

**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

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, properti perumahan yang dibangun oleh perusahaan PT, memiliki alasan untuk tidak bisa berkomentar karena beberapa dari mereka tinggal di luar kota dan memiliki waktu yang sangat terbatas. dengan 48% pelanggan mereka membatalkan pembelian. Karya Propertindo Utama Penulis membuat aplikasi wisata interaktif untuk perumahan balimbingan yang indah atas dasar kesulitan ini. Aplikasi pengembangan mengambil bentuk dua aplikasi, satu untuk pengguna dan yang lain untuk administrator. Kerangka kerja air terjun dengan kebutuhan pengguna tertentu adalah metode pengembangan perangkat lunak dalam penelitian ini. Penelitian ini bertujuan untuk menyelesaikan masalah tur virtual interaktif.

Peneliti juga melakukan tiga tes, yaitu pengujian validasi, kegunaan untuk mendeteksi kegunaan aplikasi kepada pengguna dan kompatibilitas untuk menentukan apakah aplikasi dapat dijalankan pada beberapa versi Android. Tes validasi menunjukkan bahwa validasi fungsionalitas dinyatakan valid, sementara hasil tes kegunaan menunjukkan bahwa aplikasi dengan peringkat kata sifat kegunaan yang sangat baik untuk pengguna memiliki kegunaan 82,33 persen, yang termasuk dalam kategori B. Nilai keterjangkauan aplikasi administrasi adalah 96,25%, yang termasuk dalam kategori A. Dan hasil pengujian kompatibilitas menunjukkan bahwa aplikasi dapat bekerja dengan versi 6.0 hingga 9.0 pada sistem operasi Android.

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 LAMPIRAN

LAMPIRAN A DATA CATATAN PENGUNJUNG	59
LAMPIRAN B DEMONSTRASI PRODUK	61
LAMPIRAN C KUESIONER PENGGUNA APLIKASI TUR INTERAKTIF.....	64
LAMPIRAN D KUESIONER ADMIN APLIKASI TUR INTERAKTIF.....	67

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. Penulis menggunakan metode pengembangan Waterfall SDLC untuk pengembangan aplikasi, dengan metode multifase, yaitu perencanaan sistem, analisis sistem, desain sistem, penyebaran sistem dan pemeliharaan sistem. Peneliti menggunakan tiga metode: pengujian blackbox (pengujian fungsional) dan pengujian non-fungsional (kegunaan dan pengujian kompatibilitas). Para peneliti mengemukakan data untuk pengujian kegunaan dengan menyediakan surveyor calon pembeli dan pengunjung ke rumah mereka dengan Skala Kegunaan Sistem (SUS).

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, formulasi masalah, tujuan untuk penelitian, keuntungan penelitian, kendala penelitian, dan diskusi sistematis.

1.1.2. BAB II Landasan Kepustakaan

Bab literatur pendiri memberikan gambaran penelitian saat ini dalam penggunaan realitas virtual di berbagai bidang, seperti pendidikan, pariwisata dan perumahan. Dengan berbagai metode dan penerapan metode. Penjelasan teori mencakup teori yang mempromosikan desain dan pengembangan.

1.1.3. BAB III Metodologi Penelitian

Bab ini meneliti tahapan penelitian dalam pengumpulan data dan bagaimana data, analisis teoritis dan proses penelitian diperoleh.

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

Bagian ini menjelaskan bahwa pengujian berlangsung dari awal aplikasi, setelah implementasi, ketika pengguna memilih tur 3D atau menu realitas virtual. Aplikasi mulai mengenali pola dalam brosur jika Anda memilih realitas virtual, dan kemudian menghasilkan output sebagai objek virtual. Aplikasi ini kemudian akan menampilkan tampilan di mana pengguna dapat mengunjungi lingkungan 3D virtual jika Anda memilih Tur 3D. Hasil tes kemudian ditentukan oleh apakah aplikasi memenuhi kriteria ketika menunjukkan hasil sesuai dengan apa yang diinginkan pengguna.

1.1.7. BAB VII Penutup

Bab ini menjelaskan hasil temuan ini dan menunjukkan bahwa sistem dapat memenuhi kriteria yang telah ditentukan lebih baik lagi.

BAB 2 LANDASAN KEPUSTAKAAN

Deskripsi dan penjabaran ide, konsep, model, teknik atau sistem dalam literatur ilmiah tentang subjek, masalah atau masalah penelitian termasuk dalam landasan kepastakaan. Dasar teoritis di mana beberapa sumber perpustakaan dipelajari dalam literatur untuk teori dan metodologi. Jika perlu, tinjauan literatur juga dapat dilakukan berdasarkan karakteristik penelitian dan persyaratan khusus untuk kecukupan kekhawatiran khusus yang menggambarkan penelitian sebelumnya tentang topik penelitian secara umum dan menunjukkan perbandingan dan perbedaan antara penelitian masa lalu yang ditulis dan penelitian.

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) adalah teknologi yang memungkinkan orang untuk terlibat dengan dunia imajiner, dunia komputer, nyata atau mereplikasi (Sihite 2013). Gagasan untuk VR mengacu pada serangkaian konsep, teknik, dan proses yang digunakan untuk membangun dan memproduksi perangkat lunak untuk sistem komputer multifungsi (Lacrama, 2007).Pengaturan realitas virtual aktual biasanya menawarkan pengalaman visual, baik di layar komputer atau melalui jendela bidik stereoskopis, tetapi beberapa simulasi memberikan input sensorik lainnya, termasuk suara menggunakan speaker atau headphone (Sihite, 2013).

2.3 Unity 3D

Unity 3D adalah instrumen terintegrasi untuk membangun bentuk objek 3D dalam video game atau interaktif lainnya seperti penayangan arsitektur atau animasi 3D secara real-time.Di Microsoft Windows dan Mac Os X, lingkungan pemrograman Unity 3D berjalan dan program yang dibangun oleh Unit 3D dapat berjalan di Windows, Mac, Xbox 360, Playstation 3, Wii, iPad, dan iPhone, dan tidak kembali di Android.Plug-in pemutar web Unity yang berfungsi di Mac dan Windows dan tidak berfungsi di Linux juga memungkinkan Anda untuk membuat game berbasis browser (Juniadi, 2015 dan Mutia).

2.4 Android

Android adalah sistem operasi yang digunakan pada ponsel berbasis Linux (Nazaruddin, 2012).Android awalnya dikembangkan oleh Android, Inc.dan Google dibeli Android secara resmi pada tahun 2005.Android adalah salah satu sistem operasi paling populer saat ini dan ada sejumlah besar perangkat seluler menggunakan Android.Pengembang aplikasi sekarang dapat membuat kode dengan bahasa Kotlin, Java dan C saat merancang aplikasi yang berjalan pada sistem operasi (Developers, 2019).

2.5 Pengujian Fungsional

Pengujian fungsional adalah tes penilaian yang memastikan bahwa aplikasi atau sistem yang dibangun telah memenuhi semua aplikasi atau kebutuhan sistem. Tes ini tidak berfokus pada keberhasilan prosedur sistem tetapi pada hasil sistem. Salah satu teknikya seperti kotak hitam dalam tes ini.

2.6 Pengujian *Usability*

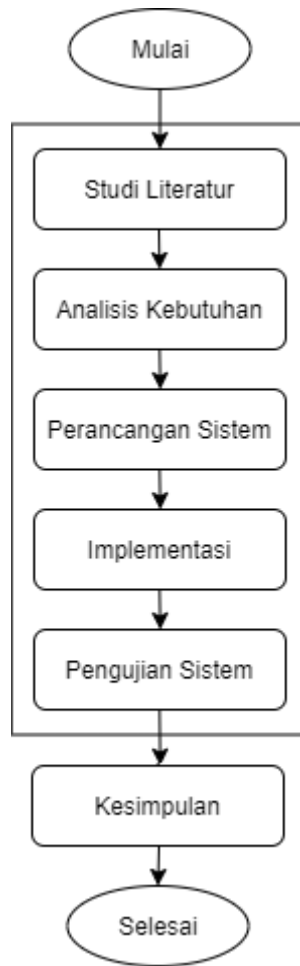
Usability adalah makna yang berasal dari istilah yang berguna dan karenanya dapat dimanfaatkan dengan benar. Dapat diklaim bahwa apa yang dibuat sangat membantu atau sangat baik untuk digunakan ketika ada beberapa atau sedikit kegagalan dalam penggunaan sehingga dapat memuaskan dan memberi orang manfaat (Jeff Rubin, 2008). Sementara itu, menurut ISO 9241:11 (1998), "kegunaan suatu produk adalah jumlah yang dapat digunakan pengguna tertentu untuk memenuhi tujuan yang dinyatakan dalam situasi tertentu dalam hal efektivitas, efisiensi, dan kepuasan."

BAB 3 METODOLOGI PENELITIAN

Metodologi studi menjelaskan teknik implementasi penelitian yang digunakan dalam desain dan pengembangan wisata interaktif di Balimbingan Permai berdasarkan virtual reality. Langkah-langkah yang terlibat dalam teknik pengembangan aplikasi akan dijelaskan dalam bab ini. Para peneliti memanfaatkan pendekatan yang sesuai dengan model Air Terjun SDLC dengan model pengembangan sistem dalam penelitian ini.

Pola pengembangan air terjun adalah pola pengembangan perangkat lunak satu arah, di mana proses pengembangan tidak akan kembali ke fase sebelumnya. Jadi, jika pengembang telah dalam posisi untuk memahami masalah dan kebutuhan aplikasi yang akan dikembangkan, model pengembangan ini cocok untuk aplikasi dan dapat memastikan bahwa kebutuhan aplikasi yang akan dikembangkan tidak berubah secara signifikan dan persyaratan yang diperoleh melalui proses elitasi dapat dipastikan selama dan di masa depan pengembangan perangkat lunak (Sommerville, 2011).

Pola pengembangan air terjun digunakan karena ada daftar kriteria yang konsisten atau jarang dimodifikasi dalam aplikasi yang akan dibangun dalam penelitian ini. Teknik air terjun terdiri dari tujuh fase: analisis literatur, analisis kebutuhan, desain sistem, implementasi, tes sistem dan kesimpulan seperti yang diilustrasikan pada gambar di bawah ini.



Gambar 3.1 Metode Pengembangan

3.1 Studi Literatur

Penelitian literatur menyediakan dasar teoritis yang diproduksi berdasarkan makalah, buku, jurnal, dan karya serupa. Studi sastra mendukung dan mendorong pengembangan tesis dan informasi lebih lanjut untuk menganalisis kebutuhan, membuat, membangun, dan menguji program untuk tujuan penelitian. Untuk mencegah plagiarisme dari proyek / penelitian masa lalu, studi literatur juga dilakukan.

3.2 Analisis Kebutuhan Sistem

Studi persyaratan sistem sangat membantu untuk menentukan kebutuhan pengembangan perangkat lunak. Analisis berorientasi objek adalah strategi yang

digunakan untuk menganalisis persyaratan perangkat lunak dalam penelitian ini (OOA). Kebutuhan dimodelkan menggunakan UML, yang terdiri dari Diagram Kasus Penggunaan dan Skenario Kasus Penggunaan, setelah analisis persyaratan dilakukan. Gunakan Diagram Kasus untuk mengekspresikan persyaratan fungsional sistem/perangkat lunak dari sudut pandang pengguna. Dan gunakan skenario kasus untuk menjelaskan setiap kasus secara rinci dalam diagram kasus aplikasi.

3.3 Perancangan Sistem

Desain sistem digunakan untuk merancang setiap kebutuhan yang ditunjukkan dalam tahap analisis kebutuhan. Tahap desain digunakan untuk desain / konstruksi tahap kerja yang mencakup isi lengkap sistem yang akan diproduksi. Desain arsitektur, desain komponen, desain data, dan desain antarmuka sekarang tersedia. Setiap kebutuhan unik dirancang dengan diagram Sequence pada tahap desain. Diagram urutan mengilustrasikan aliran sistem berdasarkan fungsi garis waktu. Selain diagram urutan, terdapat juga diagram kelas yang berguna untuk menggambarkan hubungan antar setiap kelas yang saling terkait dalam suatu perangkat lunak yang dikembangkan.

3.4 Implementasi Sistem

Implementasi sistem adalah langkah ketika semua kebutuhan yang disebutkan dirancang. Implementasinya melibatkan implementasi database menggunakan MySQL DBMS, logika program dan implementasi antarmuka. Implementasi database adalah desain database yang dibuat dengan membuat tabel DDL (Data Definition Language). Implementasi bahasa logis program ini memanfaatkan pemrograman Java untuk framework Android SDK.

3.5 Pengujian Sistem

Tes sistem dilakukan untuk menguji sistem, apakah sistem secara efektif menerapkan atau tidak semua kriteria yang disediakan. Teknologi ini juga digunakan untuk menguji kekurangan dalam program yang dibuat. Dalam pengujian sistem,

pengujian struktur perangkat lunak internal dengan whitebox (kode sumber) dan pengujian struktur program eksternal dengan blackbox (antarmuka).

3.6 Kesimpulan dan Saran

Disimpulkan bahwa semua fase metodologi berhasil dilakukan secara teratur, dimulai dengan analisis literatur, analisis persyaratan, desain sistem, implementasi sistem, pengujian sistem dan analisis hasil. Ketika perangkat lunak yang dikembangkan melewati fase pengujian, kesimpulan yang valid diperoleh. Ada saran yang berguna selain temuan untuk meningkatkan apa yang diperlukan dalam pengembangan aplikasi tur interaktif berdasarkan realitas virtual. Rekomendasi yang berguna untuk pengembangan sistem di masa depan.

BAB 4 REKAYASA KEBUTUHAN

Bagian ini menguraikan fase analisis kebutuhan pembuatan aplikasi wisata interaktif untuk Perumahan Balimbingan Permai. Deskripsi sistem keseluruhan, aktor yang menggunakan aplikasi, papan cerita, persyaratan pembangunan sistem, diagram kasus penggunaan dan skenario kasus penggunaan, dijelaskan dalam bab ini.

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 memeriksa literatur sistem saat ini. Google Maps Popular Times dan sistem canggih melalui Lesani dan Miranda-Moreno adalah sistem yang digunakan sebagai referensi (2018). Penilaian persyaratan sistem adalah penilaian keinginan utama dari sistem, selain kebutuhan untuk mendukung sistem, harus dimasukkan. Penilaian dapat diklasifikasikan ke dalam dua persyaratan sistem, yaitu persyaratan menguntungkan dan kebutuhan yang tidak sesuai. Dalam diskusi ini, kode SRS-TIVR-FN untuk kode berharga untuk persyaratan, kode SRS-TIVR-NF-N untuk kebutuhan yang tidak perlu di mana N adalah rentang digunakan untuk mengekspresikan kebutuhan yang bermanfaat dan tidak berguna.

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

Persyaratan non-fungsional adalah persyaratan yang tidak terhubung langsung ke sistem tetapi mendukung kebutuhan fungsional (Summerville, 2011). Tabel 4.4 menyediakan kebutuhan non-fungsional.

Tabel 4.4 Kebutuhan Non-Fungsional Sistem

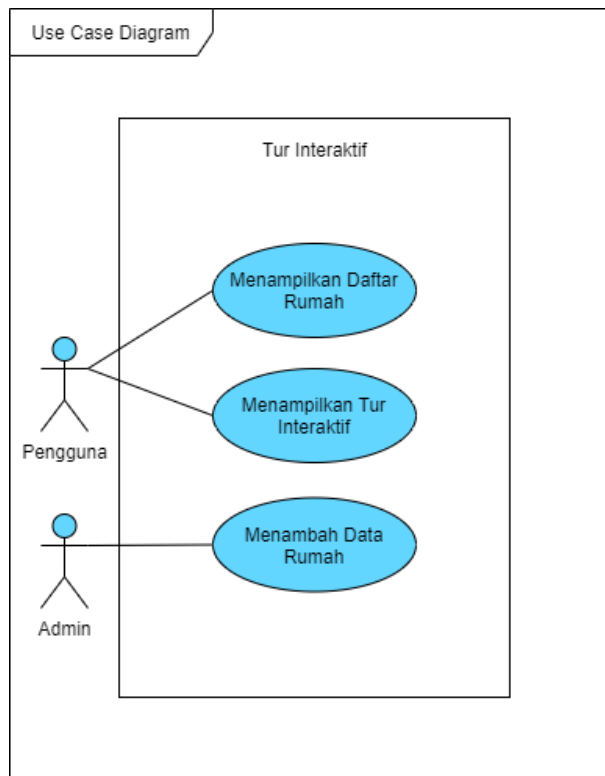
No	Kode Fungsi	Nama Fungsi	Deskripsi
1	SRS-TIVR-NF-1	<i>Usability</i>	<i>Usability</i> menjelaskan tentang kepuasan dan kemudahan pengguna terhadap pemakaian aplikasi. Usabilitas diangkat karena tujuan dari dikembangkanya aplikasi adalah untuk meningkatkan daya beli <i>home seeker</i> , maka diperlukan 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 menetapkan bagaimana subjek berinteraksi di bawah berbagai skenario dengan sistem. Use case menggambarkan perangkat lunak pengguna (Pressman, 2010).

4.3.1. Use Case Diagram

Diagram kasus untuk penggunaan adalah diagram yang mungkin berikan peserta ke sistem dan apa yang dapat mereka lakukan untuk dilakukan dalam sistem. Pengguna yang dapat berkomunikasi dengan sistem atau sistem eksternal dapat berpartisipasi (Sukamto & Salahuddin, 2016).



Gambar 4.1 Use Case Diagram Pengguna

4.3.2. Use case Scenario

Use Case Scenario adalah penjelasan lengkap dari setiap kasus yang cocok dengan diagram kasus penggunaan. Skenario Kasus Penggunaan didefinisikan sebagai tabel dengan nama kasus aplikasi, persyaratan kode, aktor, tujuan, pra-tujuan, aliran utama, pasca-tujuan, dan aliran alternatif. Bagian pra-kondisi menjelaskan kondisi sistem yang harus dipenuhi sebelum bagian aliran utama dimasukkan.

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	<ol style="list-style-type: none"> 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"> 1. Pengguna berada di halaman utama 2. Pengguna melakukan tap pada salah satu item rumah 3. Sistem menjalankan Tur Interaktif dari objek 3 dimensi rumah.
Post Condition	Sistem menampilkan Tur Interaktif <i>Unity</i>
Alternative Flow	-

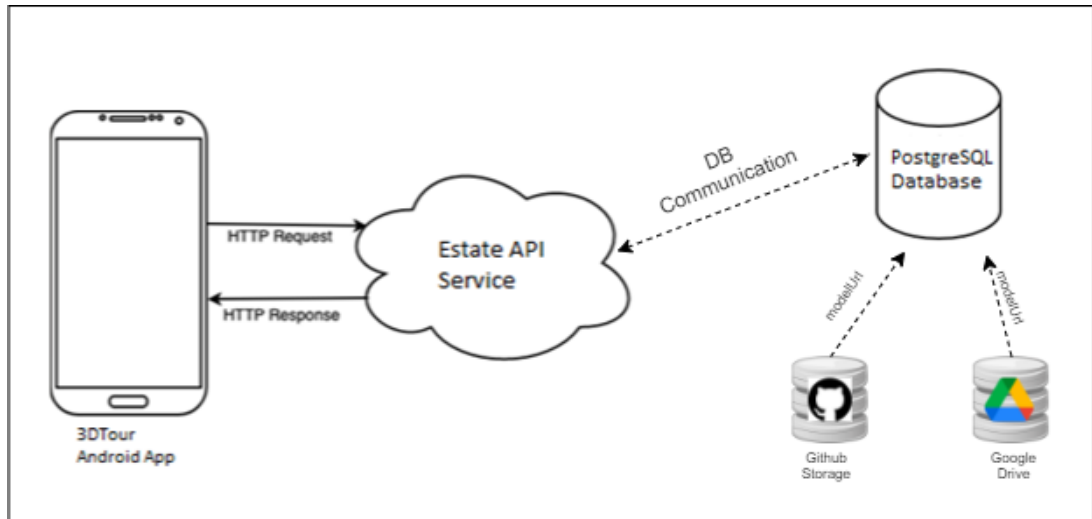
Tabel 4.7 Use Case Scenario Menambah Data Rumah

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"> 1. Admin membuka halaman utama 2. Sistem menampilkan form tambah data rumah 3. Admin melakukan pengisian data 4. Admin melakukan proses submit dan sistem memproses <i>request</i>
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

Gambar 5 menunjukkan desain arsitektur sistem yang digunakan untuk penyelidikan ini. Aplikasi 3dTour dikomunikasikan melalui HTTP melalui layanan web. Metode pertukaran data dengan menggunakan format JSON. Tambahkan data aplikasi 3DTour. Kirim data. Menggunakan desain arsitektur MVP, aplikasi 3DTour dibangun. Setiap halaman berisi dua kelas utama, View dan Presenter, berdasarkan desain arsitektur ini, yang masing-masing merujuk satu sama lain melalui antarmuka. Halaman utama tempat antarmuka MainViewContract diimplementasikan oleh kategori MainActivity dan MainPresenterContract adalah contohnya. Dengan Kelas MainPresenter.

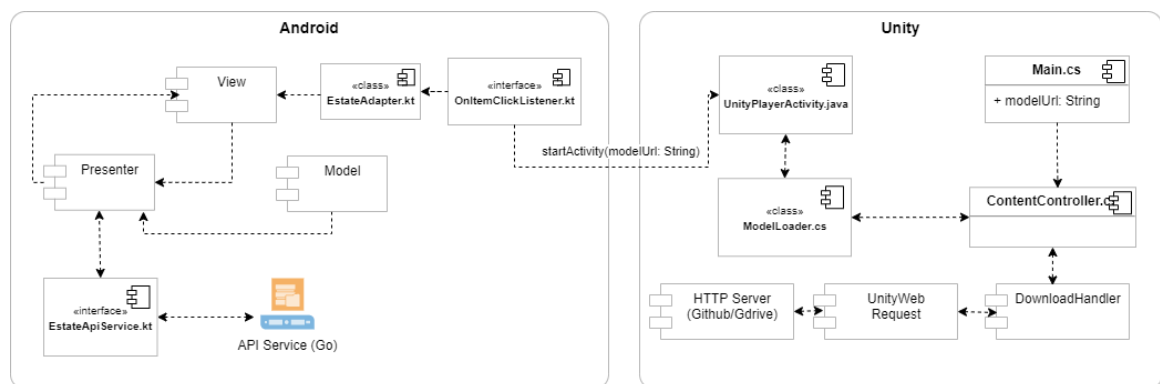
Mereka merujuk ke antarmuka bawaan, seperti halnya setiap kelas (MainActivity dan MainPresenter). Misalnya, MainActivity yang dimaksud oleh MainPresenter memiliki variabel instans HomePresenterContract dan sebaliknya. Pengembangan sistem API properti. Tidak ada desain tertentu, tetapi pemisahan terpusat masih tersedia antara pengontrol, model, pengontrol jalur HTTP dan database. Menurut permintaan pengguna, pengontrol melakukan beberapa pekerjaan. Model menunjukkan entitas yang berpartisipasi dalam sistem. Handler perutean HTTP berfungsi sebagai pengontrol

perutean untuk aplikasi HTTP masuk dan menautkannya ke pengontrol yang relevan. Objek database operasi database abstrak.

5.1.2 Perancangan Komponen

Rekayasa berbasis komponen perangkat lunak (CBSE) adalah industri dalam desain perangkat lunak yang menyoroti pemisahan masalah sehubungan dengan kemampuan luas yang dapat diakses di seluruh sistem perangkat lunak tertentu. CBSE adalah cabang pengembangan perangkat lunak. Ini adalah teknik berbasis penggunaan kembali di mana komponen individu didefinisikan, diimplementasikan dan dikompilasi secara longgar ke dalam sistem.

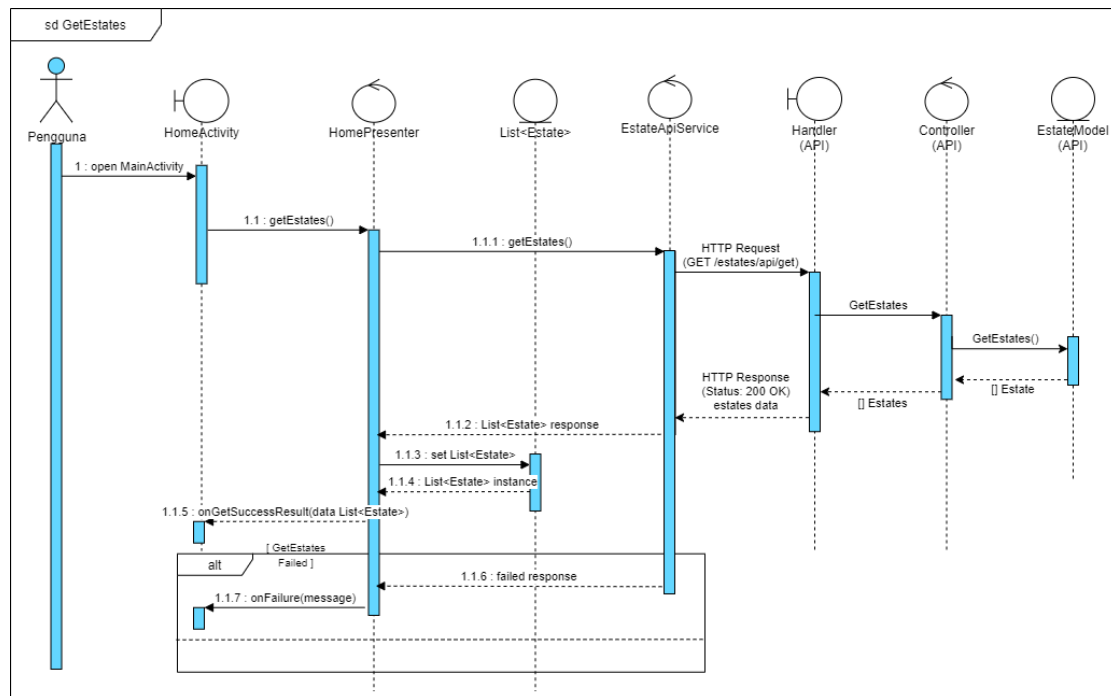
Pengaturan organisasi dan dependensi sekelompok komponen dalam sistem ditampilkan dalam diagram komponen atau diagram komponen. Diagram komponen berfokus pada komponen sistem yang diperlukan dan ada. Berikut ini adalah skema sistem untuk tur interaktif.



Gambar 5.2 Perancangan Komponen Android dan Unity

5.1.3 Sequence Diagram

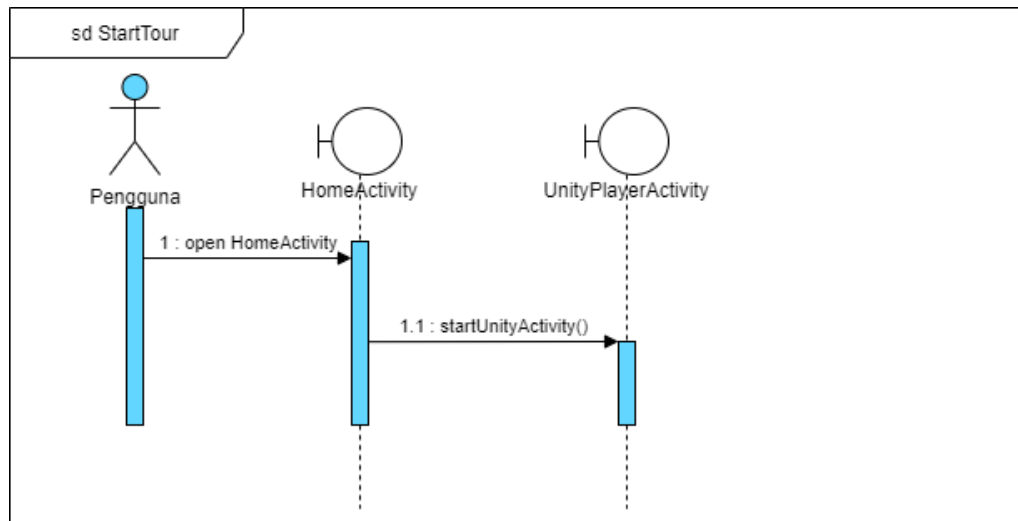
Diagram urutan dikembangkan untuk menggambarkan hubungan kelas aliran kerja. Dua proses ditampilkan dalam aplikasi Android, dan satu urutan ditampilkan dalam layanan API.



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`.



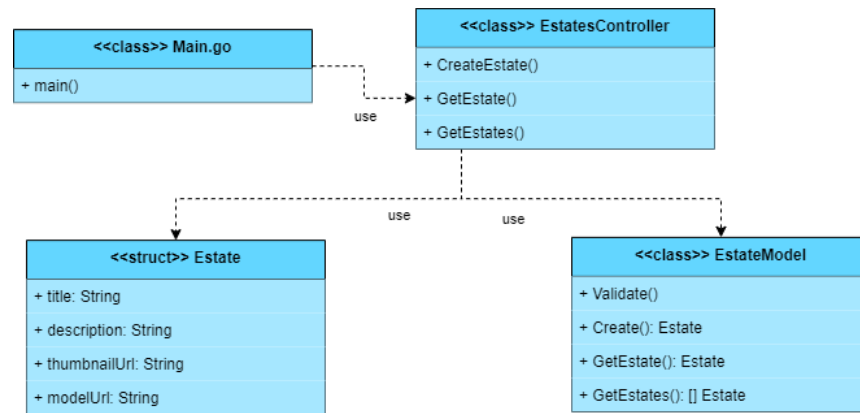
Gambar 5.4 Sequence Diagram Menampilkan Tur Interaktif

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.

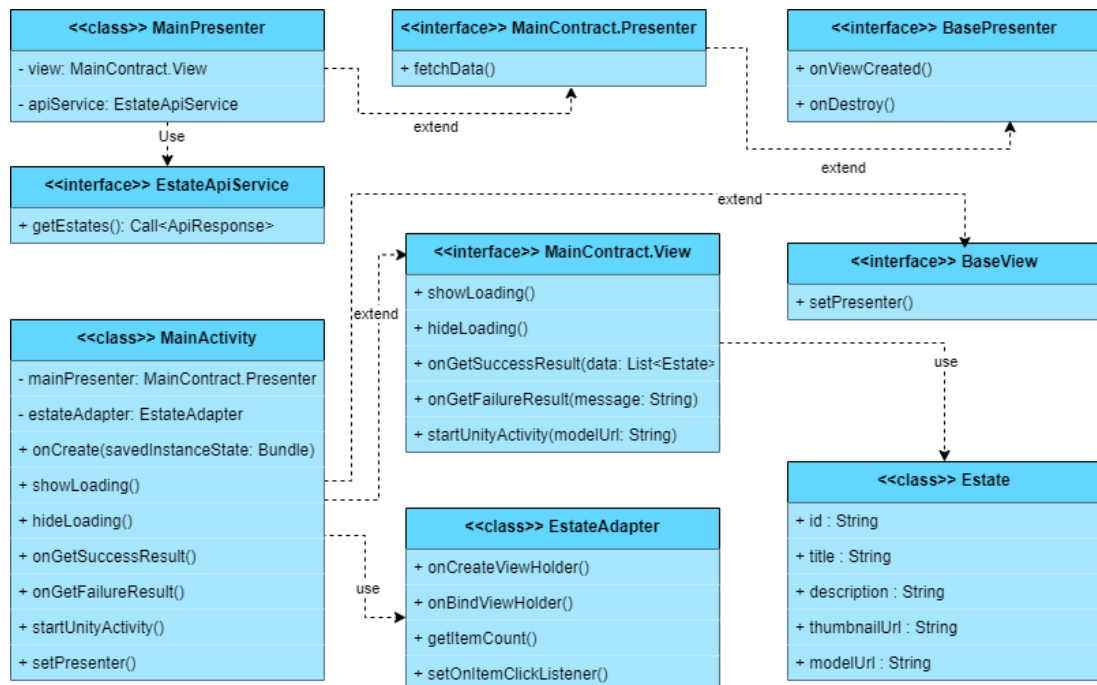
5.1.4 Class Diagram

Diagram kelas dimaksudkan untuk mewakili struktur dengan mengilustrasikan tautan antara kelas dan kelas, tindakan, dan atributnya dalam sistem desain. Bagian ini menyajikan dua diagram nilai, yaitu grafik dan diagram di Gambar 5.7. Gambar 5 memperlihatkan diagram class aplikasi Android.

Pemisahan antara Tampilan dan Penyaji dapat ditemukan di diagram kelas aplikasi Android. Begitulah pola arsitektur MVP digunakan. Kelas `renderer` yang menyertakan logika bisnis dialokasikan kelas bisnis sebagai tampilan.



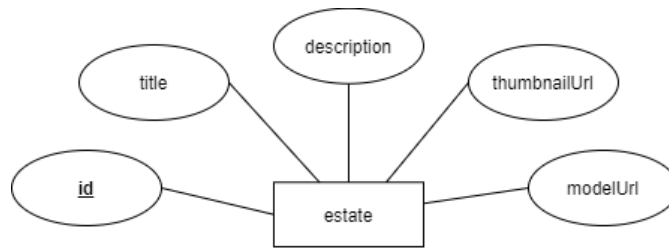
Gambar 5.5 Class Diagram API Service



Gambar 5.6 Class Diagram Aplikasi Android

5.1.5 Perancangan Basis Data

Desain database menentukan organisasi data dalam database. Seperti yang ditunjukkan oleh Gambar 5, database dibangun dalam bentuk ERD dan kemudian hasil ERD dinormalisasi ke dalam tabel. Tabel 5 memperlihatkan.



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

Algoritma ini dirancang untuk menawarkan gambaran umum proses sistem selama implementasinya. Teknik untuk mengunduh model rumah adalah metode dasar yang disebutkan dalam Tabel 5.2. Kemudian pendekatan berikutnya diberikan dalam Tabel 5.3 untuk mengenali item dalam aplikasi unit Aplikasi, yang diproduksi di dekat pemain.

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

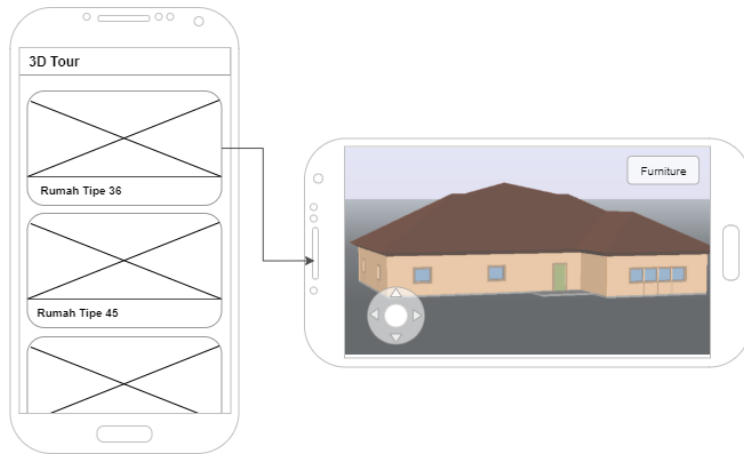
1	view = initiateView()
2	presenter = new Presenter(view)
3	dataResult: Array = presenter.getDataFromApiService()
4	recyclerView.adapter.setOnItemClickListener(new Listener())
5	
6	// onItemClick
7	Intent = Intent(UnityActivityPlayer)
8	intent.putData("modelUrl", itemData.modelUrl)
9	startActivity(intent)

Tabel 5.4 Rancangan Algoritme Menambah Data Rumah

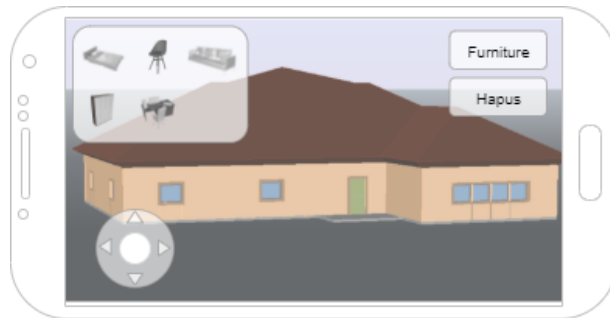
1	view = initiateView()
2	presenter = new Presenter(view)
3	
4	submitListener = onSubmit ((data) -> {
5	presenter.sendInsertDataToApiService(data, () -> {
6	presenter.onResultSuccess()
7	})
8	});
9	
10	view.setOnClickSubmit(submitListener)
11	view.setOnResultSuccess(() -> {
12	showToast("Success")
13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

5.1.7 Perancangan User Interface

Desain antarmuka berusaha untuk menawarkan ikhtisar mendasar dari sistem yang digunakan penggunaanya. Gambar 5.10 menunjukkan hasil dari desain antarmuka. Desainnya mencakup 2 halaman, 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

Pengembangan aplikasi Android Interactive Tour menggunakan desain arsitektur MVP, yang menyediakan aktivitas sebagai tampilan dan terhubung dengan kelas Penyaji yang mencakup logika aliran bahasa Go dan mengembangkan program menggunakan mesin Unity 3D dengan 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

Di lingkungan yang dijelaskan dalam Tabel 5.5 sistem dibuat dalam bentuk aplikasi dan layanan web berbasis Android. Tabel 5.6 menyediakan informasi tentang persyaratan perangkat keras untuk layanan web.

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

Tabel 5.7 menguraikan daftar perangkat lunak yang digunakan untuk membangun sistem Tur Interaktif. Tabel 5.8 menjelaskan konteks kerja layanan Web.

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"> 1. Android Studio 3.5 2. Visual Studio Code 3. Unity 3D Engine
Bahasa Pemrograman	<ol style="list-style-type: none"> 1. Kotlin 1.3.61 2. Go 1.12.5 3. C#

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

Ada item yang hanya dapat beroperasi pada perangkat berbasis android dengan versi 6.0 ke atas yang membuat batas implementasi dalam pengembangan Sistem Tur Interaktif.

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 Tour Interactive Android menggunakan Kotlin untuk pemrograman, C# untuk pengembangan mesin Unity 3D, Go untuk pengembangan layanan API. Kode program untuk aplikasi Android dijelaskan dalam Tabel 5.4-5. Mesin Unity 3D dibangun dan ditulis dalam C#. Tabel 5.8-5.11 memperlihatkan implementasi kode C#. Bahasa Go dari pemrograman layanan web. Tabel 5.12 memperlihatkan aplikasi kode buka. Untuk informasi selengkapnya lihat Tabel 5.14.

Tabel 5.10 Kode Program HomeActivity (Android)

1	class HomeActivity : AppCompatActivity(), MainContract.View {
2	private lateinit var materialDialog: MaterialDialog
3	private lateinit var mainPresenter: MainContract.Presenter
4	private lateinit var estateAdapter: EstateAdapter
5	
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 Balimbing 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 onGetSuccessResult(data: List<Estate>?) {
46	estateAdapter.data = data
47	estateAdapter.notifyDataSetChanged()
48	
49	}
50	
51	override fun onGetFailureResult(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) :
2	RecyclerView.Adapter<EstateAdapter.ViewHolder>() {
3	var data: List<Estate>? = ArrayList()
4	var itemClickListener: OnItemClickListener? = null
5	
6	override fun onCreateViewHolder(parent: ViewGroup, p1: Int):
7	ViewHolder {
8	val view =
9	LayoutInflater.from(context).inflate(R.layout.item_layout, parent,
10	false);
11	return ViewHolder(view)
12	}
13	
14	override fun onBindViewHolder(holder: ViewHolder, position: Int) {
15	holder.itemView.title.text = data?.get(position)?.title ?: ""
16	val options: RequestOptions = RequestOptions()
17	.centerCrop()
18	.placeholder(R.mipmap.ic_launcher_round)
19	.error(R.mipmap.ic_launcher_round)
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 /
25	moveInputDeadZone, 2);
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
49     void GetTouchInput()
50     {
51         for (int i = 0; i < Input.touchCount; i++)
52         {
53
54             Touch t = Input.GetTouch(i);
55
56             switch (t.phase)
57             {
58                 case TouchPhase.Began:
59                     if (t.position.x < halfScreenWidth &&
leftFingerId == -1)
60                     {
61                         leftFingerId = t.fingerId;
62                         moveTouchStartPosition = t.position;
63                     }
64                     else if (t.position.x > halfScreenWidth &&
rightFingerId == -1)
65                     {
66                         rightFingerId = t.fingerId;
67                     }
68
69                     break;
70                 case TouchPhase.Ended:
71                 case TouchPhase.Canceled:
72
73                     if (t.fingerId == leftFingerId)
74                     {
75                         leftFingerId = -1;
76                     }
77                     else if (t.fingerId == rightFingerId)
78                     {
79                         rightFingerId = -1;
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,
107	90f);
108	playerCamera.transform.localRotation =
109	Quaternion.Euler(cameraPitch, 0, 0);
110	transform.Rotate(transform.up, lookInput.x);
111	}
112	void Move()
113	{
114	if (moveInput.sqrMagnitude <= moveInputDeadZone) return;
115	
116	Vector2 movementDirection = moveInput.normalized * moveSpeed
117	* Time.deltaTime;
118	characterController.Move(transform.right * movementDirection.x + transform.forward * movementDirection.y);
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,
7	UnityAction<GameObject> callback, Transform bundleParent)
8	{
9	StartCoroutine(GetDisplayBundleRoutine(assetName, callback,
10	bundleParent));
11	}
12	string GetFilePath(string url)
13	{
14	string[] pieces = url.Split('/');
15	string filename = pieces[pieces.Length - 1];
16	return Path.Combine(Application.persistentDataPath, filename);
17	}
18	IEnumerator GetDisplayBundleRoutine(string bundleURL,
19	UnityAction<GameObject> callback, Transform bundleParent)
20	{

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,	
37	bundleParent);	
38	bundle.Unload(false);	
39	callback(arObject);	
40	}	
41	else	
42	{	
43	Debug.Log("Not a valid asset bundle");	
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		
	ProgressIndicator.text	=
56	((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 ==
43 availableItems[j].itemName)
44                 {
45                     RaycastHit hit;
46                     Ray ray =
playerController.playerCamera.ViewportPointToRay(new Vector3(0.5F, 0.5F,
47 0));
48
49                     if (Physics.Raycast(ray, out hit,
50 Mathf.Infinity))
51                     {
52                         Transform objectHit = hit.transform;
53                         Vector3 hitPoint = hit.point;
54                         hitPoint += objectHit.up;
55                         Instantiate(availableItems[j], hitPoint,
transform.rotation);
56                         // Instantiate(availableItems[j],
playerController.playerCamera.transform.position +
57 (playerController.playerCamera.transform.forward), Quaternion.identity);
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             }
115             else
116             {
117                 detectedItem = null;
118             }
119         }
120     }
121
122     void OnGUI()
123     {
124         GUI.color = detectedItem ? Color.green : Color.white;
125         GUI.DrawTexture(new Rect(Screen.width / 2 - 4, Screen.height
126         / 2 - 4, 8, 8), crosshairTexture);
127         GUI.color = Color.white;

```

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/utils"
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
payload"), false
20     }
21
22     if estate.Description == "" {
23         return u.Message(false, "Description should be on the
payload"), false
24     }
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 = ?",
id).First(estate).Error
44     if err != nil {
45         fmt.Println(err)
46         return nil
47     }
48     return estate
49 }
50
51 func GetEstates() []*Estate {
52     estates := make([]*Estate, 0)
53     err := GetDB().Table("estates").Find(&estates).Error
54     if err != nil {
55         fmt.Println(err)
56         return nil
57     }
58
59     return estates
60 }
61 }

```

5.2.5 Implementasi Antarmuka

Antarmuka aplikasi interaktif berbasis Android menggunakan format XML untuk mengatur tata letak setiap tampilan. Alur pengalihan halaman diperlihatkan dalam Gambar 5.8. Hasil implementasi ditampilkan dalam dua halaman: halaman beranda dan halaman tur interaktif, ditampilkan di Gambar 5.10 dan 5.11.

Gambar 5.10 Implementasi Antarmuka Home

Gambar 5.11 Implementasi Antarmuka Tur Interaktif

BAB 6 PENGUJIAN

6.1 Pengujian *Black-box*

Pengujian kotak hitam dilakukan untuk memverifikasi bahwa fungsi yang dihasilkan memenuhi kebutuhan fungsional. Pengujian dilakukan dalam skenario kasus dengan mengikuti alur yang ada. Temuan tes dalam tabel 6.1–6.3.

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"> 1. Pengguna berada pada halaman home. 2. Pengguna melakukan <i>tap</i> pada salah satu <i>item</i> data rumah. 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.
	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"> 1. Pengguna berada pada halaman home. 2. Pengguna melakukan <i>tap</i> pada salah satu <i>item</i> data rumah. 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.
	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"> 1. Pengguna memiliki koneksi ke jaringan internet. 2. Pengguna berada pada halaman <i>home</i> dari aplikasi Admin. 3. Pengguna mengisi form data 4. Pengguna melakukan submit untuk mengirim data ke <i>API Service</i>.
	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"> 1. Pengguna tidak memiliki koneksi ke jaringan internet. 2. Pengguna berada pada halaman <i>home</i> dari aplikasi Admin. 3. Pengguna mengisi form data 4. Pengguna melakukan submit untuk mengirim data ke <i>API Service</i>.
	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

Tes kotak hitam dilakukan untuk memeriksa apakah sistem yang dibangun telah memenuhi kebutuhan yang diidentifikasi. Diindikasikan bahwa semua ini dengan 6 kasus tes yang diambil dari aplikasi operasi sah berdasarkan tes yang dilakukan sesuai dengan yang ada di Tabel 6.1.

6.2 Pengujian Usability

Pengujian kegunaan adalah metode di mana sekelompok pengguna menguji seberapa mudah desain digunakan. Pengguna umumnya perlu diamati ketika mereka mencoba melakukan pekerjaan dan untuk banyak desain berbeda yang dapat mereka lakukan. Seringkali dilakukan lagi dan lagi, dari pengembangan asli hingga pengenalan produk. Pendekatan pengujian adalah menyediakan kemungkinan pengguna program untuk dibuat / dikembangkan dengan skenario tugas. Skenario tugas yang dilakukan oleh pengguna yang mungkin adalah langkah-langkah untuk dijalankan program, terutama langkah-langkah pada fungsionalitas program.

Dalam penelitian ini ada dua macam pengguna, yaitu pengguna dan administrator. Di Tabel 6.4 pengguna bisa menampilkan skenario tugas dan dalam tabel tugas admin Tabel 6.5 diperlihatkan.

Tabel 6.4 Task Scenario Untuk Pengguna

<i>Task</i>	Nama <i>Task</i>	Langkah-langkah
1	Menampilkan daftar rumah	1. Berada di halaman utama 2. Sistem menampilkan daftar rumah
2	Menampilkan tur interaktif	1. Pengguna menunggu proses menampilkan daftar rumah selesai 2. Pengguna melakukan <i>tap</i> pada salah satu <i>item</i> data rumah 3. Sistem menampilkan tur interaktif dengan membuka <i>activity</i> yang dikembangkan menggunakan Unity 3D Engine.

Tabel 6.5 Task Scenario untuk Admin

<i>Task</i>	Nama <i>Task</i>	Langkah-langkah
1	Menambah data rumah	1. Berada di halaman utama 2. Admin mengisi form 3. Admin menekan tombol <i>submit</i>

Seperti yang dinyatakan dalam tabel di atas, Tabel 6.4 adalah skenario tugas yang ditawarkan kepada pengguna tur interaktif, sementara Tabel 6.5 adalah skenario kerja admin.

Ada juga kuesioner yang akan diselesaikan oleh responden dan kemudian digunakan dalam penelitian ini untuk mengevaluasi kegunaan aplikasi. Kuesioner SUPR-Qm yang digunakan untuk setiap jenis pengguna aplikasi adalah kuesioner 12, seperti yang

ditunjukkan dalam Tabel 6.7 untuk pengguna aplikasi interaktif dan dalam Tabel 6.8 untuk admin aplikasi Interactive Tour untuk setiap aplikasi. Setiap pertanyaan memiliki skor dalam rentang 1 hingga 5 yang ditentukan dalam tabel 6.6 menggunakan skala Likert.

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

Dalam Persamaan 6.1, di bawah ini dijelaskan format perhitungan SUPR-Q yang digunakan dalam mendapatkan temuan kuantitatif dari hasil penghitungan nilai kuesioner SUPR-Qm pengguna.

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

(6.1)

Rumus tersebut nantinya akan digunakan dalam perhitungan SUPR-Qm, yang mana hasilnya akan diubah / konversi lagi ke dalam bentuk kategori dari nilai *usability*.

6.2.1. Analisis Hasil Pengujian Usability Aplikasi untuk Pengguna Tur Interaktif

Lima responden yang menggunakan aplikasi Interactive Tour melakukan tes kegunaan untuk pengguna. Secara acak, responden diambil sebagai pencari rumah untuk perumahan Balimbing Permai kepada konsumen potensial. Tabel 6.9 menjelaskan lima responden.

Tabel 6.9 Responden Pengujian *Usability* Pengguna Tur Interaktif

No	Nama	Latar Belakang
1		
2		
3		
4		
5		

Responden kemudian diminta untuk menjalankan program, dan moderator menganalisis perilaku responden untuk menilai apakah responden dapat memenuhi skenario tugas itu. Tabel 6.10 menggambarkan hasil dari situasi tugas yang diselesaikan.

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%

Berdasarkan Tabel 6.10, semua responden memiliki tingkat penyelesaian 100% dari kedua skenario tugas yang ditawarkan oleh moderator. Responder kemudian diminta untuk mengisi kuesioner SUPR-Qm yang telah ditentukan sebelumnya setelah menyelesaikan skenario kerja yang diberikan.

Hasil kuesioner SUPER-Qm yang diselesaikan oleh responden pelanggan tertuang dalam LAMPIRAN C dan hasil tes kuesioner SUPR-Qm terkandung dalam Tabel 6.11 untuk pengguna.

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

No	Keterangan	Skor	
----	------------	------	--

		1	2	3	4	5	Total Skor
1	Pertanyaan 1	0	0	0	5	0	20
2	Pertanyaan 2	0	0	2	3	0	18
3	Pertanyaan 3	0	0	1	3	1	20
4	Pertanyaan 4	0	0	1	2	2	21
5	Pertanyaan 5	0	0	0	4	1	21
6	Pertanyaan 6	0	0	0	3	2	22
7	Pertanyaan 7	0	1	0	3	1	19
8	Pertanyaan 8	0	0	2	1	2	20
9	Pertanyaan 9	0	0	2	0	3	21
10	Pertanyaan 10	0	0	1	2	2	21
11	Pertanyaan 11	0	0	1	3	1	20
12	Pertanyaan 12	0	0	0	1	4	24
Total Akhir							247
Total Nilai Maksimum							300
Nilai SUPR-Qm							82,33%

Skor teratas untuk nomor pertanyaan SUPR-Qm 12 (24), dan skor terendah untuk nomor pertanyaan 2 SUPR-Qm ditampilkan di Tabel 6.11. (18). Nilai akhir dari angka SUPR-Qm adalah 82,33 persen seperti yang ditunjukkan pada tabel 247.

Angka ini tergolong nilai kegunaan setelah memperoleh skor SUPR-Qm akhir sebesar 82,33%. Kategori B dengan predikat good degree diberikan kepada interpretasi 82,33% dari bidang nilai kegunaan. Sebagai konsekuensinya, perangkat lunak Interactive Tour cukup baik dan interaktif konsumen tur merangkulnya.

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

Admin aplikasi tur interaktif menerima salah satu balasan dengan pengujian kegunaan. Responden kemudian diminta untuk menjalankan program, dan moderator menganalisis perilaku responden untuk menilai apakah responden dapat memenuhi skenario tugas itu. Hasil penyelesaian skenario tugas diperlihatkan dalam tabel 6.12.

Tabel 6.12 Task Completion Rate

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

Berdasarkan Tabel 6.12, skenario pekerjaan moderator dengan tingkat penyelesaian 100 persen mungkin semua responden selesai.

Responden menyelesaikan skenario pekerjaan dan kemudian mengisi kuesioner SUPR-Qm yang ditentukan. LAMPIRAN D mencakup temuan balasan kuesioner SUPR-Qm admin aplikasi tur interaktif dan Tabel 6.13 memberikan hasil untuk administrator aplikasi turnamen interaktif tes kuesioner SUPR-Qm.

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

No	Keterangan	Skor					Total Skor
		1	2	3	4	5	
1	Pertanyaan 1	0	0	1	0	0	3
2	Pertanyaan 2	0	0	0	0	1	5
3	Pertanyaan 3	0	0	0	0	1	5
4	Pertanyaan 4	0	0	1	0	0	3
5	Pertanyaan 5	0	0	0	1	0	4
6	Pertanyaan 6	0	0	0	0	1	5
7	Pertanyaan 7	0	0	0	0	1	5
8	Pertanyaan 8	0	0	0	0	1	5
9	Pertanyaan 9	0	0	0	0	1	5
10	Pertanyaan 10	0	0	0	0	1	5
11	Pertanyaan 11	0	0	0	0	1	5
12	Pertanyaan 12	0	0	0	0	1	5

Total Akhir	55
Total Nilai Maksimum	60
Nilai SUPR-Qm	91,67%

Tabel 6.13 mengungkapkan bahwa skor terbaik dari 12 pertanyaan adalah 5, sementara skor terkecil adalah 3 untuk Pertanyaan 1 dan 4. Tabel di atas menampilkan nilai akhir 60, memperlihatkan nilai 91,67% untuk SUPR-Qm.

Jumlah ini diubah menjadi kategori kegunaan setelah hasil SUPR-Qm akhir sebesar 91,67 persen. Interpretasi 91,67 persen dalam kategori nilai kegunaan adalah Kategori A dengan peringkat kata sifat terbesar yang dapat dibayangkan. Ini membuat aplikasi tur interaktif Admin sangat baik dan dapat diterima.

6.3 Pengujian Compatibility

Pengujian aplikasi pada lima perangkat instalasi unik akan dilakukan. Beberapa kasus pengujian dan hasil uji kompatibilitas dengan berbagai aplikasi dan persyaratan operasi telah dilaporkan valid. Di mana hasil tes kompatibel. Tes akan dilakukan dengan kasus tes dan hasil tes dibandingkan dengan tes. (Zhang dan lainnya, Maret 2015).

Uji kompatibilitas dilakukan pada enam perangkat yang berbeda dan dapat digunakan untuk menampilkan layar yang cocok dengan implementasi sebelumnya. Temuan dan analisis tes kompatibilitas menunjukkan bahwa aplikasi ini sangat kompatibel dengan tingkat keberhasilan 80 persen dengan peralatan pengujian yang ditentukan. Aplikasi ini dapat dinyatakan cukup konsisten dengan kesimpulan ini.

6.3.1. Spesifikasi Perangkat Uji

Pengujian ini menggunakan satu perangkat keras dan lima perangkat virtual yang secara otomatis diproduksi dan dioperasikan menggunakan laboratorium uji firebase. Pengujian ini menggambarkan perubahan dalam versi dan resolusi Android berdasarkan data yang dikumpulkan (StatCounter, 2017). Berikut ini menjelaskan peralatan pengujian yang digunakan untuk pengujian Interactive Tour.

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)

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

6.3.2. Analisis Hasil Pengujian *Compatibility*

Berdasarkan pengujian pada 5 jenis perangkat dengan karakteristik sistem operasi yang bervariasi, aplikasi Interactive Tour tidak dieksekusi dengan 4 perangkat dan 1 perangkat. Berdasarkan data Tab 6.14 hingga Table 6.18, Android versi 6.0 hingga Android 9 adalah versi sistem operasi Android yang dapat menjalankan aplikasi Interactive Tour.

BAB 7 PENUTUP

Bagian ini menyoroti temuan dan ide untuk penelitian tentang pengembangan aplikasi tur interaktif.

7.1 Kesimpulan

Tiga temuan dicapai berdasarkan temuan analisis kebutuhan, desain, eksekusi dan pengujian studi pengembangan pariwisata interaktif :

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. Hasil penelitian mengungkapkan bahwa validasi yang dilakukan pada persyaratan fungsionalitas disertifikasi valid berdasarkan tes validasi yang dilakukan pada implementasi sistem yang dihasilkan.
3. Hasil penelitian menunjukkan bahwa aplikasi turnamen pengguna interaktif dapat digunakan pada 82,25 persen dan di bawah kategori B dengan peringkat kata sifat yang luar biasa berdasarkan pengujian kegunaan. Aplikasi tur interaktif untuk administrator dapat digunakan dengan peringkat kata sifat 96,25 persen, yang termasuk dalam kategori A. Dengan demikian aplikasi tur interaktif dapat disimpulkan berharga bagi pengguna dan administrator.

7.2 Saran

Penulis menganggap bahwa penelitian ini tidak dapat dikatakan sempurna, berdasarkan penelitian yang dilakukan. Akibatnya, ide-ide mungkin terbentuk, termasuk untuk penelitian lebih lanjut, 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.

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.

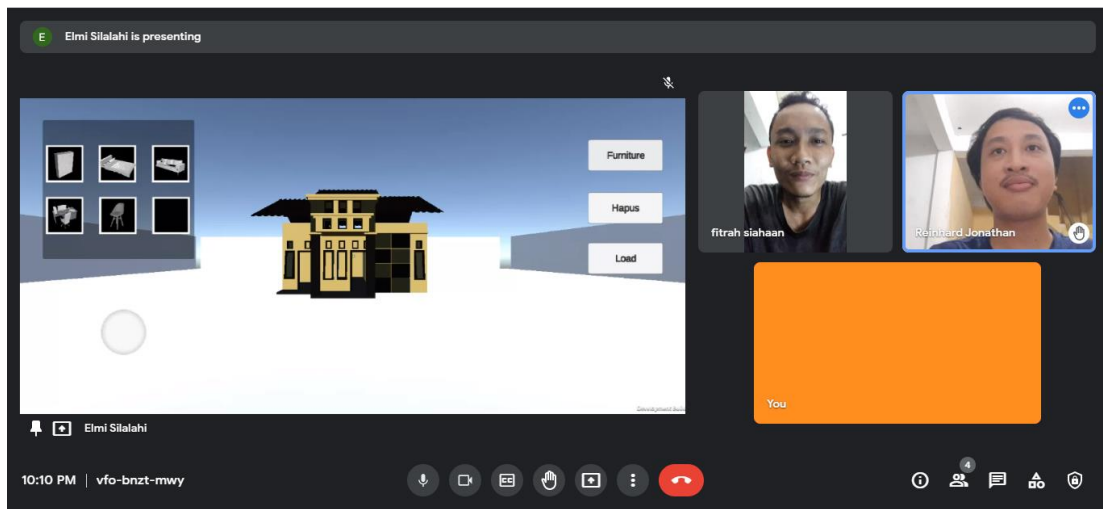
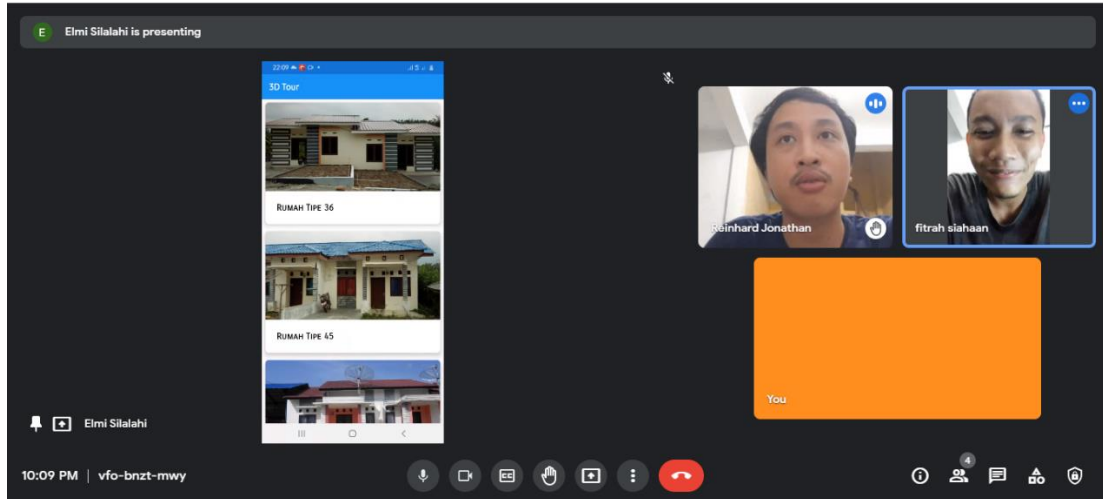
Balimbingan, 21 September 2020

PT. Karya Propertindo Utama,

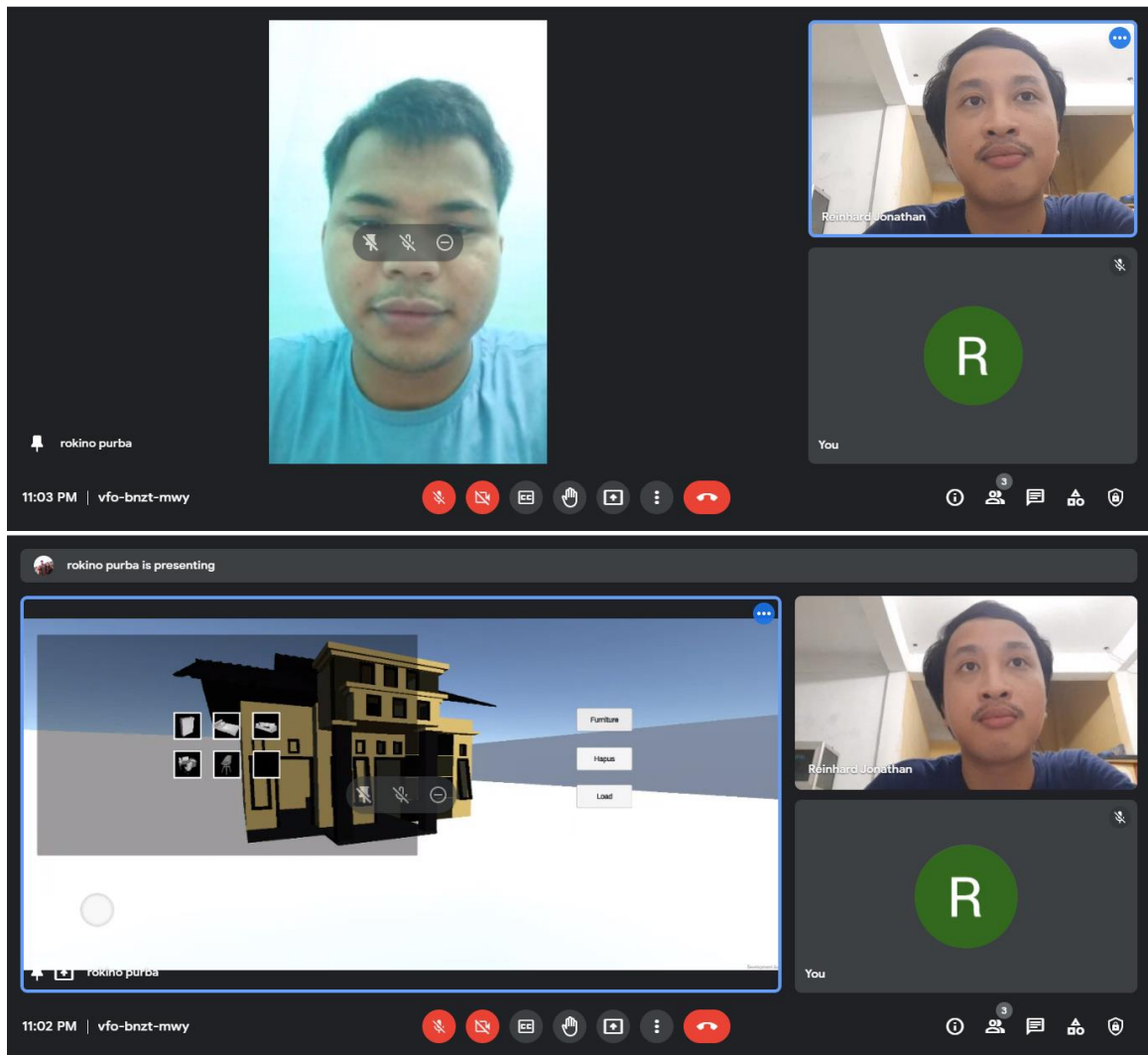

 Ir. Mangantar Silalahi, MT
 Pemilik Proyek

LAMPIRAN B DEMONSTRASI PRODUK

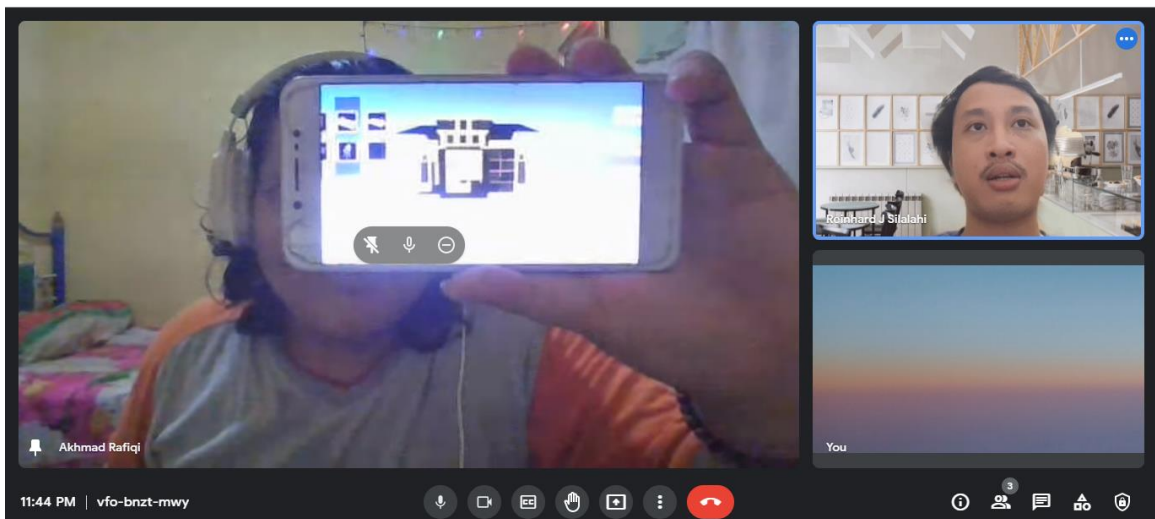
B.1 Demonstrasi Produk Responden 1 sebagai Pengguna



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