

**RANCANG BANGUN APLIKASI MEDIA PEMBELAJARAN  
RUKUN HAJI BERBASIS ANDROID DENGAN  
MENGUNAKAN TEKNOLOGI MARKERLESS  
AUGMENTED REALITY**

**TUGAS AKHIR**  
Diajukan untuk Memenuhi Salah Satu Syarat Kelulusan  
Program Pendidikan Sarjana

Oleh :  
Muhammad Supian  
2016130023



JURUSAN TEKNIK INFORMATIKA  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER – LIKMI  
BANDUNG  
2020

**RANCANG BANGUN APLIKASI MEDIA PEMBELAJARAN  
RUKUN HAJI BERBASIS ANDROID DENGAN  
MENGUNAKAN TEKNOLOGI MARKERLESS  
AUGMENTED REALITY**

Oleh :  
Muhammad Supian  
2016130023

Bandung, 15 Mei 2020  
Menyetujui,

Budi Maryanto, S.Kom., M.T.  
Pembimbing

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

JURUSAN TEKNIK INFORMATIKA  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER – LIKMI  
BANDUNG  
2020

## ABSTRAK

*Augmented Reality* merupakan teknologi yang berkembang di dunia, menggabungkan benda maya dua dimensi atau tiga dimensi ke dalam sebuah lingkungan yang nyata. Dengan perkembangan media pembelajaran saat ini yang berkembang dengan pesat, sistem operasi *hardware* pun turut ikut berkembang untuk mengimbangi, sehingga memungkinkan untuk mengimplementasikan *Augmented Reality* pada perangkat *mobile*.

Dengan dasar tersebut, maka dibangunlah sebuah media pembelajaran Rukun Haji secara *Augmented Reality*. Dalam pengembangan aplikasi ini menggunakan metode *prototype* dengan metodologi pemrograman berorientasi objek dengan *game engine* yang digunakan adalah Unity 3D dengan alasan lebih mudah membuat aplikasi media pembelajaran ini secara *Augmented Reality* karena banyak didukung oleh berbagai *Software Development Kit* seperti contoh salah satunya VOID AR.

Metode yang digunakan yaitu studi kepustakaan, yakni mencari data, tutorial serta mencari aplikasi sejenis berbagai referensi. Referensi didapatkan melalui internet, buku, dan jurnal. Pemodelan dilakukan dengan cara merancang *use case*, skenario proses, *class diagram*, dan *activity diagram* serta melakukan perancangan antar muka. Lalu, melakukan pengujian terhadap program yang dibuat serta terhadap fitur-fitur yang ada di dalam program, serta melakukan pengujian instalasi di dalam *smartphone* berbasis Android.

Dalam penelitian ini, dapat disimpulkan bahwa penelitian ini menghasilkan Aplikasi Media Pembelajaran Rukun Haji Berbasis Android Dengan Teknologi *Markerless Augmented Reality* menggunakan *game engine* Unity 3D dapat dibuat dan berjalan dengan baik dan dengan bantuan VOID AR SDK yang berfungsi untuk mengaktifkan kamera yang memiliki fitur melacak suatu target. Target yang dilacak dapat dilakukan secara *markerless*, dimana kita tidak membutuhkan sebuah marker, hanya dengan sensor *gyroscope* yang terdapat di *smartphone* suatu target dapat terlacak dan muncul pada posisi tertentu.

## KATA PENGANTAR

Bismillahirrahmanirrahim, Alhamdulillah rabbi 'alamin, Allahumma sholli 'ala Sayyidina Muhammad wa 'ala alihi Sayyidina Muhammad. Puji dan rasa syukur penulis panjatkan kepada Allah SWT. Karena rahmat dan karuniaNya yang diberikan, penulis dapat menyelesaikan Tugas Akhir yang berjudul "Rancang Bangun Aplikasi Media Pembelajaran Rukun Haji Berbasis Android Dengan Teknologi *Markerless Augmented Reality*" dengan lancar dan dengan tepat waktu.

Dalam menyelesaikan Tugas Akhir ini, penulis mendapat banyak bantuan, bimbingan, dan dukungan dari berbagai pihak. Dalam kesempatan ini, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Bapak Budi Maryanto, S.Kom, M.T. selaku dosen pembimbing yang telah meluangkan waktu untuk membimbing, memberi saran, dan membantu dalam menyusun Tugas Akhir ini.
2. Seluruh dosen STMIK LIKMI yang selama ini telah mengajarkan banyak ilmu kepada penulis selama masa perkuliahan.
3. Orang tua penulis yang selalu mendukung secara moril, meterial, dan selalu memberikan semangat serta doa sehingga dapat menyelesaikan Tugas Akhir ini.
4. KH. Hafidzin dan Ustadz Bambang Triyono, S.Pd, S.Pd.I, M.Ag. yang telah memberikan dukungan motivasi, semangat, doa serta sebagai narasumber.
5. Seluruh teman-teman Angkatan 2016 yang telah memberi semangat kepada penulis.
6. Pihak-pihak lain yang tidak dapat disebutkan satu per satu yang selalu mendukung dan menyemangati dalam penyusunan Tugas Akhir.

Penulis selalu membuka diri untuk menerima setiap saran maupun kritik yang ditujukan pada isi dari Tugas Akhir ini. Sehingga dapat melakukan yang lebih baik lagi. Akhir kata penulis memohon maaf jika masih terdapat beberapa kekurangan dalam penulisan tugas akhir ini semoga dapat bermanfaat bagi pembacanya.

Bandung, 15 Mei 2020

Penulis

## DAFTAR ISI

ABSTRAK.....	i
KATA PENGANTAR .....	ii
DAFTAR ISI.....	iv
DAFTAR GAMBAR .....	vii
DAFTAR TABEL .....	viii
DAFTAR SIMBOL .....	ix
DAFTAR LAMPIRAN .....	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Kegunaan Hasil.....	2
1.5 Metodologi Penelitian.....	3
1.6 Sistematika Penulisan.....	3
BAB II LANDASAN TEORI.....	5
2.1 Rekayasa Perangkat Lunak Berorientasi Objek .....	5
2.1.1 Pemrograman Berorientasi Objek .....	6
2.1.2 Model Pengembangan Prototype .....	7
2.2 Unified Modeling Language.....	9
2.3 Perangkat Lunak Pengembangan Software.....	10
2.3.1 Unity 3D .....	10
2.3.2 VOID AR.....	11
2.3.3 Visual Studio .....	12
2.3.4 Blender.....	13
2.3.5 Android.....	14
2.4 Augmented Reality.....	16
2.4.1 Markerless .....	17
2.4.2 Simultaneous Locatization and Mapping (SLAM) .....	20
2.5 Media Pembelajaran .....	21
2.6 Ibadah Haji.....	22
BAB III ANALISIS DAN PERANCANGAN PERANGKAT LUNAK.....	27
3.1 Gambaran Umum Perangkat Lunak .....	27
3.2 Spesifikasi Kebutuhan Perangkat Lunak .....	27
3.2.1 Kebutuhan Fungsional .....	27

3.2.2	Kebutuhan non Fungsional .....	28
3.3	Use Case Diagram .....	28
3.4	Skenario Use Case .....	29
3.5	Activity Diagram .....	31
3.5.1	Activity Diagram Pembelajaran .....	31
3.5.2	Activity Diagram Display Rujukan Ibadah Haji.....	32
3.5.3	Activity Diagram Konfigurasi .....	33
3.6	Class Diagram .....	34
3.7	Rancangan Antar Muka.....	36
BAB IV IMPLEMENTASI DAN PENGUJIAN PERANGKAT LUNAK .....		40
4.1	Spesifikasi dan Implementasi Pengujian Perangkat Lunak .....	40
4.2	Tampilan Antar Muka.....	41
4.3	Pengujian Fungsi .....	45
4.4	Hasil Implementasi dan Pengujian Fungsi .....	48
BAB V KESIMPULAN DAN SARAN .....		49
5.1	Kesimpulan .....	49
5.2	Saran... ..	49
DAFTAR PUSTAKA .....		51
LAMPIRAN .....		53

## DAFTAR GAMBAR

Gambar 2.1 Paradigma pemodelan <i>prototype</i> .....	8
Gambar 2.2 6DOF ( <i>six degree of freedom</i> ) .....	19
Gambar 2.3 Pakaian Ihram .....	23
Gambar 2.4 Suasana jamaah haji melakukan wuquf dipadang arafah .....	24
Gambar 2.5 Suasana jamaah haji melakukan <i>thawaf</i> (mengelilingi ka'bah) .....	24
Gambar 2.6 Suasana jamaah haji melakukan <i>Sa'i</i> .....	25
Gambar 2.7 Suasana jamaah haji melakukan lempar jumrah.....	26
Gambar 3.1 <i>Use Case Diagram</i> .....	28
Gambar 3.2 <i>Activity Diagram</i> Pembelajaran .....	32
Gambar 3.3 <i>Activity Diagram</i> Display Rujukan Ibadah Haji.....	33
Gambar 3.4 <i>Activity Diagram</i> konfigurasi.....	34
Gambar 3.5 <i>Class Diagram</i> Media Pembelajaran .....	35
Gambar 3.6 Rancangan Antar Muka Menu Utama .....	36
Gambar 3.7 Rancangan Antar Muka Pembelajaran .....	37
Gambar 3.8 Rancangan Antar Muka Menampilkan objek 3D .....	37
Gambar 3.9 Rancangan Antar Muka Menu Display Rujukan Ibadah Haji.....	38
Gambar 3.10 Rancangan Antar Muka Menu Konfigurasi .....	39
Gambar 3.11 Rancangan Antar Muka Konfirmasi <i>Exit</i> .....	39
Gambar 4.1 Tampilan Menu Utama.....	41
Gambar 4.2 Tampilan Pembelajaran .....	42
Gambar 4.3 Tampilan Media Pembelajaran .....	42
Gambar 4.4 Tampilan Rujukan Ibadah Haji.....	43
Gambar 4.5 Tampilan Isi Rujukan Ibadah Haji .....	43
Gambar 4.6 Tampilan Konfigurasi .....	44
Gambar 4.7 Tampilan Konfirmasi <i>Exit</i> .....	44

## DAFTAR TABEL

Tabel 3.1 Skenario utama <i>Use Case</i> Pembelajaran.....	29
Tabel 3.2 Skenario alternatif <i>Use Case</i> Pembelajaran .....	29
Tabel 3.3 Skenario utama <i>Use Case</i> Display Rujukan Ibadah Haji.....	30
Tabel 3.4 Skenario utama <i>Use Case</i> Konfigurasi.....	30
Tabel 4.1 Spesifikasi Perangkat Keras Komputer .....	40
Tabel 4.2 Tabel pengujian tombol-tombol pada Menu Utama.....	45
Tabel 4.3 Tabel pengujian tombol-tombol pada Pembelajaran .....	45
Tabel 4.4 Tabel pengujian tombol-tombol pada Media Pembelajaran .....	46
Tabel 4.5 Tabel pengujian aksi pada Media Pembelajaran.....	46
Tabel 4.6 Tabel pengujian tombol-tombol pada Rujukan Ibadah Haji.....	46
Tabel 4.7 Tabel pengujian tombol-tombol pada Konfigurasi .....	47
Tabel 4.8 Tabel pengujian tombol-tombol pada Konfigurasi .....	47
Tabel 4.9 Tabel hasil pengujian terhadap lingkungan .....	47
Tabel 4.10 Tabel hasil pengujian pada <i>smartphone</i> .....	47

## DAFTAR SIMBOL

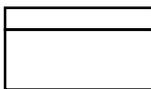
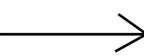
### Simbol *Use Case Diagram*

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.

### Simbol *Activity Diagram*

No	Gambar	Nama	Keterangan
1		<i>Action</i>	<i>State</i> 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 kemungkinan perbedaan transisi.

### Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2		<i>Direct Association</i>	Garis yang menghubungkan antara objek satu dengan objek lainnya.
3		<i>Association</i>	Relasi antarkelas dengan arti umum
4		<i>Generalisasi</i>	Relasi antarkelas dengan makna <i>generalisasi-spesialisasi</i> (umum khusus)

## DAFTAR LAMPIRAN

LAMPIRAN 1 CONTOH TAMPILAN PROGRAM .....	53
1. Menu Utama.....	53
2. Menu Pembelajaran .....	53
3. Thawaf AR .....	53
4. Wuquf AR.....	54
5. Isi Rujukan Thawaf.....	54
6. Isi Rujukan Wuquf.....	54
LAMPIRAN 2 LISTING PROGRAM .....	55
1. MarkerlessUI.cs.....	55
2. SimpleDrag.cs .....	55
3. SimpleRotateScale.cs .....	59
4. MarkerlessHelp.cs.....	59

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Ibadah haji ialah ibadah yang diwajibkan bagi setiap muslim, dengan syarat bagi yang mampu, mampu baik secara fisik, materi, dan keilmuan. Dan yang lebih penting adalah kemampuan untuk menyiapkan diri sebagai tamu Allah. Ibadah Haji biasanya dimulai dari bulan *Syawal* sampai sepuluh hari pertama bulan *Dzulhijjah*. Waktu tersebut sangat dinantikan bagi setiap muslim, agar kelak dapat menyempurnakan rukun islam yang kelima.

Kesempurnaan ibadah haji dapat dicapai dengan melaksanakan kewajiban-kewajiban dan persyaratannya. Syarat wajib dalam menunaikan ibadah haji dapat disebut rukun haji. Rukun haji sangat penting dalam menunaikan ibadah haji. Jika hal ini tidak sepenuhnya kita lakukan dengan baik maka ibadah haji kita kurang sempurna atau bahkan menjadi tidak sah.

Saat ini perkembangan teknologi informasi dan komunikasi terus meningkat, sehingga menciptakan inovasi-inovasi baru untuk memudahkan manusia dalam melakukan kegiatan sehari-hari. Mulai dari *smartphone* sampai laptop bukan hanya digunakan oleh kalangan muda saja, akan tetapi kalangan orang tua dan anak-anak pun mulai menggunakan peralatan canggih ini. Dengan perkembangan teknologi yang ada, *smartphone* dapat digunakan dan dimanfaatkan sebagai media pembelajaran, sehingga memudahkan pengguna untuk digunakan kapanpun dan dimanapun. Oleh karena itu, maka dibangun sebuah “**APLIKASI MEDIA PEMBELAJARAN RUKUN HAJI BERBASIS ANDROID DENGAN MENGGUNAKAN TEKNOLOGI *MARKERLESS AUGMENTED REALITY*”** yang diperuntukan untuk semua kalangan.

*Augmented reality* merupakan teknologi interaktif yang mampu memproyeksikan objek maya ke dalam objek nyata secara *real time*. *Augmented reality* sangat berpotensi sebagai sarana pembelajaran, karena di dalamnya terdapat beberapa elemen multimedia, yaitu teks, suara, animasi, dan objek 3D. Salah satu keuntungan yang dapat diperoleh dari aplikasi *augmented reality* ini, pengguna dapat mengontrol objek 3D yang berada di dalam

aplikasi. Sehingga pengguna dapat pemahaman yang lebih baik dalam pembelajaran rukun haji. Atas dasar itulah penulis berharap dengan dibuatnya aplikasi media pembelajaran rukun haji ini dapat bermanfaat dalam memahami rukun-rukun haji yang lebih interaktif dan menarik.

## 1.2 Rumusan Masalah

Rumusan masalah dalam perancangan aplikasi media pembelajaran ini adalah sebagai berikut:

1. Bagaimana merancang dan membuat aplikasi media pembelajaran rukun haji berbasis Android dengan teknologi *markerless augmented reality*?
2. Bagaimana aplikasi media pembelajaran rukun haji berbasis Android dengan teknologi *markerless augmented reality* agar nanti dapat digunakan sebagai media pembelajaran yang interaktif dan menarik bagi semua kalangan dan dapat membantu menyampaikan informasi tentang rukun haji?
3. Apa saja sumber referensi ibadah haji yang akan digunakan dalam pembuatan aplikasi?

## 1.3 Batasan Masalah

Adapun batasan masalah dalam penelitian ini :

1. Aplikasi ini dirancang dengan menggunakan *game engine Unity 3D* dan SDK VOID AR.
2. Aplikasi yang dibangun menerapkan teknologi *augmented reality* dengan metode *markerless augmented reality*.
3. Aplikasi ini hanya menggunakan sumber referensi dari Mazhab Imam Syafi'i.

## 1.4 Kegunaan Hasil

Adapun kegunaan hasil yang bisa didapat adalah sebagai media pembelajaran untuk membantu para pengguna dalam mengetahui rukun-rukun haji, pengertiannya, dan penjelasannya secara baik dan benar.

## 1.5 Metodologi Penelitian

Berikut ini adalah metodologi yang digunakan dalam penelitian tugas akhir dengan urutan pengerjaan sebagai berikut :

### 1. Studi literatur

Studi dengan referensi yang diperoleh dari berbagai sumber jurnal, buku, dan sumber lain yang tercantum dalam daftar pustaka.

### 2. Wawancara

Pengumpulan data dengan cara mengadakan tanya jawab secara langsung dengan beberapa ulama atau tokoh agama mengenai topik rukun haji.

### 3. Perancangan sistem

Mengerjakan perancangan sistem sesuai dengan skema yang telah direncanakan.

### 4. Pengujian dan Evaluasi

Melakukan pengujian sistem setelah aplikasi selesai dibuat, sehingga akan didapat data yang nantinya akan dilanjutkan pada proses evaluasi. Proses evaluasi dilakukan untuk mengetahui keberhasilan maupun kesalahan perancangan implementasi yang telah dibuat.

## 1.6 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir yang dibuat adalah sebagai berikut :

### BAB I PENDAHULUAN

Pendahuluan memuat tentang latar belakang masalah, rumusan masalah, tujuan penulisan, batasan masalah, kegunaan hasil, metodologi penelitian serta sistematika penulisan yang digunakan.

### BAB II LANDASAN TEORI

Bab ini akan menjelaskan tentang landasan teori yang digunakan untuk pembuatan aplikasi dan penulisan tugas akhir, yaitu Penjelasan *Augmented Reality*, Media Pembelajaran, Rekayasa Perangkat Lunak Berorientasi Objek, Model Pengembangan *Prototype*, UML, Unity3D, VOID AR, Blender, dan Android.

### BAB III ANALISIS DAN PERANCANGAN

Bab ini menjelaskan tentang hasil analisis dari penelitian untuk pembuatan aplikasi berupa *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan membahas perancangan *User Interface*.

### BAB IV IMPLEMENTASI DAN PENGUJIAN

Berisi mengenai implementasi dan pengujian aplikasi yang telah dibuat dari model yang telah dirancang.

### BAB V KESIMPULAN DAN SARAN

Bab ini berisikan penjelasan tentang kesimpulan dari hasil perancangan yang dibuat dan saran untuk pengembangan perancangan lebih lanjut.

## BAB II LANDASAN TEORI

### 2.1 Rekayasa Perangkat Lunak Berorientasi Objek

Pengertian rekayasa perangkat lunak menurut Sommerville dalam buku *“Software Engineering, Ninth Edition”*, Rekayasa perangkat lunak adalah:

*“Software developing is an engineering process that is focused with all views of software production from the first stages of specification system through to maintaining the system after it will use”* (Sommerville, Ian, 2011:7)

Rekayasa perangkat lunak menurut Pressman dalam buku *“Software Engineering: A Practitioner’s Approach Seventh Edition”*, memiliki definisi sebagai berikut:

*“Software engineering is the establishment and use of sound engineering principles in order reach economically software that reliable and runs efficiently on any machines.”* (Pressman, Roger S, 2010:13)

Menurut Rosa dan Shalahudin dalam buku *“Rekayasa perangkat lunak: Terstruktur dan Berorientasi Objek”*, rekayasa perangkat lunak adalah:

*“Rekayasa perangkat lunak adalah perancangan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak bernilai ekonomis yang dapat dipercaya dan bekerja secara efisien menggunakan mesin.”* (Rosa, A. S dan Shalahuddin M, 2018:4)

Maka dapat disimpulkan rekayasa perangkat lunak merupakan proses produksi atau pengembangan perangkat lunak yang dimulai dari tahap perancangan hingga tahapan implementasi dengan menggunakan pendekatan sistematis, disiplin, dan dapat diukur sehingga menghasilkan sebuah perangkat lunak yang handal, ekonomis, dan juga dapat seefisien mungkin penggunaannya.

Menurut Rosa dan Shalahuddin (Rosa, A. S dan Shalahuddin M, 2018:100) pengertian metodologi berorientasi objek ialah suatu strategi perancangan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diterapkan padanya. Metodologi ini adalah suatu cara di mana sistem perangkat lunak dibangun melalui pendekatan objek sistematis. Metodologi berorientasi objek didasarkan dengan penerapan

prinsip-prinsip manajemen kompleksitas. Metodologi ini meliputi beberapa rangkaian kegiatan analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek.

Menurut Erick Kurniawan (Kurniawan, Erick, 2011:65), pemrograman berorientasi objek memiliki kelebihan, antara lain:

1. Membuat suatu representasi teknis sedekat mungkin dengan pandangan konseptual dari dunia nyata
2. Membuat kerangka analisis dan spesifikasi yang stabil
3. Memudahkan pengembangan dan perubahan pemrograman.

Menurut Indrajani dan Martin (Indrajani dan Martin, 2007:9-12) pemrograman berorientasi objek memiliki 3 konsep, yaitu:

#### 1. *Enkapsulasi*

*Enkapsulasi* adalah suatu mekanisme untuk menyembunyikan atau melindungi suatu proses dari kemungkinan gangguan atau penyalahgunaan dari luar sistem sekaligus menyederhanakan penggunaan sistem itu sendiri.

#### 2. Pewarisan

Suatu proses dimana suatu kelas diturunkan dari kelas yang lain sehingga ia bisa mendapatkan ciri atau sifat dari kelas tersebut.

#### 3. *Polymorphism*

*Polymorphism* merupakan sebuah konsep yang dapat digunakan oleh antarmuka yang sama untuk memerintahkan objek untuk melakukan aksi atau tindakan yang akan memberikan hasil akhir yang sama, tetapi melalui proses yang berbeda.

### 2.1.1 Pemrograman Berorientasi Objek

Pengertian pemrograman berorientasi objek menurut Dan Clark dalam buku "*Beginning C# Object-Oriented Programming*", yaitu sebagai berikut:

*Object-oriented programming (OOP) is an approach to software development in which software structures are based on objects that interact with each other to complete a task. This interaction takes the form of messages that move back and forth between objects. In response to a message, an object can perform an action or method.* (Clark, Dan, 2011:1)

Konsep pemrograman berorientasi objek mulai muncul pada pertengahan 1960 dengan menggunakan bahasa pemrograman yang disebut *Simula* dan selanjutnya berkembang pada tahun 1970 dengan munculnya *Smalltalk*. Pada pertengahan 1980 bermunculan minat dalam metodologi berorientasi objek, bahasa yang digunakan pada saat itu adalah C++ dan *Eiffel*. Pemrograman berorientasi objek terus berkembang pada tahun 1990, terutama dengan munculnya *Java*. Dan pada tahun 2002, bersamaan dengan rilisnya *NET. Framework*, *Microsoft* memperkenalkan bahasa pemrograman berorientasi objek yang baru yaitu C# dan menjadikan bahasa ini terus berkembang sehingga menjadi pilihan utama pemrograman modern. (Clark, Dan, 2011:2)

Menurut Adi Nugroho (Nugroho, 2009:17) dalam terminologi pemrograman berorientasi objek terdapat konsep objek (*object*) dan kelas (*class*). Kedua konsep ini sesungguhnya mempunyai hubungan yang sangat erat. Definisi objek adalah orang, benda, tempat, kejadian, atau konsep-konsep di dunia nyata dan penting untuk suatu aplikasi (perangkat lunak dan/atau sistem informasi). Sedangkan kelas menurut Adi Nugroho (Nugroho, 2009:18) didefinisikan sebagai kumpulan/himpunan objek yang memiliki kesamaan dalam properti/atribut, perilaku, dan cara berhubungan dengan objek lain.

### 2.1.2 Model Pengembangan Prototype

Model pengembangan yang akan digunakan untuk perangkat lunak ini adalah model pengembangan *prototype*. Menurut Sommerville dalam buku "*Software Engineering, Ninth Edition*", *prototype* adalah:

*The prototype is an early version of a software system that is used to demonstrate concepts, try out design options, and find out more about problems and possible solutions. Rapid and repeatable prototype development is very important so that costs can be controlled and system stakeholders can experiment with prototypes early in the software process.* (Sommerville, Ian, 2011:45)

Menurut Setiyani (Setiyani, 2018:27-28), 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 yang digunakan dalam pengembangan perangkat lunak yang dibutuhkan oleh pengguna. Pada penelitian ini, pendekatan yang paling baik adalah menggunakan paradigma pembuatan *prototype*. Dalam hal ini paradigma pembuatan prototipe sering membantu tim pengembang dan para *stakeholder* untuk lebih memahami kebutuhan perangkat lunak yang akan dikembangkan.

Pembuatan *prototype* dimulai dengan komunikasi antara tim pengembang dan pelanggan, kemudian menetapkan sasaran pengembangan secara keseluruhan dan mengidentifikasi spesifikasi kebutuhan. Pembuatan *prototype* direncanakan dan dirancang dengan cepat. Rancangan kemudian di kontruksi untuk membuat *prototype*. *Prototype* kemudian diserahkan kepada *stakeholder* dan mereka melakukan evaluasi terhadap *prototype* tersebut, yang pada akhirnya memberikan umpan balik untuk digunakan dalam memperbaiki atau mengembangkan spesifikasi kebutuhan perangkat lunak. Pengulangan yang terjadi pada saat *prototype* diperbaiki secara tidak langsung memenuhi kebutuhan *stakeholder*, dan pada saat bersamaan memungkinkan pengembang memahami lebih dalam kebutuhan dari perangkat lunak yang dikerjakan. Semakin banyak pengembangan maka perangkat lunak berevolusi memenuhi kebutuhan pengguna. Paradigma pembuatan prototipe digambarkan sebagai berikut:



Gambar 2.1 Paradigma pemodelan prototype.  
(Setiyani, 2018:29)

Menurut Rosa dan Shalahudin (Rosa, A. S dan Shalahuddin M, 2018:33) maka dapat disimpulkan model prototipe cocok digunakan untuk menjabarkan kebutuhan pelanggan secara lebih detail karena pelanggan sering kali kesulitan menyampaikan kebutuhannya secara detail tanpa melihat gambaran yang 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 akan menjadi patokan sehingga spesifikasi persyaratan sistem masih dalam lingkup proyek.

## 2.2 Unified Modeling Language

Pengertian *Unified Modeling Language* menurut Rumbaugh dan kawan-kawan dalam buku "*The Unified Modeling Language Reference Manual Second Edition*", yaitu sebagai berikut:

*"The Unified Modeling Language (UML) is a generic purpose of illustrate modeling of any language that used to describe, analyze, develop, and report the artifacts of a software system."* (Rumbaugh, 2005:3)

Menurut Dharwiyanti (Dharwiyanti, 2003:2), *Unified Modeling Language* (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk memvisualisasikan, merancang dan mendokumentasikan sistem perangkat lunak. UML menawarkan sebuah standar dalam merancang sebuah model sistem.

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

### 1. Use Case Diagram

*Use case* diagram ialah suatu sarana dalam melakukan pengorganisasian spesifikasi kebutuhan pengguna dengan cara yang mudah untuk dikelola dan dapat dimengerti oleh para pengguna. (Nugroho, 2010:93)

### 2. Skenario Use Case Diagram

Setiap *use case* akan dilengkapi dengan skenario. Skenario *use case* ialah alur jalannya proses *use case* yang telah dibuat dari sisi aktor dan sistem. Skenario *use case* dibuat per *use case* dari yang terkecil, misalkan untuk generalisasi maka skenario

yang telah dibuat adalah *use case* yang lebih khusus. Skenario normal adalah skenario bila sistem berjalan normal tanpa terjadi kesalahan atau *error*. Sedangkan skenario alternatif adalah skenario bila sistem tidak berjalan normal, atau mengalami *error*. Skenario normal dan skenario alternatif dapat dapat dibuat lebih dari satu. Alur dari skenario inilah nantinya menjadi dasar pembuatan diagram sekuen. (Rosa, A. S dan Shalahuddin M, 2018:161)

### 3. *Activity* Diagram

*Activity* diagram ialah menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana setiap aliran dimulai dari *decision* yang mungkin terjadi, dan bagaimana mereka akan berakhir. *Activity* diagram juga dapat digambarkan dalam proses paralel yang dapat terjadi dalam beberapa eksekusi. (Dharwiyanti, 2003:7)

### 4. *Class* Diagram

*Class* diagram adalah spesifikasi yang jika diinstansiasi akan menghasilkan objek dan merupakan inti dari pengembangan dan proses desain berorientasi objek. *Class* diagram menggambarkan keadaan (*properti/atribut*) suatu sistem, sekaligus akan menawarkan layanan untuk memanipulasi keadaan tersebut. (Dharwiyanti, 2003:5)

## 2.3 Perangkat Lunak Pengembangan Software

Adapun perangkat lunak yang akan digunakan dalam pengembangan *software* adalah Unity 3D, VOID AR, Visual Studio, Blender, dan Sistem Operasi Android.

### 2.3.1 Unity 3D

Menurut Creighton dalam buku "*Unity 3D Game Development by Example: Beginner's Guide*", *unity* adalah :

*"Unity 3D is a new technology that strives to make life better and easier for game developers. Unity is a game engine or a game-making tool that allows creative people like you to make video games."* (Creighton, 2010:7)

Menurut Asfari dan kawan-kawan dalam jurnal yang berjudul *Pembuatan Aplikasi Tata Ruang Tiga Dimensi Gedung Serba Guna Menggunakan Teknologi Virtual Reality*

(Studi Kasus: Graha ITS Surabaya), (Asfari, 2012:541) mengatakan bahwa *unity engine* suatu *game engine* yang akan terus berkembang. *Engine* ini ialah salah satu *game engine* dengan lisensi *source proprietary*, namun untuk lisensi pengembangan dibagi menjadi 2, yaitu gratis (*free*) dan membayar sesuai dengan perangkat target pengembangan aplikasi.

Unity tidak membatasi publikasi aplikasi, pengguna *unity* dengan lisensi gratis dapat mempublikasikan aplikasi yang telah dibuat tanpa harus membayar biaya lisensi kepada unity. Tetapi penggunaan versi gratis akan dibatasi oleh beberapa fitur yang dikurangi atau bonus modul / prefab tertentu yang dihapus dan hanya tersedia untuk pengguna membayar.

Seperti dengan *game engine* lainnya, *unity engine* juga dapat mengolah beberapa data seperti objek 3D, tekstur, suara, dan lain-lain. Keunggulan dari *engine* ini dapat menangani grafik 2D dan 3D. Namun *engine* ini lebih berkonsentrasi membuat grafis 3D. Dari beberapa *game engine* yang bersama-sama menangani grafik 3D, *unity engine* dapat menangani lebih banyak. Beberapa diantaranya yaitu Windows, iOS, MacOS X, Wii, PS3, Xbox 360, dan Android.

### 2.3.2 VOID AR

Dalam situs [www.voidar.com](http://www.voidar.com), *VOID AR* adalah sebuah SDK (*Software Development Kit*) yang dikembangkan oleh Chengdu Miyo Network Technology Co., Ltd, sebuah perusahaan berbasis Tiongkok yang bergerak di bidang teknologi. *VOID AR* dikembangkan untuk membantu para peneliti untuk merancang perangkat lunak yang menggunakan teknologi *Augmented Reality*. Untuk mengembangkan perangkat lunak *markerless augmented reality*, peneliti dapat menggunakan fitur *VOID SLAM SDK* secara gratis dan dapat dikembangkan dengan aplikasi *Unity*. Beberapa fitur yang terdapat dalam *VOID AR*, yaitu:

#### 1. *Void Cloud*

Pusat komputasi awan *VOID AR* dapat mengenali dan melacak target dalam sepuluh juta gambar dalam 1 detik.

## 2. *Image Recognition*

Fitur ini dapat mengenali gambar yang sangat terperinci seperti kartu, uang kertas, majalah, dan lain-lain.

## 3. *Multi Targer Recognition*

Fitur ini dapat mengidentifikasi dan melacak multi target dalam adegan dengan stabilitas tinggi.

## 4. *User defined images*

Pengguna dapat menentukan marker sendiri, bukan hanya pada majalah, poster, dan lain-lain.

## 5. *Entended tracking*

Di lingkungan yang penuh dengan fitur, bahkan jika targetnya tidak terlihat oleh kamera, efek AR tetap bertahan.

## 6. *Dynamic loading*

Memuat dan mengenali target secara dinamis berdasarkan kebutuhan pengguna. Agar aplikasi menjadi lebih ringan dan fleksibel.

## 7. *Video Playback*

Memutar video langsung pada permukaan target.

## 8. *Online Video Play*

Video dapat diputar secara *online*. Data video dapat diperoleh secara *online* dengan disematkan dan dikemas ke *client server*.

## 9. *AR Recording*

Satu rekaman untuk menjaga momen *augmented reality* anda.

## 10. *Development Platform*

Mudah digunakan dan manajemen sumber daya yang efektif.

### 2.3.3 Visual Studio

Pengertian Visual Studio menurut Joe Mayo dalam buku "*Microsoft Visual Studio 2010: A Beginner's Guide*", yaitu sebagai berikut:

*Visual Studio is an integrated development environment (IDE), a set of tools in one application that helps you write programs. Without Visual Studio, you need to open a text*

*editor, write all the code, and then run the command line compiler to create an executable application. The problem with text editors and command line compilers is that you will lose a lot of productivity through manual processes.* (Mayo, 2010:4)

Menurut Simanjuntak dan kawan-kawan dalam jurnal yang berjudul *Perancangan Antarmuka Pendulum Terbalik Menggunakan Visual Studio*, mengatakan bahwa Microsoft Visual Studio merupakan sebuah perangkat lunak yang lengkap (*suite*) dan dapat digunakan untuk mengembangkan aplikasi, baik aplikasi personal, aplikasi bisnis, ataupun komponen aplikasinya dalam bentuk aplikasi *console*, aplikasi Web, ataupun aplikasi *windows*. Microsoft Visual Studio juga dapat digunakan untuk mengembangkan aplikasi dalam *native code* (bentuk bahasa mesin yang berjalan pada *windows*) atau *managed code* (bentuk *Microsoft Intermediate Language* di atas .NET Framework). (Simanjuntak, 2019:2819)

#### **2.3.4 Blender**

Menurut situs [www.blender.org](http://www.blender.org), blender adalah aplikasi *open source* dan gratis yang bisa digunakan untuk membuat pemodelan 3D. Aplikasi blender mempunyai beberapa kelebihan, yaitu sebagai berikut:

1. *Open source* dan *free*

Blender ialah salah satu 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, seperti 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* dan *motion tracking*.

### 2.3.5 Android

Menurut Anna Arthdi putra (Putra, 2012:6), dalam buku "*Android dan Anak Tukang Sayur: Buku Praktis Belajar Pemrograman Android*", Android merupakan sistem operasi untuk telepon seluler yang berbasis Linux. Android juga menyediakan platform terbuka bagi pengembang untuk membuat aplikasi mereka sendiri untuk digunakan oleh berbagai perangkat seluler. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat perangkat lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, T-Mobile, Qualcomm, dan Nvidia. Di dunia ini ada dua jenis distributor sistem operasi Android. Yang pertama adalah untuk mendapatkan dukungan penuh dari Google atau Google Mail Services (GMS) dan yang kedua adalah sepenuhnya bebas dari distribusi tanpa dukungan langsung dari Google atau dikenal sebagai Open Handset Distribution (OHD).

Menurut Nazruddin (Nazruddin, 2012:3) dalam buku "*Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android: Edisi Revisi*", Android adalah generasi baru *platform mobile*, *platform* yang dapat memberikan pengembang pengembangan yang mereka harapkan. Sistem operasi yang mendasari Android dilisensikan di bawah GNU, *General Public Lisensi* Versi z (GPLvz), yang lebih sering dikenal dengan istilah "*copyleft*" lisensi di mana setiap peningkatan pihak ketiga harus terus jatuh di bawah *terms*. Android didistribusikan di bawah *Lisensi Apache Software* (ASL/Apache2), yang memungkinkan untuk distribusi kedua dan seterusnya.

Aplikasi Android dapat dikembangkan pada sistem operasi berikut: (Nazruddin, 2012:4)

1. Windows XP Vista/Seven.
2. Mac OS X (Mac OS X ro.4.8) atau lebih baru.
3. Linux.

Arsitektur Android terdiri dari bagian-bagian seperti berikut (Nazruddin, 2012:6-8):

1. *Applications* dan *Widgets* ialah *layer* di mana kita berhubungan dengan aplikasi saja, di mana biasanya kita *download* aplikasi kemudian kita lakukan instalasi dan jalankan

aplikasi tersebut. Di *layer* terdapat aplikasi inti termasuk *clien email*, kalender, peta, program SMS, kontak, *browser*, dan lain-lain. Semua aplikasi akan ditulis menggunakan bahasa pemrograman Java.

2. *Applications Frameworks*, Android merupakan "*Open Development Platform*" yaitu Android akan menawarkan kepada pengembang atau memberikan pengembang kemampuan untuk membuat aplikasi yang baik dan inovatif. Pengembang akan bebas mengakses perangkat keras, mengakses informasi *resources*, menjalankan *seruice background*, mengatur *alarm*, dan menambahkan status *notifications*, dan sebagainya. Pengembang memiliki akses penuh menuju *API framework* seperti yang dilakukan oleh aplikasi kategori inti. Arsitektur aplikasi dirancang agar kita dapat dengan mudah menggunakan kembali komponen yang telah digunakan (*reuse*).

Sehingga kita dapat menyimpulkan bahwa *Applications Frameworks* ini adalah *layer* di mana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada *layer* inilah aplikasi dapat dirancang dan dibuat, seperti *content providers* yang berupa sms dan panggilan telepon. Ada beberapa komponen yang termasuk di dalam *Applications Frameworks*, yaitu sebagai berikut:

- a. *Views*
  - b. *Content Prouider*
  - c. *Resource Manager*
  - d. *Notification Manager*
  - e. *Activity Manager*
3. *Libraries* merupakan *layer* di mana fitur-fitur Android berada, biasanya para pengembang aplikasi mengakses *libraries* untuk menjalankan aplikasi mereka. Berjalan di atas *kernel*, *layer* ini meliputi berbagai *library* C/C++ inti seperti Libc dan SSL, serta:
    - a. *Libraries* media untuk pemutaran media audio dan video
    - b. *Libraries* SQLite untuk dukungan *database*
    - c. *Libraries Graphics* mencakup SGL dan OpenGL untuk grafis 2D dan 3D
    - d. *Libraries* untuk manajemen tampilan

- e. *Libraries* SSL dan WebKit terintegrasi dengan *web browser* dan *security*
  - f. *Libraries* 3D yang mencakup implementasi OpenGL ES 1.0 API's
  - g. *Libraries LiveWebcore* mencakup modern *web browser* dengan *engine embeded web view*
4. Android *Run Time, Layer* yang membuat aplikasi Android dapat dijalankan di mana dalam prosesnya menggunakan *Implementasi Linux. Dalvik Virtual Machine (DVM)* adalah mesin yang membentuk dasar kerangka kerja aplikasi Android. Di dalam Android *Run Time* dibagi menjadi dua bagian, yaitu sebagai berikut:
- a. *Core Libraries*. Aplikasi Android yang dibangun dalam bahasa java, sementara *Dalvik* sebagai virtual mesinnya bukan *Virtual Machine Java*, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menerjemahkan bahasa java/c yang akan ditangani oleh *Core Libraries*.
  - b. *Dalvik Virtual Machine*. *Virtual* mesin berbasis *register* yang dioptimalkan untuk menjalankan fungsi secara efisien, di mana proses pengembangannya mampu membuat *linux kernel* untuk melakukan *threading* dan manajemen tingkat rendah.
5. *Linux Kernel* adalah *layer* di mana inti dari operasi sistem dari Android itu berada. Berisi file sistem yang mengatur sistem *processing, resource, memory, drivers*, dan sistem-sistem lainnya. *Linux kernel* yang digunakan Android adalah *linux kernel release 2.6*.

## 2.4 Augmented Reality

Pengertian *augmented reality* menurut Borko Furht dalam buku "*Handbook of Augmented Reality*", yaitu sebagai berikut:

*We Determine Augmented Reality (AR) as a direct or indirect view of the real-time physical environment of the real world that has been enhanced / added by adding information generated by virtual computers into it. AR is interactive and registered in 3D and combines real and virtual objects. (Furht, 2011:3)*

*Augmented reality* menurut Gregory Kipper dan Joseph Rampolla (Kipper dan Rampolla, 2013:1) yaitu:

*Augmented Reality (AR) is a variation of the Virtual Environment (VE), or Virtual Reality (VR) because it is more commonly referred to. Virtual Reality technology really immerses the user in a synthetic environment and while immersed, the user cannot see the real world around him. Instead, Augmented Reality takes information that is generated digitally or*

*computationally, whether it is in the form of images, audio, video, touch, or haptic sensation and put it in a real-time environment. Augmented Reality can be technically used to enhance the five senses, but its use today is the most common visual. Unlike Virtual Reality, Augmented Reality allows users to see the real world, with virtual objects superimposed or combined with the real world.*

Menurut Ronald T. Azuma (Azuma, 1997:355-356), *Augmented Reality* merupakan teknologi yang dapat menggambarkan dan menggabungkan dunia nyata dan dunia virtual yang diciptakan melalui komputer sehingga batas antara keduanya menjadi sangat tipis. *Augmented Reality* (AR) adalah variasi dari *Virtual Environment* (VE) atau yang sering disebut *Virtual Reality* (VR). *Augmented reality* memiliki karakteristik sebagai berikut:

1. Menggabungkan lingkungan nyata dan virtual.
2. Berjalan secara interaktif dalam waktu nyata.
3. Integrasi dalam 3D.

Berdasarkan pendapat dari beberapa ahli maka dapat disimpulkan, bahwa *augmented reality* adalah sebuah teknologi yang mampu menciptakan lingkungan baru dengan menggabungkan lingkungan nyata dan virtual sehingga pengguna akan merasa bahwa lingkungan yang telah dibuat itu nyata. Dengan kata lain, pengguna tidak akan merasakan perbedaan antara *augmented reality* dan apa yang mereka lihat di lingkungan nyata. Dengan bantuan *augmented reality*, lingkungan nyata disekitar kita akan dapat berinteraksi dalam bentuk virtual/digital. Informasi tentang objek dan lingkungan sekitarnya dapat dimasukkan ke dalam sistem *augmented reality* yang kemudian informasi tersebut akan ditampilkan pada layar dunia nyata seolah-olah informasi itu nyata. Informasi yang ditampilkan oleh objek virtual akan membantu pengguna melakukan aktivitas di dunia nyata.

#### **2.4.1 Markerless**

Menurut Borko Furht (Furht, 2011:256-257) pelacakan berbasis *computer vision* dibagi menjadi dua, yaitu *marker based* dan *markerless*. Dengan menggunakan metode *markerless* pengguna tidak perlu lagi menggunakan sebuah *marker* dalam menampilkan elemen-elemen virtual/digital. Metode *markerless* mempunyai beberapa fitur pendeteksi, yaitu :

1. *Fast computational time.*

Memungkinkan untuk menghitung besaran waktu nyata untuk menampilkan gambar yang dapat diterima.

2. *Robustness with respect to changing lighting conditions and image blurring.*

Pencahayaan tidak boleh kurang yang akan berakibat gambar menjadi buram.

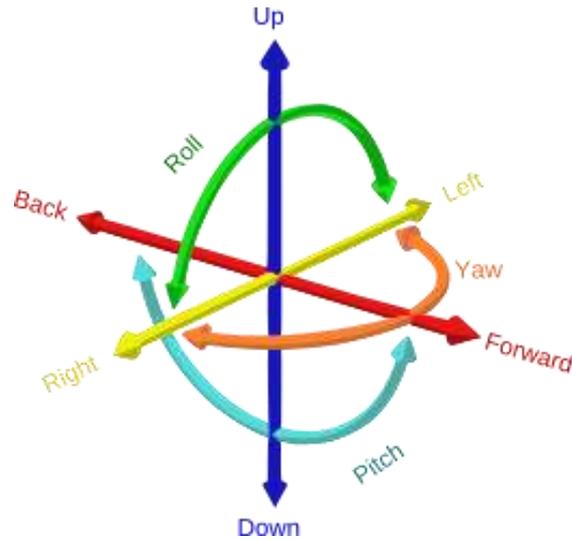
3. *Robustness against observation from different viewing angles.*

Dalam lingkungan *augmented reality*, pengguna biasanya tidak terlalu dibatasi oleh posisi, orientasi, dan sudut pandang.

4. *Scale invariance*

Dalam *augmented reality* objek yang menyediakan titik fitur yang diperlukan sering diamati jarak yang berbeda. Pelacakan harus bekerja dalam jangkauan luas dan tidak terbatas

Menurut Jon Peddie (Peddie, 2017:262-263) masalah utama dalam *augmented reality* adalah pelacakan gerakan dan menentukan lokasi. Dalam hal ini ada banyak sensor yang bisa digunakan, antara lain *mechanical devices, ultrasonic devices, magnetic sensors, inertial devices, GPS, compass, and obviously, optical sensors*. *Augmented reality* didasarkan pada perhitungan posisi kamera secara akurat dalam ruang 3D, juga dikenal dengan sebutan *pose* atau 6DOF (*degrees of freedom*). Pose atau 6DOF mengacu pada posisi dan orientasi kamera untuk menentukan dimana dan arah apa dalam ruang 3D yaitu maju/mundur (*surge*), naik/turun (*heave*), dan kiri/kanan (*sway*), lalu dikombinasikan dengan perubahan orientasi melalui rotasi sekitar tiga sumbu tegak lurus yang sering disebut *pitch, yaw, dan roll*.



Gambar 2.2 6DOF (six degree of freedom)  
(Peddie,2017:263)

Menurut Rahman dan kawan-kawan (Rahman, 2014:66) dalam jurnal yang berjudul *Rancang Bangun Aplikasi Informasi Universitas Bengkulu Sebagai Panduan Pengenalan Kampus Menggunakan Metode Markerless Augmented Reality Berbasis Android* dikutip oleh Mario dalam buku *Membuat Aplikasi Android Augmented Reality Menggunakan Vuforia SDK dan Unity*, mengatakan salah satu metode *augmented reality* yang saat ini sedang berkembang adalah metode *Markerless augmented reality*, dengan metode ini pengguna tidak perlu lagi mencetak sebuah marker untuk menampilkan elemen digital. Dalam hal ini, marker yang akan dikenali berbentuk posisi, lokasi, maupun arah perangkat. Total Immersion dan Qualcomm adalah salah satu perusahaan yang mengembangkan *Augmented Reality* dengan berbagai macam-macam teknik *Markerless*, antara lain:

#### 1. *Face Tracking*

Dengan menggunakan algoritma ini, komputer akan dapat mengenali wajah manusia secara umum dengan cara mengenali posisi mata, hidung, dan mulut manusia, kemudian akan mengabaikan benda-benda lain di sekitarnya seperti pohon, rumah, gedung, dan lain sebagainya.

## 2. 3D Object Tracking

Tidak seperti algoritma *face tracking* yang hanya mengenali wajah manusia secara umum, teknik 3D *object tracking* ini dapat mengenali semua bentuk objek yang ada disekitar pengguna, seperti mobil, meja, televisi, dan benda-benda lainnya.

## 3. Motion Tracking

Dalam teknik algoritma ini komputer dapat menangkap gerakan. Teknik ini sudah mulai digunakan secara luas untuk memproduksi film-film yang mencoba mensimulasikan gerakan.

## 4. GPS Based Tracking

Pengembangan teknik ini lebih diarahkan pada perangkat *smartphone*, karena teknologi GPS dan kompas yang tertanam di dalam *smartphone*. Dengan memanfaatkan fitur GPS yang berfungsi sebagai penentu lokasi pengguna pada saat itu sehingga lokasi terdekat yang ingin dituju dapat dilihat melalui penerapan *augmented reality*.

### 2.4.2 Simultaneous Localization and Mapping (SLAM)

Menurut Jon Peddie (Peddie, 2017:265) komponen penting untuk *augmented reality* adalah mengetahui dimana anda berada dan apa yang ada disekitar. Salah satu teknologi yang menggunakan kapabilitas tersebut adalah *Simultaneous Localization and Mapping* (SLAM), yaitu suatu sistem dan proses dimana perangkat membuat peta lingkungannya dan mengarahkan dirinya sendiri kedalam peta secara *real time*.

SLAM dimulai dengan lingkungan yang tidak dikenal dimana perangkat *augmented reality* mencoba membuat peta dan melokalkan dirinya didalam peta. Melalui serangkaian komputasi yang kompleks dan algoritma yang menggunakan sensor IMU untuk membangun peta dari lingkungan yang tidak diketahui saat menggunakannya pada saat yang sama untuk mengidentifikasi dimana ia berada.

Menurut Borko Furht (Furht, 2011:269-270) pada teknik SLAM mereka membuat dan memperluas peta fitur mereka sendiri selama proses pelacakan *ad-hoc*. Dengan demikian algoritma pelacakan berbasis SLAM dimulai tanpa pengetahuan sebelumnya

tentang lingkungan saat ini, selama proses pelacakan mereka mengumpulkan semua informasi yang diperlukan seperti posisi titik fitur 3D. Biasanya inialisasi hanya membutuhkan beberapa detik untuk membuat peta fitur awal yang sangat kasar.

## 2.5 Media Pembelajaran

Menurut Wibawanto (Wibawanto, Wandah, 2017:5) dalam buku "*Desain dan Pemrograman Multimedia Pembelajaran Interaktif*" mengutip dalam buku *Media Pembelajaran* oleh Azhar Arsyad, mengatakan bahwa kata media berasal dari bahasa latin *medius* yang secara harfiah berarti "tengah", "perantara", atau "pengantar". Dalam bahasa Arab, media ialah pengantar atau perantara pesan dari pengirim kepada penerima pesan.

Media pembelajaran memiliki peranan yang sangat besar dan juga berpengaruh terhadap pencapaian tujuan pendidikan yang diinginkan. Beberapa penggunaan Media/alat pembelajaran dalam proses belajar mengajar diantaranya (Wibawanto, Wandah, 2017:6-7):

1. Memperjelas penyajian pesan agar tidak terlalu verbalitas, (dalam bentuk kata-kata tertulis atau hanya dalam bentuk kata-kata yang diucapkan).
2. Mengatasi keterbatasan ruang, waktu dan indera, misalnya; benda yang terlalu besar, bisa digantikan dengan realita, film bingkai, film, gambar, atau model. Objek yang kecil, dapat dibantu dengan proyektor mikro, film bingkai, film, atau gambar. Gerakan yang terlalu lambat atau terlalu cepat, dapat dibantu oleh *timelapse* atau *high-speed photography*. Peristiwa atau kejadian yang terjadi dimasa lalu dapat ditampilkan lagi lewat rekaman video, film, film bingkai, atau foto objek yang terlalu kompleks, dapat disajikan dengan model, diagram atau melalui program animasi komputer. Konsep yang terlalu luas (gempa bumi, gunung berapi, planet, iklim dan lain sebagainya) dapat divisualisasikan dalam bentuk gambar, film dan lain sebagainya.
3. Dengan menggunakan media pendidikan yang tepat dan bervariasi, kita dapat mengatasi sikap pasif anak didik. Dalam hal ini media pembelajaran berguna untuk membangkitkan motivasi belajar, memungkinkan interaksi langsung antara anak didik

dan lingkungan dengan cara yang nyata, memungkinkan peserta didik belajar secara mandiri sesuai dengan kemampuan dan minat mereka.

4. Dengan berbagai latar belakang dan pengalaman yang berbeda antara peserta didik, sementara kurikulum dan materi pelajaran yang di tentukan sama untuk semua peserta didik dapat diatasi dengan media pendidikan yaitu: memberikan perangsang yang sama, mempersamakan pengalaman, serta menimbulkan persepsi yang sama.

## 2.6 Ibadah Haji

Pengertian haji menurut buku "*Tuntunan Manasik Haji Dan Umroh*" yang dibuat oleh Kementerian Agama RI, Haji adalah berkunjung ke Baitullah/Ka'bah untuk melakukan beberapa amalan-amalan antara lain: *wukuf, mabit, thawaf, sa'i*, dan lain sebagainya pada waktu tertentu, untuk memenuhi panggilan Allah Swt serta dengan mengharapkan ridha-Nya. (Kementerian Agama RI, 2016:66).

Menurut Syekh Syamsuddin Abu Abdillah (Abdillah, 2010:147-148) kata Haji menurut arti (bahasa) ialah: sengaja (menuju). Menuju istilah syara': Haji adalah sengaja mengunjungi Baitul Haram/Tanah Suci untuk menunaikan ibadah.

Syarat wajib Haji ada 7 perkara, yaitu:

1. Beragama islam
2. Cukup umur (baligh)
3. Sehat akal nya, dan
4. Merdeka

Penyimpangan dari syarat-syarat tersebut (tidak terpenuhi), maka tidak wajib haji.

5. Adanya bekal dan tempat yang diperlukan, (tapi) ada juga yang tidak memerlukan syarat ini, misalnya seorang warga atau penduduk yang tempatnya di Mekkah. Dan diisyaratkan pula adanya persediaan air yang cukup (bekal air) dalam bejana yang biasa digunakan dengan harga yang memadai (umum).
6. Adanya kendaraan yang layak (baik memperolehnya dengan cara) membeli atau menyewa. Hal ini berlaku bagi orang yang jauh dari Mekkah, 2 farsakh, atau bahkan melebihi ukuran tersebut baik yang dapat berjalan kaki maupun tidak. Kalau hanya

berjarak kurang 2 farsakh, dan orang tersebut menempuh dengan jalan kaki, maka ia wajib menunaikan haji walaupun tanpa kendaraan. Diisyaratkan bekal tersebut tersedia setelah melunasi hutang-piutang dan persediaan yang menjadi tanggungan selama dan kembalinya.

7. Sepinya jalan atau perjalanan aman dari hal-hal yang tidak diinginkan. Seandainya tidak terciptanya rasa aman dari segala macam gangguan, maka seseorang tidak wajib haji.

Rukun haji ada 4, sebagai berikut (Abdillah, 2010:148-149):

1. *Ihram* dengan berniat memasuki (menjalankan-ibadah) haji.

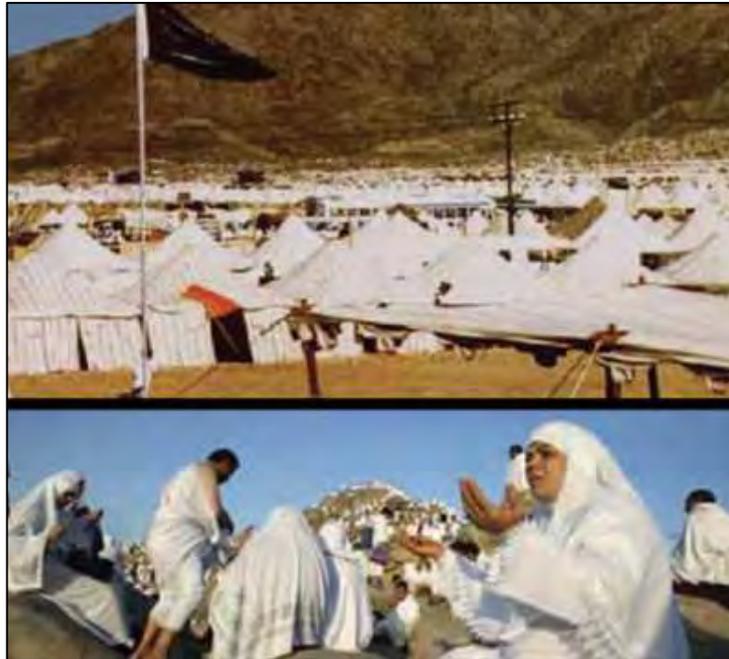


Gambar 2.3 Pakaian Ihram  
(Kementerian Agama RI, 2016:77-80)

Keterangan : 01 Kain *ihram* untuk laki-laki hanya pada waktu thawaf  
02 Kain *ihram* untuk laki-laki di luar waktu thawaf  
03 & 04 Kain *ihram* untuk perempuan

2. *Wuquf* di padang Arafah, maksudnya (orang yang ber-*Ihram* haji itu) datang di Arafah hampir tergelincirnya matahari, (sebentar lagi masuk waktu *dzuhur*) pada hari ke 9

*Dzulhijjah* dengan syarat pelakunya (yang ber*wuquf*) itu adalah ahli ibadah, tidak gila dan tidak menderita penyakit *ayan*.



Gambar 2.4 Suasana jamaah haji melakukan wuquf dipadang arafah.  
(Kementerian Agama RI, 2016:107)

3. *Thawaf* 7 kali (mengelilingi Ka'bah) Baitullah, dimulai dari Hajar Aswad (arah yang lurus dengannya) melewatinya dengan seluruh tubuhnya. Sebab jika tidak dimulai dari arah Hajar Aswad, *thawafnya* tidak ada artinya.



Gambar 2.5 Suasana jamaah haji melakukan *thawaf* (mengelilingi ka'bah)  
(Kementerian Agama RI, 2016:88)

4. *Sa'i* (yakni lari-lari kecil) dari bukit Shafa ke Marwah sebanyak 7 kali. Adapun syaratnya pertama memulai dari bukit Shafa dan berakhir (tiba) di Marwah. Dan 1 kali mulai dari Shafa menuju Marwah terhitung 1 kali *Sa'i*. demikian kembalinya dari Marwah ke Shafa terhitung 1 kali. Shafa (dibaca pendek) artinya ujung bukit (yang namanya) *Qubaisy*, sedangkan Marwah (huruf *Mim fathah*) adalah sebuah tempat yang namanya terkenal di Makkah.



Gambar 2.6 Suasana jamaah haji melakukan *Sa'i*  
(Kementerian Agama RI, 2016:97)

Menurut Syekh Syamsuddin Abu Abdillah (Abdillah, 2010:150-151), wajib-wajib dalam menunaikan ibadah haji, namun ini bukan termasuk rukun haji, yaitu:

- 1 *Ihram* dari *miqat* (batas) yang benar sesuai masa dan tempat dari mana jamaah berasal atau dari sudut mana mereka memasuki tanah suci.
- 2 Melempar jumrah yang tiga, dimulai pada Jumrah Kubra atau Jumrah Ula, Jumrah Wustha, dan Jumrah Aqabah. Dari ketiga jumrah, jumlah masing-masing melontarkan atau melemparkan dengan 7 batu kecil (kerikil) satu persatu.



Gambar 2.7 Suasana jamaah haji melakukan lempar jumrah  
(Kementerian Agama RI, 2016:119)

Mencukur atau memotong rambut, pria diutamakan mencukurnya, sedang wanita cukup memotong sebagian rambut saja. Mencukur rambut paling sedikit 3 helai rambut kepala, baik dengan menggunakan gunting atau mencabutnya. Bagi yang gundul kepalanya maka sunnah baginya memperagakannya (melewatkan alat cukur diatas kepalanya).

## **BAB III**

### **ANALISIS DAN PERANCANGAN PERANGKAT LUNAK**

#### **3.1 Gambaran Umum Perangkat Lunak**

Perangkat lunak Media Pembelajaran Rukun Haji adalah perangkat lunak yang menjelaskan mengenai pembelajaran pengenalan rukun-rukun ibadah haji dengan media bantu *markerless augmented reality*, yang akan menampilkan objek-objek rukun haji sehingga membawa pengguna seakan-akan berada ditempat pelaksanaan ibadah haji. Setiap rukun ibadah haji akan menampilkan objek-objek 3D, animasi, dan mengeluarkan suara bacaan-bacaan yang akan diucapkan oleh pelaku ibadah haji. Pengguna juga dapat mengontrol objek-objek 3D, dengan memperbesar atau memperkecil dan pengguna dapat melihat objek dari sudut mana pun, sehingga pengguna dapat melihat dengan detail objek-objek yang ada dirukun ibadah haji.

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 pembelajaran ini memakai layar landscape. Sehingga kualitas gambar dan resolusi kamera perangkat *mobile* akan memengaruhi tampilan di layar perangkat *mobile*.

#### **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**

Adapun kebutuhan yang akan dibangun harus memiliki fitur seperti :

1. Menampilkan peragaan semua rukun ibadah haji berupa objek 3D.
2. Mengeluarkan suara bacaan/do'a yang harus diucapkan oleh pelaku ibadah haji.
3. Membawa pengguna seakan-akan berada di tempat pelaksanaan ibadah haji.

### 3.2.2 Kebutuhan non Fungsional

Kebutuhan non fungsional yang dibutuhkan adalah sebagai berikut :

#### 1. Perangkat lunak

Perangkat lunak yang dibutuhkan untuk menjalankan media pembelajaran ini adalah :

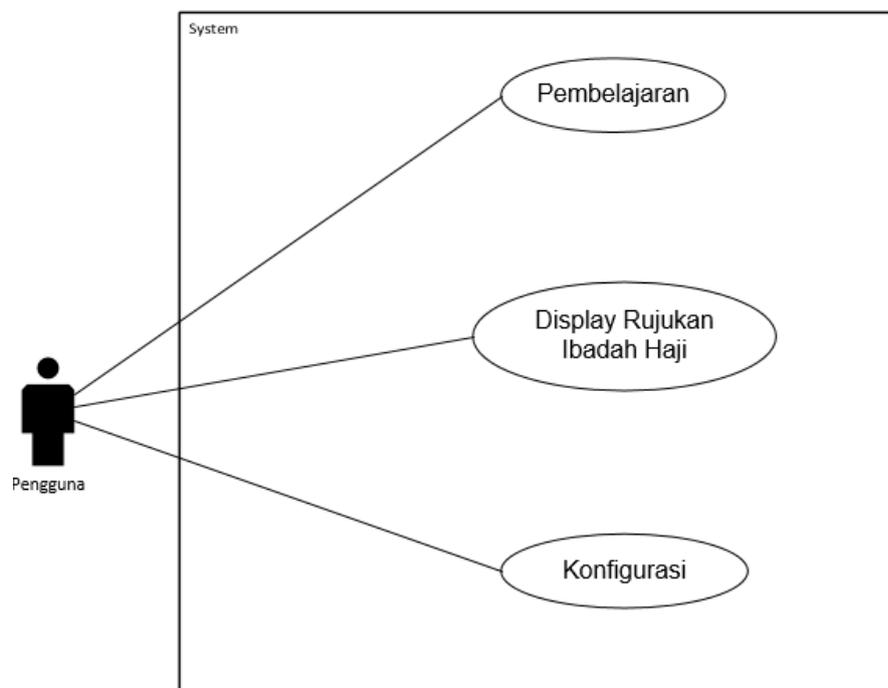
- a. Android sebagai sistem operasi
- b. Minimal sistem operasi *Lollipop* (5.0)

#### 2. Perangkat keras

Perangkat keras yang dibutuhkan untuk menjalankan media pembelajaran ini adalah :

- a. RAM 3000 MB
- b. Minimal storage *smartphone* adalah 150 MB

### 3.3 Use Case Diagram



Gambar 3.1  
Use Case Diagram

Diagram pada gambar 3.1 menggambarkan interaksi antara pengguna atau aktor dengan sistem. Penjelasan masing-masing case sebagai berikut:

### 1. Pembelajaran

Menampilkan pembelajaran rukun-rukun haji dengan mendeteksi objek-objek 3D yang telah dibuat.

### 2. Display Rujukan Ibadah Haji

Menampilkan rujukan/referensi ibadah haji, berupa dalil-dalil Al-Qur'an dan Hadits.

### 3. Konfigurasi

Pengaturan konfigurasi suara (*ON/OFF*) dan pengaturan volume suara dari perangkat yang sedang dijalankan.

## 3.4 Skenario Use Case

Dalam subbab ini akan menjelaskan mengenai skenario utama dan skenario alternatif *use case* diagram.

### 1. Skenario *Use Case* Pembelajaran

Nama *Use Case* : Pembelajaran

Aktor : Pengguna

Kondisi awal : Aplikasi telah diaktifkan

Deskripsi : Proses Pembelajaran rukun-rukun haji

Tabel 3.1  
Skenario utama *Use Case* Pembelajaran

Pengguna/Aktor	Sistem
1. Tekan tombol pembelajaran	
	2. Masuk halaman rukun haji
3. Memilih rukun haji	
	4. Mengaktifkan kamera
	5. Object Tracking
	6. Load Data
	7. Objek Ditemukan
	8. Menampilkan objek 3D

Tabel 3.2  
Skenario alternatif *Use Case* Pembelajaran

Pengguna/Aktor	Sistem
1-6 sama seperti skenario	
	7. Objek tidak terdeteksi
	8. Tampilkan pesan kesalahan

2. Skenario *Use Case* Display Rujukan Ibadah Haji

Nama *Use Case* : Display Rujukan Ibadah Haji

Aktor : Pengguna

Kondisi awal : Aplikasi telah diaktifkan

Deskripsi : Proses menampilkan rujukan ibadah haji berupa qur'an dan hadits

Tabel 3.3  
Skenario utama *Use Case* Display Rujukan Ibadah Haji

Pengguna/Aktor	Sistem
1. Tekan tombol Display Rujukan Ibadah Haji	
	2. Menampilkan beberapa tombol
3. Memilih salah satu tombol	
	4. Menampilkan rujukan tentang ibadah haji atau rukun haji yang telah dipilih

3. Skenario *Use Case* Konfigurasi

Nama *Use Case* : Konfigurasi

Aktor : Pengguna

Kondisi awal : Sistem perangkat telah dijalankan

Deskripsi : Proses mengatur suara

Tabel 3.4  
Skenario utama *Use Case* Konfigurasi

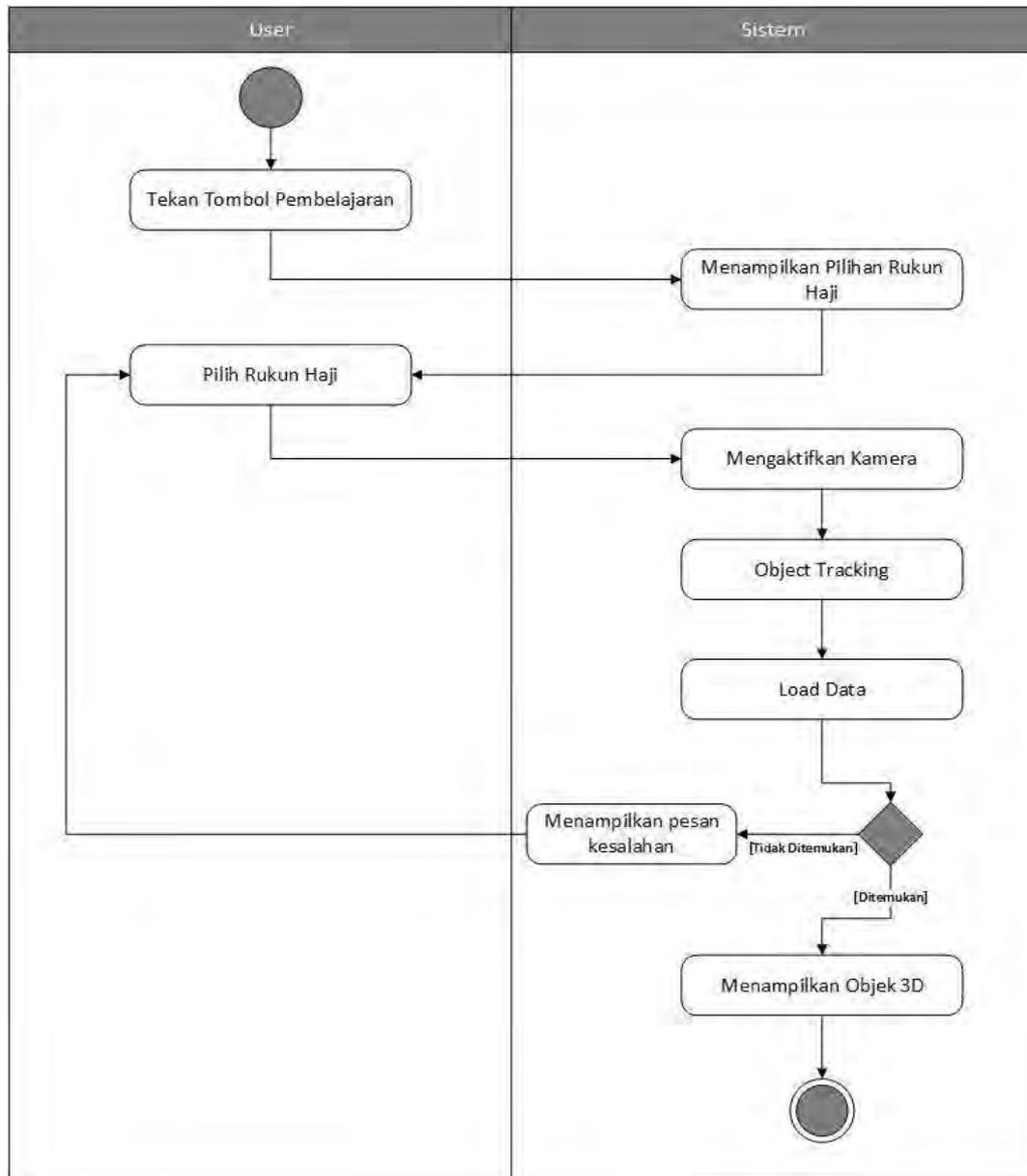
Pengguna/Aktor	Sistem
1. Tekan tombol Konfigurasi	
	2. Menampilkan konfigurasi suara
3. Menekan tombol <i>ON/OFF</i> pada konfigurasi	
	4. Sistem menjalankan sesuai perintah
5. Pengguna mengatur volume suara	
	6. Sistem menjalankan sesuai perintah

### 3.5 Activity Diagram

*Activity diagram* adalah diagram yang menggambarkan aktivitas yang terlibat dalam suatu sistem. *Activity diagram* pada perangkat lunak ini digambarkan sebagai berikut:

#### 3.5.1 Activity Diagram Pembelajaran

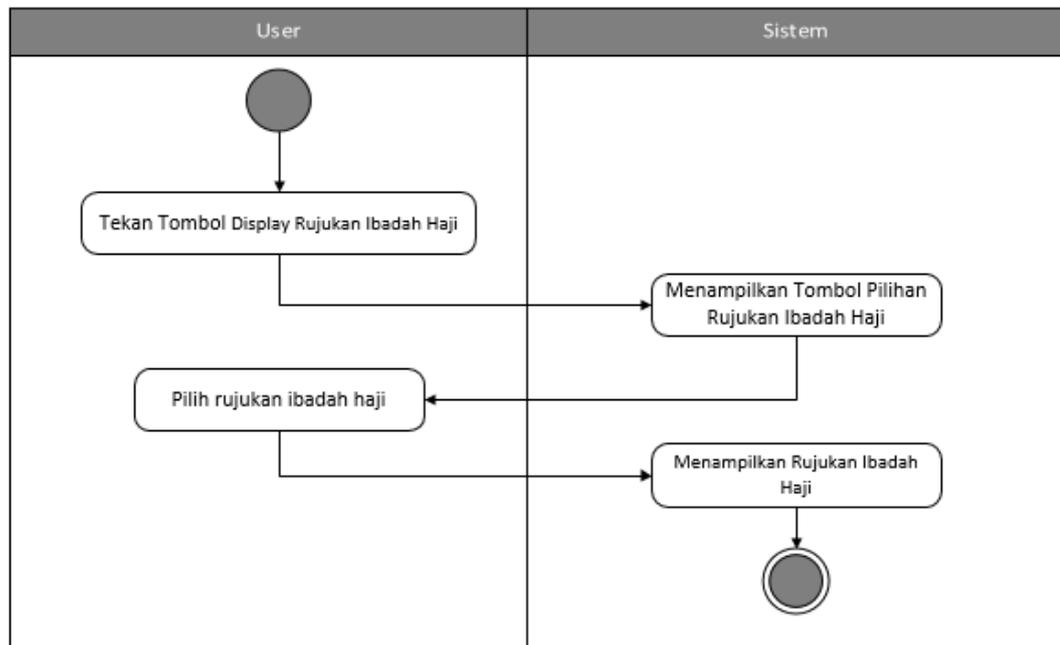
Gambar 3.2 merupakan aktivitas pada perancangan Media Pembelajaran Rukun Haji, dimana pengguna menekan tombol Pembelajaran maka sistem akan berpindah dengan menampilkan objek-objek atau rukun-rukun haji, untuk menampilkan objek 3D, pengguna terlebih dahulu memilih salah satu rukun haji, maka sistem akan mengaktifkan kamera, *object tracking*, *load data*, setelah objek ditemukan akan menampilkan objek 3D rukun haji yang telah dipilih.



Gambar 3.2  
Activity Diagram Pembelajaran

### 3.5.2 Activity Diagram Display Rujukan Ibadah Haji

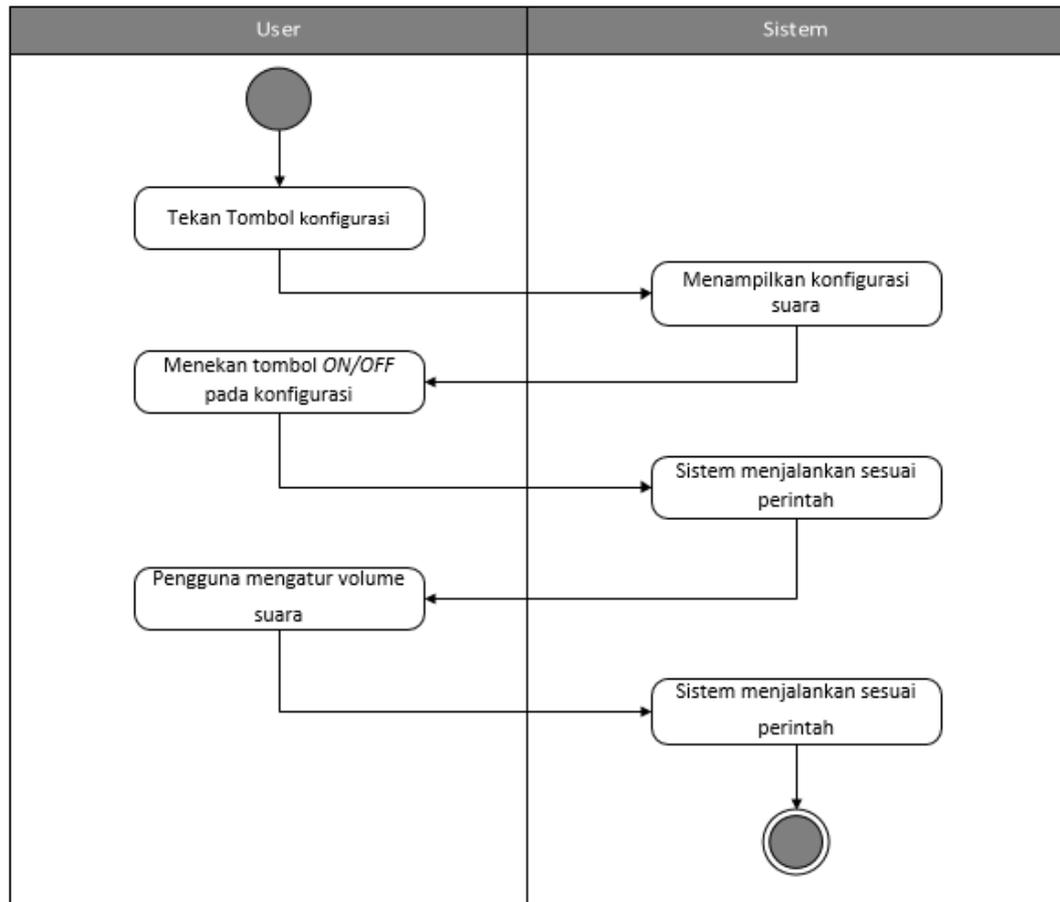
Gambar 3.3 merupakan *activity diagram* untuk menampilkan menu Display Rujukan Ibadah Haji, dimana pengguna menekan tombol Rujukan Ibadah Haji, maka sistem akan berpindah ke menu selanjutnya, didalam halaman tersebut terdapat beberapa pilihan rujukan ibadah haji, lalu pengguna memilih salah satu yang tersedia, maka sistem akan menampilkan rujukan tentang ibadah haji.



Gambar 3.3  
Activity Diagram Display Rujukan Ibadah Haji

### 3.5.3 Activity Diagram Konfigurasi

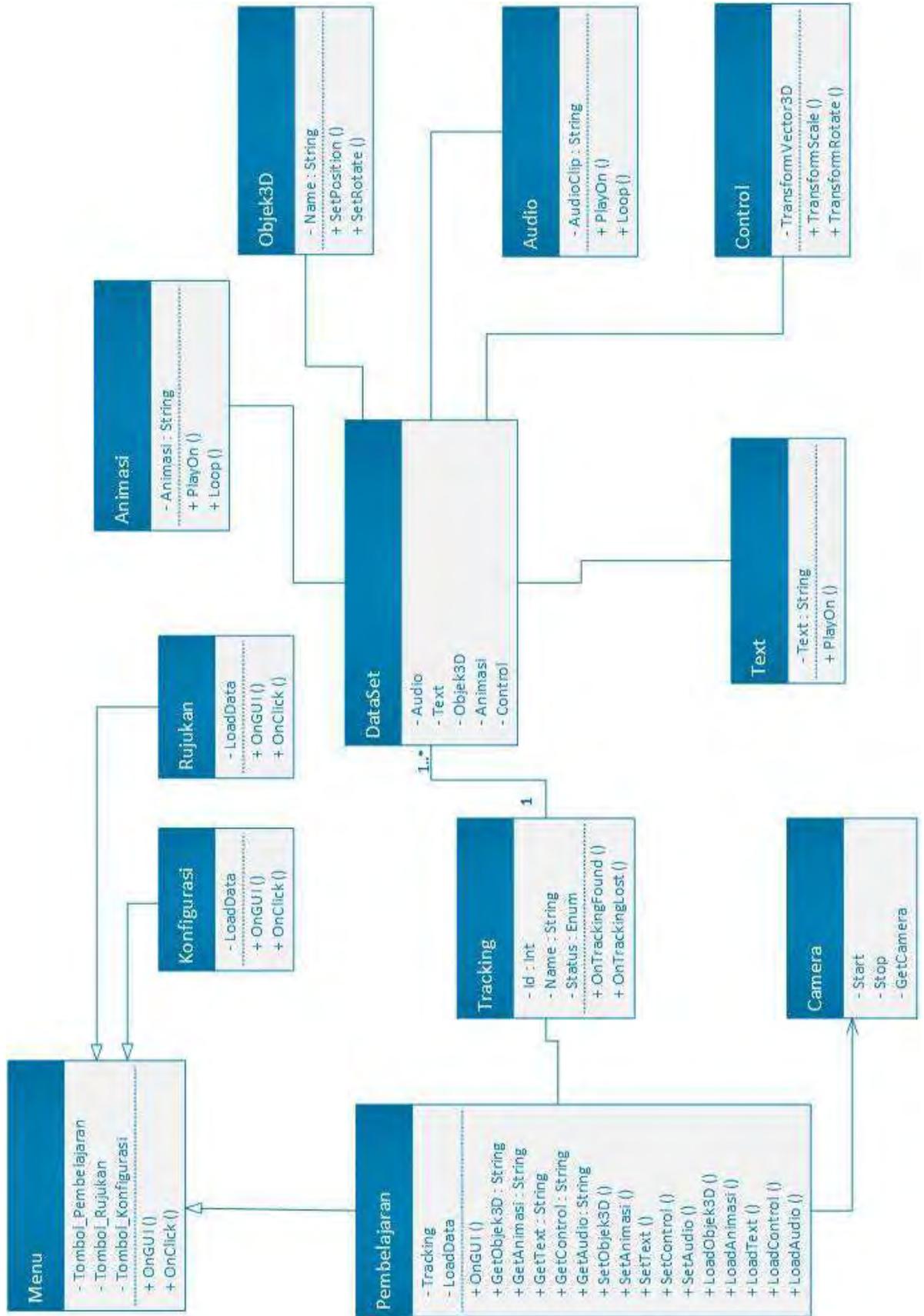
Gambar 3.4 merupakan *activity diagram* untuk menu konfigurasi, dimana pengguna menekan tombol konfigurasi, maka sistem akan menampilkan konfigurasi suara *ON/OFF* dan pengaturan volume suara.



Gambar 3.4  
Activity Diagram konfigurasi

### 3.6 Class Diagram

*Class diagram* adalah diagram yang menggambarkan hubungan antar tiap kelas yang ada pada setiap sistem. *Class diagram* untuk perancangan media pembelajaran adalah sebagai berikut seperti pada Gambar 3.5.



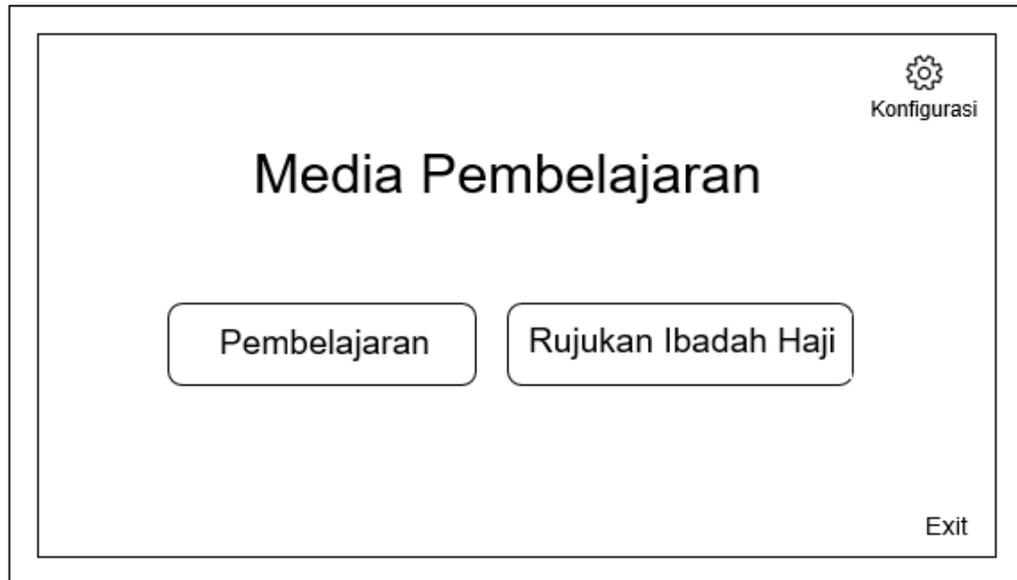
Gambar 3.5  
Class Diagram Media Pembelajaran

### 3.7 Rancangan Antar Muka

Berikut adalah rancangan antar muka dari media pembelajaran yang akan dibuat.

#### 1. Rancangan antar muka Menu Utama

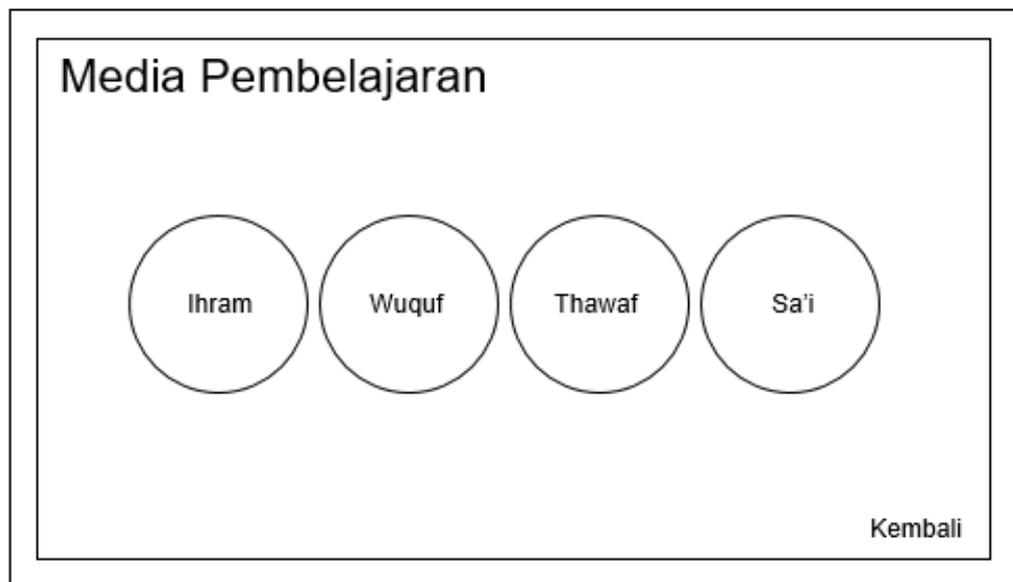
Gambar 3.6 merupakan rancangan antar muka menu utama yang terdiri dari beberapa tombol seperti tombol Pembelajaran, tombol Rujukan Ibadah Haji, Tombol Konfigurasi dan tombol Exit.



Gambar 3.6  
Rancangan Antar Muka Menu Utama

#### 2. Rancangan antar muka Pembelajaran

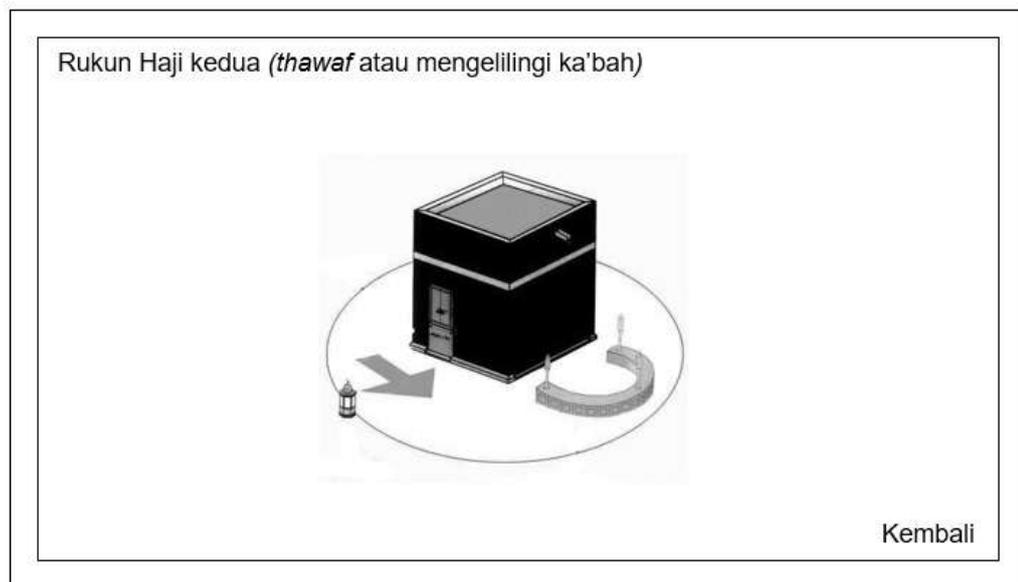
Gambar 3.7 merupakan rancangan antar muka pada *scene* Pembelajaran dimana di dalamnya terdapat beberapa tombol seperti Rukun Haji Pertama (*Ihram*), Rukun Haji Kedua (*Wuquf*), Rukun Haji Ketiga (*Thawaf*), dan Rukun Haji Keempat (*Sa'i*). Setelah pengguna memilih salah satu maka sistem akan masuk kedalam pembelajaran, dimana pengguna dapat mengontrol (memperbesar atau memperkecil) objek, memilih objek yang ada didalam lalu sistem akan menjelaskan objek yang telah dipilih.



Gambar 3.7  
Rancangan Antar Muka Pembelajaran

3. Rancangan antar muka menampilkan objek 3D

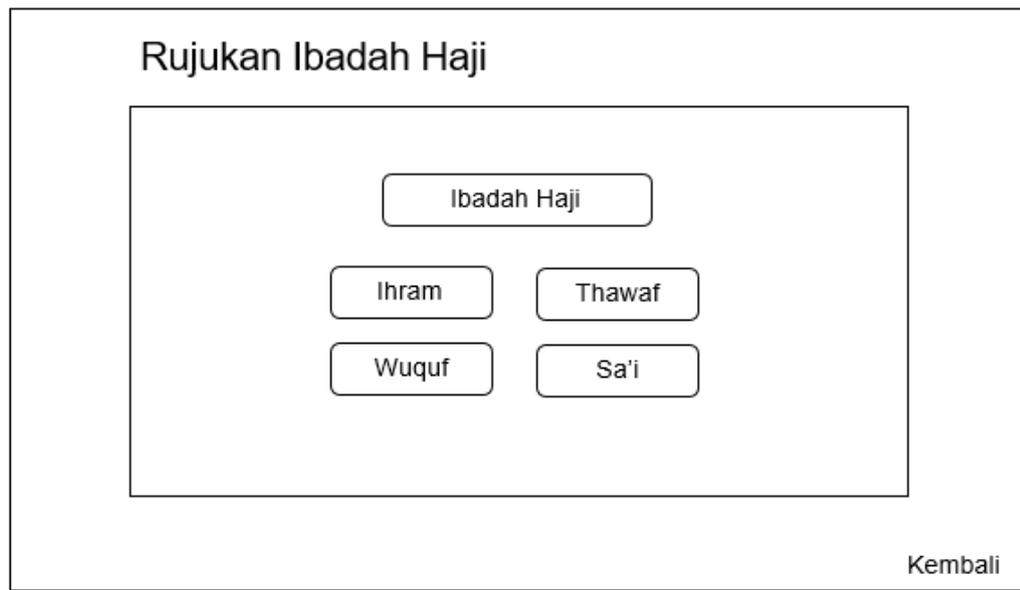
Gambar 3.8 merupakan rancangan antar muka untuk menampilkan objek 3D, dimana dalam rancangan tersebut sistem akan menampilkan rukun haji yang telah dipilih oleh pengguna.



Gambar 3.8  
Rancangan Antar Muka Menampilkan objek 3D

#### 4. Rancangan antar muka menu Display Rujukan Ibadah Haji

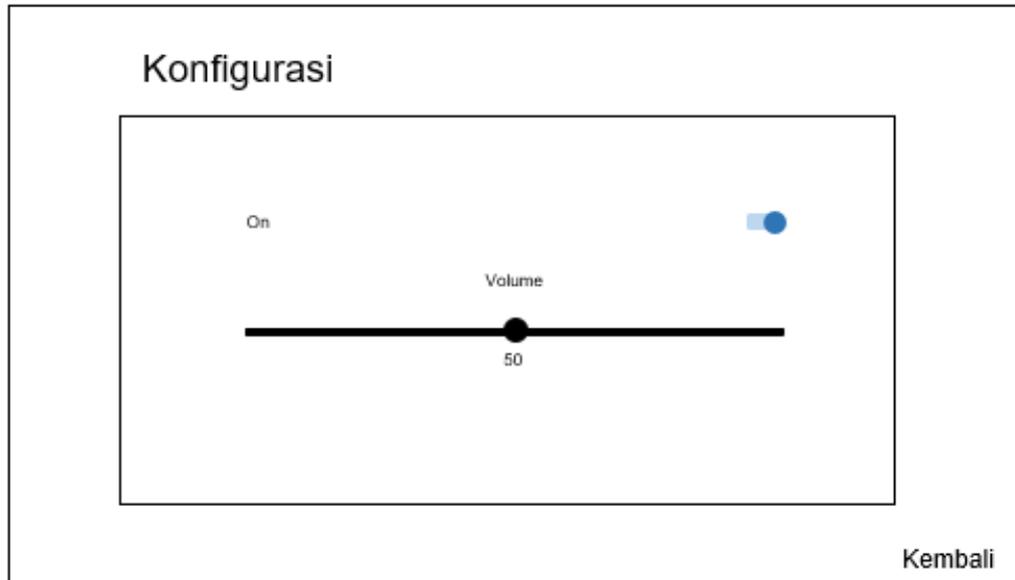
Gambar 3.9 merupakan rancangan antar muka untuk menu Display Rujukan Ibadah Haji, dimana dalam rancangan tersebut terdapat beberapa tombol, pada saat pengguna memilih salah satu tombol, maka sistem akan berpindah ke halaman selanjutnya yang berisi penjelasan singkat tentang Ibadah Haji.



Gambar 3.9  
Rancangan Antar Muka Menu Display Rujukan Ibadah Haji

#### 5. Rancangan antar muka Konfigurasi

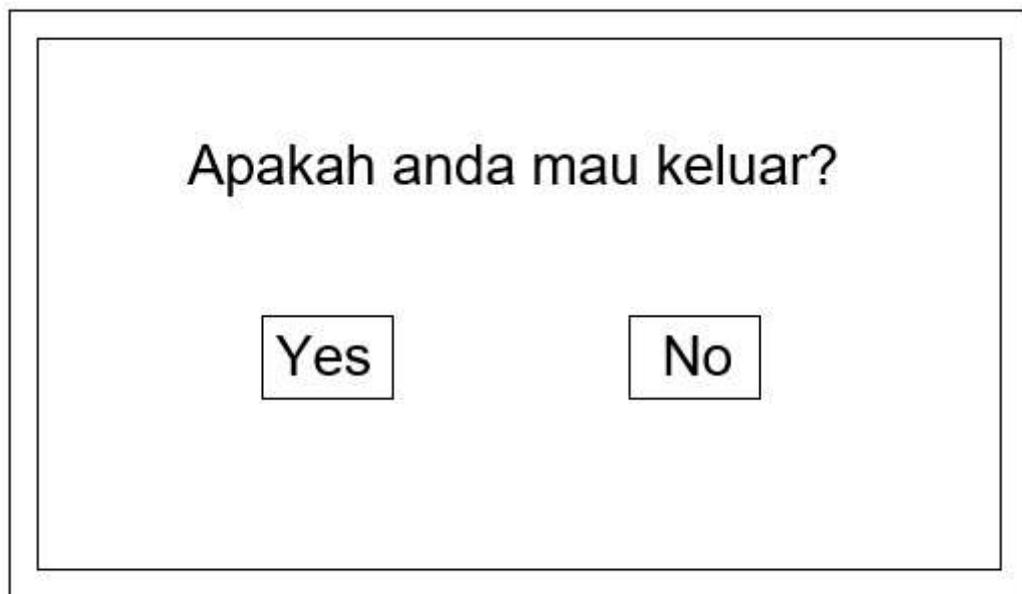
Gambar 3.10 merupakan rancangan antar muka untuk konfigurasi, dimana apabila pengguna menekan tombol konfigurasi yang berada dalam menu utama maka akan muncul konfigurasi suara. Terdapat beberapa tombol di dalamnya yaitu *ON/OFF* dan pengaturan volume suara. Apabila pengguna menekan tombol *ON*, maka sistem akan mengeluarkan suara. Apabila pengguna menekan tombol *OFF*, maka sistem tidak akan mengeluarkan suara.



Gambar 3.10  
Rancangan Antar Muka Menu Konfigurasi

6. Rancangan antar muka Konfirmasi *Exit*

Gambar 3.11 merupakan rancangan antar muka untuk konfirmasi *exit*, dimana apabila pengguna menekan tombol *exit* yang berada dalam menu utama maka akan muncul konfirmasi *exit*. Terdapat dua buah tombol di dalamnya yaitu *YES* dan *NO*. Apabila pengguna menekan tombol *YES*, maka akan keluar dari aplikasi media pembelajaran. Apabila pengguna menekan tombol *NO*, maka sistem akan kembali menu utama.



Gambar 3.11  
Rancangan Antar Muka Konfirmasi *Exit*

## BAB IV IMPLEMENTASI DAN PENGUJIAN PERANGKAT LUNAK

### 4.1 Spesifikasi dan Implementasi Pengujian Perangkat Lunak

Pembuatan aplikasi media pembelajaran untuk perangkat *mobile* ini dirancang menggunakan sebuah laptop yang memiliki spesifikasi :

Tabel 4.1  
Spesifikasi Perangkat Keras Komputer

Konfigurasi Perangkat Lunak	Spesifikasi
1. Processor	Intel® Core™ i5-6200U CPU 2.3 GHz
2. VGA Card	NVIDIA GeForce GT930MX
3. Memori	12GR DDR4
4. Harddisk	1 TB
5. Monitor	14 Inch

Penulis menerapkan aplikasi media pembelajaran yang dirancang, menggunakan perangkat lunak sebagai berikut :

1. Sistem Operasi Windows 10

Sistem operasi yang digunakan dalam pembuatan aplikasi media pembelajaran ini adalah Windows 10, karena sistem operasi ini sudah banyak mendukung banyak perangkat lunak untuk membangun / mengembangkan berbagai macam aplikasi.

2. Unity Editor

Unity editor adalah perangkat lunak untuk merancang berbagai aplikasi yang salah satunya adalah aplikasi yang berbasis Android.

3. Sistem Operasi Android 5.0 (*Lollipop*)

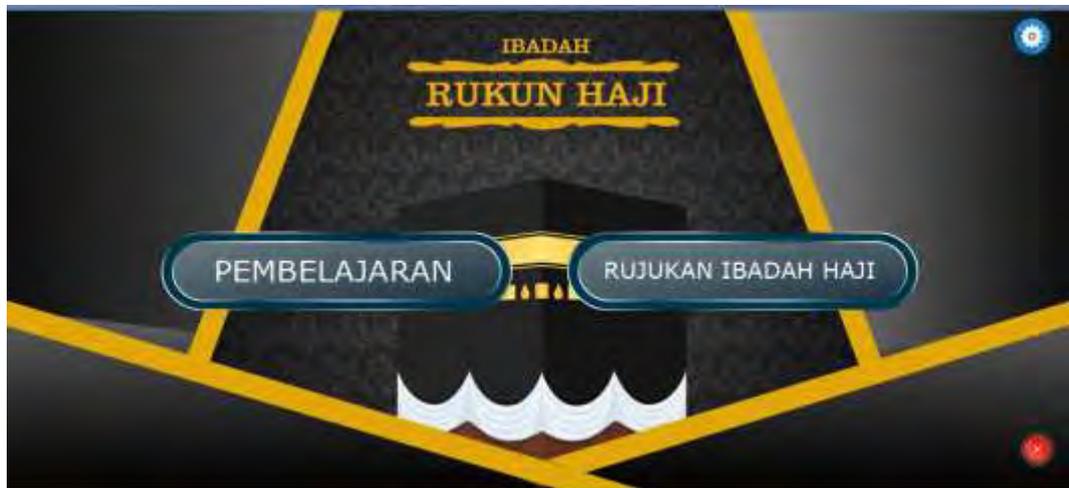
Penulis menerapkan sistem Android versi 5.0, karena *smartphone* dengan sistem operasi ini sebagian besar sudah mendukung sensor *gyroscope* yang diperlukan untuk menjalankan aplikasi media pembelajaran ini.

## 4.2 Tampilan Antar Muka

Berikut ini adalah *screen shoot* dari seluruh kemungkinan yang terjadi dalam media pembelajaran

### 1. Tampilan Menu Utama

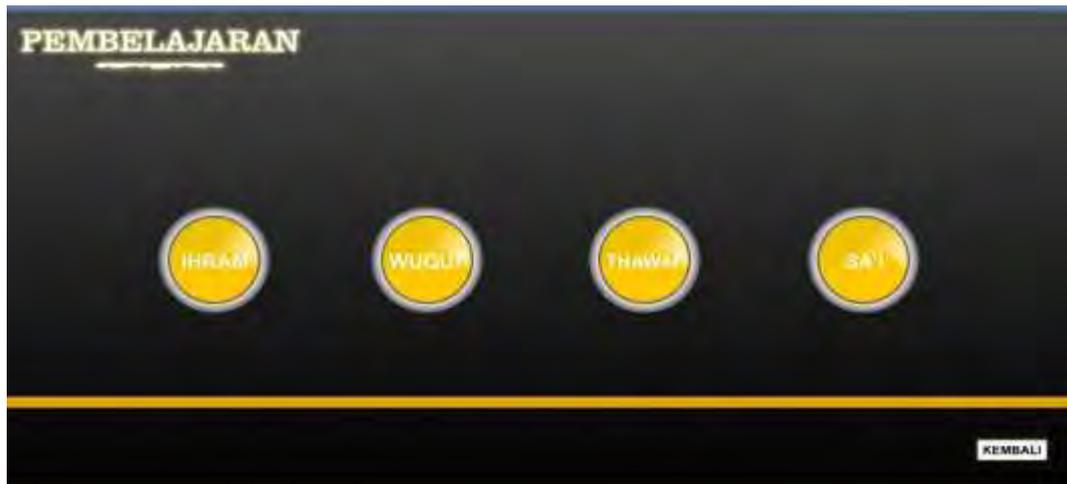
Gambar 4.1 merupakan tampilan menu utama, dimana pengguna dapat memilih berbagai menu yang tersedia seperti menu pembelajaran untuk berpindah ke menu pembelajaran, menu rujukan ibadah haji untuk berpindah ke menu rujukan ibadah haji, menu konfigurasi untuk mengatur *sound*, dan menu *exit* untuk berpindah ke menu konfirmasi *exit*.



Gambar 4.1  
Tampilan Menu Utama

### 2. Tampilan Pembelajaran

Gambar 4.2 merupakan tampilan menu pembelajaran, dimana pengguna dapat memilih berbagai menu yang tersedia seperti menu ihram, wuquf, thawaf, serta sa'l untuk memulai pembelajaran dan menu kembali untuk berpindah ke menu utama.



Gambar 4.2  
Tampilan Pembelajaran

### 3. Tampilan Media Pembelajaran

Gambar 4.3 merupakan tampilan media pembelajaran, dimana ketika pengguna menekan tombol *start* maka objek dan animasi akan muncul, pengguna juga dapat dengan memindahkan, menzoom, serta memutar objek yang terdapat di dalamnya, dan menu kembali untuk berpindah ke menu utama.



Gambar 4.3  
Tampilan Media Pembelajaran

### 4. Tampilan Rujukan Ibadah Haji

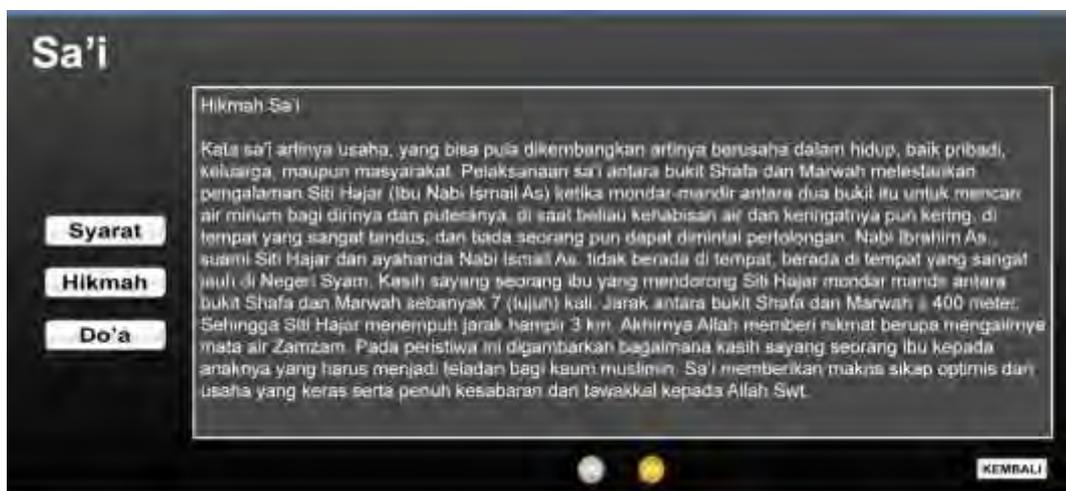
Gambar 4.4 merupakan tampilan menu rujukan ibadah haji, dimana pengguna dapat memilih berbagai menu yang tersedia seperti menu ibadah haji, ihram, wuquf, thawaf, serta sa'i untuk melihat rujukan ibadah haji berupa teks dan menu kembali untuk berpindah ke menu utama.



Gambar 4.4  
Tampilan Rujukan Ibadah Haji

#### 5. Tampilan Isi Rujukan Ibadah Haji

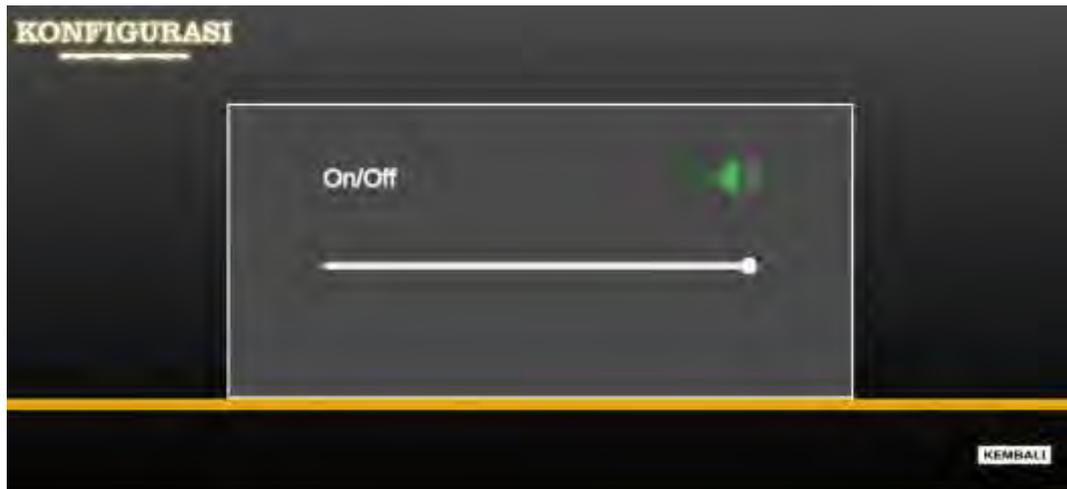
Gambar 4.5 merupakan tampilan menu isi dari rujukan ibadah haji, dimana pengguna dapat melihat isi dari rujukan ibadah haji, tersedia juga tombol *next* dan tombol *previous* untuk melihat halaman selanjutnya serta halaman sebelumnya dan menu kembali untuk berpindah ke pembelajaran.



Gambar 4.5  
Tampilan Isi Rujukan Ibadah Haji

#### 6. Tampilan Konfigurasi

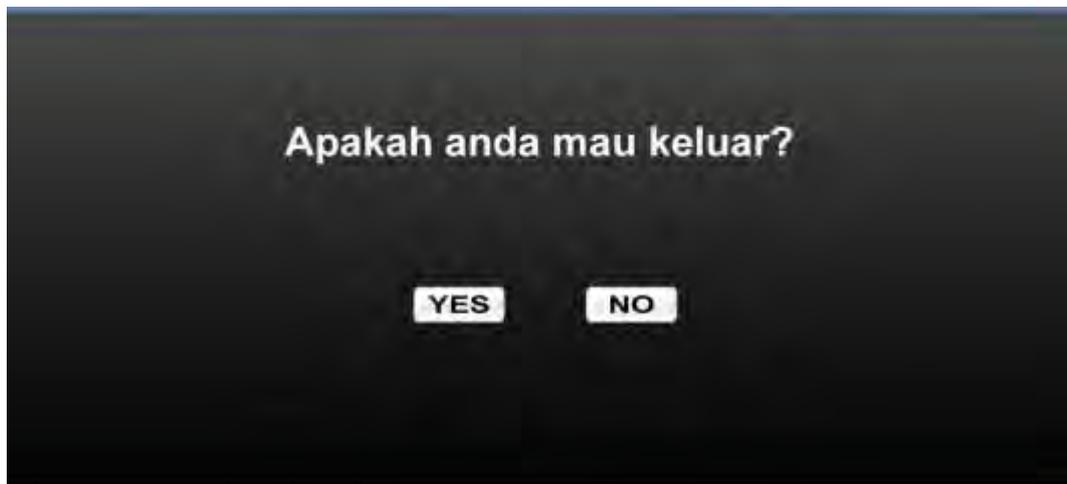
Gambar 4.6 adalah tampilan konfigurasi musik, dimana pengguna dapat mengatur musik dengan menekan tombol *ON/OFF* serta dapat mengatur volume dan menu kembali untuk berpindah ke menu utama.



Gambar 4.6  
Tampilan Konfigurasi

#### 7. Tampilan Konfirmasi *Exit*

Gambar 4.7 adalah tampilan konfirmasi untuk keluar dari aplikasi, dimana tampilan ini akan muncul apabila pengguna menekan tombol *exit* dari *scene* menu utama. Pengguna akan diberikan pertanyaan untuk konfirmasi apakah pengguna akan keluar dari aplikasi atau tidak, dimana apabila pengguna menekan tombol *YES* maka aplikasi akan keluar, apabila pengguna menekan tombol *NO* maka sistem akan kembali ke menu utama.



Gambar 4.7  
Tampilan Konfirmasi *Exit*

### 4.3 Pengujian Fungsi

Pengujian perangkat lunak dilakukan dengan menggunakan metode *black box*, sehingga pengujian fungsi-fungsi perangkat lunak menjadi fokus utama. 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:

#### 1. Tombol-tombol pada Menu Utama

Tabel 4.2  
Tabel pengujian tombol-tombol pada Menu Utama

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik tombol Pembelajaran	Sistem akan masuk ke <i>scene</i> Pembelajaran	Sistem masuk ke <i>scene</i> Pembelajaran	Berhasil
2	Klik tombol Rujukan Ibadah Haji	Sistem akan masuk ke <i>scene</i> Rujukan Ibadah Haji	Sistem masuk ke <i>scene</i> Rujukan Ibadah Haji	Berhasil
3	Klik tombol Konfigurasi	Sistem akan masuk ke <i>scene</i> Konfigurasi	Sistem masuk ke <i>scene</i> Konfigurasi	Berhasil
4	Klik tombol Exit	Menampilkan Konfirmasi Exit	Menampilkan Konfirmasi Exit	Berhasil

#### 2. Tombol-tombol pada Pembelajaran

Tabel 4.3  
Tabel pengujian tombol-tombol pada Pembelajaran

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik tombol Ihram	Sistem akan masuk ke <i>scene</i> Ihram dan mengaktifkan kamera	Sistem masuk ke <i>scene</i> Ihram dan kamera aktif	Berhasil
2	Klik tombol Wuquf	Sistem akan masuk ke <i>scene</i> Wuquf dan mengaktifkan kamera	Sistem masuk ke <i>scene</i> Wuquf dan kamera aktif	Berhasil
3	Klik tombol Thawaf	Sistem akan masuk ke <i>scene</i> Thawaf dan mengaktifkan kamera	Sistem masuk ke <i>scene</i> Thawaf dan kamera aktif	Berhasil
4	Klik tombol Sa'i	Sistem akan masuk ke <i>scene</i> Sa'l dan mengaktifkan kamera	Sistem masuk ke <i>scene</i> Sa'l dan kamera aktif	Berhasil

5	Klik tombol Kembali	Sistem akan masuk ke <i>scene</i> Menu Utama	Sistem masuk ke <i>scene</i> Menu Utama	Berhasil
---	---------------------	--	---	----------

### 3. Tombol-tombol pada Media Pembelajaran

Tabel 4.4  
Tabel pengujian tombol-tombol pada Media Pembelajaran

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik tombol <i>Start</i>	Sistem akan menampilkan objek	Sistem menampilkan objek	Berhasil
2	Klik tombol Kembali	Sistem akan masuk ke <i>scene</i> Pembelajaran	Sistem masuk ke <i>scene</i> Pembelajaran	Berhasil

Tabel 4.5  
Tabel pengujian aksi pada Media Pembelajaran

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	<i>Move</i>	Dapat memindahkan objek	Objek dapat Berpindah	Berhasil
2	<i>Zoom in/out</i>	Dapat memperbesar / memperkecil objek	Objek dapat diperbesar / diperkecil	Berhasil
3	<i>Rotate</i>	Dapat memutar objek	Dapat memutar objek	Berhasil

### 4. Tombol-tombol pada Rujukan Ibadah Haji

Tabel 4.6  
Tabel pengujian tombol-tombol pada Rujukan Ibadah Haji

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik tombol Ibadah Haji	Sistem akan masuk ke <i>scene</i> Ibadah Haji	Sistem masuk ke <i>scene</i> Ibadah Haji	Berhasil
2	Klik tombol Ihram	Sistem akan masuk ke <i>scene</i> Ihram	Sistem masuk ke <i>scene</i> Ihram	Berhasil
3	Klik tombol Thawaf	Sistem akan masuk ke <i>scene</i> Thawaf	Sistem masuk ke <i>scene</i> Thawaf	Berhasil
4	Klik tombol Wuquf	Sistem akan masuk ke <i>scene</i> Wuquf	Sistem masuk ke <i>scene</i> Wuquf	Berhasil
5	Klik tombol Sa'i	Sistem akan masuk ke <i>scene</i> Sa'i	Sistem masuk ke <i>scene</i> Sa'i	Berhasil
6	Klik tombol Kembali	Sistem akan masuk ke <i>scene</i> Menu Utama	Sistem masuk ke <i>scene</i> Menu Utama	Berhasil
7	Klik tombol <i>next</i>	Sistem berpindah ke halaman selanjutnya	Sistem berpindah ke halaman selanjutnya	Berhasil
8	Klik tombol <i>previous</i>	Sistem berpindah ke halaman sebelumnya	Sistem berpindah ke	Berhasil

			halaman sebelumnya	
--	--	--	-----------------------	--

## 5. Tombol-tombol pada Konfigurasi

Tabel 4.7  
Tabel pengujian tombol-tombol pada Konfigurasi

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik tombol <i>ON</i>	Sistem akan mengaktifkan Musik	Sistem mengaktifkan Musik	Berhasil
2	Klik tombol <i>OFF</i>	Sistem akan menonaktifkan Musik	Sistem menonaktifkan Musik	Berhasil
3	Volume Kontrol	Sistem mengatur volume	Sistem menampilkan volume	Berhasil
4	Klik tombol Kembali	Sistem akan masuk ke <i>scene</i> Menu Utama	Sistem masuk ke <i>scene</i> Menu Utama	Berhasil

## 6. Tombol-tombol pada Konfirmasi Exit

Tabel 4.8  
Tabel pengujian tombol-tombol pada Konfigurasi

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik tombol Yes	Sistem akan keluar dari aplikasi	Keluar aplikasi	Berhasil
2	Klik tombol No	Sistem akan masuk ke <i>scene</i> Menu Utama	Sistem masuk ke <i>scene</i> Menu Utama	Berhasil

## 7. Hasil pengujian terhadap lingkungan

Tabel 4.9  
Tabel hasil pengujian terhadap lingkungan

No	Lingkungan	Hasil
1	Lingkungan cahaya terang	Semua animasi / objek muncul
2	Lingkungan cahaya redup	Semua animasi / objek muncul
3	Lingkungan gelap	Semua animasi / objek muncul

8. Hasil pengujian dengan tipe *smartphone* berbeda

Tabel 4.10  
Tabel hasil pengujian pada *smartphone*

No	Tipe <i>Smartphone</i>	Hasil
1	Vivo V15 ( <i>Pie</i> )	Semua animasi / objek muncul
2	Samsung Galaxy J2 Pro ( <i>Nougot</i> )	Semua animasi / objek muncul
3	Xiaomi Redmi 4 Prime ( <i>Marshmallow</i> )	Semua animasi / objek muncul

#### 4.4 Hasil Implementasi dan Pengujian Fungsi

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 media pembelajaran secara *Augmented Reality* dengan sistem operasi Android dapat dilakukan dan berjalan baik dengan menggunakan *software* Unity dan akses kamera pada *smartphone* berjalan baik dengan menggunakan *license key* untuk mengaktifkan kamera dari *VOID AR SDK*.

## BAB V KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Dari penulisan Tugas Akhir ini, penulis mengambil beberapa kesimpulan sebagai berikut:

1. Pengimplementasian *Augmented Reality* pada *platform* Android sebagai media pembelajaran yang bersifat *portable*, memberikan kemudahan dalam penggunaan serta dapat digunakan dimana saja. Dengan kemudahan itu, maka aplikasi ini dibuat. Adapun keuntungan-keuntungan yang ada adalah pengguna dapat mengontrol objek 3D dengan cara *move*, *zoom in/out*, dan *rotate*.
2. Pembuatan Media Pembelajaran Rukun Haji secara *Augmented Reality* dengan sistem operasi Android dapat dilakukan menggunakan *game engine* yang bernama *Unity*, karena *Unity* merupakan *game engine multi platform* yang salah satunya untuk pembuatan sebuah aplikasi berbasis Android dan juga dengan bantuan *Software Development Kit* VOID AR yang berfungsi untuk mengaktifkan kamera *smartphone* yang memiliki banyak fitur untuk melacak suatu target.
3. Sebuah animasi 3D dapat dilacak dan ditampilkan dengan cara *markerless* menggunakan metode VOID SLAM dimana *smartphone* membutuhkan sensor *gyroscope* sehingga apabila suatu objek berada pada posisi tertentu dan pengguna menggerakkan *smartphone* ke kiri, kanan, atas, atau bawah maka objek itu tetap berada pada porosnya.

### 5.2 Saran

Perancangan Media Pembelajaran *Augmented Reality* 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 yang lain, seperti kewajiban-kewajiban haji dan sunah-sunah haji.
2. Melengkapi teks rujukan ibadah haji, seperti kewajiban-kewajiban haji dan sunah-sunah haji.
3. Menambah efek dan variasi pada tombol-tombol agar lebih menarik.

## DAFTAR PUSTAKA

- Abdillah, Syamsuddin, Abu, 2010. *"Terjemah Fathul Qarib: Pengantar Fiqih Imam Syafi'i"*, Subaraya : Mutiara Ilmu.
- Asfari, Bambang, dkk, 2012, *"Pembuatan Aplikasi Tata Ruang Tiga Dimensi Gedung Serba Guna Menggunakan Teknologi Virtual Reality (Studi Kasus: Graha ITS Surabaya)"*, Jurnal Teknik ITS, Vol. 1 No. 1.
- Azuma, Ronald T, 1997, *"A Survey of Augmented Reality"*. Presence Vol. 6 No. 4.
- Clark, Dan, 2011, *"Beginning C# Object-Oriented Programming"*, New York : Apress.
- Creighton, 2010, *"Unity 3D Game Development by Example: Beginner's Guide"*, Birmingham - Mumbai : Packt Publishing.
- Dharwiyanti, Sri, 2003, *"Pengantar Unified Modeling Language (UML)"*, Ilmu Komputer.
- Furht, Borko, 2011, *"Handbook of Augmented Reality"*, New York : Springer.
- Indrajani, Martin, 2007, *"Pemrograman Berbasis Objek dengan Bahasa Java"*, Jakarta : PT Elex Media Komputindo.
- Kementerian Agama RI, 2016, *"Tuntunan Manasik Haji dan Umrah"*. Jakarta.
- Kipper, Joseph, 2013, *"Augmented Reality: An Emerging Technologies Guide to AR First Edition"*, United States : Elsevier.
- Kurniawan, Erick, 2011, *"Membangun Aplikasi Mobile dengan Qt SDK"*, Yogyakarta : Andi.
- Mayo, Joe, 2010, *"Microsoft Visual Studio 2010: A Beginner's Guide"*, New York : The McGraw-Hill Companies.
- Peddie, Jon, 2017, *"Augmented Reality: Where We Will All Live"*, Switzerland : Springer.
- Pressman, Roger S, 2010, *"Software Engineering: A Practitioner's Approach Seventh Edition"*, New York : The McGraw-Hill Companies, Inc.
- Putra, Anna, Arthdi, 2012, *"Android dan Anak Tukang Sayur: Buku Praktis Belajar Pemrograman Android"*, Lubuklinggau-Sumatera Selatan.
- Rahman, Ernawati, dkk, 2014, *"Rancang Bangun Aplikasi Informasi Universitas Bengkulu Sebagai Panduan Pengenalan Kampus Menggunakan Metode Markerless Augmented Reality Berbasis Android"*, Jurnal Rekursif, Vol. 2 No. 2.
- Rosa, Shalahuddin, 2018, *"Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek Edisi Revisi"*, Bandung : Informatika.
- Rumbaugh, Ivar, dkk, 2005, *"The Unified Modeling Language Reference Manual Second Edition"*, Addison-Wesley.
- Nazruddin, 2012, *"Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android: Edisi Revisi"*. Bandung : Informatika.
- Nugrohu, Adi, 2009, *"Rekayasa Perangkat Lunak Menggunakan UML dan Java"*, Yogyakarta : Andi.
- Nugrohu, Adi, 2010, *"Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP"*, Yogyakarta : Andi.
- Setiyani, Lila, 2018, *"Rekayasa Perangkat Lunak (Software Engineering)"*, Karawang, Jawa Barat : Jatayu Catra Internusa.

Simanjuntak, Erwin, dkk, 2019, "*Perancangan Antarmuka Pendulum Terbalik Menggunakan Visual Studio*", e-Proceeding of Engineering, Vol. 6 No. 2.

Sommerville, Ian, 2011, "*Software Engineering, Ninth Edition*", Addison-Wesley.

Wibawanto, Wandah, 2017, "*Desain dan Pemrograman Multimedia Pembelajaran Interaktif*", Jember – Jawa Timur : Cerdas Ulet Kreatif.

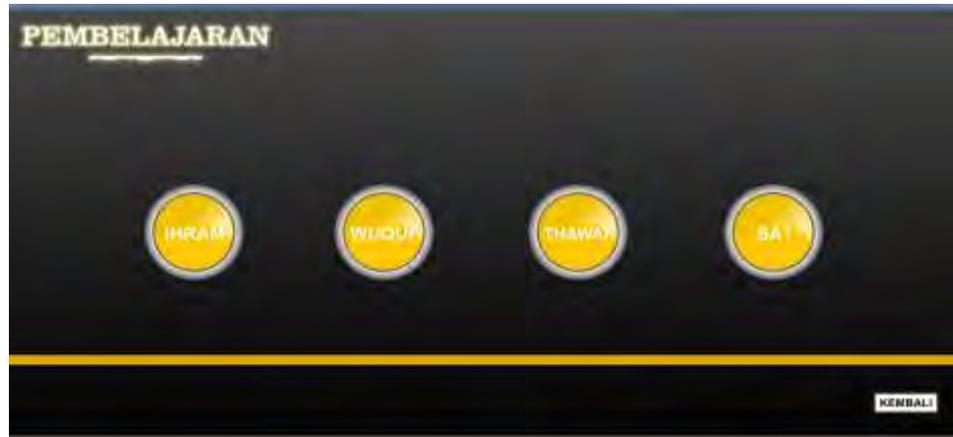
<https://www.blender.org/about/>, Dikutip 27 Januari 2020, Jam 21:30.

<https://www.codepolitan.com/unified-modeling-language-uml>, Dikutip 03 November 2019, Jam 22:00.

[https://www.voidar.com/sdk\\_E.php](https://www.voidar.com/sdk_E.php), Dikutip 25 November 2019. Jam 23:30

**LAMPIRAN 1**  
**CONTOH TAMPILAN PROGRAM**

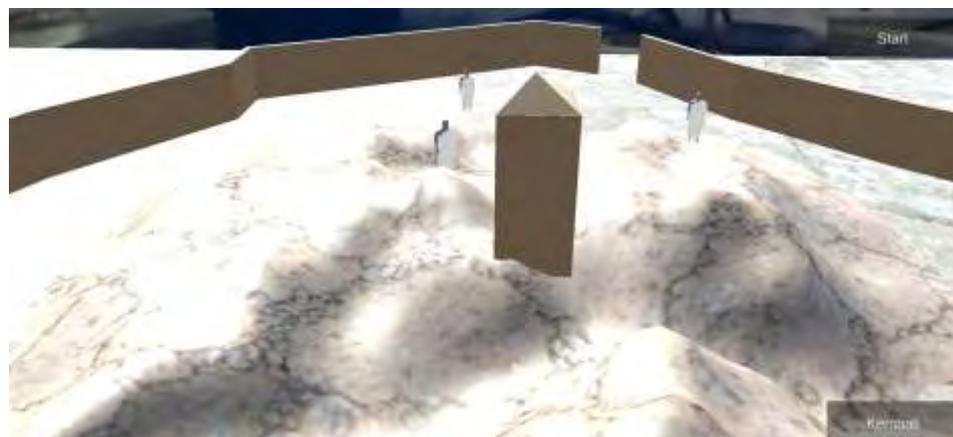
1. Menu Utama



2. Menu Pembelajaran



3. Thawaf AR



## 4. Wuquf AR



## 5. Isi Rujukan Thawaf

## Thawaf

1. Dosa thawaf putaran pertama

Di baca dan hajar aswas sampai rukun yamani

سُبْحَانَ اللَّهِ وَالْحَمْدُ لِلَّهِ وَلَا إِلَهَ إِلَّا اللَّهُ وَاللَّهُ أَكْبَرُ وَلَا يُدْرِكُهُ الْبَصَرُ وَلَا يَحْزُلُ وَلَا يَنْوَلُّ وَلَا يَبُوءُ إِلَّا بِاللَّهِ الْعَلِيِّ الْعَظِيمِ وَالْمَلَائِكَةُ وَالرُّسُلُ عَلَى رُسُولِ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ فَاللَّهُ إِيْمَانًا بِكَ وَتَحَلُّفًا بِحَبْلِكَ وَوَعْدًا بِعَهْدِكَ وَأَمَانًا لِحَبْلِكَ مُحَمَّدٌ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ لَهُمْ أَمْرٌ أَسْقَطَ الْعُقُوبَ وَالْعُقُوبَةَ وَالْمَغْفِرَةَ الْبَاطِنَةَ فِي الْقَبْرِ وَالنَّارَ الْآخِرَةَ وَالْفَوْزَ بِحُجَّةٍ وَالنَّجَاةَ مِنَ النَّارِ

Artinya : "Maha Suci Allah. Engkau puji bagi Allah. Tidak ada Tuhan selain Allah. Allah Maha Besar. Tidak daya untuk menggambar, menyalin ( dan tidak bersampan ) untuk memakainya ( kecuali bersumber dari Allah yang Maha Tinggi dan Maha Agung, sholat dan salam bagi Rasulullah SAW. Ya Allah, aku beresolusi karena beriman kepada-Mu, membenarkan kitab-Mu, dan memercu jani-Mu, dan mengakui simbah Nabi-Mu Muhammad SAW. Ya Allah, sesungguhnya aku mohon kepada-Mu ampunan, keabadian dan kesejahteraan yang kekal dalam menandakan agama, di dunia dan di akhirat dari segala perkara"

**Syarat**

**Hikmah**

**Do'a**

KEMBALI

## 6. Isi Rujukan Wuquf

## Wuquf

Di Padang Arafat

1. Di Arafat (pada tanggal 9 Dzulhijjah hingga 10 Dzulhijjah menjelang wukuf):

- Menunggu waktu wukuf dengan berzikir, tasbeeh dan membaca Al-Qur'an
- Memperbanyak bacaan talbiyah dan berdoa

2. Wukuf tanggal 9 Dzulhijjah dimulai pada zawal hingga terbit fajar tanggal 10 Dzulhijjah. Kaitan jama wukuf menurut Mazhab Syafi'i cukup sesat baik siang maupun malam. Menurut Mazhab Maliki dan Mazhab Hanafi wukuf harus menemui waktu siang dan waktu malam.

**Kegiatan**

**Hikmah**

**Do'a**

Kegiatan wukuf adalah:

1. Didahului dengan mendengarkan khutbah wukuf
2. Sholat Zhuhur dan Ashar jama' taqdim qasar di lanjutkan dengan melaksanakan wukuf
3. Selama wukuf memperbanyak talbiyah, zikir, mem baca Al-Qur'an dan ber do'a
4. Wukuf diakhiri dengan sholat Maghrib dan Isya' jama' taqdim dan qasar, selanjutnya berslap-siap menuju Muzdalifah.

KEMBALI

## LAMPIRAN 2 LISTING PROGRAM

### 1. *MarkerlessUI.cs*

```
using UnityEngine;

public class MarkerlessUI : MonoBehaviour {

    private int toolSelectIndex = 1;
    private int modelIndex = 0;
    private float delay = 0.020f;
    private bool isInitOK = false;

    private int scanMode = 1;
    private bool isStart = false;

        private int recordSelectIndex = 1;

        private string[] fileNames;
        int selectIndex = -1;
    public void OnBackHandler()
    {
        Application.LoadLevel("Menu");
    }

    public void startMarkerlessTracking() {
        VoidAR.GetInstance().startMarkerlessTracking();
    }
}
}
```

### 2. *SimpleDrag.cs*

```
using UnityEngine;
using System.Collections;
using System.Collections.Generic;
using UnityEngine.Events;
using UnityEngine.UI;
using UnityEngine.EventSystems;

public class SimpleDrag : MonoBehaviour
{
    public LayerMask LayerMask = 1<<8;//User-Defined Layer for Arrows;
    public LayerMask LayerMask2 = 1<<9;//User-Defined Layer for Objects;

    private Lean.LeanFinger draggingFinger;
    private int status;
    public Transform targetOfPosition;
    public Transform targetOfSizeAndRotation;
    int mode;
    float timer;
```

```

public GameObject red;
public GameObject green;
public GameObject blue;

void Awake(){
    status = 0;
    mode = 0;
    timer = 0;
}
protected virtual void OnEnable()
{
    if (MoveScript.mode != DEMO_STATUS.READY) {
        return;
    }
    Lean.LeanTouch.OnFingerDown += OnFingerDown;

    Lean.LeanTouch.OnFingerUp += OnFingerUp;
}

protected virtual void OnDisable()
{
    if (MoveScript.mode != DEMO_STATUS.READY) {
        return;
    }
    Lean.LeanTouch.OnFingerDown -= OnFingerDown;

    Lean.LeanTouch.OnFingerUp -= OnFingerUp;
}

protected virtual void LateUpdate()
{
    if (MoveScript.mode != DEMO_STATUS.READY) {
        return;
    }
    if (status == 0) {
        timer += Time.deltaTime;
    }
    if (draggingFinger != null)
    {
        if (mode == 0) {
            if (Mathf.Abs (Lean.LeanTouch.PinchScale - 1) > 0.05f) {
Lean.LeanTouch.ScaleObject(targetOfSizeAndRotation,Lean.LeanTouch.PinchScale);

            } else {
                if (status == 1) {
                    Vector2 p = Camera.main.WorldToScreenPoint (new Vector3 (0, 0, 0.4f));
                    Vector2 o = Camera.main.WorldToScreenPoint (new Vector3 (0, 0, 0));
                    float d = Vector2.Dot (Vector2.ClampMagnitude ((p - o), 1),
                    Vector2.ClampMagnitude (draggingFinger.DeltaScreenPosition, 1));
                    Lean.LeanTouch.MoveObjectZ (targetOfPosition, d * Mathf.Sqrt
                    (draggingFinger.DeltaScreenPosition.x * draggingFinger.DeltaScreenPosition.x +
                    draggingFinger.DeltaScreenPosition.y * draggingFinger.DeltaScreenPosition.y));

                } else if (status == 2) {
                    Lean.LeanTouch.MoveObject (targetOfPosition, new Vector2 (0,
                    draggingFinger.DeltaScreenPosition.y));
                }
            }
        }
    }
}

```

```

    } else if (status == 3) {
        Vector2 p = Camera.main.WorldToScreenPoint (new Vector3 (0.4f, 0, 0));
        Vector2 o = Camera.main.WorldToScreenPoint (new Vector3 (0, 0, 0));
        float d = Vector2.Dot (Vector2.ClampMagnitude ((p - o), 1),
Vector2.ClampMagnitude (draggingFinger.DeltaScreenPosition, 1));
        Lean.LeanTouch.MoveObjectX (targetOfPosition, d * Mathf.Sqrt
(draggingFinger.DeltaScreenPosition.x * draggingFinger.DeltaScreenPosition.x +
draggingFinger.DeltaScreenPosition.y * draggingFinger.DeltaScreenPosition.y));

    } else if (status == 4) {
        Lean.LeanTouch.RotateObjectY(targetOfSizeAndRotation,
draggingFinger.DeltaScreenPosition.x);

    } else if(status == 5){
        Lean.LeanTouch.ScaleObject(targetOfSizeAndRotation,
Lean.LeanTouch.PinchScale);

    }else {
        Lean.LeanTouch.RotateObjectY(targetOfSizeAndRotation,
draggingFinger.DeltaScreenPosition.x);
    }
}

Lean.LeanTouch.ScaleObject(targetOfSizeAndRotation, Lean.LeanTouch.PinchScale);

if (timer > 3) {
    red.GetComponent<MeshRenderer> ().enabled = false;
    green.GetComponent<MeshRenderer> ().enabled = false;
    blue.GetComponent<MeshRenderer> ().enabled = false;
} else {
    red.GetComponent<MeshRenderer> ().enabled = true;
    green.GetComponent<MeshRenderer> ().enabled = true;
    blue.GetComponent<MeshRenderer> ().enabled = true;
}

}

public void OnFingerDown(Lean.LeanFinger finger)
{
    if (MoveScript.mode != DEMO_STATUS.READY) {
        return;
    }
    timer = 0;
    // Raycast information
    var ray = finger.GetRay();
    var hit = default(RaycastHit);
    if (draggingFinger != null) {
        draggingFinger = finger;
        status = 5;
        return;
    }
    // Was this finger pressed down on a collider?
    if (Physics.Raycast (ray, out hit, float.PositiveInfinity, LayerMask) == true)
    {
        if (hit.transform.tag == "RED") {
            status = 1;

```

```

        draggingFinger = finger;
        red.GetComponent<MeshRenderer> ().material.SetColor
("_Diffuse",Color.yellow);
    } else if (hit.transform.tag == "GREEN") {
        status = 2;
        draggingFinger = finger;
        green.GetComponent<MeshRenderer>().material.SetColor
("_Diffuse",Color.yellow);

        } else if (hit.transform.tag == "BLUE") {
            status = 3;
            draggingFinger = finger;
            blue.GetComponent<MeshRenderer>().material.SetColor
("_Diffuse",Color.yellow);

        } else {

            if (Physics.Raycast (ray, out hit, float.PositiveInfinity, LayerMask2) == true)
        {
                draggingFinger = finger;
                status = 4;

                } else {
                    status = 0;
                }
            }

        }else if (Physics.Raycast (ray, out hit, float.PositiveInfinity, LayerMask2) == true) {

            draggingFinger = finger;
            status = 4;

        }
    }

    public void OnFingerUp(Lean.LeanFinger finger)
    {
        if (MoveScript.mode != DEMO_STATUS.READY) {
            return;
        }

        if (status == 1) {
            red.GetComponent<MeshRenderer>().material.SetColor
("_Diffuse",Color.red);

        } else if (status == 2) {
            green.GetComponent<MeshRenderer>().material.SetColor
("_Diffuse",Color.green);

        } else if (status == 3) {
            blue.GetComponent<MeshRenderer>().material.SetColor
("_Diffuse",Color.blue);

        } else {
            red.GetComponent<MeshRenderer>().material.SetColor
("_Diffuse",Color.red);
            green.GetComponent<MeshRenderer>().material.SetColor
("_Diffuse",Color.green);

```

```

        blue.GetComponent<MeshRenderer>().material.SetColor
        ("_Diffuse",Color.blue);
    }
    // Was the current finger lifted from the screen?
    if (finger == draggingFinger)
    {
        // Unset the current finger
        draggingFinger = null;
        status = 0;
    }
}

```

### 3. SimpleRotateScale.cs

```

using UnityEngine;

public class SimpleRotateScale : MonoBehaviour
{
    protected virtual void LateUpdate()
    {
        Lean.LeanTouch.RotateObject(transform,Lean.LeanTouch.TwistDegrees);

        Lean.LeanTouch.RotateObject(transform,Lean.LeanTouch.TwistRadians);

        Lean.LeanTouch.ScaleObject(transform, Lean.LeanTouch.PinchScale);
    }
}

```

### 4. MarkerlessHelp.cs

```

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

public class MarkerlessHelp : MonoBehaviour {

    public bool isPlay;
    public AudioSource SuaraAR;

    void Start()
    {
        isPlay = false;
    }
    void OnGUI()
    {
        var btnHeight = Screen.height * 0.1f;
        var btnWidth = btnHeight * 3.0f;
        var gap = 20;
        GUI.skin.button.fontSize = 36;
        if (GUI.Button(new Rect(Screen.width - btnWidth, gap, btnWidth, btnHeight), "Start"))
        {
            VoidAR.GetInstance().startMarkerlessTracking();
        }
    }
}

```

```
SuaraAR.Play();
isPlay = !isPlay;
this.SuaraAR.volume = 1;
}
if (GUI.Button(new Rect(Screen.width - btnWidth, Screen.height - btnHeight,
btnWidth, btnHeight), "Kembali"))
{
    SceneManager.LoadScene("Pembelajaran");
    isPlay = !isPlay;
    AudioListener.pause = !isPlay;
}
}
}
```