

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Dasar Media Pembelajaran**

##### **2.1.1 Pengertian Media Pembelajaran**

Gerlach dan Peeley menjelaskan media dari perspektif luas dan sempit. Dari sudut pandang luas, media adalah orang, materi, atau peristiwa yang dapat menciptakan kondisi dimana siswa dapat memperoleh pengetahuan, keterampilan, atau sikap baru. Dalam pengertian ini pembelajaran, buku, dan lingkungan sekolah merupakan media. Dari perspektif yang lebih sempit, media mengacu pada grafik, potret, gambar, dan perangkat mekanis dan elektronik yang digunakan untuk menangkap, memproses, dan mengirimkan informasi visual dan linguistik. Semua media adalah alat untuk mencapai tujuan. [1].

Gagne dan Briggs juga menjelaskan secara rinci bahwa media pembelajaran meliputi alat-alat yang digunakan secara fisik untuk menyampaikan isi materi pendidikan, antara lain buku, alat perekam, film, slide (bingkai), foto, gambar, grafik, televisi, dan komputer [1]. Hamalik menjelaskan bahwa Media pembelajaran adalah alat, metode dan teknik yang digunakan untuk membantu komunikasi dan interaksi antara guru dan siswa agar lebih efektif dalam proses pendidikan dan pengajaran di sekolah [2].

Dari berbagai teori yang dikemukakan, kita dapat menyimpulkan bahwa yang dimaksud dengan media adalah suatu benda, alat atau komponen yang dapat digunakan untuk menyampaikan pesan dari pengirim ke penerima. Secara umum

media pembelajaran adalah suatu alat yang menyediakan bahan pembelajaran sebagai bagian dari proses belajar mengajar.

### **2.1.2 Jenis Media Pembelajaran**

Menurut Sriyanti dalam M. Hanafi [3], ada beberapa jenis media yang digunakan dalam pembelajaran, yaitu :

1. Media Cetak

Kelebihan dari media cetak yaitu dapat diraba, dilihat, dan dibaca secara langsung. Ini memberikan keuntungan dalam menyampaikan pesan dengan cara yang lebih keterlihatan dan dapat ditemukan di tempat-tempat tertentu untuk dibaca kapan saja. Namun kekurangannya yaitu Media cetak bersifat statis, artinya pembaca hanya dapat menerima informasi yang dicetak di halaman tersebut. Tidak ada interaksi langsung atau kemampuan untuk berpartisipasi secara aktif.

2. Transparasi

Media ini memiliki sifat yang praktis dan gampang untuk dioperasikan dan media ini juga mendukung pembelajaran dengan secara langsung (tatap muka).

3. Multi Media Interaktif

Media pembelajaran ini sangat efektif sebagai proses pendukung pembelajaran. Karena media ini memiliki kemampuan untuk berinteraksi dengan pengguna. Siswa dapat terlibat aktif dalam proses belajar dengan melakukan tindakan seperti mengklik, mengetik, atau memilih opsi tertentu. Dan media ini juga dapat menciptakan pengalaman belajar yang menarik

dan menantang. Pengguna dapat terlibat dalam simulasi, permainan, atau latihan interaktif yang membuat pembelajaran lebih menarik.

#### 4. *E-learning*

Media ini dapat mencakup berbagai format, termasuk teks, video, audio, animasi, simulasi, dan lainnya. Ini memberikan variasi dalam metode penyampaian materi dan memenuhi kebutuhan pembelajaran yang beragam. Media ini juga dapat menyediakan elemen-elemen interaktif seperti kuis online, forum diskusi, dan tugas daring.

#### 5. *M-learning*

Media ini mengarah pada penggunaan perangkat mobile, seperti smartphone, tablet, atau laptop, sebagai alat pembelajaran. Media ini juga memberikan fleksibilitas kepada peserta didik untuk mengakses materi pembelajaran kapan saja dan di mana saja, selama mereka memiliki perangkat mobile dan koneksi internet.

### **2.1.3 Bangun Ruang**

Dalam Kamus Besar Bahasa Indonesia, bangun datar merupakan salah satu cabang ilmu matematika yang menjelaskan sifat-sifat garis, sudut, bidang, dan ruang. Geometri juga bisa disebut ilmu pengukuran. Mata pelajaran geometri terbagi menjadi dua kelompok, yaitu geometri yang berbentuk bangun datar dan geometri yang berbentuk bangun ruang. Dalam hal ini pada bagian bangun datar pembelajaran memfokuskan untuk mempelajari suatu bangun yang bersifat dua dimensi, sedangkan bangun ruang berfokus pada bangun yang sifatnya tiga dimensi.

Menurut Saputra dan Permata dalam Alyusfitri Geometri merupakan sesuatu yang sangat erat kaitannya dengan desain, karena ruang lingkupnya meliputi garis dan sudut, bentuk bidang, bentuk ruang, simetri, kemiripan, kongruensi, dan geometri analitik [4]. Rahayu dan hidayat dalam Alyusfitri juga menyebutkan bahwa dalam pembelajaran geometri, siswa perlu memiliki konsep yang matang untuk mampu menerapkan keterampilan geometri seperti visualisasi, pengenalan bangun datar dan bangun ruang, serta mendeskripsikan gambar. [4].

Adapun sesuai kurikulum yang terdapat pada SD 35 Bilah Hulu yakni kurikulum 2013, maka didapat kompetensi dasar yang berkaitan dengan bangun ruang didalamnya.

**Tabel 2. 1 KD Materi Bangun Ruang SD Kelas VI**

<b>Kompetensi Dasar Pengetahuan</b>	<b>Kompetensi Dasar Keterampilan</b>
1. Menjelaskan taksiran keliling dan luas lingkaran	1. Menaksir luas dan keliling lingkaran serta digunakan untuk menyelesaikan masalah
2. Membandingkan prisma, tabung, limas, kerucut dan bola	2. Mengidentifikasi prisma, tabung, limas, kerucut dan bola
3. Menjelaskan bangun ruang yang tersusun dari beberapa bangun, serta luas dan volumenya	3. Mengidentifikasi bangun ruang yang tersusun dari beberapa bangun, serta luas dan volumenya

Sumber : (<https://ciptacendekia.com/>)

#### 2.1.4 *Augmented Reality*

Menurut Abdulghani dalam S. Yulianti menjelaskan bahwa *Augmented reality* (AR) adalah istilah untuk suatu lingkungan yang menggabungkan dunia nyata dan dunia maya yang diciptakan oleh komputer sehingga batas antara kedua dunia tersebut menjadi sangat rapuh (tipis) [5]. Ronald T. Azuma dalam Indah Purnama Sari menjelaskan *Augmented reality* adalah kombinasi objek nyata dan virtual dalam lingkungan nyata yang berinteraksi secara real time dan dapat diintegrasikan dan divirtualisasikan dengan teknologi tampilan yang sesuai, dapat berinteraksi melalui beberapa perangkat input dan integrasi yang baik memerlukan penjelasan yang efektif [6]. Sedangkan dalam buku yang berjudul *Augmented reality a partial guide* karya Stephen Coward dan Mark Faila menjelaskan bahwa *Augmented reality* adalah cara alami untuk mengeksplorasi objek dan data 3D, AR adalah sebuah konsep yang menggabungkan realitas visual dengan realitas dunia.

Jadi, dengan objek virtual berteknologi AR dua dimensi (3D), pengguna dapat melihat dunia nyata di sekitarnya dengan tambahan objek virtual yang dihasilkan komputer. Berdasarkan definisi yang telah dijabarkan bisa ditarik kesimpulan bahwa, secara umum *Augmented Reality* bisa diartikan sebagai dunia nyata yang dimasukkan kedalamnya objek virtual dengan integrasi teknologi komputer.

Dan juga disimpulkan bahwa sistem *augmented reality* merupakan sistem yang digabungkan dengan teknologi kamera sehingga dapat memunculkan objek 3D pada lingkungan dunia nyata yang dapat dilihat pada *device* seseorang yang menggunakannya.

### 2.1.5 Metode *Augmented Reality*

Menurut Lyu dalam M. Hanafi [3], ada 2 jenis metode dalam *augmented reality*, yaitu :

#### 1. *Marker Based Tracking*

Metode ini sudah dikenal lama dalam pembuatan teknologi *augmented reality*. Dalam sistem ini untuk memunculkan bentuk *Reality* maka diperlukan marker (penanda) berupa gambar yang nantinya akan dianalisis oleh sistem sehingga pengguna (*user*) dapat melihat objek virtual. Sama halnya seperti yang diutarakan oleh Reynaldo bahwa Cara kerja metode ini adalah masuk ke mode kamera lalu arahkan perangkat ke penanda yang telah ditentukan, seperti gambar cetakan binatang. Sistem kemudian akan mengenali lokasi dan penanda untuk menampilkan objek 3D. *Augmented reality* (AR) memiliki konsep dasar. Konsep AR adalah kombinasi objek nyata dan awan dalam lingkungan nyata. Hasil ditampilkan secara interaktif dan real time. [7]. Contoh metode marker seperti pada Gambar 2.1.



Sumber : (<https://wear-studio.com/ar-qr-codes/>)

**Gambar 2. 1 Contoh Tampilan *Marker Based AR***

## 2. *Markerless Augmented Reality*

Pada metode ini tidak perlu memerlukan bantuan sebuah *Marker* untuk memunculkan objek virtual. Sistem *Markerless* dapat memanfaatkan fitur dalam *smartphone*, contohnya seperti pada Teknik *GPS Based Tracking* yang menggunakan fitur *GPS* pada *smartphone*, sehingga aplikasi bisa secara *real-time* menunjukkan arah yang diinginkan. Saputra juga menjelaskan bahwa Metode *markerless* tidak bergantung pada pola atau pola tertentu seperti pada metode *marker*, metode ini dapat dideteksi oleh kamera *smartphone* pengguna pada objek apapun yang dapat dideteksi oleh kamera asalkan ada model objeknya, seperti objek yang dapat diamati secara tiga dimensi dan akan dimunculkan di layar [8]. Contoh *markerless* AR pada Gambar 2.2.



Sumber : (<https://www.fi.edu/difference-between-ar-vr-and-mr>)

**Gambar 2. 2 Contoh Tampilan *Markerless Based AR***

### **2.1.6 *Android***

Menurut Andi, *Android* adalah sistem operasi perangkat seluler berbasis Linux yang mencakup sistem operasi, middleware, dan aplikasi. Android menyediakan platform terbuka bagi pengembang untuk membuat aplikasinya sendiri [9].

Perusahaan pertama kali yang mengembangkan sistem operasi *android* adalah *Android Inc*, namun pada tahun 2005 perusahaan Google mengakuisisinya. Adapun dalam pengembangannya Android dibuat melalui sistem operasi *linux*. Penjelasan ini juga sejalan dengan apa yang disampaikan Safaat dalam D. Savitri bahwa *Android* adalah sistem operasi untuk perangkat pada *smartphone* yang berbasis *Linux*, meliputi sistem operasi, *middleware* dan suatu aplikasi. [10]. Setiap perkembangan *Android* akan ditandai dengan diberinya nama sesuai dengan nama hidangan makanan.

## **2.2 Metode Perancangan Sistem**

Pada bagian metode perancangan sistem akan berisikan hal-hal yang berkaitan dengan metode yang digunakan dalam membangun suatu aplikasi. Fungsi dari pada penggunaan metode perancangan sistem saat membuat atau membangun suatu aplikasi yaitu agar *developer* atau pengembang dapat dengan mudah melihat alur kerja sistem yang akan dibuat, pengembang juga dapat membuat aplikasi dengan lebih terstruktur dikarenakan adanya tahapan-tahapan yang saling berurutan. Adapun pada penelitian kali ini peneliti akan mengembangkan suatu aplikasi bangun ruang berbasis *augmented reality* menggunakan metode perancangan sistem *waterfall*.

### **2.2.1 System Development Life Cycle**

Siklus hidup pengembangan sistem (SDLC) adalah proses logis yang digunakan oleh seorang analis sistem untuk mengembangkan sistem informasi yang mencakup persyaratan, validasi, pelatihan, dan pemilik sistem [11]. Pola dalam metode pengembangan ini adalah terdiri dari 6 tahap, yaitu *planning* (perencanaan), *analyst* (analisis), *design* (desain), *implementation* (implementasi), *testing* (uji coba), dan *maintenance* (pemeliharaan).

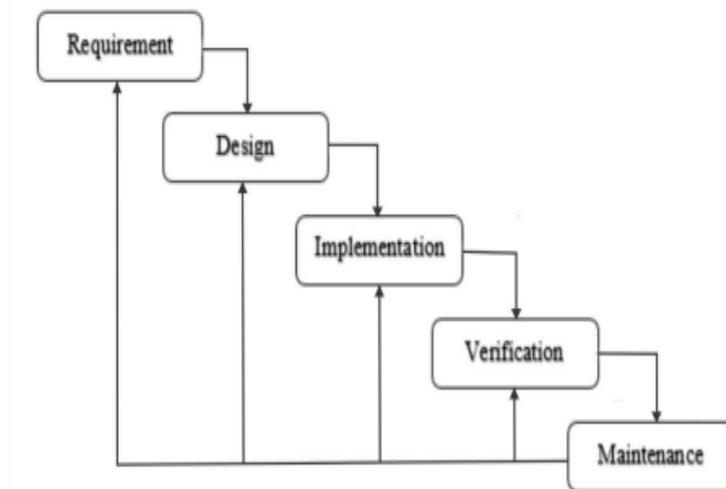
### **2.2.2 Metode Waterfall**

Sukanto & Shalahuddin dalam Y. Handrianto menjelaskan bahwa Model Air Terjun (*waterfall*) adalah “model yang didalamnya memberikan pendekatan berurutan atau teratur terhadap siklus hidup perangkat lunak yang dimulai dengan analisis, desain, pengkodean, dan pengujian” [12]. Metode air terjun atau bisa juga kita sebut dengan metode *waterfall* sering disebut dengan siklus hidup klasik, nama model ini sebenarnya adalah “Linear Sequential Model” yang didalamnya menggambarkan bagaimana pendekatan pendekatan sistematis dan sekuensial dalam pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut ke dalam tahapan-tahapan perencanaan (*planning*), permodelan (*modelling*), konstruksi (*contruction*), serta penyerahan sistem ke para pengguna (*deployment*), yang diakhiri dengan dukungan pada *software* lengkap yang dihasilkan. Adapun mengapa peneliti memakai metode perancangan sistem *waterfall* ini dikarenakan alur-alur pengembangannya yang terstruktur sehingga memudahkan dalam merancang suatu aplikasi, dan mudah dalam melihat fungsi-fungsi dalam sistemnya.

### 2.2.3 Tahap-tahap *Metode Waterfall*

Metode waterfall memiliki beberapa tahap, dapat dilihat pada Gambar 2.3.

Dibawah ini



*Sumber : (A. A. Wahid, Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi, 2020) [11]*

**Gambar 2.3 Contoh Tahap *Waterfall***

#### 1. *Requirement*

Pada tahap ini, pengembang sistem perlu berkomunikasi dengan tujuan memahami apa yang diharapkan pengguna dari perangkat lunak serta keterbatasan perangkat lunak. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi tersebut dianalisis untuk mendapatkan data yang diperlukan bagi pengguna.

#### 2. *Design*

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan persyaratan perangkat keras dan sistem, serta menentukan arsitektur sistem secara keseluruhan.

### 3. *Implementation*

Pada tahap ini, sistem dikembangkan terlebih dahulu menjadi program-program kecil yang disebut unit, yang kemudian diintegrasikan pada tahap-tahap berikutnya. Setiap unit yang dikembangkan dan diuji fungsinya disebut pengujian unit. Selain itu, pada tahap ini juga dilakukan pengujian dan verifikasi fungsional terhadap modul yang dibuat, apakah memenuhi kriteria yang diinginkan atau tidak.

### 4. *Verification*

Pada titik ini, sistem diperiksa dan diuji untuk melihat apakah sistem memenuhi persyaratan sistem secara keseluruhan atau sebagian. Pengujian dapat diklasifikasikan menjadi pengujian unit (dilakukan pada modul kode tertentu) dan pengujian sistem (untuk melihat bagaimana sistem bereaksi) ketika semua modul terintegrasi) dan pengujian penerimaan (dilakukan dengan atau atas nama pelanggan untuk melihat apakah semua kebutuhan pelanggan terpenuhi).

### 5. *Maintenance*

Ini adalah tahap akhir dari metode *waterfall*. Perangkat lunak yang telah selesai diimplementasikan dan dipelihara. Pemeliharaan mencakup perbaikan kesalahan yang tidak ditemukan pada langkah sebelumnya. Pemeliharaan memungkinkan pengembang untuk memperbaiki kesalahan yang tidak terdeteksi pada langkah sebelumnya. Pemeliharaan mencakup perbaikan kesalahan, perbaikan implementasi unit sistem, serta peningkatan dan penyesuaian sistem bila diperlukan.

## **2.3 Alat Bantu Perancangan**

### **2.3.1 *Unified modelling language***

*Unified modelling language* (UML) merupakan suatu standar bahasa yang digunakan untuk menuliskan perancangan (*blueprints*) suatu perangkat lunak. Menurut Ary Budi Warsito, Muhamad Yusup., Moh. Iqbal Awi Makaram dalam A. Setiawan, mengatakan bahwa *Unified Modeling Language (UML)* adalah seperangkat kerangka kerja dan teknik untuk memodelkan desain program berorientasi objek (OOP) dan aplikasinya [13]. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok alat yang mendukung pengembangan sistem tersebut. Pada penelitian ini desain UML yang akan digunakan adalah *use case diagram*, *activity diagram*, *state machine diagram*, *class diagram* dan *sequence diagram*.

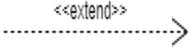
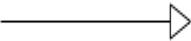
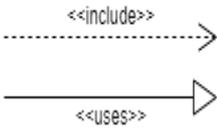
### **2.3.2 *Use Case Diagram***

Menurut A.S & Shalahuddin, *Use case* dideskripsikan sebagai sebuah interaksi antara satu aktor atau lebih dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada pada suatu sistem informasi serta siapa yang berhak untuk dapat mengakses fungsi tersebut [14]. *Use case diagram* memiliki fungsi untuk mempermudah perancangan dalam menentukan fitur dan fungsi aplikasi dari sudut pandang *user* (pengguna). *Use case* memodelkan perilaku sistem informasi yang akan dibuat. *Use case* bekerja dengan mendeskripsikan interaksi khas antara pengguna suatu sistem dan sistem itu sendiri melalui cerita tentang bagaimana sistem tersebut digunakan. [15].

Berikut adalah simbol-simbol yang ada pada *use case* diagram dapat dilihat pada Tabel 2.2 berikut ini :

**Tabel 2. 2 Simbol-simbol *Use Case* Diagram**

Simbol	Nama	Deskripsi
	<i>Use Case</i>	Fungsionalitas yang disediakan oleh sistem ketika unit bertukar pesan antar unit atau aktor sering diungkapkan dengan menggunakan kata kerja di awal kalimat nama use case.
	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi akan tercipta di luar sistem informasi yang akan dibuat.
	Asosiasi	penghubung antar aktor dan <i>use case</i> yang terhubung pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor

	<p>Ekstensi</p>	<p>Hubungan use case tambahan dengan use case yang mana use case tersebut ditambahkan dapat bersifat independen bahkan jika tidak ada use case tambahan. mirip dengan prinsip pewarisan dalam pemrograman berorientasi objek.</p>
	<p>Generalisasi</p>	<p>Hubungan generalisasi dan spesialisasi antara dua kasus penggunaan di mana satu fungsi lebih umum dibandingkan fungsi lainnya.</p>
	<p>Include / Uses</p>	<p>Hubungan antara use case dimana use case yang ditambahkan tersebut mengharuskan use case tersebut untuk menjalankan fungsinya atau sebagai syarat agar use case tersebut dapat dieksekusi disana.</p>

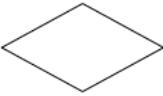
Sumber : (<https://www.dicoding.com/blog/contoh-use-case-diagram/>)

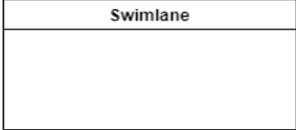
### 2.3.3 Activity Diagram

*Activity diagram* memiliki fungsi sebagai penggambar Langkah-langkah dinamis dari suatu sistem melalui suatu *flow control* antara aksi pada sistem. Pressman dalam M. Hanafi mengatakan bahwa diagram ini memiliki kemiripan dengan *flowchart*, namun yang membedakan adalah diagram aktivitas dapat mewakili proses-proses yang terjadi secara bersamaan. [3].

Berikut merupakan simbol-simbol yang terdapat pada *activity diagram* dapat dilihat pada Tabel 2.3 berikut ini :

**Tabel 2. 3 Simbol-simbol Activity Diagram**

Simbol	Nama	Deskripsi
	Status awal	Awal mula dimulai aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Proses aktivitas yang dilakukan sistem, aktivitas biasanya dimulai dengan kata kerja.
	Percabangan	Percabangan merupakan suatu simbol yang berguna jika ada pilihan aktivitas lebih dari satu.

	Penggabungan	Penggabungan merupakan symbol yang berfungsi untuk menyatukan lebih dari satu aktivitas menjadi satu.
	Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	<i>Swimlane</i>	Simbol ini berguna untuk memisahkan organisasi bisnis yang memiliki tanggung jawab terhadap aktivitas yang terjadi.

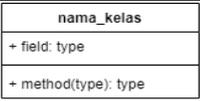
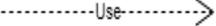
Sumber : (<https://www.dicoding.com/blog/apa-itu-activity-diagram/>)

### 2.3.4 Class Diagram

Diagram kelas (*class diagram*) merupakan gambaran struktur suatu sistem dalam hal mengidentifikasi kelas-kelas yang akan dibuat untuk membangun sistem tersebut. Diagram kelas terdiri dari properti dan operasi dengan tujuan agar pemrogram dapat membuat hubungan antara dokumen desain dan perangkat lunak yang sesuai.[15].

Berikut adalah simbol-simbol yang ada pada *class* diagram dapat dilihat pada Tabel 2.4 berikut ini :

**Tabel 2. 4 Simbol-simbol *Class Diagram***

Simbol	Nama	Deskripsi
	Kelas	Kelas yang terdapat pada struktur pada sistem
	Antarmuka	Tampilan sistem yang mirip konsepnya dengan antarmuka pada pemrograman berorientasi objek
	Asosiasi	Hubungan antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
	Asosiasi berarah	Penghubung antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
	Generalisasi	Penghubung antar kelas dengan makna umum sampai khusus
	Kebergantungan	Penghubung antarkelas dengan makna keterkaitan pada suatu antarkelas
	Agregasi	Relasi antarkelas dengan makna pada semua bagian ( <i>whole-part</i> )

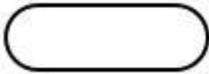
Sumber : (<https://accurate.id/teknologi/class-diagram-adalah/>)

### 2.3.5 State Machine Diagram

*State Machine Diagram* adalah diagram yang menggambarkan perubahan dan transisi suatu objek dari satu keadaan ke keadaan lain dalam suatu program yang diterima oleh suatu proses. Tujuan diagram ini adalah untuk memodelkan perilaku dinamis kasus penggunaan antarmuka pengguna (UI) dan kontrol sistem. [16].

Berikut adalah simbol-simbol yang ada pada *state machine diagram* dapat dilihat pada Tabel 2.5 berikut ini :

**Tabel 2. 5 Simbol-simbol State Machine Diagram**

Simbol	Nama	Deskripsi
	<i>State</i>	Memiliki bentuk segi empat sudut bulat dan memiliki nama sesuai situasinya.
	Titik awal	Menandakan awal dari alur kejadian pada diagram.
	Titik akhir	Menandakan akhir dari alur kejadian pada diagram.
[Guard]	<i>Guard</i>	Merupakan syarat transisi pada system yang berkaitan.
	Point	Menunjukkan apakah masuk kedalam status atau keluar dari status.
<i>event</i>	<i>event</i>	Menunjukkan kondisi yang menyebabkan sesuatu pada status.

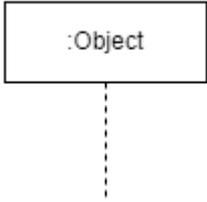
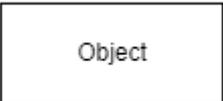
Sumber : (<https://www.ansoriweb.com/2020/04/pengertian-statemachine-diagram>)

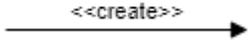
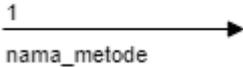
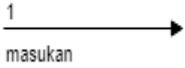
### 2.3.6 *Sequence Diagram*

Diagram ini difungsikan sebagai penunjuk komunikasi yang dinamis antara objek pada saat menjalankan tugas. Pressman dalam M. Hanafi menyebutkan bahwa *sequence diagram* bisa diartikan juga untuk menampilkan hubungan interaksi dari suatu use case atau suatu rancangan perangkat lunak [3].

Berikut merupakan simbol-simbol yang terdapat pada pada *sequence* diagram dapat dilihat pada Tabel 2.6 berikut ini :

**Tabel 2. 6 Simbol-simbol *Sequence Diagram***

Simbol	Nama	Deskripsi
	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem dibuat di luar sistem informasi yang akan dibuat itu sendiri.
	Garis hidup	Menandakan alur kehidupan suatu objek
	Objek	Memiliki artian objek yang menyampaikan pesan

	Waktu aktif	Menyatakan objek dalam keadaan aktif berinteraksi dengan semua yang terhubung didalamnya.
	Pesan tipe <i>create</i>	Menjelaskan bahwa suatu objek membuat objek yang lain
	Pesan tipe <i>call</i>	Objek memanggil metode yang terdapat dalam objek lain atau diri sendiri
	Pesan tipe <i>send</i>	Menjelaskan bahwa objek mengirimkan informasi kepada objek yang lain
	Pesan tipe <i>return</i>	Memiliki arti bahwa suatu objek yang telah menjalankan suatu operasi atau metode akan menghasilkan suatu kembalian ke objek tertentu
	Pesan tipe <i>destroy</i>	Menjelaskan bahwa suatu objek memutus aliran hidup dari objek yang lain

Sumber : (<https://revou.co/kosakata/sequence-diagram>)

## 2.4 *Tools* Pendukung

### 2.4.1 *Unity* 3D

John Riccitello dalam M. Hanafi mengatakan bahwa *Unity* memiliki misi yaitu “*democratize game development*”, maksud dari kata-kata itu adalah *Unity* akan menciptakan alat pengembangan yang mudah digunakan, memberikan kualitas game 3D yang bagus, dan dapat berjalan di banyak *platform* berbeda [3]. Helgason yang merupakan *Co-founder* dan *CEO Unity* tahun 2013 juga menyebutkan dalam M. Hanafi bahwa *Unity* adalah toolkit yang dapat digunakan untuk membuat game dengan berbagai macam teknologi, termasuk teknologi grafis, suara, fisika, interaksi, dan jaringan [3]. Melalui beberapa penjelasan diatas *unity* memiliki makna sebuah *software engine* yang bisa di gunakan untuk mengembangkan sebuah *game* dengan mudah.

Selain itu *Unity* juga membawa kemajuan baru yaitu *Unity* tidak hanya dapat digunakan untuk membuat game saja, tetapi juga dapat digunakan sebagai alat pengembangan perangkat lunak 3D atau 2D yang interaktif, seperti pelatihan simulasi kedokteran, visualisasi arsitektur, mobile, desktop, web, konsol, dan aplikasi lain berdasarkan banyak platform lain. Dengan dukungan *Vuforia Qualcomm*, *Unity* juga dapat digunakan sebagai alat untuk membuat aplikasi berbasis augmented reality. *Vuforia* secara berkala merilis berbagai *extension* yang dapat digunakan sebagai *tools* untuk mengembangkan aplikasi berbasis AR di *Unity*, antara lain *vuforia-unity android-ios-3-0-9 unitypackage* sebagai *open tool* yang banyak digunakan untuk mengembangkan aplikasi berbasis AR di Android dengan *Unity*.

Unity memiliki kerangka kerja untuk mengembangkan berbagai teknologi bisnis. Sistem alat ini menggunakan beberapa bahasa pemrograman, termasuk C#, JavaScript dan Boo.

Adapun beberapa fitur terdapat pada Unity diantaranya adalah sebagai berikut :

1. *Scripting*

pada fitur ini para programmer dapat dengan mudah untuk melakukan pengkodean pada aplikasi atau game yang hendak dibuat.

2. *Animation*

Pada fitur animation yaitu merupakan fitur dimana para developer dapat dengan mudah dan dengan langsung menanamkan animasi pada aplikasi atau game yang dibuat pada *software unity* secara langsung.

Tampilan *Home Unity 3D* pada Gambar 2.4.



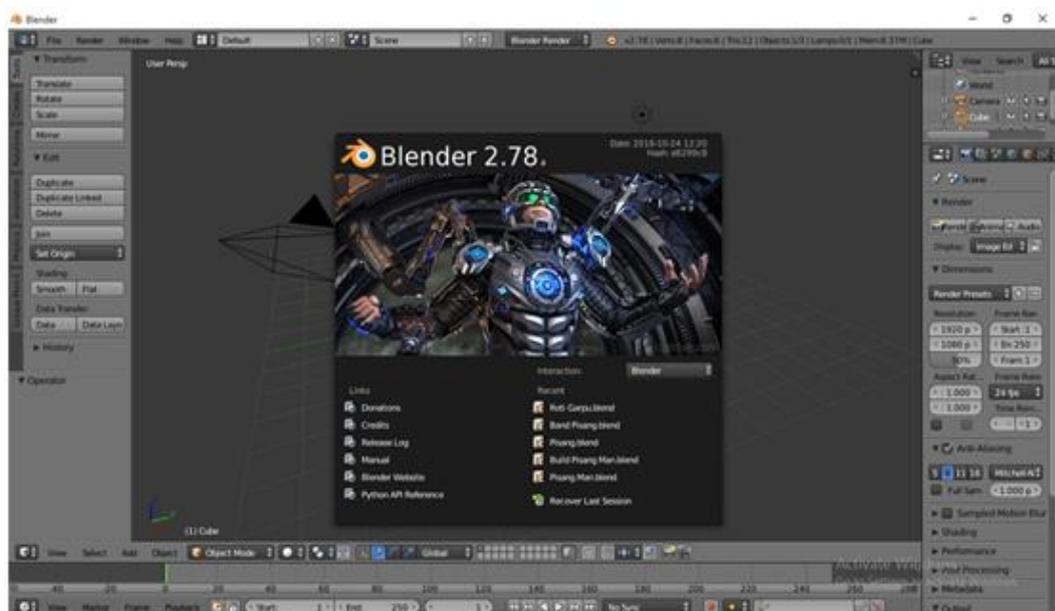
Sumber : (<https://belajar-unity3d.blogspot.com/2015/06/contoh-game-dari-unity3d.html>)

**Gambar 2. 4 Tampilan *Home Unity 3D***

## 2.4.2 Blender

Blender adalah alat pembuatan 3D sumber terbuka dan gratis. *Blender* mendukung semua alur kerja 3D seperti pemodelan, tali-temali, animasi, simulasi, rendering, pengomposisian dan pelacakan gerak, bahkan pengeditan video dan pembuatan game. *Blender* sangat bagus untuk individu dan studio kecil, sangat berguna dalam proyek 3D [17]. Ditujukan untuk profesional media dan seniman, aplikasi *Blender* dapat digunakan untuk membuat gambar 3D, gambar diam serta video berkualitas siaran dan bioskop, sementara menggabungkan mesin 3D real-time memungkinkan pembuatan konten 3D interaktif secara otomatis. Blender memiliki banyak kegunaan, antara lain pemodelan, animasi, *rendering*, tekstur, *layering*, pembentukan, pembobotan, pengeditan non-linier, pembuatan skrip, pengomposisian, pasca produksi, dan lainnya.

Berikut tampilan *Home Software Blender* pada Gambar 2.5.



Sumber : (<https://steemit.com/utopian-io/>)

**Gambar 2.5** Tampilan *Home Software Blender*

### 2.4.3 *Qualcomm Vuforia*

*Qualcomm* adalah *platform* perangkat lunak untuk mengembangkan aplikasi berbasis *augmented reality* (AR). *Platform AR Qualcomm*, termasuk *Vuforia SDK*, menggunakan teknologi *computer vision* untuk memproses gambar grafik dan memodifikasinya dengan objek lain yang muncul di dunia nyata. Dari uraian tersebut dapat disimpulkan bahwa dengan *Qualcomm*, pengembang aplikasi dapat dengan mudah membuat perangkat lunak interaktif berbasis *augmented reality*. Hal ini didukung dengan pendapat Jay Wright dalam M. Hanafi, CEO dari *Qualcomm* yang menjelaskan bahwa *Qualcomm* dapat memudahkan pengembang aplikasi atau game untuk membuat konten 3D interaktif yang diimplementasikan di lingkungan nyata [3].

### 2.4.4 *Visual studio code*

Visual Studio Code adalah editor teks kode yang memiliki ukuran ringan namun kuat yang berjalan di desktop dan tersedia untuk *Windows*, *macOS*, dan *Linux*. Muncul dengan dukungan bawaan untuk *JavaScript*, *TypeScript*, dan *Node.js*, dan memiliki ekosistem ekstensi yang kaya untuk bahasa lain (seperti C++, C#, Java, Python, PHP, Go) dan lingkungan runtime (seperti .NET dan *Unity*) [18] yang tentunya sangat bermanfaat bagi developer dikarenakan banyaknya platform yang didukung didalamnya dan beragam bahasa pemrograman.

## 2.5 Metodologi Penelitian

Penelitian dan pengembangan aplikasi ini menggunakan metode penelitian *Research and Development (R&D)*. Metode Penelitian dan Pengembangan merupakan suatu metode penelitian yang bertujuan untuk menghasilkan produk

tertentu dan menguji keefektifannya seperti yang dikatakan Sudaryono dalam M. Hanafi[3]. Produk yang dihasilkan adalah aplikasi Bangun Ruang yang merupakan aplikasi media pembelajaran geometri berbasis Android dengan menggunakan teknologi *augmented reality*. Sasaran pengguna aplikasi ini adalah siswa kelas 6 SD. Untuk mendapatkan produk yang tepat, pengembangan perangkat lunak harus didasarkan pada model metode pengembangan ini.

### **2.5.1 Penelitian Terdahulu**

Sebelum melakukan penelitian, perlu adanya hasil-hasil penelitian yang relevan dan telah dilakukan sebelumnya, khususnya yang berkaitan dengan pengembangan media dengan menggunakan teknologi *augmented reality*.

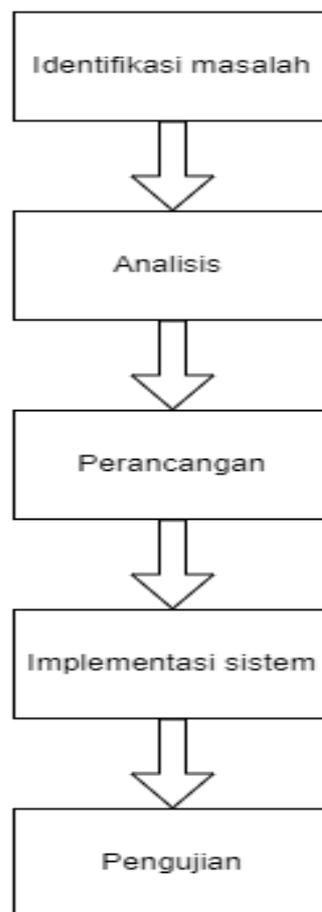
Dalam penelitian yang dilakukan Mifta Hanafi Rizqi, yang berjudul “analisis dan perancangan aplikasi geometra, media pembelajaran geometri mata pelajaran matematika berbasis android menggunakan teknologi augmented reality” Analisis, komunikasi, dan kolaborasi yang cermat pada awal pembuatan media simulasi merupakan faktor kunci validitas konten media yang dihasilkan. Namun kekurangan dari penelitian ini adalah tidak adanya fungsi kuis pada aplikasi yang dibuat serta banyak kendala seperti kamera menunjuk ke *image target* tetapi tidak muncul.

Dari referensi penelitian sebelumnya, diharapkan pengembang dapat mengembangkan aplikasi berbasis AR yang lebih baik, khususnya aplikasi *mobile* yang sangat interaktif dan terlihat bagus pada halaman menu dan tampilan *user interface* lainnya.

## 2.5.2 Kerangka Kerja Penelitian

Produk dikembangkan dengan metode air terjun (*waterfall*). Metode *waterfall* memiliki model pengembangan berurutan untuk menyelesaikan proses pengembangan perangkat lunak.

Berikut alur dari kerangka kerja penelitian pada Gambar 2.6.



**Gambar 2. 6 Kerangka Kerja Penelitian**