

**PEMBANGUNAN APLIKASI TUR INTERAKTIF  
PERUMAHAN BALIMBINGAN PERMAI PT. KARYA PROPERTINDO  
UTAMA BERBASIS VIRTUAL REALITY**

**SKRIPSI**

Keminatan Multimedia, Game, dan Mobile

Disusun oleh

REINHARD JONATHAN SLAHI

175150200111040



**TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

**MALANG**

**2020**

# **PENGESAHAN**

**PEMBANGUNAN APLIKASI TUR INTERAKTIF  
PERUMAHAN BALIMBINGAN PERMAI PT. KARYA PROPERTINDO UTAMA  
BERBASIS VIRTUAL REALITY**

**SKRIPSI**

**Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer**

**Disusun Oleh :  
Reinhard Jonathan Silalahi  
NIM: 175150200111040**

*Skripsi ini telah diuji dan dinyatakan lulus pada  
30 Juni 2020*

**Telah diperiksa dan disetujui oleh:**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Komang Candra Brata, S.Kom., M.T., M.Sc.**  
**NIK: 201607 890711 1 000**

**Adam Hendra Brata, S.Kom., M.T., M.Sc.**  
**NIP: 199001052019031009**

**Mengetahui  
Ketua Jurusan Teknik Informatika**

**Achmad Basuki, S.T., M.MG., Ph.D.**  
**NIP: 197411182003121002**

## **PERNYATAAN ORISINALITAS**

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 18 Juni 2020

Reinhard Jonathan Silalahi

NIM: 175150200111040

## **PRAKATA**

Segala puji dan syukur bagi Tuhan Yang Maha Esa atas berkatnya penulis dapat menyelesaikan skripsi yang berjudul “Pembangunan Aplikasi Tur Interaktif Perumahan Balimbingan Permai PT. Karya Propertindo Utama Berbasis Virtual Reality”. Dalam kesempatan ini, penulis ingin memberikan ucapan terima kasih kepada seluruh pihak yang telah membantu penulis dalam proses penyusunan skripsi ini, diantaranya:

1. Kedua orang tua penulis, yaitu Bapak Mangantar Silalahi dan Ibu Helena Panjaitan beserta seluruh keluarga besar yang terus memberikan dukungan dan motivasi serta doa yang tidak kunjung putus.
2. Bapak Komang Candra Brata, S.Kom., M.T., M.Sc. sebagai dosen pembimbing I dan Bapak Adam Hendra Brata, S.Kom., M.T., M.Sc. sebagai dosen pembimbing II, atas segala bimbingan, masukan, dan saran yang bermanfaat bagi penulis selama proses penelitian skripsi ini.
3. Seluruh Bapak dan Ibu Dosen Fakultas Ilmu Komputer, Universitas Brawijaya atas ketersediaannya membimbing dan memberikan ilmu yang bermanfaat kepada penulis.
4. Seluruh teman dan sahabat penulis yang secara tidak langsung memberikan ilmu serta motivasi dalam penyusunan skripsi ini.
5. Semua pihak yang tidak semuanya bisa dituliskan disini yang terlibat secara langsung maupun tidak langsung dalam penyusunan skripsi ini.

Penulis sadar masih banyak kekurangan dan kesalahan yang terdapat dalam penyusunan skripsi ini baik dari sisi teknis penelitian maupun penyajian materi. Maka dari itu, penulis mengharapkan kritik dan saran yang membangun dari pembaca. Semoga karya tulis ini bermanfaat dan dapat memberikan manfaat bagi pihak yang membutuhkan.

Malang, 18 Juni 2020

Penulis

reinhardjs@student.ub.ac.id

## ABSTRAK

**Reinhard Jonathan Silalahi, Pembangunan Aplikasi Tur Interaktif Perumahan Balimbingan Permai PT. Karya Propertindo Utama Berbasis Virtual Reality**

**Pembimbing: Komang Candra Brata, S.Kom., M.T., M.Sc. dan Adam Hendra Brata, S.Kom., M.T., M.Sc.**

Balimbingan Permai, merupakan perumahan yang dibangun oleh perusahaan PT. Karya Propertindo Utama dengan 48% konsumennya batal beli rumah memiliki alasan/kendala tidak dapat melakukan pengamatan dikarenakan beberapa dari mereka berdomisili di luar kota dan memiliki waktu yang sangat terbatas. Berdasarkan permasalahan tersebut, penulis membuat aplikasi tur interaktif perumahan balimbingan permai. Aplikasi yang dikembangkan berupa dua aplikasi, yaitu aplikasi untuk pengguna dan aplikasi untuk admin. Penelitian ini menggunakan metode pengembangan perangkat lunak dengan kerangka kerja waterfall dengan kebutuhan pengguna yang spesifik. Hasil dari penelitian ini berfokus pada penyelesaian masalah melakukan tur interaktif secara virtual. Selain itu, peneliti juga melakukan tiga pengujian, yaitu pengujian validasi, *usability* untuk mengetahui tingkat kebergunaan aplikasi bagi pengguna, dan *compatibility* untuk mengetahui aplikasi mampu dijalankan pada beberapa versi android. Dari pengujian validasi didapatkan hasil bahwa validasi yang dilakukan terhadap kebutuhan fungsionalitas dinyatakan valid, sedangkan dari hasil pengujian *usability* didapatkan hasil bahwa aplikasi untuk pengguna mendapatkan nilai *usability* sebesar 82,33% yang masuk ke dalam kategori B, dengan *adjective rating excellent*. Aplikasi admin mendapatkan nilai *usability* sebesar 96,25% yang masuk ke dalam kategori A, dengan *adjective rating best imaginable*. Dan dari hasil pengujian *compatibility* didapatkan hasil bahwa aplikasi dapat dijalankan pada sistem operasi Android dengan versi 6.0 sampai dengan 9.0.

Kata kunci: Pembangunan aplikasi, *virtual reality*, Waterfall, *usability*, *compatibility*, Android Studio, Unity 3D Engine.

## ABSTRACT

**Reinhard Jonathan Silalahi, Development of an Interactive Tour Application for Balimbingan Permai PT. Propertindo Utama Based on Virtual Reality**

**Supervisors: Komang Candra Brata, S.Kom., M.T., M.Sc. and Adam Hendra Brata, S.Kom., M.T., M.Sc.**

*Balimbingan Permai, is a housing estate built by the company PT. Karya Propertindo Utama with 48% of its consumers canceling buying a house has reasons/obstacles that they cannot make observations because some of them live outside the city and have very limited time. Based on these problems, the author makes an interactive tour application for the beautiful balimbingan housing. The application developed is in the form of two applications, namely an application for users and an application for admins. This study uses a software development method with a waterfall framework with specific user needs. The result of this research focuses on solving the problem of doing a virtual interactive tour. In addition, the researcher also conducted three tests, namely validation testing, usability to determine the level of usability of the application for users, and compatibility to determine whether the application can run on several versions of android. From the validation test, it was found that the validation carried out on the functionality requirements was declared valid, while the usability test results showed that the application for users got a usability value of 82.33% which was included in category B, with an adjective rating of excellent. The admin application gets a usability value of 96.25% which falls into category A, with an adjective rating of best imaginable. And from the results of compatibility testing, it is found that the application can be run on the Android operating system with versions 6.0 to 9.0.*

*Keywords: application development, virtual reality, Waterfall, usability, compatibility, Android Studio, Unity 3D Engine.*

## DAFTAR ISI

<b>PENGESAHAN.....</b>	<b>ii</b>
<b>PERNYATAAN ORISINALITAS.....</b>	<b>iii</b>
<b>PRAKATA.....</b>	<b>iv</b>
<b>ABSTRAK.....</b>	<b>v</b>
<b>DAFTAR ISI .....</b>	<b>vii</b>
<b>DAFTAR GAMBAR.....</b>	<b>x</b>
<b>DAFTAR TABEL .....</b>	<b>xi</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>xiii</b>
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah .....	5
1.3 Rumusan Masalah .....	5
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian .....	6
1.6 Batasan Penelitian/Ruang Lingkup Penelitian .....	6
1.7 Sistematika Pembahasan/Laporan.....	7
<b>BAB 2 LANDASAN KEPUSTAKAAN.....</b>	<b>9</b>
2.1 KAJIAN PUSTAKA .....	9
2.2 Virtual Reality .....	11
2.3 Unity 3D.....	11
2.4 Android.....	12
2.5 Pengujian Fungsional .....	12
2.6 Pengujian <i>Usability</i> .....	12
<b>BAB 3 METODOLOGI PENELITIAN .....</b>	<b>13</b>
3.1 Studi Literatur.....	14
3.2 Analisis Kebutuhan Sistem .....	15
3.3 Perancangan Sistem .....	15

3.4	Implementasi Sistem .....	15
3.5	Pengujian Sistem .....	16
3.6	Kesimpulan dan Saran .....	16
<b>BAB 4</b>	<b>REKAYASA KEBUTUHAN .....</b>	<b>17</b>
4.1.	Gambaran Umum Aplikasi .....	17
4.1.1.	Deskripsi .....	17
4.1.2.	Cara Penggunaan.....	17
4.2.	Analisis Kebutuhan Sistem .....	17
4.2.1	Identifikasi Aktor .....	18
4.2.2	Kebutuhan Fungsional .....	18
4.2.3	Kebutuhan Non Fungsional.....	19
4.3.	Pemodelan Kebutuhan .....	19
4.3.1.	<i>Use Case Diagram</i> .....	19
4.3.2.	<i>Use case Scenario</i> .....	20
<b>BAB 5</b>	<b>PERANCANGAN DAN IMPLEMENTASI .....</b>	<b>22</b>
5.1	Perancangan .....	22
5.1.1	Perancangan Arsitektur Sistem.....	22
5.1.2	Perancangan Komponen .....	23
5.1.3	<i>Sequence Diagram</i> .....	24
5.1.4	Class Diagram .....	25
5.1.5	Perancangan Basis Data .....	26
5.1.6	Perancangan Algoritme .....	27
5.1.7	Perancangan User Interface .....	28
5.2	Implementasi Perangkat Lunak.....	30
5.2.1	Spesifikasi Sistem .....	30
5.2.2	Batasan Implementasi .....	32
5.2.3	Implementasi Basis Data .....	32
5.2.4	Implementasi Kode Program .....	32
5.2.5	Implementasi Antarmuka .....	45
<b>BAB 6</b>	<b>PENGUJIAN.....</b>	<b>46</b>



6.1	Pengujian <i>Black-box</i> .....	46
6.2	Pengujian Usability .....	49
6.3	Pengujian Compatibility .....	56
<b>BAB 7</b>	<b>PENUTUP</b> .....	<b>59</b>
7.1	Kesimpulan .....	59
7.2	Saran.....	59
<b>DAFTAR PUSTAKA</b>	.....	<b>60</b>

## DAFTAR GAMBAR

Gambar 3.1 Metode Pengembangan.....	14
Gambar 4.1 <i>Use Case Diagram</i> Pengguna .....	20
Gambar 5.1 Rancangan Arsitektur Sistem .....	22
Gambar 5.2 Perancangan Komponen Android dan Unity .....	23
Gambar 5.3 Sequence Diagram Menampilkan Daftar Rumah .....	24
Gambar 5.4 Sequence Diagram Menampilkan Tur Interaktif.....	25
Gambar 5.5 <i>Class Diagram</i> API Service.....	26
Gambar 5.6 <i>Class Diagram</i> Aplikasi Android .....	26
Gambar 5.7 Rancangan Basis Data dalam ERD.....	27
Gambar 5.8 Hasil Perancangan Kerangka Antarmuka Daftar Rumah dan Tur Interaktif .	29
Gambar 5.9 Hasil Perancangan Antarmuka Inventory .....	29
Gambar 5.10 Implementasi Antarmuka Home.....	45
Gambar 5.11 Implementasi Antarmuka Tur Interaktif .....	45

## DAFTAR TABEL

Tabel 4.1 Identifikasi Aktor .....	18
Tabel 4.2 Penjelasan Format Kode Kebutuhan.....	18
Tabel 4.3 Kebutuhan Fungsional Sistem .....	18
Tabel 4.4 Kebutuhan Non-Fungsional Sistem .....	19
Tabel 4.5 Use Case Scenario Menampilkan Daftar Rumah .....	20
Tabel 4.6 <i>Use Case Scenario</i> Menampilkan 3D Tour .....	21
Tabel 4.7 <i>Use Case Scenario</i> Menambah Data Rumah .....	21
Tabel 5.1 Rancangan Tabel <i>estate</i> .....	27
Tabel 5.2 Rancangan Algoritme Mendeteksi Objek Perabotan.....	27
Tabel 5.3 Spesifikasi Perangkat Keras Lingkungan Pengembangan Sistem .....	30
Tabel 5.4 Spesifikasi Perangkat Keras Lingkungan <i>Api Service</i> .....	30
Tabel 5.5 Spesifikasi Perangkat Lunak Lingkungan Pengembangan Sistem .....	31
Tabel 5.6 Spesifikasi Perangkat Lunak Lingkungan <i>Web Service</i> .....	31
Tabel 5.7 DDL Basis Data Tur Interaktif .....	32
Tabel 5.8 Kode Program HomeActivity (Android) .....	32
Tabel 5.9 Kode Program HomePresenter (Android).....	33
Tabel 5.10 Kode Program EstateAdapter (Android) .....	34
Tabel 5.11 Kode Program EstateApiService (Android) .....	35
Tabel 5.12 Kode Program Main (Unity) .....	35
Tabel 5.13 Kode Program FirstPersonController (Unity) .....	36
Tabel 5.14 Kode Program API untuk mengunduh model 3 dimensi rumah (Unity) .....	38
Tabel 5.15 Kode Program Inventory Perabotan (Unity) .....	40
Tabel 5.16 Kode Program Main.go (API Service) .....	42
Tabel 5.17 Kode Program EstateController.go (API Service) .....	43
Tabel 5.18 Kode Program Estate.go (API Service) .....	44
Tabel 6.1 Hasil Pengujian Black-box Menampilkan Daftar Rumah.....	46
Tabel 6.2 Hasil Pengujian Menampilkan Tur Interaktif .....	47

Tabel 6.3 Hasil Pengujian Black-box Menambah Daftar Rumah .....	48
Tabel 6.4 Task Scenario Untuk Pengguna .....	49
Tabel 6.5 Task Scenario untuk Admin.....	49
Tabel 6.6 Skor Skala Likert dari Setiap Pertanyaan.....	50
Tabel 6.7 Kuesioner SUPR-Qm untuk Pengguna.....	50
Tabel 6.8 Kuesioner SUPR-Qm untuk Admin .....	51
Tabel 6.9 Responden Pengujian <i>Usability</i> Pengguna Tur Interaktif .....	52
Tabel 6.10 Task Completion Rate .....	53
Tabel 6.11 Hasil Pengujian Kuesioner SUPR-Qm untuk Pengguna Tur Interaktif.....	53
Tabel 6.12 Task Completion Rate .....	54
Tabel 6.13 Hasil Pengujian Kuesioner SUPR-Qm untuk <i>Admin</i> aplikasi Tur Interaktif .....	55
Tabel 6.14 Tabel Spesifikasi Perangkat Uji 1.....	57
Tabel 6.15 Tabel Spesifikasi Perangkat Uji 2.....	57
Tabel 6.16 Tabel Spesifikasi Perangkat Uji 3.....	57
Tabel 6.17 Tabel Spesifikasi Perangkat Uji 4.....	57
Tabel 6.18 Tabel Spesifikasi Perangkat Uji 5.....	57

## DAFTAR LAMPIRAN

LAMPIRAN A DATA CATATAN PENGUNJUNG .....	62
LAMPIRAN B DEMONSTRASI PRODUK .....	64
LAMPIRAN C KUESIONER PENGGUNA APLIKASI TUR INTERAKTIF.....	67
LAMPIRAN D KUESIONER ADMIN APLIKASI TUR INTERAKTIF.....	70

# **BAB 1 PENDAHULUAN**

## **1.1 Latar Belakang**

Pemasaran merupakan hal penting dalam penjualan properti dan harus benar-benar dipertimbangkan oleh perusahaan. Dalam hal pemasaran, tentunya pihak perusahaan pengembang perumahan harus menentukan strategi pemasaran yang baik untuk menggapai konsumen dan harus mampu mendorong konsumen sehingga tertarik untuk membeli rumah yang ditawarkan. Pemasaran bisnis properti atau perumahan saat ini umumnya masih menggunakan media brosur, seperti pada perumahan Balimbingan Permai, PT. Karya Propertindo Utama. Pemasaran menggunakan brosur memiliki keterbatasan di mana konsumen atau pembeli hanya bisa melihat bentuk rumah berupa 2 dimensi yang mana gambar rumah tidak bisa dilihat secara detail dari berbagai arah. Balimbingan Permai, merupakan perumahan yang dibangun oleh perusahaan PT. Karya Propertindo Utama yang berlokasi di Kabupaten Balimbingan Pematang Siantar, Sumatera Utara. Perumahan ini terdiri dari 80 rumah dari berbagai tipe. Mulai dari tipe 36, 45, 54 dan tipe 70. Untuk harga rumah dari perumahan Balimbingan Permai ini sendiri yaitu berkisar antara 130 sampai dengan 260 juta per unitnya. Berdasarkan data dari PT. Karya Propertindo Utama, jumlah pemilik rumah di perumahan Balimbingan Permai, Pematang Siantar saat ini masih mencapai 53 penghuni. Hal tersebut berbeda dengan capaian jumlah pembelian rumah yang diharapkan oleh perusahaan PT. Karya Propertindo yaitu dengan terjualnya semua rumah. Sampai sekarang masih tersisa 27 rumah atau lahan kosong yang belum dibeli oleh konsumen. Tersisanya rumah yang masih belum terjual berdampak akan tidak didapatkannya keuntungan terhadap rumah yang sudah dibangun oleh perusahaan, serta tidak berkembangannya dan tidak terjadinya pembangunan lanjut perumahan Balimbingan Permai. Berdasarkan data beberapa tahun dari perusahaan PT. Karya Propertindo Utama, terdapat 48% dari 25 orang yang batal membeli rumah memberikan informasi yang jelas terkait kendala atau alasan batal membeli kepada pihak perusahaan PT. Karya Propertindo Utama. Jika dilakukan perhitungan terhadap potensial omset yang bisa didapatkan, maka

perusahaan PT. Karya Propertindo Utama berpotensi untuk mendapatkan keuntungan total sekitar 300 juta atau lebih. Jumlah tersebut akan didapat apabila 48% orang konsumen tersebut tidak membatalkan untuk membeli rumah. Berdasarkan keterangan informasi yang diberikan oleh perusahaan PT. Karya Propertindo, 48% konsumen yang batal beli rumah memiliki alasan/kendala tidak dapat melakukan pengamatan dikarenakan beberapa dari mereka berdomisili di luar kota dan beberapa dari mereka memiliki waktu yang sangat terbatas karena berhalangan dengan pekerjaan di kantor atau tempat kerja. Beberapa dari konsumen yang berdomisili di luar kota juga mengajukan saran untuk diberikan akses pengamatan model 3 dimensi perumahan yang akan dibeli. Dikarenakan menampilkan objek model 3 dimensi yang dibangun oleh perusahaan PT. Karya Propertindo menggunakan perangkat lunak yang memerlukan komputer dengan kebutuhan spesifikasi khusus, maka sulit bagi perusahaan untuk mengarahkan konsumen melakukan pengamatan model 3 dimensi rumah yang ingin dibeli melalui komputer.

Dengan kemajuan teknologi yang semakin maju, media kini beralih ke media virtual reality (VR) (Moura, 2017). Realitas virtual adalah tampilan gambar 3D yang dihasilkan komputer yang dibuat secara realistis menggunakan perangkat tertentu dan membuat seolah-olah pengguna terlibat langsung dengan lingkungan (Puto, 2015). Realitas virtual telah menjadi konsep interaktif yang sangat mudah digunakan dengan perkembangan teknologi seluler yang dapat memainkan peran pendukung (Pius, 2017). Dengan pemanfaatan Teknologi Virtual Reality (VR) diharapkan konsumen akan mengetahui lebih detail produk rumah yang akan dibeli. Karena menurut survei yang telah terjadi, kebanyakan konsumen ketika memesan produk, belum mengetahui bentuk asli dan nyata seperti apa, itu dikarenakan media promosi yang digunakan masih menggunakan katalog gambar dua dimensi yang hanya bisa dilihat dari satu arah saja (Fitrana, 2019). Berdasarkan data survey yang dilakukan pada 56 responden dari penelitian yang berjudul “Pengembangan Aplikasi Katalog Rumah Berbasis Virtual Reality Menggunakan Algoritma FAST”, oleh Eis Akmeliny Fitrana, didapatkan bahwa 96,2% responden mengatakan mereka tertarik apabila ada sebuah aplikasi yang memuat tentang konsep 3

dimensi pada katalog perumahan yang berbasis android. Dan 100% dari konsumen/calon pembeli menyatakan bahwa dengan adanya konsep 3 dimensi pada aplikasi katalog perumahan mempermudah mereka dalam memvisualisasikan sebuah bangunan/rumah yang nantinya akan dipilih. Berdasarkan hasil survei tersebut, aplikasi AR katalog memberikan dampak yang signifikan dalam kegiatan promosi untuk meningkatkan minat para konsumen.

Adapun penelitian sebelumnya yang mengembangkan aplikasi serupa, seperti pada penelitian pertama yang membuat model arsitektur virtual dari objek perumahan menggunakan game engine Unity 3D dengan tujuan untuk menghemat uang dan saat proses jual beli rumah. Mereka menggunakan perangkat seperti Google Cardboard dan Oculus Rift (Deaky dan Parv, 2017). Meskipun demikian, tidak didapatkan informasi terkait software yang mereka gunakan apakah bisa didapat secara gratis atau tidak. Penelitian kedua yaitu pengembangan aplikasi katalog virtual reality penjualan rumah berbasis android yang mana pada aplikasi memiliki fitur untuk mengubah warna cat dinding, pintu, dan jendela pada model rumah 3D (Husniah, L dkk, 2016). Pada penelitian ketiga yang berjudul “A Low-Cost and Lightweight 3D Interactive Real Estate-Purposed Indoor Virtual Reality Application”, oleh Kasim Ozacar, terdapat data survey yang menyimpulkan bahwa setelah melakukan tur secara virtual, responden berkeinginan untuk membeli rumah. Beberapa dari responden juga menyarankan tur dengan penggunaan kontrol arah panah memberikan pengalaman yang lebih baik daripada harus melakukan *teleporting* pada saat melakukan perpindahan lokasi. Berdasarkan ketiga penelitian tersebut penulis ingin mengajukan pembangunan aplikasi yakni penggabungan dari ketiga penelitian dengan beberapa perbaikan/modifikasi, yaitu aplikasi berbasis android yang mampu melakukan tur secara virtual menggunakan kontrol joystick pada *3d walkthrough*-nya. Dalam proses pembangunan, penulis menggunakan teknologi Unity 3D Engine untuk membangun model 3D dan tur virtual yang lingkungannya berupa objek 3 dimensi.

Teknologi Tur Interaktif berbasis Virtual Reality dari Unity 3D ini dapat menjadi solusi terhadap permasalahan konsumen yang berdomisili di luar kota dan memiliki



waktu sangat terbatas untuk melakukan pengamatan jarak jauh. Dengan adanya aplikasi ini konsumen dapat melihat objek 3 dimensi dari rumah dan dapat melakukan tur secara virtual terhadap lingkungan rumah. Aplikasi ini juga membantu developer dalam memasarkan rumah yang akan ditawarkan ke konsumen. Oleh karena itu, berdasarkan kemampuan teknologi ar dan 3D tour dari Unity 3D yang mampu menyelesaikan permasalahan di atas, maka penulis mengajukan untuk melakukan pembangunan aplikasi tur interaktif perumahan balimbingan permai PT. Karya Propertindo Utama berbasis virtual reality, dengan harapan dapat mengatasi permasalahan konsumen tersebut dan juga dapat mendukung proses promosi perusahaan kepada konsumen sehingga dapat meningkatkan potensial pembelian rumah di perumahan Balimbingan Permai, Pematang Siantar. Dalam pembangunan aplikasi, peneliti menggunakan metode pengembangan dengan model Waterfall SDLC, dimana metode tersebut terdiri dari beberapa fase yaitu perencanaan sistem, analisis sistem, perancangan sistem, implementasi sistem, dan pemeliharaan sistem. Dalam pengujiannya, peneliti menggunakan 3 metode, yaitu pengujian dengan metode Blackbox Testing (pengujian fungsional), dan Pengujian non-fungsional (pengujian usability dan compatibility). Untuk pengujian usability peneliti melakukan pengumpulan data dengan memberikan kuesioner System Usability Scale (SUS) kepada responden calon pembeli / pengunjung perumahan.

## 1.2 Identifikasi Masalah

Berdasarkan uraian latar belakang di atas, maka ada beberapa permasalahan yang dapat diidentifikasi, di antaranya sebagai berikut :

- 1) Konsumen terkendala melakukan pengamatan rumah yang akan dibeli dikarenakan konsumen berlokasi di luar kota.
- 2) Konsumen terkendala masalah pengamatan rumah yang akan dibeli dikarenakan waktu yang terbatas.
- 3) Perumahan yang terletak di kabupaten berjarak jauh dari tempat asal konsumen sehingga membutuhkan waktu yang lama bagi konsumen datang ke lokasi perumahan untuk melakukan pengamatan dan pengunjungan kembali rumah yang akan dibeli.

## 1.3 Rumusan Masalah

Permasalahan yang akan diteliti pada penelitian ini dapat dirumuskan sebagai berikut:

- 1) Bagaimana hasil analisis kebutuhan untuk sistem Aplikasi Tur Interaktif ?
- 2) Bagaimana hasil rancangan sistem Aplikasi Tur Interaktif dengan arsitektur MVP ?
- 3) Bagaimana hasil implementasi sistem Aplikasi Tur Interaktif dengan arsitektur MVP?
- 4) Bagaimana hasil pengujian *usability* dari Aplikasi Pameran dan Tur Virtual Perumahan Balimbingan Permai?
- 5) Bagaimana hasil pengujian *compatibility* dari Aplikasi Pameran dan Tur Virtual Perumahan Balimbingan Permai?

## 1.4 Tujuan Penelitian

- 1) Tujuan Umum

Dengan dibangunnya aplikasi berbasis Virtual Reality ini, maka diharapkan mampu mengatasi permasalahan pada konsumen yang memiliki keterbatasan untuk melakukan survey jarak jauh. Dengan adanya aplikasi ini diharapkan dapat menjadi solusi untuk permasalahan tersebut dan juga meningkatkan minat beli konsumen sehingga mampu menambah jumlah properti/rumah yang terjual dari perumahan Balimbingan Permai PT. Karya Propertindo Utama.

- 2) Tujuan Khusus

- a. Dibangunnya Aplikasi berbasis Virtual Reality yang memiliki kemudahan dalam menampilkan model 3 dimensi perumahan dan melakukan tur secara virtual.
- b. Mampu membantu konsumen untuk melakukan pengamatan jarak jauh dan membayangkan bentuk asli dari properti/rumah yang ingin dibeli.

## **1.5 Manfaat Penelitian**

Penelitian ini diharapkan dapat bermanfaat bagi beberapa pihak, antara lain sebagai berikut:

- 1) Konsumen
  - a. Mampu membantu konsumen untuk melakukan pengamatan jarak jauh melalui objek maya yang berbentuk 3 dimensi dan melakukan pengamatan lingkungan sekitar rumah melalui 3D tour.
  - b. Membantu konsumen untuk lebih mudah membandingkan tipe-tipe perumahan yang akan dibeli.
- 2) Peneliti
  - a. Mengetahui teknik membangun aplikasi secara umum.
  - b. Sebagai bahan referensi untuk penelitian dan pengembangan aplikasi Android selanjutnya.

## **1.6 Batasan Penelitian/Ruang Lingkup Penelitian**

Permasalahan dalam penelitian ini cukup luas, sehingga perlu dilakukan pembatasan masalah. Permasalahan yang dibahas antara lain sebagai berikut:

- 1) Pengembangan Aplikasi berbasis Virtual Reality dengan Unity 3D yang mampu menampilkan objek properti/rumah secara virtual berupa model 3 dimensi dan digunakan untuk melakukan tur lingkungan properti/rumah secara sehingga dapat melihat secara detail bagian dalam dan luar rumah.
- 2) Aplikasi digunakan hanya pada perumahan Balimbingan Permai PT. Karya Propertindo Utama.
- 3) Aplikasi diterapkan pada bidang properti/perumahan saja.

## **1.7 Sistematika Pembahasan/Laporan**

### **1.1.1. BAB I Pendahuluan**

Bab Pendahuluan menjelaskan latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, dan sistematika pembahasan.

### **1.1.2. BAB II Landasan Kepustakaan**

Bab landasan kepustakaan menjelaskan tentang kajian pustaka terkait penelitian yang telah ada seperti penelitian tentang penggunaan virtual reality di berbagai bidang seperti edukasi, pariwisata, maupun di bidang perumahan. Dengan macam-macam metode dan penggunaan metode. Penjelasan teori berisi teori-teori yang mendukung dalam pengembangan dan perancangan.

### **1.1.3. BAB III Metodologi Penelitian**

Bab ini membahas tentang tahap-tahap penelitian baik itu dalam pengumpulan data yang akan digunakan dan bagaimana cara mendapatkan datanya, analisis teoritis terhadap suatu metode dan juga proses penelitian.

### **1.1.4. BAB IV Perancangan Sistem**

Bab ini membahas perancangan sistem menggunakan metode yang telah dipilih.

### **1.1.5. BAB V Implementasi Sistem**

Bab ini menjelaskan setelah merancang sistem maka dilakukanlah implementasi terhadap metode yang telah dipilih.

### **1.1.6. BAB VI Pengujian**

Bab ini menjelaskan setelah di implementasikan maka terjadi pengujian dari tahap pengguna memulai aplikasi untuk memilih menu virtual reality atau 3D tour. Jika memilih virtual reality, aplikasi akan mulai mengenali pola yang ada pada brosur dan kemudian menghasilkan output berupa objek virtual. Kemudian jika memilih 3D Tour maka aplikasi akan menampilkan tampilan di mana pengguna bisa melakukan tur berlingkungan 3 dimensi secara virtual. Kemudian untuk hasil pengujiannya yaitu didapat dari apakah dalam menampilkan hasil, aplikasi sudah memenuhi kriteria, sesuai dengan apa yang diinginkan pengguna.

#### **1.1.7. BAB VII Penutup**

Bab ini menjelaskan hasil dari Kesimpulan yang di dapatkan dari penelitian ini dan memberikan saran agar sistem dapat lebih baik lagi untuk memenuhi kriteria yang telah ditentukan.

## **BAB 2 LANDASAN KEPUSTAKAAN**

Landasan kepustakaan berisi uraian dan pembahasan tentang teori, konsep, model, metode, atau sistem dari pustaka ilmiah, yang berkaitan dengan tema, masalah, atau pertanyaan penelitian. Dalam landasan kepustakaan terdapat landasan teori dari berbagai sumber pustaka yang terkait dengan teori dan metode yang digunakan dalam penelitian. Jika dibutuhkan sesuai dengan karakteristik penelitiannya dan syarat kecukupan khusus keminatan tertentu, bisa juga terdapat kajian pustaka yang menjelaskan secara umum penelitian-penelitian terdahulu yang berhubungan dengan topik skripsi dan menunjukkan persamaan dan perbedaan skripsi tersebut terhadap penelitian terdahulu yang dituliskan.

### **2.1 KAJIAN PUSTAKA**

Berikut adalah beberapa penelitian terkait yang berhubungan dengan topik penelitian, diantaranya adalah penelitian yang berjudul “Aplikasi Virtual Reality untuk Pengenalan Pola Satwa Menggunakan Vuforia” oleh Uning dan Annafi, “Implementasi Virtual Reality untuk Android Sebagai Media Promosi Menggunakan Unity dan Vuforia” oleh Affix Endy Abidita, “Aplikasi Virtual tour Berbasis Multimedia Interaktif Menggunakan AutoDesk 3Ds Max” oleh Akip Suhendar, dan “Interaktif Virtual Reality untuk Katalog Penjualan Rumah Berbasis Android” oleh Husniah.

Penelitian yang berjudul “Aplikasi Virtual Reality untuk Pengenalan Pola Satwa Menggunakan Vuforia” adalah aplikasi Virtual Reality pada tempat wisata kebun binatang Gembira Loka Yogyakarta yang dibuat oleh Uning Lestari dan Annafi pada tahun 2015. Aplikasi ini dikembangkan untuk membantu pengunjung mendapatkan informasi yang lebih detail mengenai satwa-satwa yang ada pada kebun binatang Gembira Loka Yogyakarta. Adapun cara penggunaannya yaitu dengan mengarahkan kamera handphone ke gambar marker dan aplikasi melakukan proses pemindaan untuk pencocokan pola yang kemudian akan ditampilkan gambar 2 dimensi dari hewan dan informasi detail tentang hewan tersebut.

Penelitian yang berjudul “Implementasi Virtual Reality untuk Android Sebagai Media Promosi Menggunakan Unity” yang dilakukan oleh Affix Endy Abidita pada tahun 2015. Aplikasi ini memiliki fitur yang menggunakan teknologi AR untuk menampilkan informasi kampus bangunan STMIK AKAKOM berbentuk 3 dimensi, serta memiliki tambahan elemen multimedia berupa audio. Aplikasi ini ditujukan untuk calon mahasiswa yang ingin mengetahui informasi STMIK AKAKOM meliputi S1 Teknik Informatika, S1 Sistem Informasi, D3 Komputerisasi Akuntansi, D3 Manajemen Informatika, D3 Teknik Komputer, Syarat Daftar dan Fasilitas.

Penelitian yang berjudul “Aplikasi Virtual tour Berbasis Multimedia Interaktif Menggunakan AutoDesk 3Ds Max” oleh Akip Suhendar pada tahun 2016, merupakan aplikasi yang dapat melakukan tur secara virtual yang berjenis virtual model 3 dimensi pada Gedung 1 Universitas Serang Raya. Yang mana dalam pengembangannya, peneliti menggunakan tool Game Engine Unity 3D dan 3D Studio Max. Aplikasi tour ini dikembangkan menggunakan first person controller yang mana membuat pengguna mampu untuk melakukan tur berjalan di dalam ruangan.

Penelitian yang berjudul “Interaktif Virtual Reality untuk Katalog Penjualan Rumah Berbasis Android” oleh Husniah pada tahun 2016. Aplikasi berbasis virtual reality dapat digunakan untuk pemasaran dalam penjualan rumah berbasis android yang mana aplikasi ini merupakan pengembangan dari penelitian serupa sebelumnya, dengan penambahan pada fitur yaitu aplikasi mampu mengubah warna cat dinding, pintu, dan jendela pada model rumah 3D.

## **2.2 Virtual Reality**

*Virtual Reality* (VR) atau realitas maya adalah teknologi yang membuat pengguna dapat berinteraksi dengan suatu lingkungan yang disimulasikan oleh computer (computer-simulated environment), suatu lingkungan sebenarnya yang ditiru atau benar-benar suatu lingkungan yang hanya ada dalam imajinasi (Sihite, 2013). Konsep VR mengacu pada system prinsip-prinsip, metode dan teknik yang digunakan untuk merancang dan menciptakan produk-produk perangkat lunak untuk digunakan oleh bantuan dari beberapa sistem computer multimedia dengan sistem perangkat khusus (Lacrama, 2007).

Lingkungan realitas maya terkini umumnya menyajikan pengalaman visual, yang ditampilkan pada sebuah layar komputer atau melalui sebuah penampil stereoskopik, tapi beberapa simulasi mengikut sertakan tambahan informasi hasil pengindraan, seperti suara melalui speaker atau headphone. Beberapa sistem haptic canggih sekarang meliputi informasi sentuh, biasanya dikenal sebagai umpan balik kekuatan pada aplikasi berjudi dan medis (Sihite, 2013).

## **2.3 Unity 3D**

Unity 3D merupakan sebuah tools yang terintegrasi untuk membuat bentuk obyek 3D pada video games atau untuk konteks interaktif lain seperti Visualisasi Arsitektur atau animasi 3D real-time. Lingkungan dari pengembangan Unity 3D berjalan pada Microsoft Windows dan Mac Os X, serta aplikasi yang dibuat oleh Unity 3D dapat berjalan pada Windows, Mac, Xbox 360, Playstation 3, Wii, iPad, iPhone dan tidak ketinggalan pada platform Android. Unity juga dapat membuat game berbasis browser yang menggunakan Unity web player plugin, yang dapat bekerja pada Mac dan Windows, tapi tidak pada Linux. (Mutia dan Djuniadi ,2015).



## **2.4 Android**

Android merupakan sebuah sistem operasi yang digunakan pada telepon seluler yang dikembangkan berbasis linux (Nazaruddin, 2012). Android pertama kali dibuat oleh Android, Inc. dan pada tahun 2005 Google pun resmi membeli Android. Android menjadi salah satu sistem operasi yang sangat populer saat ini dan sudah banyak perangkat seluler yang menggunakan sistem operasi Android. Dalam membuat aplikasi yang berjalan pada sistem operasi Android, pengembang aplikasi saat ini dapat menuliskan kode dengan menggunakan bahasa Kotlin, Java, dan C++ (Developers, 2019).

## **2.5 Pengujian Fungsional**

Pengujian fungsional adalah pengujian yang bertujuan untuk memberikan hasil evaluasi berupa kepastian bahwa aplikasi atau sistem yang dikembangkan telah memenuhi seluruh kebutuhan aplikasi atau sistem. Pengujian ini bukan berfokus pada keberhasilan proses sistem, melainkan berfokus pada hasil dari proses sistem. Salah satu metode dalam pengujian ini yaitu seperti black box.

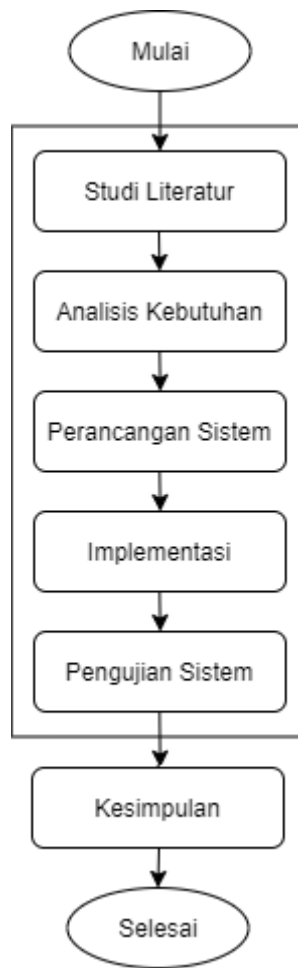
## **2.6 Pengujian *Usability***

Usability merupakan suatu arti yang berasal dari kata usable, artinya dapat digunakan dengan baik. Suatu yang dihasilkan dapat dikatakan berguna atau baik digunakan apabila kemungkinan terjadinya kegagalan saat penggunaan kecil atau minimal sehingga harus mampu memberi kepuasan dan manfaat kepada pengguna (Jeff Rubin, 2008). Sedangkan menurut ISO 9241:11 (1998), usability yaitu “sejauh mana suatu produk dapat digunakan oleh pengguna tertentu untuk mencapai target yang ditetapkan dengan efektifitas, efisiensi, dan mencapai kepuasan penggunaan dalam konteks tertentu”.

### **BAB 3 METODOLOGI PENELITIAN**

Metodologi penelitian menjelaskan mengenai metode pelaksanaan penelitian yang digunakan dalam merancang dan mengembangkan aplikasi tur interaktif pada perumahan Balimbingan Permai berbasis *Virtual Reality*. Pada bab ini akan dijelaskan terkait tahapan-tahapan yang dilakukan dalam pengimplementasian metode pengembangan aplikasi. Dalam penelitian ini, peneliti menggunakan metode dengan model pengembangan sistem yang mengadaptasi model Waterfall SDLC.

Pola pengembangan waterfall merupakan pola pengembangan perangkat lunak yang bersifat searah, di mana proses pengembangan hanya akan berjalan maju dan tidak dapat kembali ke tahapan sebelumnya. Sehingga pola pengembangan ini cocok diterapkan apabila pengembang telah mampu memahami permasalahan serta kebutuhan dari aplikasi yang akan dikembangkan serta dapat dipastikan bahwa kebutuhan-kebutuhan dari aplikasi yang akan dikembangkan serta dapat dipastikan bahwa kebutuhan-kebutuhan yang didapatkan dari proses elisitasi tidak akan mengalami perubahan secara signifikan baik dalam masa pengembangan perangkat lunak maupun di masa yang akan datang (Sommerville, 2011). Pola pengembangan waterfall diterapkan karena aplikasi yang hendak dikembangkan dalam penelitian ini memiliki daftar kebutuhan yang konsisten atau jarang berubah. Metode pengembangan waterfall memiliki 7 fase yang terdiri dari studi literatur, analisis kebutuhan, perancangan sistem, implementasi, pengujian sistem, dan kesimpulan seperti pada diagram di bawah ini.



**Gambar 3.1 Metode Pengembangan**

### **3.1 Studi Literatur**

Studi literatur menjelaskan dasar-dasar teori yang disusun berdasarkan referensi yang diperoleh dari artikel, buku, jurnal serta penelitian-penelitian terkait yang sejenis. Studi literatur digunakan untuk penunjang dan pendukung dalam penulisan skripsi, serta sebagai pengetahuan tambahan dalam melakukan analisis kebutuhan, perancangan, implementasi serta pengujian terhadap suatu perangkat lunak sebagai penelitian. Selain itu, studi literatur dilakukan untuk mencegah plagiasi dari karya/penelitian yang telah ada sebelumnya.

### **3.2 Analisis Kebutuhan Sistem**

Analisis kebutuhan sistem berguna untuk mendefinisikan kebutuhan yang diperlukan dalam pengembangan perangkat lunak. Metode yang digunakan dalam analisis kebutuhan perangkat lunak pada penelitian ini adalah Object Oriented Analysis (OOA). Setelah melakukan analisis kebutuhan, kemudian kebutuhan dimodelkan dengan menggunakan bahasa pemodelan UML (Unified Modeling Language) yang meliputi Use Case Diagram dan Use Case Scenario. Use Case Diagram digunakan untuk mendeskripsikan kebutuhan fungsional sistem/perangkat lunak dari sisi pengguna. Dan Use Case Scenario digunakan untuk menjelaskan lebih detail dari tiap-tiap case pada use case diagram.

### **3.3 Perancangan Sistem**

Perancangan sistem digunakan untuk merancang tiap-tiap kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan. Tahap perancangan digunakan sebagai bahan perancangan/pembentukan langkah kerja yang meliputi seluruh isi sistem yang akan dikembangkan. Pada tahap ini terdapat perancangan arsitektur, perancangan komponen, perancangan data dan perancangan antarmuka. Tiap-tiap kebutuhan yang telah dijabarkan secara spesifik, dirancang pada tahap perancangan dengan menggunakan diagram Sequence. Sequence diagram menjelaskan alur dari sistem berdasarkan fungsi yang digambarkan dengan garis waktu. Selain Sequence diagram, adapun class diagram yang berguna menggambarkan hubungan antara tiap-tiap kelas yang saling terkait satu sama lain pada sebuah perangkat lunak yang dikembangkan

### **3.4 Implementasi Sistem**

Implementasi sistem merupakan tahapan yang dilakukan ketika semua kebutuhan yang telah dijelaskan selesai dirancang. Implementasi yang dilakukan meliputi implementasi basis data menggunakan DBMS MySQL, implementasi logika program dan implementasi antarmuka. Implementasi basis data merupakan perancangan basis data yang dilakukan dengan mendefinisikan tabel yang digunakan dengan DDL (Data

Definition Language). Implementasi logika program menggunakan framework Android SDK dengan bahasa pemrograman Java.

### **3.5 Pengujian Sistem**

Pengujian sistem dilakukan untuk menguji sistem yang dikembangkan, apakah semua kebutuhan yang telah dijabarkan telah berhasil diimplementasikan pada sistem atau tidak. Pengujian sistem ini juga digunakan untuk mencari kesalahan/kesalahan yang terdapat pada perangkat lunak yang sedang dikembangkan. Pada pengujian sistem, menggunakan pengujian whitebox untuk menguji struktur perangkat lunak bagian dalam (source code) dan menggunakan pengujian blackbox untuk menguji struktur perangkat lunak bagian luar (interface).

### **3.6 Kesimpulan dan Saran**

Kesimpulan didapat setelah semua tahapan metodologi mulai dari studi literatur, analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian sistem, dan analisis hasil berhasil dilaksanakan dengan teratur. Kesimpulan yang valid diperoleh ketika perangkat lunak yang dikembangkan telah lolos tahap pengujian. Selain kesimpulan, adapun saran yang berguna untuk memperbaiki apa-apa saja yang kurang dalam pengembangan aplikasi tur interaktif berbasis virtual reality. Saran berguna sebagai pengembangan lebih lanjut untuk sistem kedepannya.

## BAB 4 REKAYASA KEBUTUHAN

Bagian ini menjelaskan mengenai fase analisis kebutuhan dari pembangunan aplikasi tur interaktif Perumahan Balimbingan Permai PT. Karya Propertindo Utama berbasis *virtual reality*. Pada bab ini akan dijelaskan gambaran umum sistem yang akan dibangun, aktor yang akan menggunakan aplikasi, storyboard, kebutuhan dalam membangun sistem, use case diagram dan use case scenario.

### 4.1. Gambaran Umum Aplikasi

Gambaran umum aplikasi menjelaskan tentang aplikasi Pameran dan Tur Virtual Perumahan Balimbingan Permai dan cara penggunaannya.

#### 4.1.1. Deskripsi

Aplikasi yang dikembangkan pada penelitian ini adalah aplikasi untuk pengguna *home seeker* yang mana pengguna dapat menggunakan aplikasi untuk mengamati bentuk 3 dimensi rumah dan melakukan *3d tour* secara virtual. Aplikasi ini dapat menampilkan objek 3 dimensi secara virtual melalui *virtual reality* yang dikembangkan menggunakan *Unity 3D Engine*.

#### 4.1.2. Cara Penggunaan

Untuk menggunakan aplikasi Pameran dan Tur Virtual ini, dibutuhkan aplikasi yang terinstal pada *smartphone* Android. Kemudian pengguna melakukan pencarian rumah yang diinginkan dan memilih salah satu dari hasil pencarian. Setelahnya akan terdapat opsi untuk pengguna dapat menampilkan tur interaktif secara virtual.

### 4.2. Analisis Kebutuhan Sistem

Analisis kebutuhan dilakukan dengan cara studi pustaka terkait sistem yang sudah ada. Sistem yang dijadikan acuan ialah aplikasi Google Maps Popular Times dan sistem yang dikembangkan oleh Lesani dan Miranda-Moreno (2018). Analisis kebutuhan sistem merupakan analisis kebutuhan dasar sebuah sistem yang harus tersedia maupun kebutuhan pendukung sistem. Analisis kebutuhan sistem dapat dibagi menjadi dua, yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Pada penelitian ini, kode fungsi digunakan untuk merepresentasikan kebutuhan fungsional dan non-fungsional tertentu dengan kode SRS-TIVR-F-N untuk kode kebutuhan fungsional dan SRS-TIVR-NF-

N untuk kebutuhan non-fungsional, dimana N adalah angka dari 1 sampai jumlah kebutuhan masing-masing.

#### 4.2.1 Identifikasi Aktor

Aktor adalah seseorang atau sistem yang dapat melakukan interaksi dengan sistem. Pada penelitian ini, aktor yang ada deskripsikan pada Tabel 4.1.

**Tabel 4.1 Identifikasi Aktor**

Aktor	Deskripsi
Pengguna	Aktor yang dapat melakukan pengamatan bentuk 3 dimensi dari rumah dan melakukan tur secara interaktif. Pengguna berupa pengguna umum yang dapat menggunakan aplikasi secara langsung.
Admin	Aktor yang dapat melakukan penambahan data rumah yang akan tampil di halaman Daftar Rumah pada aplikasi di sisi pengguna.

#### 4.2.2 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang terkait dengan fungsi produk (Sukamto & Shalahuddin, 2016). Kebutuhan fungsional pada pengembangan lanjut ini dapat dilihat pada Tabel 4.3

**Tabel 4.2 Penjelasan Format Kode Kebutuhan**

Kode Format	Deskripsi
SRS	SRS merupakan inisial dari <i>Software Requirement Specification</i> .
TIVR	TIVR merupakan inisial untuk aplikasi Tur Interaktif berbasis Virtual Reality
F/NF	F merupakan kode penunjuk untuk kebutuhan fungsional sedangkan NF untuk kebutuhan non-fungsional
XXX	XXX menyatakan nomor kebutuhan

**Tabel 4.3 Kebutuhan Fungsional Sistem**

No	Kode Fungsi	Nama Fungsi	Deskripsi
1	SRS-TIVR-F-1	Menampilkan daftar rumah	Sistem dapat menampilkan daftar rumah.
2	SRS-TIVR-F-2	Menampilkan Tur Interaktif	Sistem dapat menampilkan Tur Interaktif berbasis Virtual Reality

			dengan menampilkan objek 3 dimensi rumah.
3	SRS-TIVR-F-3	Menambah data rumah	Sistem dapat menambah data rumah

#### 4.2.3 Kebutuhan Non Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang tidak langsung berhubungan dengan sistem namun menjadi penunjang bagi kebutuhan fungsional (Summerville, 2011). Kebutuhan non-fungsional dapat dilihat pada Tabel 4.4.

**Tabel 4.4 Kebutuhan Non-Fungsional Sistem**

No	Kode Fungsi	Nama Fungsi	Deskripsi
1	SRS-TIVR-NF-1	<i>Usability</i>	<i>Usability</i> menjelaskan tentang kepuasan dan kemudahan pengguna terhadap pemakaian aplikasi. Usabilitas diangkat karena tujuan dari dikembangkannya aplikasi adalah untuk meningkatkan daya beli <i>home seeker</i> , maka diperlukan kebutuhan akan usabilitas yang menjamin kepuasan dan kemudahan pengguna dalam menggunakan aplikasi.
2	SRS-TIVR-NF-2	<i>Compatibility</i>	<i>Compatibility</i> menjelaskan tentang banyaknya jenis perangkat atau versi teknologi yang bisa digunakan untuk menjalankan aplikasi.

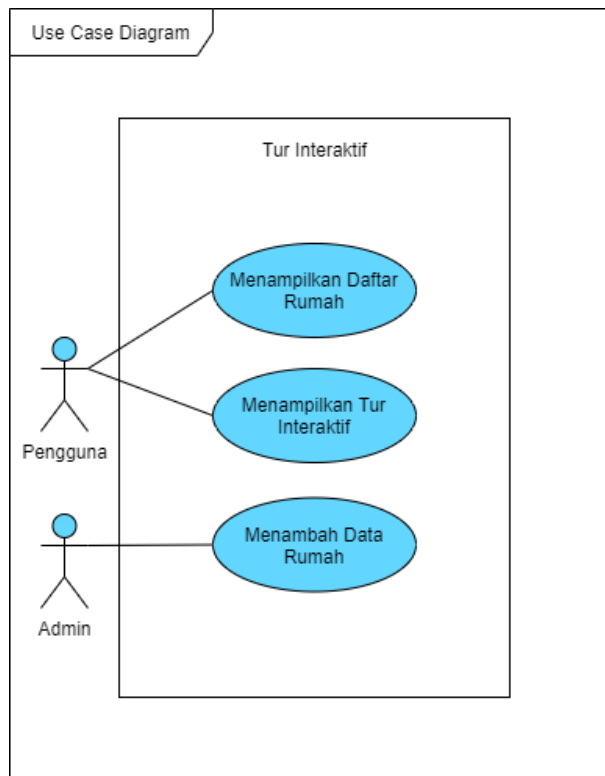
#### 4.3. Pemodelan Kebutuhan

Use case mendefinisikan tentang bagaimana aktor berinteraksi dengan sistem pada kondisi tertentu. Use case menggambarkan perangkat lunak dari sudut pandang aktor (Pressman, 2010).

##### 4.3.1. Use Case Diagram

. Use case diagram adalah diagram yang dapat menunjukkan actor dan hal yang dapat dilakukan oleh actor kepada sistem untuk melakukan sesuatu pada sistem. Actor dapat berupa pengguna maupun sistem eksternal yang dapat berinteraksi kepada sistem (Sukanto & Shalahuddin, 2016).





**Gambar 4.1 Use Case Diagram Pengguna**

#### 4.3.2. Use case Scenario

*Use Case Scenario* adalah penjelasan yang detail terhadap setiap use case yang sudah dipetakan pada *use case diagram*. *Use Case Scenario* digambarkan dalam bentuk table yang terdiri dari nama use case, kode kebutuhan, aktor, tujuan, pre condition, main flow, post condition, dan alternative flow. Bagian pre condition mendeskripsikan tentang kondisi sistem yang harus dipenuhi sebelum masuk ke bagian main flow.

**Tabel 4.5 Use Case Scenario Menampilkan Daftar Rumah**

Nama Use Case	Menampilkan Daftar Rumah
Kode Kebutuhan Terkait	SRS-TIVR-F-1
Aktor	Pengguna
Tujuan	Menampilkan daftar rumah berdasarkan filter yang diberikan oleh pengguna
Pre Condition	Aktor telah berada pada halaman utama
Main Flow	1. Pengguna membuka halaman utama 2. Sistem menampilkan daftar rumah berdasarkan filter yang diberikan oleh pengguna
Post Condition	Sistem menampilkan daftar rumah
Alternative Flow	-

**Tabel 4.6 Use Case Scenario Menampilkan 3D Tour**

Nama <i>Use Case</i>	Menampilkan 3D Tour
Kode Kebutuhan Terkait	SRS-TIVR-F-2
Aktor	Pengguna
Tujuan	Menampilkan Tur Interaktif berbasis Virtual Reality dengan menampilkan objek 3 dimensi rumah.
Pre Condition	Aktor telah berada pada halaman utama
Main Flow	<ol style="list-style-type: none"> <li>1. Pengguna berada di halaman utama</li> <li>2. Pengguna melakukan tap pada salah satu item rumah</li> <li>3. Sistem menjalankan Tur Interaktif dari objek 3 dimensi rumah.</li> </ol>
Post Condition	Sistem menampilkan Tur Interaktif <i>Unity</i>
Alternative Flow	-

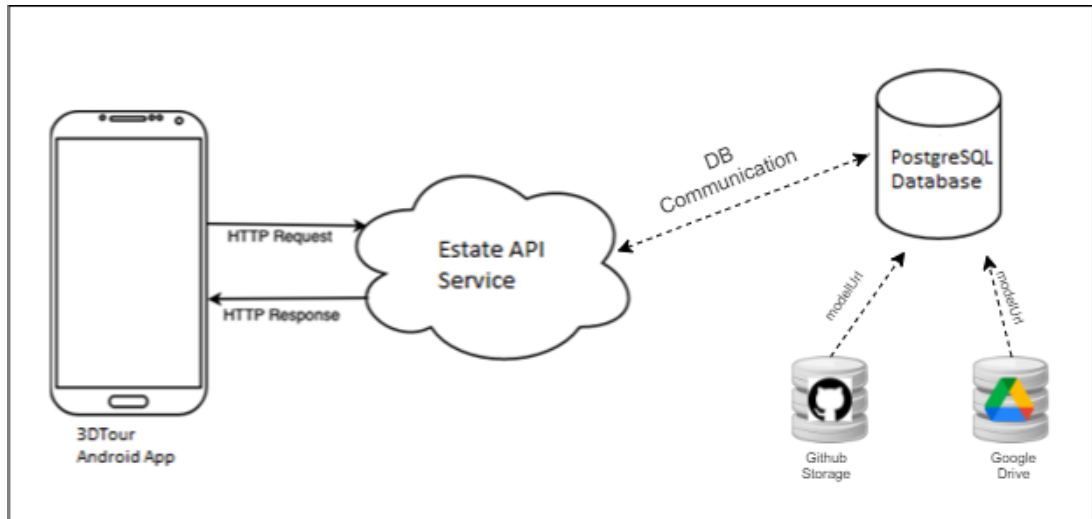
**Tabel 4.7 Use Case Scenario Menambah Data Rumah**

Nama <i>Use Case</i>	Menambah Data Rumah
Kode Kebutuhan Terkait	SRS-TIVR-F-3
Aktor	Admin
Tujuan	Menambah data rumah
Pre Condition	Aktor telah berada pada halaman utama
Main Flow	<ol style="list-style-type: none"> <li>1. Admin membuka halaman utama</li> <li>2. Sistem menampilkan form tambah data rumah</li> <li>3. Admin melakukan pengisian data</li> <li>4. Admin melakukan proses submit dan sistem memproses <i>request</i></li> </ol>
Post Condition	Sistem menampilkan pesan berhasil menambah data
Alternative Flow	-

## BAB 5 PERANCANGAN DAN IMPLEMENTASI

### 5.1 Perancangan

#### 5.1.1 Perancangan Arsitektur Sistem



**Gambar 5.1 Rancangan Arsitektur Sistem**

Rancangan arsitektur sistem yang digunakan pada penelitian ini digambarkan pada Gambar 5.1. Komunikasi yang digunakan untuk menghubungkan aplikasi 3DTour dengan web service adalah HTTP. Adapun proses pertukaran data menggunakan format JSON. Web service berfungsi untuk menyimpan data pada database serta mengagregasi data yang telah dikirim oleh aplikasi 3DTour.

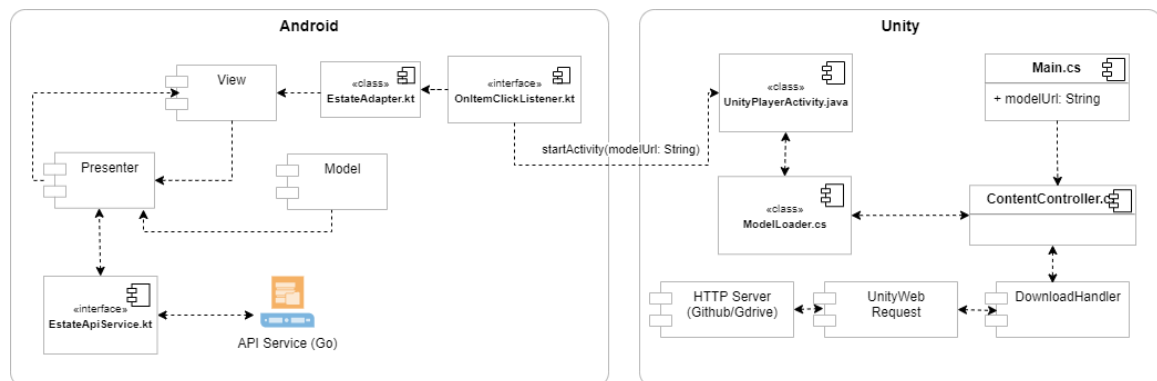
Aplikasi 3DTour dirancang dengan menggunakan pola arsitektur MVP. Berdasarkan pola arsitektur tersebut, setiap halaman memiliki dua buah class utama, yaitu View dan Presenter, yang saling mereferensi satu sama lain melalui interface. Contohnya adalah halaman main yang memiliki interface `MainViewContract` yang diimplementasikan oleh class `MainActivity` dan `MainPresenterContract` yang diimplementasikan oleh class `MainPresenter`. Adapun masing-masing class (`MainActivity` dan `MainPresenter`) saling mereferensi satu sama lain dengan bersandar pada interface yang diimplementasi. Sebagai contoh, `MainActivity` yang memiliki referensi kepada `MainPresenter` dengan memiliki sebuah instance variable bertipe `HomePresenterContract` dan sebaliknya.

Selanjutnya, sistem Estate API Service dirancang tanpa menggunakan arsitektur khusus tetapi tetap terdapat pemisahan konsentrasi antara controller, model, HTTP

route handler, dan konektivitas ke database. Controller berfungsi untuk menjalankan pekerjaan spesifik berdasarkan request yang dikirim dari user. Model merepresentasikan entitas yang terlibat dalam sistem. HTTP route handler berperan sebagai pemroses rute HTTP request yang datang dan menghubungkannya ke controller yang sesuai. Selanjutnya, objek database yang mengabstraksikan operasi-operasi terhadap database.

### 5.1.2 Perancangan Komponen

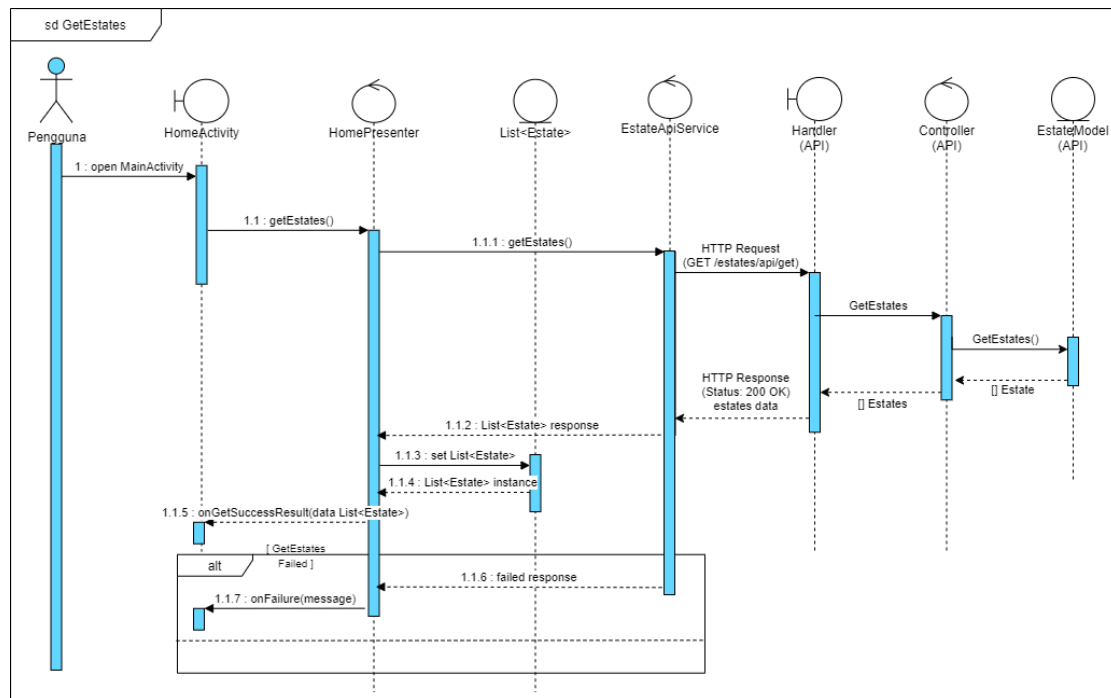
Rekayasa perangkat lunak berbasis komponen (CBSE), juga disebut pengembangan berbasis komponen (CBD), adalah cabang rekayasa perangkat lunak yang menekankan pemisahan perhatian sehubungan dengan fungsionalitas luas yang tersedia di seluruh sistem perangkat lunak yang diberikan. Ini adalah pendekatan berbasis penggunaan kembali untuk mendefinisikan, mengimplementasikan, dan menyusun komponen independen yang digabungkan secara longgar ke dalam sistem. Diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem. Berikut diagram komponen dari sistem Aplikasi Tur Interaktif.



**Gambar 5.2 Perancangan Komponen Android dan Unity**

### 5.1.3 Sequence Diagram

Sequence Diagram dibuat untuk menjelaskan interaksi kelas-kelas yang terlibat dalam satu alur kerja. Terdapat 2 buah sequence diagram yang dibuat untuk menjelaskan alur pada aplikasi Android dan 1 buah sequence diagram untuk menjelaskan alur pada api service.

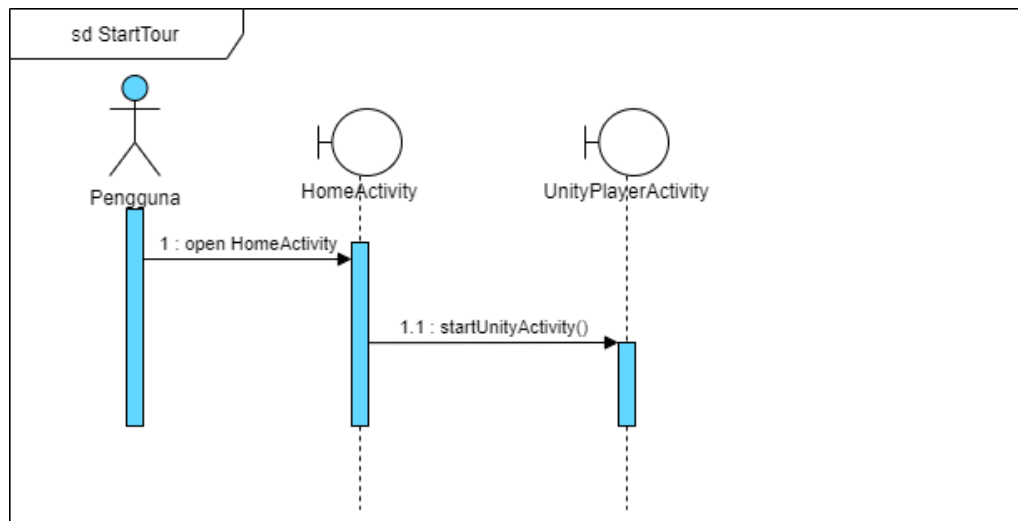


**Gambar 5.3 Sequence Diagram Menampilkan Daftar Rumah**

Gambar 5.4 menggambarkan alur untuk melihat daftar rumah. Saat pengguna membuka aplikasi, akan dipanggil fungsi `getEstates()` pada `HomePresenter` diikuti oleh pemanggilan fungsi `getEstates()` pada `ApiService`. Aplikasi Tur memanggil endpoint `"/estates/api/get"`. Selanjutnya, kelas `Handler` yang menangani *request* yang masuk akan memanggil fungsi `GetEstates` dari kelas `Controller`. Fungsi tersebut akan memanggil fungsi `GetEstates()` dari kelas `EstateModel`. Fungsi `GetEstates()` dari kelas `Model` ini akan mengembalikan data berupa `list Estate`.

Selanjutnya `API Service` mengembalikan data respon berupa daftar rumah berbentuk `json`. Yang mana bentuk `json` diubah secara otomatis menjadi objek dengan tipe `List<Estate>`. Kemudian setelah berhasil dari `HomePresenter` memanggil callback `onGetSuccessResult()` dengan mengembalikan objek data bertipe `List<Estate>` yang

diinisialisasi di kelas HomeActivity. Pada method onGetSuccessResult() ini dilakukan populasi data ke dalam recyclerview.



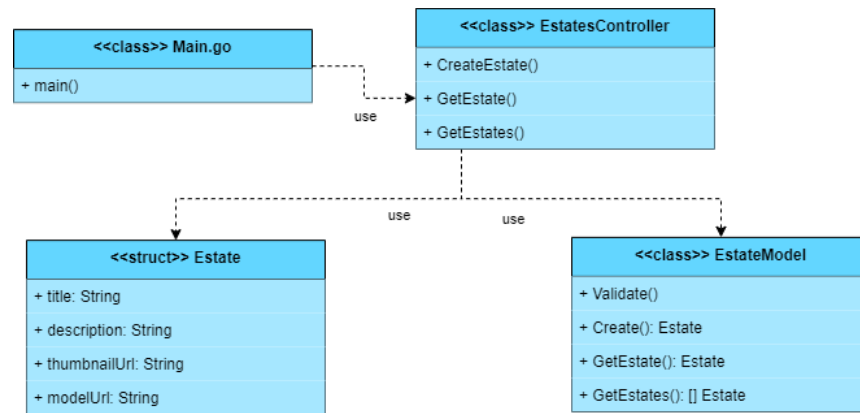
**Gambar 5.4 Sequence Diagram Menampilkan Tur Interaktif**

Gambar 5.5 menggambarkan alur menampilkan tur interaktif. Pengguna berada HomeActivity dengan pre-condition data daftar rumah telah ditampilkan pada halaman HomeActivity. Kemudian user melakukan tap ke salah satu item data daftar rumah. Kemudian method startUnityActivity dipanggil dari HomeActivity dengan menyisipkan data berupa modelUrl dengan objek Intent saat memanggil method startActivity(). Kemudian sistem menampilkan unity activity yang dikembangkan menggunakan unity 3D Engine dan di-embed di projek Android Studio.

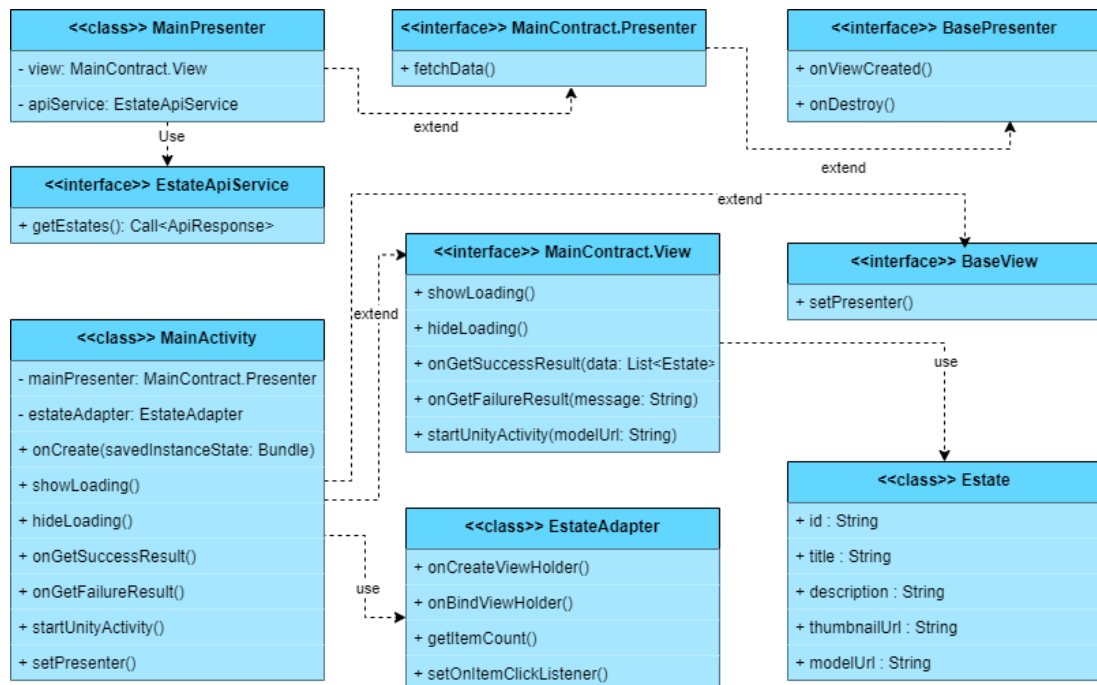
#### 5.1.4 Class Diagram

*Class diagram* bertujuan untuk memodelkan struktur dengan mendeskripsikan kelas-kelas beserta atribut, operasi, dan hubungan antar kelasnya terhadap sistem yang sedang dirancang. Pada bagian ini, terdapat dua *class diagram* yaitu *class diagram* untuk *web service* yang ada pada Gambar 5.7 dan *class diagram* untuk aplikasi berbasis Android seperti pada Gambar 5.8.

Pada *class diagram* untuk aplikasi berbasis Android, terdapat pemisahan antara View dan Presenter. Hal ini merupakan dampak dari penggunaan pola arsitektur MVP. Kelas Activity yang berperan sebagai View dihubungkan dengan kelas Presenter yang berisi mengenai *business logic*.



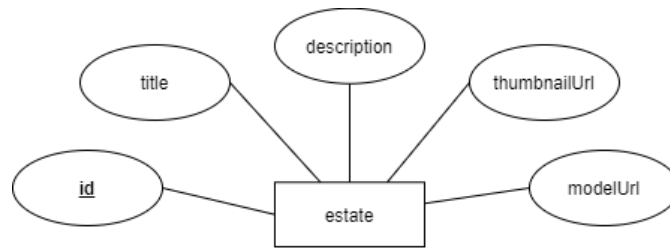
**Gambar 5.5 Class Diagram API Service**



**Gambar 5.6 Class Diagram Aplikasi Android**

### 5.1.5 Perancangan Basis Data

Perancangan basis data dilakukan untuk menentukan bagaimana data diorganisasi dalam *database*. Basis data dirancang dalam bentuk *entity-relationship diagram* (ERD) seperti yang tergambar pada Gambar 5.9. Kemudian, hasil ERD akan dinormalisasi satu table yang didefinisikan pada Tabel 5.1.



**Gambar 5.7 Rancangan Basis Data dalam ERD**

**Tabel 5.1 Rancangan Tabel *estate***

No.	Nama Kolom	Tipe Data	Deskripsi
1	id	Int	ID rumah
2	title	String	Judul rumah
3	description	String	Deskripsi rumah
4	thumbnailUrl	String	Alamat situs gambar yang akan ditampilkan pada <i>item</i> daftar rumah.
5	modelUrl	String	Alamat situs model 3 dimensi yang akan ditampilkan pada aplikasi unity yang di- <i>embed</i> di aplikasi Android.

### 5.1.6 Perancangan Algoritme

Perancangan algoritme dilakukan untuk memberikan gambaran pada tahap implementasi terhadap alur kerja sistem. Algoritme utama yang dijelaskan pada Tabel 5.2 adalah algoritme untuk mengunduh model 3 dimensi rumah. Kemudian algoritme selanjutnya dijelaskan pada Tabel 5.3 untuk mendeteksi objek perabotan yang berada di dekat *player* pada aplikasi unity yang dikembangkan.

**Tabel 5.2 Rancangan Algoritme Menampilkan Daftar Rumah**

1	<code>view = initiateView()</code>
2	<code>presenter = new Presenter(view)</code>
3	<code>dataResult: Array = presenter.getDataFromApiService()</code>
4	<code>recyclerView.setAdapter(new Adapter())</code>
5	<code>recyclerView.adapter.setData ( dataResult )</code>
6	<code>recyclerView.adapter.refreshView()</code>



**Tabel 5.3 Rancangan Algoritme Menampilkan Tur Interaktif**

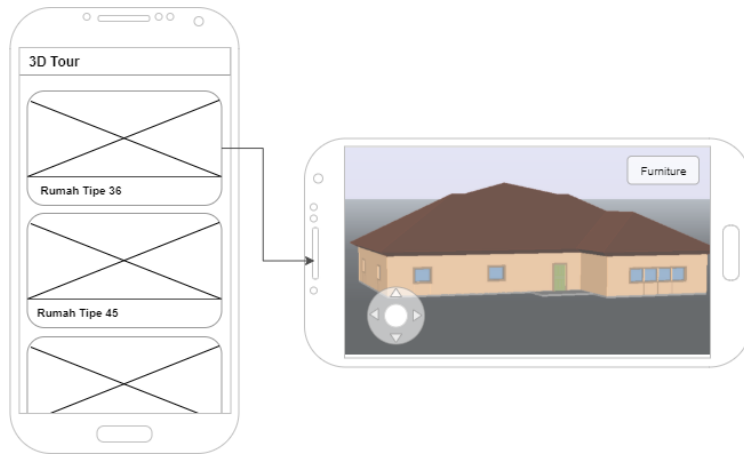
1	<code>view = initiateView()</code>
2	<code>presenter = new Presenter(view)</code>
3	<code>dataResult: Array = presenter.getDataFromApiService()</code>
4	<code>recyclerView.adapter.setOnItemClickListener(new Listener())</code>
5	
6	<code>// onItemClick</code>
7	<code>Intent = Intent(UnityActivityPlayer)</code>
8	<code>intent.putData("modelUrl", itemData.modelUrl)</code>
9	<code>startActivity(intent)</code>

**Tabel 5.4 Rancangan Algoritme Menambah Data Rumah**

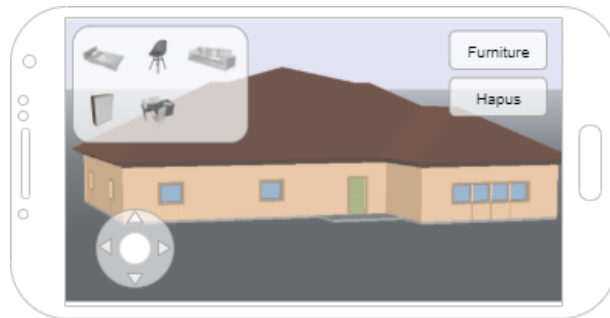
1	<code>view = initiateView()</code>
2	<code>presenter = new Presenter(view)</code>
3	
4	<code>submitListener = onSubmit ( (data) -&gt; {</code>
5	<code>    presenter.sendInsertDataToApiService(data, () -&gt; {</code>
6	<code>        presenter.onResultSuccess()</code>
7	<code>    })</code>
8	<code>});</code>
9	
10	<code>view.setOnClickSubmit(submitListener)</code>
11	<code>view.setOnResultSuccess( () -&gt; {</code>
12	<code>    showToast("Success")</code>
13	<code>});</code>

### 5.1.7 Perancangan User Interface

Perancangan antarmuka bertujuan untuk memberi gambaran dasar terkait sistem yang akan digunakan oleh penggunanya. Hasil perancangan antarmuka dituangkan pada Gambar 5.10. Rancangan tersebut terdiri dari 2 halaman, yaitu halaman *home* sebagai halaman utama dan tur virtual.



**Gambar 5.8 Hasil Perancangan Kerangka Antarmuka Daftar Rumah dan Tur Interaktif**



**Gambar 5.9 Hasil Perancangan Antarmuka Inventory**

## 5.2 Implementasi Perangkat Lunak

Implementasi aplikasi Android Tur Interaktif menggunakan pola arsitektur MVP yang mana Activity berperan sebagai View dan terhubung dengan *class* Presenter yang berisi logika-logika alur. Kemudian, terdapat kelas Model yang berfungsi sebagai representasi entitas-entitas yang digunakan dalam sistem. Kemudian, program *api service* diimplementasikan dengan bahasa Go dan program yang dikembangkan menggunakan Unity 3D Engine menggunakan C#.

### 5.2.1 Spesifikasi Sistem

Bagian ini menjabarkan tentang lingkungan pengembangan sistem dan lingkungan di mana *api service* berjalan.

#### 5.2.1.1 Spesifikasi Perangkat Keras

Sistem yang berupa aplikasi berbasis Android dan *web service* dikembangkan di lingkungan yang dijelaskan pada **Error! Reference source not found..** Adapun **Error! reference source not found.** menjabarkan spesifikasi perangkat keras di mana *web service* berjalan.

**Tabel 5.5 Spesifikasi Perangkat Keras Lingkungan Pengembangan Sistem**

Perangkat	Asus Vivobook S14 S410 UN
Prosesor	2,7 GHz Dual-Core Intel Core i5
RAM	8 GB 1867 MHz DDR3
GPU	Nvidia MX 150
Ruang Penyimpanan	256 GB
OS	Windows 10 Home Single language

**Tabel 5.6 Spesifikasi Perangkat Keras Lingkungan *Api Service***

Prosesor	1 vCPU
----------	--------

RAM	1 GB
Ruang Penyimpanan	20 GB SSD
Lokasi	Singapura

#### 5.2.1.2 Spesifikasi Perangkat Lunak

Daftar perangkat lunak yang digunakan pada pengembangan sistem RestoCrowd dijabarkan pada

. Adapun lingkungan di mana *web service* berjalan dijabarkan pada **Error! Reference source not found..**

**Tabel 5.7 Spesifikasi Perangkat Lunak Lingkungan Pengembangan Sistem**

Pengolah Kata	Microsoft Word for Mac 2016 Ver 16.16.16.
<i>Editor</i> Pemrograman	<ol style="list-style-type: none"> <li>1. Android Studio 3.5</li> <li>2. Visual Studio Code</li> <li>3. Unity 3D Engine</li> </ol>
Bahasa Pemrograman	<ol style="list-style-type: none"> <li>1. Kotlin 1.3.61</li> <li>2. Go 1.12.5</li> <li>3. C#</li> </ol>

**Tabel 5.8 Spesifikasi Perangkat Lunak Lingkungan *Web Service***

<i>Host OS</i>	Debian GNU/Linux 9.11 (Stretch)
DBMS	MariaDB 10.2.27
<i>Container Engine</i>	Docker 19.03.4 Community Edition
Docker Image OS	Alpine Linux 3.10.3

### 5.2.2 Batasan Implementasi

Terdapat hal yang menjadi batasan implementasi pada pengembangan sistem TurInteraktif ini, yaitu aplikasi TurInteraktif ini hanya dapat berjalan pada *smartphone* berbasis Android dengan versi 6.0 ke atas.

### 5.2.3 Implementasi Basis Data

**Tabel 5.9 DDL Basis Data Tur Interaktif**

1	CREATE TABLE `estates` (
2	`id` bigint(20) unsigned NOT NULL AUTO_INCREMENT,
3	`title` varchar(255) NOT NULL,
4	`description` varchar(255) NOT NULL,
5	`thumbnailUrl` varchar(255) NOT NULL,
6	`modelUrl` varchar(255) NOT NULL,
7	PRIMARY KEY (`id`)
8	) ENGINE=InnoDB DEFAULT CHARSET=utf8;

### 5.2.4 Implementasi Kode Program

Aplikasi berbasis Android Tur Interaktif diprogram dengan menggunakan bahasa Kotlin, C# pada pengembangan dengan *Unity 3D Engine*, dan Go pada pengembangan *API Service*-nya. Adapun kode program yang di implementasikan untuk pengembangan aplikasi Androidnya dijelaskan pada Tabel 5.4 hingga Tabel 5.7.

Program tur interaktif yang dikembangkan menggunakan Unity 3D Engine diprogram dengan menggunakan bahasa C#. Implementasi kode C# dijelaskan pada tabel 5.8 hingga Tabel 5.11. Program *web service* diprogram dengan menggunakan bahasa Go. Implementasi kode program Go dijelaskan pada Tabel 5.12 hingga Tabel 5.14.

**Tabel 5.10 Kode Program HomeActivity (Android)**

1	class HomeActivity : AppCompatActivity(), MainContract.View {
2	
3	private lateinit var materialDialog: MaterialDialog
4	private lateinit var mainPresenter: MainContract.Presenter
5	private lateinit var estateAdapter: EstateAdapter
6	override fun onCreate(savedInstanceState: Bundle?) {
7	super.onCreate(savedInstanceState)
8	setContentView(R.layout.activity_main)
9	
10	setPresenter(
11	MainPresenter(
12	this,
13	EstateApplication.getDependencyInjection(this).getApiService()
14	)

15	)
16	
17	val layoutManager = LinearLayoutManager(this)
18	layoutManager.orientation = LinearLayoutManager.VERTICAL
19	recyclerView.layoutManager = layoutManager
20	
21	estateAdapter = EstateAdapter(baseContext)
22	estateAdapter.setOnItemClickListener(object : EstateAdapter.OnItemClickListener {
23	override fun onItemClick(data: Estate) {
24	startUnityActivity(data.modelUrl)
25	}
26	})
27	recyclerView.adapter = estateAdapter
28	
29	mainPresenter.fetchData()
30	}
31	
32	override fun showLoading() {
33	materialDialog = MaterialDialog.Builder(this)
34	.title("Perumahan Balimbingan Permai")
35	.content("loading...")
36	.progress(true, 0)
37	.show()
38	
39	}
40	
41	override fun hideLoading() {
42	materialDialog.dismiss()
43	}
44	
45	override fun onSuccessResult(data: List<Estate>?) {
46	estateAdapter.data = data
47	estateAdapter.notifyDataSetChanged()
48	
49	}
50	
51	override fun onFailureResult(message: String) {
52	Toast.makeText(this, message, Toast.LENGTH_SHORT).show()
53	}
54	
55	override fun startUnityActivity(modelUrl: String) {
56	val intent = Intent(this, UnityPlayerActivity::class.java)
57	intent.putExtra("modelUrl", modelUrl)
58	startActivity(intent)
59	}
60	
61	override fun setPresenter(presenter: MainContract.Presenter) {
62	mainPresenter = presenter
63	}
64	
65	}

**Tabel 5.11 Kode Program HomePresenter (Android)**

1	class HomePresenter(private val view: MainContract.View, private val apiService: EstateApiService) :
2	MainContract.Presenter {
3	
4	override fun fetchData() {
5	

```

6      view.showLoading()
7      val call: Call<ApiResponse> = apiService.getEstates()
8
9      call.enqueue(object : Callback<ApiResponse> {
10
11          override fun onResponse(call: Call<ApiResponse>?, response:
Response<ApiResponse>?) {
12              try {
13                  Log.d("SKRIPSHIT", response?.toString())
14
15                  if (response!!.isSuccessful) {
16
17                      val data: ApiResponse? = response.body()
18                      view.onGetSuccessResult(data?.result)
19                      view.hideLoading()
20
21                  } else {
22                      view.hideLoading()
23                      view.onGetFailureResult(response.message())
24                  }
25
26              } catch (e: Exception) {
27                  view.hideLoading()
28                  view.onGetFailureResult(e.message!!)
29              }
30          }
31
32          override fun onFailure(call: Call<ApiResponse>?, t:
Throwable?) {
33              view.hideLoading()
34              view.onGetFailureResult(t!!.message!!)
35          }
36      })
37  }
38
39  override fun onViewCreated() {
40      TODO("Not yet implemented")
41  }
42
43  override fun onDestroy() {
44      TODO("Not yet implemented")
45  }
46  }

```

**Tabel 5.12 Kode Program EstateAdapter (Android)**

```

1  class EstateAdapter(var context: Context) :
RecyclerView.Adapter<EstateAdapter.ViewHolder>() {
2
3      var data: List<Estate>? = ArrayList()
4      var itemClickListener: OnItemClickListener? = null
5
6      override fun onCreateViewHolder(parent: ViewGroup, pl: Int):
ViewHolder {
7          val view =
LayoutInflater.from(context).inflate(R.layout.item_layout, parent,
8          false);
9          return ViewHolder(view)
10     }
11
12     override fun onBindViewHolder(holder: ViewHolder, position: Int) {

```

13	holder.itemView.title.text = data?.get(position)?.title ?: ""
14	val options: RequestOptions = RequestOptions()
15	.centerCrop()
16	.placeholder(R.mipmap.ic_launcher_round)
17	.error(R.mipmap.ic_launcher_round)
18	
19	
20	Glide.with(context).load(data?.get(position)?.thumbnailUrl).apply(options
21	)
22	.into(holder.itemView.image)
23	}
24	
25	override fun getItemCount(): Int {
26	return data?.size!!
27	}
28	
29	fun setOnItemClickListener(itemClickListener: OnItemClickListener) {
30	this.itemClickListener = itemClickListener
31	}
32	
33	inner class ViewHolder(itemView: View) :
34	RecyclerView.ViewHolder(itemView) {
35	
36	init {
37	itemView.setOnClickListener {
38	data?.get(adapterPosition)?.let { it2 ->
39	itemClickListener?.onItemClicked(it2) }
40	}
41	}
42	
43	}
44	
45	interface OnItemClickListener {
46	fun onItemClicked(data: Estate)
47	}
	}

**Tabel 5.13 Kode Program EstateApiService (Android)**

1	interface EstateApiService {
2	@GET("api/estates/get")
3	fun getEstates(): Call<ApiResponse>
4	}

**Tabel 5.14 Kode Program Main (Unity)**

1	public class Main : MonoBehaviour
2	{
3	
4	private bool hasExtra = false;
5	public string modelUrl = "";
6	public AndroidJavaObject extras;
7	public AndroidJavaObject intent;
8	
9	void Start()
10	{



11	Application.targetFrameRate = 60;
12	
13	AndroidJavaClass UnityPlayer = new
	AndroidJavaClass("com.unity3d.player.UnityPlayer");
14	AndroidJavaObject currentActivity =
	UnityPlayer.GetStatic<AndroidJavaObject>("currentActivity");
15	intent = currentActivity.Call<AndroidJavaObject>("getIntent");
16	hasExtra = intent.Call<bool>("hasExtra", "modelUrl");
17	Debug.Log("start");
18	
19	if (hasExtra)
20	{
21	Debug.Log("has extra");
22	extras = intent.Call<AndroidJavaObject>("getExtras");
23	modelUrl = extras.Call<string>("getString", "modelUrl");
24	Debug.Log("Model URL : " + modelUrl);
25	}
26	else
27	{
28	Debug.Log("no extra");
29	}
30	}
31	
32	void FixedUpdate()
33	{
34	if (Application.platform == RuntimePlatform.Android)
35	{
36	if (Input.GetKey(KeyCode.Escape))
37	{
38	Application.Quit();
39	}
40	}
41	}
42	}

**Tabel 5.15 Kode Program FirstPersonController (Unity)**

1	namespace Scripts
2	{
3	public class FirstPersonController : MonoBehaviour
4	{
5	public Camera playerCamera;
6	public Transform cameraTransform;
7	public CharacterController characterController;
8	public float cameraSensitivity;
9	public float moveSpeed;
10	public float moveInputDeadZone;
11	int leftFingerId, rightFingerId;
12	float halfScreenWidth;
13	Vector2 lookInput;
14	float cameraPitch;
15	Vector2 moveTouchStartPosition;
16	Vector2 moveInput;
17	protected Joystick Joystick;
18	
19	void Start()
20	{
21	leftFingerId = -1;
22	rightFingerId = -1;

```

23         halfScreenWidth = Screen.width / 2;
24         moveInputDeadZone = Mathf.Pow(Screen.height /
moveInputDeadZone, 2);
25     }
26
27     void Awake()
28     {
29         Joystick = FindObjectOfType<Joystick>();
30     }
31
32     void Update()
33     {
34         GetTouchInput();
35
36         if (rightFingerId != -1)
37         {
38             LookAround();
39         }
40
41         if (leftFingerId != -1)
42         {
43             Vector2 movementDirection = Joystick.AxisNormalized *
moveSpeed * Time.deltaTime;
44             characterController.Move(transform.right *
Joystick.AxisNormalized.x * 0.15f + transform.forward *
movementDirection.y);
45         }
46     }
47
48     void GetTouchInput()
49     {
50         for (int i = 0; i < Input.touchCount; i++)
51         {
52             Touch t = Input.GetTouch(i);
53
54             switch (t.phase)
55             {
56                 case TouchPhase.Began:
57                     if (t.position.x < halfScreenWidth &&
leftFingerId == -1)
58                     {
59                         leftFingerId = t.fingerId;
60                         moveTouchStartPosition = t.position;
61                     }
62                     else if (t.position.x > halfScreenWidth &&
rightFingerId == -1)
63                     {
64                         rightFingerId = t.fingerId;
65                     }
66                     break;
67                 case TouchPhase.Ended:
68                 case TouchPhase.Canceled:
69                     if (t.fingerId == leftFingerId)
70                     {
71                         leftFingerId = -1;
72                     }
73                     else if (t.fingerId == rightFingerId)
74                     {
75                         rightFingerId = -1;
76                     }
77             }
78         }
79     }

```

```

80         }
81
82         break;
83         case TouchPhase.Moved:
84             if (t.fingerId == rightFingerId)
85             {
86                 lookInput = t.deltaPosition *
cameraSensitivity * Time.deltaTime;
87             }
88             else if (t.fingerId == leftFingerId)
89             {
90                 moveInput = t.position -
moveTouchStartPosition;
91             }
92
93             break;
94             case TouchPhase.Stationary:
95                 if (t.fingerId == rightFingerId)
96                 {
97                     lookInput = Vector2.zero;
98                 }
99                 break;
100             }
101         }
102     }
103
104     void LookAround()
105     {
106         cameraPitch = Mathf.Clamp(cameraPitch - lookInput.y, -90f,
90f);
107         playerCamera.transform.localRotation =
Quaternion.Euler(cameraPitch, 0, 0);
108
109         transform.Rotate(transform.up, lookInput.x);
110     }
111
112     void Move()
113     {
114         if (moveInput.sqrMagnitude <= moveInputDeadZone) return;
115
116         Vector2 movementDirection = moveInput.normalized * moveSpeed
* Time.deltaTime;
117         characterController.Move(transform.right *
movementDirection.x + transform.forward * movementDirection.y);
118     }
119 }
120
121 }
122 }

```

**Tabel 5.16 Kode Program API untuk mengunduh model 3 dimensi rumah (Unity)**

```

1 public class API : MonoBehaviour
2 {
3     public Text ProgressIndicator;
4     public Image LoadingBar;
5
6     public void GetBundleObject(string assetName,
UnityAction<GameObject> callback, Transform bundleParent)
7     {
8         StartCoroutine(GetDisplayBundleRoutine(assetName, callback,

```

```

bundleParent));
9      }
10
11      string GetFilePath(string url)
12      {
13          string[] pieces = url.Split('/');
14          string filename = pieces[pieces.Length - 1];
15          return Path.Combine(Application.persistentDataPath, filename);
16      }
17
18      IEnumerator GetDisplayBundleRoutine(string bundleURL,
UnityAction<GameObject> callback, Transform bundleParent)
19      {
20          Debug.Log("Requesting bundle at " + bundleURL);
21          UnityWebRequest www =
UnityWebRequestAssetBundle.GetAssetBundle(bundleURL);
22          yield return www.SendWebRequest();
23
24          StartCoroutine(ShowDownloadProgress(www));
25
26          if (www.isNetworkError)
27          {
28              Debug.Log("Network error");
29          }
30          else
31          {
32              AssetBundle bundle =
DownloadHandlerAssetBundle.GetContent(www);
33              if (bundle != null)
34              {
35                  string rootAssetPath = bundle.GetAllAssetNames()[0];
36                  GameObject arObject =
Instantiate(bundle.LoadAsset(rootAssetPath) as GameObject,
bundleParent);
37                  bundle.Unload(false);
38                  callback(arObject);
39              }
40              else
41              {
42                  Debug.Log("Not a valid asset bundle");
43              }
44          }
45      }
46  }
47
48  IEnumerator ShowDownloadProgress(UnityWebRequest req)
49  {
50      float downloadDataProgress;
51      while (!req.isDone)
52      {
53          downloadDataProgress = req.downloadProgress * 100;
54          Debug.Log("Download: " + downloadDataProgress);
55
56          ProgressIndicator.text =
((int)downloadDataProgress).ToString() + "%";
57          LoadingBar.fillAmount = downloadDataProgress / 100;
58
59          yield return null;
60      }
61  }

```

**Tabel 5.17 Kode Program Inventory Perabotan (Unity)**

```

1 namespace Scripts
2 {
3     public class SC_InventorySystem : MonoBehaviour
4     {
5         public Texture crosshairTexture;
6         public FirstPersonController playerController;
7         public SC_PickItem[] availableItems; //List with Prefabs of all
the available items
8         int[] itemSlots = new int[6];
9         bool showInventory = false;
10        SC_PickItem detectedItem;
11        int detectedItemIndex;
12        Vector2 firstTouchPosition;
13        List<TouchField> touchedFields = new List<TouchField>();
14        GameObject inventory;
15
16        void Start()
17        {
18            for (int i = 0; i < itemSlots.Length; i++)
19            {
20                itemSlots[i] = -1;
21            }
22
23            for (int i = 0; i < availableItems.Length; i++)
24            {
25                itemSlots[i] = i;
26            }
27
28            inventory = GameObject.Find("Inventory");
29            inventory.SetActive(false);
30
31        }
32
33        void Update()
34        {
35
36            if (showInventory && touchedFields.Count > 0)
37            {
38                TouchField touchField = touchedFields[0];
39
40                for (int j = 0; j < availableItems.Length; j++)
41                {
42                    if (touchField.itemName ==
43availableItems[j].itemName)
44                    {
45                        RaycastHit hit;
46                        Ray ray =
playerController.playerCamera.ViewportPointToRay(new Vector3(0.5F, 0.5F,
470));
48
49                        if (Physics.Raycast(ray, out hit,
50Mathf.Infinity))
51                        {
52                            Transform objectHit = hit.transform;
53                            Vector3 hitPoint = hit.point;
54                            hitPoint += objectHit.up;
55                            Instantiate(availableItems[j], hitPoint,
transform.rotation);
// Instantiate(availableItems[j],
playerController.playerCamera.transform.position +

```

```

56 (playerController.playerCamera.transform.forward), Quaternion.identity);
57     }
58
59     break;
60 }
61 }
62 touchedFields.RemoveAt(0);
63 }
64 }
65
66 public void addTouchedField(TouchField touchField)
67 {
68     touchedFields.Add(touchField);
69 }
70
71 public void removeItemClicked()
72 {
73     if (detectedItem && detectedItemIndex > -1)
74     {
75         detectedItem.PickItem();
76     }
77 }
78
79 public void toggleShowInventory()
80 {
81     showInventory = !showInventory;
82     inventory.SetActive(showInventory);
83 }
84
85 void FixedUpdate()
86 {
87     RaycastHit hit;
88     Ray ray =
89     playerController.playerCamera.ViewportPointToRay(new Vector3(0.5F, 0.5F,
90     0));
91
92     if (Physics.Raycast(ray, out hit, 5f))
93     {
94         Transform objectHit = hit.transform;
95
96         if (objectHit.CompareTag("Respawn"))
97         {
98             if ((detectedItem == null || detectedItem.transform
99             != objectHit) && objectHit.GetComponent<SC_PickItem>() != null)
100             {
101                 SC_PickItem itemTmp =
102                 objectHit.GetComponent<SC_PickItem>();
103
104                 for (int i = 0; i < availableItems.Length; i++)
105                 {
106                     if (availableItems[i].itemName ==
107                     itemTmp.itemName)
108                     {
109                         detectedItem = itemTmp;
110                         detectedItemIndex = i;
111                     }
112                 }
113             }
114             else
115             {
116                 detectedItem = null;
117             }
118         }
119     }
120 }

```

115	}
116	else
117	{
118	detectedItem = null;
119	}
120	}
121	
122	void OnGUI()
	{
123	GUI.color = detectedItem ? Color.green : Color.white;
124	GUI.DrawTexture(new Rect(Screen.width / 2 - 4, Screen.height
125	/ 2 - 4, 8, 8), crosshairTexture);
126	GUI.color = Color.white;
127	
128	if (detectedItem)
	{
129	GUI.color = new Color(0, 0, 0, 0.84f);
130	GUI.Label(new Rect(Screen.width / 2 - 75 + 1,
	Screen.height / 2 - 50 + 1, 150, 20), detectedItem.objectName);
131	GUI.color = Color.green;
132	GUI.Label(new Rect(Screen.width / 2 - 75, Screen.height /
133	2 - 50, 150, 20), detectedItem.objectName);
134	}
	}
	}

**Tabel 5.18 Kode Program Main.go (API Service)**

1	package main
2	import (
3	"fmt"
4	"net/http"
5	"os"
6	"simple-api/app"
7	"simple-api/controllers"
8	"github.com/gorilla/mux"
9	)
10	
11	func main() {
12	router := mux.NewRouter()
13	router.HandleFunc("/api/user/new",
	controllers.CreateAccount).Methods("POST")
14	router.HandleFunc("/api/user/login",
	controllers.Authenticate).Methods("POST")
15	router.HandleFunc("/api/estates/new",
	controllers.CreateEstate).Methods("POST")
16	router.HandleFunc("/api/estate/{id}",
	controllers.GetEstate).Methods("GET")
17	router.HandleFunc("/api/estates/get",
	controllers.GetEstates).Methods("GET") // e.g : user/2/contacts
18	
19	//attach JWT auth middleware
20	router.Use(app.JwtAuthentication)
21	
22	port := os.Getenv("PORT")
23	if port == "" {
24	port = "8000" //localhost
25	}
26	
27	fmt.Println(port)

28	err := http.ListenAndServe(":"+port, router) //Launch the app, visit localhost:8000/api
29	if err != nil {
30	fmt.Print(err)
31	}
32	}

**Tabel 5.19 Kode Program EstateController.go (API Service)**

1	package controllers
2	
3	import (
4	"encoding/json"
5	"net/http"
6	"simple-api/models"
7	u "simple-api/utills"
8	"strconv"
9	"github.com/gorilla/mux"
10	)
11	
12	var CreateEstate = func(w http.ResponseWriter, r *http.Request) {
13	estate := &models.Estate{}
14	
15	err := json.NewDecoder(r.Body).Decode(estate)
16	if err != nil {
17	u.Respond(w, u.Message(false, "Error while decoding request body"))
18	return
19	}
20	
21	resp := estate.Create()
22	u.Respond(w, resp)
23	}
24	
25	var GetEstate = func(w http.ResponseWriter, r *http.Request) {
26	params := mux.Vars(r)
27	id, err := strconv.Atoi(params["id"])
28	
29	if err != nil {
30	u.Respond(w, u.Message(false, "There was an error in your request"))
31	return
32	}
33	
34	data := models.GetEstate(uint(id))
35	resp := u.Message(true, "success")
36	resp["result"] = data
37	u.Respond(w, resp)
38	}
39	
40	var GetEstates = func(w http.ResponseWriter, r *http.Request) {
41	data := models.GetEstates()
42	resp := u.Message(true, "success")
43	resp["result"] = data
44	u.Respond(w, resp)
45	}



**Tabel 5.20 Kode Program Estate.go (API Service)**

```

1 package models
2
3 import (
4     "fmt"
5     u "simple-api/utils"
6     "github.com/jinzhu/gorm"
7 )
8
9 type Estate struct {
10     gorm.Model
11     Title      string `json:"title"`
12     Description string `json:"description"`
13     ThumbnailUrl string `json:"thumbnailUrl"`
14     ModelUrl    string `json:"modelUrl"`
15 }
16
17 func (estate *Estate) Validate() (map[string]interface{}, bool) {
18     if estate.Title == "" {
19         return u.Message(false, "Estate title should be on the
20 payload"), false
21     }
22     if estate.Description == "" {
23         return u.Message(false, "Description should be on the
24 payload"), false
25     }
26     return u.Message(true, "success"), true
27 }
28
29 func (estate *Estate) Create() map[string]interface{} {
30     if resp, ok := estate.Validate(); !ok {
31         return resp
32     }
33
34     GetDB().Create(estate)
35
36     resp := u.Message(true, "success")
37     resp["estate"] = estate
38     return resp
39 }
40
41 func GetEstate(id uint) *Estate {
42     estate := &Estate{}
43     err := GetDB().Table("estates").Where("id = ?",
44 id).First(estate).Error
45     if err != nil {
46         fmt.Println(err)
47         return nil
48     }
49     return estate
50 }
51
52 func GetEstates() []*Estate {
53     estates := make([]*Estate, 0)
54     err := GetDB().Table("estates").Find(&estates).Error
55     if err != nil {
56         fmt.Println(err)
57         return nil
58     }
59 }

```

58	
60	return estates
61	}

### 5.2.5 Implementasi Antarmuka

Antarmuka aplikasi berbasis Android TurInteraktif diimplementasikan dengan menggunakan format XML untuk mengatur layout dari masing-masing tampilan. Alur perpindahan halaman digambarkan pada Gambar 5.8. Terdapat 2 halaman hasil implementasi, yaitu halaman home, dan halaman Tur Interaktif yang digambarkan pada Gambar 5.10 dan Gambar 5.11**Error! Reference source not found.** berturut-turut.

**Gambar 5.10 Implementasi Antarmuka Home**

**Gambar 5.11 Implementasi Antarmuka Tur Interaktif**

## BAB 6 PENGUJIAN

### 6.1 Pengujian *Black-box*

Pengujian *Black-box* dilakukan untuk memastikan bahwa fitur yang dikembangkan sudah memenuhi kebutuhan fungsional yang sudah ditentukan. Pengujian dilakukan dengan mengikuti alur yang ada pada *use case scenario*. Hasil pengujian diringkas pada Tabel 6.1 hingga Tabel 6.3.

**Tabel 6.1 Hasil Pengujian Black-box Menampilkan Daftar Rumah**

Kode Kebutuhan		SRS-TIVR-F-1
Kasus uji 1	Nama Kasus Uji	Menampilkan daftar rumah saat terhubung ke internet
	Prosedur	1. Pengguna memiliki koneksi ke jaringan internet. 2. Pengguna berada pada halaman <i>home</i> .
	Hasil yang diharapkan	Sistem menampilkan data daftar rumah yang di dapat dari respon <i>request API Service</i> .
	Hasil yang didapat	Sistem menampilkan data daftar rumah yang di dapat dari respon <i>request API Service</i> .
	Status	Valid
Kasus uji 2	Nama kasus uji	Menampilkan daftar rumah saat tidak terhubung ke internet
	Prosedur	1. Pengguna tidak memiliki koneksi ke jaringan internet. 2. Pengguna berada pada halaman <i>home</i> .
	Hasil yang diharapkan	Sistem tidak menampilkan data daftar rumah yang di dapat dari respon <i>request API Service</i> dan memberikan <i>toast</i> pesan error tidak ada koneksi internet.
	Hasil yang didapat	Sistem tidak menampilkan data daftar rumah yang di dapat dari respon <i>request API Service</i> dan memberikan <i>toast</i> pesan error tidak ada koneksi internet.
	Status	Valid

**Tabel 6.2 Hasil Pengujian Menampilkan Tur Interaktif**

Kode Kebutuhan		SRS-TIVR-F-2
Kasus uji 1	Nama Kasus Uji	Menampilkan Tur Interaktif dengan <i>url</i> model berisi alamat yang valid
	Prosedur	<ol style="list-style-type: none"> <li>1. Pengguna berada pada halaman home.</li> <li>2. Pengguna melakukan <i>tap</i> pada salah satu <i>item</i> data rumah.</li> <li>3. Sistem memanggil method <i>startActivity</i> untuk menjalankan program Unity dengan mengirimkan <i>string</i> yang berisi <i>url</i> dari model 3 dimensi rumah.</li> </ol>
	Hasil yang diharapkan	Sistem menampilkan tur interaktif dengan menjalankan <i>activity</i> Unity.
	Hasil yang didapat	Sistem menampilkan tur interaktif dengan menjalankan <i>activity</i> Unity.
	Status	Valid
Kasus uji 2	Nama kasus uji	Menampilkan Tur Interaktif dengan <i>url</i> model berisi alamat yang tidak valid
	Prosedur	<ol style="list-style-type: none"> <li>1. Pengguna berada pada halaman home.</li> <li>2. Pengguna melakukan <i>tap</i> pada salah satu <i>item</i> data rumah.</li> <li>3. Sistem memanggil method <i>startActivity</i> untuk menjalankan program Unity dengan mengirimkan <i>string</i> yang berisi nilai kosong atau nilai yang tidak valid.</li> </ol>
	Hasil yang diharapkan	Sistem tidak menampilkan tur interaktif.
	Hasil yang didapat	Sistem tidak menampilkan tur interaktif.
	Status	Valid

**Tabel 6.3 Hasil Pengujian Black-box Menambah Daftar Rumah**

Kode Kebutuhan		SRS-TIVR-F-3
Kasus uji 1	Nama Kasus Uji	Menambah data rumah saat terhubung ke internet
	Prosedur	<ol style="list-style-type: none"> <li>1. Pengguna memiliki koneksi ke jaringan internet.</li> <li>2. Pengguna berada pada halaman <i>home</i> dari aplikasi Admin.</li> <li>3. Pengguna mengisi form data</li> <li>4. Pengguna melakukan submit untuk mengirim data ke <i>API Service</i>.</li> </ol>
	Hasil yang diharapkan	Sistem berhasil mengirimkan <i>request</i> untuk menambah data rumah ke <i>API Service</i> .
	Hasil yang didapat	Sistem berhasil mengirimkan <i>request</i> untuk menambah data rumah ke <i>API Service</i> .
	Status	Valid
Kasus uji 2	Nama Kasus Uji	Menambah data rumah saat tidak terhubung ke internet
	Prosedur	<ol style="list-style-type: none"> <li>1. Pengguna tidak memiliki koneksi ke jaringan internet.</li> <li>2. Pengguna berada pada halaman <i>home</i> dari aplikasi Admin.</li> <li>3. Pengguna mengisi form data</li> <li>4. Pengguna melakukan submit untuk mengirim data ke <i>API Service</i>.</li> </ol>
	Hasil yang diharapkan	Sistem menampilkan pesan error tidak ada internet melalui <i>toast android</i> .
	Hasil yang didapat	Sistem menampilkan pesan error tidak ada internet melalui <i>toast android</i> .
	Status	Valid

### 6.1.1 Analisis Hasil Pengujian Black-box

Pengujian *Black-box* dilakukan untuk menguji apakah sistem yang dibuat sudah memenuhi kebutuhan yang didefinisikan. Pengujian dilakukan dengan menentukan kasus uji dan hasil yang diharapkan, kemudian aplikasi diuji sesuai dengan kasus yang ditentukan. Hasil pengujian dari 6 kasus uji bernilai valid.

## 6.2 Pengujian Usability

Pengujian *usability* dilakukan untuk menguji kebergunaan aplikasi bagi pengguna. Pengujian dilakukan dengan cara memberikan *task scenario* kepada pengguna, kemudian pengguna akan melakukan aktivitas untuk menjalankan *task scenario* yang diberikan. *Task scenario* yang diberikan kepada pengguna merupakan langkah-langkah dalam menjalankan aplikasi. Langkah-langkah tersebut mengacu pada fungsionalitas aplikasi.

Pada pengujian ini, *task scenario* diberikan kepada dua jenis pengguna, yaitu pengguna dan *admin*. *Task scenario* untuk pengguna dapat dilihat pada Tabel 6.4 dan *task scenario* untuk admin dapat dilihat pada Tabel 6.5.

**Tabel 6.4 Task Scenario Untuk Pengguna**

<i>Task</i>	Nama <i>Task</i>	Langkah-langkah
1	Menampilkan daftar rumah	<ol style="list-style-type: none"><li>1. Berada di halaman utama</li><li>2. Sistem menampilkan daftar rumah</li></ol>
2	Menampilkan tur interaktif	<ol style="list-style-type: none"><li>1. Pengguna menunggu proses menampilkan daftar rumah selesai</li><li>2. Pengguna melakukan <i>tap</i> pada salah satu <i>item</i> data rumah</li><li>3. Sistem menampilkan tur interaktif dengan membuka <i>activity</i> yang dikembangkan menggunakan Unity 3D Engine.</li></ol>

**Tabel 6.5 Task Scenario untuk Admin**

<i>Task</i>	Nama <i>Task</i>	Langkah-langkah
1	Menambah data rumah	<ol style="list-style-type: none"><li>1. Berada di halaman utama</li><li>2. Admin mengisi form</li><li>3. Admin menekan tombol <i>submit</i></li></ol>

Pada Tabel 6.4 merupakan 2 *task scenario* yang diberikan kepada pengguna aplikasi Tur Interaktif, sedangkan pada Tabel 6.5 merupakan 1 *task scenario* yang diberikan kepada admin aplikasi Tur Interaktif. Dalam melaksanakan *task scenario* yang diberikan, masing-masing pengguna harus menjalankan *task scenario* tersebut secara berurutan.

Kemudian responden diminta untuk mengisi kuesioner sebagai penilaian tentang usabilitas pada aplikasi ini. Pengujian pada penelitian ini menggunakan kuesioner SUPER-Qm yang terdiri dari 12 pertanyaan untuk masing-masing jenis pengguna yang dapat dilihat pada Tabel 6.7 untuk pengguna aplikasi interaktif dan pada Tabel 6.8 untuk

admin aplikasi Tur Interaktif. Setiap pertanyaan yang diberikan memiliki skor dengan menggunakan skala likert dalam rentang 1 sampai 5 yang dijelaskan pada Tabel 6.6.

**Tabel 6.6 Skor Skala Likert dari Setiap Pertanyaan**

Skor	Keterangan
1	Sangat Tidak Setuju (STS)
2	Tidak Setuju (TS)
3	Netral (N)
4	Setuju (S)
5	Sangat Setuju (SS)

**Tabel 6.7 Kuesioner SUPR-Qm untuk Pengguna**

No	Pertanyaan
1	Aplikasi ini penting untuk saya
2	Aplikasi ini merupakan aplikasi tur interaktif terbaik yang pernah saya gunakan
3	Saya tidak tahu apakah ada aplikasi tur interaktif yang lebih baik dari aplikasi ini
4	Saya tidak akan menghapus aplikasi ini dari smartphone saya
5	Saya akan menyarankan aplikasi ini ke teman saya
6	Saya suka mengeksplorasi fitur-fitur yang terdapat dalam aplikasi ini
7	Aplikasi ini memiliki seluruh fitur dan fungsi yang saya inginkan pada aplikasi tur interaktif
8	Desain dari aplikasi ini memudahkan saya dalam menemukan informasi yang saya inginkan
9	Menurut saya aplikasi ini menarik
10	Aplikasi ini sesuai dengan kebutuhan saya
11	Melakukan navigasi dalam aplikasi ini mudah bagi saya

12	Aplikasi ini mudah digunakan
----	------------------------------

**Tabel 6.8 Kuesioner SUPR-Qm untuk Admin**

No	Pertanyaan
1	Aplikasi ini penting untuk saya
2	Aplikasi ini merupakan aplikasi tur interaktif terbaik yang pernah saya gunakan
3	Saya tidak tahu apakah ada aplikasi tur interaktif yang lebih baik dari aplikasi ini
4	Saya tidak akan menghapus aplikasi ini dari smartphone saya
5	Saya akan menyarankan aplikasi ini ke teman saya
6	Saya suka mengeksplorasi fitur-fitur yang terdapat dalam aplikasi ini
7	Aplikasi ini memiliki seluruh fitur dan fungsi yang saya inginkan pada aplikasi tur interaktif
8	Desain dari aplikasi ini memudahkan saya dalam menemukan informasi yang saya inginkan
9	Menurut saya aplikasi ini menarik
10	Aplikasi ini sesuai dengan kebutuhan saya
11	Melakukan navigasi dalam aplikasi ini mudah bagi saya
12	Aplikasi ini mudah digunakan

Pada Tabel 6.7 dan Tabel 6.8 merupakan 12 buah pertanyaan dengan menggunakan kuesioner SUPR-Qm yang nantinya akan didapatkan hasil secara kuantitatif terhadap kebergunaan dan kemudahan aplikasi yang dikembangkan.

Perhitungan SUPR-Q yang dilakukan untuk mendapatkan hasil kuantitatif dari skor pada pertanyaan kuesioner SUPR-Qm dilakukan dengan menggunakan rumus pada Persamaan 6.1.



$$\text{Nilai SUPR} - Q = \frac{\text{jumlah nilai diperoleh}}{\text{jumlah nilai maksimal}} \times 100\%$$

**(6.1)**

Persamaan SUPR-Q pada Persamaan 6.1 digunakan untuk perhitungan nilai SUPR-Qm. Kemudian nilai hasil perhitungan tersebut akan dikonversikan ke dalam kategori penilaian *usability*.

#### 6.2.1. Analisis Hasil Pengujian Usability Aplikasi untuk Pengguna Tur Interaktif

Pengujian *usability* untuk pengguna dilakukan kepada tiga responden yang merupakan pengguna aplikasi Tur Interaktif. Pengambilan responden dilakukan secara acak kepada pengguna yang berpotensi sebagai home seeker perumahan Balimbingan Permai. Adapun lima responden tersebut dijelaskan pada Tabel 6.9.

**Tabel 6.9 Responden Pengujian *Usability* Pengguna Tur Interaktif**

No	Nama	Latar Belakang
1	Insan Nurzaman	Pelanggan yang kurang dari tiga kali dalam setahun memesan menu di restoran Wong Solo. Pengguna <i>smartphone</i> Android yang sering menggunakan <i>smartphone</i> -nya untuk penggunaan kamera atau GPS
2	Zakwan Dhiyaulhaq	Pelanggan yang sekali dalam setahun memesan menu di restoran Wong Solo. Pengguna <i>smartphone</i> Android yang menggunakan <i>smartphone</i> -nya untuk penggunaan finansial dan kamera atau GPS
3	Saiful Wardi Lubis	Pelanggan yang lebih dari lima kali dalam satu tahun memesan menu di restoran Wong Solo. Pengguna <i>smartphone</i> Android yang menggunakan <i>smartphone</i> -nya untuk penggunaan finansial dan hiburan

Kemudian responden diminta untuk mengoperasikan aplikasi berdasarkan *task scenario* dan moderator mengamati perilaku responden untuk mengetahui apakah responden dapat menyelesaikan *task scenario* yang diberikan atau tidak. Hasil dari penyelesaian *task scenario* dapat dilihat pada Tabel 6.10.

**Tabel 6.10 Task Completion Rate**

	<i>User 1</i>	<i>User 2</i>	<i>User 3</i>	<i>User 4</i>	<i>User 5</i>	<i>Completion Rate</i>
<i>Task 1</i>	√	√	√	√	√	100%
<i>Task 2</i>	√	√	√	√	√	100%

Pada Tabel 6.10 dapat disimpulkan bahwa semua responden dapat menyelesaikan 2 task scenario yang diberikan oleh moderator dengan *completion rate* bernilai 100%.

Setelah responden menyelesaikan *task scenario* yang diberikan, selanjutnya responden diminta untuk mengisi kuesioner SUPR-Qm yang telah didefinisikan sebelumnya. Adapun hasil kuesioner SUPR-Qm yang telah diisi oleh responden pelanggan dapat dilihat pada LAMPIRAN C dan hasil dari pengujian kuesioner SUPR-Qm untuk pelanggan restoran dapat dilihat pada Tabel 6.11.

**Tabel 6.11 Hasil Pengujian Kuesioner SUPR-Qm untuk Pengguna Tur Interaktif**

No	Pertanyaan	Skor					Total Skor
		1	2	3	4	5	
1	Aplikasi ini penting untuk saya	0	0	0	5	0	20
2	Aplikasi ini merupakan aplikasi tur interaktif terbaik yang pernah saya gunakan	0	0	2	3	0	18
3	Saya tidak tahu apakah ada aplikasi tur interaktif yang lebih baik dari aplikasi ini	0	0	1	3	1	20
4	Saya tidak akan menghapus aplikasi ini dari smartphone saya	0	0	1	2	2	21
5	Saya akan menyarankan aplikasi ini ke teman saya	0	0	0	4	1	21
6	Saya suka mengeksplorasi fitur-fitur yang terdapat dalam aplikasi ini	0	0	0	3	2	22
7	Aplikasi ini memiliki seluruh fitur dan fungsi yang saya inginkan pada aplikasi tur interaktif	0	1	0	3	1	19
8	Desain dari aplikasi ini memudahkan saya dalam menemukan informasi yang saya inginkan	0	0	2	1	2	20

9	Menurut saya aplikasi ini menarik	0	0	2	0	3	21
10	Aplikasi ini sesuai dengan kebutuhan saya	0	0	1	2	2	21
11	Melakukan navigasi dalam aplikasi ini mudah bagi saya	0	0	1	3	1	20
12	Aplikasi ini mudah digunakan	0	0	0	1	4	24
Total Akhir							247
Total Nilai Maksimum							300
Nilai SUPR-Qm							82,33%

Tabel 6.11 menunjukkan penilaian tertinggi pada pertanyaan SUPR-Qm nomor 12 dengan skor 24 dan penilaian terendah didapat pada pertanyaan SUPR-Qm nomor 2 dengan skor 18. Pada tabel juga menunjukkan total akhir yang bernilai 247, sehingga didapatkan nilai SUPR-Qm sebesar 82,33%.

Setelah mendapatkan nilai akhir SUPR-Qm sebesar 82.33%, kemudian nilai tersebut diinterpretasi ke kategori nilai *usability*. Penginterpretasian nilai 82,33% pada kategori nilai *usability* mendapatkan kategori B dengan *adjective rating excellent*. Sehingga dapat disimpulkan bahwa aplikasi Tur Interaktif sangat baik dan dapat diterima oleh pengguna aplikasi Tur Interaktif.

#### 6.2.2. Analisis Hasil Pengujian Usability Aplikasi untuk Admin Tur Interaktif

Pengujian *Usability* kepada Admin Aplikasi Tur Interaktif diberikan kepada satu responden. Kemudian responden diminta untuk mengoperasikan aplikasi berdasarkan *task scenario* dan moderator mengamati perilaku responden untuk mengetahui apakah responden dapat menyelesaikan *task scenario* yang diberikan atau tidak. Hasil dari penyelesaian *task scenario* dapat dilihat pada Tabel 6.12.

**Tabel 6.12 Task Completion Rate**

	<i>User 1</i>	<i>Completion Rate</i>
<i>Task 1</i>	√	100%

Pada Tabel 6.12 dapat disimpulkan bahwa semua responden dapat menyelesaikan task scenario yang diberikan oleh moderator dengan *completion rate* bernilai 100%.

Responden menyelesaikan *task scenario* kemudian responden mengisi kuesioner SUPR-Qm yang telah didefinisikan sebelumnya. Berikut hasil kuesioner SUPR-Qm responden *admin* aplikasi tur interaktif dapat dilihat pada LAMPIRAN D dan hasil dari pengujian kuesioner SUPR-Qm untuk admin aplikasi tur interaktif dapat dilihat pada Tabel 6.13

**Tabel 6.13 Hasil Pengujian Kuesioner SUPR-Qm untuk *Admin* aplikasi Tur Interaktif**

No	Pertanyaan	Skor					Total Skor
		1	2	3	4	5	
1	Aplikasi ini penting untuk saya	0	0	1	0	0	3
2	Aplikasi ini merupakan aplikasi tur interaktif terbaik yang pernah saya gunakan	0	0	0	0	1	5
3	Saya tidak tahu apakah ada aplikasi tur interaktif yang lebih baik dari aplikasi ini	0	0	0	0	1	5
4	Saya tidak akan menghapus aplikasi ini dari smartphone saya	0	0	1	0	0	3
5	Saya akan menyarankan aplikasi ini ke teman saya	0	0	0	1	0	4
6	Saya suka mengeksplorasi fitur-fitur yang terdapat dalam aplikasi ini	0	0	0	0	1	5
7	Aplikasi ini memiliki seluruh fitur dan fungsi yang saya inginkan pada aplikasi tur interaktif	0	0	0	0	1	5
8	Desain dari aplikasi ini memudahkan saya dalam menemukan informasi yang saya inginkan	0	0	0	0	1	5
9	Menurut saya aplikasi ini menarik	0	0	0	0	1	5
10	Aplikasi ini sesuai dengan kebutuhan saya	0	0	0	0	1	5
11	Melakukan navigasi dalam aplikasi ini mudah bagi saya	0	0	0	0	1	5
12	Aplikasi ini mudah digunakan	0	0	0	0	1	5
Total Akhir							55
Total Nilai Maksimum							60
Nilai SUPR-Qm							91,67%

Tabel 6.13 menunjukkan bahwa 12 pertanyaan mendapatkan skor tertinggi yaitu 5, sedangkan skor terendah ditunjukkan pada pertanyaan nomor 1 dan 4 dengan skor 3. Pada tabel di atas menunjukkan total akhir yang bernilai 60, sehingga didapatkan nilai SUPR-Qm sebesar 91,67%.

Setelah mendapatkan nilai akhir SUPR-Qm sebesar 91,67%, kemudian nilai tersebut diinterpretasi ke kategori nilai *usability*. Penginterpretasian nilai 91,67% pada kategori nilai *usability* mendapatkan kategori A dengan *adjective rating best imaginable*. Sehingga dapat disimpulkan bahwa aplikasi Tur Interaktif untuk admin sangat baik dan dapat diterima.

### 6.3 Pengujian Compatibility

Pengujian Compatibility akan dilakukan dengan menguji aplikasi dengan pemasangan pada lima perangkat berbeda. Dengan aplikasi operasi dan spesifikasi yang berbeda akan dilakukan beberapa kasus uji dan hasil dari pengujian compatibility akan dikatakan valid.

Jika hasil pengujian sesuai dengan implementasi yang telah dilakukan. Pengujian compatibility akan dilakukan dengan cara membuat kasus uji dan kemudian akan dilakukan proses pengujian serta akan dibandingkan dengan hasil pengujian (Zhang, et al., March 2015).

Pengujian compatibility yang telah dilakukan, aplikasi mampu untuk melakukan menampilkan tampilan yang sesuai dengan implementasi sebelumnya pada enam perangkat yang berbeda dengan baik. Kesimpulan dari hasil dan analisis pengujian compatibility ini bahwa aplikasi memiliki compatibility yang baik untuk perangkat uji yang telah disediakan dengan tingkat keberhasilan 80%. Dengan adanya hasil tersebut, aplikasi ini memiliki tingkat compatibility yang tinggi pada perangkat lainnya.

#### 6.3.1. Spesifikasi Perangkat Uji

Pengujian ini menggunakan satu jenis perangkat keras dan lima virtual device yang telah disiapkan serta dijalankan secara otomatis menggunakan *firebase test lab* dan mewakili perbedaan versi aplikasi operasi *android* dan resolusi layar berdasarkan data yang diambil dari (StatCounter, 2017). Penjelasan spesifikasi perangkat uji yang digunakan dalam uji coba aplikasi implementasi Tur Interaktif akan dijelaskan sebagai berikut.

**Tabel 6.14 Tabel Spesifikasi Perangkat Uji 1**

No	Jenis Perangkat Uji	Spesifikasi
1.	Model	Nexus 5
2.	Sistem Operasi	Android 6.0 (API Level 23)
3.	Resolusi	1080 x 1920
4.	<i>Density Resolution</i>	480 dpi
Keterangan		Berjalan dengan baik

**Tabel 6.15 Tabel Spesifikasi Perangkat Uji 2**

No	Jenis Perangkat Uji	Spesifikasi
1.	Model	LG G6 (VS988)
2.	Sistem Operasi	Android 7.0 (API Level 24)
3.	Resolusi	1440 x 2880
4.	<i>Density Resolution</i>	640 dpi
Keterangan		Berjalan dengan baik

**Tabel 6.16 Tabel Spesifikasi Perangkat Uji 3**

No	Jenis Perangkat Uji	Spesifikasi
1.	Model	Sony Xperia F8332
2.	Sistem Operasi	Android 8.0 (API Level 26)
3.	Resolusi	1080 x 1920
4.	<i>Density Resolution</i>	480 dpi
Keterangan		Berjalan dengan baik

**Tabel 6.17 Tabel Spesifikasi Perangkat Uji 4**

No	Jenis Perangkat Uji	Spesifikasi
1.	Model	Mi 8
2.	Sistem Operasi	Android 9.x (API Level 28)
3.	Resolusi	1080 x 2248
4.	<i>Density Resolution</i>	440 dpi
Keterangan		Berjalan dengan baik

**Tabel 6.18 Tabel Spesifikasi Perangkat Uji 5**

No	Jenis Perangkat Uji	Spesifikasi
1.	Model	Google Pixel 4
2.	Sistem Operasi	Android 11 (API Level 30)

<b>3.</b>	Resolusi	1080 x 2280
<b>4.</b>	<i>Density Resolution</i>	440 dpi
Keterangan		Gagal dijalankan

### **6.3.2. Analisis Hasil Pengujian *Compatibility***

Berdasarkan pengujian yang telah dilakukan terhadap 5 jenis perangkat dengan spesifikasi sistem operasi yang berbeda beda, terdapat 4 perangkat yang berhasil dijalankan dan 1 perangkat yang gagal menjalankan aplikasi Tur Interaktif. Maka berdasarkan hasil pengujian pada Tabel 6.14 sampai dengan Tabel 6.18, versi sistem operasi Android yang mampu menjalankan aplikasi Tur Interaktif yaitu Android versi 6.0 sampai dengan versi Android 9.

## BAB 7 PENUTUP

Pada bagian ini membahas kesimpulan dan saran terhadap penelitian pengembangan aplikasi tur interaktif.

### 7.1 Kesimpulan

Berdasarkan hasil dari analisis kebutuhan, perancangan, implementasi, dan pengujian pada penelitian pengembangan tur interaktif didapatkan tiga kesimpulan, antara lain:

1. Permasalahan survey lokasi atau fisik perumahan yang sebelumnya harus mendatangi lokasi secara langsung dapat diganti dengan menggunakan aplikasi tur interaktif sebagai aplikasi yang dapat melakukan tur secara interaktif dan virtual tanpa harus datang langsung ke lokasi.
2. Berdasarkan pengujian validasi yang telah dilakukan terhadap implementasi sistem yang dibuat, didapatkan hasil bahwa validasi yang dilakukan terhadap kebutuhan fungsionalitas dinyatakan valid.
3. Berdasarkan pengujian *usability* yang telah dilakukan kepada pengguna secara langsung, didapatkan hasil bahwa aplikasi tur interaktif untuk pengguna mendapatkan nilai *usability* sebesar 82,25% yang masuk ke dalam kategori B, dengan *adjective rating excellent*. Aplikasi tur interaktif untuk admin mendapatkan nilai *usability* sebesar 96,25% yang masuk ke dalam kategori A, dengan *adjective rating best imaginable*. Sehingga dapat disimpulkan bahwa aplikasi tur interaktif sangat berguna bagi pengguna dan admin.

### 7.2 Saran

Berdasarkan penelitian yang telah dilakukan, penulis merasa bahwa penelitian ini belum dapat dikatakan sempurna. Oleh karena itu, didapatkan saran yang dapat dikembangkan untuk penelitian selanjutnya, antara lain:

1. Perlu ditambahkan lebih banyak lagi interaksi terhadap objek 3 dimensi rumah selain menambah furniture, seperti mengganti warna objek dan lain-lain untuk meningkatkan usability dari aplikasi.
2. Perlu dilakukan penelitian lanjut untuk menggali kebutuhan dan pengujian lebih spesifik dikarenakan jumlah instrumen yang sedikit, agar aplikasi tur interaktif ini lebih tepat sasaran dan sesuai dengan kebutuhan pengguna.



## DAFTAR PUSTAKA

- Sommerville, I. (2016). *Software Engineering* (10th ed.). Harlow, England: Pearson Education
- Hendro Trieddiantoro Puto (2015). *Kajian Virtual Reality*. Makalah Studi Mandiri: Universitas Teknologi Yogyakarta, Januari 2015.
- Sherman, W. R., Craig, A. B. (2003). *Understanding Virtual Reality*. Morgan Kaufmann Publishing: Interface Application and Design. San Fransisco, CA.
- Amin, D. & Golvikar, R. (2015). Comparative Study of Augmented Reality SKD's . *International Journal on Computational Sciences & Application (IJCSA)*, Hal 2-7.
- Ozacar, K., Ortakci, Y., dkk (2017). A Low-Cost and Lightweight 3D Interactive Real Estate-Purposed Indoor Virtual Reality Application. *International Journal on Photogrammetry, Remote Sensing and Spatial Information Sciences*. Hal 308-309.
- Fitrana, E. A., dkk (2019). Pengembangan Aplikasi Katalog Rumah Berbasis Augmented Reality Menggunakan Algoritma FAST. *JISKa*. Vol. 4, No. 1. Hal 10. ISSN : 2527-5836.
- Husniah, L. dkk. (2016). Interaktif Augmented Reality untuk Katalog Penjualan Rumah Berbasis Android. *KINETIK*. Vol. 1, No. 1. Hal 33. ISSN : 2503-2259.
- Sutrisno, Adam dkk. (2015). Implementasi Teknologi Augmented Reality pada Agen Penjualan Rumah. *E-Journal Teknik Elektro dan Komputer*. Hal 19-20. ISSN : 2301-8402.
- Rumajar, R. (2015). Perancangan Brosur Interaktif Berbasis Augmented Reality. *E-Journal Teknik Elektro dan Komputer*. Vol 4, No. 6. Hal 1-3. ISSN : 2301-8402.
- Wirawan, Raden dkk. (2016). Aplikasi Augmented Reality pada Sistem Informasi Smart Building. *JNTETI*. Vol. 5, No. 3. Hal 20. ISSN : 2301 – 4156
- Suhendar, Akip dkk. (2016). Aplikasi Virtual tour Berbasis Multimedia Interaktif Menggunakan AutoDesk 3Ds Max. *Jurnal ProTekInfo*. Vol. 3, No. 1. Hal 30-35. ISSN : 2406-7741.

Pramono, Basworo Ardi, 2012, Desain dan Implementasi Augmented Reality Berbasis Web Pada Aplikasi Furniture Shopping Manager Sebagai Alat Bantu Belanja Online. Jurnal Transformatika. Vol.10, No.1, Hal 28.

Gede Wahya Dhiyatmika, I., Putra, I., & Mandenni, N. (2015). Aplikasi Augmented Reality Magic Book Pengenalan Binatang Untuk Siswa TK. Lontar Komputer: Jurnal Ilmiah Teknologi Informasi, 120-127.

## LAMPIRAN A DATA CATATAN PENGUNJUNG

### PERUMAHAN BALIMBINGAN PERMAI PT. KARYA PROPERTINDO UTAMA

Berikut data pengunjung (home seeker) perumahan balimbingan, yang ingin membeli rumah namun batal. Terdapat data nama, tahun, tipe rumah yang ingin dibeli, serta kendala / alasan batal membeli rumah, seperti tertera pada tabel di bawah.

No	NAMA	ALAMAT / DOMISILI	TAHUN	TIPE RUMAH	KENDALA
1	Ronald Heron Siahaan	Palembang	2016	45	Batal berkunjung / survey dikarenakan jarak jauh, meminta model rumah berbentuk 3-dimensi.
2	Handoko	Kebun PTP IV Balimbingan	2016	45	Batal Berkunjung karena kerja kantor, tidak ada waktu kosong untuk survey lapangan
3	Riko Silalahi	Balige - Kab. Toba Samosir	2016	36	Batal Berkunjung karena kerja kantor, tidak ada waktu kosong untuk survey lapangan, meminta model rumah berbentuk 3-dimensi
4	Nihon Halawa	Bah Kisat - Tanah Jawa	2016	54	Tanpa kabar kembali
5	Jhon Harman Sinaga	Jakarta	2017	54	Batal berkunjung dikarenakan masalah jarak yang jauh untuk berkunjung, meminta model rumah berbentuk 3-dimensi
6	Jagusman Siregar	P. Siantar	2017	70	Batal berkunjung karena tidak ada waktu libur kerja kantor
7	Effendi Sitanggang	Tanah Jawa	2017	36	Tanpa kabar kembali
8	Muchtar Simanjuntak	P. Siantar	2017	54	Tidak jadi berkunjung
9	Silvi	P. Siantar	2017	45	Tidak jadi berkunjung
10	Sriantis Butar-Butar	Balige	2017	45	Tidak jadi berkunjung
11	Sariati Silitonga	Balimbingan	2017	45	Tanpa Alasan
12	Helmiati Tanjung	Rumah Sakit Balimbingan	2017	54	Tanpa kabar kembali
13	Renata Siagian	P. Siantar	2017	54	Tanpa kabar kembali
14	Judir Nadapdap	Tanjung Pasir - Tanah Jawa	2017	45	Tanpa kabar kembali
15	Desi Silalahi	Jakarta	2018	45	Batal berkunjung / survey dikarenakan jarak jauh, meminta model rumah berbentuk 3-dimensi.
16	Azis	Tonduhan - Kebun	2018	54	Tidak jadi berkunjung karena kerja kantor kebun
17	Arifin Sipayung	Tanah Jawa	2018	54	Tanpa kabar kembali
18	Lusi Simanjuntak	Tanah Jawa	2018	45	Tanpa kabar kembali

19	Robin Sinabariba	Balimbingan	2018	45	Tanpa kabar kembali
20	Darwis	TebingTinggi	16 Maret 2019	36	Batal survey dikarenakan masalah jarak yang jauh untuk berkunjung, minta 3D
21	Erick Tampubolon	Perdagangan	2019	36	Batal Berkunjung karena kerja kantor, tidak ada waktu kosong untuk survey lapangan
22	Pungua Sidabutar	Hutabayu	2019	54	Tanpa kabar kembali
23	Heuda P. Silitonga	Medan	2019	36	Batal Survey kendala jarak jauh, Minta 3D
24	Maslon Sirait	Kebun PTP IV Mandoge	2020	70	Batal berkunjung karena membatasi diri keluar rumah akibat covid-19
25	Riski Silalahi	P. Siantar	2020	54	Perlu model 3D rumah, membatasi diri untuk berkunjung karena pandedmi Covid-19
26	Bangun Sihombing, S.STP.M.Si	P. Siantar	2020	54	Perlu Model 3D
27	Betterpen. S	Huta Bayu	2020	36	Perlu Model 3D

Berikut data harga jual perumahan balimbingan permai, seperti pada tabel di bawah yang ditampilkan berdasarkan tipe, harga, beserta estimasi keuntungan.

No	Tipe Rumah	Harga Jual	Estimasi Keuntungan
1	36	Rp 130.000.000	Rp 18.000.000
2	45	Rp 160.000.000	Rp 30.000.000
3	54	Rp. 210.000.000	Rp. 45.000.000
4	70	Rp. 260.000.000	Rp. 52.000.000

Demikian data yang tertera di atas adalah benar-benar data perumahan Balimbingan Permai yang sah diberikan oleh perusahaan PT. Karya Propertindo Utama dan akan digunakan sebagai data pendukung untuk penelitian skripsi.

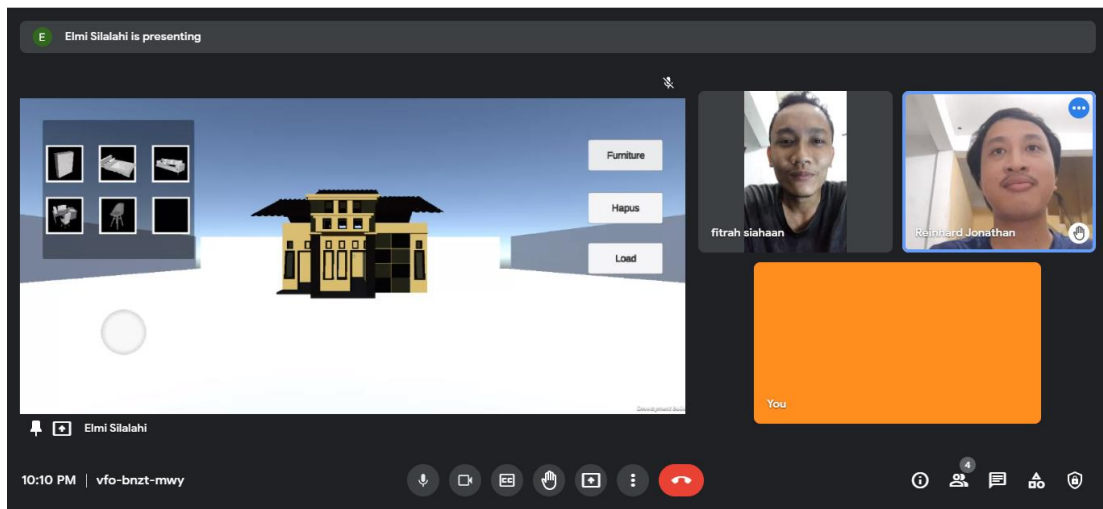
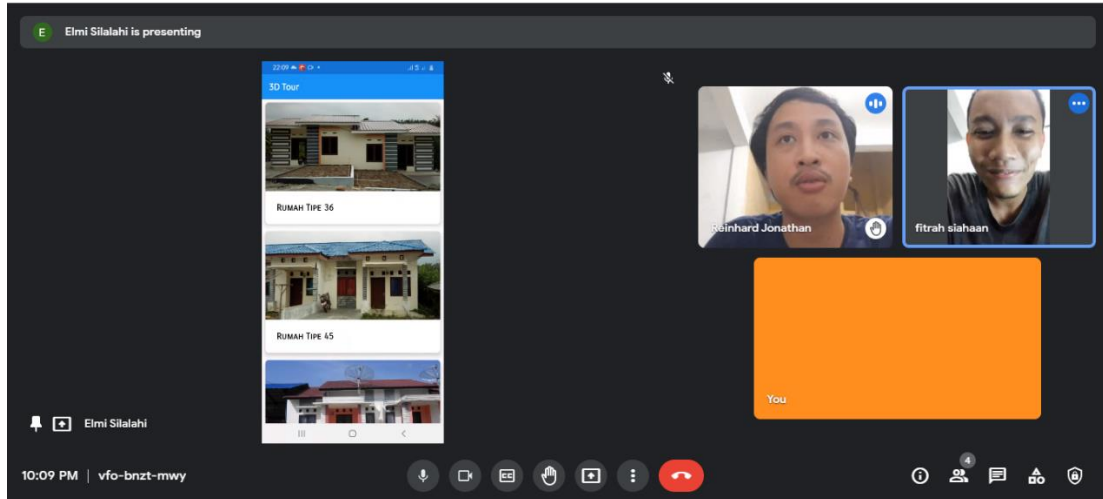
Balimbingan, 21 September 2020

PT. Karya Propertindo Utama,

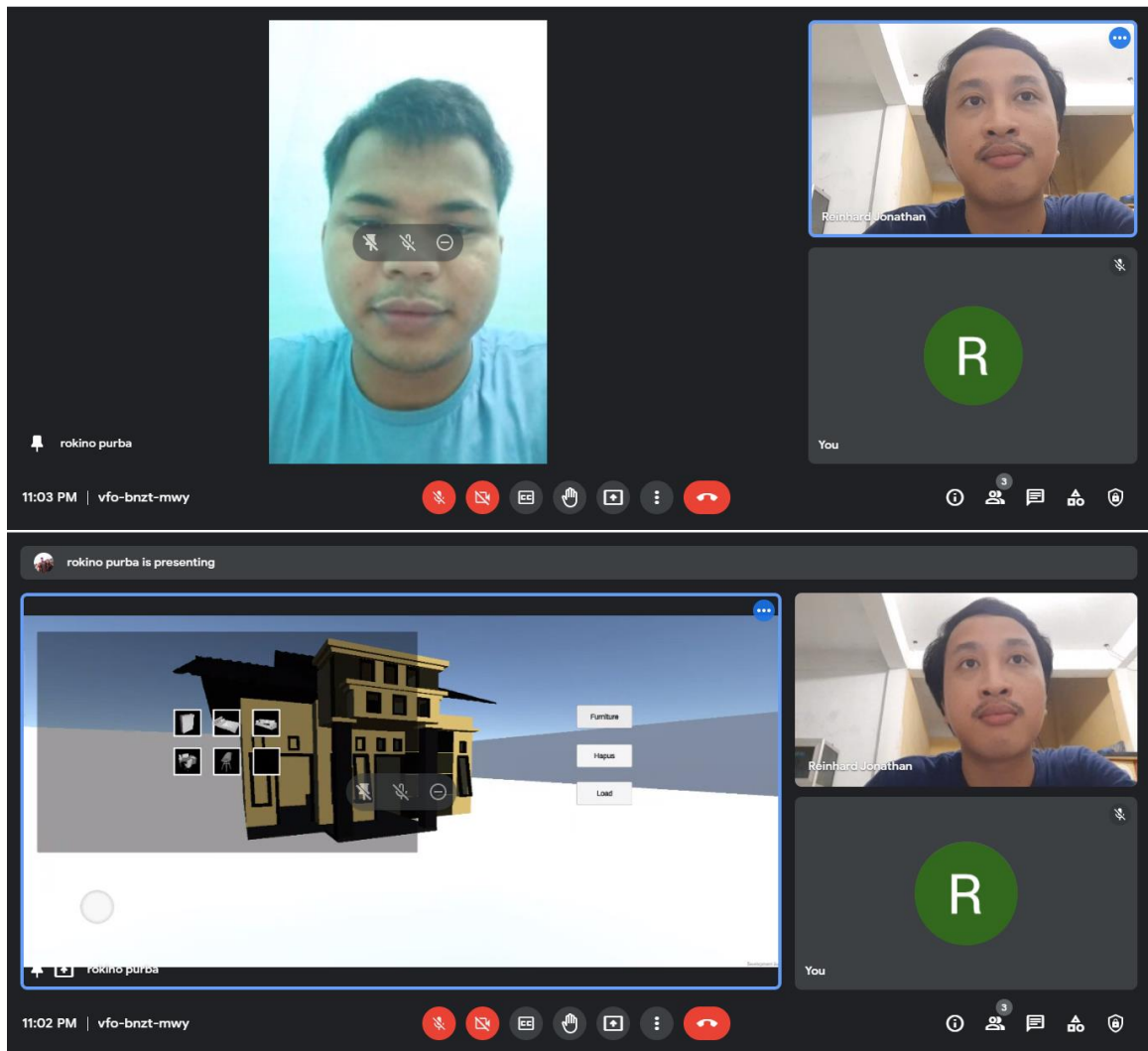
  
 Ir. Mangantar Silalahi, MT  
 Pemilik Proyek

## LAMPIRAN B DEMONSTRASI PRODUK

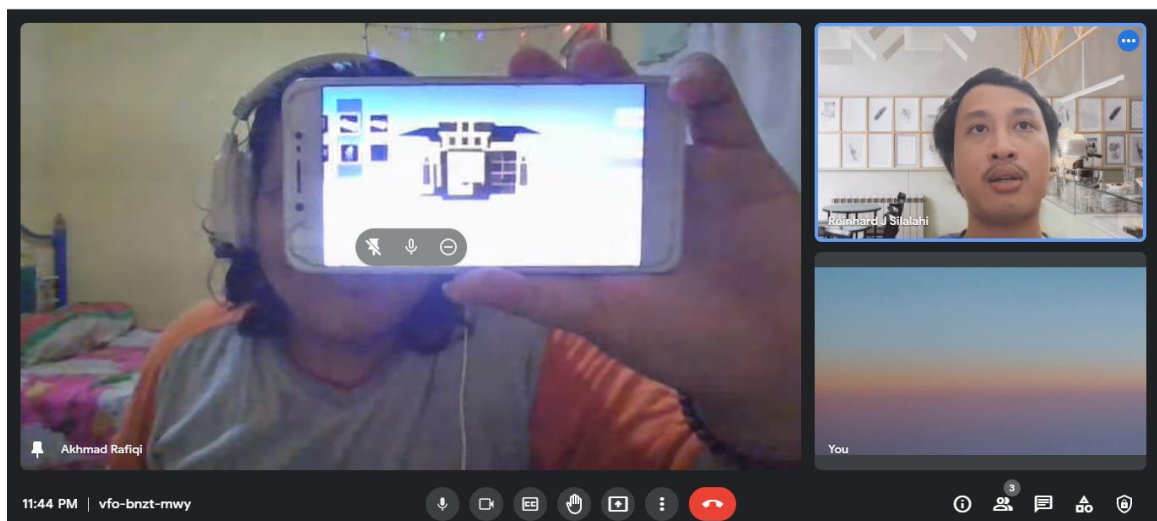
### B.1 Demonstrasi Produk Responden 1 sebagai Pengguna



## B.2 Demonstrasi Produk Responden 2 Sebagai Pengguna



### B.3 Demonstrasi Produk Responden 3 Sebagai Pengguna



## LAMPIRAN C KUESIONER PENGGUNA APLIKASI TUR INTERAKTIF

### C.1 Kuesioner Responden 1

Nama
Fitra Haposan Siahaan

Umur
26 tahun

Lokasi
Medan

Aplikasi ini penting untuk saya				
1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Aplikasi ini merupakan aplikasi tur interaktif terbaik yang pernah saya gunakan				
1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>



Saya tidak tahu apakah ada aplikasi tur interaktif yang lebih baik dari aplikasi ini

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>

Saya tidak akan menghapus aplikasi ini dari smartphone saya

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Saya akan menyarankan aplikasi ini ke teman saya

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Saya suka mengeksplorasi fitur-fitur yang terdapat dalam aplikasi ini

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Aplikasi ini memiliki seluruh fitur dan fungsi yang saya inginkan pada aplikasi tur interaktif

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Desain dari aplikasi ini memudahkan saya dalam menemukan informasi yang saya inginkan

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Menurut saya aplikasi ini menarik

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Aplikasi ini sesuai dengan kebutuhan saya

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Melakukan navigasi dalam aplikasi ini mudah bagi saya

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Aplikasi ini mudah digunakan

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Submitted 6/14/21, 10:14 PM

## **LAMPIRAN D KUESIONER ADMIN APLIKASI TUR INTERAKTIF**

### **D.1 Kuesioner Responden 1**