BAB II
TINJAUAN PUSTAKA

2.1 Profil TK Karya Cendikia

TK Karya Cendikia merupakan sekolah taman kanak-kanak yang beralamat di Jl. Trs. Bojongkoneng Atas Ciharalang No. 88 Rt. 04 / 02 Kab. Bandung dengan SK Pendirian: 421.1/2.659-Disdikbud/2015 pada 28 September 2015.

2.2 Landasan Teori

Landasan teori bertujuan memberikan gambaran dari teori yang terkait dalam pengembangan aplikasi.

2.2.1 Sampah

Sampah adalah bahan atau material yang dihasilkan sebagai hasil dari aktivitas manusia dan tidak lagi memiliki nilai atau kegunaan yang langsung. Sampah dapat berupa bahan padat, cair, atau gas. Berikut adalah beberapa jenis sampah yang umum ditemui:

1. Sampah Organik: Merupakan jenis sampah yang berasal dari bahan-bahan organik, seperti sisa makanan, dedaunan, kertas, dan sebagainya. Sampah organik dapat diuraikan secara alami oleh mikroorganisme dan menjadi pupuk.
2. Sampah Anorganik: Merupakan jenis sampah yang tidak mudah terurai secara alami. Contohnya adalah plastik, kaca, logam, karet, dan sebagainya. Sampah anorganik seringkali membutuhkan pengolahan lebih lanjut untuk daur ulang atau pembuangan yang aman.
3. Sampah Berbahaya: Merupakan jenis sampah yang mengandung bahan-bahan beracun, berbahaya, atau berpotensi merusak lingkungan dan kesehatan manusia. Contohnya adalah baterai, bahan kimia berbahaya, limbah medis, limbah elektronik (e-waste), dan sebagainya. Sampah berbahaya memerlukan penanganan khusus agar tidak mencemari lingkungan.
4. Sampah Elektronik (E-Waste): Merupakan jenis sampah yang berasal dari peralatan elektronik yang sudah tidak digunakan lagi atau rusak. Contohnya adalah komputer, telepon seluler, televisi, printer, dan sebagainya. E-waste

sering mengandung bahan-bahan berbahaya dan harus didaur ulang dengan benar.

5. Sampah Medis: Merupakan jenis sampah yang dihasilkan oleh rumah sakit, klinik, atau fasilitas kesehatan lainnya. Sampah medis dapat berupa jarum suntik, perban, obat-obatan kadaluarsa, bahan kimia medis, dan limbah biologi. Sampah medis harus diolah dan dibuang dengan prosedur yang ketat untuk mencegah penyebaran penyakit.
6. Sampah Padat Lainnya: Selain jenis sampah di atas, masih ada banyak jenis sampah lainnya, seperti kertas bekas, kain, kayu, karet, dan sebagainya.

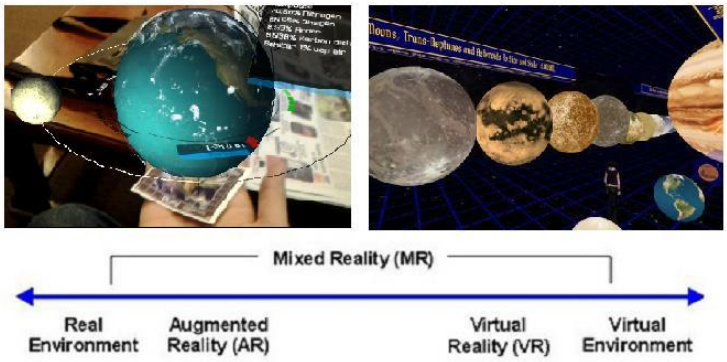
Pengelolaan sampah yang baik sangat penting untuk menjaga kebersihan lingkungan dan mencegah dampak negatif terhadap kesehatan manusia dan ekosistem. Pengurangan, daur ulang, dan pembuangan yang aman merupakan prinsip penting dalam pengelolaan sampah yang berkelanjutan.

2.2.2 *Augmented Reality*

Augmented reality adalah Teknologi yang membawa dunia digital ke dunia nyata. Augmented reality memungkinkan pengguna untuk melihat dua dimensi atau Objek virtual 3D yang diproyeksikan secara realistis dunia. (Teknologi berasal dari augmented reality. antarmuka dan desain). Anda dapat memasukkan teknologi AR ini Menampilkan informasi spesifik di dunia virtual Dunia nyata menggunakan perangkat seperti webcam Komputer, ponsel Android, kacamata khusus. pengguna atau Pengguna nyata tidak dapat melihat objek virtual Identifikasi objek secara visual yang membutuhkan perantara dalam bentuk komputer dan kamera untuk digunakan nanti Membawa objek virtual ke dunia nyata. Tujuan augmented reality untuk mendasarinya di dunia nyata Tambahkan teknologi virtual dan data kontekstual ke Pemahaman yang lebih jelas tentang manusia sebagai pengguna. ini Data kontekstual dapat berupa deskripsi audio. Data lokasi, latar belakang sejarah, dll [5].

Augmented Reality (AR) berbeda dengan *Virtual Reality* (VR), di mana VR menambahkan objek nyata didalam dunia maya, sedangkan konsep AR menambahkan objek maya ke dalam dunia nyata [4]. *Virtual Reality* membutuhkan alat-alat yang mahal untuk saat ini, seperti Oculus Rift berharga USD 1.000 atau Rp 15 jutaan. Berbeda dengan AR yang relatif lebih murah dan hanya membutuhkan Android untuk menjalankannya. *Software* yang digunakan untuk

pembuatannya juga gratis. Objek dalam AR pun bisa dibuat interaktif dengan menambahkan *script* tertentu. Walaupun tidak seinteraktif VR, namun cukup untuk meningkatkan *experience* pengguna [6].



Gambar 2. 1 Perbedaan AR dan VR (diadaptasi dari [7])

Dalam kehidupan sehari-hari, *Augmented Reality* juga sudah mulai mendominasi pasar dunia. Salah satu dari kegunaan AR adalah sebagai sarana dalam berbisnis. Banyak perusahaan yang mulai menggunakan AR sebagai media promosi produk mereka. Sebagai contoh perusahaan mobil yang ingin menunjukan produk mobil mereka akan menggunakan AR yang berbentuk mobil yang didesain mirip dengan produk mereka untuk diperlihatkan kepada para calon pembeli. Hal tersebut merupakan sebuah efisiensi yang sangat unik karena dengan AR, para sales pun tidak harus membawa produk yang besar, hanya perlu membawa sebuah *smartphone* dan sebuah brosur untuk di-*scan* di depan calon pembeli [8].

Tujuan utama dari AR adalah untuk menciptakan lingkungan baru dengan menggabungkan interaktivitas lingkungan nyata dan virtual sehingga pengguna merasa bahwa lingkungan yang diciptakan adalah nyata. Dengan kata lain, pengguna merasa tidak ada perbedaan yang dirasakan antara AR dengan apa yang mereka lihat/rasakan di lingkungan nyata. Dengan bantuan teknologi AR (seperti visi komputasi dan pengenalan pola) lingkungan nyata di sekitar kita akan dapat berinteraksi dalam bentuk digital (virtual). Informasi tentang objek dan lingkungan di sekitar kita dapat ditambahkan ke dalam sistem AR yang kemudian informasi tersebut ditampilkan di atas layer dunia nyata secara *real-time* seolah-olah informasi tersebut adalah nyata. Informasi

yang ditampilkan oleh subjek virtual membantu pengguna melaksanakan kegiatan-kegiatan dalam dunia nyata [9].

2.2.3 Pemanfaatan *Augmented Reality*

Augmented Reality dapat meningkatkan persepsi dan interaksi pengguna dengan dunia nyata melalui objek virtual yang dapat menampilkan informasi yang tidak dapat dideteksi secara langsung oleh indra penggunanya. Informasi yang disampaikan oleh objek virtual membantu pengguna melakukan tugas dunia nyata. AR adalah contoh spesifik dari apa yang Fred Brooks sebut sebagai *Intelligence Amplification* (IA), yaitu menggunakan komputer sebagai alat untuk membuat tugas lebih mudah dilakukan oleh manusia. AR telah banyak diaplikasikan dalam berbagai bidang, seperti medis, militer, manufaktur, hiburan, visualisasi, dan robotika. Selain itu, kini pengaplikasiannya menjadi lebih luas, mencakup pendidikan, pemasaran, geospasial, navigasi dan perencanaan jalur, pariwisata, perencanaan kota dan teknik sipil [10].

Kemungkinan baru untuk belajar mengajar dilakukan oleh AR semakin dikenal oleh para peneliti di bidang pendidikan. Koeksistensi objek virtual dan lingkungan nyata memungkinkan peserta didik untuk memvisualisasikan hubungan spasial yang kompleks dan konsep abstrak, mengenai fenomena yang tidak mungkin terjadi di dunia nyata, berinteraksi dengan objek sintetik dua dan tiga dimensi dalam realitas campuran, dan mengembangkan praktik-praktik penting yang tidak dapat dikembangkan dan diterapkan di lingkungan pembelajaran. Manfaat di bidang pendidikan ini telah menjadikan AR sebagai salah satu teknologi utama yang muncul untuk pendidikan selama beberapa tahun ke depan [9].

2.2.4 *Marker*

Marker merupakan sebuah penanda khusus yang memiliki pola tertentu sehingga saat kamera mendeteksi marker, objek 3 dimensi dapat ditampilkan. *Augmented reality* saat ini melakukan perkembangan besar-besaran, salah satunya pada bagian *marker*. *Marker* pertama adalah *marker based tracking*. *Marker based tracking* ini sudah lama dikembangkan sejak 1980-an dan pada awal 1990-an mulai dikembangkan untuk penggunaan *augmented reality*. Kemudian *markerless*, perkembangan terbaru marker ini merupakan salah satu

metode *augmented reality* tanpa menggunakan *frame marker* sebagai objek yang dideteksi. Dengan adanya *markerless augmented reality*, maka pengguna *marker* sebagai *tracking object* yang selama ini menghabiskan ruang akan digantikan dengan gambar atau permukaan apapun yang berisi dengan tulisan, logo, atau gambar sebagai *tracking object* (objek yang dilacak) agar dapat langsung melibatkan objek yang dilacak tersebut sehingga dapat terlihat hidup dan interaktif, juga tidak lagi mengurangi efisiensi ruang [8].

2.2.5 Vuforia Engine

Vuforia adalah sebuah *Software Development Kit* (SDK) yang disediakan oleh Qualcomm untuk membantu para *developer* dalam mengembangkan aplikasi-aplikasi *Augmented Reality* (AR) pada mobile phone (iOS dan Android). Vuforia berperan sebagai *library* yang digunakan sebagai pendukung adanya *Augmented Reality* pada Android. Vuforia menganalisa gambar dengan menggunakan pendeteksi *marker* serta menghasilkan informasi 3D dari marker yang sudah dideteksi via API. *Programmer* juga dapat menggunakannya untuk membangun objek 3D virtual pada kamera. Vuforia mempunyai banyak fitur dan kemampuan yang dapat membantu developer untuk mewujudkan pemikiran mereka tanpa adanya batas secara teknikal. Dengan *support* untuk iOS, Android, dan Unity3D, platform Vuforia mendukung para *developer* untuk mengembangkan aplikasi yang dapat digunakan di hampir seluruh jenis *smartphone* dan tablet. Vuforia telah sukses dipakai di beberapa aplikasi-aplikasi *mobile* salah satunya yaitu James May‟s Science Stories yang dikembangkan oleh International Celebrity Networks [13], *game mobile* lainnya yang dihasilkan dengan program ini adalah Temple Run yang bisa dijalankan di Android dan iPhone [6].

Vuforia juga memungkinkan *developer* untuk merancang dan membuat aplikasi yang mempunyai berbagai kemampuan antara lain teknologi *computer vision* tingkat tinggi yang mengizinkan developer untuk membuat efek khusus pada *mobile service*, terus menerus mengenali *multiple image*, *tracking* dan *detection* tingkat lanjut, serta pengaturan *database* gambar yang fleksibel [12].

2.2.6 Model Target dalam Vuforia Engine

Model Target adalah salah satu fitur kunci dalam Vuforia yang memungkinkan pengguna untuk menggunakan model 3D fisik sebagai target untuk interaksi AR. Dengan Model Target, pengguna dapat mengenali dan berinteraksi dengan objek fisik dalam lingkungan mereka [13].

Model Target dalam Vuforia bekerja dengan mengenali karakteristik unik dalam model 3D fisik yang digunakan sebagai target. Ini termasuk poin-poin kunci atau tekstur yang diidentifikasi dan digunakan untuk melacak posisi dan orientasi target dalam lingkungan fisik [18].

2.2.7 *Global Positioning System (GPS)*

GPS adalah sistem navigasi global yang mengandalkan jaringan satelit yang mengorbit bumi. Sistem ini dirancang untuk menentukan posisi, waktu, dan navigasi dengan tingkat akurasi yang tinggi di berbagai lokasi di bumi [16]. Konsep dasar utama yang digunakan dalam GPS adalah trilaterasi, di mana jarak antara penerima GPS dan beberapa satelit yang terlihat digunakan untuk menghitung posisi. Penerima GPS menerima sinyal dari setidaknya empat satelit untuk menentukan posisi tiga dimensi [17].

GPS melibatkan beberapa komponen kunci, termasuk:

- Satelit GPS: Sekelompok satelit yang mengorbit bumi dan mengirimkan sinyal radio yang digunakan oleh penerima GPS.
- Penerima GPS: Perangkat yang digunakan untuk menerima sinyal GPS dan menghitung posisi berdasarkan data yang diterima.
- Sinyal GPS: Sinyal radio yang dikirimkan oleh satelit untuk menyebarkan informasi posisi dan waktu [17].

Dari beberapa penjelasan di atas, GPS dapat bekerja dengan pengukuran jarak dan Trilaterasi. Berikut adalah penjelasan lebih lengkapnya.

a. Pengukuran Jarak

GPS mengukur jarak dengan mengukur perbedaan waktu antara waktu yang dihasilkan oleh satelit dan waktu yang diterima oleh penerima GPS. Waktu yang dibutuhkan sinyal radio untuk

perjalanan dari satelit ke penerima digunakan untuk menghitung jarak relatif [19].

b. Trilaterasi

Trilaterasi adalah metode matematika yang digunakan oleh GPS untuk menentukan posisi berdasarkan pengukuran jarak dari beberapa satelit. Dengan membandingkan waktu tempuh sinyal dari setidaknya empat satelit, GPS dapat menghitung posisi dengan akurasi tinggi [20].

2.2.8 *Augmented Reality GPS (AR GPS)*

"AR GPS" mengacu pada penggabungan antara teknologi Augmented Reality (AR) dengan sistem navigasi Global Positioning System (GPS). AR GPS berarti menggabungkan informasi navigasi dari sistem GPS dengan elemen visual augmented reality, memungkinkan pengguna untuk melihat informasi navigasi atau data terkait lokasi di atas tampilan dunia nyata yang dilihat melalui perangkat. Contohnya, Anda dapat menggunakan aplikasi AR GPS pada *smartphone* untuk menunjukkan arah di atas pandangan dunia nyata yang dilihat melalui kamera ponsel Anda, memberikan panduan visual saat berjalan atau mengemudi.

Integrasi GPS dalam AR memungkinkan penentuan posisi dan orientasi perangkat dalam dunia fisik. Ini memungkinkan objek AR untuk disematkan ke lokasi geografis yang tepat dalam realitas nyata [21].

Cara kerja AR GPS adalah sebagai berikut:

- Pelacakan Lokasi

  GPS digunakan untuk melacak lokasi perangkat AR dan pengguna dalam dunia nyata. Informasi lokasi ini digunakan untuk menentukan tempat objek AR harus ditampilkan.

- Peningkatan Pengalaman Pengguna

  Integrasi GPS dalam AR dapat meningkatkan pengalaman pengguna dengan memberikan konteks geografis yang relevan dalam pengalaman AR. Ini dapat digunakan dalam aplikasi seperti navigasi, wisata, dan pendidikan.

2.2.9 Bahasa Pemrograman Visual C Sharp

Visual C-Sharp atau C# merupakan salah satu bahasa pemrograman berorientasi objek yang dikeluarkan Microsoft. Proyek pembuatan ditangani oleh Anders Helsberg dan diperkenalkan untuk pertama kali pada bulan Juli 2000. Visual C# merupakan bahasa pemrograman modern berorientasi Objek yang menjadi bahasa pemrograman utama dalam platform Microsoft.NET Framework. Visual C# dianggap sebagai kombinasi antara efisiensi pemrograman C++, kesederhanaan pemrograman java, dan penyederhanaan dari pemrograman Visual Basic. Saat ini, visual C# dapat ditemukan dalam paket Microsoft Visual Studio [12].

Standar *European Computer Manufacturers Association* (ECMA) mencantumkan beberapa tujuan desain bahasa pemrograman C#, sebagai berikut:

a) Bahasa pemrograman C# dibuat sebagai bahasa pemrograman yang bersifat bahasa pemrograman *general-purpose* (untuk tujuan jamak), berorientasi objek, modern, dan sederhana.
b) Bahasa pemrograman C# ditujukan untuk digunakan dalam mengembangkan komponen perangkat lunak yang mampu mengambil keuntungan dari lingkungan terdistribusi.
c) Portabilitas *programmer* sangatlah penting, khususnya bagi *programmer* yang telah lama menggunakan bahasa pemrograman C dan C++.
d) Dukungan untuk internasionalisasi (*multi-language*) juga sangat penting.
e) C# ditujukan agar cocok digunakan untuk menulis program aplikasi baik dalam sistem klien-server (*hosted system*) maupun sistem *embedded* (*embedded system*), mulai dari perangkat lunak yang sangat besar yang menggunakan sistem operasi yang canggih hingga kepada perangkat lunak yang sangat kecil yang memiliki fungsi-fungsi terdedikasi.
f) Meskipun aplikasi C# ditujukan agar bersifat 'ekonomis' dalam hal kebutuhan pemrosesan dan memori komputer, bahasa C# tidak ditujukan untuk bersaing secara langsung dengan kinerja dan ukuran perangkat lunak yang dibuat dengan menggunakan bahasa pemrograman C dan bahasa rakitan.

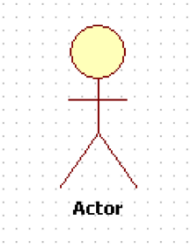
g) Bahasa C# harus mencakup pengecekan jenis (*type checking*) yang kuat, pengecekan larik (*array*), pendeteksian terhadap percobaan terhadap penggunaan Variabel-variabel yang belum diinisialisasikan, portabilitas kode sumber, dan pengumpulan sampah (*garbage collection*) secara otomatis

2.2.10 *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi yang dibuat. Use case diagram menggambarkan interaksi antara satu atau lebih aktor terhadap sebuah sistem. Proses deteksi *markerless augmented reality* ini melalui beberapa tahapan. Secara garis besarnya, dalam perancangan ada tiga bagian utama yaitu *Object Tracking*, *Image Processing* dan *Rendering camera preview*. *Use Case* diagram terdiri dari beberapa elemen pemodelan utama, yaitu *Actor, Use Case, Association, Dependency,* dan *Generalization.*

1. *Actor*

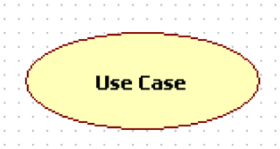
   Pada dasarnya *actor* bukanlah bagian dari *use case* diagram, namun untuk dapat terciptanya suatu *use case* diagram diperlukan beberapa *actor*. *Actor* tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. *Actor* hanya berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atas *use case*. *Actor* digambarkan dengan stick man. Pada gambar berikut menunjukan Simbol *Actor* pada *Use Case* Diagram:



2. *Use Case*

   Use case adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan

sistem yang akan dibangun. Pada gambar berikut merupakan Simbol *Use Case pada Use Case Diagram:*



3. *Association*

Associaton menghubungkan *link* antar *element,* dan bukan menggambarkan aliran data / informasi pada sistem. *Association* digunakan untuk menggambarkan bagaimana aktor terlibat dalam *usecase.* Ada 4 jenis relasi yang bisa timbul pada *usecase* diagram, yaitu *Association* antara aktor dan *usecase, Association* antara *usecase, Generalization/Inheritance* antara *usecase, Generalization/Inheritance* antara aktors.

4. *Dependency*

Dependency adalah sebuah *element* bergantung dalam beberapa cara ke *element* lainnya.

5. *Generalization*

Generalization disebut juga *inheritance* (pewarisan), sebuah *element* dapat merupakan spesialisasi dari *element* lainnya.

2.2.11 Class Diagram

Class diagram adalah diagram yang menunjukan *class-class* yang ada dari sebuah sistem dan hubungannya secara logika dan menggambarkan struktur statis dari sebuah sistem. *Class* diagram digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/perangkat lunak yang sedang digunakan.

Class diagram memberikan gambaran (diagram statis) tentang sistem/perangkat lunak dan relas-relasi yang ada didalamnya. Sebuah *class* digambarkan seperti sebuah bujur sangkar dengan tiga bagian ruangan yaitu bagian atas adalah bagian nama dari *class*, bagian tengah mendefinisikan *property*/*atribut class*, bagian akhir mendefinisikan *method-method* dari sebuah *class*. *Class* sebaiknya diberi

nama menggunakan kata benda sesuai dengan domain atau bagian atau kelompoknya.

2.2.12 Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang disusun dalam suatu urutan waktu. Secara khusus, diagram ini berasosiasi dengan *use case*. *Sequence* diagram menggambarkan kelakuan objek pada internal sebuah sistem dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek [6].

2.2.13 Unity 3D

Unity merupakan *game engine* yang dikembangkan oleh Unity Technologies. *Software* ini pertama kali diluncurkan pada tahun 2005 dan menjadi salah satu dari sekian banyak *game engine* yang dipakai oleh banyak pengembang *game professional* di dunia. Unity merupakan alat bantu pengembangan game dengan kemampuan rendering yang terintegritasi di dalamnya. Dengan menggunakan kecanggihan fitur — fiturnya dan dengan kecepatan kerja yang tinggi, unity dapat menciptakan sebuah program interaktif tidak hanya dalam 2 dimensi, tetapi juga dalam bentuk 3 dimensi [14]. Unity juga dibuat khusus untuk mendukung pengembang menggunakan *plugin* dari *software* pihak ketiga. Unity juga menyediakan toko aset (*Asset Store*) sendiri yang menyediakan berbagai *plugin* yang diperlukan untuk pengembang *game* dari pengembang, oleh pengembang dan untuk pengembang [15].

Plugin adalah unit ekstensi biner untuk aplikasi yang mana arsitekturnya memungkinkan fungsionalitas untuk diperkenalkan kepada *end-users* setelah instalasi aplikasi. *Plugin* adalah entitas perangkat lunak yang berhubungan erat dengan komponen. *Component-based development* (pengembangan berbasis komponen) biasanya tidak mempertimbangkan bahwa komponen- komponen tersebut dapat ditambahkan ke dalam aplikasi setelah proses instalasi aplikasi. Komponen biasanya digunakan untuk memfasilitasi pembangunan aplikasi itu sendiri [15].

Unity tidak hanya didesain untuk membuat *game* di *personal computer* (PC) saja, tetapi juga untuk platform yang berbeda seperti android, iOS webplayer, Mac, Linux Standalone, Xbox 360, PS3 dan juga Wii. Oleh karena itu, Unity sering disebut sebagai *game engine* yang multiplatform karena bisa digunakan untuk membuat *game* di berbagai macam platform. Unity memiliki dua versi yang berbeda. Yakni versi gratis dan versi pro yang berbayar. Walaupun berbeda jika membandingkan keduanya, versi gratis ini juga memiliki fitur — fitur yang bagus untuk membuat *game* dengan kualitas tinggi [14].

Adapun fitur-fitur yang dimiliki oleh Unity 3D antara lain sebagai berikut [13]:

1. *Integrated development environment* (IDE) atau lingkungan pengembangan terpadu.
2. Penyebaran hasil aplikasi pada banyak platform.
3. Engine grafis menggunakan Direct3D (Windows), OpenGL (Mac, Windows), OpenGL ES (iOS), dan proprietary API (Wii).
4. Game Scripting melalui Mono. Scripting yang dibangun pada Mono, implementasi open source dari NET Framework. Selain itu pemrograman dapat menggunakan UnityScript (bahasa kustom dengan sintaks JavaScriptinspired), bahasa C# atau Boo (yang memiliki sintaks Python-inspired).

2.2.12 Blender 3D

Blender adalah 3D *creation suite* gratis dan *open source*. Ini mendukung keseluruhan dari 3D *pipeline* seperti *modeling*, *rigging*, *animation*, *simulation*, *rendering*, *compositing* dan *motion tracking*, bahkan *editing video* dan pembuatan *game*. Pengguna mahir menggunakan API Blender untuk pembuatan skrip Python untuk menyesuaikan aplikasi dan menulis alat khusus. Blender sangat cocok untuk individu dan studio kecil yang mendapatkan manfaat dari saluran terpadu dan proses pengembangan yang responsif [14].

Blender adalah *cross-platform* dan berjalan dengan baik di komputer Linux, Windows, dan Macintosh. Antarmukanya menggunakan OpenGL untuk memberikan pengalaman yang konsisten. Untuk mengonfirmasi kompatibilitas

tertentu, daftar platform yang didukung menunjukkan yang secara rutin diuji oleh tim pengembangan [14].

2.2.13 Android Studio

Android Studio adalah *Integrated Development Enviroment* (IDE) untuk sistem operasi Android, yang dibangun di atas perangkat lunak JetBrains IntelliJ IDEA dan didesain khusus untuk pengembangan Android. IDE ini merupakan pengganti dari Eclipse Android Development Tools (ADT) yang sebelumnya merupakan IDE utama untuk pengembangan aplikasi android. Android studio sendiri pertama kali diumumkan di Google I/O conference pada tanggal 16 Mei 2013. Ini merupakan tahap preview dari versi 0.1 pada Mei 2013, dan memasuki tahap beta sejak versi 0.8 dan mulai diliris pada Juni 2014 [14].