

**INTEGRASI AUGMENTED REALITY DAN ANALISIS POLA STRING UNTUK
MENINGKATKAN PENGALAMAN BERBELANJA
PRODUK SECARA ONLINE**

SKRIPSI

SAFUTRA SETYA ADINATA

201402066



PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS SUMATERA UTARA

2025

**INTEGRASI AUGMENTED REALITY DAN ANALISIS POLA STRING
UNTUK MENINGKATKAN PENGALAMAN BERBELANJA
PRODUK SECARA ONLINE**

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah Sarjana
Teknologi Informasi

SAFUTRA SETYA ADINATA

201402066



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

2025

PERSETUJUAN

Judul

: Integrasi Augmented Reality dan Analisis Pola String
untuk Meningkatkan Pengalaman Berbelanja Produk
Secara Online

Kategori

: Computer Vision and Multimedia

Nama Mahasiswa

: Safutra Setya Adinata

Nomor Induk Mahasiswa

: 201402066

Program Studi

: Teknologi Informasi

Fakultas

: Ilmu Komputer dan Teknologi Informasi Universitas
Sumatera Utara

Medan, 10 Januari 2025

Komisi Pembimbing :

Pembimbing 2



Baihaqi Siregar S.Si., M.T.

Diketahui/disetujui oleh

Teknologi Informasi

Ketua,


Dedy Arisandi S.T., M.Kom.
197908312009121002

Pembimbing 1



Ivan Jaya S.Si., M.Kom.

PERNYATAAN

**INTEGRASI AUGMENTED REALITY DAN ANALISIS POLA STRING UNTUK
MENINGKATKAN PENGALAMAN BERBELANJA PRODUK SECARA ONLINE**

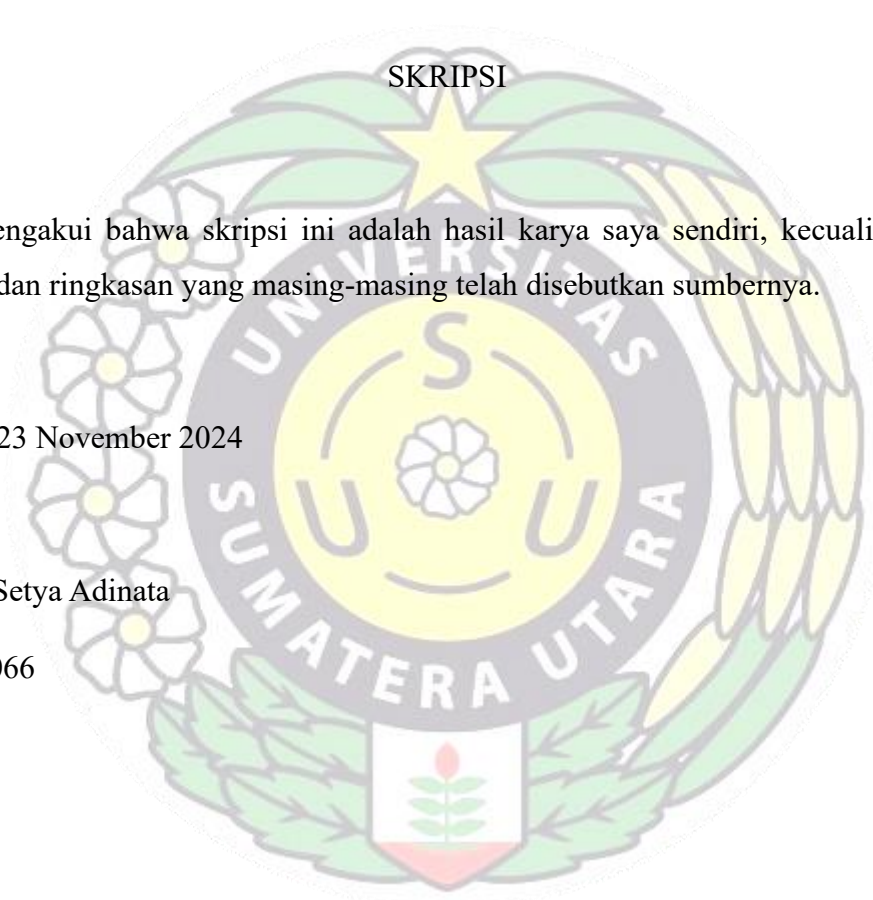
SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 23 November 2024

Safutra Setya Adinata

201402066



UCAPAN TERIMAKASIH

Penulis bersyukur atas nikmat Allah SWT yang Maha Pengasih dan Maha Penyayang, yang senantiasa memberkahi, memberik rezeki, rahmat, serta ridho-Nya sehingga penulis dapat menyelesaikan skripsi ini sebagai salah satu syarat untuk lulus dan mendapatkan gelar Sarjana Komputer dalam program studi S1 Teknologi Informasi di bawah Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Penulis ingin menyampaikan terimakasih sebesar-besarnya kepada semua pihak atas segala dukungan , bantuan dan serta doa yang diberikan kepada penulis dalam menyelesaikan skripsi ini. Penulis dengan tulus mengakui bahwa pencapaian ini tidak akan tercapai tanpa bantuan dan dukungan dari semua pihak yang terlibat yaitu:

1. Penulis bersyukur kepada Allah SWT, karena dengan izin dan berkat-Nya penulis dapat menyelesaikan skripsi ini dengan baik.
2. Keluarga penulis Bapak Ir. Erly Hidayat Nurli dan Lely Safrida selaku orang tua penulis yang selalu memberikan doa dan dukungan tanpa balasan dan memberatkan penulis dalam menyelesaikan skripsi ini.
3. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Sumatera Utara
4. Bapak Dedy Arisandy S.T., M.Kom. selaku Ketua Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
5. Bapak Ivan Jaya S.Si., M.Kom. selaku Dosen Pembimbing I penulis yang telah memberikan dukungan dan bimbingan serta saran kepada penulis
6. Bapak Dr. Baihaqi Siregar S.Si., M.T. selaku Dosen Pembimbing II penulis yang telah memberikan dukungan kepada penulis
7. Saudara penulis Safelinda Setya Ningrum yang selalu mendukung serta memberikan doa kepada penulis
8. Teman-teman dekat penulis yaitu Immanuel Lamhot Parhorasan, Zhafran Alviansyah, Simon Bungaran Sihole, M. Fadil Ramadan, M. Naufal Surur Berutu yang selalu membatntu dan berjuang bersama penulis dalam menyelesaikan perkuliahan.

9. Penulis menyampaikan terima kasih kepada semua pihak yang telah berpartisipasi secara langsung dan yang secara tidak langsung , meskipun tidak dapat disebutkan satu per satu.

Penulis menyadari bahwa penulisa skripsi ini masih belum sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi perbaikan penyusunan Tugas Akhir Skripsi dimasa yang akan datang. Harapan penulis yaitu skripsi ini memberikan manfaat yang besar kepada pembaca dan juga dalam pengembangan ilmu pengetahuan bagi masyarakat.

Medan, 23 November 2024

Penulis



ABSTRAK

Belanja secara daring merupakan sebuah aktivitas yang memiliki batasan yang cukup besar dibandingkan belanja secara luring atau langsung. Salah satu batasan tersebut adalah tidak adanya cara untuk melihat barang secara langsung sebelum mengambil keputusan untuk membeli barang tersebut. Ini menghasilkan masalah yang memerlukan solusi yang kuat untuk dilewati. Teknologi *AR* bisa mengatasi masalah tersebut dengan memberikan sebuah gambaran visual bagaimana produk terlihat di dunia nyata melalui perangkat mobile pembeli. *Augmented Reality (AR)* adalah teknologi yang memberikan gambaran visual *3D* seolah-olah objek *3D* tersebut terletak di lingkungan dunia nyata. Penelitian ini menggunakan analisis pola *string* sebagai metode untuk mengambil ukuran produk dari situs yang ditentukan. Analisis *string* ini akan menghasilkan *regex* yang mengandung pola *string* ukuran produk yang bisa digunakan untuk mengidentifikasi ukuran produk dari *string* yang panjang. Implementasi *regex* ini digunakan di sebuah server *python* guna untuk mesimplifikasikan proses kerja sistem aplikasi. Aplikasi ini menggunakan *ARCore* dari *Unity* sebagai *engine* aplikasi di perangkat *mobile*. Penelitian ini menghasilkan informasi bahwa pengguna aplikasi memberikan nilai 90.41% pada aspek penggunaan, 86.66% untuk aspek kemudahan, 89.52% untuk aspek kepuasan. Sehingga kita bisa mengambil kesimpulan, aplikasi hasil dari integrasi *augmented reality* dan analisis pola *string* untuk meningkatkan pengalaman berbelanja secara *online* layak untuk digunakan.

Kata Kunci: *Augmented Reality, regex, python, string, Unity, ARCore, engine.*

INTEGRATION OF AUGMENTED REALITY AND STRING PATTERN ANALYSIS TO IMPROVE THE ONLINE SHOPPING EXPERIENCE

ABSTRACT

Online shopping is an activity with significant limitations compared to offline or in-person shopping. One of these limitations is the inability to view products directly before making a purchase decision. This challenge can be addressed with Augmented Reality (AR) technology, which provides a visual representation of how products appear in the real world through the buyer's mobile device. Augmented Reality (AR) is a technology that delivers 3D visual representations, making objects appear as though they are situated in a real-world environment. This research utilizes string pattern analysis as a method to extract product dimensions from a specified website. The string analysis produces regular expressions (regex) containing product dimension patterns, which can be used to identify product sizes from lengthy strings. The regex implementation is deployed on a Python server to simplify the application's workflow. This application leverages ARCore from Unity as its engine for mobile devices. The study reveals that application users rated it 90.41% in terms of usability, 86.66% for ease of use, and 89.52% for satisfaction. Therefore, it can be concluded that the application resulting from the integration of augmented reality and string pattern analysis is feasible for enhancing the online shopping experience.

Keywords: Augmented Reality, regex, Python, string, Unity, ARCore, engine.

DAFTAR ISI

	Halaman
PERSETUJUAN	i
PERNYATAAN	ii
UCAPAN TERIMAKASIH.....	iii
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Penelitian.....	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Metode Penelitian	4
1.7 Sistematika Penulisan	5
BAB 2 LANDASAN TEORI	7
2.1 Augmented Reality.....	7
2.2 <i>Marker-less AR dan Marker-based AR</i>	7
2.3 ARCore.....	8
2.4 Unity.....	8
2.5 Selenium.....	8
2.6 Flask	9
2.7 Regex.....	9
2.8 Penelitian Terdahulu.....	10
BAB III ANALISIS DAN PERANCANGAN SISTEM.....	12
3.1 Data yang Digunakan	12
3.2 Analisis Sistem.....	14
3.3 Perancangan Sistem.....	15
3.3.1 Arsitektur Umum	15

3.3.2 Flowchart Diagram	28
3.3.3 Activity Diagram.....	29
3.4 Perancangan Antarmuka.....	30
BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM	32
4.1 Implementasi Sistem	32
4.1.1 Spesifikasi Perangkat Keras.....	32
4.1.2 Spesifikasi Perangkat Lunak.....	32
4.1.3 Instalasi Aplikasi pada Perangkat <i>Mobile</i>	33
4.2 Tampilan Aplikasi	34
4.2.1 Tampilan Aktivasi akses Kamera.....	34
4.2.2 Tampilan Awal Aplikasi	35
4.2.3 Tampilan Halaman <i>AR</i>	36
4.3 Pengujian Kinerja Sistem.....	37
4.3.1 Pengujian Kinerja Antarmuka.....	38
4.3.2 Pengujian Komponen Aplikasi	39
4.3.3 Pengujian Penggunaan Aplikasi Terhadap Pengguna	42
BAB V KESIMPULAN DAN SARAN	48
5.1 Kesimpulan.....	48
5.2 Saran.....	49
DAFTAR PUSTAKA	50

DAFTAR GAMBAR

Gambar 3.1 Arsitektur Umum.....	16
Gambar 3.2 Input Link pada Menu Utama	17
Gambar 3.3 Input Marker Menggunakan Kamera Pengguna	18
Gambar 3.4 Input Dimensi Objek 3D Produk	18
Gambar 3.5 Input Posisi dan Rotasi Objek.....	19
Gambar 3.6 Tampilan Proses Mencari Marker	20
Gambar 3.7 Skrip Marker	20
Gambar 3.8 Skrip <i>API Requester</i>	21
Gambar 3.9 Skrip <i>Spawner</i>	22
Gambar 3.10 Skrip <i>UI Control</i>	25
Gambar 3.11 Ukuran <i>font</i> pada saat objek menjauhi kamera(a) dan saat objek mendekati kamera(b) sama.....	26
Gambar 3.12 Model produk berbentuk umum.....	27
Gambar 3.13 Flowchart Sistem ARView	28
Gambar 3.14 Activity Diagram.....	29
Gambar 4.1 Instalasi Aplikasi ARView.....	34
Gambar 4.2 Aplikasi meminta akses kamera.....	35
Gambar 4.3 Menu Aplikasi.....	36
Gambar 4.4 Berbagai mode di halaman <i>AR</i>	37
Gambar 4.5 Objek merah menandakan titik orientasi.	40
Gambar 4.6 Aplikasi sedang melakukan permintaan <i>API</i>	40
Gambar 4.7 Pengujian dilakukan oleh pengguna.	42
Gambar 4.8 Grafik tanggapan rekomendasi pengguna.....	46

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu.....	10
Tabel 3.1 Data <i>String</i> dan <i>Regex</i> dari Tokopedia.....	12
Tabel 3.2 Data String dan Regex yang Ditambahkan Oleh Penulis	13
Tabel 3.3 <i>Regex</i> Pencari Dimensi Produk yang Universal	14
Tabel 3.4 Desain Aplikasi	30
Tabel 4.1 Rancangan Pengujian Komponen	38
Tabel 4.2 Hasil Pengujian Antarmuka	38
Tabel 4.3 Tabel Percobaan Perbandingan Layar dan Besar Marker di Layar Perangkat Mobile	41
Tabel 4.4 Tanggapan Pengguna Terkait Aplikasi	43
Tabel 4.5 Nilai Kepuasan Aplikasi Berdasarkan Aspek.....	45



BAB I

PENDAHULUAN

1.1 Latar Belakang

Kegiatan yang dilaksanakan di abad ke-21 sangat berbeda dengan cara berbelanja di abad sebelumnya. Di masa ini, masyarakat di kebanyakan negara sudah saling terhubung satu sama lain melalui jaringan internet menorebus semua hambatan-hambatan fisik yang sebelumnya ada. Kini, segala kegiatan dianjurkan secara online dikarenakan kepraktisannya dan kemudahannya untuk hanya tetap di rumah dan berada di bawah atap yang sejuk. Berbelanja bukanlah pengecualian.

Di zaman ini, memesan sebuah produk atau membelinya dilakukan secara *online* melalui aplikasi toko *online* yang sudah bisa diakses dimana saja. Sangat mudah untuk kita mengerti bagaimana sebuah metode berbelanja ini sangat digemari orang banyak dikarenakan kemudahannya. Pembeli hanya perlu membuka aplikasi, mengisi data profil, pilih atau cari barang yang disukai lalu membelinya dengan uang elektronik atau *mobile-banking*. Tetapi, ada sisi buruk yang muncul dari metode berbelanja ini. Salah satunya adalah pembeli tidak bisa membeli produk setelah mengecek fisik produk terlebih dahulu. Karena membeli sebuah barang di toko *online* sangat berisiko akan penipuan, beberapa metode untuk mengamankan pembeli dibangun untuk menjaga kepercayaan pembeli. Namun tetap saja, walaupun penjual sudah memberikan bagaimana sebuah produk terlihat melalui foto dan deskripsi, pembeli masih mengalami kesulitan untuk memperkirakan bagaimana bentuk dan ukuran produk tersebut di dunia nyata hingga semua sudah terlambat. Ini semua bisa diatasi dengan mengimplementasikan teknologi *Augmented Reality* sebagai cara kita memperkirakan dimensi produk tersebut di dunia nyata.

Teknologi *AR* adalah sebuah teknologi interaksi antar manusia dan komputer yang mengintegrasikan dunia digital ke dalam dunia nyata (Zhao, 2018). Ponsel pintar yang keluar akhir-akhir ini sudah banyak mendukung *AR* sebagai sarana untuk mencoba hal-hal baru seperti menggambar di wajah seseorang melalui kamera depan hingga memperlihatkan bagaimana bentuk sebuah bangunan terlihat di dunia nyata. Teknologi

ini bisa menjadi jembatan untuk menyelesaikan masalah yang dialami pembeli yang sebelumnya sudah dijelaskan.

Penggunaan *AR* untuk memperlihatkan bagaimana sebuah produk terlihat memiliki banyak ketertarikan dari sisi pembeli. Di penelitian *The Intention to Use E-Commerce Using Augmented Reality - The Case of IKEA Place* oleh Alves (2020), terdapat ketertarikan oleh pembeli untuk membeli sebuah produk jika dibarengi dengan teknologi *AR* tersebut. Penelitian ini menunjukkan bahwa konsumen akan menjadi lebih percaya diri dan lebih berani untuk membeli sebuah produk jika sebelum membelinya konsumen bisa menggunakan aplikasi *AR* untuk melihat bagaimana bentuk sebuah produk di dunia nyata. Tak hanya itu, didalam jurnal *Augmented Reality in Real Stores: Empirical Evidence from Consumers' Interaction with AR in a Retail Format* oleh Bonetti, Pantano, Warnaby, Quinn, dan Perry di buku *Augmented Reality and Virtual Reality : The Power of AR and VR for Bussiness* oleh Dieck dan Jung (2019), kita dapat menyimpulkan bahwa ada faktor peningkatan pengalaman berbelanja dengan menggunakan lingkungan toko baru yaitu *AR*. Peserta survey di penelitian tersebut mengapresiasi pengalaman baru tersebut dan menganggapnya sebagai sesuatu yang menghibur, nyata dan nikmat. Di dalam lingkungan yang kompetitif, toko-toko pakaian eceran bahkan mulai mengadopsi teknologi-teknologi yang berinovasi untuk meningkatkan pengalaman belanja yang berujung ke peningkatan daya jual dan keuntungan kompetitif (Pantano & Priporas, 2016). Teknologi immersive seperti *AR* dan *VR* yang terus berkembang merupakan salah satu teknologi yang diadopsi di konteks tersebut (Bonetti, Warnaby, & Quinn, 2017; Javornik, 2016).

Pada penelitian *Deskkart- Furniture Mall with Augmented Reality* oleh Mintu Baruah (2019), telah dibuat sebuah aplikasi yang memungkinkan pembeli untuk melihat bagaimana sebuah produk terlihat di dunia nyata menggunakan teknologi *AR* sebelum pembeli tersebut memutuskan membeli sebuah barang. Tetapi, aplikasi dari penelitian ini merupakan sebuah platform shopping *online* dimana penjual harus memiliki akun untuk aplikasi tersebut. Penjual lalu diharuskan untuk mengupload model *3D* dari produk yang ingin dijual agar konsumen bisa memakai teknologi *AR* yang dimaksud. Ini merupakan hal yang bisa menghambat kegunaannya karena insentif untuk menggunakan teknologi *AR* berada di tangan penjual dan tidak di tangan konsumen.

Dari latar belakang diatas dilakukan sebuah penelitian dengan judul Integrasi *Augmented Reality* Dan Analisis Pola *String* Untuk Meningkatkan Pengalaman Berbelanja Produk Secara *Online* untuk mengembangkan sebuah aplikasi dimana pengguna atau pembeli bisa melihat bagaimana sebuah produk terlihat secara umum menggunakan *AR* dengan bermodal link produk yang diinginkan tanpa mengharuskan penjual untuk mengupload model *3D* dari produk tersebut. Dengan cara ini, insentif bisa diarahkan ke arah pembeli dan pembeli bisa menjadi lebih yakin untuk jadi membeli produk atau tidak.

1.2 Rumusan Masalah

Terdapat kekecewaan konsumen ketika membeli sebuah produk secara *online* hanya untuk mendapatkan produk yang telah dibeli tidak sesuai dengan yang dibayangkan atau dengan apa yang tertera di halaman produk seperti yang terjadi di berita yang dilansir di *suara..com* sehingga dibutuhkan sebuah sistem untuk memperkirakan bagaimana sebuah produk itu terlihat di dunia nyata tanpa harus membeli terlebih dahulu.

1.3 Batasan Penelitian

Pada penelitian ini, penulis membatasi cakupan penelitian untuk memfokuskan dan menghindari ketidak-relevan-an ruang lingkup permasalahan pada penelitian ini. Batasan masalah ini yaitu :

- A. Aplikasi *AR* yang ditampilkan hanya memberikan sebuah bentuk umum dari produk yang ingin ditampilkan.
- B. Aplikasi *AR* yang ditampilkan berfokus ke penerapan *markerbased AR tracking*.
- C. Aplikasi ini hanya menggunakan *Regular Expression* untuk mengidentifikasi *string* dimensi yang ada pada deskripsi produk.
- D. Aplikasi dibuat untuk perangkat *mobile* yang menggunakan sistem operasi *Android* yang telah didukung oleh *Google* untuk menggunakan *ARCore*.
- E. Aplikasi hanya akan menggunakan *website* Tokopedia sebagai *website scraping*.
- F. Aplikasi *AR* hanya berfokus untuk produk-produk furnitur.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah untuk mengintegrasikan teknologi *Augmented Reality* dan analisis pola *string* untuk meningkatkan pengalaman berbelanja produk secara *online*.

1.5 Manfaat Penelitian

1. Integrasi *Augmented Reality* Dan Analisis Pola *String* Untuk Meningkatkan Pengalaman Berbelanja Produk Secara *Online* diharapkan bisa menjadi manfaat bagi pengguna aplikasi belanja *online* untuk mengestimasi bagaimana sebuah produk terlihat di dunia nyata sebelum membelinya.
2. Integrasi *Augmented Reality* Dan Analisis Pola *String* Untuk Meningkatkan Pengalaman Berbelanja Produk Secara *Online* diharapkan bisa menjadi manfaat bagi pengembang aplikasi *AR* sebagai referensi untuk mengembangkan aplikasi *AR* dalam bidang *e-commerce*.

1.6 Metode Penelitian

1. Studi Literatur
Pada tahap ini penulis akan mencari dan mengumpulkan dokumen-dokumen yang berkaitan dengan penelitian dan pengembangan aplikasi *AR*, *ARCore*, *Unity* dan juga dokumen yang berkaitan dengan percobaan pembuatan aplikasi *AR* yang telah ada sebelumnya.
2. Analisis sistem
Di tahap ini penulis akan mengumpulkan informasi informasi terkait dengan teknis pembuatan aplikasi *AR* sesuai dengan apa yang dibutuhkan dalam proses pengembangannya. Informasi ini bisa berkaitan dengan perangkat keras, aplikasi perangkat lunak, dan implementasi metode.
3. Perancangan sistem
Berdasarkan evaluasi analisis sistem, penulis menyusun rancangan sistem aplikasi mulai dari arsitektur umum, penetapan data, hingga pengujian aplikasi.
4. Implementasi
Tahap ini akan diisi dengan implementasi metode untuk membuat aplikasi berdasarkan analisis sebelumnya.

5. Pengujian

Di tahap ini, penulis akan menguji aplikasi yang telah dibuat untuk memastikan aplikasi yang telah dibuat bekerja dengan baik dan sesuai dengan rancangan awal yang telah dibuat.

6. Dokumentasi dan Penyusunan Laporan

Pada tahap ini, penulis akan melakukan penyusunan laporan berdasarkan rancangan, implementasi, dan hasil akhir yang telah didapat dari pengembangan aplikasi *AR* ini dalam bentuk skripsi.

1.7 Sistematika Penulisan

Sistematika Penulisan untuk penelitian ini dibagi menjadi 5 bab pembahasan, berikut penjelasan singkat untuk tiap bab :

Bab 1 : Pendahuluan

Bab pendahuluan menjelaskan tentang latar belakang yang menjadi dasar dari judul penelitian, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

Bab 2 : Landasan Teori

Bab ini menyajikan berbagai teori yang terkait dengan penelitian sebagai dasar untuk memahami permasalahan yang dibahas dalam studi ini. Teori yang dimaksud menjelaskan beberapa hal seperti *Augmented Reality*, *ARCore*, *Unity* dan bagaimana penerapannya untuk menampilkan produk yang diinginkan.

Bab 3 : Analisis dan Perancangan Sistem

Bab ini menjelaskan detail-detail alur proses kerja dari arsitektur umum aplikasi yang dibuat. Ini termasuk bagaimana proses komunikasi antar aplikasi mobile, *API* dan server yang dijalankan.

Bab 4 : Implementasi dan Pengujian Sistem

Bab ini mengulas tentang penggunaan aplikasi dan penerapan system yang dijelaskan di bab 3.

Bab 5 : Kesimpulan dan Saran

Bab ini terdiri dari rangkuman dari penjelasan bab-bab yang sebelumnya dan saran yang ditujukan atau dari penulis untuk pengembangan penelitian selanjutnya.



BAB 2

LANDASAN TEORI

2.1 **Augmented Reality**

Augmented reality (AR) adalah sebuah teknologi yang memberikan sebuah gambaran objek digital sebagaimana bentuknya kedalam dunia nyata melalui sebuah layar. Teknologi ini berbeda dengan *Virtual Reality (VR)* yang justru membawa sang pengguna kedalam dunia digital untuk mendapatkan pengalaman di dunia digital.

Menurut Gangurde (2011), *Augmented Reality* adalah teknologi yang menyediakan dunia nyata sebuah informasi virtual yang terlihat saling berdampingan dengan lingkungan dunia nyata menggunakan inputan sensor seperti penglihatan, pendengaran dan sentuhan. Melalui inputan sensor tersebut, integrasi objek virtual 3D ke lingkungan ini terasa nyata karena dihadirkan secara real-time.

Ada tiga syarat agar sebuah sistem bisa disebut *AR*: pertama, teknologi tersebut harus menggabungkan dunia nyata dan dunia virtual, kedua, penngguna harus bisa berinteraksi secara real-time dengan sistem, ketiga, memberikan benntuk dalam 3 dimensi (Azuma, 1997).

2.2 **Marker-less AR dan Marker-based AR**

Aplikasi *AR* memiliki beberapa cara dalam membawa sebuah objek virtual ke dunia nyata. Beberapa aplikasi *AR* hanya akan bisa memunculkan objek 3D tersebut menggunakan sebuah bidang 2d tertentu yang berfungsi sebagai titik acuan lingkungan 3D di aplikasi *AR* tersebut ketika telah menerima *input* video (IPTEK, 2024). Teknik ini disebut sebagai teknik *Marker-based*. Dinamakan *Marker-based* karena tanpa sebuah titik acuan, aplikasi tersebut tidak tahu dimana objek virtual seharusnya diletakkan.

Beda hal dengan *Marker-less*, teknik ini memberikan kebebasan yang lebih luas dalam menentukan titiik acuannya karena tidak membutuhkan acuan buatan di dunia nyata untuk menentukan dimana letak objek digital tersebut. Teknik *marker-less* memanfaatkan, kamera dan sensor inersia, agar bisa memetakan titik-titik orientasi dari

sebuah *input* video menjadi sebuah *landscape 3D* dimana objek digital bisa disimulasikan seolah ada di dunia nyata (Iptek Digital, 2024).

2.3 ARCore

ARCore adalah sebuah library yang dibuat oleh *Google* yang bisa diakses diaplikasi *Unity* untuk mensimulasikan *Augmented Reality* didalam sebuah sistem. *ARCore* hadir dalam bentuk *SDK* yang bisa dipakai untuk berbagai platform alias *cross-platform*. *ARCore* memiliki berbagai fitur mulai dari *motion tracking*, *anchors*, *environmental understanding*, *depth understanding*, dan *light estimation* (Google, 2024).

2.4 Unity

Unity adalah injin untuk membuat sebuah aplikasi atau game yang dikembangkan oleh *Unity Technologies* yang diluncurkan oleh *Apple Inc.* pada bulan Juni tahun 2005. Sebelumnya *Unity* hanyalah innjin permainann yang eksklusif untuk *Mac OS X* namun seiring waktu telah dibuat dukungan dukungan untuk platform lain untuk memberikan variasi dan mengembangkan cakupan injin game tersebut. *Unity* mampu membantu pengembang untuk mengembangkan game *2D*, *3D*, *Virtual Reality*, dan *Augmented Reality*.

Unity adalah aplikasi yang digunakan untuk membuat game, baik game *3D* ataupun *2D* (Akbar Project, 2019). Ini berarti *Unity* adalah sebuah mesin telah menyediakan sebuah wadah atau lingkungan yang telah didesain untuk membangun sebuah aplikasi dan games sehingga programmer dan pengembang lainnya tidak perlu menkoding dari kertas baru untuk membuat aplikasi tersebut. *Unity* hadir dalam bahasa pemrograman *C#* dimana memiliki kemiripan dalam hal *syntax* dengan bahasa pemrograman *C++* dan *Java*. Kemiripan ini akan sangat memudahkan penulis untuk memahami cara kerja bahasa pemrograman tersebut.

2.5 Selenium

Selenium adalah *framework* yang bersifat *open-source* untuk mengotomasi browser (Moraneus, 2024). *Selenium* dimulai dari tahun 2004 di *ThoughtWorks* di Chicago dengan Jason Huggins yang telah memangung kodingan intinnya unnutk

mengetes waktu internal dan pengeluaran dari *Python* dan *Plone*. Otomisasi testing adalah inti dari *ThoughtWork* oleh karena itu test ini memang salah satu test yang harus dilakukan oleh perusahaan tersebut.

Kini Selenium hadir dalam bentuk *library python* yang bisa diunduh melalui pip atau *Python Install Package*. Selenium tidak hanya bisa mengotomasi sebuah browser, tetapi juga bisa membantu pengembang untuk melakukan *web-scraping* yang akan menjadi salah satu tahap yang penting dilakukan untuk pengembangan aplikasi AR ini.

2.6 Flask

Flask adalah sebuah kerangka kerja aplikasi web yang berseifat mikro yang ditulis dalam bahasa pemrograman *Python* untuk membantu pengembang untuk membangun sebuah server menggunakan bahasa pemrograman *Python* (DQLab, 2021). *Flask* dalam konteks pengembangan aplikasi AR ini adalah sebuah *library Python* yang memungkinkan pengembang aplikasi untuk membangun sebuah server *API REST* yang nantinya akan membantu menerima request dan mengeluarkan respon sesuai dengan *endpoint* yang diinginkan.

2.7 Regex

Regex adalah singkatan dari *Regular Expression*, yang berupa rangkaian karakter yang menyusun sebuah pola pencarian yang bisa membantu pengembang aplikasi untuk mencari, mengubah, memaipulasi, serta memvalidasikan teks untuk membuat aplikasi menjadi lebih kuat (Cmlabs, 2024). *Regex* pertama kali muncul pada tahun 1951, ketika seorang ilmuwan bernama Stephen Cole Kleene memformulasikan pola bahasa formal. *Regex* digunakan pada aplikasi komputasi di tahun 1960 dengan diterapkannya aplikasi *editor* teks pada *UNIX* yang bernama *QED*. Lalu, penggunaan *regex* meluas ke penggunaan alat-alat dan pemrograman pada tahun 1970-an. Ini bisa dibuktikan dengan adanya *grep* (*Global Regular Expression Print*) yang diintegrasikan ke sistem *UNIX*. Sejak saat itu, penggunaan *regex* meluas ke berbagai bahasa pemrograman mulai dari *Perl* hingga *Python* serta sistem-sistem basis data.

2.8 Penelitian Terdahulu

Tabel 2.1 Penelitian Terdahulu

No .	Penulis	Judul	Tahun	Perbedaan
1.	Joshi, S., Walavalkar, P., Shetkar, P., Joshi, S., Mahale, S., Bhogan, S., & Sawant, K.	<i>Product Visualization Using Augmented Reality</i>	2020	Penelitian yang dilakukan Joshi memiliki database yang dibuat berdasarkan atribut atribut yang dimiliki produk sedangkan penelitian penulis akan memanfaatkan deskripsi produk yang terdapat di halaman produk untuk mendapatkan ukuran dari produk.
2.	Carlos Alves dan José Luís Reis	<i>The Intention to Use E- Commerce Using Augmented Reality - The Case of IKEA Place</i>	2020	Penelitian Carlos Alves adalah tentang penelitian niat konsumen untuk menggunakan sebuah produk AR untuk meningkatkan pengalaman berbelanja sedangkan penelitian penulis berusaha untuk mewujudkan aplikasi AR untuk digunakan konsumen.
3.	Mohamad Safrodin, Saiful Ghozi, dan Moh. Zikky	<i>The 3D virtual drawing mobile application based on augmented reality using AR- Framework</i>	2020	Penelitian Mohammad Safrodin adalah aplikasi AR untuk menggambar secara 3D sedangkan penelitian penulis adalah aplikasi AR untuk menampilkan produk belanja online.

4.	Sriram V.P, Arun Ambikapati, Sikandar M.A, Kamal Gulati	<i>The rise of 3D E-Commerce the online shopping gets real with virtual reality and augmented reality during COVID-19</i>	2021	Penelitian yang dilakukan Sriram V.P adalah penelitian tentang implementasi VR dan AR dalam E-commerce yang semakin marak sejak terjadinya pandemi Covid dan apa yang akan terjadi kedepannya sedangkan penelitian penulis adalah mengembangkan sebuah aplikasi AR untuk meningkatkan pengalaman belanja online yaitu menampilkan bagaimana sebuah produk terlihat tanpa harus datang langsung ke toko penjual.
5.	Mintu Baruah	<i>Deskkart- Furniture Mall with Augmented Reality</i>	2019	Penelitian oleh Mintu Baruah adalah aplikasi yang mengharuskan penjual untuk mengupload data model produknya sedangkan penulis mengusahakan untuk memberikan sebuah model model yang lebih universal dan lebih merujuk ke kenyamanan konsumen tanpa menyusahkan penjual untuk menyiapkan model 3D untuk diupload.

Secara umum, penelitian penulis lebih berfokus kepada pembuatan aplikasi *AR* yang mengutamakan pengguna dalam berbelanja *online* daripada pengimplementasian teknologi *AR* ke bidang *e-commerce*. Fokus ini berbeda dengan penelitian-penelitian sebelumnya yang melakukan survey dan pembuatan platform belanja secara keseluruhan. Penelitian ini berusaha untuk memberikan insentif kepada pengguna atau konsumen untuk menggunakan teknologi *AR* agar penggunaannya bisa dilakukan tanpa menunggu penjual untuk menyediakan model 3D *AR* untuk ditampilkan.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Data yang Digunakan

Data yang digunakan dalam penelitian ini mencakup deskripsi-deskripsi yang ada di produk-produk Tokopedia yang mengandung data ukuran produk. Data inilah yang akan digunakan untuk membuat sebuah rangkaian *regex* yang nantinya digunakan untuk membantu server untuk mengenali *string* ukuran dari deskripsi produk. Berikut ini ukuran dan rangkaian *regex* yang dikenali oleh server *API* :

Tabel 3.1 Data String dan Regex dari Tokopedia

No.	Contoh String	Regex
1.	P: 120 x L: 60 x T: 95 cm	<code>r'P:\s*(\d*\.\?d+)\s*x\s*L:\s*(\d*\.\?d+)\s*x\s*T:\s*(\d*\.\?d+)'</code>
2.	150x50x85	<code>r'(\d*\.\?d+)\s*x\s*(\d*\.\?d+)\s*x\s*(\d*\.\?d+)'</code>
3.	200X80X45	<code>r'(\d*\.\?d+)\s*X\s*(\d*\.\?d+)\s*X\s*(\d*\.\?d+)'</code>
4.	29.6 cm x 10.6 cm x 3.4 cm	<code>r'(\d*\.\?d+)\s*cm\s*x\s*(\d*\.\?d+)\s*cm\s*x\s*(\d*\.\?d+)\s*cm'</code>
5.	11.42 x 3.94 x 1.57inches	<code>r'(\d*\.\?d+)\s*x\s*(\d*\.\?d+)\s*x\s*(\d*\.\?d+)\s*inches?'</code>
6	310 x 100 x 44mm	<code>r'(\d*\.\?d+)\s*\s*(\d*\.\?d+)\s*\s*(\d*\.\?d+)\s*mm'</code>
7	Panjang: 200cm X Lebar:45cm X Tinggi: 70cm.	<code>r'Panjang:\s*(\d+)\s*cm\s*X\s*Lebar:\s*(\d+)\s*cm\s*X\s*Tinggi:\s*(\d+)\s*cm'</code>

Regex pada tabel diatas memiliki 2 persamaan yaitu sebagai berikut:

- Rangkaian *regex* “`(\d*\.\?d+)`”
`“/d*”` : Berguna untuk mencocokkan angka 0 hingga 9 termasuk jika tidak ada digit.
`“\.”` : Berguna untuk mencocokkan titik desimal.

“/d+” : Berguna untuk mencocokkan angka 0 hingga 9 dengan minimal 1 digit. Jika digabungkan, *regex* tersebut akan mencocokkan semua angka dengan opsi memiliki digit desimal atau tidak. Tanda kurung yang mengurungi *regex* tersebut berguna untuk menandakan bahwa bagian ini adalah bagian yang akan diambil nilainya.

- Rangkaian *regex* “\s*x\s*”

“\s*” : Berguna untuk mencocokkan karakter kosong atau spasi baik ada ataupun tidak.

“\s*x\s*” : Mencocokkan *string* yang memiliki karakter "x", baik memiliki spasi di awal dan/atau di akhir ataupun tidak.

Selain data yang didapat dari Tokopedia, penulis menambahkan data-data *regex* yang memungkinkan namun tidak ditemukan selama pencarian data di Tokopedia. Berikut data *string* dan *regex* yang ditambahkan oleh penulis :

Tabel 3.2 Data String dan Regex yang Ditambahkan Oleh Penulis

No.	Contoh <i>String</i>	<i>Regex</i>
1	29,7 x 10,7 x 27,2	<code>r'(\d+(?:,\d+)?)\s*x\s*(\d+(?:,\d+)?)\s*x\s*(\d+(?:,\d+)?)'</code>
2	29.7 x 10.7 x 27.2	<code>r'(\d+(?:\.\d+)?)\s*x\s*(\d+(?:\.\d+)?)\s*x\s*(\d+(?:\.\d+)?)'</code>
3	29,7 X 10,7 X 27,2	<code>r'(\d+(?:,\d+)?)\s*X\s*(\d+(?:,\d+)?)\s*X\s*(\d+(?:,\d+)?)'</code>
4	29.7 X 10.7 X 27.2	<code>r'(\d+(?:\.\d+)?)\s*X\s*(\d+(?:\.\d+)?)\s*X\s*(\d+(?:\.\d+)?)'</code>
5	29,7 * 10,7 * 27,2	<code>r'(\d+(?:,\d+)?)\s*\s*(\d+(?:,\d+)?)\s*\s*(\d+(?:,\d+)?)'</code>
6	29.7 * 10.7 * 27.2	<code>r'(\d+(?:\.\d+)?)\s*\s*(\d+(?:\.\d+)?)\s*\s*(\d+(?:\.\d+)?)'</code>
7	10 * 12 * 32	<code>r'(\d+)\s*\s*(\d+)\s*\s*(\d+)'</code>

Jenis-jenis kemungkinan *string* yang dikumpulkan cukup banyak untuk dites satu persatu untuk hanya satu deskripsi produk. Oleh karena itu, dibutuhkan sebuah rangkaian *regex* yang bisa mengambil angka-angka dimensi produk yang memang relevan untuk dijadikan sebuah nilai yang dikirimkan kembali ke aplikasi. Data metrik yang digunakan untuk deskripsi produk bisa diabaikan dan dijadikan sebuah opsi ketika dimensi produk telah disampaikan ke aplikasi AR. *Regex* yang bisa dijadikan sebuah patokan untuk segala kemungkinan produk tersebut ialah sebagai berikut

Tabel 3.3 Regex Pencari Dimensi Produk yang Universal

Regex Dimensi Produk Universal
<code>r'(\d+(?:\.\d+)?)\s*(?:cm m inches in mm)?\s*[xX*]\s*(\d+(?:\.\d+)?)\s*(?:cm m inches in mm)?\s*[xX*]\s*(\d+(?:\.\d+)?)\s*(?:cm m inches in mm)?'</code>

Regex universal diatas dapat dibagi menjadi beberapa segmen berikut

1. Segmen penangkap angka

Segmen ini adalah rangkaian *regex* “(\d+(?:\.\d+)?)”.

2. Segmen satuan ukuran

Segmen ini adalah rangkaian *regex* “(?:cm|m|inches|in|mm)?” yang merupakan pola opsional untuk mencocokkan salah satu dari satuan antara lain yaitu sentimeter, meter, dan inci. Penggunaan *regex* “(?:)” adalah untuk tidak menyimpannya kedalam grup hasil *filtering*.

3. Segmen pemisah dimensi

Pemisah dimensi adalah rangkaian *regex* “\s*[xX*]\s*” yang mencocokkan pemisah berupa “x”, “X” atau “*” yang diapit oleh spasi kosong ataupun tidak.

Selain *regex*, data lain yang digunakan untuk sistem aplikasi AR ini adalah beberapa model 3D umum produk untuk divisualisasikan pada aplikasi. Model 3D ini diambil dari *Unity Asset* yang menyediakan berbagai aset aset yang bisa digunakan untuk membangun aplikasi atau permainan di *Unity*. Model-model yang diambil dari *Unity Asset Store* tersebut adalah beberapa furniture seperti meja, kursi, dan rak. Tentunya, aplikasi juga diharuskan untuk bisa mengalami perkembangan jika nantinya ada model 3D baru yang ingin dimasukkan.

3.2 Analisis Sistem

Penelitian ini bertujuan untuk membangun sebuah system dimana pengguna bisa melihat bagaimana ilustrasi sebuah produk terlihat didunia nyata menggunakan *smartphone android* masing-masing. Sistem ini bertujuan untuk memberikan sebuah solusi independen yang tidak bergantung kepada penjual untuk menyediakan model-model 3D yang digunakan untuk menampilkan produk di *smartphone* pembeli.

Sistem ini menggabungkan dua platform yang dihubungkan menggunakan *API*. Di sisi server, framework yang digunakan adalah *python flask*, sedangkan di sisi pengguna, aplikasi dibangun menggunakan *Unity* atau Bahasa pemrograman *C#*.

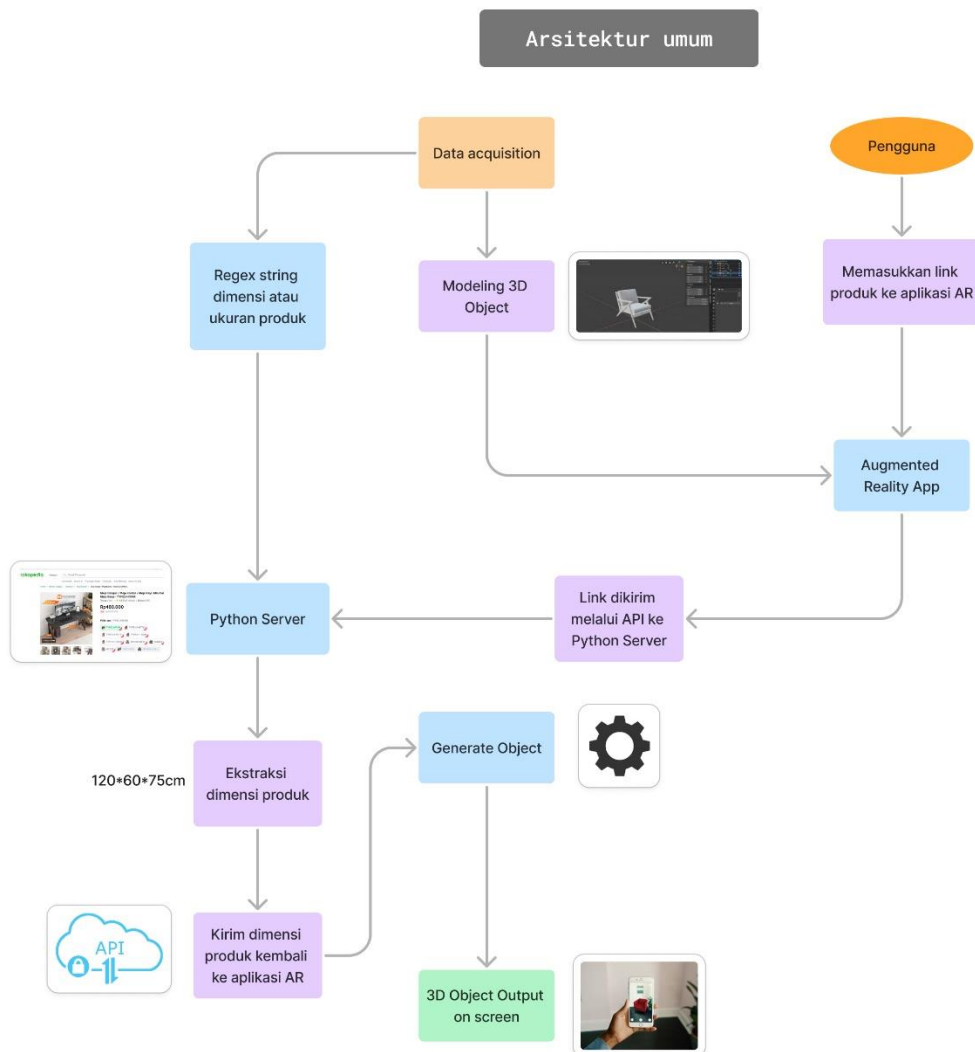
3.3 Perancangan Sistem

Sub bab ini akan menjelaskan tentang rancangan sistem aplikasi *AR* dan server serta *API* yang akan dibangun secara rinci. Aplikasi ini dirancang untuk perangkat *mobile* dengan fitur *augmented reality*. Sistem ini mengharuskan aplikasi untuk memiliki izin pengguna untuk menggunakan kamera *smartphone*. Pengguna lalu diberikan pilihan untuk memasukkan link produk Tokopedia atau langsung memasuki simulasi *AR*. Setelah memasuki simulasi *AR* pengguna harus mencari *marker AR* hingga memasuki cakupan kamera. Setelah *Marker AR* ditemukan oleh kamera pengguna, jika pengguna telah memasuki link di menu awal aplikasi, aplikasi akan menampilkan prooduk tersebut sesuai dengan ukuran yang didapat dari laman produk *Tokopedia*. Jika tidak, pengguna diberikan tombol untuk memasukkan objek secara manual.

3.3.1 Arsitektur Umum

Arsitektur umum dari sistem aplikasi yang dibangun terdiri dari beberapa langkah seperti yang bisa dilihat di Gambar 3.1 sebagai berikut :





Gambar 3.1 Arsitektur Umum

Pertama-tama penulis akan melakukan data *acquisition* untuk mengumpulkan *string-string regex* yang bisa menjadi kemungkinan *string* yang akan ada di deksripsi produk pada website belanja *online*. Setelah itu penulis akan membangun sebuah server dimana server tersebut akan menerima request dari aplikasi *AR Unity* yang nantinya akan memberikan respons berupa dimensi produk yang telah diekstrak. Pengguna aplikasi akan memasukkan link produk kedalam aplikasi *AR* yang nantinya berinteraksi dengan server *API Python* yang telah dibuat. Setelah itu, Aplikasi *AR Unity* akan

mengatur bagaimana seberapa besar produk terlihat lalu menampakkannya di layar *smartphone*.

Regex yang akan digunakan ditentukan berdasarkan analisis *string* yang telah dilakukan pada deskripsi-deskripsi produk *Tokopedia* yang memiliki *string* atau data dimensi produk. *String* dimensi produk ini akan dianalisis untuk menghasilkan *regex* yang universal agar bisa aplikasi bisa memindai teks dalam satu langkah untuk mendapatkan semua kemungkinan *string* yang mengandung dimensi produk.

a. *Input*

Tahap *Input* pada aplikasi ini adalah tahap dimana pengguna memasukkan link produk *Tokopedia* kedalam formulir yang terdapat di *main menu* aplikasi. Sebelum ini pengguna diekspektasikan sudah memiliki link produk disimpan didalam *clipboard* untuk di *paste*. Jika tidak, masih ada kesempatan untuk pengguna untuk memasukkan link walaupun *AR* sudah berjalan (tidak lagi di *main menu*). Rincian dari proses *input* pengguna bisa diuraikan sebagai berikut :

- Proses *input link* oleh pengguna kedalam formulir yang disediakan di *main menu* (Contoh *link* : <https://www.tokopedia.com/ocadoofficial/kursi-lipat-perengkapan-salon-foldable-kursi-ergonomic-serbaguna-t1-hitam-e9aac?ext>).



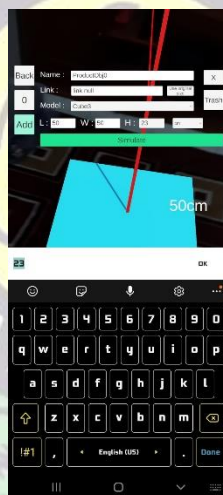
Gambar 3.2 Input Link pada Menu Utama

- Proses *input marker* kedalam cakupan kamera menggunakan *smartphone* pengguna.



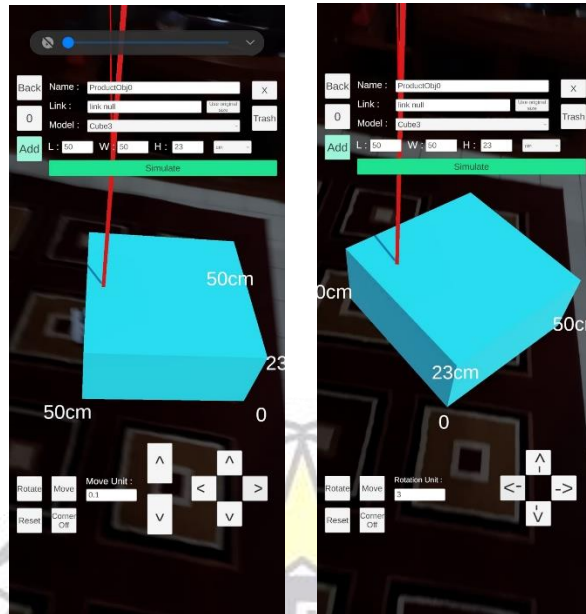
Gambar 3.3 Input Marker Menggunakan Kamera Pengguna

- Proses input data-data rincian objek yang disimulasikan di dalam seksi *AR* (panjang, lebar, dan tinggi objek, nama objek, model objek, dan link objek).



Gambar 3.4 Input Dimensi Objek 3D Produk

- Proses input posisi dan rotasi objek *AR* sesuai dengan keinginan pengguna.



(a)

(b)

Gambar 3.5 Input Posisi dan Rotasi Objek

b. Process

Pada tahap ini terdapat beberapa proses yang dilakukan oleh sistem antara lain : *Marker Tracking*, *API Request* dan *Scraping*, *Spawning* dan *Positioning*, *Size Measuring*, dan *UI Control*.

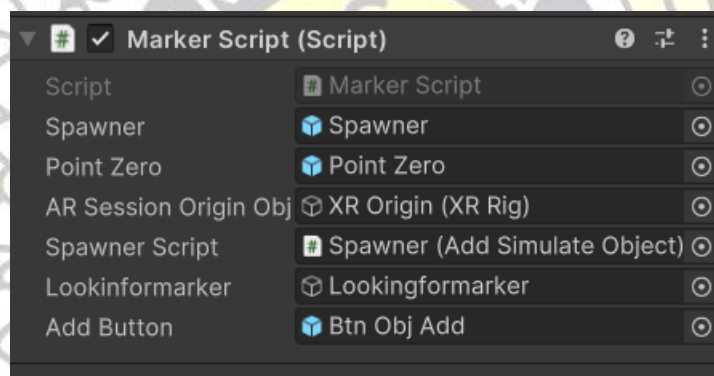
a) *Marker Tracking*

Proses ini dilakukan dengan cara memposisikan kamera android hingga *marker* aplikasi terlihat di kamera. Di proses ini, pengguna harus menggunakan dataran yang datar secara horizontal sebagai letak *marker*. Aplikasi *AR* secara otomatis akan menaruh sebuah data posisi yang nantinya akan menjadikan *marker* tersebut menjadi sebuah titik nol di sumbu X, Y dan Z. Setelah *marker* terdeteksi, tombol *Add* akan muncul dan tulisan “*Looking for marker*” akan hilang. Akan terdapat sebuah tiang vertical yang bisa menjadi sebuah titik patokan untuk menandakan bahwa posisi tersebut adalah titik nol. Proses tersebut akan terlihat seperti berikut



Gambar 3.6 Tampilan Proses Mencari Marker

Berikut adalah skrip *Unity* yang menangani proses tersebut



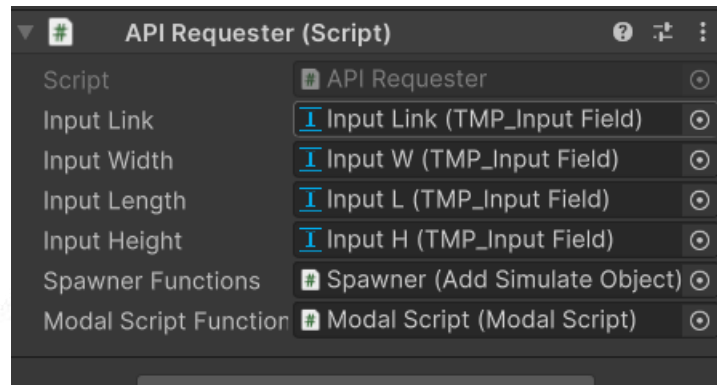
Gambar 3.7 Skrip Marker

b) *API Request* dan *Scraping*

Proses *API Request* adalah proses dimana aplikasi *AR* akan memngirim sebuah request kepada server *python* menggunakan *API* yang dibuat di server *python* tersebut. Server nantinya akan menerima request tersebut lalu akan mencari data ukuran produk dari link *Tokopedia* yang diberikan. Ketika proses ini berlangsung, aplikasi *AR* akan memberikan pemberitahuan bahwa aplikasi sedang mengambil data produk dan meminta pengguna untuk menunggu sebentar. Server akan berusaha untuk melakuka *scrapping* pada halaman tautan produk yang telah di-*input* untuk mencari *tag-tag* khusus yang menyimpan deskripsi produk pada halamannya yang dalam hal ini adalah

tag CSS yaitu “css-16inwn4”. Server lalu akan melakukan operasi tertentu kepada teks yang didapat untuk mendapatkan ukuran produk.

Ketika server selesai mendapatkan ukuran produk, ukuran produk akan dikirim sebagai *response* ke aplikasi AR. Ketika aplikasi AR telah mendapat *response* dengan data yang baik, aplikasi AR akan melanjutkan proses lebih lanjut. Berikut skrip yang digunakan untuk menangani permintaan API di aplikasi AR ini



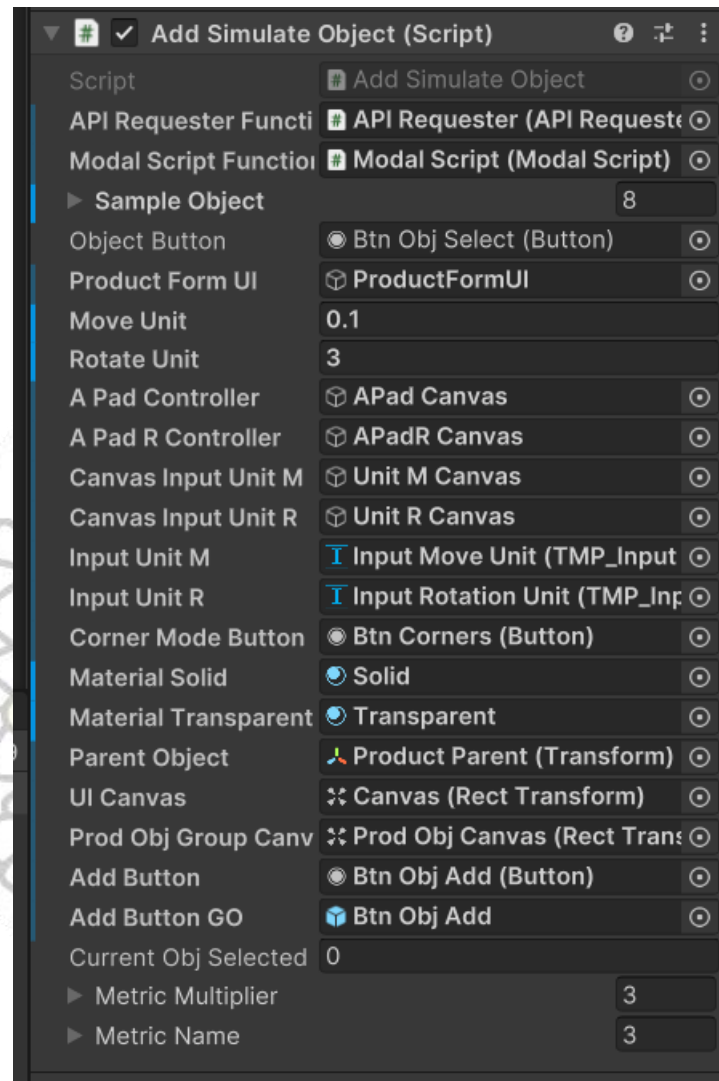
Gambar 3.8 Skrip API Requester

Input Link di skrip ini mengambil data input yang diberikan dari menu utama. *Input Width*, *Input Length*, dan *Input Height* adalah inputan yang terdapat di mode AR untuk menampilkan data panjang, lebar, dan tinggi produk kepada pengguna sekaligus mengambil masukan dari pengguna. *Spawner Functions* adalah parameter yang berguna untuk mengambil data-data terkait yang ada di fungsi *Spawner*. Ini harus dilakukan agar aplikasi tidak melakukan duplikasi kode dan pemborosan ruang penyimpanan memori. Hal yang sama juga terjadi di *Modal Script Function* yang berguna untuk melakukan operasi-operasi tertentu untuk menggunakan modal.

c) Spawning, Positioning, dan Size Measuring

Pada proses ini, aplikasi AR telah mendapatkan ukuran produk *Tokopedia* yang telah dimasukkan linknya di tahap *input*. Proses Spawning adalah penempatan objek 3D yang akan dimasukkan ke dalam lingkungan AR. Model yang akan digunakan secara *default* akan berupa model kubus. Model 3D ini akan ditempatkan di titik *zero point* yang merupakan letak *marker* yang sudah dipetakan ke ruang lingkup AR aplikasi. Letak *zero point* ini akan ditandai dengan sebuah objek balok yang memanjang keatas dan merupakan titik nol di setiap sumbu (X,Y, dan Z). Selanjutnya objek 3D ini akan

diatur ukurannya sesuai dengan ukuran yang didapat dari *API* di tahap sebelumnya. Jika data ukuran terjadi error, atau data tidak bisa didapatkan, maka ukuran akan diatur menjadi 1 x 1 x 1 unit unity (meter) secara *default*. Berikut skrip yang terkait dengan proses ini :



Gambar 3.9 Skrip *Spawner*

Skrip *Spawner* ini adalah skrip yang menangani segala hal yang berkaitan dengan penempatan, pengukuran dan pemunculan objek serta tombol-tombol UI tertentu yang terkait dengan hal tersebut. *Sample Object* adalah sebuah array yang berisi model model yang bisa digunakan untuk menampilkan produk *AR*. *Object Button* adalah *prefab* yang digunakan untuk menambahkan tombol untuk setiap objek yang ditambahkan agar objek tersebut bisa dipilih. *Product Form UI* adalah *game object* induk dari form antarmuka pengguna yang berfungsi untuk mengaktifkan atau

menonaktifkan form secara keseluruhannya sekaligus. *Move Unit* dan *Rotate Unit* adalah data bertipe *float* yang berguna untuk memberikan nilai cakupan pergerakan dan rotasi setiap kali tombol gerak atau rotasi ditekan. *A Pad Controller* dan *A Pad R Controller* adalah *game object* induk untuk mengaktifkan atau menonaktifkan antarmuka dari tombol-tombol gerak dan rotasi. *Canvas Input Unit M* dan *R* adalah sebuah *game object* induk yang mengandung *input*-an nilai untuk dijadikan ukuran pergerakan dan rotasi objek produk sedangkan *Input Unit M* dan *R* adalah *game object* itu sendiri.

Tombol *Corner Mode Button* adalah tombol yang berguna untuk menghidupkan atau mematikan tampilan sudut-sudut objek produk. *Material Solid* dan *Transparent* adalah material-material yang digunakan untuk memberikan efek tembus pandang dan efek padat untuk objek yang sedang dipilah dan yang tidak. *Product Parent* adalah sebuah objek bertipe data *Transform* yang mana berguna untuk menspesifikasikan dimana letak objek produk yang ditampilkan berada di daftar objek. Objek inilah yang akan menjadi induk dari objek-objek produk yang akan ditampilkan.

UI Canvas berguna untuk menspesifikasikan objek induk yang menjadi sebuah wadah untuk beberapa komponen *UI*. *Add button* adalah sebuah objek yang berfungsi mengatur tata letak antarmuka tombol tambah objek. Ini berguna Ketika pengguna ingin menambah objek baru kedalam *scene* aplikasi *AR*. Lalu dibawahnya, terdapat *Add Button GO* yang berguna untuk memasukkan atau memunculkan tombol *Add Button* tersebut kedalam daftar objek ketika permintaan *API* telah dipenuhi. Perbedaan *Add Button* dan *Add Button GO* ini terletak di objek yang direferensikan. *Add Button* mereferensikan tombol sedangkan *Add Button GO* mereferensikan *game object* secara keseluruhan objek tombol tersebut.

Metric Multiplier dan *Metric Namei* adalah variabel yang mengandung list nilai yang akan dikalikan dengan ukuran yang diberikan di form ukuran objek produk. Ini berguna untuk mengatur ukuran produk sesuai dengan metriknya yaitu meter, sentimeter, dan inci.

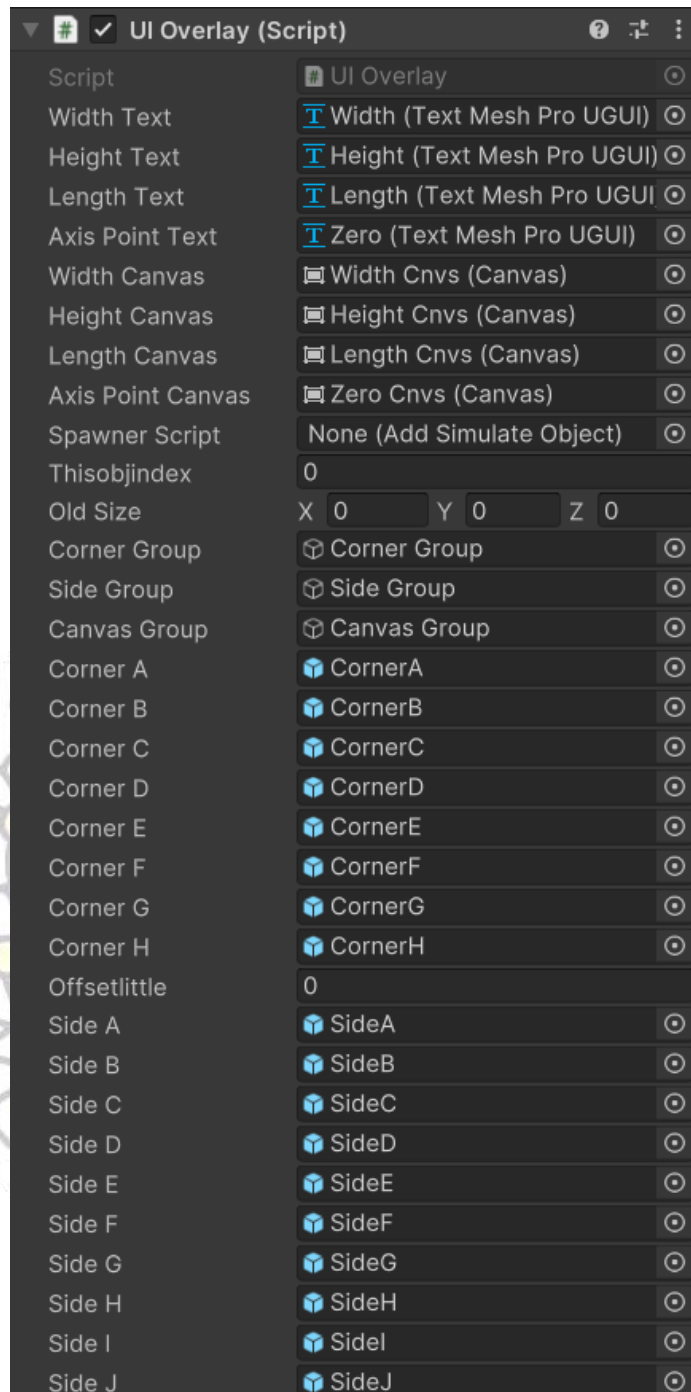
d) *UI Control*

Setelah model *3D* dimuat di lingkungan *AR*, sistem aplikasi akan memberikan sebuah interface untuk memperkaya informasi dari objek terserbut. *Interface* tambahan

ini berupa angka yang menunjukkan ukuran objek sesuai dengan sumbunya. Misalnya : jika model objek memiliki ukuran 20,30,40 pada sumbu X,Y dan Z secara berurutan, angka-angka tersebut akan menjadi indikasi pada objek dari sudut ke sudut untuk menandakan sisi manakah yang memiliki ukuran X,Y atau Z tersebut. Salah satu sudut objek akan diberikan angka '0' untuk menandakan bahwa dari sudut tersebutlah ukuran-ukuran setiap sumbu diukur.

Selain angka ukuran, aplikasi akan memberikan sebuah tanda atau indikasi sudut dan sisi untuk model 3D. Indikasi ini berguna untuk model model 3D lainnya yang berbentuk bukan kubus, seperti bola, kursi, sofa dan banyak bentuk lainnya. Indikasi sudut dan sisi ini bisa dimatikan atau dihidupkan untuk menghilangkan *clutter* jika objek terasa 'ramai' atau tidak menentu. Pengguna juga bisa mengganti model setelah ditampilkan selagi dalam mode *AR* menggunakan *UI* yang tampil di layar *smartphone* pengguna. Berikut adalah skrip *Unity* yang menangani antarmuka tersebut :



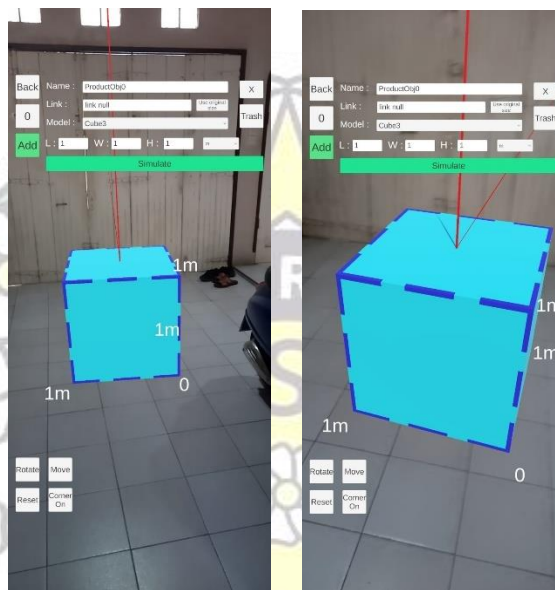


Gambar 3.10 Skrip *UI Control*

Dapat dilihat dari skrip diatas ada beberapa varibel yang mencolook yaitu *Corner* dan *Side*. *Corner* mulai dari A hingga H, adalah sudut sudut yang ada di objek produk. *Side* adalah sisi-sisi yang ada di sebuah produk. Sudut dan sisi ini memberikan sebuah bentuk kubus atau balok yang menjadi sebuah tanda dan petunjuk yang bertujuan untuk memperjelas seberapa besar sebuah produk terlihat secara garis besar. Produk mungkin tidak akan terlalu terlihat seperti apa bentuknya dari sudut ke sudut,

jadi penulis memberikan pilihan kepada pengguna untuk melihat seberapa banyak ruangan yang dipakai oleh produk tersebut.

Weight, *Height* dan *Length Text* dan *Canvas* adalah salah satu penjelas lainnya yang memberikan angka yang menunjukkan seberapa produk tersebut berukuran di dunia nyata. *Weight*, *Height*, dan *Length Canvas* digunakan untuk memberikan sebuah efek ukuran huruf yang digunakan agar tidak berubah ukuran jika objek produk mendekati kamera atau menjauhi. Berikut contoh cara kerja ukuran huruf tersebut :



(a) (b)

Gambar 3.11 Ukuran *font* pada saat objek menjauhi kamera(a) dan saat objek mendekati kamera(b) sama

Oldwhl adalah variable vektor yang berguna untuk membandingkan ukuran sebelumnya dengan yang telah diubah. Ini berguna untuk mengecek apakah ukuran berubah atau tidak ketika melakukan perubahan ukuran objek. Tujuan utama variable ini adalah untuk mengurangi tenaga proses *smartphone* pengguna agar tidak menjalani proses yang tidak sebenarnya dibutuhkan. *Thisobjectindex* adalah variabel yang menyimpan nomor indeks seberapa objek yang telah dipanggil tersebut. Ini berguna untuk mengambil informasi objek secara cepat tanpa harus menghitung kembali list yang ada di skrip *Spawner*.

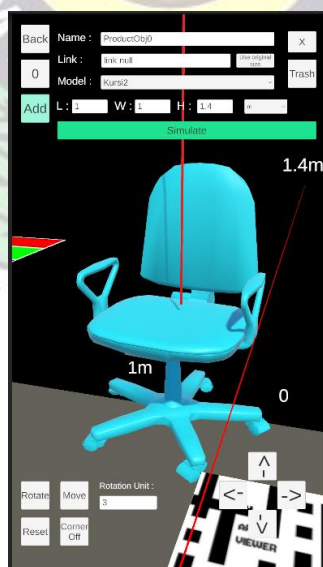
Canvas, *Side*, dan *Corner Group* adalah induk dari objek sudut dan sisi dari sebuah objek produk. Ini dibuat agar semua sisi dan sudut bisa lebih mudah

dinonaktifkan dan diaktifkan tanpa harus mengatur setiap objek sudut dan sisi satu-persatu. *Offsetlittle* adalah nilai *float* yang berguna untuk memberikan sedikit jarak antara objek produk dan setiap sudut dan sisi agar sudut dan sisi tidak bertabrakan dengan jaringan *polygon* objek produk.

c. Output

Hasil dari sistem ini adalah sebuah aplikasi *AR* yang bisa menampilkan sebuah objek *3D* melalui layar *smartphone* pengguna dengan menggunakan hanya link produk *Tokopedia* saja. Pengguna bisa mengatur tata letak setiap objek produk yang ditampilkan menggunakan tombol-tombol yang telah disediakan. *UI* untuk tombol-tombol control ini bisa melakukan perubahan terhadap posisi dan rotasi dari objek produk tersebut.

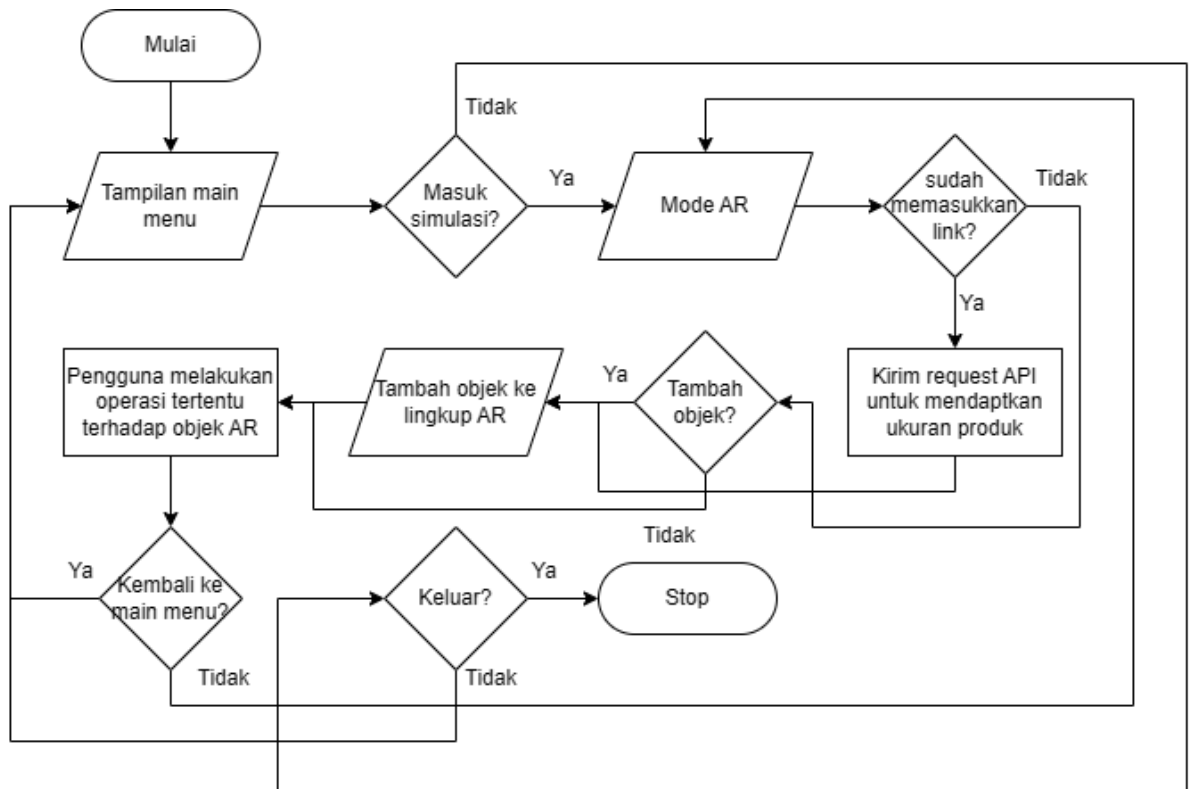
Produk-produk yang ditampilkan hanya menggunakan model-model patokan atau umum yang bisa merepresentasikan objek produk mendekati bentuk produk aslinya. Keuntungan memakai sistem ini adalah pengguna masih bisa melihat bentuk produk walaupun penjual tidak menyediakan model *3D* terlebih dahulu. Hal ini memberikan pengguna pilihan untuk menggunakan aplikasi dengan koneksi internet ataupun *offline*.



Gambar 3.12 Model produk berbentuk umum.

3.3.2 Flowchart Diagram

Flowchart diagram adalah sebuah Gambaran bagaimana sebuah proses atau aktivitas dilaksanakan mulai dari awal hingga akhir dari aktivitas tersebut. Sistem aplikasi yang dibuat oleh penulis memiliki beberapa tahapan dalam penggunaanya. Diagram ini dibuat untuk memberikan penjelasan bagaimana tahapan-tahapan tersebut bekerja.



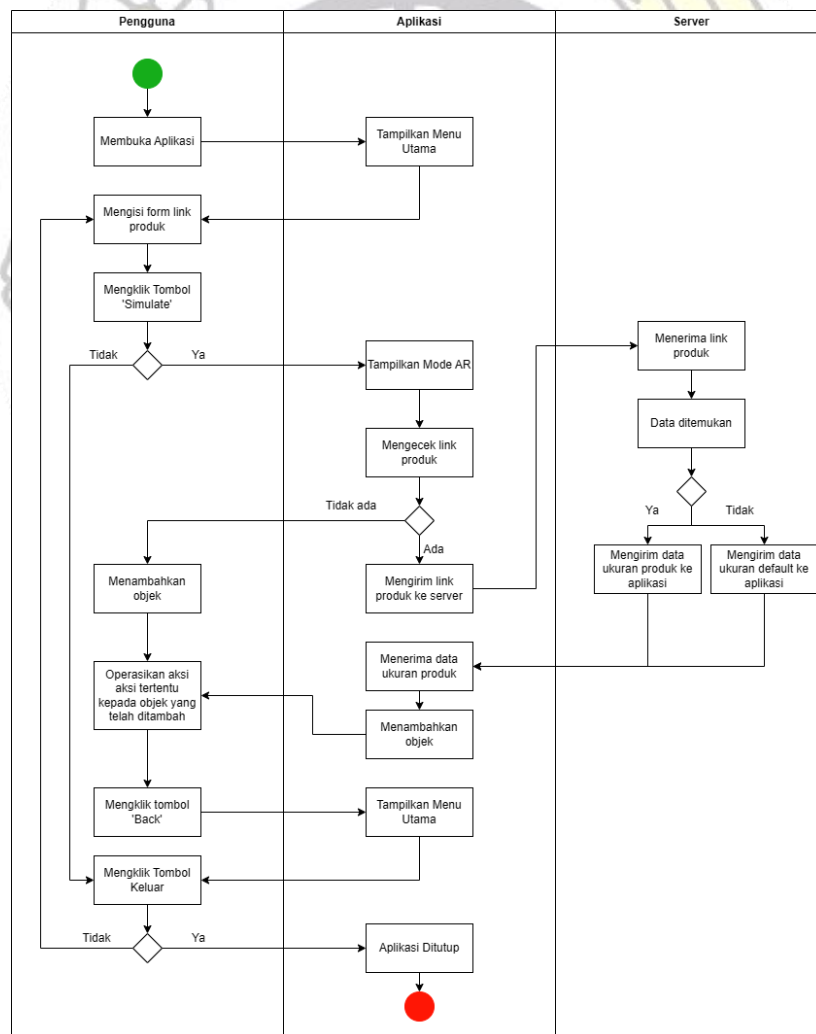
Gambar 3.13 Flowchart Sistem ARView

Ketika aplikasi pertama kali dijalankan, pengguna akan diberikan tampilan *main menu*. *Main menu* ini memiliki beberapa pilihan aksi seperti ‘mulai simulasi’ atau keluar dari aplikasi. Jika pengguna memilih untuk memulai simulasi, sistem aplikasi akan berpindah ke mode simulasi. Setelah berpindah mode, aplikasi akan mengecek apakah pengguna telah memasuki link produk atau tidak, jika ada maka akan diberikan sebuah objek *default* yang memiliki ukuran produk tersebut. Jika tidak, aplikasi akan melanjutkan proses ke aksi berikutnya yaitu memberikan pengguna sebuah opsi untuk menambah objek. Jika pengguna menekan tombol ‘*add object*’, sistem aplikasi akan menambahkan satu objek dengan model *default* berukuran 1 x 1 x 1 unit. Pengguna bisa melakukan operasi operasi yang tersedia untuk memodifikasi objek dan jumlah objek

yang ditampilkan di lingkungan *AR* aplikasi. Pengguna bisa kembali ke *main menu* jika menekan tombol 'Back'. Pilihan lain di *main menu* selain 'Simulate' adalah tombol 'Exit' yang berguna untuk keluar dari aplikasi.

3.3.3 Activity Diagram

Activity Diagram adalah diagram yang menggambarkan alur aktivitas dan Tindakan-tindakan yang diambil dalam sebuah sistem selama sistem tersebut berjalan. Diagram ini juga mencakup alur-alur pilihan yang mungkin terjadi berulang-ulang. Dengan adanya diagram ini, pengguna bisa memahami proses apa saja yang terjadi selama pengguna menggunakan aplikasi. Diagram berikut menjelaskan alur yang terjadi pada sistem aplikasi yang dibuat penulis.



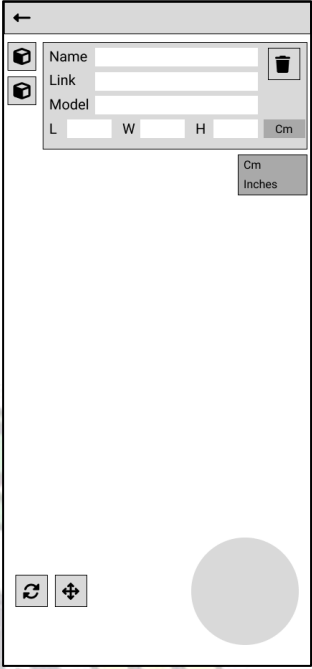
Gambar 3.14 Activity Diagram

3.4 Perancangan Antarmuka

Perancangan antarmuka adalah sebuah tahap dalam pembuatan aplikasi dimana pengembang menggambarkan kerangka antarmuka yang nantinya ditampilkan untuk pengguna aplikasi. Penggambaran antarmuka di tahap ini tidak menjadi hasil akhir namun menjadi dasar untuk membuat antarmuka sesungguhnya. Rancangan antarmuka yang dibuat untuk aplikasi *AR* yang dibuat penulis mencakup halaman menu utama dan mode *AR*.

Tabel 3.4 Desain Aplikasi

Halaman	Penjelasan	Ilustrasi
Menu Utama	Halaman menu utama memiliki sebuah <i>input</i> teks yang berguna untuk memasukkan tautan produk <i>Tokopedia</i> dan beberapa tombol yaitu <i>Simulate</i> , <i>About</i> , dan <i>Exit</i> .	

<p>Mode <i>AR/Simulator</i></p>	<p>Halaman mode <i>AR</i> adalah halaman antarmuka yang menampilkan visualisasi produk <i>Tokopedia</i>. Halaman ini memberikan berbagai opsi untuk melakukan operasi-operasi tertentu terhadap objek produk yang ditampilkan. Opsi-opsi tersebut antara lain adalah Tombol <i>Add</i>, Tombol <i>Delete</i>, <i>Input</i> Nama, <i>Input</i> Tautan Produk, <i>Input</i> Model Produk/Objek, <i>Input</i> Panjang,Lebar dan Tinggi Produk, Pilihan Unit <i>Unity</i>, Pilihan Mode Posisi, dan <i>Joystick</i> Kontrol Posisi Objek.</p>	
-------------------------------------	---	--

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap pertama adalah implementasi sistem dimana penulis akan melakukan pemasangan aplikasi *AR* ke dalam perangkat *mobile* untuk diuji.

4.1.1 Spesifikasi Perangkat Keras

Perangkat keras yang digunakan untuk menggunakan aplikasi ini adalah perangkat *mobile* seperti *smartphone* yang setidaknya memiliki kamera untuk mengambil *snapshot* lingkungan untuk dimasukkan kedalam aplikasi, sedangkan perangkat keras yang digunakan untuk mengembangkan aplikasi adalah sebuah perangkat PC(*laptop*), dan sebuah *smartphone* untuk mengetes *build* aplikasi. Adapun detail lebih lengkap untuk perangkat keras yang digunakan adalah sebagai berikut:

a. Laptop

- *Processor AMD Ryzen 7 4800H with Radeon Graphics 2.90 GHz*
- *16.0 GB Memory*
- *512GB Storage*
- *Nvidia GeForce GTX 1650 Ti Graphics Card*

b. Smartphone

- *128GB Storage*
- *6GB Memory*
- *Super AMOLED FHD Graphics*
- *Camera 48.0 MP, F2.0*
- *Accelerometer, Fingerprint Sensor, Gyro Sensor, Geomagnetic Sensor, Light Sensor, Virtual Proximity Sensing*

4.1.2 Spesifikasi Perangkat Lunak

Perangkat lunak yang digunakan untuk mengembangkan aplikasi ini adalah sebagai berikut :

a. Perangkat Pemrograman

- *Windows 10 Home*
- *Unity Game Engine 2022.3.33f.1*
- *Python Server*
- *AR Foundation 5 for Unity*
- *ARCore*
- *Newtonsoft JSON for Unity*
- *Selenium*
- *Firefox/Chrome Web Browser Driver*

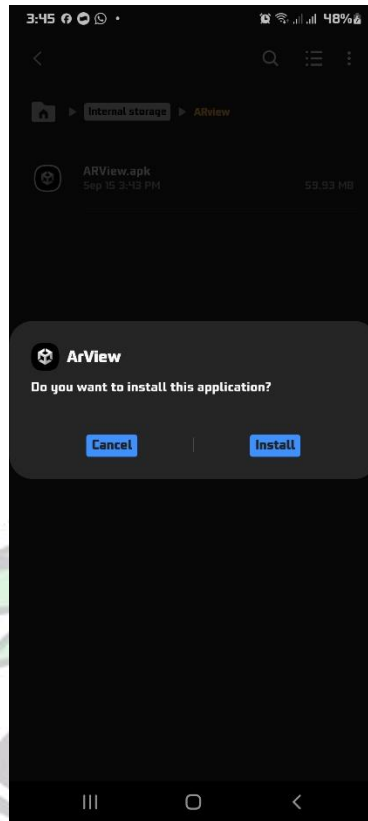
b. Perangkat Uji Program

- Sistem Operasi Android Versi 10
- Layanan Google Play *AR* untuk Android
- Perangkat *mobile* dengan layar berukuran 1080 x 2400 *pixel* dan 720 x 1480 *pixel*

4.1.3 Instalasi Aplikasi pada Perangkat *Mobile*

Sebelum memakai aplikasi, aplikasi harus terlebih dahulu dipasang ke perangkat *mobile* agar bisa dipakai. Langkah-langkah agar aplikasi bisa diinstal dengan benar adalah sebagai berikut :

1. Perangkat *mobile* pengguna harus memiliki versi *Android* 7.0 dan memiliki dukungan *ARCore*. Kesanggupan perangkat *mobile* untuk memakai *ARCore* bisa dilihat di halaman resmi *ARCore*.
2. Kondisi sensor *Accelerometer* dan *gyro* harus berfungsi dengan baik untuk memastikan aplikasi bisa melakukan visualisasi *AR* dengan baik.
3. Perangkat memiliki koneksi internet untuk memakai tautan produk *Tokopedia* (opsional). Ini berguna jika pengguna mau menggunakan aplikasi tanpa harus mengingat berapa ukuran produk *Tokopedia* tersebut.
4. Instal aplikasi pada perangkat *mobile* menggunakan instalasi *apk*. Contoh langkah ini dapat dilihat seperti gambar berikut.



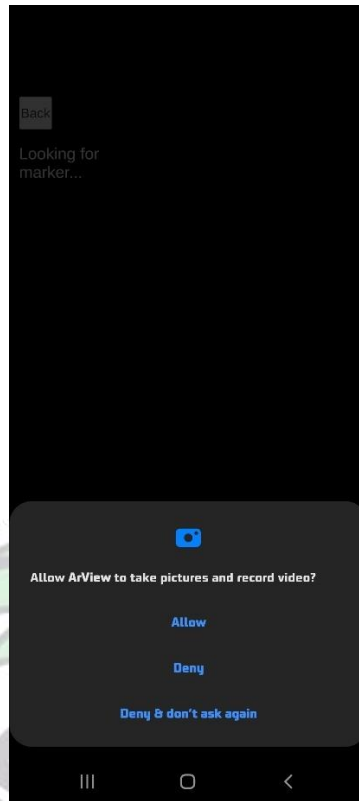
Gambar 4.1 Instalasi Aplikasi ARView

4.2 Tampilan Aplikasi

Tampilan aplikasi *AR* ini dirancang berdasarkan desain *wireframe* yang ada di bab 3 sebelumnya yaitu perancangan antarmuka. Bagian ini juga akan menjelaskan bagaimana terapan desain yang telah dibuat untuk dijadikan antarmuka di aplikasi sebagai hasil.

4.2.1 Tampilan Aktivasi akses Kamera

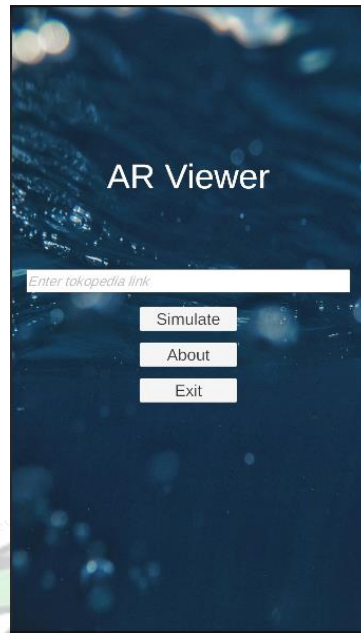
Ketika aplikasi sedang dibuka, aplikasi akan meminta izin kepada pengguna untuk menggunakan kamera *smartphone* yang ada di ponsel pengguna. Izin ini memberikan akses kepada aplikasi untuk menggunakan kamera untuk digunakan didalam aplikasi *AR*.



Gambar 4.2 Aplikasi meminta akses kamera

4.2.2 Tampilan Awal Aplikasi

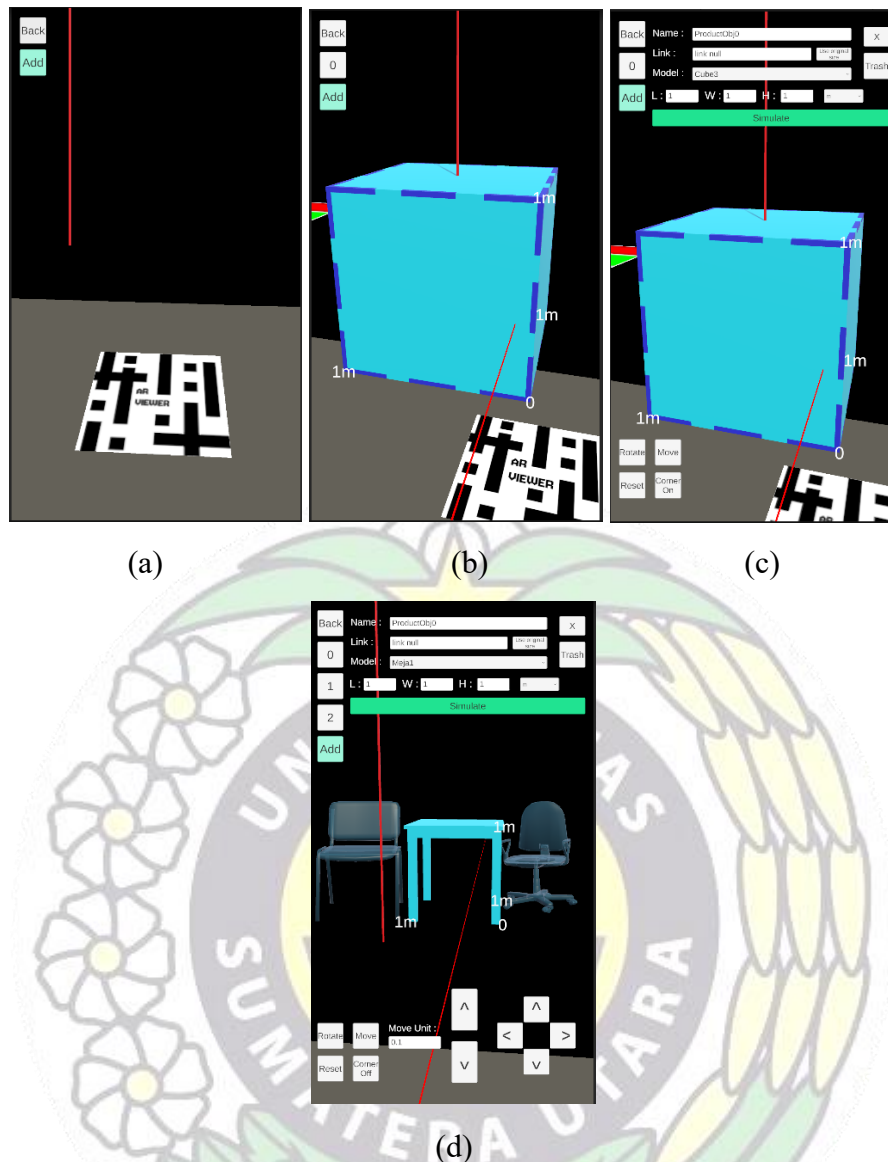
Tampilan awal aplikasi, sesuai dengan namanya, ditampilkan ketika aplikasi baru saja dibuka. Pengguna ditampilkan opsi untuk masuk ke mode simulasi, melihat halaman *About*, atau keluar dari aplikasi.



Gambar 4.3 Menu Aplikasi

4.2.3 Tampilan Halaman *AR*

Tampilan aplikasi didalam halaman *AR* memiliki beberapa pilihan. Pada gambar 4. (a), tampilan halaman *AR* terlihat kosong karena belum ada objek yang ditambahkan. Ketika objek ditambahkan, tampilan akan berubah menjadi seperti yang ada di gambar 4. (b). Pada mode ini, pengguna ditampilkan beberapa opsi operasi seperti, mengganti nama produk, mengganti link produk, menutup antarmuka opsi, mematikan dan menghidupkan sudut dan sisi, mengatur nilai rotasi dan pergerakan, dan tentunya mengganti ukuran produk. Jika pengguna menekan opsi *Corner* maka indikasi sudut dan sisi akan disembunyikan untuk memberikan gambaran lebih jelas tanpa indikasi sudut dan sisi menghalangi pengguna. Pengguna bisa juga menutup antarmuka menu opsi operasi yang ada seperti yang bisa dilihat di gambar 4. (c). Gambar 4. (d) adalah Gambaran visualisasi jika pengguna memiliki lebih dari satu objek ke dalam lingkungan *AR*.



Gambar 4.4 Berbagai mode di halaman AR

4.3 Pengujian Kinerja Sistem

Pengujian kinerja sistem adalah tahap yang dilakukan untuk memastikan komponen-komponen yang dibuat pada aplikasi dapat berfungsi dengan baik. Komponen pada sistem aplikasi yang telah melalui tahap pembuatan dan implementasi, selanjutnya komponen harus melalui tahapan pengujian *blackbox testing*. Adapun *blackbox testing* adalah pengujian aplikasi yang menilai bagaimana antarmuka aplikasi bekerja tanpa melihat program.

4.3.1 Pengujian Kinerja Antarmuka

Adapun komponen-komponen yang akan diuji pada pengujian kinerja antarmuka aplikasi dapat dilihat pada tabel 4.1

Tabel 4.1 Rancangan Pengujian Komponen

No.	Komponen yang diuji	Butir uji
1	Tampilan menu utama	Pengujian tampilan dan tombol
2	Tampilan form objek	Pengujian tombol dan input
3	Tampilan kamera <i>AR</i>	Pengujian tampilan
4	Tampilan objek virtual	Pengujian tampilan
5	Tampilan sudut aplikasi	Pengujian tampilan

Setelah komponen tersebut diuji didalam aplikasi, didapatkan hasil sebagai berikut :

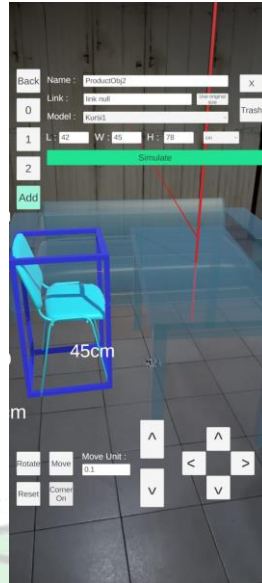
Tabel 4.2 Hasil Pengujian Antarmuka

No.	Komponen	Target Uji	Hasil	Keterangan
1	Tampilan menu utama	Pengujian tampilan menu dan tombol	Berhasil	Tombol “Simulate” berhasil mengganti mode aplikasi ke mode <i>AR</i> dan tombol “quit” berhasil menutup aplikasi.
2	Tampilan form objek	Inputan dan tombol-tombol form dan umpan balik dari form.	Berhasil	Inputan form dan tombol-tombol form bekerja dengan baik.

3	Tampilan kamera <i>AR</i>	Menguji kamera dan deteksi <i>marker</i>	Berhasil	Kamera <i>smartphone</i> berhasil mengidentifikasi <i>marker</i> yang telah ditentukan.
4	Tampilan objek virtual	Uji penampilan objek <i>3D</i> dalam mode <i>AR</i> .	Berhasil	Objek <i>3D</i> virtual berhasil ditampilkan di mode <i>AR</i> .
5	Tampilan sudut dan sisi objek virtual	Uji penampilan sudut pada objek virtual	Berhasil	Objek <i>3D</i> sudut berhasil dimapping dengan benar pada objek virtual.

4.3.2 Pengujian Komponen Aplikasi

Tahap selanjutnya adalah pengujian komponen aplikasi secara langsung yang digunakan oleh pengguna. Saat aplikasi digunakan oleh pengguna, pengguna tidak memiliki batasan dalam segi arah dan orientasi perangkat. Oleh karena itu, aplikasi menggunakan sensor *gyro* yang ada di *smartphone* pengguna untuk mendapatkan arah dan orientasi di perangkat *mobile*. Aplikasi akan menampilkan objek virtual selama gambar visual yang didapat melalui kamera jelas. Ketika kamera mendapatkan gambaran yang kurang jelas, seperti cahaya yang didapat terlalu terang atau terlalu gelap, aplikasi akan mengalami disorientasi untuk menentukan pusat ruang lingkup. Ini mengharuskan penulis harus menambahkan sebuah satu objek sederhana yang bisa menjadi sebuah titik tumpu untuk membantu pengguna melihat bagaimana ruang lingkup yang dibuat oleh aplikasi bekerja. Objek ini dapat dilihat seperti gambar berikut:



Gambar 4.5 Objek merah menandakan titik orientasi.

Aplikasi mendapatkan ukuran produk melalui sebuah server yang menganalisis halaman produk untuk mendapatkan rangkaian *string* yang menandakan dimensi produk. Ini membuat aplikasi sangat bergantung dengan server eksternal untuk memberikan fungsionalitas yang lengkap untuk benar-benar memberikan pengalaman yang bagus untuk pengguna. Sejah ini, server berjalan dengan mulus dan mampu memberikan respon yng baik kepada aplikasi.



Gambar 4.6 Aplikasi sedang melakukan permintaan *API*.

Deteksi gambar pada aplikasi *AR* bergantung pada kualitas kamera perangkat *mobile* untuk memberikan visualisasi gambar *tracker* yang lebih jelas. Kesimpulan ini diambil dikarenakan deteksi gambar tidak akan mengalami perkembangan walaupun aplikasi diberi konfigurasi memakai gambar yang memiliki resolusi lebih tinggi tetapi *ARCore* memberi instruksi untuk memberikan resolusi minimal 300 x 300 *pixel* sebagai gambar *tracker* (Ashray Pai, 2022). Penulis memberikan fokus pada hal ini dikarenakan deteksi gambar *tracker* terkadang buruk dan terkadang bagus sewaktu-waktu. Tapi, setelah diteliti kembali, penulis mendapatkan data lebih untuk memperkirakan apa yang sebenarnya menjadi tolak ukur terdeteksinya *marker* yaitu perbandingan antara *pixel* layar perangkat dan besarnya *marker* terlihat di kamera perangkat. Hal ini diasumsikan penulis karena terjadi perbedaan jarak antara layar dan *marker* ketika *marker* yang digunakan berbeda ukuran. Percobaan ini dilakukan di 2 perangkat *smartphone* yang masing masing memiliki 1080 x 2400 *px* dan 720 x 1480 *px*. Penulis melakukan 5 kali percobaan untuk masing-masing perangkat *mobile*. Berikut adalah tabel percobaan yang dilakukan untuk mencapai kesimpulan ini.

Tabel 4.3 Tabel Percobaan Perbandingan Layar dan Besar Marker di Layar Perangkat Mobile

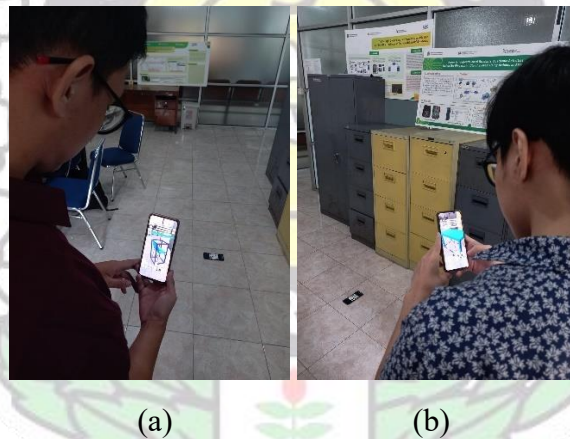
Resolusi Layar Perangkat (<i>px</i>)	Perkiraan Resolusi <i>Marker</i> Ketika Terdeteksi (<i>px</i>)	Ukuran <i>Marker</i> (cm)	Rata-rata Perbandingan Antara Layar dan Besar <i>Marker</i> di Layar
720 x 1480	325 x 320	16 x 16	0.1 dari besar layar perangkat
	375 x 375	16 x 16	
	375 x 380	6 x 6	
	300 x 305	6 x 6	
	319 x 303	6 x 6	
1080 x 2400	440 x 454	16 x 16	0.09 dari besar layar perangkat
	510 x 500	16 x 16	
	525 x 490	6 x 6	
	508 x 522	16 x 16	

	545 x 533	6 x 6	
--	-----------	-------	--

Dari percobaan diatas, dapat dilihat bahwa terjadi perbedaan yang minim pada resolusi *marker* yang terlihat dilayar walaupun ukuran *marker* berbeda pada masingmasing perangkat. Karena hal ini, penulis bisa mengambil kesimpulan bahwa terdeteksinya sebuah *marker* tidak dipengaruhi oleh jarak tetapi perbandingan antara resolusi layar perangkat dan resolusi *marker* yang terlihat di layar perangkat.

4.3.3 Pengujian Penggunaan Aplikasi Terhadap Pengguna

Pengujian dengan pengguna dilakukan untuk mengetahui tanggapan pengguna terhadap kinerja aplikasi. Ini juga membantu menjawab pertanyaan apakah pengguna merasa mudah dalam berbelanja *online* menggunakan aplikasi ini. Pengujian ini dilakukan oleh 30 pengguna yang perangkatnya memenuhi spesifikasi minimum aplikasi yang sudah dibahas di bab sebelumnya. Setelah pengujian, pengguna diberikan kuisioer untuk merekam bagaimana keberhasilan aplikasi dan tanggapan pengguna terhadap aplikasi ini.



Gambar 4.7 Pengujian dilakukan oleh pengguna.

Hasil dari kuisioer yang diberikan kepada pengguna dapat dilihat di tabel 4.2. Pertanyaan yang diberikan memuat standar USE (*Usefulness, Satisfaction, and Ease of Use*) yang dibuat oleh Lund, A.M. (2001).

Tabel 4.4 Tanggapan Pengguna Terkait Aplikasi

Tanggapan mengenai penggunaan aplikasi							
No.	Pernyataan	STS	TS	N	S	SS	Persentase
1	Aplikasi ini membantu saya menjadi lebih efektif	0	0	1	16	13	SS,S = 96.9% N = 3.3% STS,TS = 0%
2	Aplikasi ini membantu saya menjadi lebih produktif	0	0	3	13	14	SS, S = 90% N = 10% STS, TS = 0%
3	Aplikasi ini sangat berguna	0	0	1	11	18	SS, S = 96.67% N = 3.33% STS, TS = 0%
4	Aplikasi ini memberikan saya lebih banyak kontrol dalam berbelanja.	0	0	4	12	14	SS, S = 86.67% N = 13.33% STS, TS = 0%
5	Aplikasi ini membuat visualisasi produk lebih mudah untuk dicapai	0	0	2	13	15	SS, S = 93.33% N = 6.67% STS, TS = 0%
6	Aplikasi ini memenuhi kebutuhan berbelanja <i>online</i> saya	0	0	3	8	19	SS, S = 90% N = 10% STS, TS = 0%
7	Aplikasi ini menghemat waktu saya	0	1	5	6	18	SS, S = 80% N = 16.67% STS, TS = 3.33%

8	Aplikasi ini melakukan apa yg saya ekspektasikan	0	0	3	11	16	SS, S = 90% N = 10% STS, TS = 0%
Tanggapan mengenai kemudahan aplikasi							
1	Saya belajar menggunakannya dengan cepat	0	0	4	8	17	SS, S = 83.33% N = 13.33%%
2	Saya mengingat cara menggunakan aplikasi ini dengan mudah	0	0	3	13	14	SS, S = 90% N = 10% STS, TS = 0%
3	Mempelajari aplikasi ini sangat mudah	0	0	2	13	15	SS, S = 93.33% N = 6.67% STS, TS = 0%
4	Saya cepat mahir menggunakan aplikasi ini	0	0	6	8	16	SS, S = 80% N = 20% STS, TS = 0%
Tanggapan mengenai kepuasan pengguna							
1	Saya puas menggunakan aplikasi ini	0	0	1	10	19	SS, S = 96.67% N = 3.33% STS, TS = 0%
2	Saya akan merekomendasikan aplikasi ini kepada teman-teman saya	0	0	4	9	17	SS, S = 86.67% N = 13.33% STS, TS = 0%
3	Aplikasi ini sangat menyenangkan untuk dipakai	0	0	5	10	15	SS, S = 83.33% N = 16.67% STS, TS = 0%
4	Aplikasi ini bekerja sebagaimana saya	0	0	2	12	16	SS, S = 93.33% N = 6.67% STS, TS = 0%

	ingin aplikasi ini bekerja						
5	Aplikasi ini sangat bagus	0	0	3	9	18	SS, S = 90% N = 10% STS, TS = 0%
6	Saya merasa saya akan membutuhkan aplikasi ini	0	0	4	14	12	SS, S = 86.67% N = 13.33% STS, TS = 0%
7	Aplikasi nyaman digunakan	0	0	3	11	16	SS, S = 90% N = 10% STS, TS = 0%

Ketrangan Kuisisioner :

SS = Sangat Setuju

S = Setuju

N = Netral

TS = Tidak Setuju

STS = Sangat Tidak Setuju

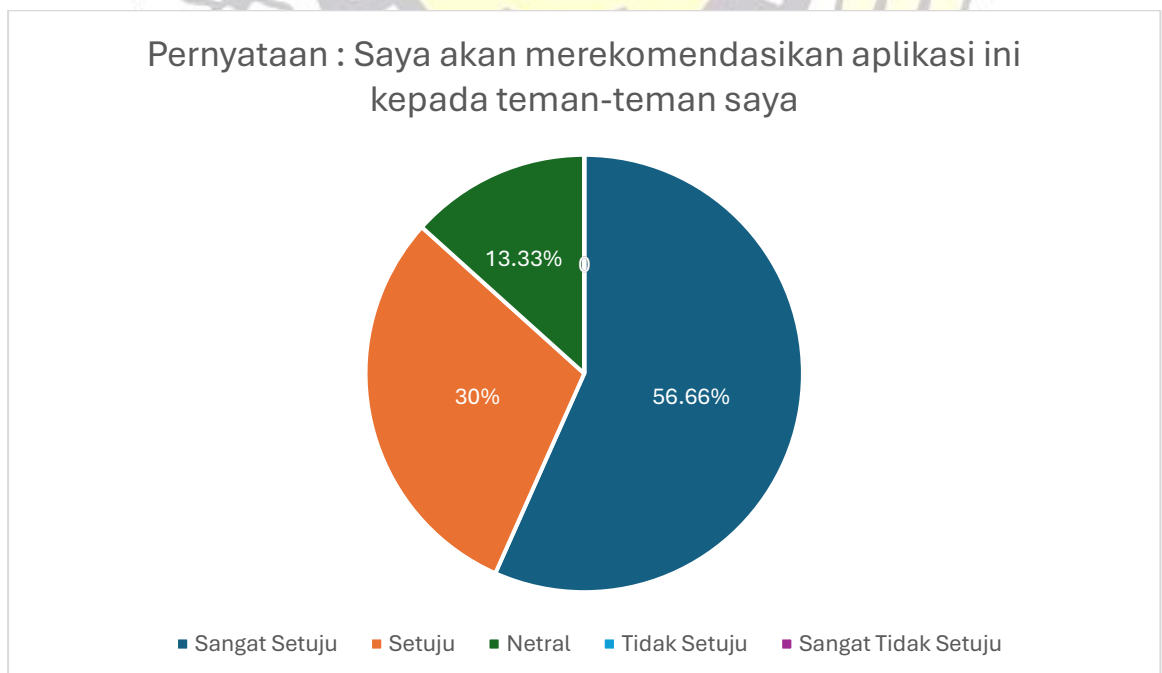
Dapat dilihat dari tabel diatas bahwa nilai kepuasan yang dihasilkan dari kusioner memiliki nilai positif secara umum untuk setiap pertanyaan. Pertanyaan-pertanyaan dibagi menjadi 3 aspek yaitu, aspek penggunaan, aspek kemudahan dan yang terakhir aspek kepuasan. Nilai persentase keseluruhan bisa dilihat di tabel 4.5 berikut:

Tabel 4.5 Nilai Kepuasan Aplikasi Berdasarkan Aspek

Aspek Penilaian	Nilai Kepuasan(%)
Tanggapan mengenai penggunaan aplikasi	$S/SS = \frac{96.67\%+90\%+96.67\%+86.67\%+93.33\%+90\%+80\%+90\%}{8} =$ $\frac{723.34}{8} = 90.41\%$

	$N = \frac{3.33\%+10\%+3.33\%+13.33\%+6.67\%+10\%+16.67\%+10\%}{8} = \frac{73.33}{8} = 9.17\%$ $TS/STS = \frac{0\%+0\%+0\%+0\%+0\%+0\%+3.33\%+0\%}{8} = \frac{3.33}{8} = 0.42\%$
Tanggapan mengenai kemudahan aplikasi	$S/SS = \frac{83.33\%+90\%+93.33\%+80\%}{4} = \frac{346.66}{4} = 86.66\%$ $N = \frac{13.33\%+10\%+6.67\%+20\%}{4} = \frac{50}{4} = 12.5\%$ $TS/STS = \frac{3.33\%+0\%+0\%+0\%}{4} = \frac{3.33}{4} = 0.83\%$
Tanggapan mengenai kepuasan pengguna	$S/SS = \frac{96.67\%+86.67\%+83.33\%+93.33\%+90\%+86.67\%+90\%}{7} = \frac{626.67}{7} = 89.52\%$ $N = \frac{3.33\%+13.33\%+16.67\%+6.67\%+10\%+13.33\%+10\%}{7} = \frac{73.33}{7} = 10.47\%$ $TS/STS = \frac{0\%+0\%+0\%+0\%+0\%+0\%+0\%}{7} = \frac{0}{7} = 0\%$

Pengguna aplikasi juga diberikan pertanyaan mengenai apakah pengguna merekomendasikan aplikasi kepada teman atau keluarga pengguna. Pertanyaan ini persentase pengguna yang merekomendasikan aplikasi ini kepada teman dan keluarga sebesar 86,67%. Perbandingan untuk pertanyaan tersebut dapat dilihat di grafik berikut :



Gambar 4.8 Grafik tanggapan rekomendasi pengguna.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari hasil implementasi penelitian pada bab sebelumnya adalah sebagai berikut :

1. Aplikasi dibuat khusus untuk perangkat *android* dengan sistem operasi *android* minum *android 7(nougat)* yang memiliki *ARCore*.
2. Aplikasi berjalan dengan baik selama kamera perangkat *mobile* berfungsi. Untuk menggunakan fungsi *link-to-model*, dibutuhkan koneksi internet untuk mengirim permintaan *API* dan menerima responsnya.
3. Fungsi pengambilan ukuran melalui tautan Tokopedia yang dimiliki sistem aplikasi bergantung kepada server *python* yang dihost secara *remote*. Jika server tersebut tidak aktif lagi, fungsi tersebut tidak bisa digunakan namun aplikasi masih bisa digunakan jika pengguna mengingat ukuran produk tersebut.
4. Kemampuan aplikasi untuk memberikan gambaran visual model *3D* akan mengalami ketidakstabilan jika kamera perangkat mendeteksi terlalu banyak cahaya atau cahaya yang ditangkap kamera terlalu sedikit yang menyebabkan aplikasi kesusahan untuk melakukan *mapping* pada *frame* yang ditangkap kamera.
5. Perangkat *mobile Android* akan dapat mendeteksi *marker* ketika *marker* berukuran sekitar sepersepuluh ($1/10$) dari resolusi layar perangkat *mobile*.
6. Sebanyak 90% pengguna yang menggunakan aplikasi setuju bahwa aplikasi ini membantu mereka lebih produktif dalam berbelanja, 10% berpendapat netral dan 0% untuk pengguna yang tidak setuju.
7. Aplikasi ini bisa dijalankan secara online ataupun *offline*.

5.2 Saran

Berdasarkan kelebihan dan kekurangan aplikasi yang didapatkan dari penelitian ini, berikut beberapa saran yang penulis berikan sebagai pertimbangan untuk penelitian selanjutnya :

1. Diharapkan kedepannya aplikasi dalam versi iOS untuk memperluas cakupan pengguna.
2. Diharapkan untuk memberikan pembaruan terhadap fitur yang mencakup *ARFoundation* dikarenakan paket *ARFoundation* akan terus mengalami pembaruan dan memiliki versi yang lebih baru.
3. Diharapkan untuk memberika model-model *3D* yang lebih banyak dan lebih umum agar mencakup lebih banyak produk untuk ditampilkan.
4. Diharapkan untuk memberikan fitur untuk mengotomisasi pemilihan model berdasarkan deskripsi produk.
5. Diharapkan untuk memberikan fitur untuk menampilkan tekstur produk agar objek produk bisa lebih mendekati produk aslinya.

DAFTAR PUSTAKA

- Akbar. 2019. Apa itu Unity ?. Diakses pada 8 Januari 2025 dari <https://akbarproject.com/apa-itu-unity/>
- Alves, Carlos & Reis, José. (2020). *The Intention to Use E-Commerce Using Augmented Reality - The Case of IKEA Place*. 10.1007/978-3-030-40690-5_12.
- Iptek Digital. 2024. *AR Marker dan AR Markerless* - Iptek Digital. Diakses pada 8 Januari 2025 dari <https://iptek.co.id/ar-marker-dan-ar-markerless/>.
- Azuma, R. (1997). *A Survey of Augmented Reality*. *Presence: Teleoperators & Virtual Environments*, 6, 355-385. <https://doi.org/10.1162/pres.1997.6.4.355>.
- Baruah, Mintu. (2019). *Deskkart- Furniture Mall with Augmented Reality*. *International Journal for Research in Applied Science and Engineering Technology*. 7. 653-660. 10.22214/ijraset.2019.5113.
- Bonetti, F., Pantano, E., Warnaby, G., Quinn, L., Perry, P. (2019). *Augmented Reality in Real Stores: Empirical Evidence from Consumers' Interaction with AR in a Retail Format*. In: tom Dieck, M., Jung, T. (eds) *Augmented Reality and Virtual Reality. Progress in IS*. Springer, Cham. https://doi.org/10.1007/978-3-030-06246-0_1
- Bonetti, F., Warnaby, G., & Quinn, L. (2017). *Augmented reality and virtual reality in physical and online retailing: A review, synthesis and research agenda*. In T. Jung & M. Tom Dieck (Eds.), *Augmented reality and virtual reality* (pp. 119–132). New York: Springer.
- Cmlabs. 2024. Apa itu Regex? Definisi, Manfaat, dan Cara Membuatnya Diakses pada 8 Januari 2025 dari <https://cmlabs.co/id-id/seo-guidelines/apa-itu-regex>
- Google. 2024. *Fundamental concepts | ARCore | Google for Developers*. Diakses pada 8 Januari 2025 dari <https://developers.google.com/ar/develop/fundamentals>.
- Gangurde, M. (2011). *Augmented reality*, 1363. <https://doi.org/10.1145/1980022.1980349>.

- Jadhav, Karuna & V.P., Sriram & Ambikapathy, Arun & M.A., Sikandar & Gulati, Kamal & Kumar, Narinder. (2021). *The rise of 3D E-Commerce: the online shopping gets real with virtual reality and augmented reality during COVID-19. World Journal of Engineering. ahead-of-print.* 10.1108/WJE-06-2021-0338.
- Javornik, A. (2016). Augmented reality: *Research agenda for studying the impact of its media characteristics on consumer behaviour. Journal of Retailing and Consumer Services*, 30, 252–261.
- Joshi, Shreyash & Walavalkar, Pallav & Shetkar, Pooja & Joshi, Sumedh & Mahale, Shubham & Bhogan, Snehal & Sawant, Kedar. (2020). *Product Visualization Using Augmented Reality.* 7. 2395-0056.
- Moraneus. 2024. *Guide to Using Python Selenium.* Diakses pada 8 Januari 2024 dari <https://medium.com/@moraneus/guide-to-using-python-selenium-873342d5fdad>
- DQLab. 2021. *Mengenai Flask, Library Machine Learning Python Idaman Developer.* Diakses pada 8 Januari 2025 dari <https://dqlab.id/mengenai-flask-library-machine-learning-python-idaman-developer>
- Pai, Ashray. 2022. *AR Foundation Image Tracking.* Diakses 18 November 2024, dari <https://immersive-insiders.com/blog/ar-foundation-image-tracking#:~:text=Here%20are%20some%20of%20the,PNG%20or%20JPEG%20file%20format.>
- Pantano, E., & Priporas, C.-V. (2016). *The effect of mobile retailing on consumers' purchasing experiences: A dynamic perspective. Computers in Human Behavior*, 61, 548–555.
- Safrodin, Mohamad & Zikky, Moh & Khozi, Saiful & Wicaksono, M. (2020). *The 3D virtual drawing mobile application based on augmented reality using AR-Framework. Journal of Physics: Conference Series.* 1450. 012078. 10.1088/1742-6596/1450/1/012078.
- Zhao, W. (2018). *Research on the Fusion of AR Technology in Home Space Design.* . <https://doi.org/10.2991/ICED-18.2018.26>.

