

BAB I

PENDAHULUAN

A. Latar Belakang

PAUD (Pendidikan Anak Usia Dini) merupakan upaya pembinaan anak semenjak lahir hingga usia enam tahun melalui pemberian rangsangan pendidikan agar memiliki kesiapan secara jasmani dan rohani saat memasuki pendidikan lebih lanjut. Pemberian rangsangan dengan media pembelajaran yang tepat akan mempengaruhi proses pertumbuhan dan perkembangannya, oleh sebab itu rangsangan perlu dilakukan sedini mungkin seperti pada Masa keemasan (*Golden Age*), *golden age* anak-anak berlangsung dari mulai lahir sampai usia 6 tahun. Media pembelajaran ini merupakan Sarana komunikasi dalam bentuk cetak maupun *audio* visual (Abidin and Haq 2023)

Pendekatan tradisional dalam pengajaran tentang buah dan sayuran sering kali terbatas pada gambar, buku, atau penjelasan *verbal* yang mungkin kurang menarik dan sulit dipahami oleh anak-anak usia dini. Keterbatasan ini dapat menghambat efektivitas pembelajaran dan mengurangi minat anak dalam mempelajari topik tersebut.

Dalam era digital saat ini, teknologi *augmented reality* (AR) menawarkan potensi besar untuk meningkatkan pengalaman belajar dengan menyediakan interaksi yang lebih dinamis dan *imersif*. AR menggabungkan elemen digital dengan dunia nyata, memungkinkan anak-anak untuk berinteraksi dengan *objek virtual* yang terlihat seolah-olah berada di dunia fisik mereka. *Teknologi* ini dapat

memberikan pengalaman belajar yang lebih menarik dan mendalam, terutama dalam *konteks* pendidikan anak usia dini.

Penggunaan *AR* dalam pendidikan PAUD belum sepenuhnya dimanfaatkan, dan terdapat peluang untuk merancang media pembelajaran yang dapat membuat pengenalan buah dan sayuran menjadi lebih interaktif dan menyenangkan. Media pembelajaran berbasis *AR* dapat menyediakan *visualisasi* 3D dari berbagai jenis tumbuhan, memungkinkan anak-anak untuk mengeksplorasi fitur-fitur buah dan sayuran secara langsung, serta memberikan informasi tambahan melalui elemen *interaktif*.

Dengan demikian Skripsi penelitian yang berjudul “**Perancangan Media Pembelajaran Pengenalan Manfaat Pada Buah-Buahan dan Sayuran Berbasis *Augmented Reality* Untuk Anak Usia Dini**”, bertujuan untuk merancang media pembelajaran berbasis *augmented reality* yang fokus pada pengenalan buah dan sayuran untuk anak usia dini. Diharapkan, media pembelajaran ini tidak hanya akan meningkatkan pemahaman dan minat anak-anak terhadap tumbuhan, tetapi juga memberikan alternatif inovatif untuk metode pengajaran *konvensional* yang ada saat ini.

B. Rumusan Masalah

Adapun rumusan masalah yang dapat dirumuskan dari latar belakang diatas ialah Bagaimana merancang media pembelajaran interaktif yang mampu mengenalkan Manfaat pada buah-buahan dan sayuran secara efektif kepada anak usia dini menggunakan teknologi *Augmented Reality*?

C. Tujuan Penelitian

Penelitian ini bertujuan untuk merancang media pembelajaran interaktif yang mampu mengenalkan Manfaat buah-buahan dan sayuran secara efektif kepada anak usia dini menggunakan teknologi *Augmented Reality*?

D. Batasan Masalah

Berdasarkan masalah yang diteliti ,peneliti juga membatasi beberapa hal dalam penelitiannya ini diantaranya ;

1. Penelitian ini akan fokus pada pengenalan sejumlah jenis buah-buahan dan sayuran yang umum dan mudah ditemukan di lingkungan sekitar anak usia dini.
2. Media pembelajaran berbasis *augmented reality* yang dirancang akan diuji dan digunakan oleh anak usia dini dalam rentang usia 6 hingga 9 tahun.
3. Desain media pembelajaran akan fokus pada elemen visual dan interaktif yang sederhana, seperti visualisasi 3D buah-buahan dan sayuran, kemudian informasi tambahan yang disajikan secara *intuitif*.

E. Manfaat Penelitian

Berdasarkan permasalahan diatas, Manfaat dari penelitian ini dapat dibagi menjadi 3(tiga) bagian yakni:

1. Bagi Penulis

Adapun manfaat bagi penulis adalah dapat mengetahui cara implementasi media pembelajaran Pengenalan Manfaat buah-buahan dan sayuran yang dilakukan untuk anak usia dini.

2. Bagi Pembaca

Manfaat bagi pembaca yaitu penelitian ini menawarkan Pembaca mendapatkan wawasan tentang manfaat dan penerapan teknologi AR dalam pendidikan anak usia dini, serta bagaimana teknologi ini dapat meningkatkan metode pembelajaran termasuk pendidik dan praktisi pendidikan, dan dapat menggunakan temuan dari rekomendasi penelitian ini sebagai referensi untuk mengimplementasikan atau mengembangkan media pembelajaran berbasis AR dalam lingkungan mereka. Penelitian ini dapat menginspirasi pembaca untuk melakukan studi lebih lanjut tentang penggunaan teknologi AR di berbagai konteks pendidikan dan disiplin ilmu lainnya.

3. Bagi Masyarakat

Manfaat bagi masyarakat yaitu penelitian ini dapat menjadi solusi terbaik dalam mempermudah pendidik untuk memberikan pemahaman kepada anak usia dini dan orang tua untuk mengajar melalui media pembelajaran interaktif yang sudah dibuat.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

1. Media Pembelajaran

Media Pembelajaran merupakan suatu kegiatan yang melibatkan seseorang dalam upaya memperoleh pengetahuan, keterampilan dan nilai-nilai positif dengan memanfaatkan berbagai sumber untuk belajar. Pembelajaran dapat melibatkan dua pihak yaitu siswa sebagai pembelajar dan guru sebagai fasilitator, yang terpenting dalam kegiatan pembelajaran adalah terjadinya proses belajar (*learning process*)(Rohani 2020).

Media pembelajaran juga dijadikan sebagai alat, metode, atau bahan yang digunakan oleh pengajar untuk membantu dalam proses transfer pengetahuan kepada siswa. Media ini berfungsi sebagai perantara yang menjembatani antara informasi yang ingin disampaikan oleh pengajar dan pemahaman yang diperoleh oleh siswa. Dengan menggunakan media pembelajaran, materi yang kompleks atau abstrak dapat disajikan secara lebih menarik, interaktif, dan mudah dipahami.

2. Buah dan Sayuran

Pengetahuan Buah dan Sayuran pada anak adalah langkah penting dalam mendukung perkembangan kesehatan fisik dan mental anak. Mengajarkan anak-anak tentang buah dan sayuran sejak usia dini membantu mereka mengembangkan kebiasaan makan sehat yang dapat bertahan seumur hidup. Pengenalan buah dan sayuran pada anak usia dini merupakan langkah penting dalam membentuk pola makan sehat yang akan membantu pertumbuhan dan perkembangan mereka.

Dengan menggunakan metode yang kreatif dan interaktif, orang tua dan pendidik dapat membuat proses belajar ini menjadi menyenangkan dan efektif. Pada akhirnya, anak-anak yang terbiasa dengan buah dan sayuran sejak dini akan memiliki pondasi yang kuat untuk menjalani kehidupan yang lebih sehat dan berkualitas (Lathifuddin, et al., 2019).

Manfaat pada buah dan sayuran sangat penting untuk anak usia dini karena mendukung pertumbuhan dan perkembangan mereka yang pesat. Berikut penjelasan mengenai beberapa nutrisi penting yang terdapat pada buah dan sayuran serta manfaatnya bagi anak usia dini:

- a. *Vitamin C* , Banyak Terdapat dalam buah-buahan seperti jeruk, stroberi, dan kiwi, serta sayuran seperti brokoli dan paprika. *Vitamin C* membantu meningkatkan sistem kekebalan tubuh, mempercepat penyembuhan luka, dan meningkatkan penyerapan *zat besi* dari makanan.
- b. *Serat* , Banyak terdapat pada sayuran berdaun hijau, wortel, apel, dan pisang. Serat sangat penting untuk kesehatan pencernaan, membantu mencegah sembelit, dan menjaga kesehatan usus anak.
- c. *Vitamin A* , banyak terdapat pada sayuran seperti wortel, bayam, dan ubi jalar, serta buah-buahan seperti mangga. Vitamin A penting untuk kesehatan mata, pertumbuhan tulang, dan mendukung sistem kekebalan tubuh.
- d. *Folat (Vitamin B9)*, Terdapat dalam sayuran hijau seperti bayam, brokoli, dan *asparagus*. *Folat* penting untuk perkembangan otak dan tulang belakang, serta berperan dalam produksi *sel* darah merah.

- e. *Kalium* , Banyak terdapat pada pisang, kentang, tomat, dan bayam. *Kalium* penting untuk menjaga tekanan darah yang sehat, fungsi otot, dan keseimbangan cairan dalam tubuh.
- f. *Zat Besi* , Terdapat dalam sayuran berdaun hijau seperti bayam dan kangkung, serta dalam beberapa buah-buahan kering. *Zat besi* penting untuk pembentukan *hemoglobin*, yang mengangkut *oksigen* dalam darah, serta mendukung perkembangan *kognitif*.

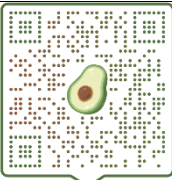
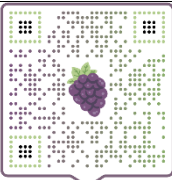
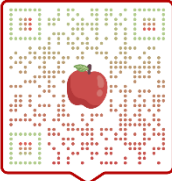
Adapun Gambar Buah yang kemudian akan dilakukan d *Scan* Buah dan Sayuran Menggunakan *Scan Qr code* dan *Scan* gambar :

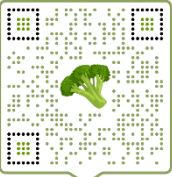
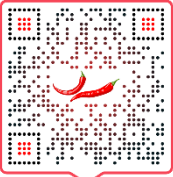
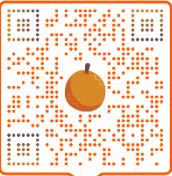
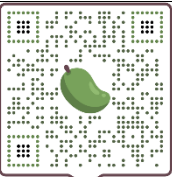
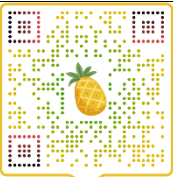
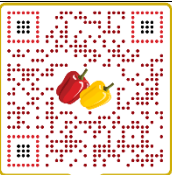
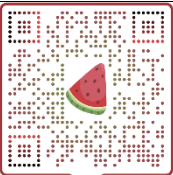
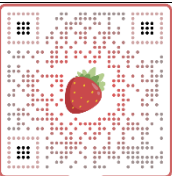
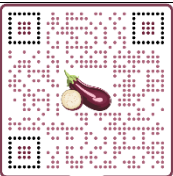
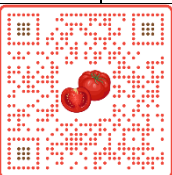
Tabel 2. 1 *Marker* Gambar

	 ALPUKAT			 ANGGUR	
	 APEL			 PAWANG PUTIH	
	 PROKOLI			 CAPAI	
	 JERUK			 LEMON	

	 MANGGA		 NANAS	
	 PAPRIKA		 SEMANGKA	
	 STRAWBERRY		 TERONG	
 TOMAT				

Tabel 2. 2 *Marker Qr Code*

 ALPUKAT	 ANGUR
 APEL	 BAWANG PUTIH

 BROKOLI	 CABAI
 JERUK	 LEMON
 MANGGA	 NANAS
 PAPRIKA	 SEMANGKA
 STRAWBERRY	 TERONG
 TOMAT	

3. *Android*

a. Pengertian *android*

Android adalah sistem operasi berbasis *linux* yang digunakan oleh telepon seluler (*mobile*) seperti telepon pintar (*smartphone*) dan komputer tablet (Yulianto and Mulyani 2019).

Android merupakan sebuah sistem operasi perangkat *mobile* berbasis *Linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. Untuk mengembangkan *Android*, dibentuk Open handset Aliance, yaitu konsorsium dari 34 perusahaan perangkat keras perangkat lunak dan Telekomunikasi termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, *Nvidia* (Menrisal and Utami 2019)

Android disebut *platform* terbuka pertama untuk perangkat *mobile*, *Android* ini sebuah lingkungan perangkat lunak yang dibangun untuk perangkat berbasis *mobile*. *Android* termasuk *kernel* berbasis *Linux*, aplikasi *enduser*, dan *framework* aplikasi. *User application* dibangun berbasiskan bahasa pemrograman *Java*. Bahkan aplikasi yang dibangun juga berbasiskan *Java*. Beberapa komponen yang ada dalam *Android*: Kapabilitas konektivitas seperti *WiFi* yang tersedia. Kode-kode *libraries* yang meliputi : *teknologi browser* contohnya *WebKit*. Dukungan database *SQLite*.

b. Kelebihan dan kekurangan *android*

Berikut kelebihan dari aplikasi *android* :

- 1) *User Friendly*, yang dimaksudkan disini adalah sistem *Android* sangat mudah untuk dijalankan. Sama halnya pada sistem operasi *Windows*

yang ada pada komputer. Bagi 9 pengguna yang belum terlalu terbiasa dengan *smartphone*, anda hanya membutuhkan waktu sebentar saja untuk bisa mempelajari sistem *Android*.

- 2) Pengguna akan sangat mudah mendapat beragam *notifikasi* dari *smartphone*. Untuk mendapatkannya, pengguna bisa mengatur beberapa akun yang anda miliki seperti *SMS*, *Email*, *Voice Dial*, dan lainnya.
- 3) Keunggulan lainnya terdapat dari segi tampilan sistem *Android* yang menarik dan tidak kalah baiknya dengan *iOS* (Apple). Hal ini dikarenakan dari awal, *Android* memang mengusung konsep dan teknologi *iOS* hanya saja *Android* merupakan *versi* murah dari *iOS*.
- 4) Sistem operasi ini memang memiliki konsep *open source* yang mana pengguna dapat bebas mengembangkan sistem *android* versi miliknya sendiri. Sehingga akan banyak sekali *Custom ROM* yang bisa anda gunakan.
- 5) Tersedia beragam pilihan aplikasi menarik, bahkan hingga jutaan aplikasi. Dari mulai aplikasi gratis hingga aplikasi berbayar. Anda dapat mendownloadnya langsung di *Google Playstore* yang tersedia pada *smartphone* anda.

Berikut kekurangan *android* :

- 1) *Update system* yang kurang efektif. Sistem *Android* memang sering kali mengalami peningkatan *versi* yang ditawarkan kepada penggunanya. Namun untuk mengupdate *system Android* ini bukanlah hal yang mudah. Pengguna diharuskan untuk menunggu masing-masing *vendor* merilis

resmi *update* terbaru dari *sistem android* tersebut. Dan bagi beberapa pengguna, hal ini kurang begitu menyenangkan.

- 2) Baterai yang cepat habis, mungkin hal ini sering dialami oleh pengguna *sistem Android* saat ini. Apalagi jika Anda sering menyalakan paket data serta menggunakan *widget* dan aplikasi yang berjalan terlalu berlebihan sehingga menyebabkan daya baterai berkurang dengan cepat. Untuk mengatasi hal tersebut, Anda bisa mengurangi aktivitas yang ada pada *smartphone* Anda.
- 3) Sering mengalami lemot atau lag, hal ini biasanya berkaitan dengan *spesifikasi* dari masing-masing perangkat seluler. Namun meskipun begitu, jika *sistem android* memang tidak bersahabat dengan aplikasi-aplikasi yang Anda miliki tentu saja akan berdampak pada leletnya penggunaan *smartphone* Anda. Hal ini dikaitkan dengan *RAM* atau *prosesor* yang memang kurang memadai. Sehingga akan lebih baik jika menyimpan aplikasi yang memang sesuai dengan *smartphone* yang Anda miliki.

Android menawarkan sebuah lingkungan yang berbeda untuk pengembang. Setiap aplikasi memiliki tingkatan yang sama. *Android* tidak membedakan antara aplikasi inti dengan aplikasi pihak ketiga. *API* yang disediakan menawarkan akses ke *hardware*, maupun data-data ponsel sekalipun, atau data *system* sendiri. Bahkan pengguna dapat menghapus aplikasi inti dan menggantikannya dengan aplikasi pihak ketiga.

Android merupakan sistem operasi yang dikembangkan untuk perangkat *mobile* berbasis *Linux*. Pada awalnya sistem operasi ini dikembangkan oleh *Android Inc.* yang kemudian dibeli oleh Google pada tahun 2005. Dalam usaha mengembangkan *android*, pada tahun 2007 dibentuklah *Open Handset Alliance (OHA)*, sebuah konsorsium dari beberapa perusahaan, yaitu *Texas Instruments, Broadcom Corporation, google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel*, dan *T-Mobile* dengan tujuan untuk mengembangkan standar terbuka untuk perangkat *mobile*. Pada tanggal 9 Desember 2008, ia diumumkan bahwa 14 anggota baru akan bergabung Proyek *Android*, termasuk *PacketVideo, ARM Holdings, Atheros communications, asustek computer Inc, garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc* (Wahana Komputer, 2019).

4. *Unity*

Unity merupakan alat bantu pengembangan game dengan kemampuan *rendering* yang terintegrasi di dalamnya. Dengan menggunakan kecanggihan fitur-fiturnya dan juga kecepatan kerja yang tinggi, *Unity* dapat menciptakan sebuah program interaktif tidak hanya dalam 2 dimensi, tetapi juga dalam bentuk 3 dimensi. *Unity 3D* adalah *software* yang baik digunakan untuk mendukung pembelajaran interaktif, buku dan observasi lapangan ataupun informasi lain di internet (Fortuna, Purnamasari, and Dikananda 2023)

Adapun fitur-fitur dalam *unity* diantaranya :

a. *Rendering*

Graphics engine yang digunakan adalah *Direct3D* (*Windows, Xbox 360*), *OpenGL* (*Mac, Windows, Linux, PS3*), *OpenGL ES* (*Android, iOS*), dan *proprietary APIs* (*Wii*). Ada pula kemampuan untuk *bump mapping*, *reflection mapping*, *parallax mapping*, *screen space ambient occlusion* (*SSAO*), *dynamic shadows using shadow maps*, *render-to-texture* and *full-screen post-processing effects*.

b. *Asset Tracking*

Unity juga menyertakan *Server Unity Asset* – sebuah solusi terkontrol untuk *Developer game asset* dan *script*. *Server* tersebut menggunakan *PostgreSQL* sebagai *backend*, sistem *audio* dibuat menggunakan *FMOD library* (dengan kemampuan untuk memutar *Ogg Vorbis compressed audio*), *video playback* menggunakan *Theora codec*, *engine* daratan dan vegetasi (dimana mensupport *tree billboarding*, *Occlusion Culling* dengan *Umbra*), *built-in lightmapping* dan *global illumination* dengan *Beast*, *multiplayer networking* menggunakan *RakNet*, dan navigasi mesh pencari jalur *built-in*.

c. *Platform*

Unity support pengembangan ke berbagai *platform*. Di dalam project, *developer* memiliki kontrol untuk mengirim perangkat *mobile*, *web browser*, *desktop*, dan *console*. *Unity* juga mengijinkan spesifikasi kompresi *texture* dan pengaturan resolusi di setiap *platform* yang didukung.

Saat ini *platform* yang didukung adalah *BlackBerry 10*, *Windows 8*, *Windows Phone 8*, *Windows*, *Mac*, *Linux*, *Android*, *iOS*, *Unity Web Player*, *Adobe Flash*, *PlayStation 3*, *Xbox 360*, *Wii U* and *Wii*. Meskipun tidak semua

terkonfirmasi secara resmi, *Unity* juga mendukung *PlayStation Vita* yang dapat dilihat pada game *Escape Plan* dan *Oddworld: New 'n' Tasty*.

d. Asset Store

Diluncurkan November 2010, *Unity Asset Store* adalah sebuah *resource* yang hadir di *Unity* editor. *Asset store* terdiri dari koleksi lebih dari 4,400 *asset packages*, beserta *3D models*, *textures* dan *materials*, sistem *particle*, musik dan efek suara, tutorial dan project, *scripting package*, *editor extensions* dan servis online.

5. Augmented Reality (AR)

Augmented Reality (AR) merupakan teknologi yang menggabungkan benda maya dua dimensi maupun tiga dimensi ke dalam sebuah lingkungan nyata lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata. *Augmented Reality* dapat juga didefinisikan sebagai teknologi yang mampu menggabungkan objek maya dalam dua dimensi dan tiga dimensi ke dalam sebuah lingkungan nyata, kemudian memproyeksikan objek-objek tersebut secara real time. *Augmented Reality* menggabungkan dunia nyata dengan dunia virtual. (Khairani, Elvitaria, and Salamun 2021).

Tujuan dari teknologi *augmented reality* adalah untuk memungkinkan integrasi materi digital yang dibuat oleh komputer secara real-time dengan lingkungan luar. Pengguna *augmented reality* dapat melihat benda-benda virtual dalam dua atau tiga dimensi yang diproyeksikan ke lingkungan sebenarnya.

6. Bahasa Program C#

C# adalah bahasa pemrograman *object oriented* dan memiliki *class library* yang sangat lengkap yang berisi *prebuilt component*, sehingga memudahkan programmer untuk *men-develop* program lebih cepat. C# juga distandarkan oleh *Ecma International* pada Desember 2002. Dengan menggunakan C#, dapat dibuat bermacam aplikasi seperti aplikasi *console*, aplikasi *windows form*, aplikasi *web*, aplikasi *web services*, dan aplikasi untuk *mobile device* (Yulius Eka Agung, 2021).

7. Python

Menurut (Raschka, 2020) "Python tetap menjadi bahasa yang paling disukai untuk komputasi ilmiah, ilmu data, dan *machine learning*, meningkatkan kinerja dan produktivitas dengan memungkinkan penggunaan *library* tingkat rendah dan antarmuka pemrograman aplikasi tingkat tinggi yang bersih".

Augmented Reality (AR) dapat diimplementasikan menggunakan Python dengan berbagai *library* yang mendukung pengolahan gambar, pelacakan objek, dan interaksi dengan dunia nyata. Salah satu library populer yang digunakan untuk pengembangan AR dengan Python adalah *OpenCV* (untuk pemrosesan gambar) dan *ARUco* (untuk deteksi marker AR). Selain itu, library seperti *PyGame* atau *Vuforia* juga sering digunakan untuk pengembangan AR.

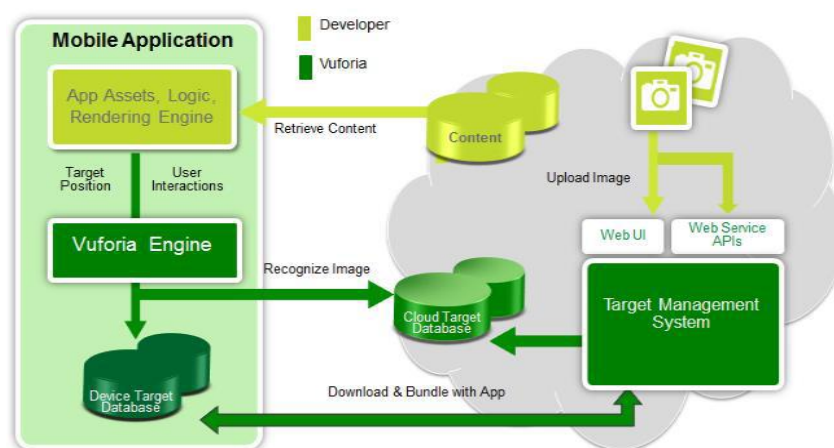
8. Vuforia

Vuforia merupakan salah satu *library* untuk *Virtual Reality*, yang menggunakan sumber yang konsisten mengenai *computer vision* yang fokus pada *image recognition*. *Vuforia* mempunyai banyak fitur – fitur dan kemampuan yang

dapat membantu pengembang untuk mewujudkan pemikiran mereka tanpa adanya batas secara teknis (Mario Fernando, 2019).

Vuforia mendukung untuk membuat aplikasi yang dapat digunakan di hampir seluruh jenis *smartphone* dan *tablet*. Pengembang juga diberikan kebebasan untuk mendesain dan membuat aplikasi yang mempunyai kemampuan antara lain

- Teknologi *computer vision* tingkat tinggi yang memungkinkan *developer* untuk membuat efek khusus pada *mobile device*.
- Dapat secara terus-menerus mengenal *multiple image*.
- Tracking* dan *detection* tingkat lanjut.
- Solusi pengaturan *database* gambar yang fleksibel seperti pada gambar berikut;



Gambar 2. 1, Struktur Vuforia

Target pada *vuforia* merupakan obyek pada dunia nyata yang dapat dideteksi oleh kamera, untuk menampilkan obyek virtual. Beberapa jenis target pada *vuforia* adalah :

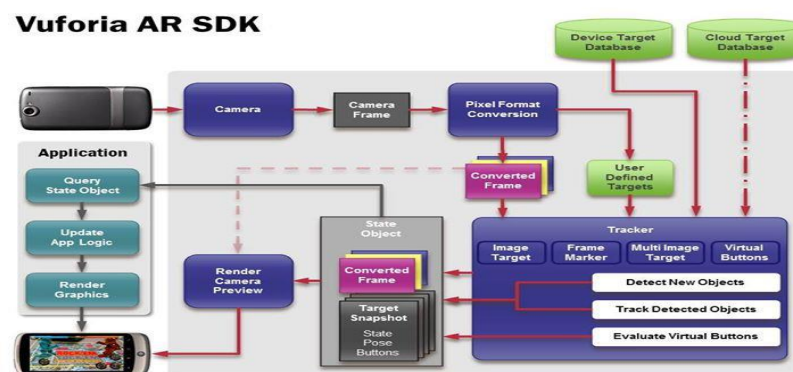
- Image target*, contoh : foto, papan permainan, halaman majalah, sampul buku, kemasan produk, poster, kartu ucapan.

- b. *Frame markers*, tipe *frame* gambar 2D dengan *pattern* khusus yang dapat digunakan sebagai permainan.
- c. *Multi-target*, contoh : kemasan produk atau produk yang berbentuk kotak ataupun persegi. Jenis ini dapat menampilkan gambar sederhana *Augmented 3D*.
- d. *Virtual buttons*, yang dapat membuat tombol sebagai daerah kotak sebagai sasaran gambar. Pada *vuforia*, ada 2 jenis *workflow* dengan dasar *database* yang dapat dipilih oleh *developer*, yaitu *cloud database* dan *device database*. Berikut ini adalah alur pembuatan aplikasi pada *vuforia*.

1. *Vuforia SDK*

Vuforia memerlukan beberapa komponen penting agar dapat bekerja dengan baik.

Berikut ini adalah gambaran dari diagram aliran data *vuforia* :



Gambar 2. 2, Diagram Aliran Data Vuforia

Beberapa penjelasan dari gambar diatas adalah sebagai berikut :

a. *Camera*

Digunakan untuk menangkap gambar per-*frame* kemudian mempersiapkan format dan ukurannya (*pixel*) menghasilkan "*camera-frame*".

b. *Pixel Format Conversion*

Setiap kamera *smartphone* memiliki format yang berbeda seperti *YUV*, *RGB*, *CMYK*, dan lain-lain. Oleh karena itu harus di *convert* menjadi format yang dapat diolah dengan baik oleh *Vuforia* yang berbasis *OpenGL*, kemudian menghasilkan "*converted frame*" yaitu format yang siap diolah oleh *Vuforia*.

c. *Tracker*

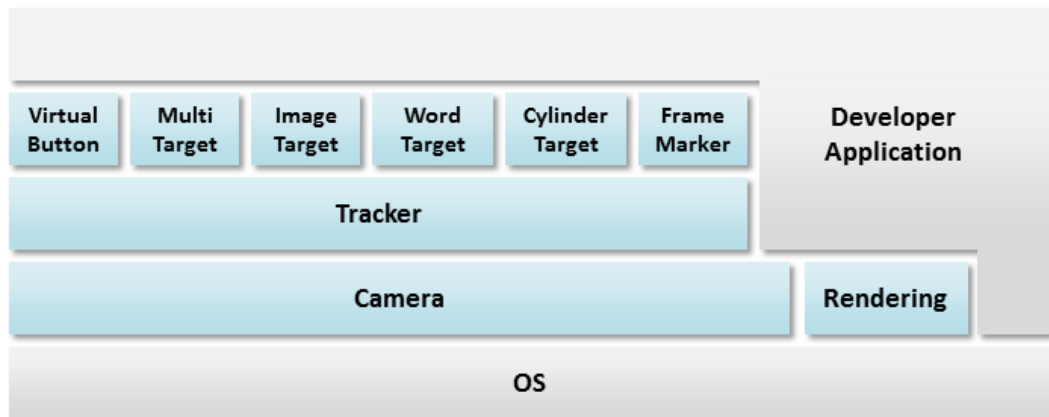
Merupakan *engine* inti dari *Vuforia*, yang berisi algoritma *computer vision* yaitu *SIFT* dan *FERNS* dengan metode *NFT (Natural Feature Tracking)*. Sehingga dapat melakukan *tracking* objek yang ada di dunia nyata (*converted frame*). *Tracking* marker dapat dilakukan pada benda seperti gambar 2D ataupun benda lainnya seperti meja, kursi, dan sebagainya. Marker yang dapat di *tracking* berasal dari database yang sudah dibuat sebelumnya, yaitu pada *cloud* ataupun pada *smartphone*).

d. *Application*

Merupakan tahapan pembangunan aplikasi bagi *developer*, pada bagian ini dilakukan pengolahan terhadap pembangunan aplikasi misalnya *coding*, mengatur *event* atau *action* yang dibutuhkan, serta mengatur objek yang akan ditampilkan pada aplikasi.

2. *Vuforia API References*

API Reference berisi informasi tentang hirarki kelas dan fungsi *member* dari *QCAR SDK*. Sistem *High-level* pada *vuforia* dapat dilihat pada Gambar dibawah ini :



Gambar 2. 3, Sistem High – Level Pada Vuforia

Sebuah gambaran dari *SDK* yang ditampilkan pada Gambar diatas ini menyediakan :

1. *Callbacks events* (Contoh : Sebuah gambar kamera baru tersedia)
2. *High – level access to hardware units* (contoh : Kamera mulai/berhenti)
3. *Multiple trackables* / jenis pelacakan yaitu dapat melalui :

a. *Image target*

Dapat mengenali gambar dengan detail yang cukup, termasuk majalah, iklan atau brosur serta kemasan yang tertera pada produk.

b. *Multi target*

Dapat mengenali lebih dari satu marker secara bersamaan.

c. *Cylinder Targets*

Dapat mengenali benda seperti botol, cangkir, kaleng, dan sebagainya.

d. *Word Targets*

Mendukung pengenalan kata bahasa inggris dari database standar 100.000 kata atau kosa kata kustom didefinisikan oleh pengembang.

e. *Frame Markers*

ID unik dari *frame marker* dikodekan ke dalam pola biner sepanjang perbatasan gambar *marker*. Sebuah *frame marker* memungkinkan gambar apapun untuk ditempatkan dalam batas-batas *marker*.

f. Real-World Interactions

Contoh : Penggunaan *virtual button* agar dapat berinteraksi dengan objek.

9. UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) adalah salah satu bahasa yang digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Putra and Andriani 2019).

Sedangkan menurut (Setiyani and Setiawan 2021) *Unified Modeling Language* (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis objek [6].

Metode *Unified Modelling Language* (UML) menggunakan tiga bangunan dasar untuk mendeskripsikan sistem atau perangkat lunak yang akan dikembangkan, yaitu:

a. Sesuatu (*things*)

Ada empat *things* dalam *Unified Modelling Language* (UML):

- 1) *Structural things*, bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.
- 2) *Behaviorial things*, bagian dinamis biasanya merupakan kata kerja dari

model UML yang mencerminkan perilaku sepanjang waktu

- 3) *Grouping things*, bagian pengorganisasian dalam UML. Dalam penggambaran model UML yang rumit diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model serta subsitem-subsistem.
- 4) *An notational things*, merupakan bagian yang memperjelas model UML. Dapat berisi komentar yang menjelaskan fungsi serta ciri-ciri tiap element dalam model UML.

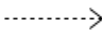
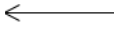
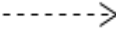
b. Relasi (*relationship*)

Ada empat *relationship* (hubungan) dalam *Unified Modelling Language* (UML):

1. Ketergantungan (*dependency*) adalah hubungan dimana perubahan yang terjadi pada suatu elemen independent akan mempengaruhi elemen yang bergantung padanya.
2. Asosiasi adalah apa dan bagaimana yang menghubungkan antara objek satu dengan yang lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.
3. Generalisasi adalah hubungan dimana objek anak berbagai perilaku dan struktur data dari objek yang ada di atasnya (objek induk). Arah dari objek induk ke objek anak dinamakan spesialisasi sedangkan arah sebaliknya dinamakan generalisasi..
4. Realisasi adalah operasi yang benar-benar dilakukan oleh suatu objek.

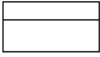
- c. Diagram, *Unified Modelling Language* (UML) menyediakan Sembilan jenis diagram yang dapat dikelompokkan berdasarkan sifatnya (statis dan dinamis). *Unified Modelling Language* (UML) sudah banyak digunakan untuk membuat desain dari suatu sistem, adapun beberapa jenis UML yang sering digunakan seperti *Use Case Diagram*, *Sequence Diagram*, *Activity Diagram* dan *Class Diagram*. Adapun simbol dari UML sebagai berikut:

Tabel 2. 3*Symbol Use Case Diagram*

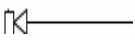
No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Orang, organisasi, atau sistem yang berinteraksi dengan sistem yang sedang dimodelkan. Ini direpresentasikan oleh gambar manusia.
2		<i>Dependency</i>	Penggunaan <i>Dependency</i> pada diagram use case membantu menyederhanakan dan mengorganisir kompleksitas hubungan antar elemen dalam sistem, yang pada gilirannya mendukung pemahaman yang lebih baik terhadap kebutuhan dan fungsionalitas sistem.
3		<i>Generalization</i>	<i>Generalization</i> membantu dalam memahami hierarki <i>use case</i> , di mana <i>use case</i> yang lebih umum dapat memiliki satu atau lebih <i>use case</i> anak yang lebih spesifik.
4		<i>Include</i>	<i>Include</i> digunakan untuk menggambarkan bagaimana suatu <i>use case</i> dapat memasukkan (menggunakan) fungsionalitas dari <i>use case</i> lain untuk melengkapi tugas atau skenario tertentu.
5		<i>Extend</i>	menggambarkan situasi di mana sebuah <i>use case</i> dapat menambah atau memperluas fungsionalitas dari <i>use case</i> lain dalam suatu sistem
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menyepesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan

No	Gambar	Nama	Keterangan
			suatu hasil yang terukur bagi suatu <i>actor</i>
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 4 *Symbol Class Diagram*

No.	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan di mana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Tabel 2. 5 *Symbol Sequence Diagram*

No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-

			informasi tentang aktivitas yang terjadi
--	--	--	--

Tabel 2. 6 *Symbol Activity Diagram*

No.	Gambar	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

B. Kajian Hasil Penelitian Terdahulu

1. (Wahyuddin and Hasnawati 2023)Universitas Muhammadiyah Parepare. “Aplikasi Media Pembelajaran Pengenalan Pancasila Menggunakan *Augmented Reality*” Penelitian ini bertujuan untuk menguji efektivitas aplikasi pengenalan nilai Pancasila berbasis augmented reality pada anak-anak sekolah dasar. Perbedaan dengan penelitian yang akan dilaksanakan yaitu penelitian ini berfokus pada media interaktif pembelajaran pengenalan nutrisi buah dan sayur untuk anak usia dini ketahui dan untuk memberikan pemahaman tentang pentingnya mengkonsumsi buah dan sayur.
2. (Triwibowo and Mangkunegara 2024)Universitas Harapan Bangsa “Perancangan *Augmented Reality* Untuk Meningkatkan Pengetahuan Orang

Tua Dan Anak Tentang Konsumsi Buah-Buahan” Penelitian ini bertujuan untuk merancang aplikasi AR yang dapat meningkatkan pengetahuan orang tua dan anak tentang Konsumsi buah buahan. Adapun perbedaan dengan penelitian yang akan dilaksanakan yaitu penelitian ini akan memberikan pemahaman tentang nutrisi yang ada pada buah dan sayur , dan menjelaskan apa saja manfaat yang terdapat pada buah dan sayur itu sendiri .

3. (Hiperakif 2023)Universitas Muhammadiyah Sidoarjo “*Augmented Reality* Sebagai Media Pembelajaran Tumbuhan di Wisata Cobaan Binangun Dengan Menggunakan Metode *Single Marker*” Penelitian ini bertujuan untuk menghasilkan sebuah aplikasi pembelajaran *Augmented Reality* tumbuhan yang diharapkan bisa digunakan untuk membantu mengenalkan jenis-jenis tumbuhan. Aplikasi ini dibuat menggunakan *Unity* dan *Vuforia* yang dijalankan berbasis *android* .Dengan adanya aplikasi ini dapat memperkenalkan berbagai jenis tumbuhan yang ada di wisata coban binangun. Adapun perbedaan dengan penelitian yang akan dilakukan yaitu fokus pada pengenalan buah dan sayur yang memberikan penjelasan untuk anak usia dini ketahui, media pembelajaran ini tidak hanya akan meningkatkan pemahaman dan minat anak-anak terhadap Buah dan Sayur , tetapi juga memberikan alternatif inovatif untuk metode pengajaran konvensional yang ada saat ini.

C. Kerangka Berpikir

Untuk mempermudah alur dari penelitian ini maka dibuatlah kerangka berpikir yang dibuat dalam bentuk diagram;

PAUD (Pendidikan Anak Usia Dini) merupakan upaya pembinaan anak semenjak lahir hingga usia enam tahun melalui pemberian rangsangan pendidikan agar memiliki kesiapan secara jasmani dan rohani saat memasuki pendidikan lebih lanjut. Pemberian rangsangan dengan media pembelajaran yang tepat akan mempengaruhi proses pertumbuhan dan perkembangannya, oleh sebab itu rangsangan perlu dilakukan sedini mungkin seperti pada Masa keemasan (*Golden Age*), *golden age* anak-anak berlangsung dari mulai lahir sampai usia 6 tahun.



Anak cenderung lebih tertarik jika dalam proses belajarnya disertai dengan suatu hal yang interaktif dan rata-rata anak-anak sekarang sangat cepat tahu akan menggunakan *smartphone*. Pembuatan aplikasi media pembelajaran dengan menerapkan manfaat buah dan sayur dengan menampilkan video yang interaktif ke dalam aplikasi tersebut merupakan solusi yang terbaik.



Maka dari itu Peneliti membuat suatu aplikasi dengan judul ” Perancangan Media Pembelajaran Pengenalan Manfaat Pada Buah-Buahan dan Sayuran Berbasis *Augmented Reality* Untuk Anak Usia Dini “Diharapkan aplikasi Pembelajaran tentang Buah dan Sayuran ini dapat mudah dipahami oleh anak-anak usia dini .



Adapun aplikasi ini berbasis *android* yang dibuat dengan menggunakan *Unity Editor* dan Bahasa pemrograman *C#* dengan memanfaatkan perangkat mobile dan *android* sebagai sistem operasinya.

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang digunakan adalah penelitian deskriptif. Penelitian deskriptif adalah jenis penelitian yang bertujuan untuk menggambarkan atau mendeskripsikan fenomena yang sedang diteliti secara sistematis dan akurat. Dalam konteks pendidikan, penelitian deskriptif sering digunakan untuk memahami situasi atau kondisi saat ini, mengidentifikasi masalah, atau menggambarkan karakteristik suatu kelompok atau fenomena.

Penelitian deskriptif akan memberikan gambaran yang komprehensif tentang bagaimana implementasi media pembelajaran berbasis AR berlangsung dan bagaimana hal itu mempengaruhi proses pembelajaran untuk anak usia dini.

B. Waktu dan Tempat Penelitian

Adapun lokasi penelitian dilaksanakan selama $\pm$ 3 bulan dengan tempat penelitian di TK Paud Melati Pare-Pare.

C. Alat dan Bahan

Untuk melakukan proses penelitian, maka yang harus diperlukan adalah alat dan bahan penelitian, guna mendukung kegiatan penelitian tersebut. Adapun alat dan bahan sebagai berikut:

a. Alat

Alat yang digunakan yaitu :

- 1) Laptop *HP* dengan spesifikasi :

a) *Processor : AMD AthlonGold 3150U(4 CPUs),2,4GHz*

b) *RAM : 8.00 GB*

c) *Ssd : 512 GB*

2) *Perangkat Android Mobile*

a) *Processor :Intel® Atom™ CPU Z2560 1.60 Ghz*

b) *RAM : 8 GB*

c) *LCD : 6.67 “*

3) *Software yang digunakan dalam pembuatan aplikasi yaitu :*

a) *Windows 11*

b) *Unity, Blender 3D*

c) *Vuforia*

d) *Adobe Illustrator, Draw.io*

e) *C#*

f) *Canva*

b. *Bahan*

Adapun bahan yang diperlukan : Marker (*image target*) Gambar Buah-Buahan dan Sayuran Itu Sendiri.

D. Metode Pengumpulan Data

Metode pengumpulan data yang tepat yaitu dengan mempertimbangkan penggunaan berdasarkan jenis data dan sumbernya, sehingga peneliti melakukan teknik pengumpulan data sebagai berikut:

Studi Literatur

Mengumpulkan data dengan mempelajari masalah dengan objek yang diteliti, bersumber dari buku-buku, literatur yang disusun oleh para ahli untuk melengkapi data yang diperlukan dalam penelitian baik secara langsung maupun *Online*.

E. Tahapan Penelitian

Tahap penelitian yang dilakukan terdiri dari tujuh tahap penelitian tersebut dan dapat dilihat sebagai berikut:

1. Persiapan Penelitian

Tahap ini peneliti mempersiapkan alat, bahan seperti yang telah dijelaskan di atas serta menyiapkan buku yang terkait dengan topik yang akan diteliti.

2. Pengumpulan Data

Tahap ini peneliti melakukan pengumpulan informasi sebanyak-banyaknya mengenai topik melalui studi literatur.

3. Analisis Data

Tahap ini peneliti melakukan analisa data yang telah didapatkan dari pengumpulan data yang sebelumnya telah dilakukan. Dalam hal ini peneliti mempersiapkan rancangan yang akan diterapkan dari sistem yang telah ada ke sistem yang baru.

4. Perancangan Sistem

Tahap ini peneliti akan mulai merancang sistem berdasarkan Analisa data yang telah dilakukan dan melakukan pembuatan desain sistem terkait desain sistem yang berjalan dan desain sistem yang

diusulkan serta membuat desain *User Interface* aplikasi .

5. Pembuatan Aplikasi

Tahap ini peneliti akan membuat aplikasi sesuai dengan rancangan sistem yang telah dirancang.

6. Pengujian

Peneliti akan melakukan pengujian aplikasi untuk mengetahui apakah hasil perancangan terdapat kekurangan, jika ada peneliti akan kembali ke tahap analisis.

7. Implementasi

Tahap ini merupakan tahap dimana peneliti mengimplementasikan aplikasi yang telah dibuat kepada anak (TK) Paud terpadu Melati.

F. Metode Pengujian

Setelah aplikasi telah dibuat maka dilakukan pula pengujian pada aplikasi tersebut. Dalam penelitian ini digunakan 2 (dua) metode untuk melakukan pengujian pada sistem aplikasi yaitu *Black box Testing*, *whitebox Testing* yang bisa dijelaskan sebagai berikut :

1. *White box Testing*

Dalam jurnal (Gusdevi et al. 2022) Dijelaskan *White-box testing* atau pengujian *white-box* dilakukan untuk menguji dan menganalisis kode program bilamana terjadi kesalahan atau tidak di sebut dengan pengujian *white box* . Terdapat pendapat lain mengenai pengertian dari pengujian *white box* ini dilakukan dengan melihat pure kode tanpa melihat tampilan *interface* dari halaman aplikasi . *White Box* sendiri mempunyai beberapa teknik di dalam

pengujiannya, seperti : *Data Flow Testing*, *Control Flow Testing*, *Basic Path / Path Testing*, dan *Loop Testing* .

Kelebihan dari penggunaan metode *white-box testing* adalah dapat memperlihatkan galat pada *kode* yang dibuat dengan menghapus baris yang tidak diperlukan serta maksimalnya cakupan pengujian aplikasi saat uji coba sebuah *scenario* .

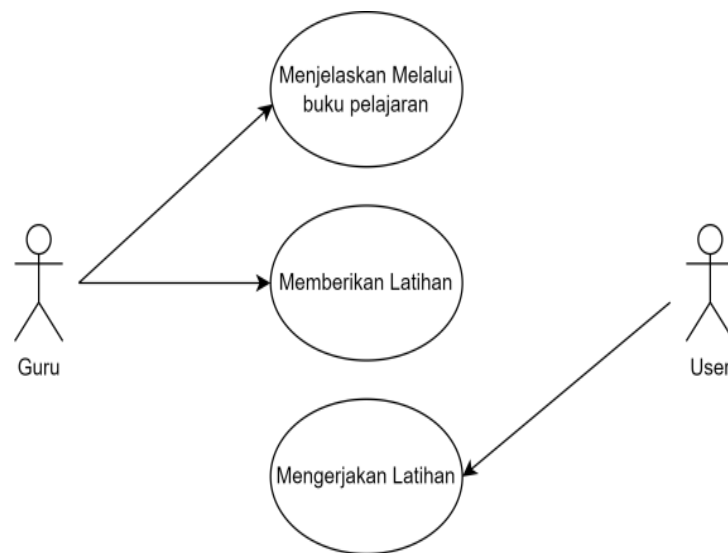
2. *Black box Testing*

Metode *Black Box Testing* merupakan pengujian untuk menunjukkan kesalahan pada system aplikasi seperti kesalahan pada fungsi system aplikasi, serta menu aplikasi yang hilang. Jadi *Black Box testing* merupakan metode uji *fungsionalitas system* aplikasi. Dalam melakukan pengujian menggunakan masukan data acak dengan tujuan untuk mendapatkan hasil yang pasti . Dikatakan pasti artinya bila salah ,maka di tolak oleh system informasi atau data input tersebut tidak dapat disimpan dalam database, sedangkan bila data input benar maka dapat di terima/masuk di database informasi (Gusdevi et al. 2022)

G. Desain Sistem

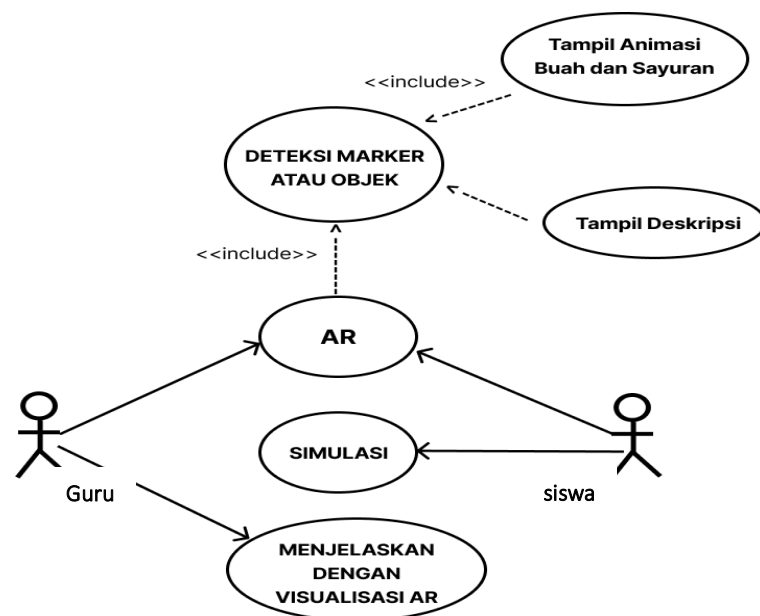
1. Desain Sistem yang berjalan

Sistem pembelajaran saat ini dapat dilihat pada gambar di bawah, di mana guru memberikan penjelasan melalui buku pelajaran dan memberikan latihan kepada siswa, kemudian siswa menyelesaikan latihan tersebut. Adapun desain sistemnya sebagai berikut ;



Gambar 3. 1 Desain yang berjalan

2. Desain Sistem Yang di Usulkan

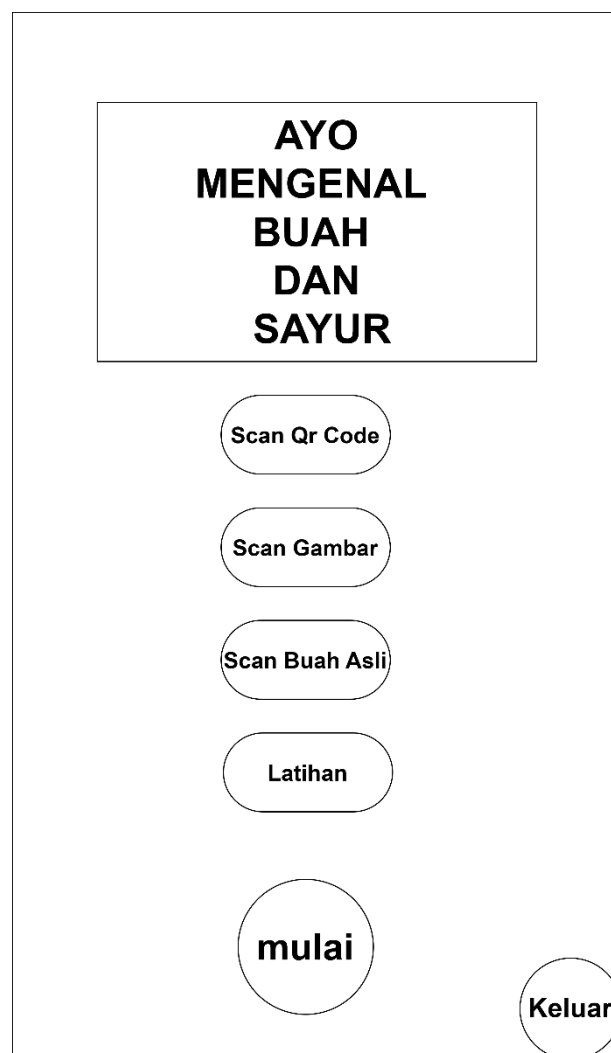


Gambar 3. 2 Desain yang di usulkan

Adapun Sistem yang diusulkan menawarkan pengalaman belajar yang seru dan interaktif bagi siswa melalui aplikasi AR. Dengan fitur Scan Buah dan

Sayur, siswa dapat menjelajahi berbagai jenis buah dan sayuran dalam bentuk 3D yang realistis, seolah-olah mereka benar-benar ada di depan mata. Selain itu, siswa juga dapat dengan mudah mengakses penjelasan mendetail tentang sistem ini, membantu mereka memahami cara kerjanya dengan lebih jelas. Semua ini menjadikan proses belajar tidak hanya *informatif*, tetapi juga penuh dengan eksplorasi yang menyenangkan.

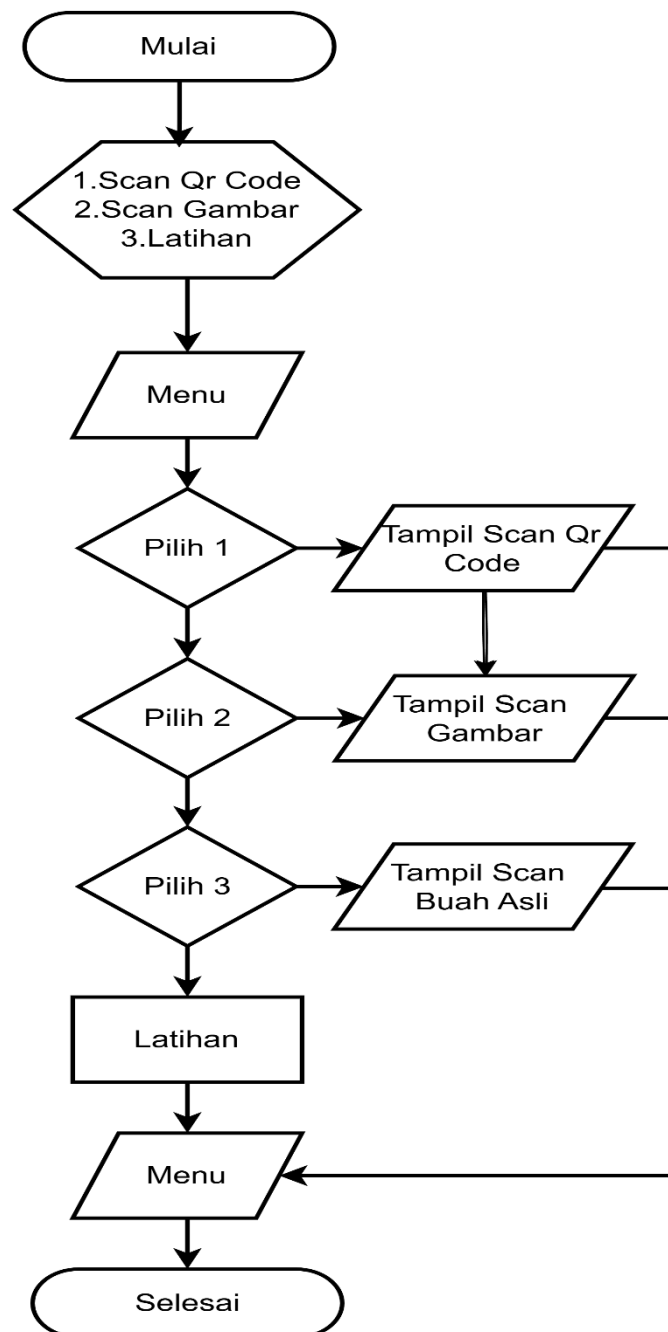
3. Desain *interface*



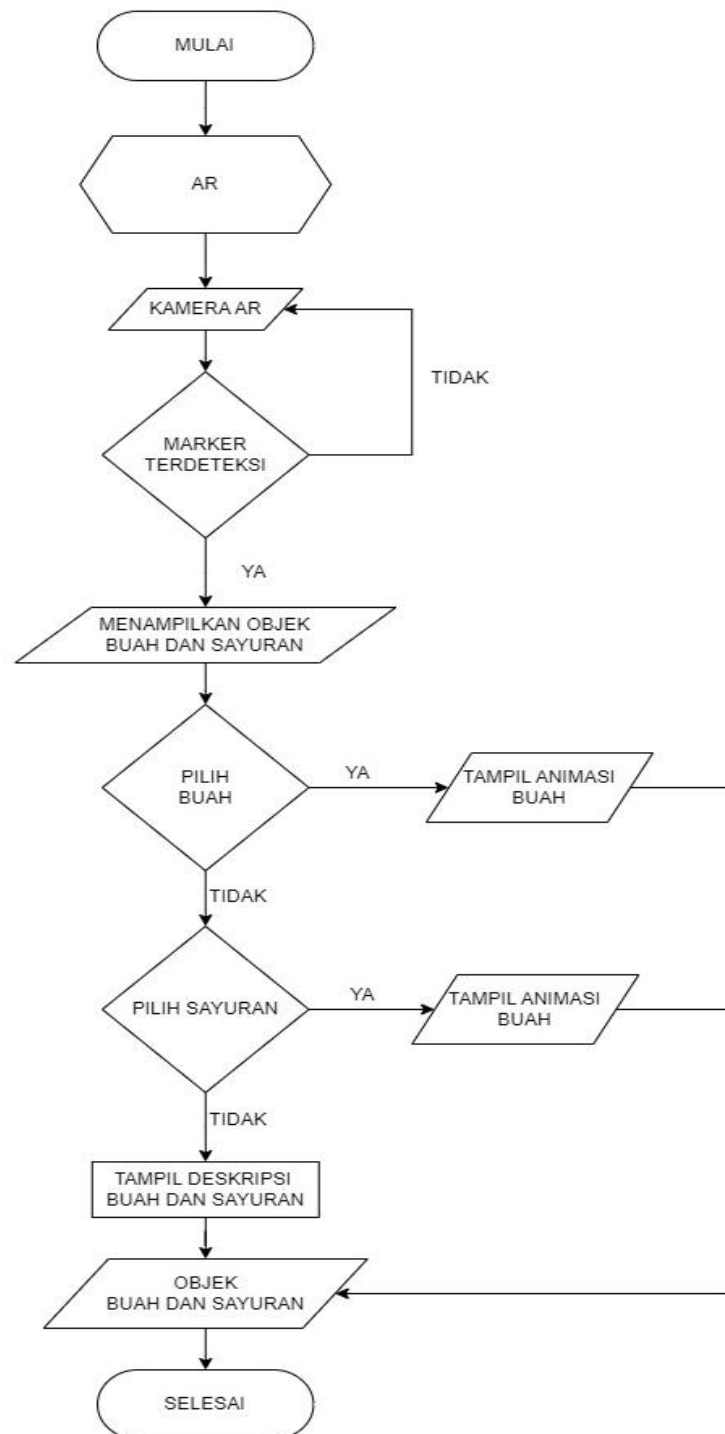
Gambar 3. 3 Desain *Interface*

4. Flowchart

a. Menu



Gambar 3. 4 *Flowchart menu*

b. Scan**Gambar 3.5** *Flowchart Scan*

BAB IV

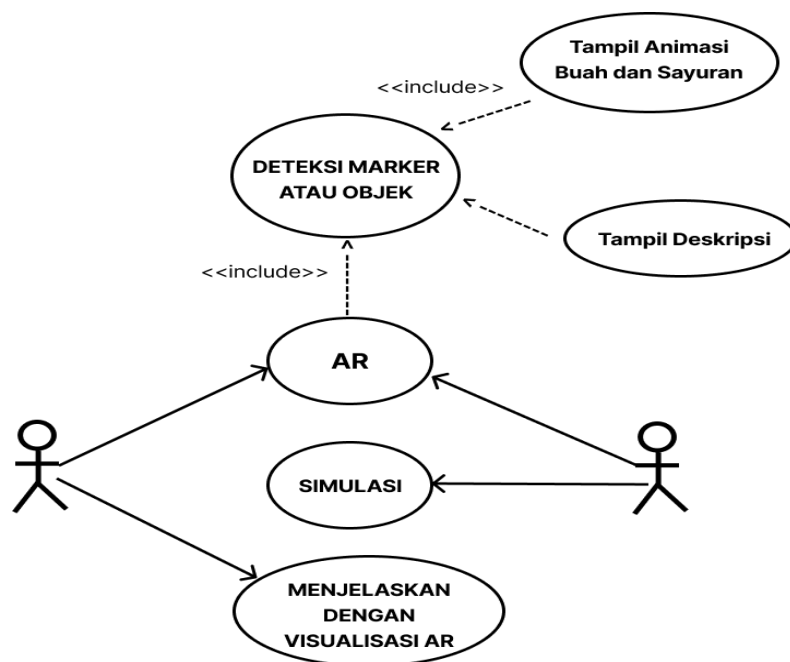
HASIL DAN PEMBAHASAN

A. Analisis Aliran Data UML

Perencanaan sistem ini memiliki tujuan yakni untuk membantu siswa dapat memahami dengan mudah materi pembelajaran bangun ruang yang dibuat dengan teknologi *Augmented reality* atau biasa disingkat AR, kemudian peneliti merancang sistem dengan menggunakan *Use Case diagram*, *Activity diagram* dan *sequence diagram*.

1. *Use Case diagram*

Use Case Case diagram dirancang guna untuk menentukan fungsi sistem oleh pengguna. Diagram



Gambar 4. 1 *Use Case Diagram*

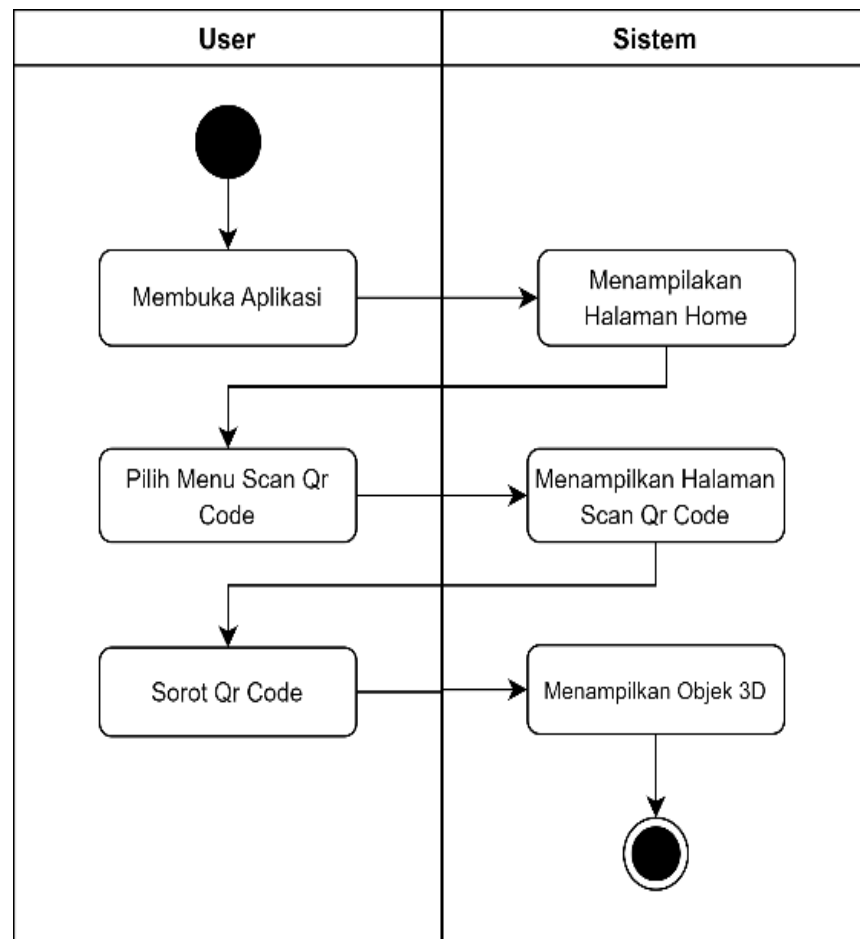
Tabel 4. 1 Deskripsi *Use Case*

Nama <i>Use Case</i>	Deskripsi
AR	<i>Use Case ini</i> menjelaskan dimana siswa dan guru dapat menggunakan fitur <i>Augmented reality</i> .
Latihan	<i>Use Case ini</i> menguraikan bagaimana siswa di latih dengan memilih buah yang benar.
Visualsasi AR	<i>Use Case ini</i> menjelaskan dimana aplikasi ini dapat menampilkan buah dan sayur secara 3D.
Deteksi <i>Marker</i> ,Qr Code dan Buah Asli	<i>Use Case ini</i> menjelaskan proses deteksi <i>marker</i> atau benda hingga menghasil objek dalam bentuk 3D
Tampil informasi buah dan sayur	<i>Use Case ini</i> menjelaskan dimana siswa dapat melihat <i>marker</i> dan Qr code yang di sorot.

2. *Activity diagram*

Activity diagram adalah skema yang mampu mensimulasikan proses sistemik. Urutan proses sistem ditampilkan secara vertikal. Kasus penggunaan dengan *activity diagram* dikembangkan menjadi beberapa *activity diagram* yang ada .

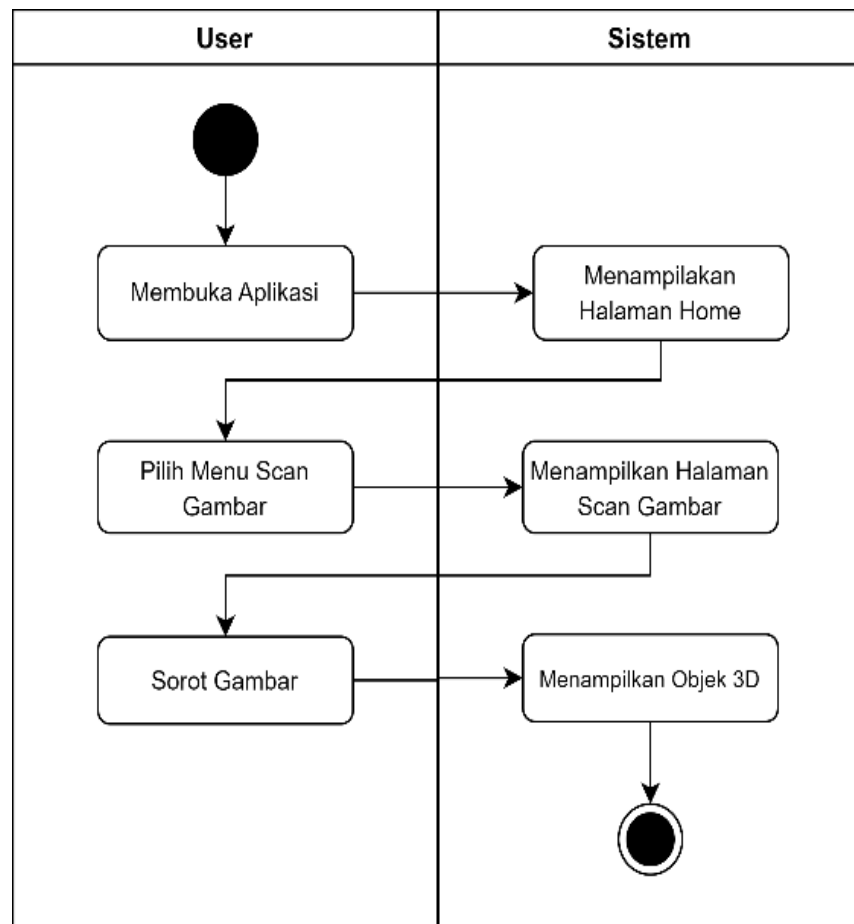
a. *Activity diagram Scan Qr code*



Gambar 4. 2 *Activity diagram Scan Qr code*

Pada gambar 4. 2 menjelaskan *Activity Scan Qr code* menggunakan *Scan marker* Buah dan Sayuran . Dimulai dari *user* membuka aplikasi dan sistem menampilkan *home*, kemudian *user* memilih atau mengklik *menu Scan Qr code* dan sistem menampilkan tampilan halaman *Scan Qr code* setelah itu *user* menyorot *Qr code* lalu sistem menampilkan objek 3D Pengenalan Buah dan Sayuran.

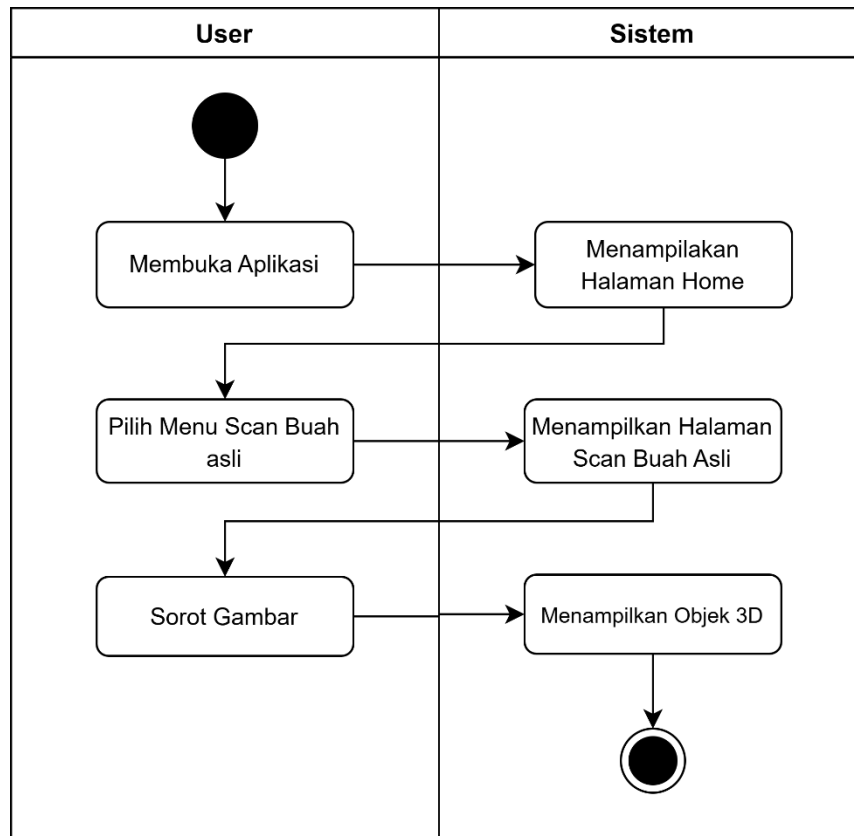
b. *Activity diagram Scan Gambar*



Gambar 4. 3 *Activity diagram Scan Gambar*

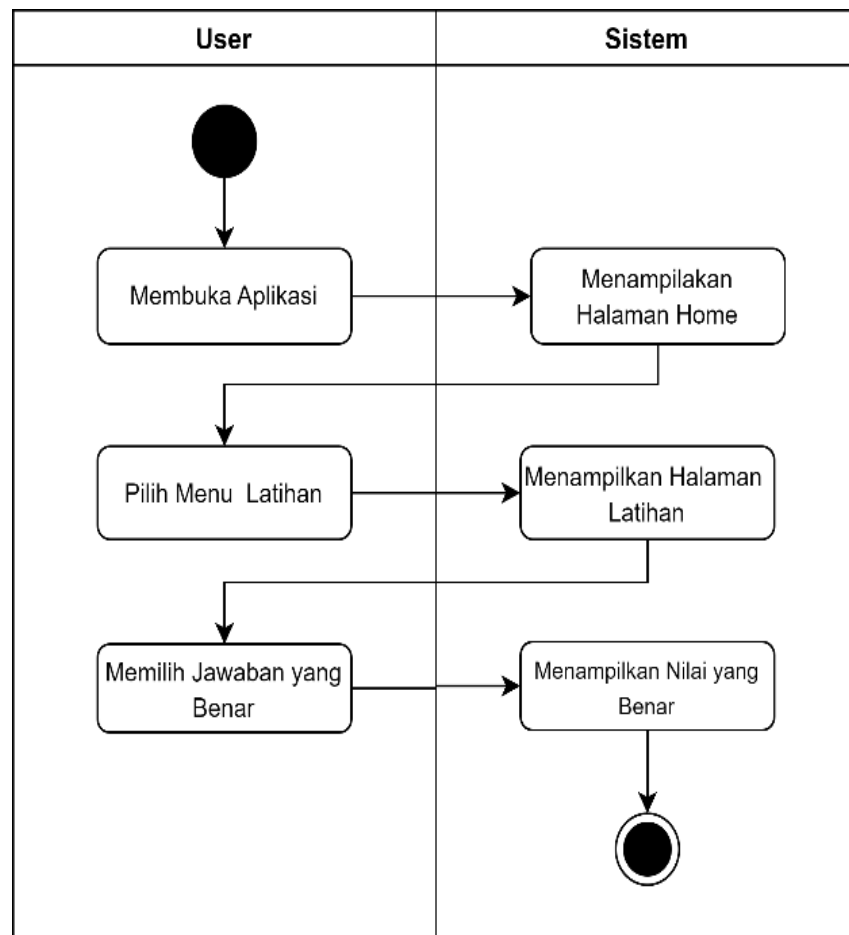
Pada gambar 4. 3 menjelaskan *Activity Scan gambar* menggunakan *Scan marker* Buah dan Sayuran . Dimulai dari *user* membuka aplikasi dan sistem menampilkan *home*, kemudian *user* memilih atau mengklik *menu Scan gambar* dan sistem menampilkan tampilan halaman *Scan gambar* setelah itu *user* menyorot gambar lalu sistem menampilkan objek 3D Pengenalan Buah dan Sayuran .

c. *Activity* diagram Buah Asli



Gambar 4. 4 *Activity* diagram Buah Asli

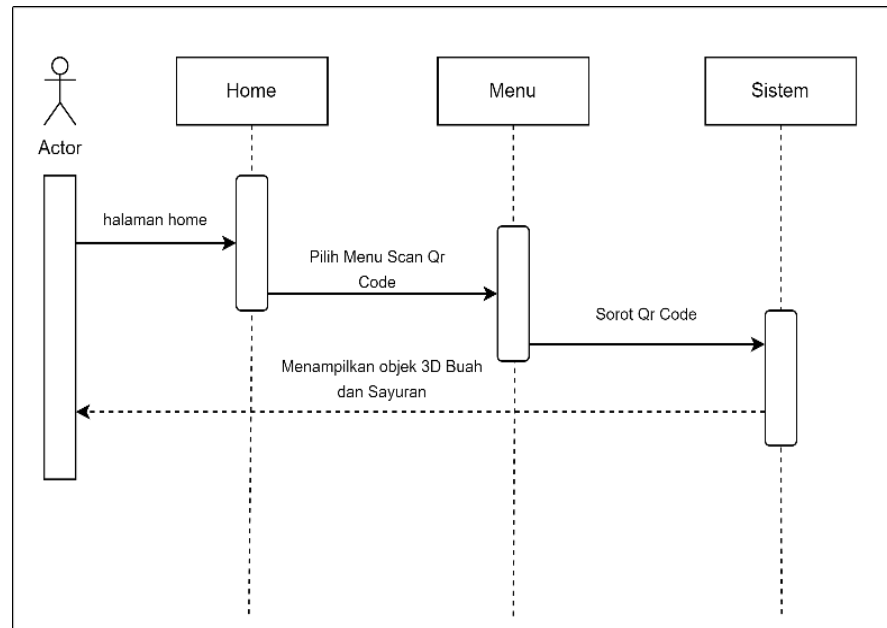
Pada gambar 4. 4 menjelaskan *Activity Scan* Buah Asli menggunakan *Scan marker* Buah . Dimulai dari *user* membuka aplikasi dan sistem menampilkan *home*, kemudian *user* memilih atau mengklik *menu Scan* Buah Asli dan sistem menampilkan tampilan halaman *Scan* Buah Asli setelah itu *user* menyorot gambar lalu sistem menampilkan objek 3D Pengenalan Buah .

d. *Activity diagram Latihan***Gambar 4. 5** *Activity diagram Latihan*

Pada gambar 4. 5 menjelaskan *Activity diagram* latihan menggunakan pilihan Buah dan Sayuran . Dimulai dari *user* membuka aplikasi dan sistem menampilkan *home*, kemudian *user* mengklik *menu* latihan dan sistem menampilkan tampilan halaman latihan setelah itu user memilih jawaban yang benar dengan 4 pilihan jawaban dan hanya satu yang benar kemudian menampilkan jawaban *user* memperoleh nilai yang benar .

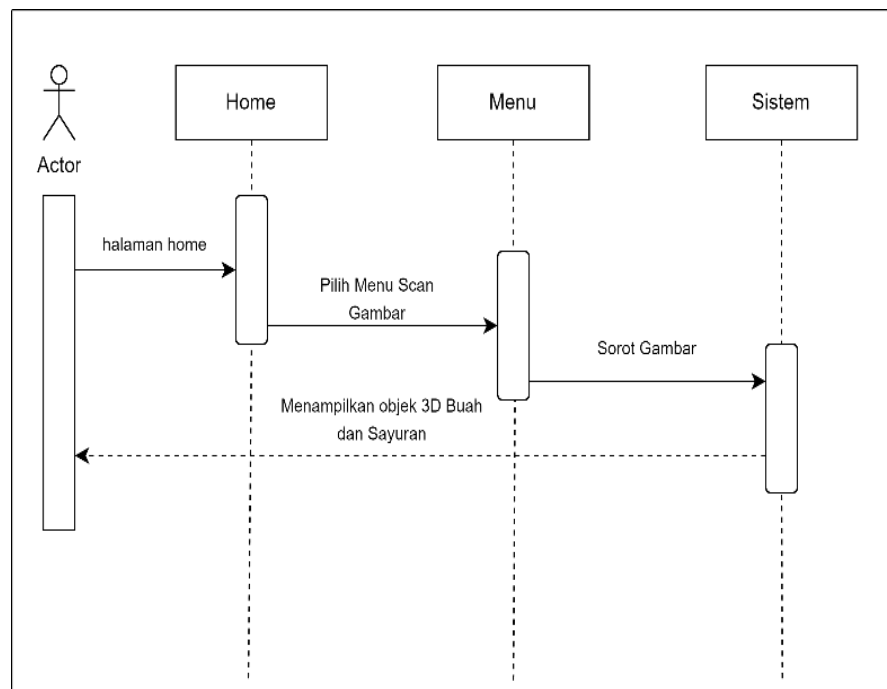
3. Sequence Diagram

a. Sequence diagram scan Qr code



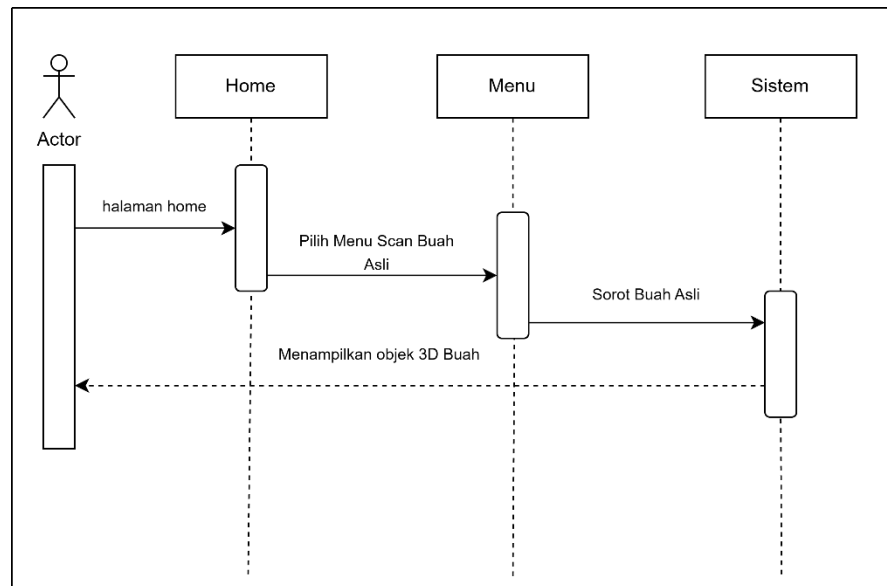
Gambar 4. 6 Sequence Diagram Scan Qr code

b. Sequence Diagram Scan Gambar



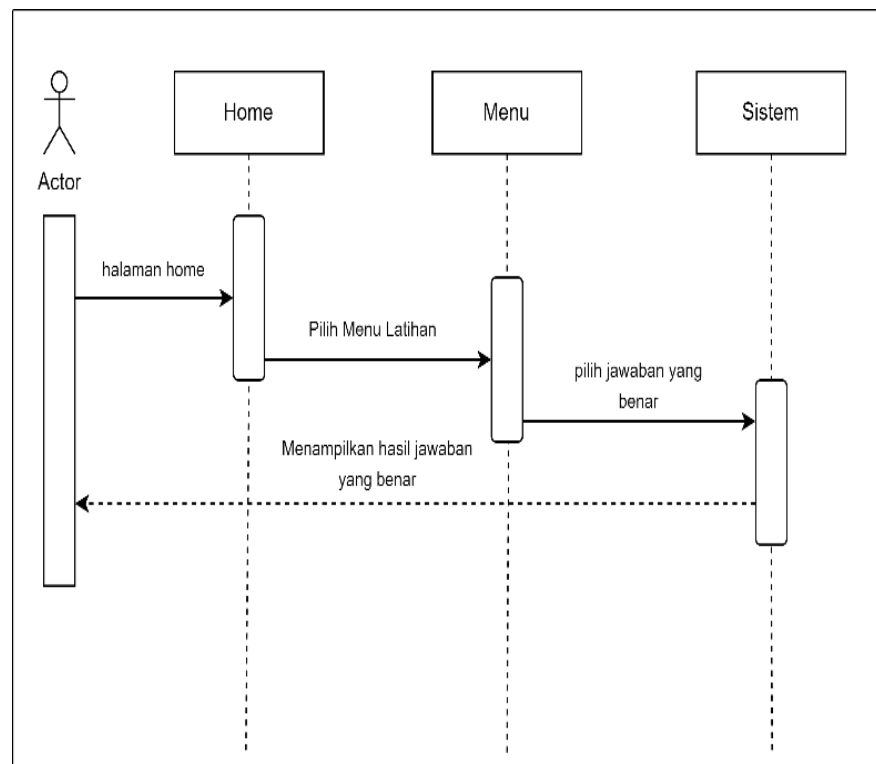
Gambar 4. 7 Sequence Diagram Scan Gambar

c. *Sequence Diagram Buah Asli*



Gambar 4. 8 *Sequence Diagram Scan Buah Asli*

d. *Sequence Diagram Latihan*



Gambar 4. 9 *Sequence Diagram Scan Latihan*

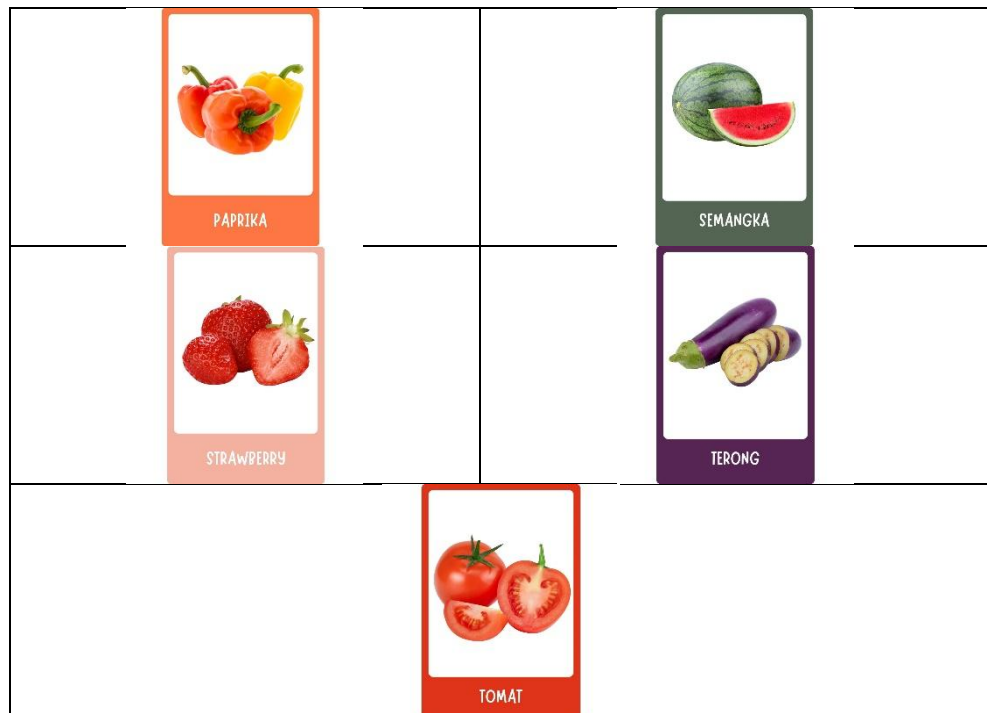
B. Detail Sistem

1. *Marker* Gambar

Pembuatan *marker* gambar dibuat menggunakan *tool* desain gambar yaitu *Canva*. Berikut *marker* gambar yang telah dibuat.

Tabel 4. 2 *Marker* Gambar

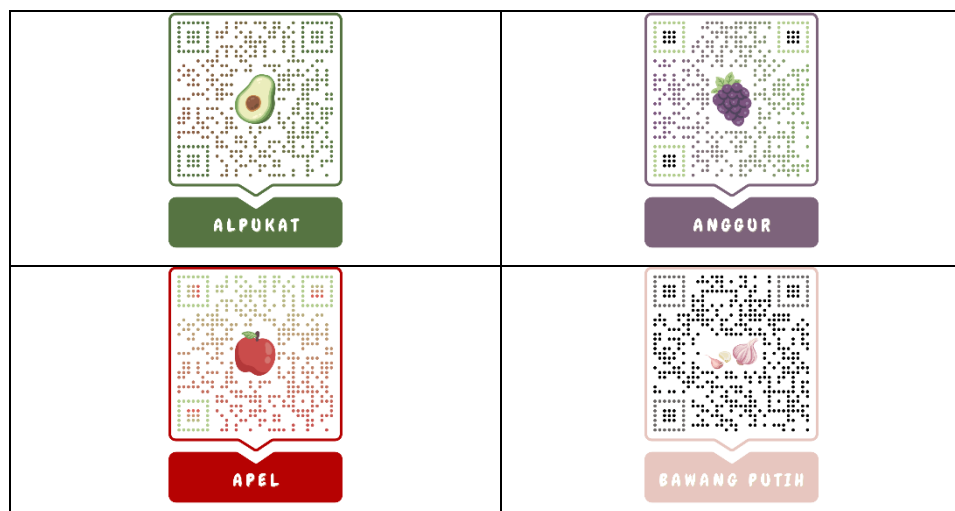
	 ALPUKAT			 ANGGUR	
	 APEL			 PAWANG PUTIH	
	 BROKOLI			 CAPAI	
	 JERUK			 LEMON	
	 MANGGA			 NANAS	

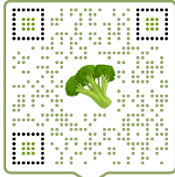
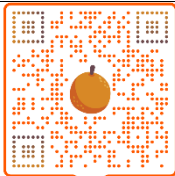
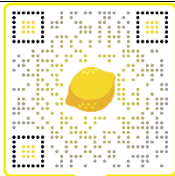
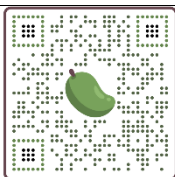
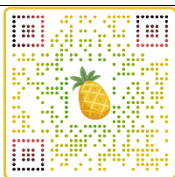
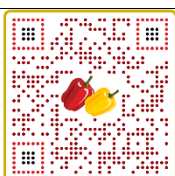
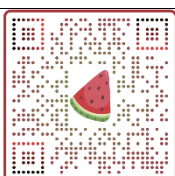
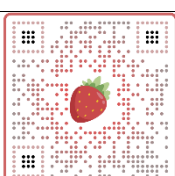
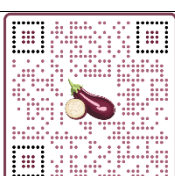
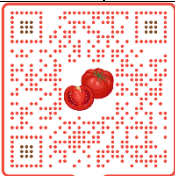


2. Marker Qr Code

Pembuatan *marker* Qr Code dibuat menggunakan *tool* desain gambar yaitu *Canva*. Berikut *marker* gambar yang telah dibuat.

Tabel 4. 3 *Marker Qr Code*



 BROKOLI	 CABAI
 JERUK	 LEMON
 MANGGA	 NANAS
 PAPRIKA	 SEMANGKA
 STRAWBERRY	 TERONG
 TOMAT	

3. Tampilan Utama Aplikasi

Masuk aplikasi *unity* - Pilih *Scene Home* - Tambahkan Komponen (*Canvas Panel* dan *Button*) - Atur ukuran Canvas dan Panel - Tambahkan Gambar 2D di *Inspector Image* pada Panel - Tambahkan juga Gambar 2D Pada *Inspector Image* Pada ke 2 *Button* .

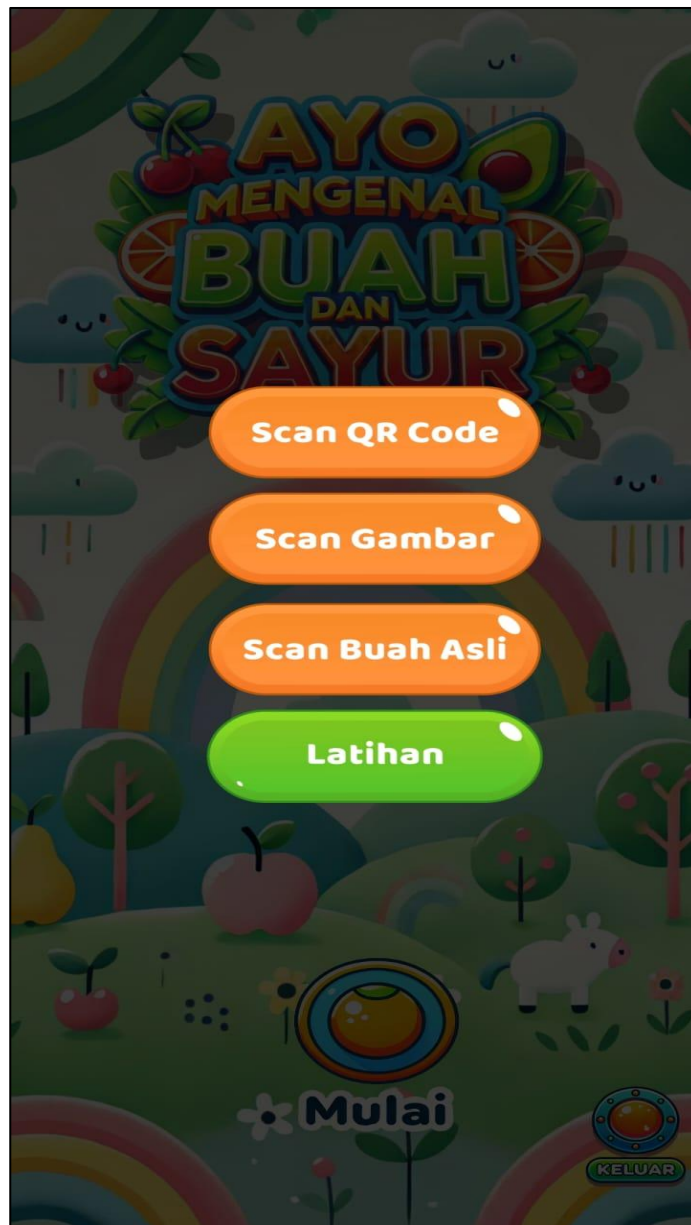
Tampilan utama aplikasi terdapat tombol mulai dan Tulisan ayo mengenal Buah dan Sayur Kemudian tombol keluar untuk keluar dari aplikasi.



Gambar 4. 10 Tampilan Utama

4. Tampilan *Pop up* Menu

Tampilan ini memperlihatkan pilihan untuk memilih yang ingin di Klik terdapat 4 pilihan ada *Marker Scan Qr Code* ,*Scan Gambar*,*Scan Buah* asli dan Latihan. di ke4 pilihan ini memiliki fungsi masing-masing.



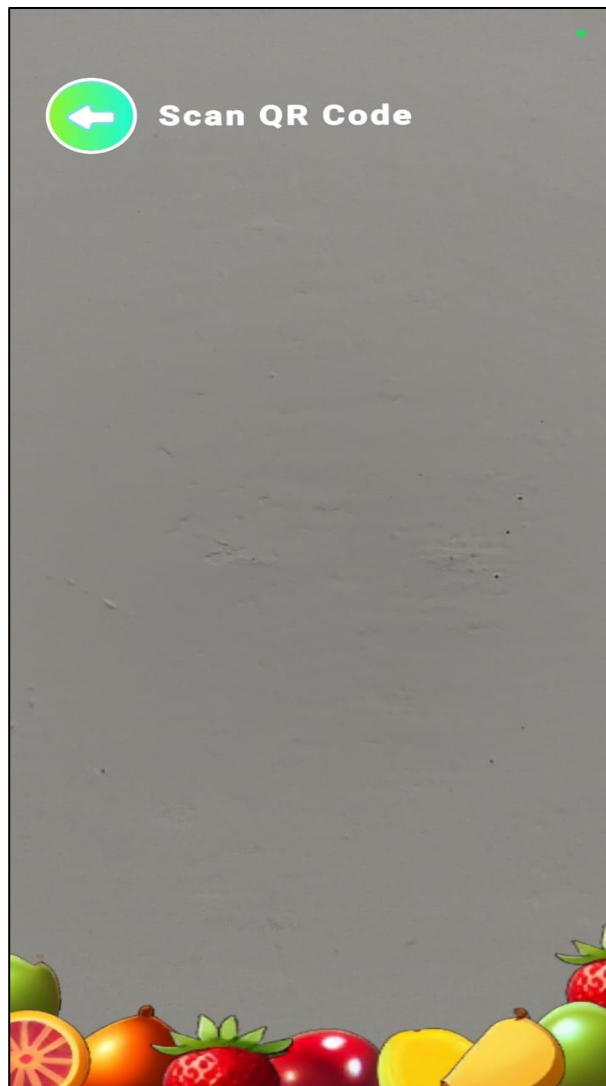
Gambar 4. 11 Tampilan *pop up* menu

Tabel 4. 4 *Script popup Controller Menu*

```
using UnityEngine;
public class PopupController : MonoBehaviour
{
    private Animator animator;
    private bool canShowPopup = true;
    void Start()
    {
        animator = GetComponent<Animator>();
    }
    public void ShowPopup()
    {
        if (canShowPopup)
        {
            animator.SetTrigger("Show");
            canShowPopup = false;
        }
    }
    public void HidePopup()
    {
        animator.SetTrigger("Hide");
    }
    // Panggil fungsi ini di akhir animasi
    "Hide"
    public void EnableShow()
    {
        canShowPopup = true;
    }
}
```

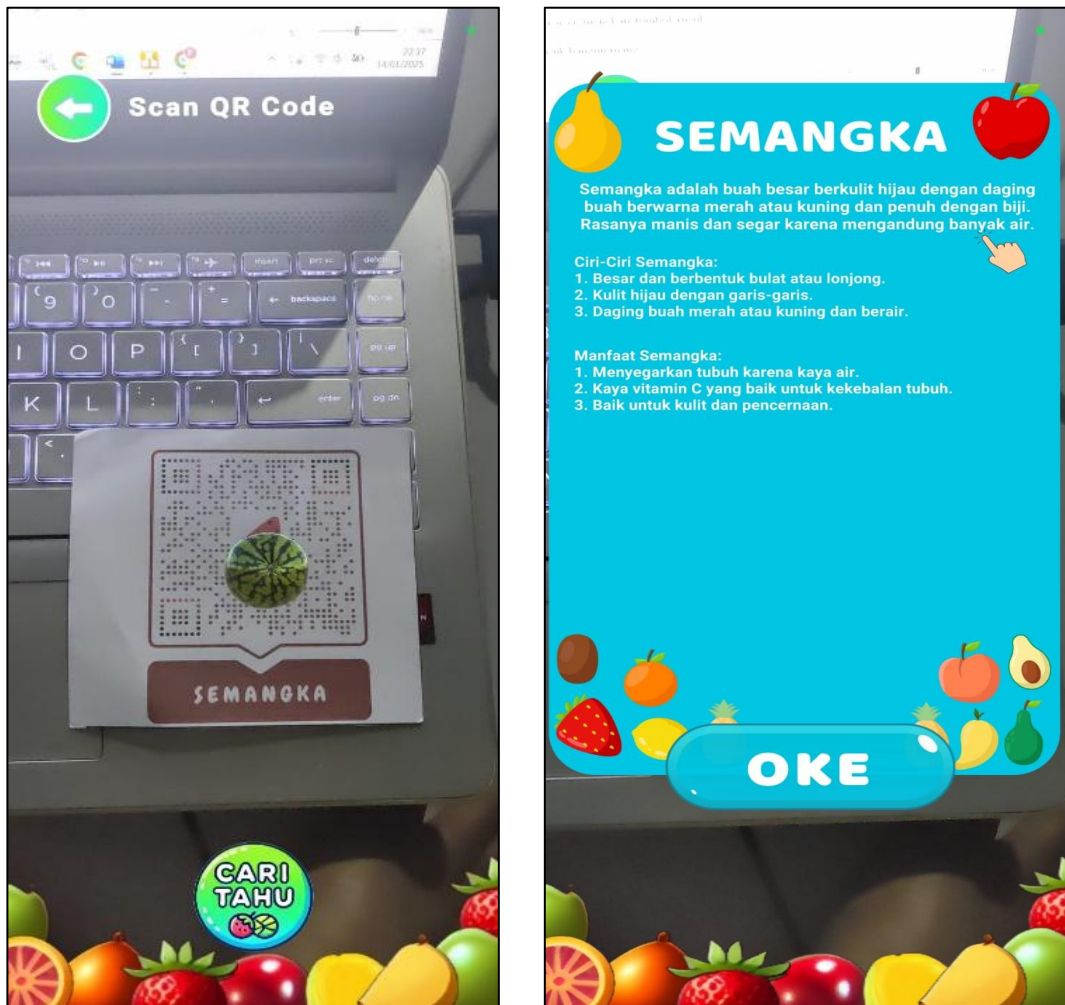
5. Tampilan Awal *Scan Marker Qr Code*

Tampilan awal *Scan Qr Code* dapat melihat objek pada target yang telah di *Scan* dengan menyorot *marker* berupa gambar Buah dan Sayuran. Namun pada Buah dan Sayuran perlu digaris bawahi bahwa tidak semua Buah dan Sayuran dapat di *scan* dan tampil objek 3D nya, harus ada proses tambahan yang diperlukan oleh peneliti agar benda dapat di *scan*.



Gambar 4. 12 Tampilan Awal *Scan*

6. Tampilan *Scan Qr Code*



Gambar 4. 13 Tampilan *Scan Qr Code*

Masih di tampilan awal *Scan Qr Code*, ketika *user* meletakkan *Marker* Buah dan Sayur maka akan muncul gambar buah atau sayur dan ketika di tekan kata *Cari tahu* maka objek target akan memberikan penjelasan mengenai *marker* gambar buah atau sayur yang dipilih dengan diberikan penjelasan berupa tulisan dan juga menggunakan suara.

Tabel 4. 5 *Script popup Controller Scan Qr Code*

```
using UnityEngine;
public class PopupController : MonoBehaviour
{
```

```

private Animator animator;
private bool canShowPopup = true;
void Start()
{
    animator = GetComponent<Animator>();
}
public void ShowPopup()
{
    if (canShowPopup)
    {
        animator.SetTrigger("Show");
        canShowPopup = false;
    }
}
public void HidePopup()
{
    animator.SetTrigger("Hide");
}
// Panggil fungsi ini di akhir animasi "Hide"
public void EnableShow()
{
    canShowPopup = true;
}
}

```

Tabel 4. 6 *Script popup Controller Scan Qr Code*

```

using UnityEngine;
using UnityEngine.UI;
using Vuforia;
public class ScanController : MonoBehaviour
{
    public GameObject ObjekPertama;
    public GameObject deskripsiPopup;
    public GameObject btnCariTahu;
    private AudioSource audioSource; // AudioSource
    sekarang menjadi variabel private
    private AudioSource backgroundMusic; //
    Menambahkan variabel untuk background music
    private ObserverBehaviour observerBehaviour;
    private float originalVolume; // Menyimpan
    volume asli background music
    void Start()
    {
        if (ObjekPertama != null)
        {
            ObjekPertama.SetActive(false); // Mulai
            dengan objek tersembunyi
        }
        if (deskripsiPopup != null)
        {
            deskripsiPopup.SetActive(false);
        }
        if (btnCariTahu != null)
        {

```

```

        btnCariTahu.SetActive(false);
    }
    // Inisialisasi ObserverBehaviour dan
    daftarkan event handler
        observerBehaviour =
GetComponent<ObserverBehaviour>();
    if (observerBehaviour != null)
    {
        observerBehaviour.OnTargetStatusChanged
+= OnTargetStatusChanged;
    }
    // Temukan AudioSource untuk background
music
        backgroundMusic =
GameObject.Find("backsoundController").GetComponen
t<AudioSource>(); // Ganti "BackgroundMusic" dengan
nama GameObject yang sesuai
    if (backgroundMusic != null)
    {
        originalVolume = backgroundMusic.volume;
// Menyimpan volume asli
        backgroundMusic.Play(); // Memutar
background music
    }
}
void OnDestroy()
{
    if (observerBehaviour != null)
    {
        observerBehaviour.OnTargetStatusChanged
-= OnTargetStatusChanged;
    }
}
// Fungsi yang dipanggil ketika tombol deskripsi
diklik
public void ShowDeskripsiPopup()
{
    if (deskripsiPopup != null)
    {
        deskripsiPopup.SetActive(true);
    }
}
// Fungsi yang dipanggil ketika objek terdeteksi
public void OnObjectScanned()
{
    if (ObjekPertama != null)
    {
        ObjekPertama.SetActive(true);
    }
    if (btnCariTahu != null)
    {
        btnCariTahu.SetActive(true);
    }
    if (deskripsiPopup != null)
    {
        deskripsiPopup.SetActive(false);
    }
}

```

```

    }
    // Temukan AudioSource di dalam Image Target
    if (observerBehaviour != null)
    {
        AudioSource =
observerBehaviour.GetComponentInChildren<AudioSource>();
        if (audioSource != null &&
!audioSource.isPlaying)
        {
            // Mengecilkan volume background
music
            if (backgroundMusic != null)
            {
                backgroundMusic.volume =
originalVolume * 0f;
            }
            audioSource.Play();
            StartCoroutine(ResetVolumeAfterSou
nd());
        }
    }
}
// Fungsi yang dipanggil ketika objek hilang
public void OnObjectLost()
{
    if (ObjekPertama != null)
    {
        ObjekPertama.SetActive(false);
    }
    if (btnCariTahu != null)
    {
        btnCariTahu.SetActive(false);
    }
    // Hentikan suara saat objek hilang
    if (audioSource != null &&
audioSource.isPlaying)
    {
        audioSource.Stop();
    }
}
// Event handler untuk perubahan status target
private void
OnTargetStatusChanged(ObserverBehaviour behaviour,
TargetStatus status)
{
    if (status.Status == Status.TRACKED ||
        status.Status ==
Status.EXTENDED_TRACKED)
    {
        // Objek ditemukan
        OnObjectScanned();
    }
    else
    {
        // Objek hilang

```

```

        OnObjectLost();
    }
}

private System.Collections.IEnumerator
ResetVolumeAfterSound()
{
    // Tunggu sampai audio source selesai
    diputar
    yield return new
    WaitForSeconds(audioSource.clip.length);
    // Mengembalikan volume background music ke
    nilai asli
    if (backgroundMusic != null)
    {
        backgroundMusic.volume = originalVolume;
    }
}
}

```

Tabel 4. 7 *Script Rotate object Touch*

```

using UnityEngine;
public class RotateObjectTouch : MonoBehaviour
{
    public Transform rotationAxis; // Poros untuk rotasi
    private Vector2 initialTouchPosition;
    private Vector3 initialRotation;
    private bool isTouching = false;
    private bool canRotate = true; // Flag untuk mengizinkan
    rotasi
    private bool isMoving = false; // Flag untuk memeriksa jika
    objek sedang dipindah
    private bool touchStartedOnObject = false; // Flag untuk
    memeriksa jika sentuhan dimulai di objek
    // Kecepatan rotasi, sesuaikan sesuai kebutuhan Anda
    public float rotationSpeed = 0.01f;
    private Vector3 originalObjectPosition; // Posisi awal objek
    yang disimpan saat pertama kali dibuka
    private Quaternion originalObjectRotation; // Rotasi awal
    objek yang disimpan saat pertama kali dibuka
    private Vector3 originalObjectScale; // Ukuran awal objek
    yang disimpan saat pertama kali dibuka
    public GameObject centerPoint; // GameObject yang menjadi
    titik pusat
    void Start()
    {
        // Simpan posisi dan rotasi awal objek
        originalObjectPosition = transform.position;
        originalObjectRotation = transform.rotation;
        originalObjectScale = transform.localScale;
    }
    void Update()
    {
        if (Input.touchCount == 1 && !isMoving)
        {

```

```

        Touch touch = Input.GetTouch(0);
        Ray ray =
Camera.main.ScreenPointToRay(touch.position);
        RaycastHit hit;
        if (touch.phase == TouchPhase.Began)
        {
            if (Physics.Raycast(ray, out hit) &&
hit.transform == transform)
            {
                initialTouchPosition = touch.position;
                initialRotation =
transform.rotation.eulerAngles;
                isTouching = true;
                touchStartedOnObject = true;
            }
        }
        else if (touchStartedOnObject)
        {
            switch (touch.phase)
            {
                case TouchPhase.Moved:
                    if (isTouching && canRotate)
                    {
                        Vector2 touchDeltaPosition =
touch.deltaPosition;
                        float rotationX = -
touchDeltaPosition.x * rotationSpeed;
                        float rotationY = -
touchDeltaPosition.y * rotationSpeed;
                        // Rotasi di sekitar poros
                        transform.RotateAround(rotati
onAxis.position, Vector3.up, rotationX);
                        transform.RotateAround(rotati
onAxis.position, Vector3.right, -rotationY);
                    }
                    break;
                case TouchPhase.Ended:
                case TouchPhase.Canceled:
                    isTouching = false;
                    touchStartedOnObject = false;
                    break;
            }
        }
    }

    public void ResetToCenterPosition()
    {
        transform.position =
centerPoint.transform.position; // Atur ulang posisi objek
        ke titik pusat
        transform.rotation = originalObjectRotation; //
Gunakan rotasi awal yang disimpan saat pertama kali dibuka
        transform.localScale = originalObjectScale; //
Atur ulang ukuran objek ke ukuran awal
    }

```

```

        public void SetRotationAxis(Transform newAxis)
        {
            rotationAxis = newAxis;
        }
    }

```

Tabel 4. 8 *Script Voice Management qr code*

```

using System.Collections;
using UnityEngine;
using UnityEngine.UI;
using Vuforia;
public class voiceManagement : MonoBehaviour
{
    // Variabel untuk menyimpan suara
    public AudioClip pengertian;
    public AudioClip ciri;
    public AudioClip manfaat;
    private AudioSource audioSource;
    private AudioSource backgroundMusic;
    private float originalVolume;
    public Button btnPengertian;
    public Button btnCiri;
    public Button btnManfaat;
    void Start()
    {
        // Inisialisasi AudioSource
        audioSource = GetComponent();
        // Tambahkan event listener ke tombol
        btnPengertian.onClick.AddListener(() =>
PlayAudio(pengertian));
        btnCiri.onClick.AddListener(() => PlayAudio(ciri));
        btnManfaat.onClick.AddListener(() =>
PlayAudio(manfaat));
        // Cari dan inisialisasi background
music backgroundMusic =
GameObject.Find("backsoundController").GetComponent<AudioSource
>();
        if (backgroundMusic != null)
        {
            originalVolume = backgroundMusic.volume; // Menyimpan
volume asli
            backgroundMusic.Play(); // Memutar background music
        }
    }
    void PlayAudio(AudioClip clip)
    {
        if (audioSource == null || backgroundMusic == null ||
clip == null)
            return;
        // Berhentikan suara yang sedang diputar di audioSource
audioSource.Stop();
        // Mengecilkan volume background music
        backgroundMusic.volume = originalVolume * 0f; //
Sesuaikan angka untuk tingkat pengecilan
    }
}

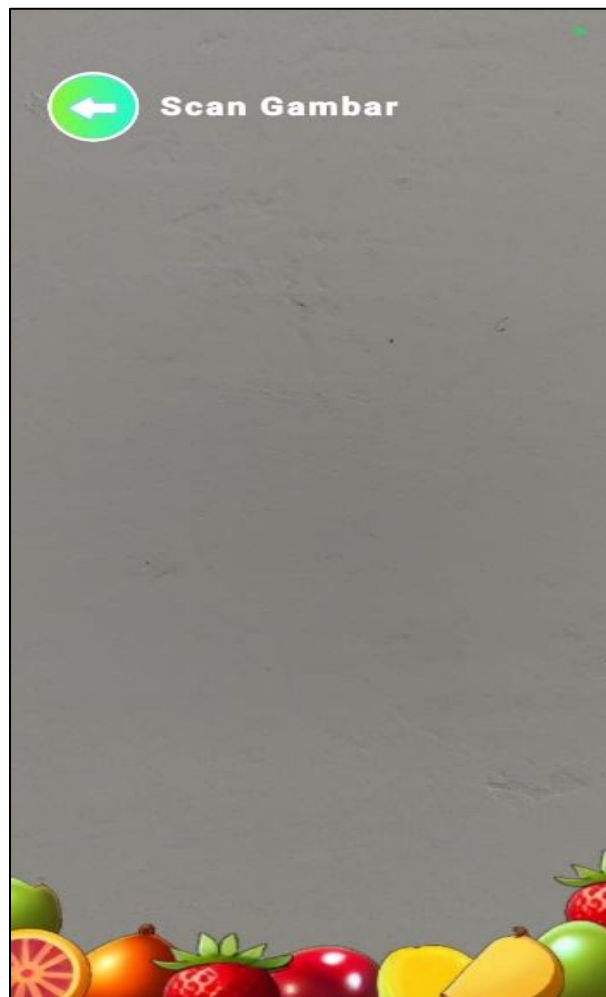
```

```

        // Mainkan suara baru di audioSource
        audioSource.clip = clip;
        audioSource.Play();
        // Kembalikan volume background music setelah suara
        selesai
        StartCoroutine(ResetVolumeAfterSound(audioSource.clip.length));
    }
    private IEnumerator ResetVolumeAfterSound(float delay)
    {
        yield return new WaitForSeconds(delay);
        // Kembalikan volume background music ke nilai asli
        if (backgroundMusic != null)
        {
            backgroundMusic.volume = originalVolume;
        }
    }
}

```

7. Tampilan Awal *Scan* Gambar

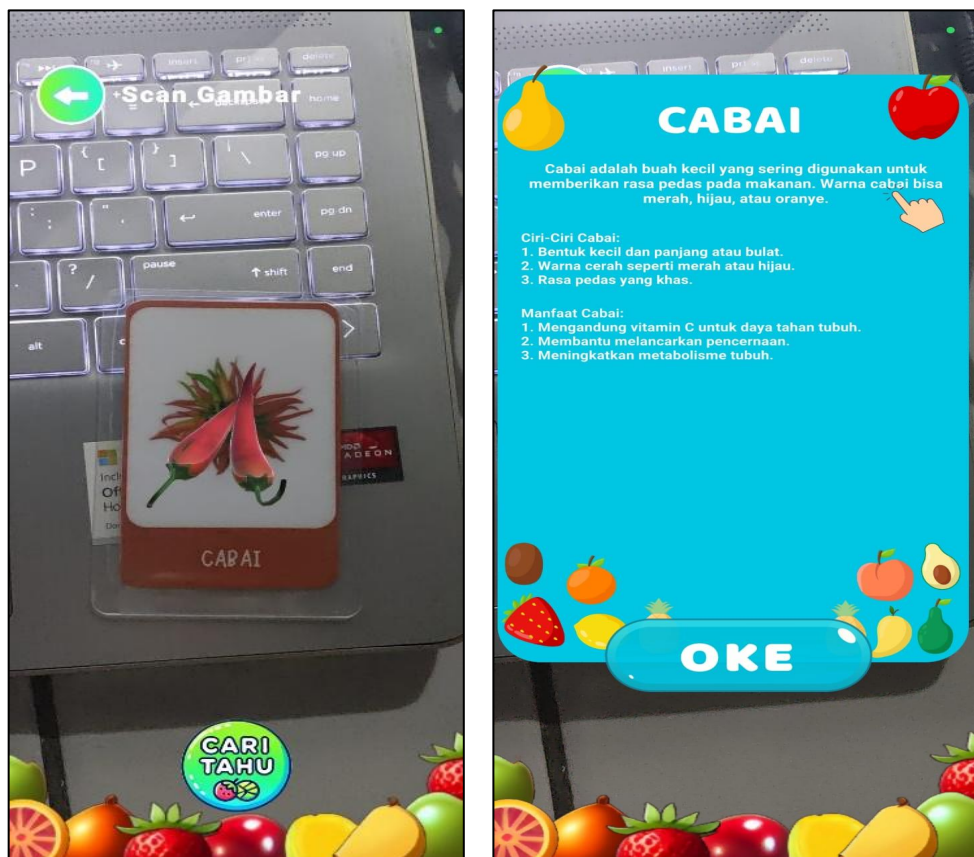


Gambar 4. 14 Tampilan Awal *Scan* Gambar

Tampilan awal *Scan Gambar* dapat melihat objek pada target yang telah di *Scan* dengan menyorot *marker* berupa gambar Buah dan Sayuran. Namun pada Buah dan Sayuran perlu digaris bawahi bahwa tidak semua Buah dan Sayuran dapat di *scan* dan tampil objek 3D nya, harus ada proses tambahan yang diperlukan oleh peneliti agar benda dapat di *scan*.

8. Tampilan *Scan* gambar

Masih di tampilan awal *Scan* gambar, ketika *user* meletakkan *Marker* Buah dan Sayur maka akan muncul gambar buah atau sayur dan ketika di tekan kata Cari tahu maka objek target akan memberikan penjelasan mengenai *marker* gambar buah atau sayur yang dipilih dengan diberikan penjelasan berupa tulisan dan juga menggunakan suara.



Gambar 4. 15 Tampilan *Scan* gambar

Tabel 4. 9 *Script popup Controller Gambar*

```

using UnityEngine;
public class PopupController : MonoBehaviour
{
    private Animator animator;
    private bool canShowPopup = true;
    void Start()
    {
        animator = GetComponent<Animator>();
    }
    public void ShowPopup()
    {
        if (canShowPopup)
        {
            animator.SetTrigger("Show");
            canShowPopup = false;
        }
    }
    public void HidePopup()
    {
        animator.SetTrigger("Hide");
    }
    // Panggil fungsi ini di akhir animasi "Hide"
    public void EnableShow()
    {
        canShowPopup = true;
    }
}

```

Tabel 4. 10 *Script Scan Controller Gambar*

```

using UnityEngine;
using UnityEngine.UI;
using Vuforia;
public class ScanController : MonoBehaviour
{
    public GameObject ObjekPertama;
    public GameObject deskripsiPopup;
    public GameObject btnCariTahu;
    private AudioSource audioSource; // AudioSource sekarang
    menjadi variabel private
    private AudioSource backgroundMusic; // Menambahkan
    variabel untuk background music
    private ObserverBehaviour observerBehaviour;
    private float originalVolume; // Menyimpan volume asli
    background music
    void Start()
    {
        if (ObjekPertama != null)
        {
            ObjekPertama.SetActive(false); // Mulai dengan
            objek tersembunyi
        }
        if (deskripsiPopup != null)

```

```

        {
            deskripsiPopup.SetActive(false);
        }
        if (btnCariTahu != null)
        {
            btnCariTahu.SetActive(false);
        }
        // Inisialisasi ObserverBehaviour dan daftarkan
event handler
        observerBehaviour =
GetComponent<ObserverBehaviour>();
        if (observerBehaviour != null)
        {
            observerBehaviour.OnTargetStatusChanged +=
OnTargetStatusChanged;
        }
        // Temukan AudioSource untuk background music
        backgroundMusic =
GameObject.Find("backgroundController").GetComponent<AudioSou
rce>(); // Ganti "BackgroundMusic" dengan nama GameObject
yang sesuai
        if (backgroundMusic != null)
        {
            originalVolume = backgroundMusic.volume; //
Menyimpan volume asli
            backgroundMusic.Play(); // Memutar background
music
        }
    }
    void OnDestroy()
    {
        if (observerBehaviour != null)
        {
            observerBehaviour.OnTargetStatusChanged -=
OnTargetStatusChanged;
        }
    }
    // Fungsi yang dipanggil ketika tombol deskripsi diklik
    public void ShowDeskripsiPopup()
    {
        if (deskripsiPopup != null)
        {
            deskripsiPopup.SetActive(true);
        }
    }
    // Fungsi yang dipanggil ketika objek terdeteksi
    public void OnObjectScanned()
    {
        if (ObjekPertama != null)
        {
            ObjekPertama.SetActive(true);
        }
        if (btnCariTahu != null)
        {
            btnCariTahu.SetActive(true);
        }
    }

```

```

        if (deskripsiPopup != null)
        {
            deskripsiPopup.SetActive(false);
        }
        // Temukan AudioSource di dalam Image Target
        if (observerBehaviour != null)
        {
            audioSource =
observerBehaviour.GetComponentInChildren<AudioSource>();
            if (audioSource != null &&
!audioSource.isPlaying)
            {
                // Mengecilkan volume background music
                if (backgroundMusic != null)
                {
                    backgroundMusic.volume = originalVolume
* 0f;
                }
                audioSource.Play();
                StartCoroutine(ResetVolumeAfterSound());
            }
        }
    }
    // Fungsi yang dipanggil ketika objek hilang
    public void OnObjectLost()
    {
        if (ObjekPertama != null)
        {
            ObjekPertama.SetActive(false);
        }
        if (btnCariTahu != null)
        {
            btnCariTahu.SetActive(false);
        }
        // Hentikan suara saat objek hilang
        if (audioSource != null && audioSource.isPlaying)
        {
            audioSource.Stop();
        }
    }
    // Event handler untuk perubahan status target
    private void OnTargetStatusChanged(ObserverBehaviour
behaviour, TargetStatus status)
    {
        if (status.Status == Status.TRACKED ||
            status.Status == Status.EXTENDED_TRACKED)
        {
            // Objek ditemukan
            OnObjectScanned();
        }
        else
        {
            // Objek hilang
            OnObjectLost();
        }
    }
}

```

```

        private System.Collections.IEnumerator
ResetVolumeAfterSound()
    {
        // Tunggu sampai audio source selesai diputar
        yield return new
WaitForSeconds(audioSource.clip.length);
        // Mengembalikan volume background music ke nilai
asli
        if (backgroundMusic != null)
        {
            backgroundMusic.volume = originalVolume;
        }
    }
}

```

Tabel 4. 11 *Script Voice Management Gambar*

```

using System.Collections;
using UnityEngine;
using UnityEngine.UI;
using Vuforia;
public class voiceManagement : MonoBehaviour
{
    // Variabel untuk menyimpan suara
    public AudioClip pengertian;
    public AudioClip ciri;
    public AudioClip manfaat;
    private AudioSource audioSource;
    private AudioSource backgroundMusic;
    private float originalVolume;
    public Button btnPengertian;
    public Button btnCiri;
    public Button btnManfaat;
    void Start()
    {
        // Inisialisasi AudioSource
        audioSource = GetComponent<AudioSource>();
        // Tambahkan event listener ke tombol
        btnPengertian.onClick.AddListener(() =>
PlayAudio(pengertian));
        btnCiri.onClick.AddListener(() => PlayAudio(ciri));
        btnManfaat.onClick.AddListener(() =>
PlayAudio(manfaat));
        // Cari dan inisialisasi background music
        backgroundMusic =
GameObject.Find("backgroundController").GetComponent<AudioSou
rce>();
        if (backgroundMusic != null)
        {
            originalVolume = backgroundMusic.volume; //
Menyimpan volume asli
            backgroundMusic.Play(); // Memutar background
music
        }
    }
}

```

```

void PlayAudio(AudioClip clip)
{
    if (audioSource == null || backgroundMusic == null
|| clip == null)
        return;
    // Berhentikan suara yang sedang diputar di
audioSource
    audioSource.Stop();
    // Mengecilkan volume background music
    backgroundMusic.volume = originalVolume * 0f; //
Sesuaikan angka untuk tingkat pengecilan
    // Mainkan suara baru di audioSource
    audioSource.clip = clip;
    audioSource.Play();
    // Kembalikan volume background music setelah suara
selesai
    StartCoroutine(ResetVolumeAfterSound(audioSource.cli
p.length));
}
private IEnumerator ResetVolumeAfterSound(float delay)
{
    yield return new WaitForSeconds(delay);
    // Kembalikan volume background music ke nilai asli
    if (backgroundMusic != null)
    {
        backgroundMusic.volume = originalVolume;
    }
}
}

```

Tabel 4. 12 *Script Rotate Object Touch Gambar*

```

using UnityEngine;
public class RotateObjectTouch : MonoBehaviour
{
    public Transform rotationAxis; // Poros untuk rotasi
    private Vector2 initialTouchPosition;
    private Vector3 initialRotation;
    private bool isTouching = false;
    private bool canRotate = true; // Flag untuk mengizinkan
rotasi
    private bool isMoving = false; // Flag untuk memeriksa jika
objek sedang dipindah
    private bool touchStartedOnObject = false; // Flag untuk
memeriksa jika sentuhan dimulai di objek
    // Kecepatan rotasi, sesuaikan sesuai kebutuhan Anda
    public float rotationSpeed = 0.01f;
    private Vector3 originalObjectPosition; // Posisi awal objek
yang disimpan saat pertama kali dibuka
    private Quaternion originalObjectRotation; // Rotasi awal
objek yang disimpan saat pertama kali dibuka
    private Vector3 originalObjectScale; // Ukuran awal objek
yang disimpan saat pertama kali dibuka
    public GameObject centerPoint; // GameObject yang menjadi
titik pusat

```

```

void Start()
{
    // Simpan posisi dan rotasi awal objek
    originalObjectPosition = transform.position;
    originalObjectRotation = transform.rotation;
    originalObjectScale = transform.localScale;
}
void Update()
{
    if (Input.touchCount == 1 && !isMoving)
    {
        Touch touch = Input.GetTouch(0);
        Ray ray =
Camera.main.ScreenPointToRay(touch.position);
        RaycastHit hit;
        if (touch.phase == TouchPhase.Began)
        {
            if (Physics.Raycast(ray, out hit) &&
hit.transform == transform)
            {
                initialTouchPosition = touch.position;
                initialRotation =
transform.rotation.eulerAngles;
                isTouching = true;
                touchStartedOnObject = true;
            }
        }
        else if (touchStartedOnObject)
        {
            switch (touch.phase)
            {
                case TouchPhase.Moved:
                    if (isTouching && canRotate)
                    {
                        Vector2 touchDeltaPosition =
touch.deltaPosition;
                        float rotationX = -
touchDeltaPosition.x * rotationSpeed;
                        float rotationY = -
touchDeltaPosition.y * rotationSpeed;
                        // Rotasi di sekitar poros rotasi
                        transform.RotateAround(rotationAxis.
position, Vector3.up, rotationX);
                        transform.RotateAround(rotationAxis.
position, Vector3.right, -rotationY);
                    }
                    break;
                case TouchPhase.Ended:
                case TouchPhase.Canceled:
                    isTouching = false;
                    touchStartedOnObject = false;
                    break;
            }
        }
    }
}

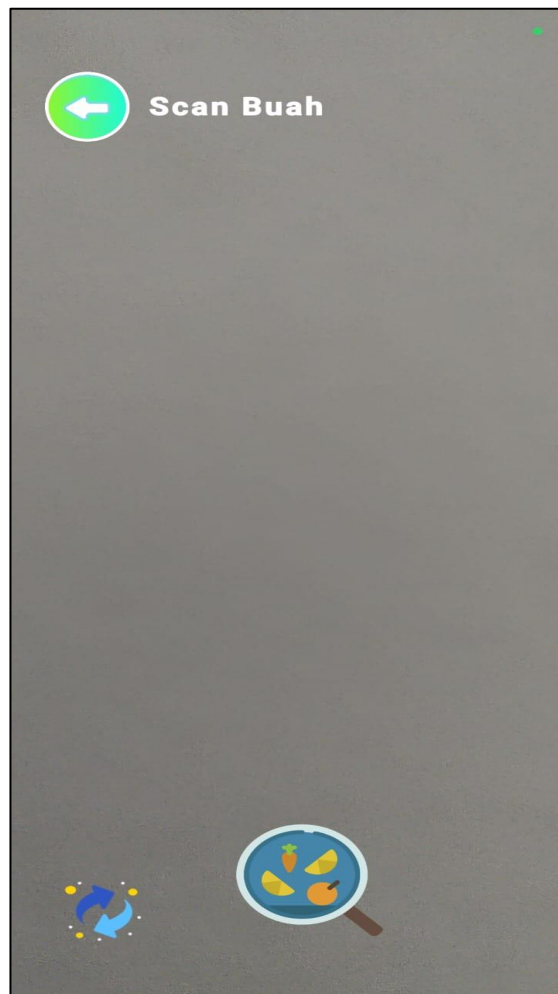
```

```

public void ResetToCenterPosition()
{
    transform.position = centerPoint.transform.position; //
    Atur ulang posisi objek ke titik pusat
    transform.rotation = originalObjectRotation; // Gunakan
    rotasi awal yang disimpan saat pertama kali dibuka
    transform.localScale = originalObjectScale; // Atur
    ulang ukuran objek ke ukuran awal
}
public void SetRotationAxis(Transform newAxis)
{
    rotationAxis = newAxis;
}
}

```

9. Tampilan Awal *Scan* Buah Asli

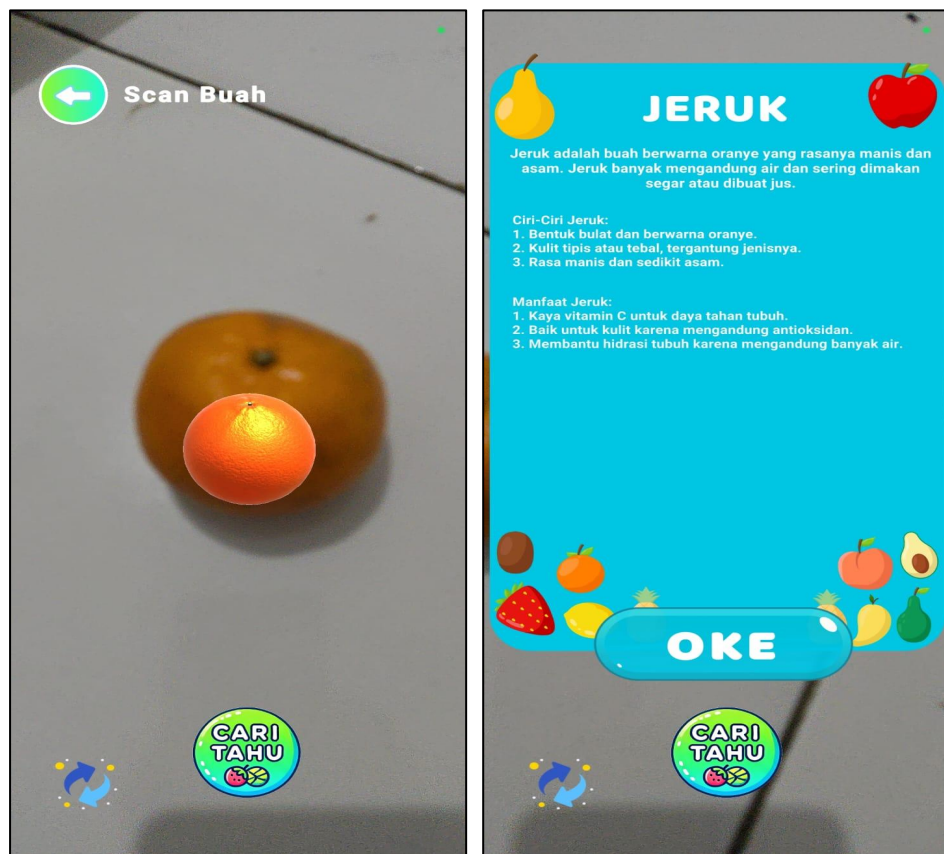


Gambar 4. 16 Tampilan Awal *Scan* Buah Asli

Tampilan awal *Scan Buah* asli dapat melihat objek pada target yang telah di *Scan* dengan menyorot buah asli. Namun pada Buah dan Sayuran perlu digaris bawahi bahwa tidak semua Buah dan Sayuran dapat di *scan* dan tampil objek 3D nya, harus ada proses tambahan yang diperlukan oleh peneliti agar buah dan sayuran asli dapat di *scan*.

10. Tampilan *Scan Buah Asli*

Masih di tampilan awal *Scan Buah Asli*, ketika *user* meletakkan *Marker Buah dan Sayur* maka akan muncul gambar buah atau sayur dan ketika di tekan kata *Cari tahu* maka objek target akan memberikan penjelasan mengenai *marker* gambar buah atau sayur yang dipilih dengan diberikan penjelasan berupa tulisan dan juga menggunakan suara.



Gambar 4. 17 Tampilan *Scan Buah Asli*

11. Tampilan Latihan

Selanjutnya pada *menu* latihan *user* akan memilih gambar buah atau sayuran sesuai dengan gambar yang menjadi pertanyaan dengan menekan tombol pilihan jawaban yang benar dengan terdapat 4 pilihan jawaban dan hanya satu yang benar dan setelah selesai dari 15 soal yang diberikan maka akan dikumpulkan hasil dari tebakan berapa soal yang benar dengan perhitungan, jumlah soal yang benar dibagi 15 dikali 100.



Gambar 4. 18 Latihan

Tabel 4. 13 *Script Nilai*

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using TMPro;
public class nilai : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {
        PlayerPrefs.SetInt("nilai", 0);
    }
    // Update is called once per frame
    void Update()
    {
        GetComponent<TextMeshProUGUI>().text =
        PlayerPrefs.GetInt("nilai").ToString();
    }
}

```

Tabel 4. 14 *Script Jawab*

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
public class jawab : MonoBehaviour
{
    public GameObject feed_benar, feed_salah;
    void Start()
    {
    }
    public void jawaban(bool jawab)
    {
        if (jawab){
            feed_benar.SetActive(false);
            feed_benar.SetActive(true);
            int nilai = PlayerPrefs.GetInt("nilai")+1;
            PlayerPrefs.SetInt("nilai", nilai);
        }else{
            feed_salah.SetActive(false);
            feed_salah.SetActive(true);
        }
        gameObject.SetActive(false);
        transform.parent.GetChild(gameObject.transform.GetSiblingIndex()+1).gameObject.SetActive(true);
    }
    // Update is called once per frame
    void Update()
    {
    }
}

```

Tabel 4. 15 *Script Nilai Akhir*

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using TMPro;
public class NilaiAkhir : MonoBehaviour
{
    public int jumlahSoal = 15; // Jumlah soal
    public int jawabanBenar;    // Jumlah jawaban benar
    private int nilaiAkhir;     // Nilai akhir
    // Start is called before the first frame update
    void Start()
    {
        PlayerPrefs.SetInt("jawabanBenar", 0);
        PlayerPrefs.SetInt("nilaiAkhir", 0);
    }
    // Update is called once per frame
    void Update()
    {
        jawabanBenar = PlayerPrefs.GetInt("nilai");
        nilaiAkhir = (jawabanBenar * 100) / jumlahSoal;
        PlayerPrefs.SetInt("nilaiAkhir", nilaiAkhir);
        GetComponent<TextMeshProUGUI>().text =
        PlayerPrefs.GetInt("nilaiAkhir").ToString();
    }
}
```

1. *Popup* Konfirmasi Keluar

Terakhir pada tampilan awal terdapat juga tombol keluar yang berguna untuk keluar dari aplikasi AR Pengenalan Buah dan Sayur



Gambar 4. 19 *Popup* Konfirmasi Keluar

Tabel 4. 16 *Script Popup Controller Keluar*

```
using UnityEngine;
public class PopupController : MonoBehaviour
{
    private Animator animator;
    private bool canShowPopup = true;
    void Start()
    {
        animator = GetComponent<Animator>();
    }
    public void ShowPopup()
    {
        if (canShowPopup)
        {
```

```

        animator.SetTrigger("Show");
        canShowPopup = false;
    }
}
public void HidePopup()
{
    animator.SetTrigger("Hide");
}
// Panggil fungsi ini di akhir animasi "Hide"
public void EnableShow()
{
    canShowPopup = true;
}
}

```

Tabel 4. 17 *Script Menu Keluar*

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;
public class menu : MonoBehaviour
{
    public void ScanButton(string scenename)
    {
        SceneManager.LoadScene(scenename);
    }
    public void BackButton(string scenename)
    {
        SceneManager.LoadScene(scenename);
    }
    public void QuitButton()
    {
        Application.Quit();
        Debug.Log("ANJAY");
    }
}

```

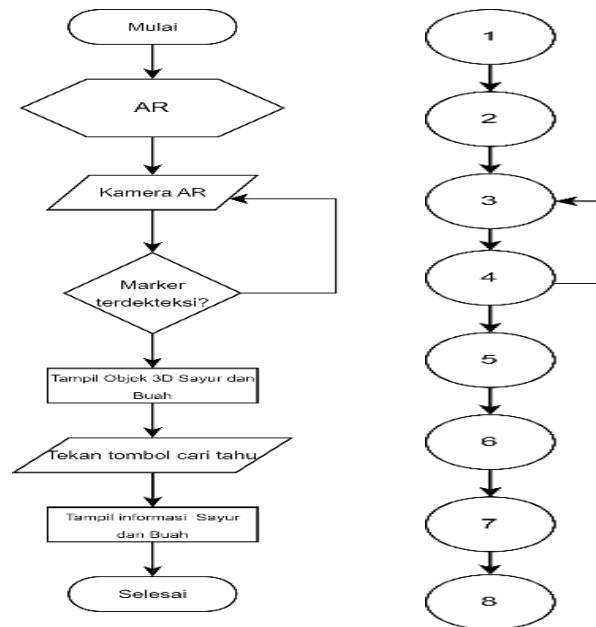
C. Pengujian Sistem

Pengujian pada sistem ini menggunakan pengujian *Black box*, *White box* dan uji tambahan. Adapun rincian pengujian sistem dapat dilihat sebagai berikut.

1. *White box Testing*

Pengujian sistem menggunakan *White box Testing* yang menggunakan *flowchart* dan *flowgraph* dari sistem. Adapun di bawah adalah hasil pengujian sistem menggunakan *White box*.

a. Pengujian *Flowchart* dan *Flowgraph Scan Qr code* dan *Scan gambar*



Gambar 4. 20 Flowchart dan Flowgraph Scan Qr code dan Scan gambar

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *egde* dan *node*

Pada rumus : $V(G) = E - N + 2$

E (*edge*) = 8

N (*node*) = 8

P (*Predikat node*) = 1

Penyelesaian :

$V(G) = E - N + 2$

$= 8 - 8 + 2$

$= 2$

$$\begin{aligned}\text{Predikat (P)} &= P + 1 \\ &= 1 + 1 \\ &= 2\end{aligned}$$

(2) Dari perhitungan *Cyclomatic Complexity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent Path* pada *flowgraph* tersebut yakni:

$$Path\ 1 = 1 - 2 - 3 - 4 - 3$$

$$Path\ 2 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

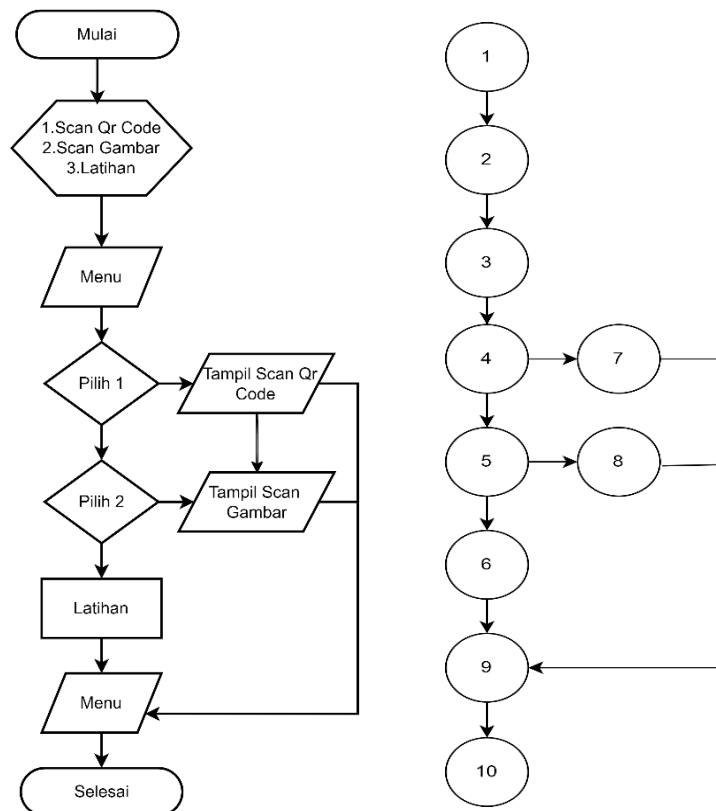
(4) *Grafik matriks Scan Qr code dan Scan gambar*

Tabel 4. 18 Grafik matriks *Scan Qr code* dan *Scan gambar*

	1	2	3	4	5	6	7	8	E - 1
1		1							$1 - 1 = 0$
2			1						$1 - 1 = 0$
3				1					$1 - 1 = 0$
4			1		1				$2 - 1 = 1$
5						1			$1 - 1 = 0$
6							1		$1 - 1 = 0$
7								1	$1 - 1 = 0$
SUM (E + 1)									$1 + 1 = 2$

b. Pengujian *Flowchart* dan *flowgraph* Menu

Dari *Flowchart* dan *flowgraph* di atas di lakukanlah perhitungan sebagai berikut:



Gambar 4. 21 *Flowchart* dan *Flowgraph* Menu

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 11$$

$$N \text{ (node)} = 10$$

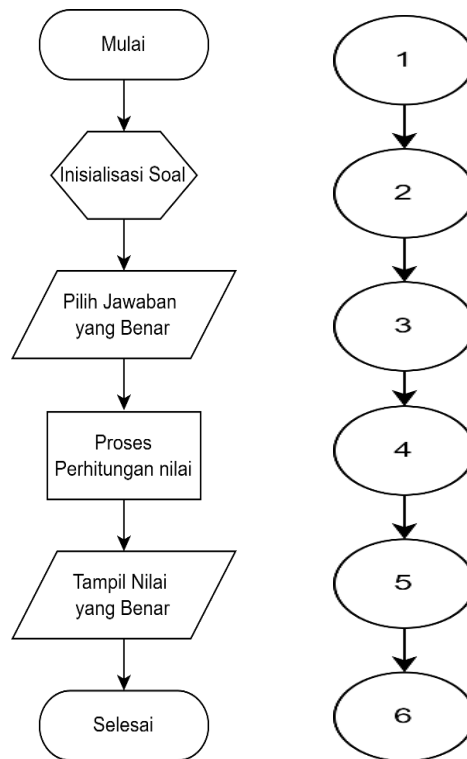
$$P \text{ (Predikat node)} = 2$$

Penyelesaian :

$$V(G) = E - N + 2$$

c. Pengujian *Flowchart* dan *flowgraph* Latihan

Dari *Flowchart* dan *flowgraph* di atas dilakukanlah



Gambar 4. 22 *Flowchart* dan *Flowgraph* Latihan

perhitungan sebagai sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *egde* dan *node*

Pada rumus : $V(G) = E - N + 2$

E (*edge*) = 5

N (*node*) = 6

P (*Predikat node*) = 0

Penyelesaian :

$V(G) = E - N + 2$

$$= 5 - 6 + 2$$

$$= 1$$

$$\text{Predikat (P)} = 2 + 1$$

$$= 0 + 1$$

$$= 1$$

(2) Dari perhitungan *Cyclomatic Complexity* dari *flowgraph* di atas memiliki *Region* = 1

(3) *Independent Path* pada *flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 5 - 6$$

(4) Grafik matriks Latihan

Tabel 4. 20 Grafik matriks Latihan

	1	2	3	4	5	6	E - 1
1		1					1-1=0
2			1				1-1=0
3				1			1-1=0
4					1		1-1=1
5						1	1-1=0
SUM (E + 1)							0+1=1

2. *Black box Testing*

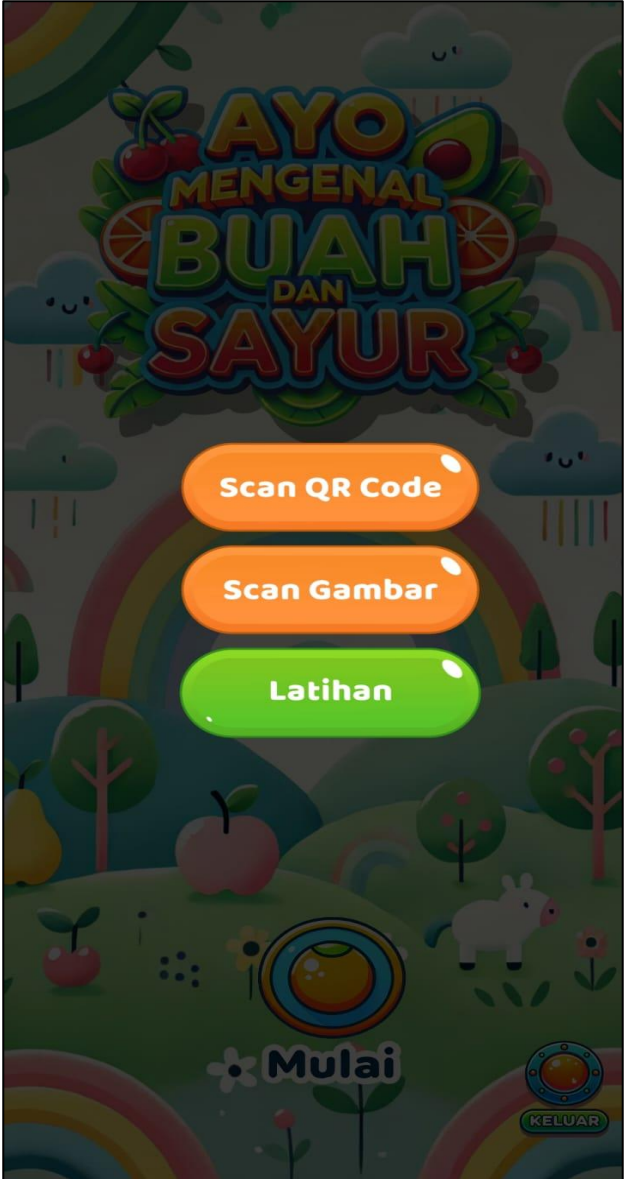
Pengujian yang dilakukan pada sistem yaitu pengujian *Black box* dimana peneliti menguji tampilan aplikasi apakah berhasil tampil atau tidak, adapun hasil dan penjelasan detailnya akan dijelaskan selanjutnya.

a. *Black box Testing* tampilan *home*

Tabel 4. 21 *Black box Testing* tampilan *home*

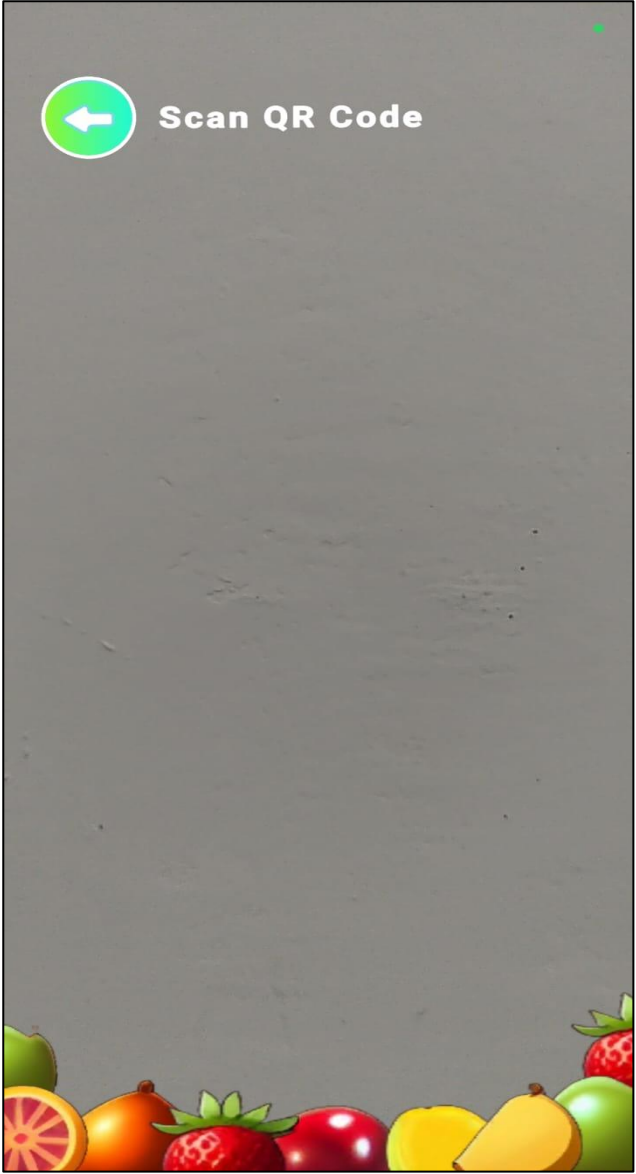
Tes Faktor	Hasil	Keterangan
User Membuka Aplikasi	✓	Berhasil Tampil Halaman <i>Home</i>
<i>Screenshot</i>		
		

b. *Black box Pop-up Menu***Tabel 4. 22** *Black box Pop-up menu*

Tes Faktor	Hasil	Keterangan
User Menekan Tombol mulai	✓	Berhasil Tampil <i>Pop-up</i> menu
Screenshot		
		

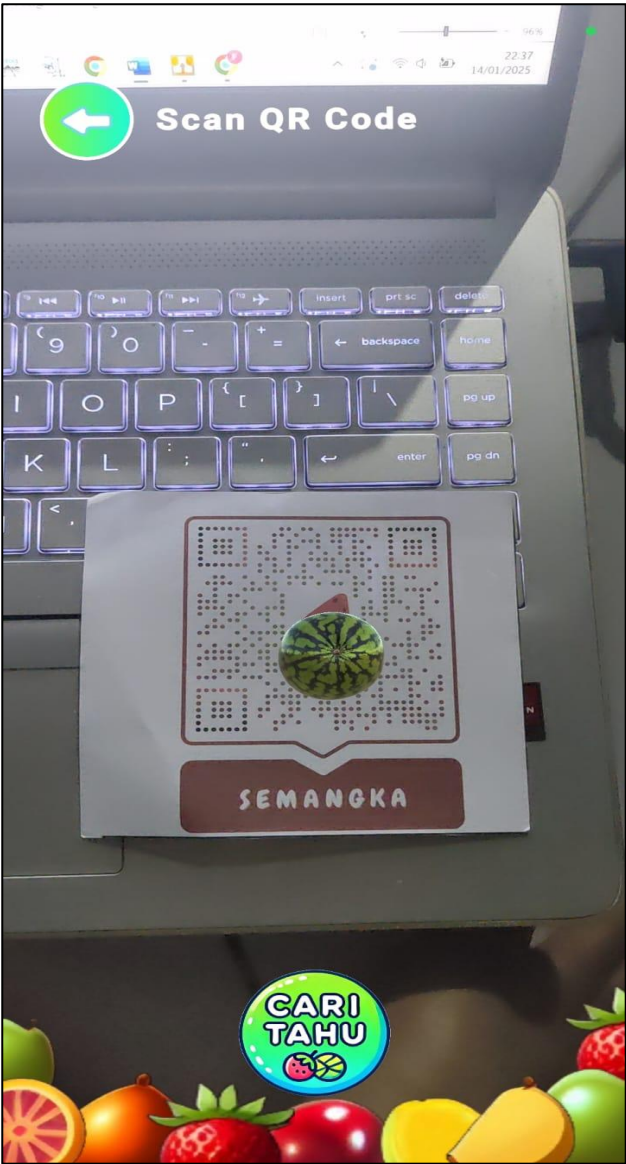
c. *Black box scene Scan Qr Code*

Tabel 4. 23 *Black box scene Scan qr code*

Tes Faktor	Hasil	Keterangan
<i>User Menekan Tombol Scan qr code</i>	✓	Berhasil Tampil <i>Scan qr code</i>
Screenshot		
		

d. *Black box Deteksi Qr Code*

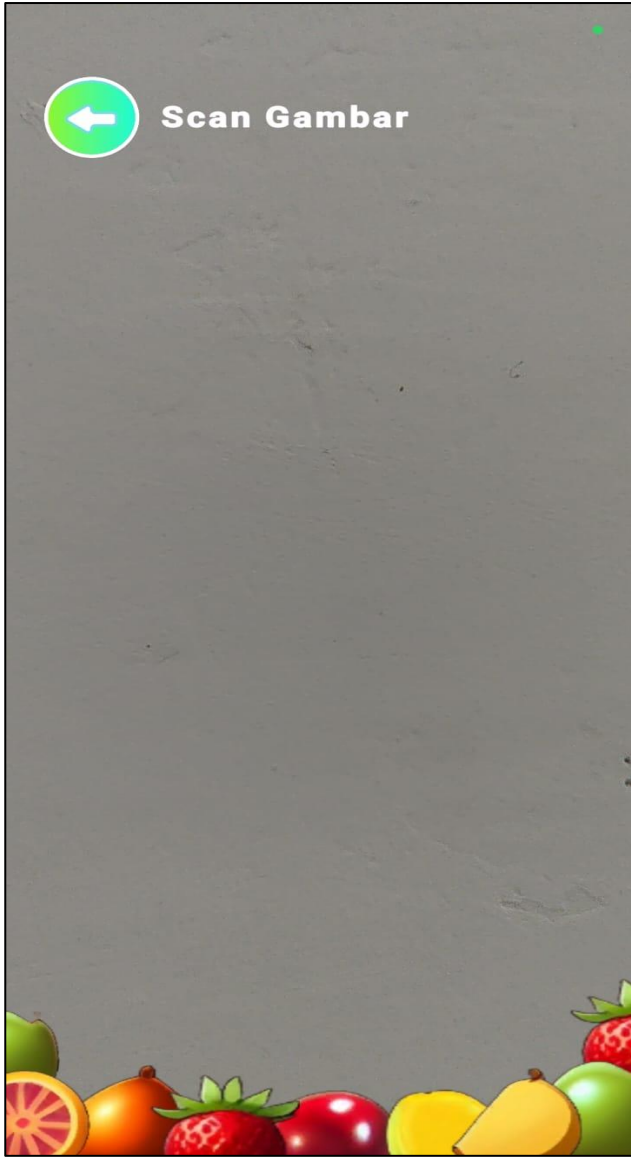
Tabel 4. 24 *Black box deteksi qr code*

Tes Faktor	Hasil	Keterangan
User Menyorot Qr Code	✓	berhasil tampil buah atau sayur secara 3d
Screenshot		
		

e. *Black box* Info Buah dan Sayur**Tabel 4. 25** *Black box* info buah dan sayur

Tes Faktor	Hasil	Keterangan
User Menekan Tombol cari tahu	✓	Berhasil Tampil info Buah atau sayur
Screenshot		
 SEMANGKA Semangka adalah buah besar berkulit hijau dengan daging buah berwarna merah atau kuning dan penuh dengan biji. Rasanya manis dan segar karena mengandung banyak air. Ciri-Ciri Semangka: 1. Besar dan berbentuk bulat atau lonjong. 2. Kulit hijau dengan garis-garis. 3. Daging buah merah atau kuning dan berair. Manfaat Semangka: 1. Menyegarkan tubuh karena kaya air. 2. Kaya vitamin C yang baik untuk kekebalan tubuh. 3. Baik untuk kulit dan pencernaan. OKE		

f. *Black box scene Scan Gambar***Tabel 4. 26** *Black box scene scan gambar*

Tes Faktor	Hasil	Keterangan
User Menekan Tombol <i>Scan Gambar</i>	✓	Berhasil Tampil <i>Scan gambar</i>
Screenshot		
		

g. *Black box Deteksi Gambar***Tabel 4. 27** *Black box deteksi gambar*

Tes Faktor	Hasil	Keterangan
<i>User Menyorot Gambar</i>	✓	berhasil tampil buah atau sayur secara 3d
Screenshot		

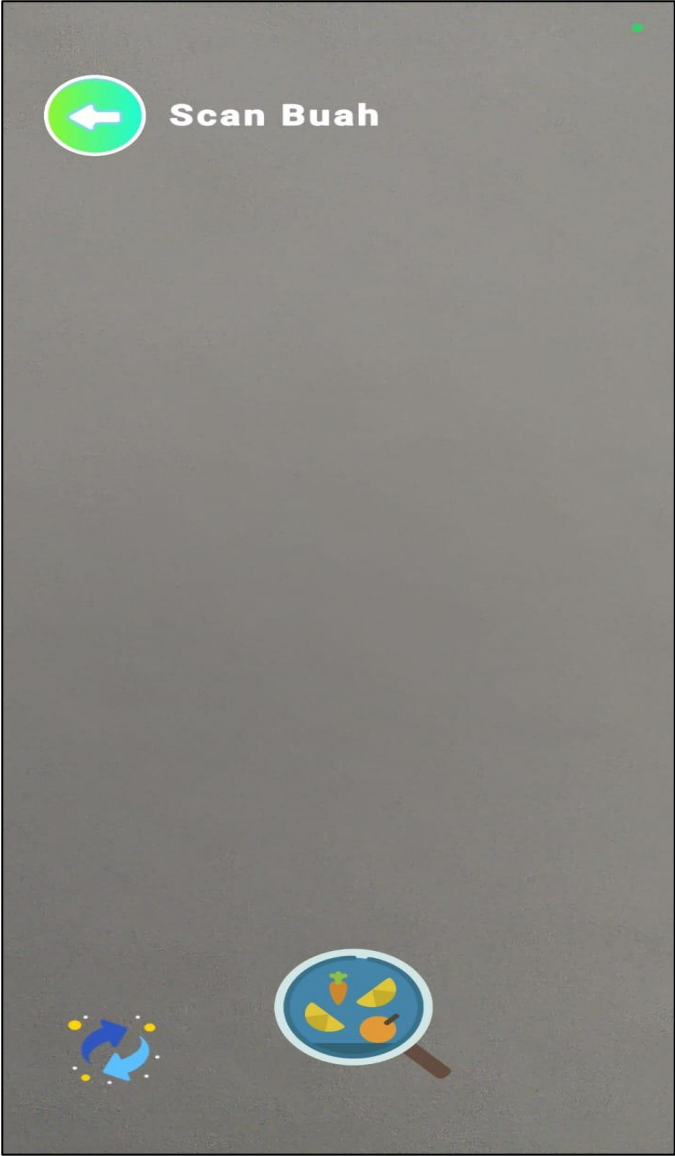


h. *Black box* Info Buah dan Sayur**Tabel 4. 28** *Black box* info buah dan sayur

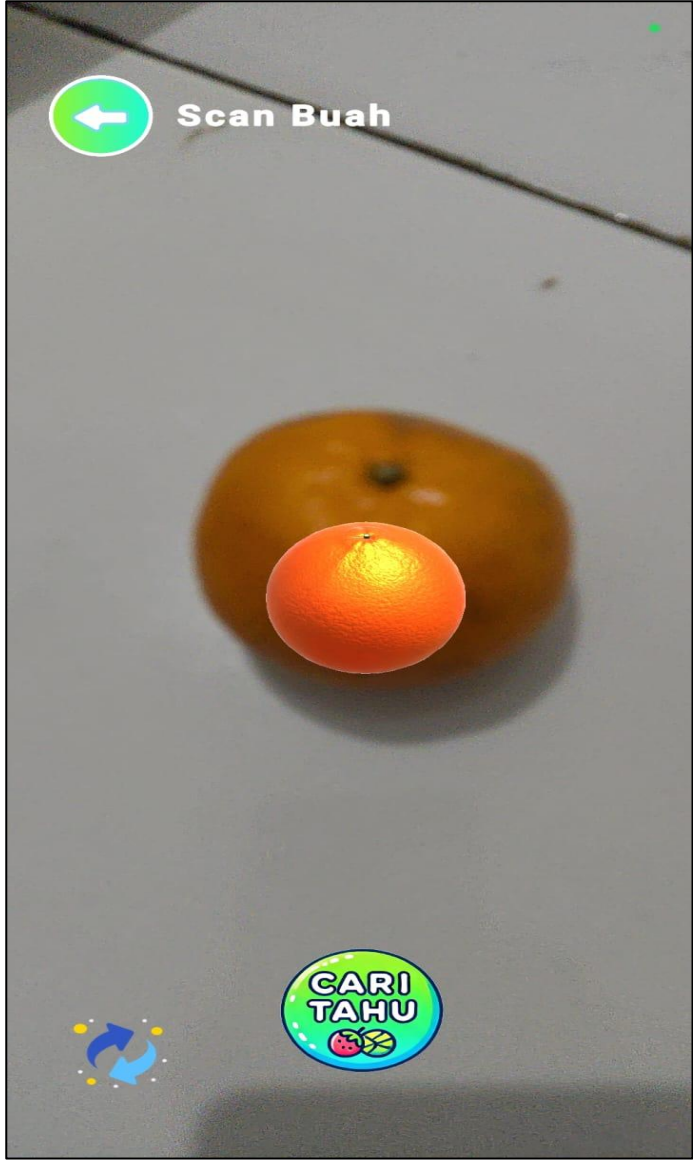
Tes Faktor	Hasil	Keterangan
User Menekan Tombol cari tahu	✓	Berhasil Tampil info Buah atau sayur
Screenshot		



i. *Black box scene Scan Buah Asli***Tabel 4. 29** *Black box scene scan Buah Asli*

Tes Faktor	Hasil	Keterangan
<i>User Menekan Tombol Scan Buah</i>	✓	Berhasil Tampil <i>Scan Buah</i>
Screenshot		
		

j. *Black box Deteksi Buah Asli***Tabel 4. 30** *Black box deteksi gambar*

Tes Faktor	Hasil	Keterangan
<i>User Menyorot Buah Asli</i>	✓	berhasil tampil buah secara 3d
Screenshot		
		

k. *Black box* Info Buah**Tabel 4. 31** *Black box* info buah dan sayur

Tes Faktor	Hasil	Keterangan
User Menekan Tombol cari tahu	✓	Berhasil Tampil info Buah
Screenshot		
		

1. *Black box Pilih jawaban Buah dan Sayur*

Tabel 4. 32 *Black box pilih jawaban buah dan sayur*

Tes Faktor	Hasil	Keterangan
User memilih jawaban yang benar	✓	Berhasil memilih jawaban yang benar
Screenshot		
		

m. *Black box* Nilai

Tabel 4. 33 *Black box* nilai

Tes Faktor	Hasil	Keterangan
User Menyelesaikan Latihan Buah dan Sayur	✓	Berhasil Tampil Nilai
Screenshot		
		

3. Uji Tambahan (Khusus Kamera AR)

Adapun uji tambahan yang dilakukan yaitu dengan cara menguji jarak kamera ke *marker* yang di sorot dengan pencahayaan lampu yang digunakan mempunyai intensitas cahaya 45 *Lux* dan dilakukan di ruangan seluas 12 m², berikut adalah tabel hasil uji kamera AR.

Tabel 4. 34 Hasil uji tambahan dengan *marker* gambar

Percobaan	Jarak							Persentase
	10 cm	20 cm	30 cm	40 cm	50 cm	60 cm	80 cm	
1	✓	✓	✓	✓	✓	✓	✗	85,71%
2	✓	✓	✓	✓	✓	✓	✗	85,71%
3	✓	✓	✓	✓	✓	✓	✗	85,71%
4	✓	✓	✓	✓	✓	✓	✗	85,71%
5	✓	✓	✓	✓	✓	✓	✗	85,71%
6	✓	✓	✓	✓	✓	✓	✗	85,71%
7	✓	✓	✓	✓	✓	✓	✗	85,71%
8	✓	✓	✓	✓	✓	✓	✗	85,71%
9	✓	✓	✓	✓	✓	✓	✗	85,71%
10	✓	✓	✓	✓	✓	✓	✗	85,71%

Berdasarkan tabel hasil uji tambahan di atas dapat disimpulkan bahwa nilai persentase rata-rata dengan 10 (sepuluh) kali percobaan secara berturut-turut adalah 87,5%. Adapun penyebab tidak terdeteksinya *marker* gambar dengan jarak 80 cm ketika kamera sudah tidak dapat membaca *marker* gambar dengan jelas akibat jarak.

Tabel 4. 35 Hasil uji tambahan dengan *marker* benda

Percobaan	Jarak					Persentase
	5 cm	10 cm	15 cm	20 cm	25 cm	
1	✓	✓	✓	✓	✗	80%
2	✓	✓	✓	✓	✗	80%
3	✓	✓	✓	✓	✗	80%
4	✓	✓	✓	✓	✗	80%
5	✓	✓	✓	✓	✗	80%
6	✓	✓	✓	✓	✗	80%
7	✓	✓	✓	✓	✗	80%
8	✓	✓	✓	✓	✗	80%

Percobaan	Jarak					Persennt ase
	5 cm	10 cm	15 cm	20 cm	25 cm	
9	✓	✓	✓	✓	✗	80%
10	✓	✓	✓	✓	✗	80%

Berdasarkan tabel hasil uji tambahan di atas dapat disimpulkan bahwa nilai persentase rata-rata dengan 10 (sepuluh) kali percobaan secara berturut-turut adalah 80%. Adapun penyebab tidak terdeteksinya *marker* benda dengan jarak 25 cm ketika kamera sudah tidak dapat membaca *marker* benda dengan jelas akibat jarak dan kurangnya cahaya yang tampak pada *marker*.

Catatan tambahan pada uji *marker* benda bahwa setiap benda memiliki pola unik dan bentuk yang berbeda-beda. Oleh karena itu, untuk pemilihan *marker* benda perlu pemilihan kriteria yang sesuai agar benda tersebut memenuhi syarat dijadikan sebagai *marker*.

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan penelitian yang telah dilakukan dapat disimpulkan sebagai berikut:

1. Penulis dapat membuat aplikasi media pembelajaran bangun ruang yang dapat mempermudah siswa/i dalam belajar bangun ruang.
2. Hasil uji yang telah dilakukan diantaranya uji *Black box* dan *White box* semuanya berjalan baik yang mana tampilan dan tombol berfungsi dengan baik.
3. Hasil implementasi yakni dengan uji langsung ke siswa/i dengan cara siswa/i menggunakan atau mencoba langsung aplikasi sebagai percobaan seberapa baik atau layak digunakannya aplikasi.

B. Saran

Penulis menyadari bahwa masih ada kekurangan yang ada pada penelitian ini dan masih perlu dikembangkan agar dapat lebih sempurna kedepannya. Daripada itu penulis memiliki beberapa saran untuk pengembang selanjutnya, sebagai berikut:

1. Mempercantik desain *interface* aplikasi sehingga dapat terlihat lebih menarik.
2. Menambah fitur-fitur aplikasi yang dapat diperlukan nantinya.

DAFTAR PUSTAKA

- Abidin, Nur, and Ahmad Fajrul Haq. 2023. "Aplikasi Media Pembelajaran Anak Usia Dini Menggunakan Teknologi Augmented Reality Berbasis Android." *Jurnal Nasional Komputasi dan Teknologi Informasi (JNKTI)* 6(1): 95–102.
- Fortuna, Sanadz, Ade Irma Purnamasari, and Arif Rinaldi Dikananda. 2023. "Game Edukasi Menyusun Kata Berbasis Android Dengan Metode MDLC Sebagai Media Pembelajaran Anak Usia Dini Pada Paud Wijaya Kusuma 1 Kota Cirebon." *Jurnal Teknologi Ilmu Komputer* 1(2): 61–65.
- Gusdevi, Harya et al. 2022. "Pengujian White-Box Pada Aplikasi Debt Manager Berbasis Android." *Naratif: Jurnal Nasional Riset, Aplikasi dan Teknik Informatika* 4(1): 11–22.
- Hiperakif, Menangani Anak. 2023. "3 (1,2,3." 08: 18–23.
<https://journal.fkpt.org/index.php/Explorer/article/view/53>.
- Khairani, Rini, Luluk Elvitaria, and Salamun Salamun. 2021. "Augmented Reality Pengenalan Kompleks Perkantoran Kota Bagansiapiapi Berbasis Andrioid." *Explorer* 1(1): 38–.
- Lathifuddin, Muhammad, Ai Nurhayati, and Rita Patriasih. 2018. "Pengetahuan 'Buah Dan Sayur' Sebagai Hasil Penyuluhan Gizi Pada Siswa SD Yang Mengalami Obesitas Di Kota Bandung." *Media Pendidikan, Gizi, dan Kuliner* 7(1): 45–54. <https://ejournal.upi.edu/index.php/Boga/article/view/11596>.
- Menrisal, Menrisal, and Nadiya Rizki Utami. 2019. "Erancangan Dan Pembuatan Media Pembelajaran Android Pada Mata Pelajaran Simulasi Dan Komunikasi Digital (Studi Kasus Kelas X Smkn 7 KerinPci)." *Jurnal Pendidikan Teknologi Informasi* 6(1): 80–86.
- Putra, Dede Wira Trise, and Rahmi Andriani. 2019. "Unified Modelling Language (UML) Dalam Perancangan Sistem Informasi Permohonan Pembayaran Restitusi SPPD." *Jurnal TeknolIf* 7(1): 32.
- Raschka, S. (2020). *Machine Learning in Python: Main Developments and Technology Trends In Data Science, Machine Learning and Artificial Intelligence. Information*.
- Rohani. 2020. "Media Pembelajaran." *Repository.Uinsu*: 234.
<https://books.google.com/books?hl=en&lr=&id=npLzDwAAQBAJ&oi=fnd&pg=PA1&dq=media+pembelajaran&ots=Nr8w9uLXRR&sig=dO9nzuMdeU76Gwa7wE2-xLcBB7I>.

- Setiyani, Lila, and Benny Setiawan. 2021. "Analisis Dan Design Manajemen Control Produksi Menggunakan Business Process Improvement Dan Unified Modelling Language (STUDI KASUS: PT. MULTISTRADA)." *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi* 16(1): 27–37.
- Triwibowo, Deny Nughoru, and Iis Setiawan Mangkunegara. 2024. "Perancangan Augmented Reality Untuk Meningkatkan Pengetahuan Orang Tua Dan Anak Tentang Nutrisi Pada Buah-Buahan." *Decode: Jurnal Pendidikan Teknologi Informasi* 4(2): 406–17.
- Wahyuddin, Wahyuddin, and Hasnawati Hasnawati. 2023. "Aplikasi Media Pembelajaran Pengenalan Pancasila Menggunakan Augmented Reality." *Jurnal Sintaks Logika* 3(3): 8–15.
- Yulianto, M T, and A Mulyani. 2019. "Aplikasi Mobile Dokter Gizi Berbasis Android." ... : *Jurnal Pengembangan Riset* ... 6(1): 10–15. <http://e-jurnal.lppmunsera.org/index.php/PROSISKO/article/view/1121%0Ahttps://e-jurnal.lppmunsera.org/index.php/PROSISKO/article/download/1121/932>.