

BAB II

TINJAUAN PUSTAKA

2.1. Teori Dasar

2.1.1. *Augmented Reality*

Augmented reality ialah sebuah teknologi yang bekerja sama secara langsung atau implikasi dari lingkungan yang nyata dengan menambahkan komputer virtual yang menghasilkan data. AR merupakan dua jenis inovasi cerdas dan terdaftar dalam 3D serta menyatukan benda asli dan virtual (Wiharto & Budihartanti, 2017). Sedangkan pendapat lainnya, *Augmented reality* ialah sebuah gagasan untuk melapisi substansi visual dimana sebuah inovasi untuk menyatukan benda virtual baik 2D maupun 3D ke dalam keadaan nyata lalu memproyeksikan objek-objek virtual ini dalam waktu sebenarnya (Kamelia, 2019).

Berdasarkan teori diatas, maka dapat diambil kesimpulan AR merupakan sebuah teknologi yang digunakan untuk berinteraksi secara langsung maupun secara tidak langsung dengan untuk memperoleh informasi berbasis teknologi komputer virtual baik dalam bentuk 3D maupun 2D.

2.1.2. Media Pembelajaran

Menurut (Steffi Adam & Muhammad Taufik Syastra, 2015) media pembelajaran ialah semua yang baik bersifat fisik maupun khusus dalam siklus pembelajaran yang bisa digunakan oleh pendidik dalam penyampaian materi kepada siswa sehingga dapat tercapai target pembelajaran yang sudah dirinci. (Hutabri et al., 2019) Berpendapat bahwa media pembelajaran merupakan

perantara yang memiliki tujuan untuk menyampaikan materi dalam proses pembelajaran. Selanjutnya (Joni Purwono, dkk, 2015) mendeskripsikan media pembelajaran yaitu mempunyai peran yang signifikan dalam mendukung sifat mendidik dan mengukur pembelajaran. Media dapat menjadikan kegiatan pembelajaran menjadi sangat menarik dan menyenangkan, Media pembelajaran sedang berkembang saat ini ialah media yang bervariasi.

Dari uraian di atas dapat disimpulkan yaitu cenderung mendefinisikan bahwa media pembelajaran merupakan suatu alat yang dimanfaatkan membantu pendidikan dan pembelajaran mengukur daya renungan, sentimen, pertimbangan dan kapasitas atau kemampuan siswa sehingga dapat memberdayakan siklus belajar.

2.1.2.1. Fungsi Media Pembelajaran

Media pembelajaran adalah perangkat khusus yang dimanfaatkan pada ukuran pendidikan dan pembelajaran. Sebagai perangkat khusus, seperti yang ditunjukkan oleh Oemar Hamalik, media pembelajaran memiliki kapasitas yang luas antara lain (Tafonao, 2018)

1. Kapasitas edukatif media korespondensi, khususnya bahwa setiap gerakan media korespondensi mengandung sifat instruktif karena memiliki dampak instruktif.
2. Kapasitas sosial media korespondensi, media korespondensi memberikan data asli dan pertemuan di berbagai bidang aktivitas publik individu.
3. Kapasitas praktis media komunikasi, media surat menyurat dapat dimanfaatkan secara serius di bidang penyalur dan industri.

4. Kapasitas politik media korespondensi, dalam bidang politik, media korespondensi dapat bekerja terutama dalam perbaikan isu-isu pemerintahan, baik material maupun mendalam.
5. Kapasitas karya dan budaya adalah media media komunikasi, kemajuan ekspresi pengalaman dan budaya manusia dapat disebarkan melalui media komunikasi.

2.1.2.2. Media Pembelajaran Interaktif

Media Pembelajaran Interaktif ialah media yang bisa menyebabkan siswa beradaptasi secara bebas tanpa adanya tugas seorang pendidik dan siswa dapat bekerja sama dengan media seperti berkomunikasi dengan seorang pengajar. Media pembelajaran interaktif bisa berupa perangkat lunak dan perangkat keras yang dimanfaatkan sebagai mediator untuk penyampaian materi kepada siswa. (Sutarti, T & Irawan, 2017).

2.1.2.3. Kriteria Media Pembelajaran

Menurut (Ambiyar, 2018) kriteria dalam pemilihan media dibagi menjadi tiga bagian yaitu:

1. Kelayakan praktis

Pada pemilihan media dimanfaatkan karena praktis yaitu: pertama pengalaman instruktur dari berbagai media, kedua aksesibilitas media lingkungan, ketiga aksesibilitas waktu dalam merencanakan, keempat aksesibilitas kantor dan dukungan. Kemungkinan khusus ketercapaian teknis, penentuannya dengan memenuhi prasyarat subjektif atau terlepas dari apakah alat tersebut dapat memperkuat dan menegakkan ukuran pembelajaran siswa.

2. Kelayakan biaya dari beberapa standar atau langkah dalam memilih media yang disampaikan oleh para ahli diatas, maka dibuat kesimpulan beberapa pertimbangan yang perlu diperbuat oleh pengajar dalam memilih media untuk lebih spesifiknya:
 1. Evaluasi para siswa
 2. Memperkirakan target pembelajaran
 3. Evaluasi tentang prosedur belajar
 4. Evaluasi kapasitas dalam membangun dan memanfaatkan media
 5. Evaluasi tentang biaya
 6. Evaluasi tentang kebutuhan
 7. Evaluasi tentang kecakapan dan kecukupan.
3. Kelayakan teknis Kelayakan teknis, penentuannya prasyarat subjektif (kualitas) atau terlepas dari apakah media tersebut dapat memajukan dan mendukung ukuran pembelajaran siswa. Ada dua jenis nilai yang perlu dipertimbangkan, untuk lebih spesifiknya:
 1. Kualitas pesan (kurikulum): pertama signifikansi terhadap tujuan/fokus pembelajaran, kedua menunjukkan strategi, ketiga tidak sulit untuk diolah/dicapai, dan keempat sistematika yang diperhitungkan.
 2. yang mengikuti standar representasi, standar ini adalah alasan untuk rencana atau format visual sebagai berikut, untuk menjadi spesifik kemegahan, kejernihan dan keterbacaan dan keseimbangan.

2.1.3. *Android*

Android adalah sebuah sistem kerja yang sifatnya terbuka terhadap *device* ponsel yang bergantung pada kerangka kerja Linux (Nazaruddin, 2012). Juli 2005 *Android* diperoleh oleh *Google* dan diluncurkan pada November 2007. Pada pengembangan 19 aplikasi *Android* memberikan *Android SDK* yang tersedia sebuah perangkat dan API (*application programming interface*) (Kasus et al., 2017).

Dari uraian pernyataan diatas dapat disimpulkan *Android* adalah sebuah *system* untuk *smartphone* berbasis Linux dengan menggabungkan kerangka kerja dan aplikasi. *Android* memberikan kebebasan kepada para pengembang untuk membuat *software*.

2.1.3.1. Versi *Android*

Android ialah *system mobile* yang bersifat terbuka mempunyai banyak versi dalam perkembangannya. Beberapa versi dari *Android* yang didapatkan dari situs *android* sebagai berikut. (Lengkong et al., 2015)

1. *Android 1.5 Cupcake*



Gambar 2.1. *Android Cupcake*
Sumber: (Ramadhani, 2019)

Android Cupcake merupakan versi *android* yang memulai pemanfaatan nama pada jenis makanan. Tanggal 30 April 2009 versi *Android* inilah yang

dikenalkan secara meluas. Versi ini adalah angkatan awal yang diluncurkan secara komersial dan dimulai dari inilah *Android* mulai memanfaatkan jenis nama makanan untuk berbagai versi *Android* yang dikeluarkannya. Fitu-fitur yang ada pada versi ini ialah adanya mendukung rotasi layar otomatis, *widget*, dan papan ketik virtual. (Lengkong et al., 2015)

2. *Android* 1.6 Donut



Gambar 2.2 *Android* Donut
sumber: (Ramadhani, 2019)

Android versi ini dikeluarkan pada 15 September 2009. Dalam bentuk ini terdapat peningkatan dan perbaikan, beberapa diantaranya adalah perbaikan pada *error reboot*, pembaruan pada antarmuka foto dan video dan penggabungan pencarian yang lebih unggul dari versi *android* sebelumnya. (Lengkong et al., 2015)

1. Proses pencarian yang lebih unggul dibandingkan versi awal.
2. Antarmuka dari galeri yang mudah digunakan.
3. Rotasi putaran pada layar WVGA.
4. *Android store* dan *apps* yang dikembangkan lebih lanjut dari versi awal
5. Pembaruan ke versi 2.1 (*Eclair*).

3. *Android* 2.0/2.1 *Eclair*



Gambar 2.3 *Android Eclair*
Sumber: (Ramadhani, 2019)

Di versi ini telah dilakukan pembaruan untuk melakukan perbaikan kesalahan dan memperbanyak fitur di tanggal 26 Oktober 2009. Versi ini mendukung fitur-fitur seperti multi *touch*, *live wallpaper*, tampilan *inteface* yang berbeda dan mendukung browser pada HTML5. (Lengkong et al., 2015). Beberapa fitur lain yang di *upgrade* pada *Android Eclair* meliputi:

1. *Camera* 3,2 MP yang mendukung adanya "*flash*"
 2. Tampilan *contact* baru yang menarik.
 3. Adaptasi HTML5 ini yang mendukung perbedaan antar muka dengan program terbaru
 4. *Bluetooth* 2.1 yang *speed* pemindahan dokumen lebih diatas.
4. *Android 2.2 Froyo*



Gambar 2.4 *Android Froyo*
Sumber: (Ramadhani, 2019)

Versi ini telah dirilis pada 20 Mei 2010 *Android*. Dalam versi ini *Android* telah dikenal secara umum oleh vendor. *Upgrade Android 2.2* menghadirkan fitur-fitur yang unggul seperti *zoom* gambar yang ada dalam galeri dengan pergerakan, USB yang telah ditingkatkan dan dikembangkan lebih lanjut, juga WIFI area serta mendukung adanya animasi GIF pada *browser* internet. (Lengkong et al., 2015)

Beberapa fitur menarik dari *Android* Froyo sebagai berikut:

1. Kamera dapat merekam dengan kualitas tinggi
2. Dapat menempatkan aplikasi pada kartu MMC/SD
3. Dapat berbagi jaringan
4. Meningkatnya performa
5. Kemampuan pembaruan otomatis di *Android* Market.

5. *Android* 2.3 *Gingerbread*



Gambar 2.5 *Android* *Gingerbread*
Sumber: (Ramadhani, 2019)

Versi ini telah menjadi kerangka kerja portabel yang terkenal. Antisipasi pada pabrik manufaktur *Samsung* dalam pembuatan item *Samsung Galaxy Series* semakin meningkatkan ketenaran *Android*. Versi ini diluncurkan pada 6 Desember 2010. (Lengkong et al., 2015) Fitur- fitur menarik *Gingerbread* antara lain:

1. UI yang tidak boros terhadap energi
2. *Keyboard* virtual dengan penentuan kata
3. *Power management*
4. Control aplikasi
5. Mendukung *multiple* kamera.

6. *Android 3.0 Honeycomb*



Gambar 2.6 *Android Honeycomb*

Sumber: (Ramadhani, 2019)

Versi *Android* ini merupakan versi yang diluncurkan sejak 22 Februari 2011. Versi ini diutamakan untuk *device tablet PC*, dengan memberikan fitur-fitur yang berfokus pada pemrosesan bisnis di *tablet*. Fitur tambahan dari *Android* ini seperti *support processor multi-pusat*, *support* panggilan video menggunakan *Google Talk* dan peningkatan kecepatan untuk bertukar aplikasi yang sedang bekerja dengan melakukan beberapa tugas aplikasi. (Lengkong et al., 2015)

7. *Android 4.1 Jellybean*



Gambar 2.7 *Android Jellybean*

Sumber: (Ramadhani, 2019)

Android Jellybean awal dirilis pada bulan Juli 2012, yang berbasis *Linux* Kernel. Pengembangan versi *Jellybean* lebih berpusat pada pengerjaan tampilan *user interface*. Fitur tambahan yang dipasang pada versi *Jellybean* adalah *keyboard* yang bisa disesuaikan dari *user* dan mendukung pergerakan pada *keyboard*, antarmuka pengguna yang lebih halus, mendukung tampilan tanpa kabel, *widget* yang dapat diubah dan dapat disesuaikan ukurannya. Versi 4.1 juga

menyertakan fitur papan ketik virtual yang lebih baik. Baterai pada sistem *Android Jellybean* versi 4.1 dinilai sangat produktif. (Lengkong et al., 2015)

8. *Android 4.4 KitKat*



Gambar 2.8 *Android Kitkat*
Sumber : (Ramadhani, 2019)

Android ini diluncurkan oleh Google tanggal 3 september 2013. Versi *Android* ini lebih mengembangkan peningkatan dengan menghasilkan *highlight* yang berkualitas. Fitur tambahan dari versi ini seperti *WebViews* berbasis *Chromium*, peningkatan performa pada *device* yang spesifikasinya rendah, *support* akan penggabungan sensor dan *step detector*. (Lengkong et al., 2015)

1. Fitur *SMS* dikoordinasikan langsung ke aplikasi *Google Hangouts*.
2. Adanya fitur *Cloud Printing*, yang memungkinkan *user* bisa mencetak tanpa kabel / perintah yang dikirim ke komputer yang tersambung dengan printer.
3. *Design icon* dan *theme* yang menarik dan nyata.
4. Menggunakan perintah suara dari *Google Now* tanpa banyak menggunakan daya baterai
5. *Navigation* dan *status bar* telah diperbarui.
6. Antarmuka yang lebih mulus.
7. Bisa membuka *camera* saat *screen* terkunci.

9. *Android Lollipop 5.0*



Gambar 2.9 *Android Lollipop*

Sumber: (Ramadhani, 2019)

Android versi berikutnya ialah versi yang diluncurkan tanggal 25 Juni 2014.

Dalam versi ini *Android* tidak lagi berpatokan pada ponsel, tetapi juga bekerja pada *system* lain seperti *Android TV* dan juga *Google Fit*.(Lengkong et al., 2015)

Ada beberapa fitur tambahan di *Android Lollipop* antara lain:

1. Kinerja yang lebih dinamis, kinerja lebih cepat, lebih responsif dan lebih mulus dengan tampilan simbol *3D* yang menarik bila dibandingkan dengan versi *android* sebelumnya.
2. Peringatan dimana lebih jelas terlihat, mudah digunakan, dan bisa untuk di *setting*.
3. Aplikasi dokumen sentris dimana menghadirkan pembaruan *Overview Space* yang lebih beradaptasi dan bermanfaat untuk melakukan banyak tugas.
4. Inovasi API terbaru mengharuskan *user* untuk menampilkan kegiatan yang dalam aplikasi sebagai *file* individual bersama *screen* terbaru lainnya.

5. Ketersediaan yang modern yaitu *Android Lollipop* meningkatkan inovasi API lain yang memungkinkan aplikasi bekerja secara sekaligus pada *Bluetooth Low Energy (BLE)*.
6. Performa grafis yang ditingkatkan
7. Kualitas suara yang lebih baik
8. Kualitas kamera dan video yang dikembangkan lebih lanjut
9. Baterai lebih hemat dan dapat menyesuaikan pemakaian baterai dengan cara menyusun jalannya aplikasi tertentu.
10. Mendukung *USB OTG*

10. *Android Marshmallow 6.0*



Gambar 2.10 *android marshmallow*

Sumber: (Ramadhani, 2019)

Android ini di luncurkan tanggal 5 Mei 2015. Fitur yang ada dalam versi ini ialah *support* akan sensor *finger print* untuk menggunakan ponsel, fasilitas untuk mengakses aplikasi yang berbeda pada desain *screen* yang *support* banyak jendela, *support platform virtual reality*, dan kapasitas untuk memperkecil penggunaan *bandwidth* pada mode *data saver*. (Lengkong et al., 2015)

1. Sensor *finge print*.
2. Pilihan kata yang dikembangkan lebih lanjut.
3. Pengaturan volume yang dikembangkan lebih lanjut.
4. Fitur peluncur *google now* terbaru.

5. Kemudahan pengaturan otorisasi aplikasi.
6. *Doze and* isi daya.
7. Dapat menghapus beberapa aplikasi *default* untuk memuaskan pengguna.
8. Pertukaran langsung *game / software* antar memori (*internal & SD Card*).

11. *Android Nougat 7.0*



Gambar 2.11 *Android Nougat*
Sumber: (Ramadhani, 2019)

Dirilis tanggal 19 Oktober 2016 *Android* ini tertuju untuk pengerjaan kinerja antarmuka pengguna sehingga akan lebih perseptif dan pemanfaatan aplikasi secara bersamaan yang banyak dalam fitur multi *window*. *Android* ini juga *upgrade* beberapa penambahan fitur seperti *support night light* atau *night mode*, papan ketik bawaan yang bisa untuk mengirim animasi *GIF* secara langsung. (Lengkong et al., 2015)

1. Fitur *Daydream* yang bertujuan *Virtual Reality*
2. Lebih sigap melakukan berbagai tugas
3. Multi jendela dan *Double tab chrome*
4. Fitur data *saver* yang dapat mengurangi pemanfaatan data internet
5. Editor pengaturan cepat

6. Tombol *cancel* pada saat mengunduh.

12. *Android Oreo* 8.0



Gambar 2.12 *Android Oreo*
Sumber: (Ramadhani, 2019)

Versi *Android* ini dirilis pada Agustus 2017. Antar muka pada *Android* ini yang lebih sederhana mempermudah pengguna untuk menggunakan aplikasi. *Upgrade* pada versi *Oreo* terdapat fitur-fitur tambahan seperti fitur *Autofill* yang memudahkan untuk melengkapi struktur, seperti dukungan gambar dalam gambar dan peningkatan *boot* untuk membuatnya lebih cepat. (Lengkong et al., 2015)

13. *Android* 9.0 Pie



Gambar 2.13 *Android 9.0 Pie*
Sumber: (Ramadhani, 2019)

Android ini adalah versi dikeluarkan pada Agustus tahun 2018. Elemen yang unggul dari versi ini ialah kompetensi dari AI. Dari inilah ponsel bisa mengetahui dan menganalisa cara pemakaian *user* menggunakan ponsel secara *automatic*. Terdapat fitur-fitur yang dihadirkan seperti *Adaptive Brightness* yang dapat mengubah penerangan *screen* secara *automatic* dan mendukung ponsel *bezel less*(Lengkong et al., 2015).

14. *Android* 10

android 10

Gambar 2.14 *Android* 10
Sumber: (Ramadhani, 2019)

Nama *Android* terbaru berikut ini tidak dinamai seperti sebelumnya yang terkait oleh makanan manis. Versi ini dinamai *Android* 10 untuk mengingat akan *Android* sudah mencapai 10 tahun secara meluas. Versi ini telah menambah perbaikan pada mode malam atau gelap dan pembaruan fitur *speaker* suara yang dapat mengubah kualitas audio. (Lengkong et al., 2015)

2.2. Teori Khusus

2.2.1. Klasifikasi Makhluk Hidup

Merupakan metode pengumpulan makhluk hidup tergantung pada kualitas dan atribut yang sama. Pengumpulan hasil keteraturan pada berbagai tingkatan atau dalam berbagai takson disebut kategorisasi ilmiah. Semakin tinggi levelnya, semakin banyak individunya, namun semakin sedikit individu tersebut tetapi kesamaan dari sifat anggotanya semakin sedikit (Anshori & Martono, 2013).

2.2.1.1. Tujuan Klasifikasi Makhluk Hidup

Klasifikasi menurut (Anshori & Martono, 2013) difungsikan sebagai sebagai alat dalam mempelajari keanekaragaman hayati. Tujuan atas adanya klasifikasi yaitu seperti dibawah ini:

1. Memperbaiki objek belajar agar lebih mudah untuk dipelajari.
2. Menggambarkan sifat-sifat makhluk hidup untuk mengenali setiap jenisnya.

3. Mengumpulkan makhluk hidup yang bergantung pada perumpamaan dalam kualitas mereka.
4. Mengetahui hubungan dan sejarah perkembangannya.

Keberadaan makhluk hidup memiliki kegunaan yang sangat banyak yang dapat dirasakan oleh manusia, lebih spesifiknya sebagai berikut:

1. Pengklasifikasian melalui pengumpulan dapat mempermudah untuk mempertimbangkan berbagai bentuk kehidupan.
2. Klasifikasi dimanfaatkan untuk melihat tingkat hubungan keterkaitan antara alam dan lainnya.

2.2.1.2. Tahapan Klasifikasi

Dalam klasifikasi makhluk hidup menurut (Anshori & Martono, 2013) penting melalui beberapa tahapan antara lain:

1. Pengamatan sifat makhluk hidup

Tahap ini adalah tahap pertama klasifikasi, dalam tahap dilakukan identifikasi terhadap antar makhluk hidup. Memperhatikan dan mengumpulkan berdasar dari perilaku, morfologi, struktur kehidupan, dan fisiologi.

2. Pengelompokkan makhluk hidup berdasarkan pada ciri-ciri

Hasil tahap analisa kemudian ditingkatkan pada pengumpulan makhluk hidup. Pada proses ini ciri dan karakter atau kesamaan dan perbedaan makhluk hidup yang diperhatikan.

3. Penamaan makhluk hidup

Penamaan makhluk hidup adalah suatu peran penting dalam klasifikasi. Terdapat berbagai macam cara pemberian nama makhluk hidup, yaitu dengan

sistem tata nama ganda (*binomial nomenclature*). Dari adanya penamaan makhluk hidup ini, kualitas dan karakter makhluk hidup tidak sulit untuk dijangkau.

2.2.1.3. Hewan Vertebrata

Seperti namanya, hewan vertebrata adalah hewan yang mempunyai ciri khas utama berupa tulang belakang. Jenis tulang ini dibentuk khusus untuk membentuk tulang punggung, yang berfungsi untuk melindungi sumsum tulang belakang dan menyambungkannya pada system saraf. Mereka juga mempunyai tengkorak untuk melindungi otak.

Tabel 2.1 Klasifikasi Hewan Vertebrata Mamalia

No	Mamalia	Nama Ilmiah
1	Anjing	<i>Canis lupus familiaris</i>
2	Badak Jawa	<i>Rhinoceros sondaicus</i>
3	Badak Sumatera	<i>Dicerorhinus sumatrensis</i>
4	Gajah sumatera	<i>Elephas maximus sumatranus</i>
5	Harimau Sumatera	<i>Panthera tigris sumatrae</i>
6	Harimau Bali	<i>Panthera tigris balica</i>
7	Harimau Jawa	<i>Panthera tigris sondaica</i>
8	Kambing Hutan Sumatera	<i>Capricornis sumatraensis</i>
9	Kambing Ternak	<i>Capra aegagrus hircus</i>
10	Kelinci Jawa	<i>Lepus nigricollis</i>
11	Kelinci Sumatera	<i>Nesolagus netscheri</i>
12	Kucing	<i>Felis catus</i>
13	Rusa Bawean	<i>Axis kuhlii</i>
14	Rusa Sambar	<i>Cervus unicolor</i>
15	Rusa Timor	<i>Cervus timorensis</i>
16	Rusa Toto1	<i>Axis axis</i>
17	Sapi Ternak	<i>Bos taurus</i>

Sumber: (Ruslan et al., 2016)

Tabel 2.2 Klasifikasi Hewan Vertebrata Ikan

No	Ikan	Nama Ilmiah
1	Belut	<i>Monopterus albus</i>
2	Ikan cupang	<i>Trichaptis vittatus</i>
3	Ikan hiu	<i>Carchareus leucas</i>
4	Ikan mas	<i>Carrasius auratus</i>
5	Ikan pari	<i>Dasyatis sp</i>
6	Ikan tuna sirip kuning	<i>Thunnus albacores</i>
7	Ikan lele local	<i>Clarias bathracus</i>

Sumber: (Ruslan et al., 2016)

Tabel 2.3 Klasifikasi Hewan Vertebrata Burung

No	Burung	Nama Ilmiah
1	Burung kasturi raja	<i>Psittrichas fulgidus</i>
2	Merpati	<i>Columba livia</i>
3	Bebek laut	<i>Esacus magnirostris</i>
4	Burung rajawali/elang	<i>Haliastur indus</i>
5	Bangau hitam	<i>Ciconia episcopus</i>
6	Burung beo	<i>Gracula religiosa</i>

Sumber: (Ruslan et al., 2016)

Tabel 2.4 Klasifikasi Hewan Vertebrata Amfibi

No	Amfibi	Nama Ilmiah
1	Kadal	<i>Mabouya multifasciata</i>
2	Katak hijau	<i>Rana macrodon</i>
3	Katak emas Panama	<i>Rachoporus renwardii</i>

Sumber: (Ruslan et al., 2016)

Tabel 2.5 Klasifikasi Hewan Vertebrata Reptil

No	Reptil	Nama Ilmiah
1	Buaya Siam	Crocodylus siamensis
2	Buaya Irian	Crocodylus novaeguineae
3	Iguana	Dermochelys coriacea
4	Komodo	Varanus komodoensis.
5	Penyu Pipih	Natator depressus
6	Kadal	Tiliqua gigas
7	Ular kobra	Naja sputatrix
8	Ular piton	Phyton reticulatus

Sumber: (Ruslan et al., 2016)

2.2.1.4. Flora Asiatis

Flora yang tumbuh di bagian barat Indonesia seperti Sumatera, Jawa, Bali dan Kalimantan dikenal sebagai Flora Asiatis, inilah yang menyebabkan tumbuhan yang hidup di daerah ini menerima pengaruh dari daratan Asia atau seperti tumbuhan yang hidup di Benua Asia. Oleh karena hal itu, nama flora ataupun fauna di daerah ini beri nama Asiatis, nama ini didapat dari daerahnya.(Kusmana & Hikmat, 2015)

Menurut (Kusmana & Hikmat, 2015) Flora Asiatis mempunyai sifat berbeda dengan flora bagian tengah ataupun timur. Flora di setiap wilayah mempunyai karakteristik tersendiri yang membuatnya berbeda dari flora di wilayah lain. Adapun karakteristik dari flora Asiatis seperti dibawah ini:

1. Flora Berupa Hutan Tropis

Flora Asiatis biasanya mengisi hutan tropis yang mempunyai beragam jenis tumbuhan atau pohon. Pohon-pohon yang memenuhi hutan tropis bagian barat Indonesia umumnya mempunyai kriteria tebal dan lembab. Beberapa jenis

tumbuhan Asiatis seperti tumbutan kamper, keruing, kayu meranti dan juga kayu mahoni.

2. Garis Khayal *Wallace* yang membatasi

Antar area yang membatasi kawasan penyebaran flora dan fauna dibatasi oleh garis yang tidak ada. Garis yang tidak ada ini membatasi antara kawasan Asiatis dan kawasan peralihan disebut garis *Wallace*. Sementara itu, batas antara kawasan peralihan dengan kawasan Australis dikenal sebagai garis *Weber*.

3. Mempunyai Daun Yang Lebat

Salah satu karakteristik yang dimiliki flora Asiatis adalah mempunyai daun yang begitu lebat. Jumlah daun pada tumbuhan ini dapat dikatakan rimbun dan lebat, bagian barat Indonesia sering ditemui hutan hujan tropis yang selalu rimbun dan lembab seiring berjalannya waktu karena daunnya membentuk penutup sehingga sinar matahari tidak bisa masuk.

4. Berukuran Besar

Karakteristik flora Asiatis lainnya ialah berukuran besar. Ukuran tumbuhan ini tentunya jauh lebih besar dibandingkan jenis tumbuhan lainnya. Terdapat beberapa tumbuhan atau flora Asiatis yang ukurannya sangat besar yaitu bunga *Rafflesia Arnoldi*, dan daun Sang.

Tabel 2.6 Klasifikasi Flora Asiat

N0	Nama Flora	Nama Ilmiah
1	Tumbuhan Kamper	Cinnamomum camphora
2	Kayu Meranti	Shorea leprosula Miq
3	Kayu Mahoni	Swietenia Macrophylla
4	Bunga Rafflesia Arnoldi	Amorphophallus titanum
5	Bunga Bangkai	Amorphophallus titanum
6	Daun Sang	Johannesteijsmannia altifrons

Sumber: (Kusmana & Hikmat, 2015)

2.2.2. Metode *Markerless-Based Tracking*

Dalam penelitian ini peneliti menggunakan metode tanpa menggunakan *marker*. Disini peneliti tidak membutuhkan *marker* untuk menampilkan objek tetapi *marker* yang dimanfaatkan adalah *marker* yang sudah terdaftar oleh *Vuforia*. (Buchari et al., 2015). Objek virtual diantisipasi dari bergantung pada sebagian sekitar sebagai tujuannya. (Mediaty Arief et al., 2019). Metode ini dapat digunakan untuk:

1. *Face tracking*

Cara ini memanfaatkan suatu perhitungan khusus dimana dari perhitungan ini kamera bisa melihat wajah manusia dengan melihat keadaan hidung, mata, dan mulut manusia, yang kemudian kamera melihat benda-benda di sekitarnya seperti lemari, kursi, pohon dan benda lain yang berbeda.

2. *3D object tracking*

Teknik *3D object tracking* adalah metode *markerless Augmented reality* yang bisa melihat berbagai macam objek yang ada di dunia nyata seperti meja, kendaraan, lemari, dan lain-lain.

3. *Motion tracking*

Dengan cara ini kamera bisa menangkap pergerakan dalam kenyataan. Teknik ini digunakan untuk pembuatan berbagai film yang menciptakan berbagai gerakan. Fitur untuk membedakan gerakan menyerupai membuat film yang menyimulasi Gerakan.

4. *Gps-based tracking*

Metode berikut menggunakan fitur GPS dan kompas yang terdapat pada ponsel. Dengan menggunakan fitur ini, aplikasi AR yang dibuat dapat memulihkan informasi dari GPS dan kompas yang kemudian menampilkan arah secara bertahap. Representasinya juga bisa dalam 3 dimensi (3D). Dengan menggunakan GPS dan sensor kompas yang kemudian menunjukkan suatu *object* kedalam bentuk arah yang dibutuhkan.

2.2.3. Metode Pengujian Sistem

Metode pengujian sistem adalah dimana aplikasi yang telah dibuat akan dilakukan pengujian kedalam beberapa metode seperti *whitebox* dan *blackbox*. Berikut beberapa penjelasan mengenai metode *whitebox* dan *blackbox*.

1. *Black Box Testing*

Merupakan teknik yang dengan tidak menguji rencana dan kode program untuk melihat apakah kapasitas, sumber informasi dan hasil produk yang tepat dengan penentuan yang diperlukan. Metode ini merupakan teknik yang tidak sulit dalam Penilaian ukuran data yang diuji dapat ditentukan dari jumlah bagian data entri yang nantinya akan dicoba, bagian yang harus diisi seperti batas atas dan

bawah bawah yang dapat diterima (Cholifah et al., 2018). Berikut ialah beberapa Teknik dari *Black-box Testing*

1. *Equivalence Partitioning*

Cara kerja metode ini ialah dengan memisahkan partisi atau membagi menjadi beberapa bagian dari menginput informasi.

2. *Boundary Value Analysis*

Metode *boundary*, dimana ada kesalahan dari perspektif eksternal atau dalam produk, dasar, atau nilai terbesar dari kesalahan yang diperoleh.

3. *Fuzzing*

Fuzzing adalah teknik untuk menemukan *bug* / impedansi dari perangkat lunak dengan memanfaatkan injeksi data yang tidak memadai maupun pertemuan semi-otomatis.

4. *Cause-Effect Graph*

Metode ini ialah percobaan yang memanfaatkan ilustrasi sebagai acuannya. Dimana pada bagan ini mendeskripsikan keterkaitan antara efek dan penyebab dari kesalahan tersebut

5. *Orthogonal Array Testing*

Bisa dimanfaatkan apabila ukuran ruang informasi cukup kecil, namun cukup besar untuk digunakan pada tingkat yang sangat besar.

7. *All Pair Testing*

Pada metode ini, semua rangkaian *test case* di rancang sedemikian rupa sehingga semua campuran diskrit yang mungkin, semua hal yang dipertimbangkan, dapat dieksekusi tergantung pada batasan informasinya.

Alasan dari pengujian ini ialah mempunyai pasangan *test case* yang sesuai dan bergabung dengan kumpulan ini.

8. *State Transition*

Tes ini bermanfaat untuk melakukan percobaan terhadap keadaan dari mesin dan navigasi dari *use interface* dalam struktur began.

2. *White Box*

Merupakan pengujian *software* dari segi desain dan kode program yang dapat memberikan informasi dan menghasilkan pekerjaan yang sesuai dengan detail prasyarat. Mengikuti program harus dimungkinkan dengan memperbaiki kode sumber program sehingga jika dicoba dalam penggunaan *White Box Testing* lagi, akan memberikan *Node, Edgeds dan Test Cases* yang tidak banyak ketimbang dengan tes dahulunya. (Mediaty, 2018)

2.2.4. UML (*Unified Modeling Language*)




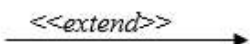

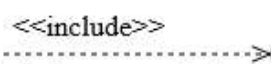
UML (Unified Modeling Language) standar yang dapat dipergunakan untuk mengarsipkan, menspesifikasi dan merancang *software*. UML adalah metode yang digunakan dalam mengembangkan sebuah system yang berorientasi objek serta yang membantu dalam pengembangan sistem. (Windu & Grace, 2013) UML tergantung pada desain atau gambar untuk menunjukkan, menunjukkan, merakit, dan mengarsipkan kerangka kerja perbaikan yang diatur pemrograman itemUML juga memberikan standar untuk menyusun kerangka kerja diagram, yang mencakup gagasan ukuran bisnis, pengiriman kelas dalam bahasa pemrograman,

penggambaran kumpulan data, dan segmen yang diperlukan untuk kerangka produk. (Amin Sofiyanto, 2020) UML terdiri dari beberapa diagram diantaranya:

1. Use Case Diagram

Diagram ini memberikan gambaran eksternal dari *framework* yang akan dibuat. Diagram ini terdiri dari *actor*, *usecase*, dan relasinya sebagai pengelompokan kegiatan yang memberikan kualitas terukur untuk *actor*.

Tabel 2.7 Simbol *Use Case Diagram*

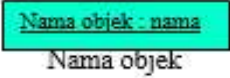
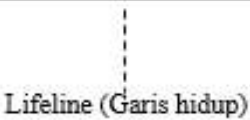


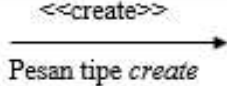
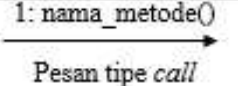
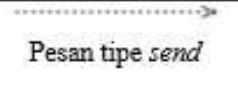
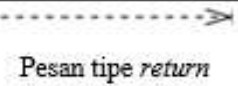
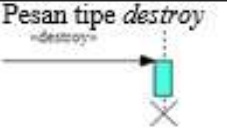
Simbol	Deskripsi
 <i>Usecase</i>	Fungsi atau kegunaan dari sistem untuk sistem pertukaran pesan antara unit dan aktor.
 Actor Nama Aktor	Aktor atau <i>user</i> merupakan yang berperan dan memiliki hubungan dengan sistem informasi yang dibuat.
 <i>Association</i> (asosiasi)	Asosiasi merupakan yang menghubungkan <i>actor</i> dengan <i>use case</i> atau lainnya.
 <i>Extend</i> (ekstensi)	Menghubungkan <i>Use Case</i> tambahan ke dalam sebuah <i>Use Case</i>
 <i>Generalization</i> (Generalisasi)	Merupakan hubungan umum – khusus (generalisasi dan spesialisasi) untuk dua use case.
 <i>Include</i> (menggunakan)	Mengekstraksi bagian-bagian <i>use case</i> yang digunakan beberapa <i>usecase</i>

Sumber: (Amin Sofiyanto, 2020)

2. Sequence Diagram

Menggambarkan perilaku objek dalam sebuah *Usecase* dengan menggambarkan objek yang saling bertukar pesan dari objek. *Sequence* dimanfaatkan untuk melihat rencana yang akan dibangun pada *Usecase*.

Tabel 2.8 Simbol *Sequence Diagram*






Simbol	Deskripsi
	Merupakan orang atau sistem lain atau proses diluar sistem yang dibuat, yang berhubungan dengan sistem yang dibuat.
	Merupakan garis hidup objek yang menerangkan kehidupan objek.
	Merupakan objek yang melakukan interaksi pesan
	Merupakan simbol yang menyatakan bahwa objek dalam keadaan berinteraksi dan keadaan aktif
	Merupakan pernyataan satu objek membuat objek lain.
	Merupakan pernyataan satu objek memanggil metode atau operasi pada objek lain atau diri sendiri.
	Merupakan pernyataan bahwa objek mengirimkan informasi atau masukan atau data ke objek lain.
	Merupakan pernyataan bahwa objek menjalankan metode atau operasi yang memberi hasil suatu pengembalian atau keluaran ke objek tertentu
	Merupakan pernyataan bahwa satu objek mengakhiri hidup dari objek lain, jika ada <i>create</i> lebih baik ada <i>destroy</i> .

Sumber: (Amin Sofiyanto, 2020)

3. Activity Diagram

Activity Diagram adalah aktifitas-aktifitas, objek, *state*, transisi *state* dan *event*. Semua yang ada dalam diagram alur kerja mendeskripsikan perilaku sistem untuk aktivitas. Diagram ini menggambarkan suatu gerakan atau kerangka kerja sebagai bermacam-macam kegiatan, bagaimana kegiatan tersebut mulai, pilihan yang mungkin dibuat dan sampai gerakan tersebut berakhir, atau juga bisa menggambarkan lebih dari satu interaksi pada saat bersamaan.

Tabel 2.9 Simbol *Activity Diagram*

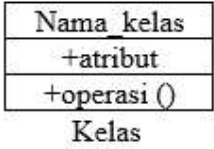
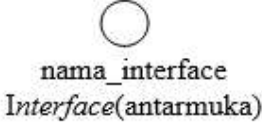
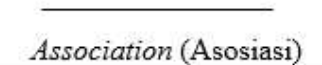
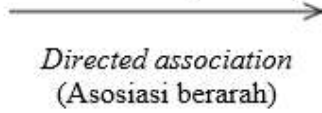
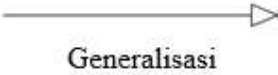
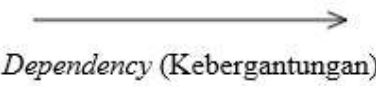
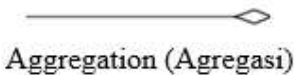
Simbol	Deskripsi
 Status awal/ Akhir	Merupakan status awal atau akhir keadaan dari sistem, setiap diagram aktivitas memiliki satu status awal.
 Aktivitas	Merupakan kegiatan yang dilakukan sistem, sering dimulai dengan kata kerja
 <i>Decision</i> (Percabangan)	Merupakan hubungan percabangan untuk keputusan aktivitas yang memiliki lebih dari satu pilihan.
 <i>Join</i> (Penggabungan)	Merupakan hubungan penggabungan jika satu atau lebih aktivitas menjadi satu. Swimlane Merupakan
 <i>Swimlane</i>	Merupakan yang memisahkan organisasi bisnis. Memiliki tanggung jawab untuk aktivitas yang terjadi.

Sumber: (Amin Sofiyanto, 2020)

4. Class Diagram

Class Diagram adalah diagram struktur untuk mendefinisikan kelas-kelas yang akan dibuat dalam membuat sebuah *system*.

Tabel 2.10 Simbol *Class Diagram*

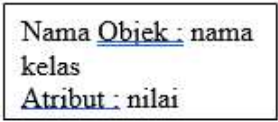
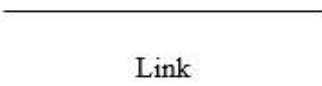
Simbol	Deskripsi
 <p>Kelas</p>	Merupakan kelas yang ada pada struktur sistem. Memiliki atribut dan operasi dalam kelas.
 <p>Interface(antarmuka)</p>	Merupakan kemiripan dengan kelas tetapi memiliki metode yang di deklarasikan tanpa isi dan tanpa atribut kelas.
 <p>Association (Asosiasi)</p>	Merupakan relasi antarkelas (umum), biasanya dilengkapi dengan <i>multiplicity</i> .
 <p>Directed association (Asosiasi berarah)</p>	Merupakan relasi antara kelas bermakna satu kelas digunakan oleh kelas yang lain, biasanya dilengkapi dengan <i>multiplicity</i> .
 <p>Generalisasi</p>	Merupakan relasi antara kelas bermakna generalisasi-spesialisasi (umum ke khusus).
 <p>Dependency (Kebergantungan)</p>	Merupakan relasi kebergantungan (<i>dependency</i>) antara kelas
 <p>Aggregation (Agregasi)</p>	Merupakan relasi antara kelas bermakna semua bagian (<i>whole-part</i>)

Sumber: (Amin Sofiyanto, 2020)

5. Object Diagram

Menggambarkan konstruksi kerangka sejauh penamaan artikel dan item dalam kerangka. Dalam grafik artikel, harus dipastikan bahwa semua kelas yang telah dicirikan dalam diagram kelas harus digunakan sebagai item, mengingat sesuatu yang lain, pencirian kelas tidak dapat ditingkatkan. Diagram objek juga berfungsi untuk menggambarkan gambaran nilai atau isi dari properti masing-masing kelas. Berikut adalah gambar dalam grafik artikel:

Tabel 2.11 Simbol *Object Diagram*

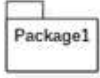
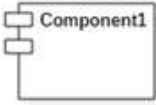



Simbol	Deskripsi
	Objek dari kelas yang berjalan saat sistem dijalankan
	Relasi antar objek

Sumber: (Amin Sofiyanto, 2020)

6. *Component Diagram*

Diagram komponen dirancang untuk menunjukkan hubungan dan ikatan antara bermacam-macam segmen dalam suatu kerangka kerja. Diagram komponen fokus di sekitar bagian kerangka kerja yang diperlukan dan ada dalam kerangka kerja.

Tabel 2.12 Simbol *Component Diagram*






Simbol	Deskripsi
 <p><i>Package</i></p>	Package merupakan bungkusan satu atau lebih elemen
 <p>Komponen <i>Dependency / kebergantungan</i></p>	Merupakan komponen sistem kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai
 <p>Nama_Interface <i>Interface/ antarmuka</i></p>	Sebagai antarmuka komponen agar tidak mengakses langsung komponen
 <p><i>Link</i></p>	Relasi antar komponen
 <p><i>File</i></p>	Dokumen dapat berupa <i>file, library</i>

Sumber: (Amin Sofiyanto, 2020)

7. Composite Structure Diagram

Diagram digunakan untuk mendeskripsikan desain bagian yang saling berhubungan atau untuk menggambarkan konstruksi pada saat *runtime* dari kejadian terkait. Dapat menggambarkan konstruksi di kelas kerjasama.

Tabel 2.13 Simbol *Composite Structure Diagram*


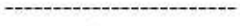
Simbol	Deskripsi
 <i>Association / Asosiasi</i>	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i>
 <i>Directed Association / Asosiasi berarah</i>	Relasi antar kelas dengan makna yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
 Generalisasi	Generalisasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
 <i>Dependency / Kebergantungan</i>	Relasi antar kelas dengan makna kebergantungan antar kelas
 <i>Aggregation / agregasi</i>	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

Sumber: (Amin Sofiyanto, 2020)

8. *Package diagram*

Package diagram memberikan metode pengumpulan komponen yang umumnya mendukung dalam diagram UML. Hampir seluruh diagram UML bisa digabungkan menggunakan *package diagram*. Berikut ini terdapat beberapa simbol dari diagram *package*.

Tabel 2.14 Simbol *Package Diagram*

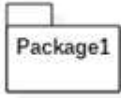
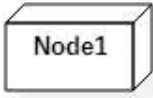


Simbol	Deskripsi
 <i>Package</i>	Sekelompok elemen-elemen model
 <i>Import</i>	Suatu <i>dependency</i> yang mengindikasikan isi tujuan paket secara umum yang ditambahkan kedalam sumber paket
 <i>Access</i>	Suatu <i>dependency</i> yang mengindikasikan isi tujuan paket secara umum yang bisa digunakan pada nama sumber paket

Sumber: (Amin Sofiyanto, 2020)

9. *Deployment Diagram*

Diagram ini menggambarkan pengaturan bagian dalam ukuran eksekusi aplikasi. Diagram ini juga bisa dipergunakan untuk menunjukkan bagian-bagian kerangka tambahan (*embedded system*) yang ditunjukan untuk merancang *device*, *node*, dan *hardware*.

Tabel 2.15 Simbol *Deployment Diagram*




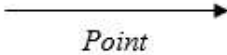
Simbol	Deskripsi
 <i>Package</i>	<i>Package</i> merupakan sebuah bungkusan dari satu node atau lebih
 <i>Node</i>	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak (<i>software</i>) yang tidak dibuat sendiri, jika didalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikuti sertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen
 <i>Dependency / Kebergantungan</i>	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
 <i>Link</i>	Relasi antar <i>node</i>

Sumber: (Amin Sofiyanto, 2020)

10. State Machine Diagram

Diagram ini disebut juga *statechart diagram* yang dimanfaatkan untuk menggambarkan perubahan keadaan atau kemajuan mesin atau kerangka kerja atau objek. Jika diagram sekuen digunakan untuk kolaborasi antar objek, oleh karena itu diagram status dimanfaatkan dalam hubungan sebuah objek.

Tabel 2.16 Simbol *State Mechine diagram*

Simbol	Deskripsi
	Digambarkan berbentuk segi empat dengan sudut membulat dan memiliki nama sesuai kondisi saat itu.
	Digunakan untuk menggambarkan awal dari kejadian dalam suatu diagram <i>flowchart</i> .
	Digunakan untuk menjelaskan/ menggambarkan akhir (<i>end</i>) dari kejadian dalam suatu diagram <i>state chart</i>
[<i>Guard</i>]	Merupakan syarat transisi yang bersangkutan.
	Digunakan untuk menggambarkan/ menjelaskan apakah akan masuk (<i>entry point</i>) ke dalam status atau keluar (<i>exit point</i>).
<i>Event</i>	Digunakan untuk menjelaskan kondisi yang menyebabkan sesuatu pada status.

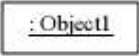

Sumber: (Amin Sofiyanto, 2020)

11. *Communication Diagram*

Diagram ini menggambarkan korespondensi antar objek/bagian dari perkembangan penyampaian pesan. Diagram komunikasi menghasilkan data yang didapat dari diagram kelas, diagram *sequence*, dan diagram *use case* untuk menggambarkan perpaduan antara konstruksi statis dan perilaku dinamis kerangka kerja. Diagram ini menguraikan banyak pesan pada diagram sekuen menjadi satu diagram. Pada diagram ini yang disusun adalah kegiatan / teknik yang dilakukan

antara satu objek dan objek yang lainnya secara umum, dengan cara ini dapat diambil dari komunikasi di semua diagram sekuen

Tabel 2.17 Simbol *Communication Diagram*

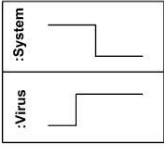
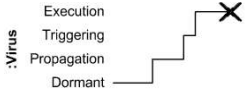
Simbol	Deskripsi
 Objek	Object merupakan <i>instance</i> dari sebuah <i>class</i> dan dituliskan tersusun secara horizontal.
 Aktor	<i>Actor</i> juga dapat berkomunikasi dengan <i>object</i> , maka <i>actor</i> juga dapat diurutkan sebagai kolom.
<i>Message</i>	<i>Message</i> digambarkan dengan anak panah yang mengarah antar objek dan diberi label urutan nomor yang mengindikasikan urutan komunikasi yang terjadi antar objek, seperti yang sudah digambarkan dan dijelaskan di atas.

Sumber: (Amin Sofiyanto, 2020)

12. *Timing Diagram*

Diagram ini adalah diagram yang menekankan pada penggambaran yang diidentifikasi dengan persyaratan waktu. Diagram ini dimanfaatkan untuk menggambarkan pelaksanaan kerangka kerja dalam jangka waktu tertentu. Dalam diagram ini bisa dimanfaatkan untuk menggambarkan aktivitas dari perangkat digital karena gambaran visual terlihat lebih jelas dari pada kata-kata.

Tabel 2.18 Simbol *Timing Diagram*

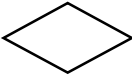




Simbol	Deskripsi
	<p><i>Lifeline</i> adalah elemen bernama yang mewakili peserta individu dalam interaksi.</p>
 <p style="text-align: center;">Notasi</p>	<p>Peristiwa penghancuran digambarkan dengan tanda silang berbentuk X di akhir garis waktu.</p>

Sumber: (Amin Sofiyanto, 2020)

13. *Interaction Overview Diagram*

Diagram ini hampir menyerupai diagram aktivitas yang digunakan untuk penggambaran beberapa urutan aktivitas. Diagram ini merupakan bentuk aktivitas diagram dimana setiap titiknya mendeskripsikan diagram interaksi.

Tabel 2.19 *Interaction Overview Diagram*

Simbol	Deskripsi
 <i>Decision</i>	<i>Decision</i> adalah suatu unsur diagram yang menunjukkan titik perkembangan bersyarat, jika kondisi benar, maka proses terus jika tidak, maka yang lain
	Mendefinisikan awal aliran Ketika sebuah kegiatan dipanggil
	Menunjukkan penyelesaian sebuah kegiatan, setelah mencapai final,
	Menunjukkan aktivitas yang sedang berlangsung
 <i>Flow Control</i>	Konektor yang menghubungkan dua <i>node</i> dalam sebuah diagram.

Sumber: (Amin Sofiyanto, 2020)

2.2.5. Software Pendukung

2.2.5.1. *Unity 3D*



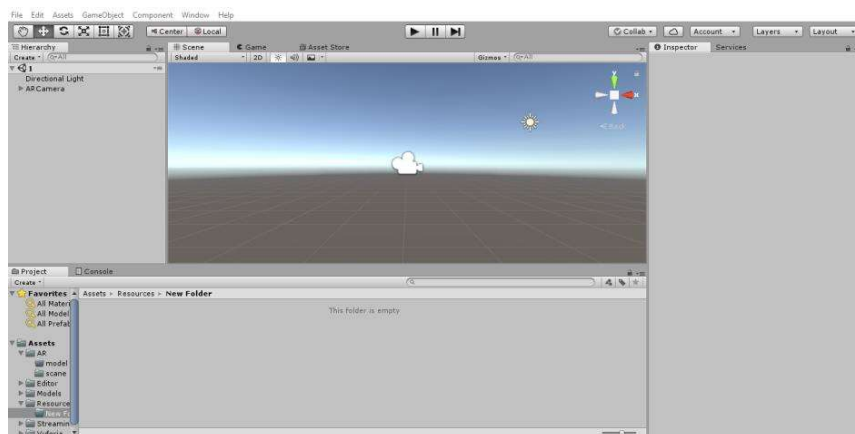
Gambar 2.15 Logo *Unity 3D*
(Sumber: Unity3D)

Unity 3D adalah sebuah sistem yang menyediakan layanan dalam pembuatan sebuah *games*. *Unity 3D* ialah perangkat yang telah terdaftar untuk

pembuatan rekaman *games* 3D atau konten interaktif lainnya seperti visualisasi arsitektur animasi 3D (Wiharto & Budihartanti, 2017). *Unity 3D* ialah sebuah *game engine* yang berbasis *cross-platform*. *Unity* dimanfaatkan dalam pembuatan sebuah *game* yang nantinya dapat dimainkan menggunakan *Computer*, ponsel, *iPhone*, PS3, dan bahkan X-BOX (Bagus & Mahendra, 2016)

Dari uraian diatas dapat disimpulkan *Unity* merupakan sebuah aplikasi yang sudah terdaftar untuk pembuatan sebuah *game*, desain arsitektur dan simulasi. *Unity* bisa dimanfaatkan dalam membuat *games computer* dan *games online*. tidak dibuat untuk mendesain karena *Unity* bukan merupakan *Tool* untuk perancangan.

Untuk *game* internet memerlukan sebuah *plugin*, seperti *Unity Web Player*, setara dengan *Flash Player* dalam programnya. Banyak hal yang dapat dikerjakan dalam *Unity*, ada *highlight* seperti zona *sound reverb*, efek *particle*, dan *Sky Box* untuk menampilkan langit. Fitur pengaturan awal yang diberikan oleh *Unity*, terdapat tiga bahasa pemrograman yaitu, *JavaScript*, *C#*, dan *Boo*.



Gambar 2.16 Tampilan *Unity 3D*
(Sumber: *Unity 3D*)

Dibawah ini merupakan beberapa penjelasan dari setiap *window*:

1. *Scene*: dimanfaatkan untuk memulai *game* yang akan dibuat.
2. *Project*: terdapat seluruh aset yang dimanfaatkan dalam pembuatan *game* yang tersimpan dalam *harddrive computer*.
3. *Hierarchy*: terdapat objek permainan atau beberapa objek *game* yang dipergunakan dalam *scene*. Urutan objek *game* dapat dipindahkan dan dikelompokkan menjadi *parent* dan *child*.
4. *Inspector*: terdapat penjelasan dari objek yang telah dipilih. Jendela ini juga terdapat informasi property atau komponen dari sebuah objek *game* maupun *asset*.
5. *Game*: komponen ini dimanfaatkan untuk menunjukkan hasil dari permainan saat di jalankan.
6. *Toolbar*: terdapat *button* yang dapat membantu dalam mengatur berbagai *segmen* pada *games*.

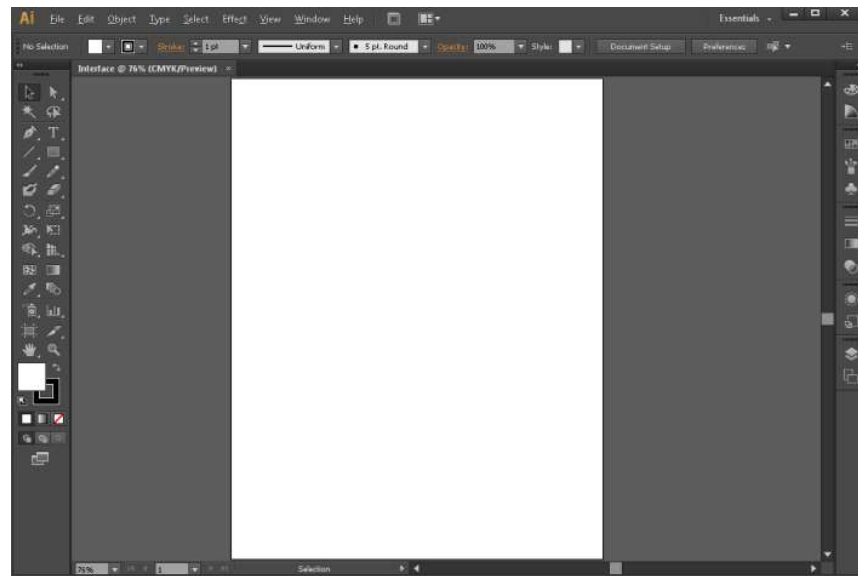
2.2.5.2. Adobe Illustrator



Gambar 2.17 Logo *Adobe Illustrator*
Sumber: *Adobe Illustrator*

(Wahana Komputer, 2011) mengatakan, *Adobe Illustrator* merupakan program desain yang dimanfaatkan untuk membuat karya desain yang sangat memikat dan imajinatif berbasis vektor. *Adobe Illustrator* memiliki tampilan yang

lebih modern yang membuatnya mudah untuk mengukur dan membuat berbagai jenis objek *vector*. *Adobe Illustrator* memiliki fitur baru yaitu fitur *Artboard* yang bisa membagi sebuah *file* kerja menjadi beberapa perspektif yang berbeda dan pada *software* ini juga terdapat banyak efek dan corak yang bisa digunakan untuk memberikan pemrosesan pada objek sehingga akan membentuk objek yang menarik.



Gambar 2.18 Tampilan *Adobe Illustrator*
Sumber: *Adobe Illustrator*

1. *Selection Tool* (v)

Digunakan untuk memilih objek, memindahkan objek dan menyalin objek.

2. *Direct-Select Tool* (A)

Digunakan untuk menyeleksi objek lebih mendasar. Dan *Tool* yang satu ini juga mmenggerakan salah satu sudut dari objek yang di pilih.

3. *Magic Wand Tool* (Y)

Digunakan untuk memilih objek berbentuk vektor yang memiliki sifat serupa seperti *colors, line, gradient*.

4. *Lasso Tool (Q)*

Digunakan untuk memilih objek dengan menandai objek tersebut.

5. *Pen Tool (P)*

Digunakan dalam pembuatan objek secara mendalam seperti yang ditunjukkan oleh apa yang dibutuhkan titik demi titik atau *anchor point* untuk membentuk sebuah kurva.

6. *Add anchor point Tool (+)*

Digunakan untuk menambahkan fokus pada objek yang membentuk vektor. Titik yang sebelumnya sudah terbentuk dapat di pindahkan menggunakan *Direct-Select Tool*.

7. *Type Tool (T)*

Digunakan untuk mengetik atau membuat teks.

8. *Area type Tool*

Digunakan untuk membuat teks pada ruang objek sebagai vektor, seperti membuat tulisan pada objek berbentuk kotak, lalu menekan garis kotak tersebut dan sudah dapat untuk mengetik di area kotak tersebut.

9. *Vertical type Tool*

Digunakan untuk membuat *text* dalam bentuk vertikal.

10. *Vertical area type Tool*

Digunakan untuk membuat teks dalam suatu objek berbagai vektor dan hasil teksnya berbentuk vertikal

11. *Vertical Type on a path Tool*

Digunakan dalam membuat teks vertikal pada objek garis yang berbentuk vektor sesuai dengan lengkungan garis tersebut.

12. *Line segmen Tool (l)*

Digunakan untuk membuat garis lurus ataupun garis miring.

13. *Arc Tool*

Digunakan untuk membuat garis lengkung, yang lengkungannya berbentuk setengah lingkaran.

14. *Spiral Tool*

Digunakan dalam membuat sebuah garis yang tikungannya seperti memutar.

15. *Rectangular Grid Tool*

Digunakan dalam membentuk sebuah garis kotak dan pada tengah kotak tersebut terdapat banyak kotak kecil.

16. *Polar grid Tool*

Digunakan dalam pembuatan garis yang membentuk 2 lingkaran.

17. *Rectangle Tool (M)*

Digunakan untuk membuat objek bentuk kotak yang sudutnya 90 derajat.

18. *Roundet Rectangle Tool*

Digunakan dalam membuat objek kotak yang sudutnya melengkung.

19. *Elips Tool (L)*

Digunakan dalam pembuatan objek lingkaran.

20. *Polygon Tool*

Digunakan untuk membuat persegi yang sudutnya sesuai dengan jumlah lekukan. *Tool* ini bisa menampilkan segitiga, segilima dll.

21. *Star Tool*

Digunakan untuk membuat bintang dan kaki bintang juga dapat ditambahkan untuk membuat lingkaran.

22. *Flare Tool*

Digunakan untuk menampilkan efek cahaya, meletakkan gambar efek cahaya di belakang layar.

23. *Pain brush Tool (B)*

Digunakan untuk menggambar sesuai keinginan.

2.2.5.3. *Vuforia SDK*



Gambar 2.19 *Vuforia SDK*
(sumber: (Grubert & Grasset, 2013).

Vuforia merupakan *Software Development Kit (SDK)* untuk *system smartphone* yang dimanfaatkan dalam membuat sebuah aplikasi *Augmented Reality*. Awalnya *Vuforia* dikenal dengan *QCAR (Qualcomm Company Augmentend Reality)*. Inovasi *Computer Vision* digunakan untuk mengetahui dan mendeteksi gambar dan objek 3D biasa, seperti kotak, secara *realtime*. Kemampuan penyetelan gambar membantu perancang memposisikan dan menempatkan *virtual* orientasi objek, misalnya model 3D dan media lain, dalam menemukan gambar yang dapat dipercaya ketika dilihat melalui kamera ponsel. Objek maya kemudian memperhatikan posisi dan orientasi dari gambar secara

real-time sehingga sudut pandang pengguna pada objek sesuai dengan sudut pandang mereka pada gambar target, dengan cara ini memberikan gagasan bahwa bagian virtual dari adegan realitas saat ini (Mario Fernandez, Qualcomm, 2012).

Vuforia SDK membutuhkan bagian-bagian penting untuk dapat bekerja dengan baik yaitu sebagai berikut:

1. Kamera, diharapkan dapat menjamin masing-masing *frame* tertangkap dan dibersihkan secara efektif ke pelacak. *Developer* mungkin memberi tahu kamera saat akan bekerja dan berhenti.
2. *Image Converter* mengubah desain *camera* menjadi desain yang telah diidentifikasi oleh *OpenGL* dan untuk mengikuti.
3. *Tracker*, perhitungan visi PC yang dapat mengenali dan mengikuti objek yang dapat disertifikasi pada *camcorder*. Mengingat gambar dari kamera, perhitungan alternatif untuk membedakan *trackbale* terbaru, dan tangkapan virtual. Hasil ini akan tersimpan pada *state* objek yang dipergunakan oleh pembuat video *foundation* dan dapat diperoleh dari kode aplikasi.
4. *Video Background Renderer*, membuat hasil dari *camera* tersimpan dalam *state object*. Tampilan dari *video background renderer* sangat ketergantungan dengan *device* yang digunakan.
5. *Application Code*, memperkenalkan semua segmen di atas dan melakukan tiga langkah signifikan dalam kode aplikasi, misalnya, *Query state object* menyatakan obejek pada tujuan marker lain yang dikenali, memperbarui

algoritma aplikasi setiap kali informasi baru dimasukkan dan *Render* grafis yang ditambahkan (*augmented*).

6. *Target Resources*, memanfaatkan *system* manajemen *on-line Target*. *Asets* yang diperlukan berisi desain *xml.Config.xml* yang memungkinkan *developer* untuk mengatur fitur-fitur dalam *trackbale* dan berkas *binary* yang berisi basis data yang dapat dideteksi.

2.2.5.4. *Android Studio*



Gambar 2.20 Logo *Android Studio*
(Sumber: *Android Studio*)

Android Studio adalah digunakan untuk mengembangkan aplikasi *Android*. *Android Studio* mempunyai beberapa keunggulan yang membuatnya populer dan banyak penggunanya. Dibawah ini terdapat keunggulan dari *Android*:

1. *Interface* yang mudah untuk digunakan
2. Terbuka untuk dikembangkan
3. Selalu dilakukan *upgrade*
4. Banyak aplikasi dan *game* gratis

Berikut kekurangan *Android Studio*:

1. Beberapa dari ponsel *ndroid* tidak mendapatkan pembaruan
2. Banyaknya *merk* dan *type* pada *android*

3. *Lag* dan Lemot

Berikut antarmuka *Android Studio*

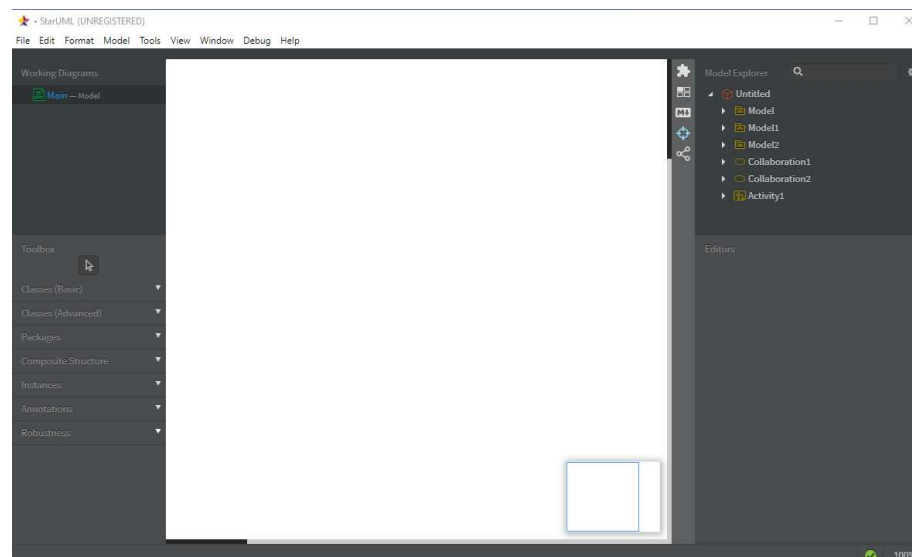
1. *Toolbar*, menunjukkan untuk melakukan atau menjalankan aktivitas dengan cepat. Di jendela ini Anda dapat menangani *SDK Manager*, *AVD Manager*, menjalankan hasil program, *build program*, hingga *debuging*. dibuat sedemikian rupa dengan simbol-simbol yang mudah dikenali sehingga memudahkan para engineer untuk melakukan aktivitas.
2. *Navigation Bar*, ampilan ini berfungsi untuk menyelidiki semua aktivitas. Disini dapat dilakukakn tindakan seperti membuat proyek baru, mengedit proyek atau membuka proyek yang telah dibuat. Sebagian kapasitas pada rute dapat diakses padapada *Toolbar* di nomor 1.
3. *Windows Editor*, dalam tampilan ini pengembang bisa menyesuaikan atau membuat kode. Semua kode dalam desain registri tugas dapat diubah di jendela ini.
4. *Windows Toolbar*, pada bagian ini, anda dapat memperkecil atau memperbesar jendela. Dapat juga untuk memilih jendela mana yang di tampilkan, dan jendela mana yang ditutupi.
5. *Windows Tool*, pada jendela ini dapat dilakukan penganturan manajemen proyek, pencarian dan *version control*. Dapat juga memperbesar atau memperkecil jendela dan bisa memilih jendela mana yang di tampilkan.
6. *Status Bar*, pada tampilan ini dapat terlihat status *project*, apakah dapat berjalan normal, ada kesalahan atau ada pemberitahuan.

2.2.5.5. Star UML



Gambar 2.21 Logo *StarUML*
(Sumber: *Star UML*)

Menurut (Amin Sofiyanto, 2020) *Star UML* adalah sebuah *software* yang dimanfaatkan untuk permodelan dalam mendukung UML. *StarUML* dilengkapi dengan 13 jenis diagram yang berbeda, mendukung notasi UML dengan dukungan sketsa UML. *Star UML* juga dapat mengoptimalkan produktivitas dan kualitas dari suatu proyek *software*.



Gambar 2.22 Tampilan *StarUML*
(Sumber: *StarUML*)

Berikut ini beberapa konsep dasar pada *StarUML*:

1. *Model, View* dan *Diagram*

Model komponen yang berisi data untuk model produk. *View* adalah artikulasi visual dari data dalam model dan diagram yang merupakan bermacam-macam komponen yang memberikan ide pengguna dalam rencana tertentu.

2. *Project* dan *Unit*

Merupakan unit administrasi dasar di dalam *StarUML*. Sebuah *project* dapat mengawasi salah satu model pemrograman. Proyek ialah bagian teratas *package* yang secara konsisten ada di beberapa model produk.

3. *File Project*

File project disimpan dengan XML dengan augmentasi “.UML”. Semua *model, views* dan *diagrams* yang dibuat dengan *StarUMLt* tersimpan dalam satu *file project*. *File project* terdapat berisi data berikut ini:

1. Profil UML
2. *Unit file*
3. Informasi untuk semua model yang ada.
4. Data untuk setiap *diagrams* dan *views*

4. *Units*

Terdapat beberapa situasi dimana sebuah proyek harus disimpan dalam beberapa dokumen kecil dengan tujuan agar para pengembang dapat menyelesaikan pekerjaan di dalam satu *project* secara bersamaan. Sebuah *unit* memiliki berbagai struktur berjenjang dan berisi beberapa bagian unit. *Unit* disimpan dengan format “.UML“ *file* dan beberapa mengacu pada dokumen

projek (.UML) atau dokumen *unit* lainnya (.UNT). Semua komponen di bawah tipe komponen *package* ini dapat tersimpan dalam setiap *file unit* (.UNT).

5. *Module*

Module ialah sebuah *package* yang memberikan kapasitas dan fitur tambahan dari *StarUML*. *Module* dibuat sebagai campuran dari komponen-komponen ekspansi dan selanjutnya membuat berbagai jenis komponen dari sebuah *module*.

2.3. Penelitian Terdahulu

1. Muntahanah, Rozali Toyib, Miko Ansyori, Penerapan Teknologi *Augmented reality* Pada Katalog Rumah Berbasis *Android*, Jurnal *Pseudocode*, Volume IV, Nomor 1, Februari 2017

Pada penelitian ini dapat dilihat permasalahan yang ada yakni, untuk dapat melihat luas rumah, pembeli terkadang tidak mempunyai waktu karena mereka memiliki kesibukan, terutama individu yang benar-benar ditempati, begitu juga dengan luas rumah yang jauh dari luas pelanggan. Dan dikarenakan rumah yang ditunjukkan dari indeks tersebut tidak terlihat pasti dan asli. Melalui media khusus ini, kebanyakan calon pembeli kebingungan untuk memprediksi gambaran atau keadaan rumah yang akan dibuat, dari teknologi tersebut sebenarnya memanfaatkan benda dua dimensi (2D), yang memiliki perspektif terbatas. Disini peneliti menggunakan metode *makerless*. Adanya sebuah aplikasi *Augmented reality* dapat menjadi media promosi perumahan yang dapat memudahkan vendor untuk mempresentasikan barang

serta memudahkan pembeli untuk memilih rumah yang ideal. Dalam penelitian ini peneliti membuat sebuah aplikasi *Augmented reality* yang berguna sebagai media promosi untuk perumahan yang dapat mempermudah vendor dalam mempresentasikan produk serta memudahkan pembeli untuk memilih rumah sesuai keinginan.

2. Lia Kamelia, Perkembangan Teknologi *Augmented reality* Sebagai Media Pembelajaran Interaktif Pada Mata Kuliah Kimia Dasar, November 2018, Volume IX, No. 1, Juni 2015

Dari latar belakang dapat dilihat permasalahan penelitian ini yaitu, sekarang ini mata kuliah kimia dasar ialah mata kuliah yang tidak menarik menurut para mahasiswa, khususnya mahasiswa fakultas teknik tingkat awal yang mendapatkan mata kuliah ini. Dari penjelasannya ialah kimia memiliki banyak gagasan yang sulit untuk dimengerti dan digambarkan, seperti struktur atom dan bentuk partikel dalam zat kimia. Sehingga membutuhkan sebuah media pembelajaran alternatif untuk menambah pemahaman gagasan kimia secara interaktif. Penelitian ini yang menggunakan teknologi AR diharapkan dapat dimanfaatkan oleh pengguna baik dosen maupun mahasiswa untuk lebih berinteraksi secara nyata dengan secara visual. Inovasi AR ini juga diciptakan untuk menggantikan kegiatan praktik kimia dasar yang selama ini dipandang menggunakan banyak uang dan waktu.

3. Tri Yuliono, Sarwanto, Peduk Rintayati, Keefektifan Media Pembelajaran *Augmented reality* Terhadap Penguasaan Konsep Sistem Pencernaan Manusia, Jurnal Pendidikan Dasar, Volume III, No.3, 2018

Dari penelitian permasalahan yang dapat dilihat yaitu, siswa terkadang tidak terlalu fokus atau membutuhkan pebaikan dalam pelajaran, menjawab pertanyaan dari guru dengan tegas tanpa premis, siswa cenderung mengantuk disebabkan dari mata pelajaran yang membosankan dan siswa tidak aktif pada saat mengikuti pelajaran. Metode penelitian yang dimanfaatkan oleh peneliti ialah *Quasi Eksperimen*. Dapat diambil kesimpulan bahwa terdapat keefektifan media *Augmented reality* terhadap gagasan kerangka yang dikuasai terkait pencernaan di tubuh manusia pada siswa kelas V SD di kabupaten Sragen. Media dari hasil penelitian tidak hanya menampilkan teks akan tetapi juga menampilkan unsur-unsur multimedia suara juga animasi yang membantu siswa dalam pemahaman materi.

4. Meyti Eka Apriyani¹, Robie Gustianto², *Augmented reality* Sebagai Alat Pengenalan Hewan Purbakala Dengan Animasi 3D Menggunakan Metode Single Marker, JURNAL INFOTEL-Informatika Telekomunikasi Elektronika, Volume VII, No.1, 2015

Dari penelitian ini permasalahan yang ada dapat dilihat yaitu, seperti sekarang ini dalam mempelajari mengenai hewan purbakala saat belajar sejarah di sekolah tingkat SMP dan SMA hanya menggunakan buku, dimana yang terdapat hanya gambar-gambar seperti fosil hewan purbakala yang telah ditemukan di beberapa wilayah di dunia tanpa mengetahui proses dimana cara hewan tersebut dapat berjalan, berlari, makan, atau saat mereka mati. Peneliti memanfaatkan metode *single marker*. Tampilan dari *Augmented reality* berupa alat untuk memperagakan pengenalan hewan purbakala sudah menarik.

5. **Aldi Apriansyah, Dhimas Mulia Anugraha, Galuh Prakoso, Kevin Nuardy Erdiham, Rudi Priyana, Aplikasi Pengenalan Hewan dengan Teknologi Marker Less *Augmented reality* Berbasis *Android*, Journal of Computer and Information Technology Vol.1, No. 1, August 2017**

Dari masalah proses pembelajaran interaktif mengenai pengenalan hewan yang dinilai kurang efektif. Peneliti menggunakan metode *markerless*. Hasil yang dapat disimpulkan ialah eksekusi dan pengujian aplikasi dengan memanfaatkan metode *black box*. Hasil yang diperoleh ialah waktu reaksi untuk mengenali *marker* sekitar 1 detik. Jarak dasar untuk mendeteksi marker sekitar 30 cm dan jarak paling jauh sekitar 100 cm. Mempresentasikan virtual *button* pada gambar target yang dilengkapi dengan suara yang diperoleh dari model 3D.

6. **Devina Mirza Nunditya, Maman Somantri, Yuli Christyono, Aplikasi Naturar Pengenalan Tanaman Berbasis *Augmented reality* Pada Perangkat Bergerak *Android*, Jurnal Teknik Informatika, Volume Ii, No. 1, 2017**

Dari penelitian ada beberapa permasalahan yaitu, dengan jumlah yang banyak pada tumbuhan di Indonesia, tidak semua orang menenal nama dan kelebihan dari berbagai jenis tanaman. Terutama tanaman di lingkungan sekitar tempat tinggal. Peneliti memanfaatkan metode *marker*. Hasil dari pengujian ialah semua komponen pengujian efektif tanpa adanya kesalahan. Nilai dari responden menghasilkan rata-rata kepuasan $89,3=89$ dengan ukuran yang dapat diterima.

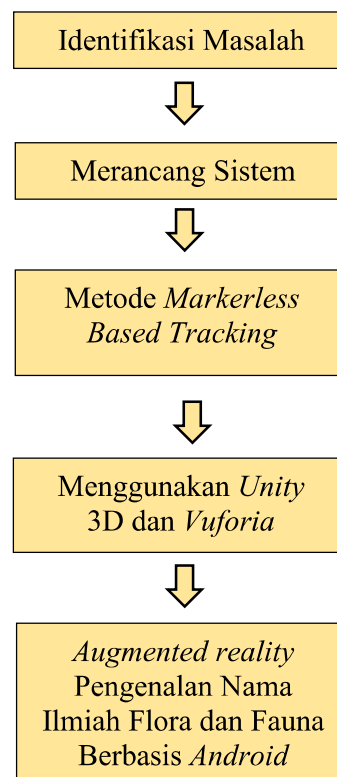
7. Ali Fakhrudin, Sri Yamtinah, dan Riyadi, Implementasi Teknologi Augmented Reality pembelajaran Ilmu Alam Di Sekolah Dasar mengoptimalkan Hasil Belajar Siswa, IJJET, Vol. 3, No. 1, Januari 2019

Dalam melakukan pemeriksaan, nilai mata pelajaran IPA berada pada peringkat paling rendah dibandingkan dengan mata pelajaran lain. Di ujian Nasional tahun pelajaran 2013/104 rata-rata bahasa Indonesia Bahasa adalah 8,26, matematika 7,38, sedangkan IPA 7,29. Kemudian, di Ujian Nasional tahun ajaran 2014/2015, nilai IPA mengalami penyusutan yang cukup relevan dengan rata-rata 62,43. Mulai dari itu, ialah sebuah tantangan untuk para pengajar agar dapat menyampaikan pelajaran IPA semaksimal mungkin kepada siswa. Upaya yang dapat diperbuat untuk meningkatkan hasil belajar ialah dengan memperbaiki metode pembelajaran. Hasil dari belajar siswa dipengaruhi oleh faktor dalam dan faktor luar.

Peneliti memanfaatkan model prosedural yang menyesuaikan dengan perbaikan model *Borg and Gall*. Peneliti merancang sebuah aplikasi untuk pembelajaran IPA dengan memanfaatkan teknologi *augmented reality*. Hal yang membuat produk dapat diterima dan relevan untuk diterap kandalam pembelajaran ialah karena produk ini telah disetujui oleh 9 ahli. Selain itu, hasil pengujian lapangan juga menunjukkan hasil yang signifikan. Hasil pengujian yang dilakukan di lapangan menunjukkan bahwa pruduk yang dibuat telah dapat memaksimalkan hasil belajar IPA siswa.

2.4. Kerangka Pikir

Kerangka Pemikiran Merupakan diagram yang mendeskripsikan secara garis besar jalannya sebuah penelitian. Kerangka pemikiran juga sering digunakan oleh mahasiswa dalam melakukan penelitian tesis. Berikut kerangka pemikiran penelitian yang dibuat oleh peneliti:



Gambar 2.23 Kerangka Pemikiran
Sumber: Data Olahan Peneliti (2021)

Dari tahapan kerangka berpikir diatas dapat disimpulkan bahwa dalam pembuatan *augmented reality* yang berbasis *android*, terdapat 5 tahapan yaitu:

1. Identifikasi masalah

Peneliti mendefinisikan masalah yang ada bersumber dari latar belakang masalah yang timbul pada perihal yang akan di kaji untuk mencari pemecahan masalah tersebut.

2. Merancang Sistem

Peneliti mendesain rancangan aplikasi yang nantinya digunakan sebagai media atau alat yang nantinya akan digunakan dalam penelitian ini, Peneliti merancang *Augmented reality* ini berdasarkan berbagai sumber yang diambil sebagai acuan yang berhubungan dengan penelitian ini.

3. Metode *Markerless Based Tracking*

Metode yang menangani pengembangan aplikasi media pembelajaran yang dibuat dalam bentuk AR atau yang dinyatakan dalam bentuk tiga dimensi tanpa menggunakan marker.

4. Menggunakan *Unity 3D* dan *Vuforia*

Dalam membuat sebuah *Augmented reality* harus mempunyai perangkat lunak yang dapat mendukung pembuatan sebuah pembuatan *augmented reality*. Oleh karena itu, penulis menggunakan *software Unity* untuk membuat dan menggunakan *Android Studio*.

5. *Augmented reality* pengenalan nama ilmiah flora fauna

Pembuatan sistem telah menghasilkan aplikasi yang dapat digunakan oleh siswa sebagai bahan pembelajaran guna mengetahui tentang nama – nama ilmiah dari flora dan fauna.