BAB II
LANDASAN TEORI

Bab ini akan membahas mengenai teori yang melandasi penelitian, meliputi *Augmented Reality*, Android, Unity, Vuforia, Blender, dan Alat Musik Tradisional. Dimana dasar teori-teori tersebut digunakan dalam pengerjaan tugas akhir ini.

2.1 *Augmented Reality*

Augmented Reality (AR) adalah bagian dari *Environment Reality* (ER) atau yang dikenal juga sebagai *Virtual Reality* (VR)[3]. *Augmented Reality* (AR) memiliki kemampuan untuk melakukan komunikasi visual seperti tulisan bentuk atau gambar dari komputer *virtual* ke dalam dunia nyata. Antarmuka AR memungkinkan pengguna melihat dunia nyata bersamaan dengan citra virtual yang terletak pada suatu lokasi tempat dan objek nyata. Antarmuka AR meningkatkan pengalaman dunia nyata, tidak seperti antarmuka VR yang menarik pengguna dari dunia nyata dan masuk ke layar visual [14]. Pada garis besarnya perbedaan *Virtual Reality* dan *Augmented Reality* terletak pada bagian interaksi terhadap pengguna.

Augmented Reality dikembangkan oleh banyak peneliti yang mempunyai sebuah harapan yang besar terhadap sebuah perkembangan dalam bidang *Virtual Reality*. Dimulai pada tahun 1962, ketika Morton Heilig yaitu seorang *cinematographer* yang membuat mesin *Augmented Reality* pertama. Heilig merancang sebuah desain simulasi sepeda motor yang ia beri nama Sensorama, dimana Sensorama memiliki sebuah sistem yang pertama kali memperkenalkan gabungan teknologi visual, suara, serta getaran pada sebuah perangkat sistem sekaligus.

Tujuan utama dari pembagunan teknologi *Augmented Reality* sendiri yaitu untuk memberikan sebuah pengertian dan informasi dalam dunia nyata, dimana *Augmented Reality* mengambil dasar yang terdapat pada dunia nyata, lalu sistem tersebut akan menambahkan data kontekstual agar lebih memperjelas pemahaman seseorang terhadap informasi yang akan diserap.

Pada saat ini teknologi *Augmented Reality* telah diterapkan dan diaplikasikan oleh beberapa bidang dalam membantu keperluanya, beberapa bidang tersebut diantaranya yaitu:

1. Bidang Kesehatan

   Bidang ini merupakan bidang yang sangat memerlukan penggunaan tekhnologi AR. AR akan digunakan pada saat melakukan pemeriksaan sebelum operasi, membantu ahli bedah memiliki gambaran mengenai kepastian gejala yang diderita pasien [3].

2. Bidang Manufaktur dan Reparasi

   Pada bidang ini, *Augmented Reality* digunakan untuk memperbaiki mesin-mesin kendaraan yang berstruktur sangat kompleks, dengan adanya AR, mekanik akan dengan mudah melakukan pembongkaran dan pemasangan mesin kendaraan dengan *real*itas tertambah, yaitu dengan menampilkan gambar-gambar tiga dimensi [3].

3. Bidang Pemasaran

   Mempromosikan produk menggunakan teknologi *Augmented Reality*.
   Contohnya pada acara 2008 *Auto Show* Nissan pernah meluncurkan AR kendaraanya, konsep yang disajikan berupa brosur yang ketika dihadapkan terhadap *webcam* menunjukkan versi gambar 3D [23].

4. Bidang Pelatihan Militer

   Kalangan militer telah bertahun-tahun menggunakan tampilan dalam kokpit yang menampilkan informasi pada kaca pelindung pilot kokpit atau kaca depan helm penerbangan mereka, memberikan informasi seperti petunjuk, peta, musuh, lokasi, dan sel-sel api menggunakan AR. Ini merupakan sebuah bentuk tampilan *real*itas tertambah [23].

Pada *Augmented Reality* ada tiga karakteristik yang menjadi dasar atas sistem tersebut, diantaranya adalah kombinasi pada dunia nyata dan virtual, interaksi yang berjalan secara *real-time*, dan karakteristik yang terakhir adalah bentuk objek yang berupa model 3 dimensi atau 3D. Bentuk data kontekstual dalam sistem *Augmented Reality* ini dapat berupa data lokasi, *audio*, video ataupun dalam

bentuk data pemodelan 3D. Pada *Augmented Reality* terdapat beberapa komponen penting yang wajib dimiliki oleh pembuat aplikasi tersebut, diantranya:

1. *Display*s

   Dalam *Augmented Reality* terdapat 3 jenis *displays* yang digunakan, yaitu:

   a. Head Mounted *Display*s (HMD)

      Biasanya perangkat ini dipasang dikepala dan menjadi bagian pada sebuah helm ataupun perangkat yang menempel dikepala lainnya. HMD bekerja dengan menampilkan 2 buah *displays* optik baik itu *monocular* maupun *binocular*. Kelemahan dari HMD adalah pengguna tidak dapat merasakan atau menyentuh gambar visual secara *real*.

   b. Handheld

      Seperti namanya, perangkat ini merupakan *displays* yang harus ditahan (dipegang) oleh tangan penggunanya, *displays* ini memiliki perangkat komputer didalamnya, dimana pengguna akan dihadapkan pada teknik melihat video secara langsung, dengan menggabungkan visual kedalam dunia nyata dengan bantuan sensor.

   c. *Spatial Augmented Reality* (SAR)

      Displays yang dimana terdapat didalamnya video *projector*, elemen optik, hologram, radio frekuensi *tags*, dan teknik *tracking* lainnya untuk menampilkan informasi grafis secara langsung kedalam objek fisik tanpa harus dibawa atau dipakai oleh pengguna.

2. *Input Devices*

   Banyak sekali inputan yang dapat digunakan dalam *Augmented Reality*, baik itu *utilize gloves*, *wireless wristband*, maupun kamera yang dapat mendukung kinerja dari aplikasi itu sendiri. Disini kamera merupakan hal terpenting jika aplikasi *Augmented Reality* yang kita bangun menggunakan *marker*, karena kamera akan difungsikan sebagai *recording* sensor, dimana kamera akan memproses gambar yang masuk, lalu gambar tersebut akan dicocokan dengan *marker* yang sebelumnya telah dimasukan kedalam

aplikasi, dan jika *marker* tersebut terdeteksi (cocok), maka aplikasi akan menampilkan obyek 3D diatas *marker* tersebut.

3. *Tracking*

   Perangkat *tracking* terdiri atas kamera digital dan atau *optical sensor*, GPS, Accelometers, *solide state compase*, *wireless sensors*, dan perangkat lainnya. Semakin baik tingkat keakuratan sensor, maka semakin baik tingkatan pemindaian (*scaning*).

   Marker atau penanda merupakan gambar (*image*) yang difungsikan untuk proses *tracking* pada saat aplikasi dijalankan. Sehingga aplikasi akan mengenali posisi orientasi dari *marker* dan akan menciptakan objek virtual secara 3D, yaitu pada titik (0,0,0) dan sumbu (X,Y,Z).

4. Komputer

   Pada Pembuatan *Augmented Reality* peran komputer menjadi hal yang penting untuk dimiliki, dikarenakan komputer mengendalikan semua proses yang akan terjadi dalam aplikasi yang dibangun, selanjutnya output aplikasi akan ditampilkan melalui *displays*.

   Pada *Augmented Reality* sendiri sebetulnya komputer yang digunakan harus memiliki RAM yang memadai, hal ini dikarenakan untuk mempercepat proses *scaning* gambar pada sistem optik. Sejauh ini pembangunan teknologi *Augmented Reality* sendiri baru memakai komputer, tetapi dengan perkembangan *smartphone* yang semakin pesat diharapkan mampu menggantikan peran dari komputer, dengan penambahan pencahayaan serta dengan tambahan kecanggihan *graphic card* yang ditanam pada sebuah *smartphone*.

Penerapan *Augmented Reality* dibagi menjadi dua tipe yaitu *marker-based* dan *markerless* [11]. Kedua tipe *Augmented Reality* tersebut membutuhkan perangkat lunak agar berfungsi. *Marker-based* biasanya lebih populer dan lebih mudah dikerjakan, sedangkan *markerless* masih terus berkembang karena terkendala akurasi sensor dan keterbatasan *bandwidth* [6].

1. *Marker-based*

   Menggunakan beberapa kode QR pada gambar 2D untuk memproduksi sebuah citra sebagai hasil data yang dikenali oleh sebuah sensor *reader*, misalnya kamera pada *smartphone* Android [25].

2. *Markerless*

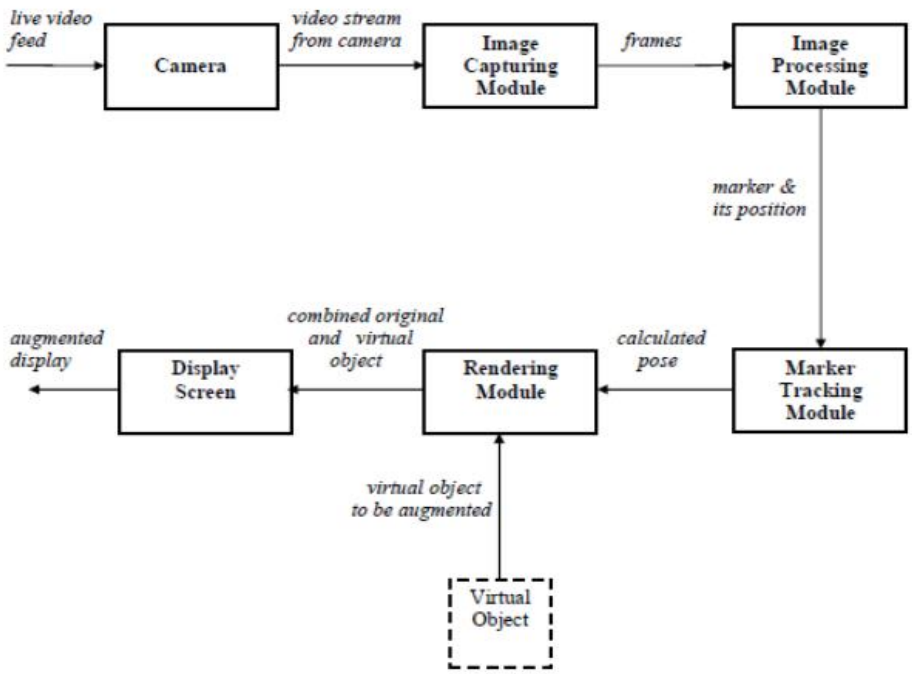
   Mengandalkan pada kemampuan peralatan yang akan digunakan sebagai *reader* misalnya lokasi GPS, pengukur kecepatan, dan lain-lain. Aplikasinya biasanya mengacu pada *Augmented Reality* berbasis lokasi atau posisi (*location-based* atau *position-based*) [6].

2.1.1 Metode *Marker*-Based *Augmented Reality*

Metode *marker-based* dari teknologi *Augmented Reality* dihadirkan dari gabungan teknologi *computer vision* dan *image processing* yang mencari informasi dari sebuah gambar secara langsung. Dimana *marker* yang didesain secara khusus, dibutuhkan untuk memunculkan objek 3D yang akan dihadirkan setelah proses *tracking* dan *positioning* dilakukan.

Marker yang akan digunakan harus terlebih dahulu dibuat, dengan cara mendaftarkan gambar kedalam sebuah *database*. Dalam hal ini penulis menggunakan Vuforia sebagai pembuat *database marker*. Gambar yang akan dijadikan *marker* harus memiliki sudut, hal ini dikarenakan vuforia melakukan *tracking* dengan proses *edge detection*.

Marker Based biasanya merupakan suatu ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang yang berwarna putih. Warna lainya juga dapat digunakan dengan syarat kontras warnanya dapat dibedakan dan dikenali oleh kamera. Bentuk pada *marker based* ini biasanya berupa seperti tato. Keuntungan menggunakan warna hitam dan putih adalah agar dapat memisahkan latar belakang *marker* dan bingkai dengan mudah. Area tengah bingkai pada sebuah *marker* biasanya menandai identitas suatu *marker based*.



Gambar 2.1 Marker based Architecture Block Diagram

Terdapat 6 modul proses dari *marker based system yaitu*[1] :

1. *Camera*

   Video langsung dunia nyata yang menjadi input dari kamera laptop ke modul Kamera. Menampilkan gambaran video langsung dunia nyata dari kamera pada laptop menjadi suatu proses awal dalam *augmented reality*. Video dunia nyata ini kemudian dijadikan input pada *Image Capturing* Modul.

2. *Image Capturing Module*

   Modul ini menganalisis inputan yang diberikan dari kamera, dengan menganalisa setiap frame dalam video. Modul ini kemudian menghasilkan gambar biner yaitu suatu gambar digital yang hanya memiliki dua nilai saja untuk setiap piksel. Biasanya dua warna yang digunakan untuk gambar biner adalah hitam dan putih. Gambar biner hitam dan putih ini disediakan sebagai inputan untuk *Image* Processing *Module*.

3. *Image Processing Module*

   Input ke *image processing module* adalah gambar biner yang merupakan output dari dari *Image Capturing Module* sebelumnya. Gambar biner ini kemudian diproses dengan menggunakan teknik pemrosesan gambar untuk mendeteksi AR *Marker*. Pendeteksian AR *Marker* sangat penting untuk menentukan posisi, dimana untuk menempatkan objek 3D virtual. Setelah AR *Marker* terdeteksi, lokasinya kemudian disediakan dan dijadikan sebagai inputan ke *Marker Tracking Module.*

4. *Marker Tracking Module*

   Marker Tracking Module merupakan "jantung" dari sistem *augmented reality*; Disini akan dihitung pose relatif kamera secara *real time*. Istilah pose disini berarti posisi enam derajat kebebasan (DOF), yaitu lokasi 3D dan orientasi 3D suatu objek. Pose yang dihitung dan diberikan sebagai masukan untuk modul *rendering*.

5. *Rendering Module*

   Dimana terdapat 2 inputan yang masuk ke *Rendering module*. Yang pertama yaitu dari pose yang dihitung pada *Marker* Tracking *Module* sebelumnya, dan yang kedua dari 3D Objek virtual untuk menjadi gambar *augmented*. Pada *rendering module* ini akan mengkombinasikan gambar asli dan komponen virtual menggunakan penghitungan pose dan kemudian menampilkan gambar *augmented* dari kedua inputan tersebut ke *display screen*.

6. *Display Screen*

   Display Screen merupakan media dimana gambar dari *augmented reality* divisualisasikan atau ditampilkan. Media ini berupa layar pada monitor atau *smartphone*.

2.2 Intensitas Cahaya

Intensitas cahaya memiliki peran sangat penting dalam proses pembacaan *marker* (*tracking).* Hal ini dikarenakan keberadaan cahaya dapat membantu kamera dalam menentukan *marker* yang dibaca, apakah *marker* tersebut cocok ataupun tidak. Jika pembacaan *marker* dilakukan diluar ruangan (ruang terbuka) hampir tidak memiliki kendala pada proses *tracking.* Berbeda halnya jika proses *tracking*

dilakukan didalam ruangan, dimana banyak faktor yang dapat mempengaruhi kecepatan maupun keberhasilan kamera dalam membaca *marker*.

Kondisi cahaya yang ada didalam ruangan sangat berbeda dengan intensitas cahaya yang berada diluar ruangan, baik dari sisi kecerahan, pengaruh bayangan objek lain, maupun titik jatuh cahaya, merupakan faktor-faktor yang dapat menghambat kamera untuk mengidentifikasi sisi dari sebuah *marker.* Pada Vuforia kecerahan menjadi faktor utama yang harus diperhatikan dalam proses *tracking,* hal ini dikarenakan intensitas cahaya didalam ruangan dapat mengganggu sistem saat melakukan proses *thresholding image* .

2.3 Android

Android merupakan sebuah sistem operasi berbasis java yang berjalan pada kernel *linux* 2.6 dan memiliki sistem yang sangat ringan dan berfitur lengkap [10]. Android dapat mencakup sistem operasi, *middleware* maupun aplikasi yang digunakan dalam perangkat mobile berbasis android [8]. Dikarenakan android merupakan *platform* terbuka bagi para *developer* dalam menciptakan aplikasi, membuat aplikasi android berkembang sangat cepat [17]. Terutama pada aplikasi *Augmented Reality,* yang belakangan ini sedang begitu ramai diperbincangkan.

Android dimulai 2003 oleh 4 orang pakar IT, Andi Rubin, Rich Minner, Nick Sears dan Chris White mendirikan Android, tujuan pembuatan aplikasi Android adalah mewujudkan *mobile device* yang lebih mengerti penggunanya, dikarenakan perkembangan Android sangat pesat, kemudian Google mengakuisisi Android pada agustus 2005, OS pada Android bersifat *open source,* dengan konsep *open source* pada OS Android, Android dapat dikembangkan oleh para pengembang aplikasi, sehingga tidak membutuhkan waktu lama bagi Android untuk bersaing mengalahkan sistem operasi *smartphone* lainya seperti *Symbian, Windows Mobile, Blackberry* dan iOS [13]. Kini sistem aplikasi Android menjadi terkenal karena sistem operasinya yang *open source* dan kegunaanya yang mudah digunakan sehingga membuat Android menjadi penguasa *operating system* bagi *smartphone* [17]. Terdapat beberapa arsitektur yang membantu pada pembentukan sistem Android diantaranya:

1. Aplikasi dan *Widget* aplikasi

   Aplikasi dan *widget* keduanya saling berhubungan. Dimana, *widget* digunakan sebagai penghubung pada saat akan melakuan penginstalan aplikasi. Dengan adanya *widget* aplikasi, *user* dapat melakukan *download* dan penginstalan aplikasi [8].

2. Aplikasi *framework*

   Framework aplikasi, merupakan sebuah *frame* yang digunakan oleh para pengembang dalam perancangan dan pembuatan aplikasi, yang dapat dijalankan pada sistem operasi Android. Fitur ini mendukung penggantian komponen dan penggunaan kembali komponen yang sudah dibuat [8].

3. *Library*

   Library digunakan untuk membantu para pengembang aplikasi dalam pembuatan maupun pengembangan aplikasi, *library* merupakan fitur-fitur yang tersedia pada Android, *library* yang sering digunakan para pengembang diantranya, C++/C, Libc dan SSL [8]. Fitur-fitur yang tersedia pada *platform* Android adalah sebagaimana diuraikan berikut : [13]

   a. *Framework Aplikasi*

      Dengan adanya *framework* aplikasi pembuat aplikasi dapat melakukan penggantian komponen dan penggunaan kembali komponen yang sudah dibuat (*reusable*).

   b. Grafis

      Dengan adanya fitur grafis, *developer* dapat membuat aplikasi grafis 2D maupun 3D. Dikarenakan Android memiliki *library Open* GL ES 1,0. Namun, pada saat ini sangat jarang aplikasi yang menggunakan grafis 2D.

   c. Dukungan Perangkat Tambahan

      Pada sistem aplikasi Android, *developer* dapat memanfaatkan kamera, layar sentuh, *accelerometer, magnetometers,* GPS, akselerasi 2D, dan akselerasi 3D dalam pembuatan aplikasi.

d. *Market*

Market merupakan tempat penjualan aplikasi pada Android, pada *market* terdapat *widget* aplikasi yang dibuat oleh beberapa *developer,* seperti halnya katalog aplikasi.

2.3.1 Fitur-Fitur Android

Fitur-fitur yang dapat digunakan oleh pengembang aplikasi dalam pembuatan aplikasi Android sebagai berikut: [24]

- *Database* dengan *SQLite*.
- Menulis dan membaca file di SD *card*.
- *Integrated browser* berbasis *web kit engine.*
- 2D *graphics library*; dan *Open* GL ES 2.0 untuk 3D *graphics*.
- Lokasi dan sensor: GPS, *compass, gyroscope*, *barometer*, *light* sensor, *accelerometer*.
- *Renderscript* untuk pemrosesan paralel memanfaatkan CPU dan GPU.
- Kamera, Audio *capture*, *Jetplayer*, media support untuk MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF.
- GSM *Telephony*: fasilitas telepon dan SMS.
- *Connectivity*: USB, *Bluetooth*, NFC, SIP (*Session Initiation Protocol*, internet *telephony* khusunya untuk video *conference* dan *instant* messaging) dan WiFi.
- *Multitouch*.

2.4 Unity 3D

Unity merupakan sebuah *game engine* yang terus berkembang dan sering *developer* gunakan untuk membuat *game,* namun tidak hanya *game* yang dapat dibuat menggunakan Unity, Unity dapat digunakan untuk mengolah objek 3 dimensi, suara dan tekstur, dengan adanya fitur untuk mengolah objek 3 dimensi, sehingga Unity dapat digunakan untuk pembuatan aplikasi *Augmented Reality* dengan bantuan Vuforia untuk melakukan *scan marker* [18]. Tidak hanya itu Unity mendukung beberapa *platform*, diantaranya Unity Web ,Windows, Mac, Android, Ios, Xbox, Playstation, dan Wii [7].

Dalam pembuatan aplikasi *Augmented Reality* menggunakan *library* Unity AR. Unity menyediakan sebuah *interface* ke AR *Toolkit* [16]. kamera yang berada pada *smartphone* digunakan sebagai sumber masukan untuk pengenalan pola *marker*. *Interface* ini untuk mendeteksi beberapa *source* dari aplikasi yang berjalan, melalui Unity AR pengguna akan diberikan informasi tentang posisi dan rotasi suatu pola *marker* yang telah terdaftar sebelumnya, dengan adanya gabungan antara AR *Toolkit* dan Unity ini dapat mempermudah pengguna untuk mengembangkan aplikasi berbasis *Augmented Reality* [11].

Untuk membuat sebuah aplikasi pada Unity terdapat beberapa *tools*, diantaranya yaitu : [17]

1. *Project*

   Merupakan kumpulan komponen dari aplikasi yang akan kita bangun, kumpulan tersebut dikumpulkan agar dapat di-*build* menjadi sebuah aplikasi. *Project* bisa berisi nama *project*, *platform building*. *Scene* aplikasi, *asset,* beberapa *pacekage* bisa digunakan dalam *project* untuk menyimpan gambar 2D, gambar 3D, *script*.

2. *Scene*

   Dapat disebut juga sebagai halaman untuk membuat layar aplikasi, *scene* dapat dianalogikan sebagai halaman yang berisi *object, text*, maupun lainya, contohnya halaman menu disimpan pada *scene* 1, *scene* 2 digunakan untuk menampilkan AR.

3. *Game Objects*

   Komponen yang ada di dalam *assets* yang dipindah ke dalam *scene*, yang dapat digerakkan, diatur ukurannya dan diatur rotasinya.

4. *Asset* dan *Package*

   Merupakan sekumpulan *object* yang disimpan, di dalamnya dapat berupa *game* objek, *terrain*, dan lainya. Dengan adanya *asset* tidak perlu membuat objek lagi jika ingin melanjutkan pembuatan aplikasi, tinggal melakukan *import.*

5. *Components*

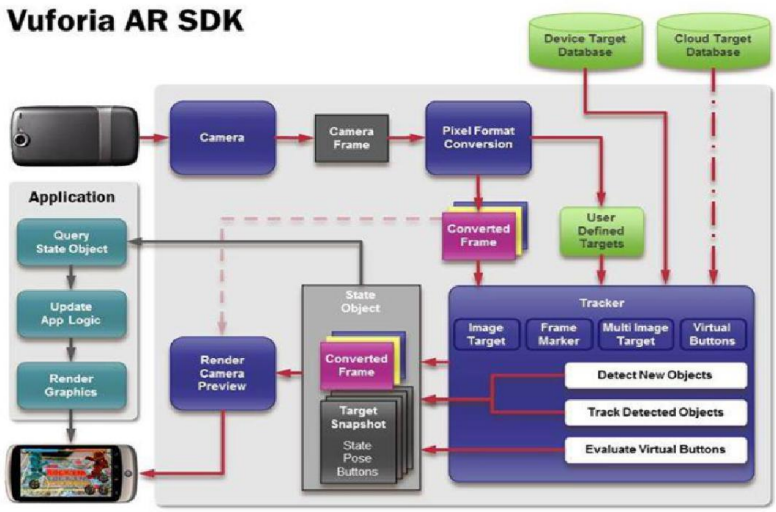
   Merupakan reaksi baru, bagi objek seperti *collision*, memunculkan partikel, dan sebagainya.

6. *Prefabs*

   Tempat untuk menyimpan satu jenis *game* objek, sehingga mudah untuk diperbanyak.

2.5 Vuforia SDK

Vuforia merupakan *Software Devlopment Kit* [19] yang sedang dikembangkan oleh Qualcomm SDK ini sangat membantu pengembang dalam menciptakan aplikasi atau *game* terutama aplikasi yang memiliki teknologi *Augmented Reality.* sehingga aplikasi yang dibuat dengan teknologi ini akan terlihat lebih interaktif dan nyata. SDK ini sendiri memiliki beberapa fitur untuk memindai seperti memindai objek, memindai teks, mengenali bingkai penanda, tombol *virtual*, mengidentifikasi permukaan objek, memindai dengan berbasis awan, mengenali target gambar, mengenali target benda silinder, dan mengenali objek target yang telah ditetapkan. Terdapatnya fitur iOS, Android, dan Unity 3D, membuat aplikasi dapat digunakan dihampir seluruh jenis *smartphone*.



Gambar 2.2 Diagram Aliran Data Vuforia

Pada gambar 2.2 menunjukan aliran data yang dilakukan pada proses *scan marker* Vuforia. Dimana pada awal proses pengguna yang terlebih dahulu mendaftarkan *marker* kedalam *database*, lalu kamera akan memindai gambar sehingga gambar yang diterima sistem akan dilanjutkan pada proses *tracker*. Jika *marker* sesuai dengan gambar yang diambil oleh kamera, maka aplikasi akan me-*render graphic*, atau dengan kata lain menampilkan objek 3D.

2.6 Blender 3D

Blender 3D merupakan aplikasi untuk membuat gambar 3D yang dapat digunakan oleh siapapun (*Opensource*) [25]. tidak hanya membuat gambar 3D, Blender dapat digunakan untuk pengeditan video, membuat animasi, bahkan membuat *game* [5].

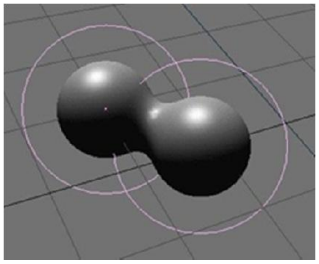
Terdapat beberapa fasilitas modeling yang dapat membantu dalam pembuatan gambar 3D pada Blender diantaranya: [4]

1. *Mesh Modeling*

   Teknik dasar *modeling* yang digunakan dalam pembuatan gambar 3D di sini terdapat teknik *box modeling* yang terdiri dari *object* primitif (*box, sphare, cylinder* dan sebagainya, Objek tersebut biasanya digunakan sebagai dasar dalam pembuatan gambar 3D. Blender mengenali *face* dengan 4 *vertek* atau 4 sisi sehingga pembuatan gambar akan lebih mudah.

2. *Metaball*

   Metaball merupakan fasilitas *modeling* yang jarang dipakai, *metaball* merupakan sebuah *object* berbentuk bola dan bersifat seperti cairan atau tanah liat, terdapat 2 buah *metaball* yaitu *metaball* positif dan negatif jika *metaball* positif didekatkan dengan *metaball* positif akan saling menyatu satu sama lain, namun penggunaanya pada Blender masih belum sempurna, *tools* masih terbatas.



Gambar 2.3 Metaball Pada Blender

3. *Curve*, NURBS (*Non-Uniform Rational B-Slines*), dan *Survace*

   Ini merupakan fasilitas umum, terdapat pula pada *software* Blender 3D, kelebihan menggunan pemodelan ini, data yang tersimpan lebih sedikit, hasil lebih bagus, namun terdapat kekurangan pada model ini, sangat sulitnya menambahkan detail yang kompleks, karena jumlah *poligonnya* sangat sulit dikendalikan.

2.7 Alat Musik Tradisional

Alat musik tradisional adalah alat musik masyarakat setempat yang dibuat dan diwariskan secara turun – temurun, dan berkelanjutan dalam masyarakat suatu daerah. Alat musik tradisional biasanya digunakan sebagai pengiring musik tradisional, pengiring seni tari daerah dan sebagai pengiring untuk ritual – ritual tertentu dalam upacara keagamaan. Tapi seiring berkembangnya jaman alat musik tradisional juga dapat digunakan sebagai pengiring untuk jenis musik lainnya ataupun suatu pertunjukan baik itu personal ataupun kelompok dengan tujuan tertentu.

Alat musik pada awalnya dibuat dari benda-benda disekitar yang mudah ditemukan seperti kerang atau kulit-kulit binatang dan juga bagian tanaman. Seiring berkembangnya zaman alat musik berevolusi dengan muncul berbagai macam variasi dan kualitas bahanpun ikut diperhatikan. Hampir semua yang terdapat dialam telah telah digunakan oleh setidaknya satu budaya untuk membuat alat musik.

Berikut adalah beberapa contoh dari alat musik tradisional di indonesia :

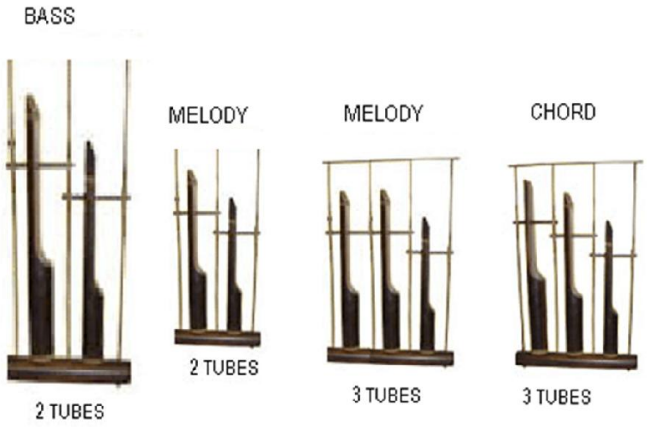
1. Angklung

   Angklung merupakan sebuah alat musik yang multitonal (bernada ganda). Alat musik tradisional ini banyak tumbuh dan berkembang di pulau Jawa bagian barat (tanah Sunda) [20]. Meskipun begitu terdapat juga angklung yang berasal dari luar pulau jawa seperti angklung bali.



Gambar 2.4 Angklung

Di indonesia angklung memiliki banyak jenis seperti angklung padaeng, angklung gubrak, angklung toel dan masih banyak lagi. Alat musik angklung ini dimainkan dengan cara menggoyangkanya, benturan antar pipa bambu akan menghasilkan bunyi. Tabung pipa bambu pada angklung pun jumlahnya berbeda beda, ada yang 2, 3, dan 4 untuk setiap nadanya, dan ada jug yang berdasarkan ukuran pipa tabungnya. Contohnya pada angklung padaeng, angklung ini terdiri dari beberapa nada seperti bass melody dan chord.



Gambar 2.5 Jenis nada angklung padaeng

Pada gambar 2.5 dapat kita lihat beberapa jenis angklung, jumlah tabung pipa bambunya berbeda berdasarkan jenis nadanya.



Gambar 2.6 Jenis nada angklung berdasarkan ukuran angklung

Pada gambar 2.6 dapat kita lihat jumlah pipa tabung bambunya sama tetapi ukurannya berbeda untuk setiap nadanya, dari yang paling kecil sampai yang ukurannya besar. Semakin kecil ukuran angklung dan tabung pipa bambunya maka semakin nyaring bunyi yang dihasilkan, begitupun sebaliknya.

Pada alat yang nantinya akan dibuat, cara memainkan alat musik ini dengan menggunakan *virtual button,* yaitu dengan cara mengarahkan jari ke suatu lokasi tertentu kemudian akan terdengar output suara nada dari media yang digunakan untuk *augmented reality* ini, speaker *handphone android* atau komputer.

2. Gamelan

Gamelan merupakan alat musik ensembel (kelompok alat musik) yang terdiri dari gambang, demung, gong, gendang, bonang, gong dan lain sebagainya. Gamelan ini merupakan alat musik yang berasal dari Jawa, namun juga bisa kita jumpai di Madura, Bali dan Lombok [20].



Gambar 2.7 Gamelan

Alat musik ini dimainkan dengan cara dipukul, ada yang menggunakan telapak jari tangan ada juga yang menggunakan tongkat khusus yang dibuat berdasarkan jenis gamelan tersebut. Pada aplikasi yang nantinya dibuat, cara memainkannya yaitu sama dengan alat musik sebelumnya yaitu menggunakan *virtual button.* Selain itu tidak semua alat musik gamelan yang nantinya akan ditampilkan, hanya beberapa saja.

3. Kecapi

Kacapi merupakan alat musik Sunda yang dimainkan sebagai alat musik utama dalam Tembang Sunda atau Mamaos Cianjuran dan kacapi suling [20]. Pada kecapi jumlah nadanya yaitu sebanyak 18 sampai 20 nada, tetapi ada kecapi yang ukurannya lebih mini atau kecil yang hanya memiliki 9 sampai 10 nada saja. Cara memainkan kecapi ini yaitu dengan cara dipetik.



Gambar 2.8 Kecapi

Pada aplikasi yang nantinya dibuat ukuran kecapinya yaitu ukuran kecil yang hanya memiliki 9 nada, dan cara memainkannya yaitu dengan menggunakan *virtual button.*

4. Suling

Suling merupakan alat musik tradisional yang dapat ditemui dihampir seluruh Provinsi di Indonesia. Akan tetapi Suling dari Jawa Barat memiliki bentuk dan suara yang unik apabila dimainkan untuk mengiringi lagu-lagu sunda.[20]



Gambar 2.9 Suling

Pada alat musik yang sebenarnya, alat musik ini dimainkan dengan cara ditiup. Sedangkan pada aplikasi yang dibuat cara memainkannya yaitu dengan menggunakan *virtual button.*

2.8 UML *(Unified Modelling Language)*

UML merupakan bahasa yang telah menjadi standar untuk melakukan perancang, *visualisasi* dan mendokumentasikan sistem pada piranti lunak, dengan UML kita dapat memodelkan aplikasi yang dapat berjalan pada sistem operasi dan dapat ditulis menggunakan berbagai macam bahasa pemrograman, namun bahasa pemrograman yang sering digunakan dalam pembuatan *software* yaitu bahasa yang berorientasi objek seperti C++, JAVA, C# atau VB.NET [22].

Notasi UML merupakan hasil pengembangan 3 notasi yang telah ada sebelumnya, yaitu : [22]

- Grady Booch OOD (*Object-Oriented Design*).
- Jim Rumbaugh OMT (*Object Modeling Technique*).
- Ivar Jacobson OOSE (*Object-Oriented Software Engineering*).

Beberapa konsep dasar UML, yaitu *structural, classification, dynamic, behavior,* dan model *management*. Dapat kita lihat pada gambar di bawah ini, sama halnya dengan apa yang ada pada gambar UML mendefinisikan beberapa diagram: [24]

Tabel 2.1 Konsepsi Dasar UML

Major Area	*View*	*Diagrams*	*Main Concepts*
structural	static view	class diagram	class, association, generalization, dependency, realization, interface
	use case view	use case diagram	use case, actor, association, extend, include, use case generalization
	implementation view	component diagram	component, interface, dependency, realization
	deployment view	deployment diagram	node, component, dependency, location
dynamic	state machine view	statechart diagram	state, event, transition, action
	activity view	activity diagram	state, activity, completion transition, fork, join
	interaction view	sequence diagram	interaction, object, message, activation
		collaboration diagram	collaboration, interaction, collaboration role, message
model management	model management view	class diagram	package, subsystem, model
extensibility	all	all	constraint, stereotype, tagged values

1. *Use Case* Diagram

   Menekankan kepada apa yang akan diperbuat sistem, sebuah *use case* menggambarkan interaksi aktor dengan sistem, aktor dapat berupa manusia maupun mesin. Sebuah *use case* dapat *include* terhadap *use case* lain, dapat digambarkan ketika ada *use case* yang *include* dengan *use case* yang di eksekusi maka *use case* yang *include* tersebut akan ikut dieksekusi, tidak hanya satu *use case* yang dapat *include* pada sebuah *use case,* lebih dari satu *use case* dapat *include* pada sebuah *use case.* Sebuah *use case* pun dapat meng-*extend use case* lain dengan *behavior-nya* sendiri.

2. *Class* Diagram

   Class Diagram adalah sebuah sepesifikasi yang diinstansikan akan menghasilkan objek, *class* diagram memiliki tiga area pokok : nama, atribut, dan metode. Pada atribut dan metode dapat memiliki salah satu sifat ini. *private, protected, public* . Hubungan antara *class* pada *class* diagram:

   a. Asosisasi, yaitu hubungan statis antara *class.* Asosiasi *class* pun memiliki atribut berupa *class* lain, panah penunjuk menunjukan arah *query* antara *class.*
   b. Agregasi, yaitu hubungan yang menyatakan bagian yang di dalamnya terdiri atas *class-class.*
   c. Pewarisan, yaitu hubungan hirarki antara *class. Class* dapat diturunkan kepada *class* lain.
   d. Hubungan dinamis, yaitu rangkaian pesan yang di-passing dalam satu *class* kepada *class* lain.

3. *Sequence* Diagram

   Diagram ini menggambarkan interaksi antara objek di dalam dan objek di sekitar sistem. Diagram ini dapat digunakan untuk menggambarkan skenario dan langkah-langkah yang dilakukan.

4. *Collaboration* Diagram

   Diagram ini menggambarkan interaksi antara objek, namun lebih menekankan pada peranan masing-masing objek seperti *sequence* diagram

dan bukan merupakan penyampaian pesan, setiap pesan memiliki *sequence number*.

2.9 Skala Likert

Skala Likert merupakan skala yang digunakan untuk mengukur pendapat, seseorang maupun kelompok mengenai sebuah objek, berdasarkan beberapa definisi yang telah ditetapkan oleh peneliti.

Peneliti mengumpulkan item-item yang sesuai dengan masalah yang sedang diteliti, terdiri dari item yang dapat memperjelas apakah objek tersebut disukai maupun tidak disukai. Kemudian item-item itu diberikan kepada sekelompok responden. Responden diminta untuk mengecek tiap item, apakah ia menyenangi (+) atau tidak menyukainya (-) [9]. Respon tersebut dikumpulkan dan jawaban yang memberikan indikasi menyenangi diberi skor tertinggi. Dengan memberikan angka 5 untuk skor tertinggi dan skor 1 untuk yang terendah. Yang terpenting adalah konsistensi dari arah sikap yang diperlihatkan. Demikian juga apakah jawaban "setuju" atau "tidak setuju" disebut yang disenangi, tergantung dari isi pertanyaan dan isi dari item-item yang disusun [12].

Total skor dari responden adalah penjumlahan dari skor masing-masing responden tersebut. Respon dianalisis untuk mengetahui item-item mana yang sangat terlihat perbedaanya antara skor tinggi dan skor rendah dalam skala total. Misalnya, responden pada *upper* 25% dan *lower* 25% dianalisis untuk melihat sampai berapa jauh tiap item dalam kelompok ini berbeda. Item-item yang menunjukkan hasil, apakah masuk dalam skor tinggi atau rendah juga dibuang untuk mendapatkan total skor yang tepat dari beberapa pertanyaan [9].

Untuk mencari presentase dari masing-masing jawaban kuesioner digunakan rumus Skala Likert sebagai berikut:

$$P = \frac{S}{\text{Skor Ideal}} \text{ x } 100\%$$

Keterangan:

P = Nilai presentase yang dicari

S = Jumlah frekuensi dikalikan dengan nilai skala jawaban

Skor Ideal = Skala tertinggi jawaban dikalikan jumlah responden

2.10 Alpha Beta Testing

Alpha testing merupakan sebuah simulasi atau pengujian terhadap *software* yang sudah dibangun yang dilakukan pengembang, calon pengguna aplikasi, pelanggan, maupun tim uji sebelum *sofware* tersebut dipasarkan. Sedangkan Beta testing merupakan pengujian yang dilakuakan setelah *software* sudah digunakan oleh pengguna, ini untuk menentukan apakah aplikasi tersebut memenuhi kebutuhan pengguna. Pengujian Beta sering digunakan untuk mendapatkan umpan balik dari pengguna *software* [22].

2.10.1 Metode Black-box

Pengujian *Black-box* merupakan peran yang sangat penting dalam pengujian *software*, membuat aplikasi berfungsi secara baik. Pengujian ini dilakukan berdasarkan keinginan pelanggan agar semua keinginanya terpenuhi dan keinginan yang tidak terduga dapat diidentifikasi dengan mudah, pengujian *black-box* digunakan berdasarkan persefektif penggunaan akhir untuk mangatasi kegagalan dan masukan dari pelanggan yang membuat kegagalan pada aplikasi [2].