

Implementasi Teknologi *Augmented Reality* Pada *Photobooth* Interaktif Tanpa Kamera Pemindai Kedalaman Untuk Meningkatkan Pengalaman Pengguna Dalam Aplikasi Jelita

1st Adam Aria Rahman
School of Applied Science
Telkom University

Bandung, Indonesia
khalilahatikaakmal@student.telkomuni-
versity.ac.id

2nd Khalilah Atika Akmal
School of Applied Science
Telkom University

Bandung, Indonesia
adamrahman@student.telkomuniversity.
.ac.id

3rd Fat'hah Noor Prawita, S.T., M.T.
School of Applied Science
Telkom University

Bandung, Indonesia
fathah@telkomuniversity.ac.id

Abstract— Modern *photobooths* increasingly incorporate *Augmented Reality* (AR) technology to enhance user interaction and visual engagement. However, most AR-based *photobooths* still rely on depth-sensing cameras, which are costly and less flexible for general deployment. This study aims to develop an AR *photobooth* application named JELITA (Jepratan Lincih Augmented Tanpa Sensor Kedalaman), designed to operate without a depth-sensing camera, thereby offering a more accessible and easily deployable solution across various devices. The application was developed using the MediaPipe framework and Kotlin programming language and tested using standard *smartphone* cameras as a practical and flexible approach for cross-device evaluation. Core features implemented include *real-time* pose tracking, virtual clothing filter selection, and automatic photo capture and storage. Functional testing demonstrated that all major features operated reliably and as designed. Furthermore, user testing indicated high usability and positive reception toward the concept of an AR *photobooth* without the need for additional hardware. These findings suggest that JELITA not only reduces dependency on external devices, but also demonstrates infrastructural efficiency by requiring fewer hardware components while maintaining performance. Therefore, the application presents a cost-effective, efficient, and scalable AR *photobooth* solution suitable for various public event settings.

Keywords: *Photobooth, Augmented Reality, MediaPipe, Smartphone Camera, Interactive experience*

I. PENDAHULUAN

A. Latar Belakang

Fotografi telah berkembang pesat dan menjadi bidang yang secara dinamis memengaruhi cara masyarakat mengabadikan dan menyimpan kenangan. Perkembangan teknologi mendorong berbagai inovasi dalam fotografi, salah satunya adalah *photobooth*, yang kini semakin populer sebagai medium interaktif di berbagai acara dan ruang publik, menawarkan pengalaman fotografi yang unik dan modern. *Photobooth* adalah sebuah area yang disediakan oleh penyelenggara acara di mana tamu dapat berpose untuk difoto dan langsung mendapatkan hasilnya saat itu juga [1]. Memasuki tahun 2018, *photobooth* mulai dipergunakan masyarakat secara umum di berbagai acara penting seperti pernikahan maupun pesta lainnya. Hal ini dikarenakan meningkatnya kegemaran masyarakat dalam mengabadikan foto di suatu kegiatan acara serta ketertarikan masyarakat terhadap *photobooth* karena keunikannya [2].

Di sini muncul kebutuhan akan solusi yang lebih praktis melalui pengembangan aplikasi bernama JELITA (Jepratan Lincih Augmented Tanpa Sensor Kedalaman), yang memanfaatkan teknologi *Augmented Reality* (AR) tanpa ketergantungan pada kamera pengukur kedalaman (*depth sensing camera*), melainkan menggunakan kamera web. Seiring dengan kemajuan dalam kemampuan perangkat lunak AR, aplikasi seperti Snapchat dan Instagram telah membuktikan bahwa efek AR dapat berfungsi secara optimal tanpa memerlukan kamera khusus. Aplikasi-aplikasi tersebut menggunakan teknologi deteksi gambar, pelacakan tubuh, dan pengenalan pola untuk menciptakan efek interaktif yang realistis.

Perlu dipahami bahwa *photobooth* berbasis AR memiliki kebutuhan berbeda dari *photobooth* konvensional. Jika *photobooth* umum mengandalkan kamera DSLR untuk menghasilkan foto berkualitas tinggi, maka *photobooth* AR lebih menekankan pada interaksi digital dan pelacakan tubuh secara real time. Untuk itu, umumnya dibutuhkan kamera dengan kemampuan *depth sensing*. Namun, berdasarkan hasil wawancara dengan praktisi di bidang multimedia dan teknologi AR/VR, solusi berbasis *depth sensing* dinilai masih memiliki hambatan dari sisi biaya dan ketersediaan perangkat. Hal inilah yang menjadi latar belakang utama dikembangkan aplikasi JELITA.

Situasi tersebut membuka peluang untuk menerapkan konsep serupa dalam pengembangan aplikasi JELITA. Dengan memanfaatkan kamera web yang lebih umum, penggunaan JELITA pada *photobooth* dapat diakses oleh lebih banyak kalangan dan diterapkan dalam berbagai acara dengan lebih mudah dan efisien. Selain itu, solusi ini juga dapat mengurangi biaya produksi serta meningkatkan fleksibilitas dalam pemasangan perangkat *photobooth* di berbagai lokasi. Pendekatan ini diharapkan mampu menghasilkan solusi yang inovatif, mudah diimplementasikan secara luas, serta lebih ekonomis dengan tingkat kompleksitas teknis yang minimal.

B. Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah yang akan dibahas adalah sebagai berikut:

1. Bagaimana *photobooth* interaktif berbasis *Augmented Reality* (AR) dapat meningkatkan *user experience* tanpa bergantung pada perangkat *depth sensing*?

2. Bagaimana solusi *photobooth* AR tanpa *depth sensing camera* dapat mengurangi biaya dan kompleksitas perangkat?

C. Tujuan

Berdasarkan rumusan masalah yang ada, tujuan yang akan dicapai adalah:

1. Menciptakan solusi teknologi *photobooth* berbasis AR melalui aplikasi JELITA yang dapat berfungsi tanpa memerlukan kamera pemindai kedalaman, sehingga menjadikannya sebagai alternatif yang lebih efisien dan hemat biaya.
2. Mengimplementasikan MediaPipe sebagai pendekatan alternatif dalam pengembangan *photobooth* AR.

D. Batasan Masalah

Batasan masalah dalam pengembangan aplikasi JELITA adalah:

1. *Photobooth* dirancang untuk berjalan menggunakan kamera web standar tanpa *depth sensing* karena keterbatasan perangkat keras yang tersedia, pengujian prototipe aplikasi dilakukan pada kamera *smartphone*.
2. Menggunakan bahasa pemrograman Kotlin.
3. Aplikasi JELITA pada *photobooth* ditujukan untuk digunakan dalam berbagai acara sosial, seperti pameran seni, dan promosi.

E. Metode Penyelesaian Masalah

Berikut adalah metodologi penyelesaian masalah yang digunakan dalam proyek akhir ini:

1. Studi Literatur Mencari dan mempelajari sumber dari buku, jurnal ilmiah, makalah, serta referensi daring yang terkait dengan AR, *photobooth* interaktif, dan teknologi pengenalan pola serta deteksi badan menggunakan kamera web standar. Selain itu, memahami penerapan teknologi AR pada perangkat *depth sensing camera* dan meninjau aplikasi AR serupa yang berhasil menerapkan AR menggunakan kamera web.
2. Analisis Kebutuhan Mengidentifikasi kebutuhan teknis dan non-teknis untuk *photobooth* interaktif berbasis AR melalui penelitian pasar dan survei terhadap calon pengguna, seperti penyelenggara acara. Selain itu, menentukan spesifikasi perangkat keras yang akan digunakan, yaitu kamera web standar, serta mengidentifikasi kemampuan AR yang bisa dicapai tanpa menggunakan sensor kedalaman.
3. Perancangan Aplikasi Merancang arsitektur aplikasi JELITA. Ini mencakup perancangan alur kerja aplikasi, algoritma pengolahan gambar untuk deteksi tubuh, serta penambahan efek visual atau filter AR.
4. Pembuatan Aplikasi Pengembangan aplikasi JELITA menggunakan bahasa pemrograman Kotlin dengan Android Studio. Selain itu, untuk menerapkan algoritma deteksi tubuh digunakan teknologi computer vision, yaitu MediaPipe.

5. Pengujian Aplikasi Pada tahapan ini dilakukan pengujian untuk mengobservasi kesalahan yang mungkin terjadi pada aplikasi, sehingga dapat dipastikan aplikasi berjalan sesuai dengan yang diharapkan. Pengujian dilakukan dalam dua tahap, pertama oleh developer aplikasi, kemudian dengan mitra dan pengguna lainnya. Pada tahap pengujian aplikasi JELITA, dilakukan pengujian internal dan eksternal untuk memastikan fungsionalitas dan performa sesuai harapan. Developer akan melakukan pengujian pertama kali, lalu dilanjutkan dalam kondisi lingkungan nyata, seperti di event atau lokasi dengan banyak pengguna.

II. PENELITIAN TERKAIT

A. Android Studio

Android Studio adalah Integrated Development Environment (IDE) resmi untuk pengembangan aplikasi Android, yang menyediakan berbagai alat dan fitur untuk membantu pengembang dalam menciptakan aplikasi yang berkualitas. IDE ini mendukung penggunaan Gradle sebagai sistem build, yang memungkinkan pengembang untuk mengelola dependensi dan konfigurasi proyek dengan lebih efisien [5]. Gradle memfasilitasi pembuatan berbagai varian build dari satu proyek, sehingga memudahkan pengembang dalam mengelola aplikasi untuk berbagai perangkat Android [6].

Pengaturan Proyek: Untuk menggunakan MediaPipe dalam Android Studio, pengembang harus membuat proyek baru atau menggunakan proyek yang sudah ada. MediaPipe menyediakan pustaka yang dapat diintegrasikan ke dalam proyek Android melalui Gradle [7]. **Integrasi MediaPipe:** MediaPipe menyediakan pre-built libraries yang dapat digunakan untuk berbagai tugas seperti *FaceMesh*, *Hands*, dan *Face Detection*. Pengembang dapat menambahkan dependensi ini ke dalam file build.gradle proyek mereka untuk mengakses fungsionalitas yang disediakan oleh MediaPipe [8]. **Penggunaan Pre-Built Lib Files:** Dengan menggunakan pre-built libraries dari MediaPipe, pengembang dapat dengan mudah mengimplementasikan fitur-fitur seperti *FaceMesh* dan *Hands* tanpa harus membangun model dari awal. Ini menghemat waktu dan usaha dalam pengembangan aplikasi [9]. Pengembang hanya perlu memastikan bahwa file model yang diperlukan tersedia di direktori aset aplikasi [10].

Implementasi Fitur: Setelah dependensi yang diperlukan ditambahkan, pengembang dapat mulai menggunakan API MediaPipe dalam kode aplikasi mereka. Misalnya, untuk menggunakan *Face Detection*, pengembang dapat memanggil API yang sesuai dan mengatur parameter yang diperlukan untuk mendeteksi wajah dalam gambar atau video [11].

Penggunaan Android Studio dengan Gradle untuk mengintegrasikan MediaPipe memberikan kemudahan bagi pengembang dalam menciptakan aplikasi yang memanfaatkan teknologi pengolahan gambar dan video. Dengan adanya pre-built libraries, pengembang dapat dengan cepat mengimplementasikan fitur-fitur canggih seperti

FaceMesh, *Hands*, dan *Face Detection*, sehingga meningkatkan efisiensi dan produktivitas dalam pengembangan aplikasi [12].

B. Kotlin

Kotlin adalah bahasa pemrograman statis yang dikembangkan oleh JetBrains, dirilis pertama kali pada tahun 2011 [13]. Kotlin mendukung pemrograman berorientasi objek dan pemrograman fungsional, sehingga memberikan fleksibilitas yang lebih besar kepada pengembang dalam menulis kode [14]. Salah satu fitur utama dari Kotlin adalah *null safety*, yang dirancang untuk mengurangi kesalahan *NullPointerException* yang sering terjadi dalam pemrograman Java. Dengan menggunakan tipe *nullable* dan *non-nullable*, Kotlin membantu pengembang untuk lebih mudah mengelola nilai yang mungkin *null* [15]. Selain itu, Kotlin juga menawarkan fitur-fitur seperti ekstensi fungsi, *data class*, dan *coroutines*, yang memungkinkan pengembang untuk menulis kode yang lebih bersih dan efisien [16].

C. Augmented Reality (AR)

Sari menyebutkan bahwa AR merupakan teknologi yang mengintegrasikan objek *virtual* dua dimensi atau tiga dimensi ke dalam lingkungan nyata tiga dimensi dan menampilkan objek-objek *virtual* tersebut secara langsung dalam waktu nyata [17].

Selain itu, Cahyaningsih menyebutkan bahwa AR adalah suatu teknologi yang mengintegrasikan objek maya dua dimensi maupun tiga dimensi ke dalam dunia nyata yang bersifat tiga dimensi, lalu memproyeksikan objek-objek maya tersebut dalam waktu nyata [18].

Selain itu, menurut Griffith, AR merupakan teknologi yang menggabungkan objek *virtual* dengan objek nyata secara bersamaan. *Augmented Reality* (AR) mampu memberikan pengalaman interaktif yang menarik bagi pengguna, karena teknologi ini memungkinkan pengguna untuk merasakan seolah-olah objek *virtual* benar-benar ada di dalam lingkungan nyata mereka [4].

Berdasarkan ketiga pendapat di atas, dapat disimpulkan bahwa AR merupakan teknologi yang menggabungkan objek *virtual*, baik dua dimensi maupun tiga dimensi, dengan dunia nyata yang memiliki tiga dimensi. Objek-objek *virtual* tersebut kemudian ditampilkan secara langsung dalam waktu nyata, sehingga memungkinkan interaksi antara dunia nyata dan objek *virtual*. Dengan demikian, AR menciptakan pengalaman yang membuat pengguna merasa seolah-olah objek *virtual* itu benar-benar ada di sekeliling mereka, memberikan pengalaman interaktif yang menarik serta memperluas cara pandang pengguna terhadap lingkungan di sekitar mereka.

D. Mediapipe

MediaPipe adalah sebuah framework untuk membangun *pipeline* yang melakukan inferensi terhadap data sensor secara umum. Dengan MediaPipe, sebuah *perception pipeline* dapat dibangun sebagai grafik dari komponen-komponen modular, termasuk inferensi model, algoritma pemrosesan media, dan transformasi data, dan sebagainya. Data sensor seperti aliran audio dan video masuk ke dalam grafik, dan deskripsi hasil persepsi seperti aliran pelokasian objek dan landmark wajah keluar dari grafik tersebut [19]. Dalam konteks MediaPipe, *pipeline* adalah alur kerja terstruktur yang menyusun proses inferensi dan pemrosesan data sensor (seperti citra atau video) menjadi grafik modular.

Pose Landmarker dalam Mediapipe menggunakan model *machine learning* yang mendeteksi hingga 33 titik landmark pada tubuh manusia. Landmark ini mencakup berbagai bagian tubuh seperti kepala, bahu, siku, pinggul, lutut, dan pergelangan kaki. Model ini berfungsi untuk mengidentifikasi posisi kunci tubuh, menganalisa postur, dan mengkategorikan gerakan [20]. Dengan kemampuan deteksi landmark yang cepat dan akurat, Pose Landmarker menjadi alat yang sangat berharga bagi pengembang yang ingin menciptakan pengalaman interaktif dan imersif bagi pengguna mereka [19].

E. Kamera Pemindai Kedalaman

Sensor kamera kedalaman adalah sensor yang dapat mendeteksi jarak objek dengan menggunakan teknologi pengukuran jarak pada kamera. Sensor ini banyak digunakan dalam berbagai aplikasi seperti robotika, otomasi industri, dan pengembangan gim [21].

Sensor kamera kedalaman bekerja berdasarkan prinsip pengukuran jarak dengan mengukur waktu tempuh cahaya. Cara kerjanya adalah sebagai berikut:

1. Sensor kamera kedalaman mengirimkan sinyal cahaya ke objek yang ingin diukur jaraknya.
2. Sinyal cahaya tersebut dipantulkan oleh objek dan kembali ke sensor.
3. Sensor kemudian mengukur waktu tempuh cahaya dari saat sinyal dikirim hingga sinyal yang dipantulkan diterima kembali.
4. Dengan menggunakan prinsip kecepatan cahaya, sensor dapat menghitung jarak objek dari kamera [22].
5. Meskipun memiliki banyak keunggulan, sensor kamera kedalaman juga memiliki beberapa kekurangan, antara lain: a. Kualitas sensor akan menurun jika digunakan dalam kondisi lingkungan dengan cahaya yang terlalu terang atau terlalu gelap.
6. Bentuk dan warna objek juga mempengaruhi akurasi pengukuran sensor.
7. Sensor memerlukan kalibrasi yang tepat agar hasil pengukuran menjadi akurat [23].

F. Bodytracking

Teknologi pelacakan tubuh (*body tracking*) telah memungkinkan pendeteksian pergerakan tubuh secara menyeluruh dengan memanfaatkan kamera dan perangkat lunak secara terpadu, sebagaimana dijelaskan dalam studi terdahulu yang dikutip dalam [24].

G. Photobooth

Istilah *photobooth* berasal dari kata photo yang merujuk pada aktivitas pengambilan gambar suatu objek (fotografi), dan booth yang dapat diartikan sebagai bilik, stan, atau panggung kecil. Secara umum, *photobooth* dipahami sebagai sebuah area khusus yang disediakan untuk keperluan pengambilan foto, baik dalam bentuk bilik tertutup maupun panggung terbuka. Area ini umumnya dirancang atau dihias sedemikian rupa agar menarik secara visual dan mendukung suasana berfoto bagi pengunjung atau peserta[25].

H. Photobooth AR

Photobooth AR (Augmented Reality) merupakan inovasi dalam teknologi *photobooth* yang menggabungkan elemen dunia nyata dengan elemen digital melalui penggunaan teknologi *Augmented Reality*. *Photobooth AR* memungkinkan pengguna untuk mengambil foto dengan efek digital yang ditambahkan secara *real-time*, menciptakan pengalaman yang lebih interaktif dan menarik dibandingkan dengan 10 *photobooth* tradisional [26]. *Photobooth AR* juga dilengkapi dengan layar interaktif yang menampilkan pratinjau gambar dan efek yang dapat dipilih oleh pengguna. Layar ini memungkinkan pengguna untuk berinteraksi dengan berbagai fitur, seperti memilih filter atau latar belakang, serta melihat hasil foto sebelum diambil [1]. Selain itu, sistem ini sering kali dilengkapi dengan kemampuan untuk mencetak foto atau membagikannya langsung ke media sosial, meningkatkan pengalaman pengguna [27].

I. Aplikasi Serupa

Aplikasi yang serupa dengan proyek akhir ini yaitu:

1. Heromirror

HeroMirror Next-gen AR *Photobooths* adalah perangkat *photobooth* inovatif yang mengintegrasikan teknologi *Augmented Reality* (AR) untuk meningkatkan pengalaman berfoto di berbagai acara. Menggunakan teknologi canggih, HeroMirror memungkinkan pengguna untuk berinteraksi dengan elemen foto secara *real-time*, dengan opsi filter, efek, dan kustomisasi yang lebih imersif. Fitur ini memberikan pengalaman visual yang lebih dinamis dan personal, serta memungkinkan pengguna untuk melihat dan mencoba berbagai tampilan sebelum mengambil foto, menciptakan momen yang lebih interaktif dan menyenangkan. Untuk kelebihan dan kekurangan aplikasi akan di sajikan dalam Tabel 2.1.

Link aplikasi: [Heromirror app](#)

TABEL 1
Kelebihan dan kekurangan Heromirror

Kelebihan Aplikasi	Kekurangan Aplikasi
Pengalaman interaktif Teknologi AR memungkinkan pengguna berinteraksi dengan foto secara langsung, memberikan pengalaman	Keterbatasan perangkat keras Memerlukan perangkat keras yang lebih canggih dan mahal untuk berfungsi secara optimal, yang dapat membatasi

yang lebih menarik dan imersif.	penggunaan di beberapa lokasi.
Kustomisasi efektif Menawarkan berbagai filter, efek, dan opsi kustomisasi yang dapat disesuaikan dengan tema acara atau preferensi pribadi.	Biaya tinggi Investasi awal dan biaya operasional untuk menggunakan HeroMirror dapat cukup mahal, terutama untuk acara kecil dengan anggaran terbatas.
Peningkatan kualitas foto Teknologi canggih meningkatkan kualitas gambar dengan pencahayaan yang lebih baik dan fitur pengeditan otomatis, menghasilkan foto yang lebih profesional.	Kompleksitas pengoperasian Bagi pengguna yang kurang berpengalaman, pengoperasian teknologi AR dan pengaturan awal dapat memerlukan waktu dan keterampilan teknis yang lebih tinggi.

2. Foto Master's Photo Booth Software

Foto Master's Photo Booth Software adalah aplikasi perangkat lunak yang dirancang untuk mengelola dan mengoperasikan perangkat *photobooth* secara efisien dalam berbagai acara, seperti pernikahan, pesta, atau acara perusahaan. Aplikasi ini menyediakan berbagai fitur yang memungkinkan pengguna untuk mengambil foto, menambahkan efek, dan mencetak foto secara langsung, serta mengatur berbagai parameter pengambilan gambar. Dengan antarmuka yang intuitif, Foto Master memungkinkan pengguna, baik penyelenggara acara maupun tamu, untuk menikmati pengalaman interaktif dan personal dalam penggunaan *photobooth*. Selain itu, aplikasi ini mendukung berbagai opsi kustomisasi, seperti desain backdrop, filter foto, dan branding khusus, yang dapat disesuaikan dengan tema acara. Foto Master's Photo Booth Software bertujuan untuk meningkatkan pengalaman tamu dalam berfoto dan membuat acara lebih menarik melalui teknologi yang mudah digunakan dan beragam pilihan kreatif. Pada Tabel 2.2 berikut terdapat kelebihan dan kekurangan dari aplikasi Foto Master's Photo Booth Software.

Link aplikasi: [Foto Master's Photo Booth Software](#)

TABEL 2
Kelebihan dan kekurangan Foto Master's Photo Booth Software

Kelebihan Aplikasi	Kekurangan Aplikasi
Antarmuka intuitif Dirancang untuk memudahkan pengguna dalam mengoperasikan perangkat <i>photobooth</i> tanpa kesulitan teknis.	Fitur terbatas pada versi dasar Beberapa fitur canggih hanya tersedia di versi berbayar.
Kustomisasi fleksibel Menyediakan berbagai opsi untuk menyesuaikan desain, filter, dan elemen acara sesuai kebutuhan.	Kebutuhan perangkat keras khusus Memerlukan perangkat keras tertentu untuk kinerja optimal.
Fitur interkatif Memungkinkan pengalaman pengguna yang lebih menarik dengan	Pengaturan awal yang kompleks Proses konfigurasi awal dapat memakan waktu, terutama bagi pengguna

berbagai efek dan opsi berbagi foto.	yang kurang berpengalaman.
--------------------------------------	----------------------------

3. *Software Virtual Fitting Room*

Software virtual fitting room merupakan aplikasi yang memungkinkan pengguna untuk mencoba berbagai jenis pakaian dan aksesoris secara digital, tanpa perlu mengenakan pakaian tersebut secara langsung. Dengan *software* ini pengguna bisa melihat tampilan mereka dengan berbagai pakaian hanya dengan beberapa tombol yang sangat menghemat waktu dibandingkan dengan berganti pakaian secara fisik. Selanjutnya dalam Tabel 2.3 akan disajikan kelebihan dan kekurangan aplikasi *Software virtual fitting room*.

Link aplikasi: [Virtual-Fitting-Room](#)

TABEL 3
Kelebihan dan kekurangan *Virtual Fitting Room*

Kelebihan Aplikasi	Kekurangan Aplikasi
Efisiensi waktu dan ruang	Ukuran baju yang terkadang tidak sesuai
Kemudahan penggunaan	Memerlukan <i>Hardware</i> khusus
Fleksibilitas Kustomisasi	-

4. Perbandingan Fitur

Setelah melakukan review, fitur-fitur dari aplikasi di atas dapat disajikan dalam Tabel 2.1. Tabel ini juga memuat rencana fitur yang akan dikembangkan di aplikasi proyek akhir ini.

TABEL 4
Perbandingan Fitur Aplikasi Serupa

No.	Fitur Aplikasi	HeroMirror	Foto Master	Virtual Fitting Room	Aplikasi PA
1	Fitur interaksi	✓	✓	✓	✓
2	Fitur Visual	✓	✓	✓	✓
3	Fitur <i>Output</i>	x	✓	✓	✓
No.	Fitur Aplikasi	HeroMirror	Foto Master	Virtual Fitting Room	Aplikasi PA
4	Fitur Kustomisasi	✓	✓	✓	✓

- a. **Fitur Interaksi**
Fitur yang mana pengguna bisa berinteraksi dengan elemen-elemen AR dalam sebuah *photobooth*.
- b. **Fitur Visual**
Fitur visual merujuk pada elemen-elemen grafis atau efek yang ditambahkan ke dunia nyata melalui teknologi AR, yang dapat dilihat melalui perangkat seperti kamera, ponsel, atau headset AR. Contohnya termasuk filter wajah, objek 3D, juga animasi yang muncul di layar dan berinteraksi dengan lingkungan nyata pengguna.
- c. **Fitur *Output***
Fitur *output* merujuk pada hasil atau produk yang dihasilkan dari interaksi pengguna dengan aplikasi AR. Dalam konteks AR *photobooth*, *output* bisa berupa foto yang sudah otomatis terunduh.
- d. **Fitur Kustomisasi**

Kustomisasi dalam konteks AR memungkinkan pengguna untuk menyesuaikan atau mengubah elemen-elemen AR sesuai dengan preferensi mereka.

III. ANALISIS KEBUTUHAN DAN PERANCANGAN

A. Analisis Kebutuhan Pengguna

Pada bagian ini dibahas mengenai analisis kebutuhan pengguna, proses perancangan aplikasi, serta spesifikasi perangkat keras dan perangkat lunak yang dibutuhkan dalam pengembangan aplikasi dan fitur *Augmented Reality* pada aplikasi JELITA.

1. Proses Menggali Informasi

Informasi kebutuhan pengguna dan karakteristiknya digali dengan metode wawancara. Wawancara dilaksanakan pada 18–19 Desember 2024 secara daring. Wawancara dilakukan terhadap tiga orang narasumber. Dokumentasi wawancara berupa foto-foto kegiatan dapat dilihat di Lampiran A.

Pertanyaan yang diajukan dalam wawancara disusun berdasarkan teori-teori yang telah ditinjau pada Bab II, aplikasi serupa yang telah di-review kelebihan dan kekurangannya, serta sumber lain yang relevan. Daftar pertanyaan yang diajukan dapat dilihat pada Tabel 3.1.

TABEL 5
Daftar Pertanyaan yang Diajukan

No.	Pertanyaan yang Diajukan	Narasumber
1	Berapa lama Bapak sudah berkecimpung di bidang <i>photobooth</i> ?	Engineer
2	Seberapa sering Bapak terlibat dalam proyek <i>photobooth</i> ? Dapatkah Bapak menyebutkan beberapa proyek yang pernah ditangani?	Engineer
3	Seberapa sering kebutuhan untuk <i>photobooth</i> muncul dalam acara-acara dalam sepekan?	Engineer
4	Apa saja jenis acara yang biasanya membutuhkan layanan <i>photobooth</i> ?	Engineer
5	Fitur apa saja yang biasanya ada di dalam sistem <i>photobooth</i> yang Bapak tawarkan?	Engineer
6	Data pengguna apa saja yang diperlukan saat proses registrasi untuk menggunakan layanan <i>photobooth</i> ?	Engineer
7	Pengalaman apa yang biasanya diharapkan oleh klien dari layanan <i>photobooth</i> ?	Engineer
8	Di akhir aplikasi, apakah ada kebutuhan tambahan yang biasanya diminta oleh klien?	Engineer

2. Karakteristik Target Pengguna

Aplikasi ini ditujukan untuk pengguna yang menginginkan pengalaman *photobooth* yang interaktif dan inovatif, khususnya dalam konteks acara atau event. Berdasarkan hasil wawancara yang dilakukan, *photobooth* terbukti menjadi salah satu fitur hiburan yang paling diminati oleh berbagai kalangan, karena mampu menciptakan momen yang menyenangkan dan dapat dibagikan di media sosial.

Dengan mengimplementasikan teknologi AR tanpa menggunakan kamera pemindai kedalaman, aplikasi ini menghadirkan solusi yang lebih terjangkau dan mudah diakses oleh perangkat umum, khususnya kamera web. Pendekatan ini tidak hanya menarik perhatian pengunjung, tetapi juga dirancang untuk meningkatkan pengalaman pengguna (*user experience*) melalui interaksi visual yang responsif, kemudahan penggunaan, dan elemen hiburan digital yang modern. Karakteristik pengguna aplikasi ini adalah individu yang mengapresiasi teknologi visual, ingin mendapatkan pengalaman unik dalam berfoto, serta mencari nilai tambah dari kehadirannya di suatu acara.

Selain itu, *photobooth* juga sering digunakan dalam acara seperti pernikahan, pameran, dan festival untuk memeriahkan suasana. Untuk mendukung operasional *photobooth*, diperlukan perangkat yang memadai, seperti Android TV dengan dukungan kamera eksternal (kamera web). Spesifikasi perangkat yang mendukung aplikasi *photobooth* dirangkum dalam Tabel 3.2 berikut ini:

TABEL 6
Spesifikasi Target Perangkat

Jenis	Spesifikasi Minimal
Perangkat keras	Android TV dengan prosesor minimal QuadCore, Ram 4GB dan penyimpanan internal 8GB
Perangkat lunak	Sistem operasi Android minimal versi 5.0 (Lollipop, API level 21)

3. Fitur yang Dibutuhkan

Berdasarkan informasi kebutuhan yang telah digali pada tabel 3.1, fitur aplikasi yang perlu dibangun sesuai kebutuhan pengguna dapat diuraikan sebagai berikut:

Fitur AR

- Bertujuan menciptakan pengalaman yang realistis sehingga elemen *Augmented Reality* tampak menyatu dengan lingkungan nyata.
- Teknologi AR akan terhubung dengan kamera eksternal (kamera web) yang diintegrasikan dengan aplikasi.

Fitur Pose Landmarker

- Berfungsi untuk mendeteksi landmark tubuh manusia pada gambar atau video.
- Memungkinkan identifikasi lokasi utama pada tubuh, analisa postur, dan pengkategorian gerakan.
- Terintegrasi dengan elemen *virtual* berupa pakaian, sehingga pakaian akan tampak seperti dikenakan oleh pengguna.

- Pakaian akan bergerak secara dinamis mengikuti gerakan pengguna, memberikan pengalaman yang lebih imersif.

Fitur pemilihan filter dan AR Interaktif

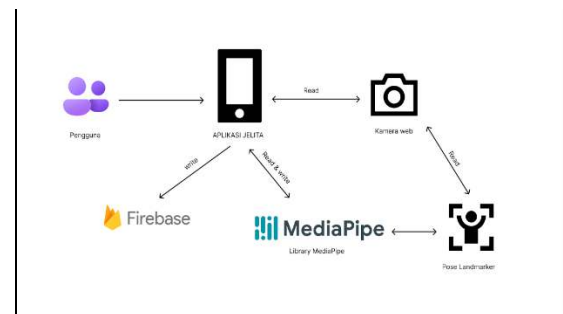
Memberikan pengalaman *photobooth* yang berbeda dengan *photobooth* konvensional yang mana menyajikan beberapa opsi filter yang dapat pengguna kenakan.

B. Perancangan Aplikasi

Setelah karakteristik target pengguna dipahami dan fitur-fitur yang dibutuhkan pengguna berhasil dirumuskan, aplikasi dirancang sebagai berikut.

1. Gambaran Umum Aplikasi

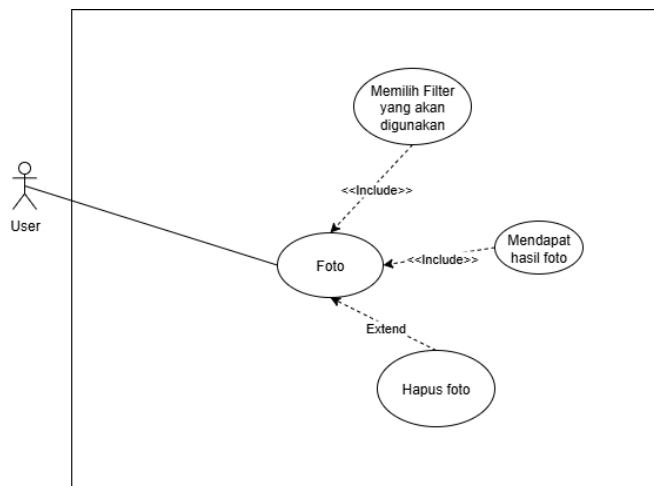
Aplikasi JELITA merupakan aplikasi mobile *photobooth* berbasis *Augmented Reality* (AR) yang dirancang untuk memberikan pengalaman interaktif kepada pengguna. Aplikasi ini dikembangkan menggunakan Android Studio dengan memanfaatkan library MediaPipe, yang mendukung pengolahan data visual secara efisien tanpa memerlukan perangkat keras kamera dengan kemampuan *depth sensing*. Visualisasi terkait arsitektur aplikasi JELITA dapat dilihat pada Gambar 3.1, yang menggambarkan struktur sistem secara menyeluruh.



GAMBAR 1
Arsitektur Aplikasi

Untuk mengakses aplikasi ini, pengguna cukup memasukkan nama dan email. Data tersebut akan disimpan secara terstruktur di *Firebase Realtime Database* untuk mendukung pengelolaan data yang efisien dan memastikan keamanan informasi pengguna. Aplikasi JELITA menghadirkan inovasi utama berupa penerapan filter AR interaktif yang dapat diintegrasikan pada *photobooth* tanpa memerlukan perangkat kamera pemindai kedalaman. Pendekatan ini memanfaatkan algoritma deteksi pose berbasis MediaPipe, yang mampu menghasilkan pengolahan visual secara akurat hanya dengan menggunakan kamera web standar. Berdasarkan analisis, solusi ini tidak hanya menurunkan biaya implementasi teknologi AR, tetapi juga meningkatkan aksesibilitas pengguna dalam memanfaatkan fitur *photobooth* berbasis AR. Implementasi ini memberikan kontribusi signifikan terhadap efisiensi sistem, sekaligus memperluas adopsi teknologi AR di berbagai platform dengan perangkat keras minimal.

2. Use case Diagram



GAMBAR 2
Use case Diagram

pengguna, yaitu individu yang menggunakan aplikasi untuk melakukan pengambilan foto dengan filter berbasis AR. Terdapat satu *use case* utama, yaitu Foto, yang menjadi pusat dari seluruh aktivitas pengguna. *Use case* ini mencerminkan proses utama ketika pengguna melakukan sesi foto dengan tampilan filter AR yang mengikuti pose tubuh secara waktu nyata (real time). Dari *use case* ini, terdapat beberapa relasi tambahan yang memperkaya proses pengambilan foto, yaitu relasi *<<include>>* dan *<<extend>>*.

a. <<include>> Memilih Filter yang Akan Digunakan

Relasi include menunjukkan bahwa aktivitas memilih filter AR merupakan bagian integral dari proses pengambilan foto. Artinya, setiap kali pengguna ingin mengambil foto, sistem akan selalu menyertakan langkah pemilihan filter terlebih dahulu. Filter dalam konteks ini merujuk pada aset AR berupa pakaian *virtual* yang ditampilkan secara waktu nyata melalui teknologi pelacakan pose (*pose tracking*).

b. <<include>> Mendapat Hasil Foto

Use case mendapat hasil foto juga direlasikan dengan include, karena *output* dari aktivitas “Foto” secara langsung menghasilkan foto akhir yang dapat ditampilkan kepada pengguna. Hasil ini merupakan keluaran visual yang menampilkan pengguna dengan efek filter AR yang telah dipilih sebelumnya.

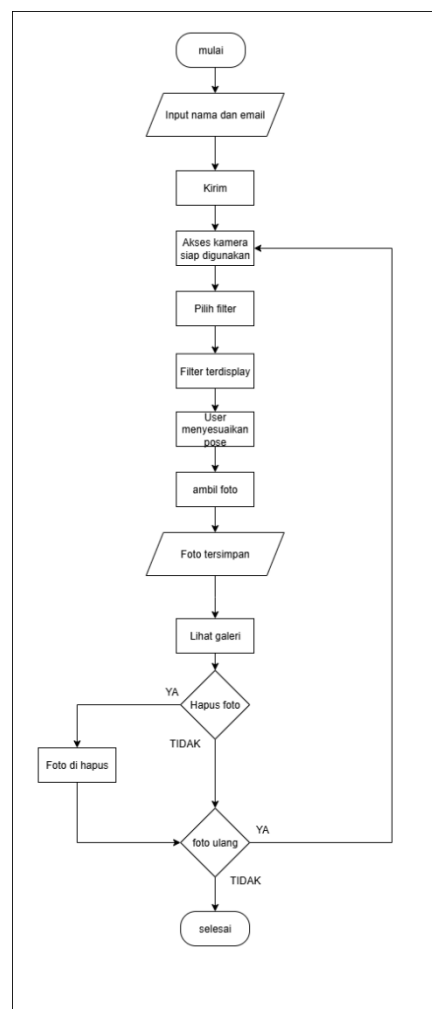
c. <<extend>> Hapus Foto

Use case hapus foto direlasikan dengan extend, yang berarti fitur ini bersifat opsional dan hanya dilakukan apabila diperlukan. Setelah pengguna memperoleh hasil foto, ia dapat memilih untuk menghapus foto tersebut apabila dirasa kurang memuaskan. Fitur ini memberikan fleksibilitas kepada pengguna untuk mengulangi proses pengambilan gambar.

Secara keseluruhan, *use case* diagram ini menunjukkan bahwa sistem dirancang dengan alur yang sederhana namun tetap fungsional, dengan fokus pada pengalaman pengguna dalam mengambil foto berbasis AR. Interaksi ini menekankan prinsip desain berpusat pada pengguna (*user-centered design*), di mana pengguna memiliki kendali penuh terhadap filter yang digunakan serta hasil akhir dari sesi fotonya.

3. Flow chart

Untuk menggambarkan tahapan proses yang terjadi di dalam aplikasi JELITA secara sistematis, maka disusunlah flowchart seperti pada Gambar 3.3.



GAMBAR 3
Flow chart Aplikasi JELITA

Proses alur flowchart dijabarkan lebih lanjut dalam bentuk uraian tabel. Setiap langkah dijelaskan secara sistematis guna menunjukkan bagaimana pengguna berinteraksi dengan sistem mulai dari awal hingga akhir penggunaan. Rincian proses tersebut dapat dilihat pada Tabel 3.3 berikut ini.

TABEL 7
Deskripsi proses *flow chart*

No.	Tahapan Proses	Deskripsi
1	Mulai	Proses diawali saat pengguna membuka aplikasi. Ditandai dengan simbol terminator sebagai titik awal.
2	Input Nama dan Email	Pengguna mengisi data berupa nama dan alamat surel untuk keperluan identifikasi atau distribusi hasil.

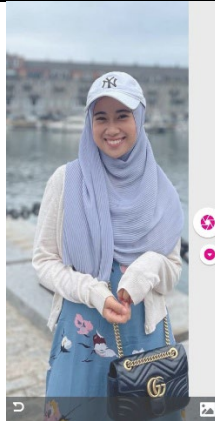
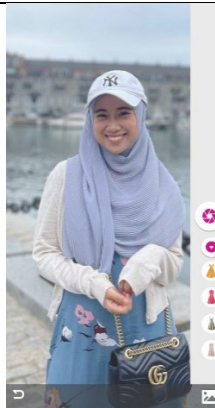
3	Kirim	Data dikirim ke sistem untuk diproses dan digunakan pada tahap berikutnya.
4	Akses Kamera Siap Digunakan	Sistem menginisialisasi kamera agar siap menampilkan visual pengguna secara <i>real-time</i> .
5	Pilih Filter	Pengguna memilih filter AR berupa pakaian <i>virtual</i> yang akan digunakan dalam proses pengambilan foto.
6	Filter Tertampil	Sistem menampilkan filter yang dipilih ke dalam antarmuka kamera menggunakan teknologi <i>pose tracking</i> .
7	Penyesuaian Pose oleh Pengguna	Pengguna menyesuaikan posisi tubuh agar pakaian <i>virtual</i> selaras secara visual.
8	Ambil Foto	Pengguna menekan tombol untuk menangkap gambar dengan filter AR yang aktif.
9	Foto Tersimpan	Foto hasil pengambilan disimpan secara otomatis ke dalam penyimpanan perangkat.
10	Lihat Galeri	Sistem menampilkan hasil foto dalam galeri internal aplikasi untuk ditinjau oleh pengguna.
11	Hapus Foto (Keputusan)	Pengguna menentukan apakah foto ingin dihapus atau tidak. Jika "Ya", foto dihapus dan proses diulang.
12	Foto Ulang (Keputusan)	Pengguna menentukan apakah foto ingin dihapus atau tidak. Jika "Ya", foto dihapus dan proses diulang.
13	Selesai	Proses interaksi dinyatakan selesai. Ditandai dengan simbol terminator pada akhir flowchart.

Flowchart di atas menggambarkan alur proses dari penggunaan aplikasi JELITA, sebuah *photobooth* berbasis teknologi *Augmented Reality* (AR) yang dikembangkan tanpa dukungan kamera pemindai kedalaman (*depth sensing*). Diagram ini disusun untuk memvisualisasikan tahapan interaksi pengguna dengan sistem secara menyeluruh, mulai dari inisialisasi hingga hasil akhir berupa foto yang telah disesuaikan dengan filter AR.

4. Perancangan Antarmuka Aplikasi

Antarmuka aplikasi yang dirancang dapat disajikan dalam Tabel 3.4. Rancangan ini dibuat dengan menggunakan prototyping tool berbasis web Figma. Setiap rancangan tampilan ini telah dicek kesesuaiannya dengan analisis kebutuhan pengguna di subbab 3.1.3 dan juga *use case diagram* yang ada di subbab 3.2.2.

TABEL 8
Rancangan Antarmuka Aplikasi

No.	Tampilan	Penjelasan
1		<i>Welcome Screen</i> Tampilan ini merupakan halaman <i>Welcome Screen</i> dari aplikasi JELITA, yang menyajikan nama dan slogan aplikasi tersebut. Selain itu, tersedia tombol untuk memulai (mulai) dan keluar dari aplikasi (keluar).
2		<i>Input Data Pengguna</i> Tampilan ini akan muncul setelah pengguna menekan tombol mulai pada halaman sebelumnya. Selanjutnya, pengguna diminta untuk mengisi email dan nama pada kolom yang tersedia, dan setelah itu tekan tombol kirim.
3		<i>Kamera Photobooth</i> Selanjutnya, pengguna akan langsung diarahkan ke halaman kamera, di mana pengguna dapat berfoto menggunakan filter AR. Untuk menggunakan efek AR, pengguna dapat menekan tombol dropdown di sebelah kanan.
4		<i>Efek Augmented Reality (AR)</i> Jika pengguna sudah menekan tombol dropdown maka pengguna bisa memilih opsi baju yang telah disediakan untuk di coba di <i>photobooth</i> .

5. Perancangan Basis Data

Untuk menunjang performa aplikasi JELITA dalam pengelolaan data pengguna, digunakan *Firestore Realtime Database* sebagai sistem manajemen basis data. Sistem ini berfungsi menyimpan data email dan nama pengguna yang mengakses aplikasi JELITA.

Pada sisi lain, pelacakan pose tubuh pengguna dilakukan dengan memanfaatkan *MediaPipe Pose Landmarker*, yang mampu mendeteksi titik-titik landmark tubuh secara *real-time* melalui tangkapan kamera. Data hasil deteksi ini kemudian digunakan untuk menyesuaikan posisi objek visual pakaian *virtual*, agar tampil selaras dengan tubuh pengguna.

Guna meningkatkan efisiensi proses pemuatan dan perenderan efek visual, seluruh aset gambar pakaian disimpan secara lokal di dalam direktori proyek Android Studio. Secara keseluruhan, struktur data *Firestore* dan sistem penyimpanan lokal bekerja secara terpadu. *Firestore* bertanggung jawab dalam pengelolaan data pengguna, sementara aset visual pakaian dimasukkan secara manual ke dalam direktori proyek di Android Studio.

C. Kebutuhan Pengembangan Aplikasi

Untuk mengimplementasikan aplikasi sesuai rancangan yang telah dibuat, dibutuhkan perangkat keras dan perangkat lunak berikut.

1. Kebutuhan Perangkat Keras

Perangkat keras yang dibutuhkan dapat dilihat pada Tabel 3.4. Seluruh perangkat yang tercantum telah tersedia sehingga tidak diperlukan pengeluaran biaya tambahan untuk pengadaan alat.

TABEL 9
Kebutuhan Perangkat Keras

No.	Spesifikasi Perangkat	Ketersediaan
1	Laptop Lenovo Legion slim 5	Tersedia, milik pribadi
2	Intel Core™ i7 dan RAM 16GB	Tersedia, milik pribadi
3	Smartphone Xiaomi X6 pro 5G	Tersedia, milik pribadi

2. Kebutuhan Perangkat Lunak

Perangkat lunak yang diperlukan tercantum pada Tabel 3.5. Seluruh perangkat lunak tersebut merupakan versi resmi berlisensi dan bukan merupakan perangkat lunak ilegal. Selain itu, seluruhnya dapat digunakan tanpa memerlukan biaya tambahan dalam proses pengembangan aplikasi

TABEL 10
Kebutuhan Perangkat Lunak

No.	Spesifikasi Perangkat	Ketersediaan
1	Figma	Open source
2	Firestore Realtime Database	Spark plan (free)
3	MediaPipe Pose Landmarker	Open source
4	Android Studio	Open source

Dalam proses pengembangan aplikasi JELITA, beberapa platform telah dipertimbangkan untuk membangun pengalaman *Augmented Reality*, seperti Unity dan Lens Studio. Berdasarkan eksplorasi awal, kedua platform tersebut belum sepenuhnya sesuai dengan kebutuhan sistem yang ingin dikembangkan. Pada Unity, belum ditemukan metode yang tepat untuk mengintegrasikan aset visual, seperti pakaian *virtual*, secara presisi ke dalam sistem, khususnya dalam hal penyesuaian posisi terhadap tubuh pengguna menggunakan *MediaPipe*. Selain itu, proses integrasi *MediaPipe* ke dalam Unity masih tergolong kompleks dan terbatas.

Di sisi lain, Lens Studio memiliki sejumlah keterbatasan, terutama dalam hal fleksibilitas pengembangan aplikasi secara mandiri, distribusi ke platform Android, serta pengelolaan data pengguna. Penggunaan kedua platform ini juga memerlukan pemahaman teknis lanjutan, sementara penguasaan terhadap alat-alat tersebut masih dalam tahap awal eksplorasi.

Berdasarkan pertimbangan tersebut, Android Studio dipilih sebagai platform utama dalam pengembangan aplikasi JELITA. Platform ini memberikan kontrol penuh terhadap alur kerja pengembangan, serta mendukung integrasi langsung dengan *MediaPipe Pose Landmarker*. Android Studio juga dilengkapi dokumentasi resmi dan contoh implementasi dari Google, yang sangat membantu selama proses pengembangan. Selain itu, latar belakang pembelajaran selama dua semester perkuliahan terkait Android Studio turut menjadi faktor pendukung dalam memilih platform ini, sehingga pengembangan aplikasi dapat dilakukan secara lebih optimal dan terarah.

IV. IMPLEMENTASI DAN PENGUJIAN

A. Implementasi Aplikasi

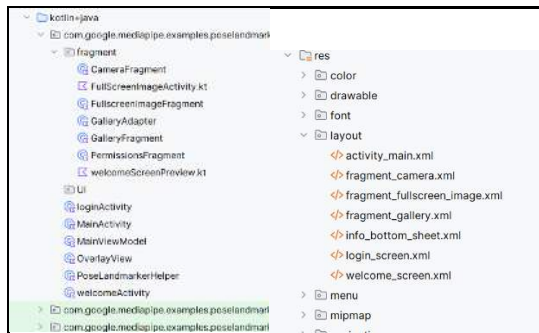
Pengujian aplikasi bertujuan untuk meningkatkan performa, mengidentifikasi bug, serta memastikan kesesuaian dengan kebutuhan pengguna. Proses ini sangat penting untuk menjamin bahwa aplikasi JELITA tidak hanya berfungsi dengan baik, tetapi juga memenuhi kebutuhan dan harapan pengguna. Selain itu, pengujian berperan sebagai indikator kelayakan pengguna aplikasi JELITA, yang membantu menentukan apakah aplikasi tersebut siap untuk diluncurkan ke publik.

Pada tahap perancangan awal, aplikasi JELITA dirancang agar dapat berjalan pada perangkat yang mendukung kamera web, seperti Android TV. Akan tetapi, selama proses pengembangan berlangsung, keterbatasan perangkat uji coba menjadi kendala utama yang menyebabkan implementasi hanya dilakukan menggunakan kamera *smartphone*. Kendala ini tidak berasal dari sisi teknis aplikasi, melainkan karena tidak tersedianya perangkat yang sesuai. Meski demikian, struktur sistem tetap disusun secara fleksibel agar dapat menyesuaikan dengan berbagai jenis *input* kamera, termasuk kamera eksternal, melalui pengaturan konfigurasi tambahan pada sistem.

1. Struktur Kode Project

Pengembangan aplikasi JELITA dilakukan menggunakan Android Studio dengan bahasa pemrograman Kotlin sebagai basis utama. Dalam proses pengembangan, aplikasi ini memanfaatkan plugin *MediaPipe* dari Google untuk melakukan *tracking* pose landmark pada tubuh secara *real-time*. *MediaPipe* dipilih karena kemampuannya dalam

menghasilkan respon yang cepat dan akurat tanpa menggunakan kamera sensor kedalaman. Struktur kode aplikasi menerapkan pemrograman fungsional, yang memungkinkan penyusunan kode melalui fungsifungsi terdefinisi dengan baik sehingga memudahkan debugging, pemeliharaan, dan optimasi performa. Struktur kode project ditunjukkan pada Gambar 4.1.



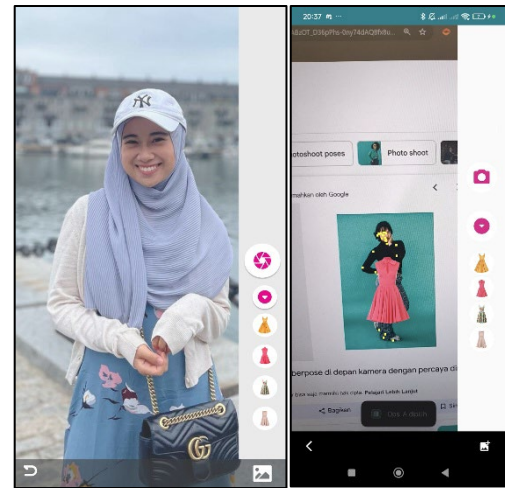
GAMBAR 4
Struktur Kode Project

Selain itu, implementasi Pose Landmarker didasarkan pada repository GitHub resmi Google yang dimodifikasi sesuai dengan rancangan pada Bab III, termasuk penyesuaian antarmuka pengguna (UI) agar lebih intuitif serta penambahan fitur pemilihan baju yang terintegrasi dengan arsitektur dan komponen sistem yang telah dirancang. Sesuai dengan best practice di industri, project ini juga telah menerapkan *version control system* (VCS) sehingga setiap perubahan pada kode akan tersimpan riwayatnya. VCS yang digunakan adalah git, dan project dihosting di Github untuk memudahkan kolaborasi dalam tim. Repository bersifat private dan dapat diakses di link berikut jika diperlukan: <https://github.com/PA-D3RPLA/d3if46-npr-jelita>

2. Kesesuaian Terhadap Rancangan

Proses perancangan aplikasi JELITA melibatkan kolaborasi antara berbagai pihak, termasuk dosen pembimbing, dosen reviewer, dan calon pengguna. Dalam hal ini, tidak ada perubahan signifikan pada aplikasi hasil implementasi. Semua fitur dan fungsionalitas tetap sesuai dengan rancangan yang dijelaskan dalam Bab III, meskipun terdapat beberapa perbedaan kecil dalam antarmuka pengguna. Perbedaan antara tampilan rancangan awal dan hasil implementasi aplikasi dapat dilihat pada Gambar 4.2.

Saat aplikasi dijalankan, pengguna akan diarahkan ke *Welcome Screen* yang menampilkan tombol "Mulai" dan "Keluar." Setelah memilih "Mulai," pengguna diminta mengisi nama dan email, kemudian diarahkan ke tampilan kamera untuk 31 menggunakan filter AR dan mengambil foto. Data pengguna disimpan secara realtime menggunakan *Firebase Firestore*.



GAMBAR 5
Perbedaan UI Rancangan dan Implementasi

Namun demikian, terdapat satu penyesuaian penting pada sisi perangkat yang digunakan dalam pengujian. Pada tahap perancangan, aplikasi JELITA direncanakan untuk dijalankan menggunakan kamera web, lalu disambungkan pada perangkat seperti Android TV. Akan tetapi, karena keterbatasan alat selama proses pengembangan, pengujian dilakukan menggunakan kamera *smartphone* yang tersedia. Kendala ini tidak berasal dari sistem aplikasi, melainkan disebabkan oleh keterbatasan perangkat keras yang dapat digunakan selama tahap implementasi. Ke depannya, penggunaan kamera web masih dimungkinkan, tetapi memerlukan penyesuaian lebih lanjut pada pengaturan sistem agar dapat mendukung integrasi dengan perangkat lain.

3. Hasil Implementasi

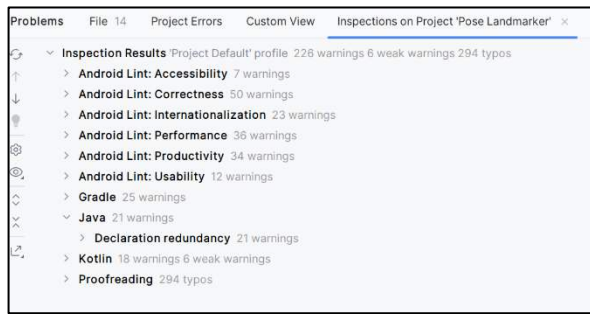
Hasil implementasi proyek akhir ini, yaitu aplikasi JELITA, tersedia dalam bentuk berkas APK. Untuk mempermudah pengguna dalam mengoperasikan aplikasi, tautan tersebut juga menyediakan buku panduan penggunaan, video demo aplikasi, serta video promosi:

<https://drive.google.com/drive/folders/1L9Gt9aFkeNgS75lz4BjcD7tpUr08xbKp?usp=sharing>

B. Pengujian Aplikasi

1. Pengujian Kualitas Kode

Pengujian kualitas kode project dilakukan dengan menggunakan tools *Inspect Code* yang disediakan oleh Android Studio. Berdasarkan hasil pengujian, ditemukan 266 warning dan enam weak warning, dan 294 typo yang tersebar pada beberapa kategori. Meskipun terdapat sejumlah peringatan, tidak ditemukan error kritis yang mengganggu fungsi utama aplikasi. Mayoritas adalah bersifat minor, yang tidak secara langsung memengaruhi kestabilan sistem. Hal ini menunjukkan bahwa secara keseluruhan, kode ditulis dengan struktur yang baik, dengan tetap mempertimbangkan hasil inspeksi sebagai bahan perbaikan. Visualisasi hasil inspeksi kode dapat dilihat pada Gambar 4.3.



GAMBAR 6
Hasil *Inspect Code*

2. Pengujian Fungsionalitas

Pengujian fungsionalitas aplikasi dilakukan dengan metode *white box*. Proses pengujian dimulai dengan pembuatan *test case* secara mendetail berdasarkan struktur internal dan logika setiap modul atau fitur aplikasi. Setelah *test case* disusun, pengujian dilakukan secara bertahap pada setiap modul atau fitur dengan menggunakan scenario test yang telah dirancang. Pengujian ini bertujuan untuk memastikan bahwa setiap bagian kode berjalan sesuai dengan alur dan kondisi yang diharapkan. Seluruh pengujian aplikasi dilakukan menggunakan *smartphone* Poco x6 pro 5g dengan sistem operasi Android 14 untuk mendapatkan hasil yang akurat dan relevan. Hasil dari proses pengujian tersebut dapat dilihat pada Tabel 4.1.

TABEL 11
Test case JELITA

Modul	Jenis Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Tanggal Pengujian	Hasil Akhir
Welcome Screen	Positif	Periksa apakah tombol 'Mulai' dari 'Mulai' terlihat di Welcome screen	Tombol 'Mulai' dan 'Kembali' ditampilkan dengan benar	09/05/2025	Muncul di layar Welcome screen
Welcome Screen	Luas	Verifikasi pengguna dapat menginput email dan nama, lalu menekan tombol 'Kirim' untuk melanjutkan	Pengguna diarahkan ke halaman berikutnya setelah input valid	09/05/2025	Berhasil diarahkan dan terdapat di analisis ke halaman selanjutnya
Welcome Screen	Negatif	Verifikasi sistem menolak saat nama atau email dimasukkan dan tombol 'Kirim' ditekan	Tampilan error muncul dan pengguna tidak dapat melanjutkan	09/05/2025	Notif error muncul
Kamera	Positif	Periksa apakah tombol 'Kamera' dapat menangkap foto saat ditekan	Foto berhasil diambil dan ditampilkan	09/05/2025	Dalam proses
Kamera	Positif	Verifikasi apakah pengguna dapat mengambil 4 foto AR dress saat ditekan	4 foto dress ditampilkan dalam aplikasi	09/05/2025	Berhasil dan tampil sesuai UI
Kamera	Filter	Periksa apakah filter AR dress yang dipilih muncul dengan benar di kamera	AR dress yang dipilih muncul dengan benar di kamera	09/05/2025	Berhasil dan tampil sesuai dengan yang dipilih
Navigasi	Positif	Verifikasi tombol 'Kembali' mengembalikan pengguna ke halaman sebelumnya	Pengguna kembali ke halaman sebelumnya	09/05/2025	Berhasil diarahkan ke halaman sebelumnya
Galeri	Positif	Periksa apakah tombol 'Galeri' menampilkan foto yang sudah diambil	Pengguna dapat melihat hasil foto di galeri	09/05/2025	Dalam proses

Berdasarkan serangkaian pengujian yang telah dilakukan, dapat disimpulkan bahwa seluruh fitur yang diimplementasikan dalam aplikasi ini telah berfungsi sebagaimana mestinya. Hasil yang diperoleh pun menunjukkan kesesuaian dengan tujuan dan ekspektasi awal

pengembangan, sehingga aplikasi dinilai telah memenuhi indikator keberhasilan yang ditetapkan.

3. Pengujian ke Pengguna

Pengujian pengguna dilakukan untuk memperoleh masukan langsung terhadap kualitas, kenyamanan, dan efektivitas fitur-fitur dalam aplikasi JELITA. Enam responden dipilih secara purposive berdasarkan keterlibatan mereka dalam bidang multimedia dan teknologi interaktif. Pengujian terhadap pengguna dalam penelitian ini dilakukan dengan pendekatan kualitatif deskriptif. Responden diberikan video demo aplikasi JELITA untuk ditinjau, kemudian diminta mengisi kuisioner daring berupa pertanyaan terbuka (esai) yang dirancang untuk menggali tanggapan mereka secara mendalam terhadap pengalaman penggunaan aplikasi.

Analisis terhadap hasil kuisioner dilakukan dengan metode tematik, yaitu mengelompokkan pola-pola tanggapan yang muncul menjadi beberapa tema utama. Sebagai pelengkap, data disajikan dalam bentuk tabulasi frekuensi sederhana, seperti jumlah atau persentase responden yang memberikan tanggapan serupa. Penyajian numerik ini tidak dimaksudkan untuk mengubah pendekatan penelitian menjadi kuantitatif, melainkan untuk memperjelas kecenderungan respon dan mendukung interpretasi hasil secara lebih transparan.

a. Ringkasan Hasil Kuantitatif

Agar hasil lebih terukur, analisis tematik dari tanggapan responden dikombinasikan dengan data kuantitatif seperti pada tabel 4.1 sebagai berikut:

TABEL 12
Hasil Kuantitatif

Aspek yang Dievaluasi	Aspek yang Dievaluasi
Aplikasi mudah digunakan	100% (6 dari 6 responden)
AR tanpa <i>depth sensing</i> dinilai cukup baik	50% menyatakan cukup, 50% beri masukan teknis
Tampilan UI sederhana & mudah dipahami	5 dari 6 responden
Layak digunakan di event nyata	5 dari 6 responden
Saran untuk pengembangan lebih lanjut	Tambah filter, dukungan web, dan peningkatan AR <i>tracking</i>

b. Ringkasan Tematik

1. Kemudahan Penggunaan

Seluruh responden menyatakan bahwa aplikasi JELITA mudah digunakan dan tidak memerlukan banyak instruksi. Hal ini menunjukkan bahwa desain interaksi pengguna telah sesuai dengan prinsip usability.

2. Fitur AR Tanpa *Depth sensing*

- Tiga dari enam responden menyatakan bahwa meskipun tidak menggunakan *depth sensing*, performa AR cukup baik untuk kebutuhan *photobooth*.
- Tiga lainnya menyampaikan bahwa *tracking* masih kurang presisi, seperti pose yang tidak

menyesuaikan orientasi tubuh dengan akurat. Meskipun demikian, semua responden menganggap pendekatan ini unik dan efisien secara biaya.

3. User Interface (UI)

Sebagian besar responden (83%) menilai UI JELITA sederhana, intuitif, dan enak dipandang. Namun, satu responden menyarankan agar tampilan lebih “berkesan” seperti *photobooth* pada umumnya.

4. Kelayakan Sebagai *Photobooth* Modern

Lima dari enam responden menyatakan bahwa JELITA layak digunakan sebagai alternatif *photobooth* modern, terutama karena aplikasi ini dapat berjalan di perangkat *smartphone* tanpa alat tambahan. Responden lainnya menyarankan penambahan fitur lanjutan agar bisa bersaing dengan platform sosial media yang sudah populer.

5. Saran Pengembangan

Masukan dari responden meliputi:

- Dukungan untuk tablet atau desktop karena *photobooth* umumnya menggunakan layar besar.
- Versi berbasis web untuk akses lebih luas.
- Peningkatan presisi pose *tracking*, khususnya orientasi tubuh.
- Opsi unggah filter pakaian sendiri.
- Integrasi visual AR dan objek foto yang lebih halus.

c. Kesimpulan Pengujian

Hasil pengujian pengguna menunjukkan bahwa aplikasi JELITA mendapat tanggapan positif secara umum. Aplikasi dinilai:

- Mudah digunakan, bahkan oleh pengguna awam. Memberikan pengalaman *photobooth* yang menarik dan fungsional tanpa memerlukan kamera pemindai kedalaman.
- Memiliki potensi untuk diterapkan secara luas, terutama di acara publik dengan anggaran terbatas.

Meskipun terdapat beberapa masukan mengenai akurasi *tracking* dan fleksibilitas platform, keseluruhan hasil menunjukkan bahwa JELITA telah memenuhi ekspektasi dasar sebagai solusi *photobooth* AR berbasis perangkat standar. Saran dari responden menjadi referensi penting dalam pengembangan versi lanjutan. Dokumentasi lengkap tersedia pada Lampiran B.

4. Diskusi Hasil Pengujian

Bagian ini membahas interpretasi hasil pengujian yang telah dilakukan, baik dari sisi teknis maupun pengalaman pengguna, serta mengaitkannya dengan rumusan masalah dan tujuan proyek akhir sebagaimana dijelaskan dalam Bab I.

Tujuan utama dari proyek ini adalah menciptakan solusi *photobooth* berbasis *Augmented Reality* (AR) yang tidak memerlukan kamera pemindai kedalaman, melalui pengembangan aplikasi JELITA. Solusi ini diharapkan dapat menyederhanakan kompleksitas perangkat, mengurangi biaya

operasional, serta memberikan pengalaman pengguna yang optimal.

a. Pengujian Kualitas Kode

Pengujian kualitas kode dilakukan menggunakan fitur *Inspect Code* pada Android Studio. Hasilnya menunjukkan tidak terdapat error atau warning kritis dalam struktur program. Hal ini menandakan bahwa kode ditulis mengikuti standar praktik yang baik dan mendukung stabilitas sistem. Dengan kualitas kode yang bersih dan terstruktur, potensi gangguan teknis pada saat aplikasi dijalankan dapat diminimalkan.

b. Pengujian Fungsionalitas

Uji fungsional dilakukan dengan metode *white-box* testing terhadap seluruh fitur utama aplikasi, meliputi:

- Navigasi antar tampilan (*Welcome Screen*, input data pengguna, dan kamera).
- Input nama dan email pengguna.
- Pemilihan dan penerapan filter AR menggunakan MediaPipe.
- Pengambilan gambar dan penyimpanan foto ke galeri.

Hasil pengujian menunjukkan bahwa seluruh fitur utama dapat dijalankan sesuai dengan ekspektasi tanpa kendala. Aplikasi JELITA terbukti mampu berjalan pada perangkat dengan kamera standar (*smartphone*), tanpa memerlukan kamera pemindai kedalaman. Hal ini mendukung tujuan pengembangan yaitu menyediakan solusi *photobooth* AR yang fleksibel dan hemat biaya, tanpa mengurangi fungsionalitas inti.

c. Pengujian Kepada Pengguna

Pengujian terhadap pengguna dilakukan secara kualitatif terhadap enam responden yang memiliki latar belakang di bidang multimedia dan teknologi 3D visual. Untuk memperjelas hasil, data disajikan dalam bentuk tematik dan tabulasi frekuensi sederhana.

Kemudahan Penggunaan: Seluruh responden (100%) menyatakan aplikasi mudah digunakan dan tidak membutuhkan kurva belajar yang tinggi. Ini menunjukkan bahwa desain interaksi aplikasi telah memenuhi prinsip usability.

Performa AR tanpa *Depth sensing*: Sebanyak 50% responden menilai performa AR sudah cukup baik meskipun tidak menggunakan sensor kedalaman. Sisanya memberikan masukan terkait peningkatan akurasi *tracking*, namun tetap menganggap pendekatan ini unik dan efisien.

Antarmuka Pengguna (UI): Sebanyak 83% responden menilai UI aplikasi sederhana, ramah pengguna, dan mudah dipahami. Satu responden menyarankan agar tampilan dibuat lebih menarik secara visual agar sesuai dengan ekspektasi *photobooth*.

Kelayakan Sebagai Aplikasi *Photobooth*: Sebanyak lima dari enam responden menyatakan bahwa JELITA layak digunakan sebagai solusi *photobooth* modern berbasis perangkat standar. Mereka juga menilai pendekatan ini sesuai

dengan tren efisiensi biaya dan kemudahan akses pada berbagai event publik.

Hasil ini menunjukkan bahwa aplikasi JELITA telah menjawab rumusan masalah yang diajukan, yaitu menciptakan *photobooth* interaktif berbasis AR tanpa kebutuhan perangkat keras khusus, dan mampu memberikan pengalaman yang menyenangkan bagi pengguna.

Perlu disampaikan bahwa seluruh pengujian yang dilakukan, baik dari sisi fungsionalitas maupun pengujian kepada pengguna, dilaksanakan menggunakan kamera *smartphone*. Hal ini merupakan bentuk penyesuaian dari rencana awal yang semula menargetkan penggunaan kamera web (seperti pada perangkat Android TV). Keterbatasan perangkat yang tersedia saat implementasi menjadi alasan utama dilakukannya pengujian alternatif ini.

Meskipun demikian, hasil pengujian tetap relevan dan dapat dijadikan dasar evaluasi karena fitur-fitur utama aplikasi mampu berjalan dengan stabil, serta memperoleh respons positif dari pengguna. Pendekatan ini juga sekaligus membuktikan bahwa aplikasi JELITA cukup fleksibel untuk dijalankan di perangkat dengan spesifikasi umum dan tidak bergantung pada hardware khusus.

d. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan proses perancangan, pengembangan, dan pengujian yang telah dilaksanakan, dapat disimpulkan bahwa aplikasi JELITA berhasil dikembangkan sebagai solusi *photobooth* berbasis *Augmented Reality* (AR) tanpa menggunakan kamera pemindai kedalaman. Pendekatan ini dirancang untuk memberikan alternatif yang lebih sederhana dan hemat biaya dibandingkan dengan sistem *photobooth* AR konvensional yang umumnya membutuhkan perangkat keras tambahan.

Seluruh proses pengujian dilakukan menggunakan kamera bawaan *smartphone*, mengingat keterbatasan perangkat pengujian yang tersedia. Kendati demikian, secara teknis aplikasi tetap memungkinkan untuk diadaptasi pada perangkat lain seperti kamera web, termasuk yang terintegrasi dalam Android TV, melalui modifikasi sistem lebih lanjut.

Pemanfaatan MediaPipe sebagai teknologi ini memungkinkan pelacakan wajah dan tubuh secara waktu nyata dengan performa yang stabil. Hasil pengujian menunjukkan bahwa seluruh fitur utama, seperti pengambilan gambar, pemilihan filter AR, dan penyimpanan hasil foto, berjalan sesuai dengan rancangan. Berdasarkan evaluasi terhadap pengalaman pengguna, aplikasi ini dinilai cukup mudah digunakan dan mampu menghadirkan interaksi yang menarik serta responsif.

Selain itu, hasil implementasi menunjukkan bahwa sistem memiliki efisiensi yang signifikan dalam aspek penggunaan perangkat. Seluruh fungsi dapat dijalankan secara optimal tanpa ketergantungan pada kamera DSLR atau sensor kedalaman. Hal ini membuktikan bahwa jumlah dan kompleksitas perangkat yang diperlukan dapat diminimalkan, tanpa mengorbankan fungsionalitas utama. Dengan demikian, tujuan dari pengembangan aplikasi JELITA sebagai prototipe *photobooth* berbasis AR yang fungsional, sederhana, dan efisien dari sisi perangkat keras, dapat dikatakan telah tercapai.

B. Saran

Berdasarkan hasil pengujian serta umpan balik dari pengguna, berikut beberapa saran yang dapat dipertimbangkan dalam pengembangan lanjutan aplikasi JELITA:

1. Peningkatan akurasi pelacakan AR

Disarankan untuk menyempurnakan fitur pelacakan agar lebih presisi, terutama dalam mengikuti orientasi tubuh dan ekspresi wajah pengguna secara *real-time* guna meningkatkan realisme visual.

2. Penyempurnaan antarmuka pengguna (UI)

Perlu dilakukan peningkatan dari sisi tampilan agar lebih sesuai dengan suasana *photobooth*, misalnya melalui penambahan elemen dekoratif, transisi yang halus, atau animasi pendukung.

3. Penambahan variasi filter AR

Aplikasi dapat ditingkatkan dengan menyediakan lebih banyak pilihan filter digital, termasuk opsi kostumisasi atau unggah filter pribadi oleh pengguna untuk meningkatkan keterlibatan.

4. Dukungan lintas perangkat

Perluasan dukungan terhadap berbagai perangkat seperti tablet, desktop, atau pengembangan versi berbasis web akan membuka peluang pemanfaatan aplikasi di berbagai jenis event berskala besar.

5. Pendekatan edukatif bagi pengguna awam

Mengingat tidak semua pengguna familiar dengan istilah teknis seperti *depth sensing*, disarankan agar aplikasi menyediakan penjelasan singkat atau *tooltips* yang membantu pemahaman terhadap teknologi yang digunakan.

6. Optimisasi performa pada berbagai jenis kamera

Dikarenakan keterbatasan penggunaan kamera web standar, pengujian lebih lanjut terhadap berbagai jenis kamera *smartphone* dan kondisi pencahayaan diperlukan agar aplikasi dapat bekerja lebih konsisten di berbagai situasi.

REFERENCES

- [1] S. Balandin, I. Oliver, S. Boldyrev, A. Smirnov, N. Shilov, and A. Kashevnik, "Multimedia services on top of M3 Smart Spaces," Proc. - 2010 IEEE Reg. 8 Int. Conf. Comput. Technol. Electr. Electron. Eng. Sib., vol. 13, no. 2, pp. 728–732, 2010.
- [2] G. Dan Mulut Pada Anak, "Penggunaan Teknologi *Augmented Reality* Sebagai Media Pembelajaran Dalam Meningkatkan Kesehatan," vol. 6, pp. 611–619, 2023.
- [3] O. A. Egaji, I. Asghar, M. G. Griffiths, and D. Hinton, "An *Augmented Reality*-based system for improving quality of services operations: a study of educational institutes," TQM J., vol. 34, no. 2, pp. 330–354, 2022.
- [4] Google Developers, "Android Studio." [Online]. Available: <https://developer.android.com/studio>. [Accessed: 14-Dec-2024].
- [5] Gradle, "Gradle User Manual." [Online]. Available: <https://docs.gradle.org/current/userguide/userguide.html>. [Accessed: 14-Dec-2024].
- [6] Google AI for Developer, "MediaPipe Setup for Android." [Online]. Available: https://ai.google.dev/edge/mediapipe/solutions/setup_android. [Accessed: 14-Dec-2024].

- [7] [7] Google AI for Developer, "MediaPipe Tasks Vision." [Online]. Available: <https://ai.google.dev/edge/mediapipe/tasks/vision>. [Accessed: 14-Dec-2024].
- [8] [8] Google AI for Developer, "Using Pre-Built Libraries in MediaPipe." [Online]. Available: https://ai.google.dev/edge/mediapipe/framework/getting_started/android_archive_library. [Accessed: 14-Dec-2024].
- [9] [9] Google AI for Developer, "MediaPipe Assets Management." [Online]. Available: https://ai.google.dev/edge/mediapipe/solutions/assets_management. [Accessed: 14-Dec-2024].
- [10] [10] Google AI for Developer, "MediaPipe Face Detection API." [Online]. Available: https://ai.google.dev/edge/mediapipe/tasks/vision/face_detection. [Accessed: 14-Dec-2024].
- [11] [11] K. Febry Ardika et al., "Sistemasi: Jurnal Sistem Informasi Analisis Sentimen pada Aplikasi Android menggunakan MediaPipe untuk Klasifikasi Teks Sentiment Analysis on Android Applications using MediaPipe for Text Classification," vol. 14, no. 1, pp. 2540–9719, 2025.
- [12] [12] JetBrains, "Kotlin Programming Language." [Online]. Available: <https://kotlinlang.org/>. [Accessed: 14-Dec-2024].
- [13] [13] V. Polara, C. Mahant, and N. B. Dalwadi, "Kotlin: The new Android programming language," Int. J. Sci. Res. Dev., vol. 6, no. 1, 2018.
- [14] [14] JetBrains, "Null safety." [Online]. Available: <https://kotlinlang.org/docs/null-safety.html#what-s-next>. [Accessed: 14-Dec-2024].
- [15] [15] J. J. Sanjaya and J. Susilo, "Perbandingan Performa Kotlin vs Java dalam Pengembangan Android dengan Metode Iterasi While," bit-Tech, vol. 7, no. 2, pp. 545–553, 2024.
- [16] [16] I. P. Sari, I. H. Batubara, A. H. Hazidar, and M. Basri, "Pengenalan Bangun Ruang Menggunakan *Augmented Reality* sebagai Media Pembelajaran," Hello World J. Ilmu Komput., vol. 1, no. 4, pp. 209–215, 2022.
- [17] [17] D. A. Romadlon and A. S. Aji, "PENGUNAAN TEKNOLOGI *AUGMENTED REALITY* SEBAGAI MEDIA PENGENALAN JENIS MOBIL PADA SHOWROOM," J. TEKNOM, vol. 6, no. 2, 2023.
- [18] [18] Viso.ai, "MediaPipe: A Guide Google's *Open source* Framework." [Online]. Available: <https://viso.ai/computer-vision/mediapipe/>. [Accessed: 14-Dec-2024].
- [19] [19] Moravio, "Moravio on Implementing the MediaPipe." [Online]. Available: <https://www.moravio.com/blog/moravio-on-implementing-the-mediapipe%0A>. [Accessed: 14-Dec-2024].
- [20] [20] S. T. Nursaputro, D. Pramadihanto, and ..., "Rekonstruksi 3d untuk pemetaan ruangan Menggunakan Kamera Stereo," JTET (Jurnal Tek. ..., 2021.
- [21] [21] M. Fuad, "Estimasi Jarak Menggunakan Sensor Kinect," J. Ilm. Miktrotek, vol. 1, no. 1, pp. 5–10, 2013.
- [22] [22] F. M. Zulkarnaen, "Sistem Deteksi Objek Manusia Menggunakan Algoritma YOLOv8 Berbasis Kamera Depth Sensor (Studi Kasus: Cv. Ateri Global Teknologi)," Fak. Tek., p. 1, 2024.
- [23] [23] A. Setyawan and Norhikmah, "Aplikasi *Augmented Reality* Pada Photo Booth Di Kagungan Dalem Wahanarata Karaton Ngayogyakarta Hadiningrat," J. ICT Inf. Commun. Technol., vol. 24, pp. 139–145, 2024.
- [24] [24] F. Immanuel and A. P. Widodo, "Pengembangan Aplikasi *Photobooth* Berbasis *Augmented Reality*," J. Masy. Inform., vol. 11, no. 1, pp. 22–34, 2020.
- [25] [25] M. Leeson, J. Purnama, and M. Galinium, "Bildbox! – Virtual Photo Booth that implements *Augmented Reality*," J. Appl. Information, Commun. Technol., vol. 7, no. 1, pp. 27–43, 2021.