

## **BAB II**

### **KAJIAN PUSTAKA**

#### **2.1 Teori Dasar**

Beberapa teori dasar dalam melakukan penelitian ini yaitu *Mobil Application*, *Android*, *Augmented Reality*, Teknik *Tracking*, bahasa *C#*, UML dan Pengembangan perangkat lunak.

##### **2.1.1 Mobile Application**

Aplikasi seluler (*Mobile application*) adalah skema penerapan dan pengelolaan fungsi kerja untuk perangkat seluler seperti telepon seluler dan tablet. Skema ini terdiri dari kebijakan masing-masing dari pengembang aplikasi seluler (Hayes et al, 2020). Adapun menurut (Herlandy et al., 2019) aplikasi seluler yang di kembangkan oleh para *developer* aplikasi yang ditanamkan perangkat komunikasi, khususnya yang memiliki telepon pintar dapat dimanfaatkan untuk media pembelajaran. Bentuk dari teknologi pembelajaran ditampilkan di ponsel yakni berupa *e-book*, *game*, dan simulasi.

Aplikasi seluler merupakan sebuah aplikasi yang memungkinkan untuk melakukan mobilitas terhadap perangkat telepon pintar yakni PDA dan *smartphone*. Aplikasi seluler dapat juga dikatakan sebagai komponen untuk mendukung sistem kerja dari perangkat seluler untuk melakukan fungsi dan tugas yang di arahkan oleh pengguna. Para pengembang aplikasi seluler biasanya memasarkan dan menjual produk mereka melalui sistem online seperti *Apps Store* dan *Google Play*. Pengguna dapat mengunduh nya secara gratis dan berbayar sesuai

dengan kebijakan dari pengembang aplikasi. Aplikasi seluler yang sudah di unduh masih dalam kondisi pra-intsal, untuk mengkatifkannya di dalam telepon seluler, PDA, *smartphone* dan *tablet* maka pengguna di haruskan untuk melakukan instalasi terlebih dahulu (Rendi & Handoko, 2021).

### 2.1.2 Android

Android pertama sekali dikembangkan oleh Android Inc dengan berbasis linux, tetapi kini sudah di miliki oleh Google dan pada tahun 2007 google meluncurkan AOSP (*Android OpenSource Project*). Pengembangan android dilakukan oleh perusahaan pengembangan perangkat keras, perangkat lunak dan telekomunikasi yang termasuk *Intel, Google, NVIDIA, Qualcomm, Motorola, HTC* dan *T-Mobile*, yang tergabung disebut *Open Handset Alliance* (OHC), dimana android adalah OS yang dapat beradaptasi Didirikannya *Open Handset Alliance* (OHC) bertujuan membangun peralatan canggih yang menyesuaikan dengan teknologi untuk dapat menghemat waktu dan biaya serta meningkatkan layanan dan penyedia fitur terbaik untuk pengguna. Perangkat lunak untuk pengembangan android diluncurkan di bawah *lisensi Apache* sebagai *Open Source* yaitu setiap kegiatan untuk pengembangan aplikasi *smartphone* dan *tablet* dapat dilakukan secara gratis oleh para pengembang aplikasi (Haris et al., 2018).



**Gambar 2.1** Logo *Android*  
**Sumber:** (Data penelitian, 2021)

Android menyediakan 4 komponen utama yang harus dimiliki oleh sistem operasi android, sebagai berikut:

1. **Aktivitas:** berfungsi untuk menyediakan antarmuka untuk ditampilkan kepada pengguna dengan setiap aktivitas aplikasi dapat menyesuaikan dengan tampilan yang berbeda-beda.
2. **Layanan:** berfungsi mengatur untuk sistem pemrosesan pada latar belakang.
3. **Penyedia konten:** berfungsi sebagai dukungan untuk aplikasi terhadap data yang menawarkan kemampuan penyimpanan, pengambilan dan dapat berbagi dalam satu komponen aplikasi atau banyak aplikasi.
4. **Siaran penerima:** merespons secara tidak sinkron ke seluruh sistem siaran pesan google. Berfungsi sebagai jalur masuk untuk menghubungkan dari komponen satu ke komponen dan meneruskan pesan ke aktivitas dan layanan untuk dilakukan penanganan (Siyamto, 2019).

*Android* merupakan sistem operasi yang dikembangkan dengan berbasis *linux* untuk disematkan pada setiap perangkat *mobile device* yang mencakup sistem operasi, aplikasi dan *middleware*. Komponen yang diterapkan ke dalam *android* meliputi *boot loader*, *device driver*, fungsi-fungsi pustaka, hingga API (*Application Programming Interface*) yakni *Software Development Kit* untuk pengembang aplikasi. Para pengembang berbasis *android* diuntungkan dengan sistem operasi *android* yang *compatibility* dengan perangkat pintar sejenis seperti *smartphone*, *tablet*, *smart TV*, *smart watch* dan perangkat pintar lainnya (Soewito et al., 2019).

**Tabel 2.1** Versi dan fitur OS *Android*

No	Versi <i>Android</i>	API level	Tahun	Fitur
1	<i>Cupcake</i>	3	2009	<ul style="list-style-type: none"> <li>▪ Dukungan <i>Widget</i></li> <li>▪ Navigasi pencarian aplikasi</li> </ul>
2	<i>Donut</i>	4	2009	<ul style="list-style-type: none"> <li>▪ Tangkapan layar</li> <li>▪ Perintah suara</li> </ul>
3	<i>Eclair</i>	5	2010	<ul style="list-style-type: none"> <li>▪ Hemat Batrey,</li> <li>▪ <i>Keyboard auto-correct</i></li> <li>▪ Resolusi 320x240</li> </ul>
4	<i>Froyo</i>	8	2010	<ul style="list-style-type: none"> <li>▪ Keamanan tinggi</li> <li>▪ API <i>cloud</i></li> <li>▪ Deteksi gerak.</li> </ul>
5	<i>Ginger-bread</i>	9	2011	<ul style="list-style-type: none"> <li>▪ Rentang NFC dalam 10 cm</li> <li>▪ Panggilan video dan audio</li> <li>▪ Didukung 3G</li> <li>▪ Sensor giroskop</li> </ul>
6	<i>Honey-comb</i>	11	2011	<ul style="list-style-type: none"> <li>▪ Multiprosesor simetris</li> <li>▪ <i>Multitasking</i></li> <li>▪ <i>Google talk</i></li> <li>▪ Efek 3D</li> </ul>
7	<i>Ice cream-sandwich</i>	14	2011	<ul style="list-style-type: none"> <li>▪ Pemberitahuan muncul dilayar kunci</li> <li>▪ Tangkapan gambar cepat</li> <li>▪ NFC (<i>Near Field Communication</i>) di aktifkan</li> </ul>
8	<i>Jelly bean</i>	16	2012	<ul style="list-style-type: none"> <li>▪ <i>Speedy</i></li> <li>▪ Panggilan audio dan video</li> <li>▪ Akses jarak jauh</li> </ul>

**Tabel 2.1** (Lanjutan)

9	<i>Kit-kat</i>	19	2013	<ul style="list-style-type: none"> <li>▪ Responsif</li> <li>▪ CPU 3 core</li> <li>▪ Dukungan <i>cloud</i></li> </ul>
10	<i>Lollipop</i>	21	2015	<ul style="list-style-type: none"> <li>▪ Dukungan tampilan 3D</li> <li>▪ 64bit MIPS dan ARM</li> <li>▪ <i>Heart rate</i></li> <li>▪ <i>Swipe pinch</i></li> <li>▪ Tangkapan layar</li> </ul>
11	<i>Marhs-mallow</i>	23	2015	<ul style="list-style-type: none"> <li>▪ Deteksi sidik jari</li> <li>▪ Kamera sensor gerakan dan suara</li> <li>▪ Resolusi tinggi</li> <li>▪ Cadangan di <i>cloud</i></li> <li>▪ Mode hemat daya</li> </ul>
12	<i>Nougat</i>	25	2016	<ul style="list-style-type: none"> <li>▪ Deteksi sidik jari dan telapak tangan</li> <li>▪ Sensor gerakan dan suara,</li> <li>▪ <i>Cloud</i></li> <li>▪ Mode hemat daya</li> </ul>

**Sumber:** (Haris et al., 2018)

*Android Lollipop* awal mula menjadi sistem operasi *android* yang mendukung teknologi *Augmented Reality*. *Android Lollipop* dukungan API level 21 meningkatkan pengalaman pengguna dengan tampilan 3D yang dapat melihat elemen secara langsung. Didukung dengan transisi elemen *visual* dari keadaan satu ke keadaan lain sehingga dapat menampilkan objek secara mulus meskipun terjadi penundaan di atas utama antarmuka pengguna akan tetapi tidak mengganggu tampilan dari objek.

Versi *Lollipop* disematkan arsitektur ARM dan OS 64-bit MIPS guna untuk mengurangi sampah secara efisien dan menjaga kinerja aplikasi tetap responsif. Untuk mempertegas perbedaan dari versi *Kit-kat* maka versi *Lollipop* melakukan peningkatan pada fitur *audio* yang sinkronisasi *video* di dukung oleh *pipelining* guna untuk kelancaran keluaran. Pada versi android ini juga diperkenalkan teknologi sensor baru untuk detak jantung, deteksi aktifitas sentuhan (jentikan, gesek, cubit), sensor interaksi, dan sensor deteksi rekam aktivitas. API 21 yang ditambahkan pada versi *Lollipop* guna untuk meningkatkan kinerja baterai, aksesibilitas, tampilan *web*, tangkapan layar dan fitur-fitur untuk kamera (Haris et al., 2018).

### **2.1.3 Augmented Reality**

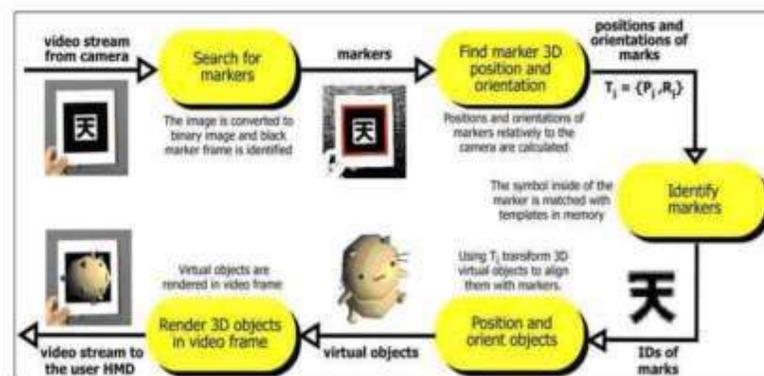
*Augmented Reality* merupakan teknologi yang dapat menampilkan objek dalam bentuk 2D dan 3D yang dihasilkan dari penggabungan antara dunia maya dan nyata yang digunakan untuk hiburan, teknik desain dan sebagai media informasi. Perubahan bentuk dari tekstur objek yang nyata menjadi kedalam bentuk 3D membuat penyampaian suatu informasi terhadap suatu objek menjadi lebih menarik serta memudahkan proses pelaksanaan pemberian informasi (Adnan et al., 2020).

Beberapa komponen yang diperlukan pada saat melakukan pembuatan dan pengembangan aplikasi berbasis *Augmented Reality* yaitu:

- A. Komputer, alat utama untuk perancangan dan pembuatan aplikasi yang menerapkan teknologi *Augmented Reality*.

- B. *Marker*, alat yang berupa gambar untuk objek deteksi dan sebagai tempat untuk menampilkan objek *virtual*.
- C. Kamera, berfungsi untuk *recording* sensor yang sebagai wadah dalam menampilkan objek virtual melalui kamera smartphone. Pada saat aplikasi mendeteksi *marker* yang dibuat maka pada aplikasi akan menampilkan objek *virtual* diatas *marker* yang sudah dideteksi sebelumnya.

Prinsip kerja teknologi AR dalam menampilkan objek ke dalam dunia *virtual* terdiri dari beberapa tahapan yang saling berhubungan, sebagai berikut:



**Gambar 2.2** Sistem kerja AR

**Sumber:** (Maldanop et al., 2017)

Pada aplikasi akan mengaktifkan kamera dan meng-*capture marker* lalu mengirimkannya ke *processor* untuk menentukan pola deteksi. Aplikasi akan membuka kamera selanjutnya proses deteksi *marker* kemudian memilah objek *virtual* yang akan ditampilkan yang menyatukan dengan informasi yang dimiliki oleh aplikasi. Melalui aplikasi objek *virtual* akan di tampilkan dan diletakkan diatas *marker* yang telah di-*capture* sebelumnya. Keduanya tergabung menjadi realitas campuran/*mixed reality* (Maldanop et al., 2017).

### 2.1.2 *Markerless Tracking*

Teknik *Tracking* pada teknologi *Augmented Reality* yang terbaru dan populer pada saat sekarang yaitu *Markerless Tracking*. *Markerless* merupakan alat yang berupa suatu tulisan, gambar dan objek langsung dengan menggunakan sistem *tracking* objek melalui metode *Defined Target*. Metode ini secara otomatis melakukan pencarian titik pola (*Pattern Recognition*) untuk menentukan perhitungan posisi yang tepat antara kamera dengan objek pada dunia nyata, lalu merefleksikan objek menggunakan titik-titik pada fitur alami MAR seperti: *edge*, *corner*, garis atau objek 3D (Christoffel et al., 2019).



**Gambar 2.3** Contoh *Markerless*  
**Sumber:** (Data penelitian, 2021)

### 2.1.4 Bahasa Pemrograman C#

Bahasa pemrograman C# (baca: *C-sharp*) merupakan bahasa pemrograman dengan mengikuti perkembangan *framework*. *NET* yang di ciptakan oleh perusahaan *Microsoft Corp*. Penggunaan bahasa pemrograman ini untuk mendukung perancangan aplikasi yang berskala kecil dan besar dengan memberikan kemudahan kepada pengembang aplikasi supaya dalam melakukan

masukan kode program untuk lebih mudah di pelajari dan aman. (Heriyanto et al., 2018).

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class NewBehaviourScript : MonoBehaviour {
6
7     // Use this for initialization
8     void Start () {
9
10    }
11
12    // Update is called once per frame
13    void Update () {
14
15    }
16 }
17

```

**Gambar 2.4** Struktur penulisan bahasa C# pada *Unity*  
**Sumber :** (Data penelitian, 2021)

Fitur penting bahasa pemrograman C# yang diterapkan dalam *Unity*:

```

using UnityEngine; ❶
namespace MyGame { ❷
    [RequireComponent(typeof(SpriteRenderer))] ❸
    class Alien : MonoBehaviour { ❹
        public bool appearsPeaceful; ❺
        private int cowsAbducted;
        public void GreetHumans() {
            Debug.Log("Hello, humans!");
            if (appearsPeaceful == false) {
                cowsAbducted += 1;
            }
        }
    }
}

```

**Gambar 2.5** Contoh *script* bahasa C# di *Unity*  
**Sumber :** (Manning & Buttfield-Addison, 2017)

1. *Using* (Deklarasi Pustaka), menunjukkan tipe penggunaan *Unity* yaitu *UnityEngine*.
2. *Namespace* (Nama proyek), dalam menghindari penamaan yang sama setiap nama diletakkan pada tanda kurung {}.

3. *Attribute* (Atribut), guna menambah informasi lebih mengenai struktur program.
4. *Class* (Kelas), di dalamnya ada dua tipe kelas yaitu *private class* dan *superclass*.
5. Terdapatnya sebuah variabel yang dinyatakan kedalam kelas tersebut (Manning & Buttfield-Addison, 2017).

### 2.1.5 Unified Modeling Language (UML)

Penggunaan UML sangat penting dalam membuat setiap alur kerja aplikasi yang bermodelkan *Object Oriented Analysis*. Pendekatan dengan menggunakan OOA merupakan pendekatan yang lebih dekat kepada sistem atau aplikasi, hal ini dikarenakan dengan pendekatan ini setiap proses sistem dapat di analisa dengan seksama sehingga memenuhi kebutuhan sebuah sistem (Maharani, 2018).

#### 1. *Use case diagram*

*Usecase Diagram* akan menggambarkan setiap alur proses yang nantinya akan dilakukan oleh aktor.

**Tabel 2.2** Simbol-simbol *Use case diagram*

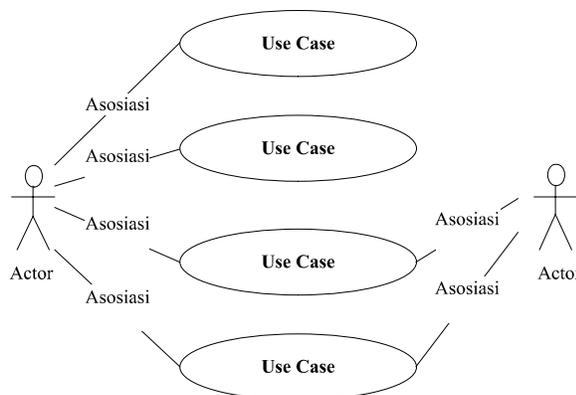
Simbol	Keterangan
	<p><i>Use Case</i></p> <ul style="list-style-type: none"> <li>▪ Mewakili setiap langkah</li> <li>▪ Kegiatan sistem berada di dalamnya.</li> <li>▪ Nama mengharuskan label atau kata kerja kemudian di ikuti oleh kata kerja.</li> </ul>

**Tabel 2.2** (Lanjutan)

	<p><i>System Boundary</i></p> <ul style="list-style-type: none"> <li>▪ Pada bagian atas memuat nama sistem.</li> <li>▪ Mendeskripsikan dari sistem termasuk ruang lingkungannya.</li> </ul>
	<p>Asosiasi Boundary</p> <ul style="list-style-type: none"> <li>▪ Penghubung antara aktor dan kegiatan sistem (<i>Usecase</i>)</li> </ul>
<p>&lt;&lt;include&gt;&gt;</p> 	<p><i>Include</i></p> <ul style="list-style-type: none"> <li>▪ Sebuah <i>sub usecase</i> akan aktif jika terlebih dahulu menjalankan fungsi <i>main usecase</i>.</li> <li>▪ Arah panah harus menuju <i>main usecase</i>.</li> </ul>
<p>&lt;&lt;extend&gt;&gt;</p> 	<p><i>EXTEND</i></p> <ul style="list-style-type: none"> <li>▪ Sebuah <i>main usecase</i> yang dapat di jalankan sendiri tanpa aktif fungsional dari <i>usecase</i>.</li> </ul>
	<p>GENERALISASI / <i>GENERALIZATION</i></p> <ul style="list-style-type: none"> <li>▪ Penghubung dari <i>usecase</i> umum ke <i>usecase</i> khusus.</li> </ul>

Sumber : (Maharani, 2018)

Contoh *Usecase Diagram* dapat di gambarkan sebagai berikut:

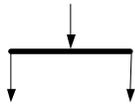
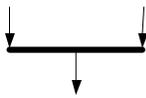
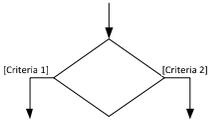
**Gambar 2.6** Contoh *Use case diagram*

Sumber : (Data penelitian, 2021)

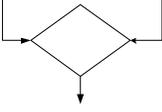
## 2. Activity diagram

*Activity Diagram* untuk menunjukkan dan menjelaskan serangkaian aksi awal hingga menentukan hasil aksi tersebut.

**Tabel 2.3** Simbol-simbol *Activity diagram*

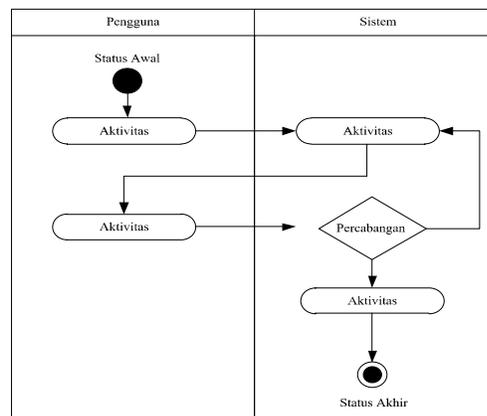
Simbol	Keterangan
	<p><i>Start poin</i></p> <ul style="list-style-type: none"> <li>Awal aktivitas (<i>star point</i>)</li> <li>Digunakan 1 <i>star point</i> untuk satu aktivitas</li> </ul>
	<p><i>End Poin</i></p> <ul style="list-style-type: none"> <li>Akhir aktivitas (<i>end point</i>)</li> <li>Digunakan 1 <i>star point</i> untuk satu aktivitas</li> </ul>
	<p><i>Activities</i></p> <ul style="list-style-type: none"> <li>Suatu aktivitas dari pengguna dan sistem</li> <li>Aktivitas haruslah dengan nama kata kerja agar menggambarkan aktivitas</li> <li>Setiap aktivitas harus satu jalur masuk dan keluar</li> </ul>
	<p><i>Fork</i></p> <ul style="list-style-type: none"> <li>Percabangan</li> <li>Pada satu aliran akan mengerjakan dua atau lebih aktivitas maka menggunakan percabangan</li> </ul>
	<p><i>Join</i></p> <ul style="list-style-type: none"> <li>Penggabungan</li> <li>Aktivitas berada pada beberapa aliran maka dapat dilakukan penggabungan</li> </ul>
	<p><i>Decision poin</i></p> <ul style="list-style-type: none"> <li>Berada di tengah belah ketupat yang tidak memiliki aktivitas atau keterangan</li> </ul>

Tabel 2.3 (Lanjutan)

[ ]	<p><i>Guard</i></p> <ul style="list-style-type: none"> <li>▪ Transisi aktif ketika mendapati kondisi benar</li> </ul>
	<p><i>Merge</i></p> <ul style="list-style-type: none"> <li>▪ Setelah melewati <i>decision poin</i> akan berbeda kondisi setelah jalur kembali</li> </ul>
<i>Swimlane</i>	<p><i>Swimlane</i></p> <ul style="list-style-type: none"> <li>▪ Aktor pada setiap aktivitas dapat dikelompokkan dengan dasar ditulis dengan nama aktor</li> <li>▪ <i>Swimlane</i> sebuah aktivitas yang dapat digambarkan secara horizontal dan vertikal</li> </ul>

Sumber : (Maharani, 2018)

Contoh *Activity Diagram* dapat di gambarkan sebagai berikut:



Gambar 2.7 Contoh *Activity diagram*

Sumber : (Data penelitian, 2021)

### 3. *Sequence diagram*

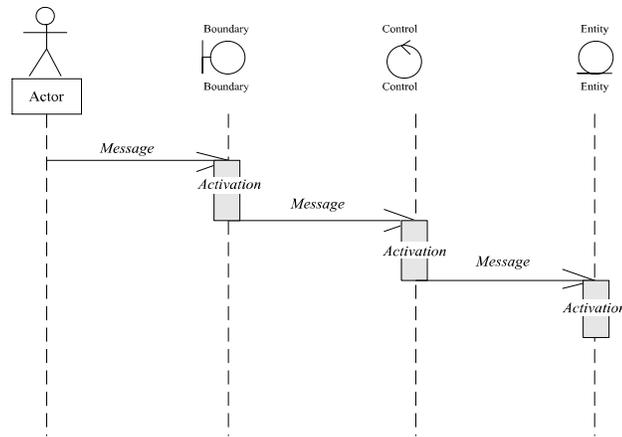
*Sequence diagram* alur proses yang menggambarkan setiap satu tindakan terhadap tindakan lain.

Tabel 2.4 Simbol-simbol *Sequence diagram*

Simbol	Keterangan
	<i>Actor</i> <ul style="list-style-type: none"> <li>▪ Pengguna yang sedang berinteraksi dengan sistem</li> </ul>
	<i>Boundary</i> <ul style="list-style-type: none"> <li>▪ Penghubung aktor dan sistem</li> </ul>
	<i>Control</i> <ul style="list-style-type: none"> <li>▪ Mengatur perilaku sistem</li> </ul>
	<i>Entity</i> <ul style="list-style-type: none"> <li>▪ Menyimpan informasi sistem</li> <li>▪ Struktur data dari sistem akan digambarkan <i>entity</i>.</li> </ul>
	<i>Activation</i> <ul style="list-style-type: none"> <li>▪ Kondisi saat interaksi objek</li> <li>▪ Sebuah kondisi yang aktif dengan durasi posisi lurus dengan panjang simbol.</li> </ul>
	<i>Message</i> <ul style="list-style-type: none"> <li>▪ Pesan antar objek di gambarkan dengan memuat urutan kejadian</li> </ul>

Sumber : (Maharani, 2018)

Contoh *Usecase Diagram* dapat di gambarkan sebagai berikut:



**Gambar 2.8** Contoh *Sequence diagram*

Sumber : (Data penelitian, 2021)

#### 4. *Class diagram*

Kelas adalah kumpulan objek yang memiliki kesamaan atribut dan perilaku sistem. Dilenkapi dengan 3 nilai pokok, yaitu nama, atribut dan operasi sistem.

Tabel 2.5 Simbol-simbol *Class diagram*

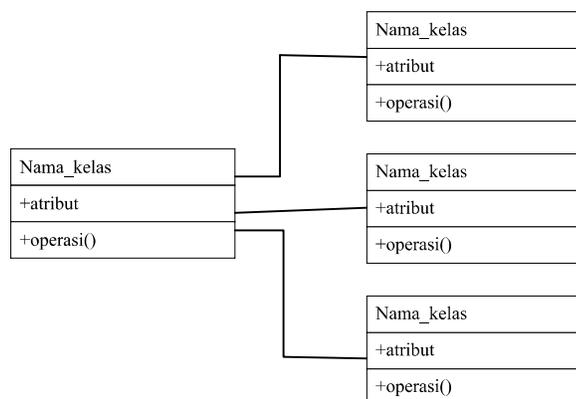
Simbol	Deskripsi
<div style="border: 1px solid black; padding: 5px; width: fit-content;">           Nama_kelas            +atribut            +operasi()         </div>	<ul style="list-style-type: none"> <li>Struktur sistem dalam kelas</li> </ul>
Antarmuka / interface  Nama_interface	<ul style="list-style-type: none"> <li>Menggunakan proses yang berorientasi objek untuk menggambarkan antarmuka sistem</li> </ul>
Asosiasi/ association 	<ul style="list-style-type: none"> <li>Relasi antar kelas yang memuat makna umum</li> <li>Asosiasi biasanya dilengkapi dengan <i>multiplicity</i></li> </ul>

**Tabel 2.5** (Lanjutan)

Asosiasi berarah / <i>directed association</i> →	<ul style="list-style-type: none"> <li>Relasi yang bermakna dari satu kelas dengan kelas lain</li> </ul>
generalisasi →▷	<ul style="list-style-type: none"> <li>Relasi umum ke khusus yang menunjukkan relasi antar kelas</li> </ul>
Kebergantungan/ <i>dependency</i> .....>	<ul style="list-style-type: none"> <li>Relasi antar kelas bergantung dari makna antar kelas</li> </ul>
Agregasi/ <i>aggregation</i> .....◊	<ul style="list-style-type: none"> <li>Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)</li> </ul>

Sumber : (Maharani, 2018)

Contoh *Class Diagram* dapat di gambarkan sebagai berikut:

**Gambar 2.9** Contoh *Class diagram*

Sumber : (Data penelitian, 2021)

### 2.1.6 Pengembangan perangkat lunak

Pengembangan perangkat lunak merupakan suatu tahapan demi tahapan kegiatan dengan tujuan untuk meninjau kebutuhan atas aplikasi yang akan dibangun sehingga terciptanya aplikasi handal dan akurat untuk pengguna dan untuk pengembangan lanjutan.



**Gambar 2.10** Gambaran pengembangan perangkat lunak  
**Sumber :** (Tabrani & Pudjiarti, 2018)

1. Analisis kebutuhan aplikasi  
 Penentuan kebutuhan aplikasi yang akan dibangun di titik beratkan kepada peran kebutuhan dari pengguna. Hal inilah yang membuat layaknya aplikasi tersebut di buat.
2. Desain  
 Proses desain aplikasi termasuk yang penting dalam membangun aplikasi yang meliputi dari kegiatan struktur data dan representasi antarmuka untuk pengguna.
3. Pembuatan kode program  
 Kode program aplikasi akan berpanduan kepada desain aplikasi yang telah dibuat sehingga selaras dengan fungsi masing-masing menu.
4. Pengujian  
 Tahapan pengujian aplikasi akan di lakukan uji coba secara logis dan fungsional aplikasi sehingga memastikan aplikasi sudah sesuai dengan kebutuhan awal (Tabrani & Pudjiarti, 2018).

## 2.2 *Sparepart Motor*

Objek penelitian ini ditetapkan untuk membuat alur penelitian jelas dan terarah. Pada kendaraan bermotor maka fokus dari penelitian ini berupa *sparepart* motor beat sebagaimana motor ini yang sangat umum pemakaiannya di kalangan masyarakat. Adapun beberapa objek *sparepart*-nya sebagai berikut:

- 1) ECU



**Gambar 2.11** ECU motor beat

**Sumber:** (Data penelitian, 2021)

ECU (*Elektronik Control Unit*) adalah perangkat elektronik yang otomatis mempunyai tugas untuk mengatur proses pembakaran. ECU punya peran mengontrol beberapa fungsi dan fitur pada motor injeksi dan dapat dikatakan juga sebagai pusatnya dari motor injeksi yang semua berkaitan dengan sistem injeksi yang dikontrolnya di ECU, seperti *riding mode*.

Dalam beberapa kasus terdapat ada beberapa tanda yang menjadi ciri utama sebagai indikasi jika ada kerusakan pada bagian ECU motor, seperti contoh berikut:

- a. Lampu indikator mati namun kondisi kontak motor hidup. Kondisi ini aki motor dalam keadaan baik-baik saja akan tetapi sistem lampu senja, lampu rem, lampu sein, klakson, speedometer dan starter tidak bisa diaktifkan maka pertanda kuat dari ECU yang mengalami masalah.

- b. Tidak berfungsinya pompa dan busi. Secara normalnya ketika kunci kontak dalam kondisi hidup maka secara otomatis pompa bahan bakar akan mengeluarkan bunyi seperti dengungan. Bunyi dengungan dari pompa bahan bakar ini sudah bisa dijadikan sebagai patokan dalam menentukan dan mengetahui kondisi terkini pada sistem injeksi motor.
  - c. Adanya kebocoran arus listrik. Masalah lain yang menjadi tanda kuat jika ECU motor mengalami kerusakan bisa jadi dikarenakan adanya kebocoran arus listrik. Kondisi ini dapat dilihat pada saat posisi kontak sudah dimatikan namun klakson dan lampu-lampu masih aktif dan menyala hanya saja arus listriknya kecil.
- 2) Kampas Kompling Ganda



**Gambar 2.12** Kampas Kompling Ganda

**Sumber:** (Data penelitian, 2021)

Terkait dengan kampas ganda yang menggunakan gaya sentrifugal yakni komponen ini sangat berperan penting untuk membantu motor matic supaya berjalan secara sempurna yang memiliki peran yang signifikan untuk Continuously Variable Transmission (CVT). Peran kampas ganda berfungsi untuk menyalurkan dan memutuskan tenaga putaran mesin dari CVT depan ke belakang. Fungsi kampas ganda menggunakan gaya sentrifugal. Mengecek fungsi kampas ganda motor matic dapat dilakukan dengan mudah yaitu dengan membongkar CVT sudah

terlihat aus atau belum. Dapat juga dengan mengacu kepada jarak yang sudah ditempuh, biasanya sudah terjadi keausan pada kampas kopling ganda dengan jarak tempuh berkisar 24.000 kilometer hingga 35.000 kilometer.

Kampas kopling ganda motor matic yang mulai menipis mengalami gejala seperti:

- a. Getar pada saat dipakai berkendara
- b. Akselerasi awal terasa menjadi lebih berat
- c. Terjadi perubahan suara di area CVT

### 3) *Relay Starter*



**Gambar 2.13** *Relay Starter*

**Sumber:** (Data penelitian, 2021)

*Relay starter* memiliki fungsi sebagai jembatan penghubung antara kontak, aki dan *dinamo starter* di motor yang bertujuan untuk menghidupkan mesin. Komponen ini sangatlah penting yang mana ketika kunci kontak diputar ke posisi “*On*” maka arus listrik dari aki sudah disalurkan sistem *electric starter*. Tapi belum sampai membuat mesin menyala karena masih ditahan oleh relay starter. Barulah ketika tombol *starter* ditekan, *relay starter* menjembatani arus listrik ke dinamo

starter dan mesin pun menyala. Seiring penggunaan, *relay starter* bisa mengalami kerusakan. Hal ini dikarenakan komponen tersebut kerap menerima arus listrik yang besar dari aki sehingga menyebabkan motor tidak bisa menyala.

Adapun ciri-ciri relay starter motor rusak dapat diketahui dari beberapa komponen di bawah ini.

- a. Saat tombol start ditekan muncul bunyi “Cetak”
- b. Mesin tidak bisa menyala, padahal aki normal
- c. Aki mendadak tekor saat di *starter*
- d. Dinamo masih berfungsi, tapi mesin tidak menyala

#### 4) *Ring Piston*



**Gambar 2.14** *Ring Piston*

**Sumber:** (Data penelitian, 2021)

*Ring Piston* atau umumnya ring seher merupakan sebuah komponen mesin yang berbentuk bulat seperti cincin yang melingkar guna untuk mencegah terjadinya kebocoran kompresi dan oli di antara celah *piston* dengan dinding

silinder serta berperan juga untuk memindahkan panas dari hasil pembakaran *piston* ke dinding silinder.

*Ring piston* yang menempel pada satu *piston* tapi mempunyai beberapa kegunaan yang berbeda-beda, sebagai berikut:

1. *Top Compression Ring*: Terletak pada posisi yang paling atas yang berfungsi untuk mencegah terjadinya kebocoran kompresi campuran antara udara dan bensin dari ruang bakar selama siklus pembakaran terjadi.
2. *Second Compression Ring*: Berada pada posisi kedua guna untuk *backup* kerja dari *Top Compression Ring* serta melakukan fungsi sebagai penyeka sisa oli.
3. *Ring Oli*: Dalam satu jalur *piston*, *ring oli* terdiri dari 3 cincin sekaligus yang terletak pada urutan paling bawah guna untuk menampung dan membawa oli serta melumasi dinding liner silinder.

Berikut adalah tanda-tanda ring piston bermasalah yang dapat untuk diketahui sebagai berikut:

- a. Oli mesin menjadi cepat habis
- b. Keluarnya asap putih dari knalpot motor
- c. *Kick starter* lebih ringan kadang ngelos

5) *Roller*

**Gambar 2.15** *Roller*

**Sumber:** (Data penelitian, 2021)

*Roller* merupakan termasuk kedalam komponen vital yang ada pada sistem CVT (*Continous Variable Transmission*) guna untuk membuat motor bisa bergerak tanpa melakukan akselerasi manual. *Roller* mempunyai bentuk seperti tabung-tabung yang berukuran kecil. Untuk penggunaannya pada sepeda motor terdiri dari 6 buah sekaligus yang dipasang kedalam rumah *roller* yang ada di CVT. Daftar harga yang di tawarkan berkisar 9.000/pcs.

Beberapa tanda kerusakan *roller* yang sudah rusak/aus dapat dilihat dari ciri-ciri berikut ini:

- a. Tenaga berkurang, ini dirasa ketika motor di gas lalu ada terasa seperti tersendat-sendat, terlebih jika menaiki tanjakan akan membutuhkan tenaga lebih untuk menaikinya.
- b. Adanya suara krek-krek di bagian CVT

6) *Sliding shave*

**Gambar 2.16** *Sliding Shave*

**Sumber:** (Data penelitian, 2021)

Perilaku yang buruk dari pengendara sepeda motor dengan mengentahkan membuka gas, maka akan mempengaruhi komponen sliding shave. Hal ini akan membuat komponen *Sliding shave* menjadi cepat rusak (tidak lancar). Pengaruh ini akan menyebabkan munculnya suara ketika pengguna membuka gas. Tanda kerusakan yang dapat dikenali oleh pengguna jika *Sliding Shave* mengalami kerusakan yakni:

- a. Munculnya suara kasar seperti gredek-gredek sewaktu mengegas.
- b. Akselerasi putaran menengah terasa tertahan, tapi akan normal kembali sejalan kenaikan putaran mesin

## 7) Saringan Udara



**Gambar 2.17** Saringan Udara

**Sumber:** (Data penelitian, 2021)

Saringan udara menjadi salah satu komponen yang sangat penting dalam rangkaian teknologi injeksi. Pada saat pembakaran untuk menghasilkan tenaga atau dorongan maka membutuhkan peranan udara, untuk itu saringan udara berfungsi untuk menyaring udara dari kotoran atau debu yang akan masuk ke mesin atau ruang bakar. Adapun gejala kerusakan jika saringan udara mengalami kerusakan sebagai berikut:

- a. Dalam waktu dekat secara bertahap motor menjadi boros bensin
- b. Motor sulit untuk dinyalakan
- c. Muncul suara bising pada mesin
- d. Pada knalpot mengeluarkan asap hitam
- e. Motor sering mengalami macet secara tiba-tiba

### 2.3 *Software* Pendukung

Dukungan *software* yang berhubungan dengan konsep *Augmented Reality* seperti *CorelDRAW*, *Blender*, *Vuforia* dan *Unity*.

#### 2.3.4 *CorelDraw*

*CorelDraw* adalah perangkat lunak keluaran perusahaan *Corel Corporation* masuk kedalam jenis perangkat lunak *editor graphic vektor*. *CorelDraw* merupakan *Suite Grafis Corel* yang memiliki cakupan untuk *editor bitmap-image*, *corel photo-paint*, dan pemrograman terkait grafis sejenis guna untuk dapat mengedit masukan gambar dua dimensi seperti logo dan poster (Adnan et al., 2020).



**Gambar 2.18** Logo *CorelDraw*

**Sumber :** (Data penelitian, 2021)

Aplikasi *CorelDraw* akan digunakan untuk mendesain *marker* yang menjadi objek deteksi pada aplikasi yang sudah dibuat. *Marker* dirancang dengan melakukan masukan berupa gambar 2D yang sesuai dengan objek yang nantinya akan dijadikan kedalam bentuk 3D lalu ditampilkan pada dunia *virtual*.

### 2.3.5 Blender

*Blender* merupakan perangkat lunak yang berjalan pada sistem operasi *Linux*, *Macintosh* dan *Windows*, berfungsi sebagai perangkat lunak pembuatan 3D yang dilengkapi dengan *pipeline* 3D termasuk membuat, memodelkan, animasi, simulasi, dan pelacakan gerakan yang memungkinkan untuk pengeditan video dan membuat *game* (Adnan et al., 2020).



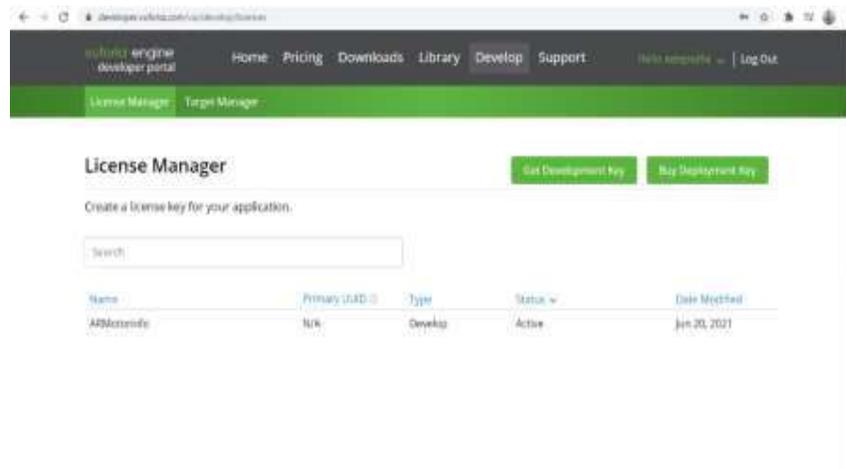
**Gambar 2.19** Logo *Blender*  
**Sumber :** (Data penelitian, 2021)

Pengolahan objek bermodelkan 3D seperti aplikasi *Blender* digunakan untuk membuat permodelan tiga dimensi terhadap objek yang sudah ditetapkan sebelumnya. Objek-objek tersebut akan dibentuk dengan sedemikian rupa mirip dengan bentuk aslinya. Ketika selesai melakukan proses edit maka objek-objek yang sudah dibuat lalu disimpan dengan menggunakan ekstensi *file fbx*, *obj* dan *3ds* yang akan di baca oleh aplikasi *Unity*.

### 2.3.6 Vuforia

*Vuforia* adalah sebuah pustaka untuk pengembangan *Augmented Reality Kit*. *Vuforia* SDK secara resmi disediakan melalui *vuforia website* (<https://developer.vuforia.com/>.) yang bisa didapatkan secara gratis dan dapat dipilih sesuai dengan kebutuhan perangkat lunak yang dibangun, diantaranya *Unity Extension*, *iOS*, atau *Android*. *Vuforia* memiliki fitur *Image Target* guna

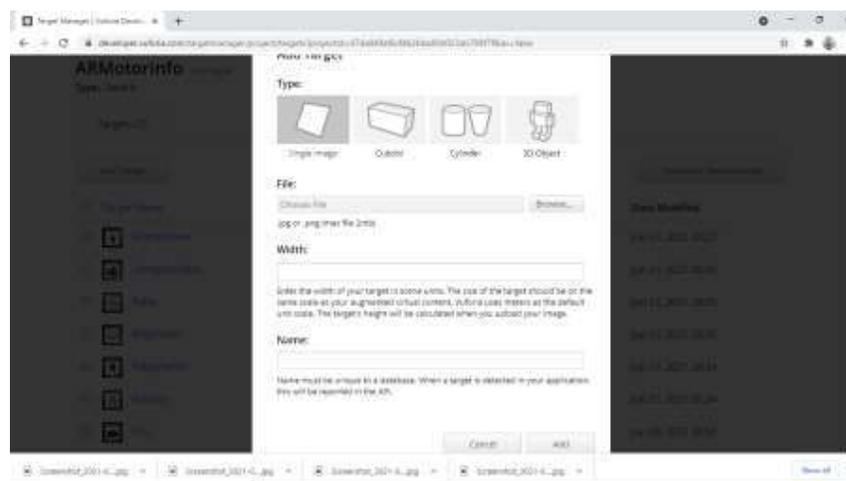
mendukung pengembangan aplikasi berbasis *Augmented Reality* (Rendi & Handoko, 2021).



**Gambar 2.20** Tampilan *website Vuforia engine*  
**Sumber :** (Data penelitian, 2021)

Ada 2 fitur utama didalam Vuforia yang menjadikan objek gambar sebagai database untuk mendukung teknologi *Augmented Reality*, sebagai berikut:

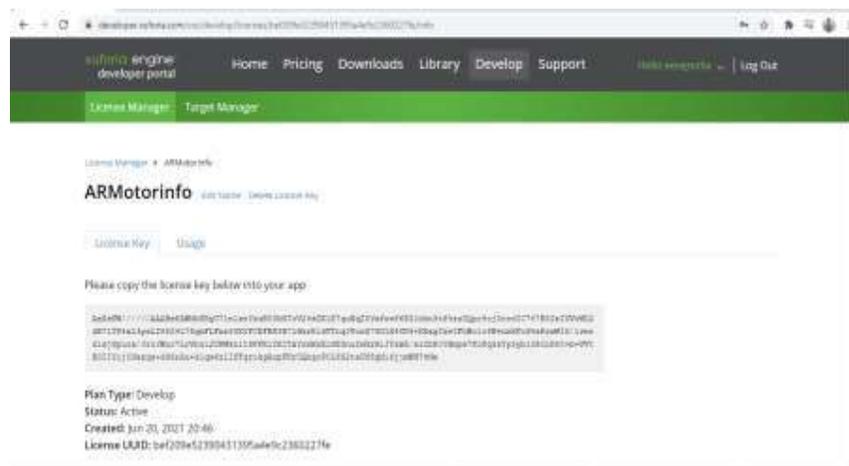
### 1. *Target Manager*



**Gambar 2.21** Halaman *input marker*  
**Sumber :** (Data penelitian, 2021)

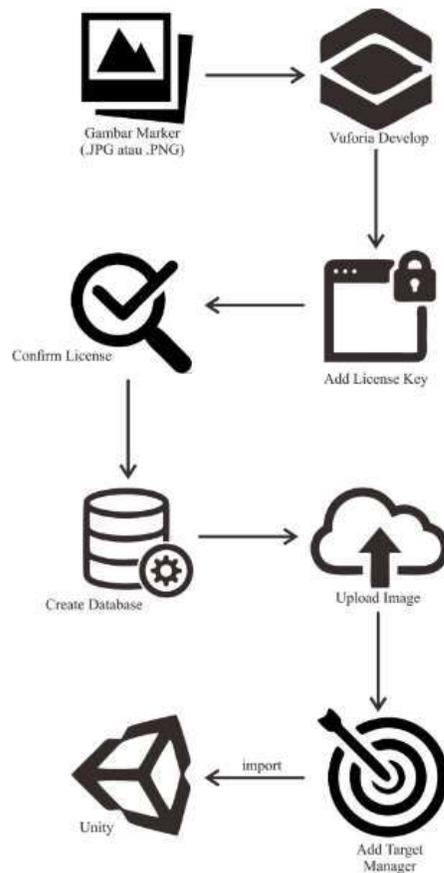
*Marker* haruslah dengan format .jpg atau .png. Proses menambahkan *marker* Dimulai “Add Target, Pilih *single image* untuk jenis *file* yang di unggah satu persatu. Pada kotak dialog pilih “choose file” lalu pilih *marker* yang sudah di tentukan dan secara satu persatu. Pilih “Width” untuk panjang *marker* dan pilih “Name” untuk nama *marker* kemudian pilih Add, *marker* sudah ditambahkan.

## 2. License Manager



**Gambar 2.22** Halaman *license manager*  
Sumber : (Data penelitian, 2021)

Pada fitur yang kedua ini adalah saat *marker* telah ditambahkan. Maka *vuforia* secara otomatis akan menandai *marker* tersebut dengan sebuah lisensi yang berfungsi sebagai kunci yang akan digunakan oleh *Unity* dalam memanggil *marker* yang sudah diunggah ke dalam *Vuforia*. Di mana *marker* tersebut sebagai *database* yang dapat diakses oleh *Unity* dalam menampilkan objek 3 dimensi.



**Gambar 2.23** Struktur dalam membuat *database Unity 3D*

**Sumber :** (Rendi & Handoko, 2021)

*Vuforia* melakukan masukan terhadap gambar 2D dari objek 3D. *Database* berupa gambar dengan *Vuforia* yang sebagai *library*. Pada *Vuforia* secara otomatis akan merekam dan menuliskan rincian *marker* dan menentukan *rating marker* dengan di tandai oleh banyak dan sedikitnya bintang. *Marker* yang sudah diupload ke dalam *website Vuforia* akan membentuk sebuah *Packet*. Selanjutnya *Packet* akan di download lalu di *import* ke dalam *Unity*, guna nantinya sebagai wadah untuk menampilkan objek 3D dengan memadukan antara dunia nyata dan *virtual*. Dapat dikatakan *Vuforia* adalah SDK yang merupakan *computer vision based* dari *Augmented Reality* (Rendi & Handoko, 2021)

### 2.3.7 Unity

*Unity* adalah sebuah mesin permainan berbasis *lintas-platform* untuk membangun aplikasi dan *game* yang berbasis komputer, *Android*, *iPhone*, *PS3*, dan *X-Box*. Karakteristik *game* di haruskan adanya sebuah *plugin*, yaitu *Unity Web Player* sama dengan *Flash Player* dibaca *browser*. Sebuah *tools* yang terintegrasi untuk dapat membuat model objek 3 dimensi untuk konteks interaktif seperti di *videogame*, visualisasi arsitektur dan animasi tiga dimensi (Rendi & Handoko, 2021).



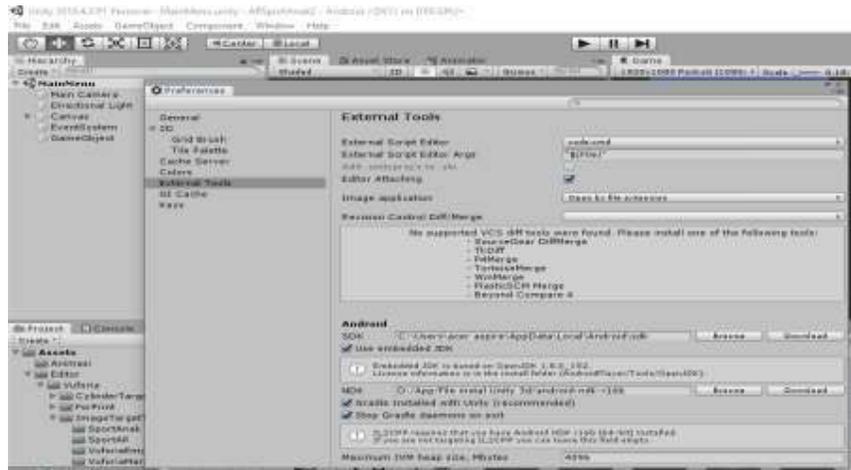
**Gambar 2.24** Logo *Unity 3D*  
**Sumber :** (Data penelitian, 2021)

Saat proses instalasi berjalan, maka akan dijumpai tampilan *custom*. Pada tampilan ini secara otomatis *Unity 3D* akan mengklaim bahwa pengembangan aplikasi *Unity 3D* akan menggunakan Bahasa pemrograman yang dikemukakan oleh perusahaan *Microsoft Visual Studio Community* yakni Bahasa pemrograman C# (*C-Sharp*). Lalu ada hal yang harus di ingat bahwa fitur *platforms* yang dicentang harus lah 2 fitur, yakni “*Android Build Support*” dan “*Vuforia Augmented Reality Support*” guna untuk bisa melakukan pengembangan aplikasi yang berbasis *android* dan sebagai konektor penghubung kepada *Vuforia Development Key* untuk pengembangan aplikasi berbasis *Augmented Reality*. Setelahnya lakukan proses instalasi sesuai dengan arahan.



**Gambar 2.25** Tampilan menu awal *Unity*  
**Sumber :** (Data penelitian, 2021)

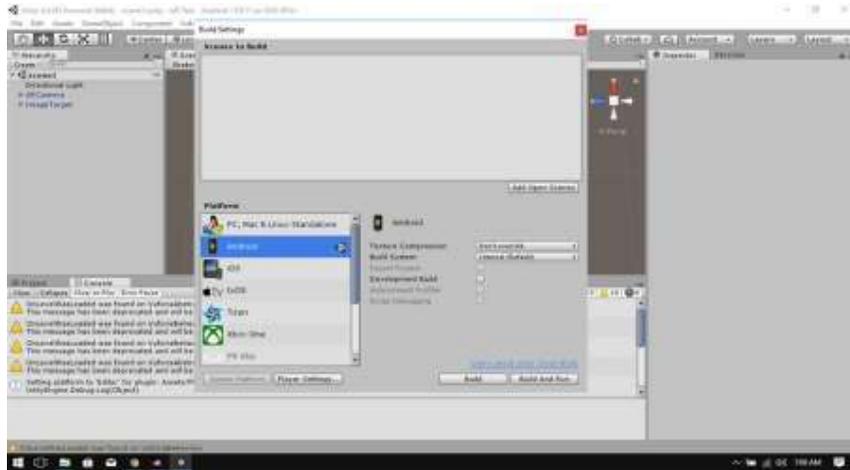
Ketika *Unity* selesai di install maka tidak bisa langsung digunakan untuk merancang aplikasi. Diharuskan juga menginstall aplikasi tambahan lain yang secara otomatis sudah disediakan fitur *Browse* dan *Download* oleh *Unity*. Dimulai dengan masuk ke *Unity* pilih menu *Edit-> Preferences-> External Tools*. Seperti gambar dibawah ini terdapat *setting* untuk mengubah atau mengarahkan SDK *Android* dan *NDK java* ke *Unity*. Jika sudah menginstall *Android Studio* dan *Java* tinggal klik *Browse* dan arahkan ke SDK *Android* dan *NDK Java* yang sudah di install. Akan tetapi jika belum maka bisa langsung *download* pada tombol *Download* yang disediakan. *File* akan berbentuk *.exe* yang mempunyai ukuran tidak jauh beda dengan SDK pada umumnya.



**Gambar 2.26** Tampilan untuk unduh SDK *Android* dan NDK *Java*  
**Sumber :** (Data penelitian, 2021)

*Android* SDK dan NDK *java* merupakan sistem operasi yang sangat penting untuk pengembangan aplikasi berbasis *android*. Hal ini dikarenakan SDK *Android* dan NDK *Java* memasukan sistem operasi *linux* untuk memberikan fungsi sistem yang terdapat pada *smartphone*. SDK *Android* termasuk *tools* yang dibutuhkan untuk pengembangan aplikasi *android*. Sedangkan NDK *Java* merupakan perangkat lunak yang dikembangkan *oracle* yang memiliki fungsi untuk melengkapi aplikasi yang ada di *android* (Erick & Ekawati, 2021).

Setelah selesai semua persiapan yang dibutuhkan maka untuk memulai melakukan pembuatan projek di mulai klik *File-> Build Setting*. Untuk pengembangan aplikasi berbasis *Android* maka pilih *Platform Android*. Setelah itu klik tombol *Switch Platform*. Secara otomatis akan masuk ke dalam pembuatan dan pengembangan aplikasi berbasis *android*.



**Gambar 2.27** Tampilan memulai proyek  
**Sumber :** (Data penelitian, 2021)

Dengan *Unity 3D* kita dapat membuat *game 3D*, FPS dan *game 2D* serta *Game Online*, fitur *Unity* lainnya adalah sebagai berikut: 1. Buat *Game 2D / 3D*. 2. Buat Aplikasi FPS, Simulasi dan Realita. 3. Buat *Game Online*. 4. Dukungan Konversi: Ponsel *Android, Iphone, Blackberry, Windows, Linux, Flash, Webplayer*. 5. Publikasikan *Google Play Online*, pasar *Android*. 6. Kode Dukungan: C #, *Javascript* dan *Boo*. 7. Mendukung ekstensi *file, 3ds, obj, fbx* (Nugroho & Pramono, 2017).

## 2.4 Penelitian Terdahulu

Penelitian terdahulu akan di jadikan bahan perbandingan dan referansi dalam melakukan penelitian guna sebagai sarana pendukung penelitian:

1. (Haris et al., 2018) *Evolution Of Android Operating System: A Review*, ISBN: 978-0-6481172-0-9. Evolusi dari ponsel yang dikembangkan oleh Google dulunya hanya digunakan untuk melakukan panggilan hal ini menambah fungsi dengan diperkenalkanya telepon pintar dengan didukung

oleh sistem operasi untuk ponsel yang memiliki penyimpanan dan pemrosesan sendiri. Pada saat sekarang ponsel telah mempunyai fungsi mulai dari menelepon dan mengirim pesan teks, hingga multimedia, *email*, aplikasi sosialisasi, pengolah kata, lembar kerja (*Word*, *Excel* dan lainnya) dan mendukung untuk bermain *multiplaying*. Setiap dikeluarkannya versi dari OS *Android* akan mengevaluasi untuk meningkatkan kemampuan beradaptasi dan memenuhi kebutuhan pengguna. Fokus *google* dalam mengembangkan sistem operasi android harus mengutamakan keamanan dan privasi yang merupakan ancaman utama bagi sistem operasi dan memastikan kerentanan terhadap sistem operasi dari para penyusup.

2. (Adnan et al., 2020) ***Analysis development of augmented reality in android-based computer learning in vocational schools***, ISSN: 1757-899X. Penerapan teknologi *Augmented Reality* guna mendukung pembelajaran yang interaktif bagi siswa sehingga membuat proses belajar menjadi menyenangkan, menantang, memotivasi dan memberikan kesempatan untuk mengembangkan kreativitas dan kemandirian siswa. Semakin banyak variasi media pembelajaran yang dibangun untuk menunjang kegiatan pembelajaran disekolah diharapkan dapat meningkatkan pengetahuan bagi siswa.
3. (Rendi & Handoko, 2021) **Penerapan Augmented Reality Pengenalan Jenis Olahraga Berbasis Android**, ISSN: 2715-6265. Teknologi *Augmented Reality* yang ditanamkan ke dalam sebuah *mobile phone* memberikan pengalaman baru untuk penggunaan media pembelajaran. Pengalaman baru yang dirasakan adalah proses pengenalan jenis olahraga mengalami

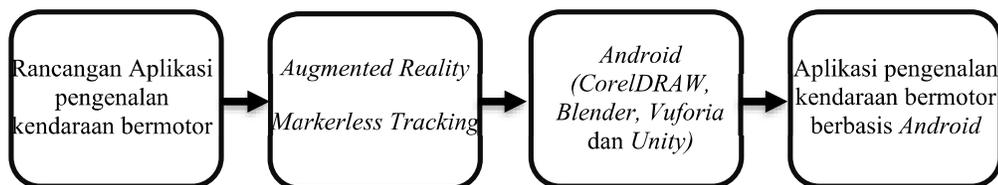
pembaharuan dari yang dulunya menggunakan media buku teks dan bergambar sehingga mengalami perubahan menjadi bentuk 3D yang ditampilkan pada waktu dan lingkungan secara *realtime*. Bentuk dari penerapan teknologi AR menjadi sebuah aplikasi yang dapat dijadikan sebagai alat peraga tepat guna yang dapat di akses secara langsung oleh siswa.

4. (Christoffel et al., 2019) **Aplikasi Augmented Reality Pengenalan Rambu Lalu Lintas menggunakan Metode *User Defined Target***, ISSN: 2685-6131. *Markerles* termasuk teknik deteksi pengembangan dari teknologi *Augmented Reality* dengan metode yang digunakan yakni *User Defined Target*. Metode ini dalam melakukan pelacakan objek dunia nyata dengan menggunakan tahapan pengenalan pola (*Pattern Recognition*) pada objek dunia nyata lalu menampilkannya ke dalam dunia maya. Aplikasi *Augmented Reality* dengan objek pengenalan rambu lalu lintas menggunakan metode *User Defined Target* dapat memberikan informasi dari ke 4 jenis rambu lalu lintas seperti rambu petunjuk, rambu peringatan, rambu larangan dan rambu perintah.

## 2.5 Kerangka Pemikiran

Metode kerangka pemikiran merupakan model teori yang mendefinisikan sebuah masalah menjadi hal sesuatu fokus utama. Proses mengidentifikasi kerangka pemikiran dalam pengenalan kendaraan bermotor dengan melakukan wawancara langsung, untuk nanti di buat kedalam media yang menarik seperti menggunakan *smartphone* yang kemudian dipasang aplikasi pengenalan kendaraan bermotor yang

menerapkan teknologi AR dengan masukan objek 3D, lalu pada *smartphone* dapat di lihat secara nyata. Pada gambar di bawah ini adalah bagan untuk kerangka pemikiran:



**Gambar 2.28** Kerangka pemikiran

**Sumber :** (Data penelitian, 2021)

1. Pada kerangka pemikiran dapat digambarkan pencarian sumber pustaka melalui media buku, jurnal ilmiah dan observasi. Media buku dan jurnal di khususkan mempunyai kaitan dengan teknologi *Augmented Reality*, Teknik *Markerless Tracking* dan materi *sparepart* motor serta melakukan obeservasi langsung dengan cara wawancara kepada pengelola bengkel *Motor Speed Factory*.
2. Teknologi yang digunakan adalah teknologi *Augmented Reality* di mana teknologi ini teknologi yang sedang berkembang yang mendukung untuk setiap bagian seperti pengenalan dan pemasaran, teknologi *Augmented Reality* melakukan penggabungan dunia nyata dan dunia *virtual*. *Markerless tracking* menjadi Teknik deteksi yang akan digunakan untuk menemukan objek nyata dan menampilkan objek tersebut ke dalam bentuk tiga dimensi.
3. Perancangan aplikasi menggunakan beberapa *software* seperti *CorelDraw* guna untuk membuat *marker*. Aplikasi *Blender* untuk pengolahan Objek 3D, setiap *sparepart* motor tersebut akan di jadikan menjadi objek dengan bentuk

menjadi 3 dimensi. *Vuforia* sebagai *library* atau database untuk menyimpan *marker* guna mendukung untuk menampilkan jenis olahraga dalam bentuk tiga dimensi dan *Unity 3D* sebagai proses utama dalam membangun aplikasi *Augmented Reality* yang berperan sebagai perancangan aplikasi dan masukan kode program.

4. Aplikasi yang dihasilkan yaitu sebuah aplikasi yang menerapkan teknologi *Augmented Reality* untuk menampilkan jenis kendaraan bermotor yakni *sparepart* motor beat dalam bentuk tiga dimensi di rancang dengan menarik untuk kemudian dilakukan implementasi kepada bengkel motor dan masyarakat umum.