

BAB 2

TINJAUAN PUSTAKA

2.1 Profil Instansi

2.1.1 Visi

Berdasarkan analisis kondisi eksternal dan internal di SD Negeri 007 Sekupang Kota Batam Provinsi Kepulauan Riau maka ditetapkan Visi Sebagai berikut :

“Terwujudnya siswa yang beriman, prestasi, dan kreatif serta berbudaya melayu”.

Visi ini tentunya ingin menjadikan **Sekolah Dasar Negeri 007 Sekupang** – Kota Batam, sebagai lembaga pendidikan dasar yang profesional dan berkualitas melalui penerapan serta *mengedepankan mutu*, kondusif dalam pembelajaran dengan Budaya Belajar, Budaya Religius serta Kerjasama Kemitraan dengan berbagai unsur dan elemen masyarakat sekitar sekolah.

2.1.2 Misi

Memandang kepada visi tersebut di atas, “Misi” **Sekolah Dasar Negeri 007 Sekupang** – Kota Batam, adalah sebagai berikut:

1. Meningkatkan penguasaan ilmu pengetahuan dan teknologi dengan mengoptimalkan potensi akademik yang dimiliki oleh setiap siswa.
2. Meningkatkan pelayanan pembelajaran kepada siswa melalui proses KBM yang aktif dan menyenangkan.
3. Membina budi pekerti yang luhur dan berbudaya melayu sesuai dengan tutunan agama, dan budaya bangsa.
4. Membina, menggali, memupuk, mengembangkan bakat, minat prestasi siswa dalam bidang seni, olah raga, keterampilan melalui penyelenggaraan ekstrakurikuler yang efektif.

Media sebagai salah satu komponen dalam sistem itu, mempunyai fungsi sebagai sarana komunikasi non-verbal. Sebagai salah satu komponen sistem, berarti media mutlak harus ada atau harus dimanfaatkan di dalam setiap pembelajaran[8]. Adapun jenis media pembelajaran terdiri dari 2 metode yaitu metode konvensional dan non konvensional. Dalam metode konvensional alat yang digunakan dalam proses belajar mengajar yaitu berbentuk ceramah, pengadaan bukubuku dan alat peraga, serta pada metode non konvensional alat yang digunakan berupa media visual meliputi gambar atau foto, grafik, peta konsep, poster, dan media audio seperti penggunaan suara, musik serta audio visual[4]

2.2.2 Augmented Reality

Augmented Reality (AR) adalah teknologi yang menggabungkan informasi virtual dua dimensi ataupun tiga dimensi ke dalam lingkungan dunia nyata. Secara teknis *Augmented Reality* termasuk dalam *multimedia*, Prinsipnya adalah menerapkan informasi komputer visual seperti teks, gambar, model 3D, musik dan video. Dengan cara seperti ini, kedua jenis informasi tersebut saling melengkapi sehingga benda-benda maya tersebut ke dalam waktu nyata (*real-time*)[9].

augmented Reality merupakan salah satu teknologi yang memanfaatkan kamera *smartphone* untuk menerapkannya. Teknologi ini sudah banyak dikembangkan oleh *developer software smartphone* karena sebelumnya pernah dikembangkan pada perangkat PC[10]. Ada beberapa metode yang terdapat pada *Augmented Reality* salah satunya adalah metode *Marker-Based*.

2.2.2.1 Marker-Based Augmented Reality

Marker-based tracking adalah metode pelacakan menggunakan marker atau gambar. Penggunaan *marker* pada aplikasi *Augmented Reality* sering dipilih selain cocok diimplementasikan pada

lingkungan nyata sebagai aspek pembelajaran dan juga cenderung cepat dalam hal mendeteksi objek pada proses munculnya objek 3D[11].

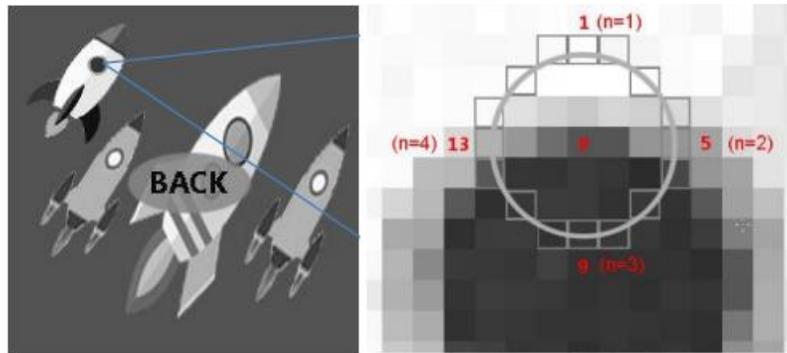
Untuk membuat aplikasi *Augmented Reality Marker Based Tracking* hal penting paling utama yang diperlukan adalah membuat *marker* kartu yang berisi gambar, gambar yang dipakai bisa berupa hitam-putih ataupun warna lain asalkan gambar yang dihasilkan dapat terdeteksi oleh kamera sebagai tempat/lokasi sehingga aplikasi dapat memproses gambar tersebut untuk menampilkan object 3D yang sudah ditambahkan sebelumnya di dalam aplikasi *unity*[12].

2.2.3 Vuforia

Vuforia adalah *Software Development Kit (SDK) Augmented Reality* untuk perangkat mobile yang mendukung pembuatan aplikasi *Augmented Reality*. *Software Development Kit (SDK) Vuforia* bisa digabungkan dengan aplikasi Unity yang bernama *Vuforia AR Extension for Unity*, Vuforia merupakan SDK yang disediakan oleh Qualcomm untuk membantu para *developer* membuat aplikasi *Augmented Reality (AR)* di *mobile phones* (iOS, Android). Terdapat dua jenis *workflow* dengan dasar *database* yang dapat dipilih oleh *developer*, adalah *Cloud Database* dan *Device Database*[13].

AR Vuforia memberikan cara berinteraksi yang memanfaatkan kamera *mobile phones* untuk digunakan sebagai perangkat masukan, sebagai mata elektronik yang mengenali penanda tertentu, sehingga di layar bisa ditampilkan perpaduan antara dunia nyata dan dunia yang digambar oleh aplikasi[12]. Pengenalan mengenai objek 3D menggunakan Algoritma *fast corner* yang sudah disediakan oleh Vuforia sehingga mampu mengenali *marker* maupun *non-marker* secara spesifik. Algoritma *Fast Corner* merupakan sebuah algoritma untuk mendeteksi sudut-sudut yang ada pada objek. Memiliki tujuan agar tingkat akurasi menurun yang ada pada deteksi

sudut secara *real-time*. Aturan di algoritma menetapkan titik pada sudut dengan titik sudut p yang akan di masukkan data dengan 16 piksel pada titik p . Contoh pada gambar 2 yaitu menentukan pada setiap masing-masing perbedaan.



Gambar 2. 2 Algoritma *Fast corner*

$$C = \begin{cases} I_p - I_n < t & \text{normal} \\ I_n - I_p > t & \text{brighter} \\ I_p - I_n > t & \text{darker} \end{cases}$$

Keterangan :

I_p = intensitas titik pusat,

I_n = titik intensitas tetangga ke- n

t = *threshold*

2.2.4 Unity 3D

Unity Engine adalah game engine yang terus berkembang sejak dibangun pada tahun 2004. Unity tidak membatasi publikasi aplikasi, pengguna unity dengan lisensi gratis dapat mempublikasikan aplikasi yang dibuat. Unity *Engine* memiliki kerangka kerja (*framework*) lengkap untuk pengembangan profesional. Sistem unity engine ini menggunakan beberapa pilihan bahasa pemrograman, diantaranya C#, javascript maupun boo. Unity3D editor

menyediakan beberapa alat untuk mempermudah pengembangan yaitu Unity *Tree* dan *terrain creator* untuk mempermudah pembuatan vegetasi dan terrain serta *MonoDevelop* untuk proses pemrograman[14]. 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 3D[15].

2.2.5 Android

Android merupakan sistem operasi yang dikembangkan untuk perangkat mobile berbasis Linux. Pada awalnya sistem operasi ini dikembangkan oleh Android Inc. yang kemudian dibeli oleh Google pada tahun 2005. Pada tahun 2007 dibentuklah Open Handset Alliance (OHA), sebuah konsorsium dari beberapa perusahaan, yaitu Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, dan T-Mobile dengan tujuan untuk mengembangkan standar terbuka untuk perangkat mobile[16].

Android adalah sistem operasi mobile dan bersifat *open source* yang dikembangkan Google Corporation yang merupakan perusahaan mesin pencari terkemuka di dunia. Para pengembang dapat membuat aplikasi dengan menggunakan platform Android untuk berbagai perangkat bergerak. Android menjadi sistem operasi yang sangat populer karena tingkat efektivitas dan efisiensinya yang lebih baik dibandingkan dengan program sejenis lainnya, sehingga Android juga populer digunakan untuk kepentingan pendidikan karena kemudahan dan fleksibilitasnya[17]. Berikut adalah sistem operasi Android yang telah rilis:

Tabel 2. 1 Android

Versi	Rilis
Android 1,0 (Alpha)	23 September 2008
Android 1.1 (Beta)	9 Februari 2009
Android 1,5 (Cupcake)	27 April 2009
Android 1,6 (Donut)	15 September 2009
Android 2.0-2.1 (Elcair)	23 Oktober 2009
Android 2.2 (Froyo)	20 May 2010
Android 2.3 (Gingerbread)	6 December 2010
Android 3.0-3.2 (Honeycomb)	22 Februari 2011
Android 4.0 (Ice Cram Sandwich)	19 Oktober 2011
Android 4.1-4.3 (Jelly Bean)	27 Juni 2012
Android 4.4 (KitKat)	3 September 2013
Android 5.0	25 Juni 2014
Android 6.0 (Marshmallo)	5 Oktober 2015
Android 7.0-7.1 (Nougat)	9 Maret 2016
Android 8.0-8.1 (Oreo)	7 Agustus 2017
Android 9 (Pie)	6 Agustus 2018
Android 10 (Android Q)	3 September 2019
Android 11	8 September 2020
Android 12	4 Oktober 2021
Android 13	10 Februari 2022

2.2.6 Model 3D

Model 3D adalah perwujudan atau representasi hasil dari proses secara matematika yang membentuk objek 3D. *Modelling* adalah proses pembuatan bentuk- bentuk virtual yang melengkapi suatu *scene*. Maka dari itu, dapat disimpulkan bahwa Model 3D merupakan sebuah proses pembuatan bentuk virtual yang melengkapi suatu *scene* dalam bentuk 3D atau 3 dimensi. Berdasarkan aspek produksi, konstruksi, dan klasifikasi evaluasi, model 3D dibagi menjadi 2 kelompok yaitu objek *Hard Surface (Non-Organic)* dan *Organic*[18].

Sejarah singkat model 3D dimulai dari penggunaan *computer-aided design (CAD)* untuk sistem simulator penerbangan militer, ruang angkasa, dan industri otomotif pada tahun 1950-an dan 60-an. Pada tahun 1970-an dan awal 80-an terdapat peningkatan teknologi yang pada akhirnya memunculkan peningkatan teknik realisme 3D pula, contohnya sutradara George Lucas mulai menggunakan material 3D untuk menghasilkan *scene* yang sulit pada film bergenre fantasi-nya [19]. Kebutuhan dalam industri hiburan jelas berperan penting dalam pengembangan teknik model 3D. Pada tahun 1990-an, terdapat perkembangan teknologi komputer yang memberikan masyarakat akses terhadap *software* 3D yang lebih mudah digunakan, dan sampai sekarang teknologi 3D familiar digunakan masyarakat luas untuk berbagai bidang, seperti industri hiburan, medis, dan pendidikan.

2.2.7 Bahasa Pemrograman C#

Bahasa pemrograman C#, (dibaca: C Sharp), merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh Microsoft pada awal 2000-an, dipimpin oleh Anders Hejlsberg. Pemrograman berorientasi objek adalah pendekatan pemrograman yang memecah masalah pemrograman menjadi objek yang berinteraksi satu sama lain[20]. bahasa pemrograman ini merupakan bagian dari kerangka .Net yang bertujuan untuk

menjadi bahasa pemrograman yang umum dan sederhana yang dapat digunakan untuk mengembangkan berbagai jenis aplikasi, termasuk aplikasi konsol, windows, web, dan seluler. Seperti pada bahasa pemrograman lainnya, bahasa pemrograman C# pun harus diubah menjadi bahasa mesin menggunakan *compiler* agar dapat dipahami komputer.

2.2.8 Use Case Diagram

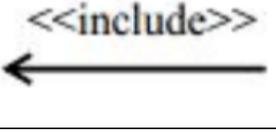
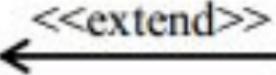
Use case adalah metodologi yang digunakan dalam analisis sistem untuk mengidentifikasi, mengklarifikasi, dan mengatur persyaratan sistem. *Use Case Diagram* diusulkan oleh Ivar Jacobson pada tahun 1986 dan digunakan dalam UML (*Unified Modeling Language*) yang merupakan notasi standar untuk pemodelan objek dunia nyata dan sistem [21].

Use Case Diagram memiliki beberapa komponen, yaitu:

1. Aktor, yang merupakan perseorangan, kelompok individu, organisasi, atau sistem eksternal yang memainkan peran dan berinteraksi dalam suatu sistem. Aktor dapat berupa jaringan, komunikasi, perangkat, komputer, atau program lain di computer yang sama.
2. *Use case*, yang merupakan urutan dari berbagai aksi atau tindakan. Tindakan tersebut harus memiliki nilai terukur yang bermakna kepada actor.
3. Asosiasi, yang merupakan interaksi yang dijelaskan oleh *use case*.
4. Batas sistem, yang merupakan ruang batas pada use case. Apapun yang termasuk dalam batas sistem bersifat fungsional dalam sistem.

Tabel 2. 2 Use Case Diagram

Simbol	Keterangan
--------	------------

	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case
	Use Case : Abstraksi dan interaksi antara sistem dan aktor
	Association : Abstraksi dari penghubung antara aktor dan use case
	Generalisasi : Menunjukkan spesialisasi actor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
	Extend menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi

2.2.9 Black Box Testing

Pengujian merupakan suatu proses pelaksanaan suatu program dengan tujuan menemukan suatu kesalahan. Pengujian dianggap baik atau sukses apabila berhasil mengungkap kesalahan yang tidak diketahui awalnya. Salah satu bentuk pengujian adalah *Black Box Testing*. *Black Box Testing* memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya sehingga para *tester* memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi cukup dikenai proses testing di bagian luar [23]. Metode *Black Box Testing* sendiri terdiri dari beberapa cara, yaitu *Equivalence Partitioning*, *Boundary Value Analysis*,

Comparison Testing, Sample Testing, Robustness Testing, dan lain-lain[24].

Black Box Testing juga cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.