

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Berikut beberapa pembandingan dari penelitian sebelumnya yang memiliki konsep penerapan *augmented reality* sebagai media pembelajaran sistem pencernaan, yaitu:

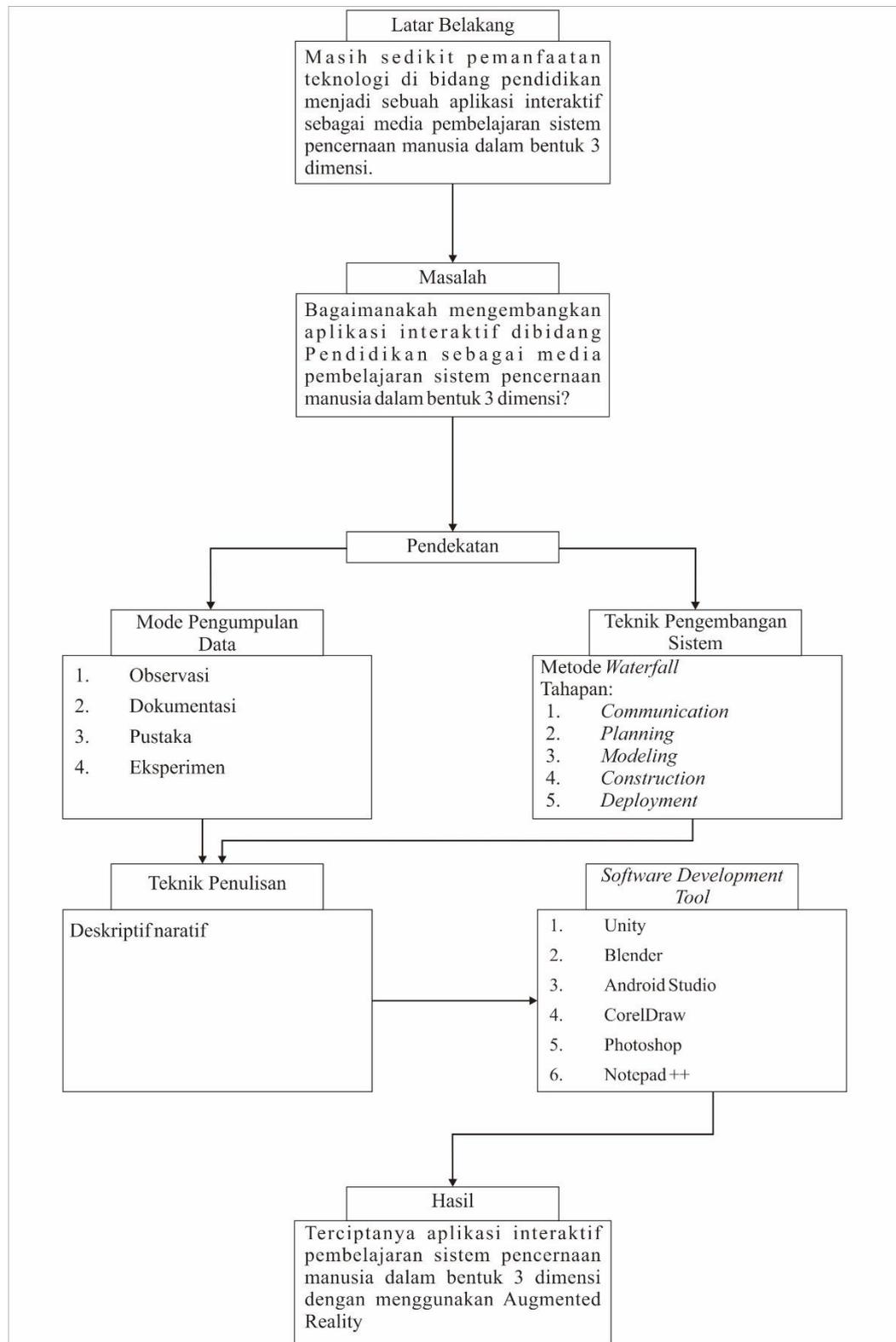
Berdasarkan judul penelitian “Penerapan *Augmented Reality* Sebagai Media Pembelajaran Sistem Pencernaan Pada Manusia Dalam Mata Pelajaran Biologi” yang termuat dalam jurnal edukasi dan penelitian informatika (jepin) vol. 3, no. 2 tahun 2017 dan disusun oleh Rizqi Mauludin, Anggi Srimurdianti Sukamto, Hafiz Muhandi. Aplikasi AR Pencernaan yang dibangun merupakan penerapan dari teknologi *Augmented Reality* berbasis Android. Dibangunnya aplikasi AR Pencernaan bertujuan sebagai media pembelajaran multimedia sistem pencernaan pada manusia. Menjelaskan bahwa hasil penelitian berbentuk aplikasi yang terdiri atas menu utama sebagai tampilan utama, berisikan navigasi untuk berpindah scene. Pada menu utama terdapat beberapa tombol yang yaitu, tombol AR pencernaan yang akan mengarahkan ke scene AR pencernaan. Pada scene AR objek yang ditampilkan adalah objek 3D keseluruhan dari sistem pencernaan, tombol kuis yang akan mengarahkan ke scene menu kuis, tombol petunjuk yang akan menampilkan pop up informasi cara menggunakan aplikasi, tombol tentang yang akan menampilkan pop up informasi mengenai aplikasi, dan tombol keluar yang akan menampilkan pop up pilihan untuk keluar dari aplikasi atau tidak. Ketika marker terdeteksi dan objek 3D tampil pada layar android, maka tombol untuk menjalankan animasi serta informasi dari objek pencernaan akan aktif dan tampil bersamaan dengan tampilnya objek 3D.

Berdasarkan judul penelitian “Penerapan Teknologi *Augmented Reality* Pada Media Pembelajaran Sistem Pencernaan Berbasis Android” yang termuat dalam jurnal teknik komputer amik bsi vol. ii no. 1 februari 2016 yang disusun oleh Feby Zulham Adami, dan Cahyani Budihartanti.

Menjelaskan bahwa hasil penelitian yang berbentuk aplikasi terdapat berbagai scene yang memiliki fungsi tersendiri, sebagai berikut: Halaman Splash Screen merupakan tampilan awal pada saat aplikasi dijalankan, setelah tampilan splash Screen selesai tampilan aplikasi akan berpindah ke tampilan menu utama. Halaman Menu Utama, terdapat 5 buah button yang digunakan untuk menuju scene masing-masing halaman AR pencernaan dimana objek 3D sistem pencernaan akan tampil, berdasarkan jurnal di informasikan bahwa model objek 3D ditampilkan dalam bentuk keseluruhan tubuh organ pencernaan manusia. Halaman Materi, berisikan informasi mengenai materi sistem pencernaan pada manusia. Halaman Cara Penggunaan, menginformasikan tata cara penggunaan aplikasi untuk mempermudah pengguna. Halaman About Aplikasi, mengenai semua hal yang berhubungan dengan aplikasi dari informasi pembuatan hingga pendukung sistem dalam implementasi.

Berdasarkan judul penelitian “Pengembangan Media Pembelajaran Sistem Pencernaan Manusia Untuk Kelas 8 Smp Dengan Fitur *Augmented Reality* Berbasis Android (Studi Kasus: Smpn 7 Depok)” yang termuat dalam jurnal pinter vol. 1 no 1 juni 2017 serta disusun oleh Juannita, dan Bambang Prasetya Adhi. Menjelaskan bahwa hasil penelitian ini adalah sebuah media pembelajaran sistem pencernaan manusia untuk kelas 8 SMP dengan fitur *Augmented Reality* berbasis Android. Dalam aplikasi berisi materi sistem pencernaan manusia yang meliputi materi organ pencernaan, enzim pencernaan, dan penyakit yang berhubungan dengan sistem pencernaan. Terdapat fitur yang menampilkan objek 3D organ pencernaan dan video cara kerja sistem pencernaan. Penggunaan aplikasi AR dalam pembelajaran bertujuan membuat siswa lebih aktif sehingga menarik minat siswa untuk mempelajari materi sistem pencernaan manusia.

2.2 Kerangka Pemikiran



Gambar 2.1. Kerangka Pemikiran

Keterangan:

1. Latar Belakang Masalah

Pokok permasalahan yang mendasari perlunya dibangun suatu aplikasi interaktif sebagai media pembelajaran sistem pencernaan manusia dalam bentuk 3 dimensi.

2. Perumusan Masalah

Perumusan masalah adalah rumusan dari hasil simpulan masalah dan solusi untuk menyelesaikan permasalahan yang didapatkan berdasarkan pokok permasalahan yang dibahas pada latar belakang masalah sebelumnya.

3. Pengumpulan Data

Mengumpulkan semua data yang dibutuhkan, baik melalui observasi, dokumentasi, literatur dan eksperimen.

a. Observasi

Mengamati beberapa aplikasi interaktif pembelajaran sistem pencernaan manusia 3 dimensi yang sudah ada, baik dari karya ilmiah, buku, atau internet yang bisa dijadikan referensi

b. Dokumentasi

Pengumpulan data yang berhubungan dalam pembuatan aplikasi pembelajaran sistem pencernaan manusia.

c. Pustaka

Pengumpulan data yang diarahkan kepada pencarian data dan informasi melalui dokumen-dokumen.

d. Eksperimen

Penelitian yang dilakukan secara sengaja oleh peneliti dengan cara memberikan perlakuan tertentu terhadap subjek penelitian.

4. Pengembangan Sistem

Perencanaan sistem apa yang akan dibangun, menganalisis dan merancang aplikasi interaktif pembelajaran sistem pencernaan manusia, menerapkan atau mengimplementasikan dalam bentuk aplikasi kemudian penggunaan aplikasi secara umum.

5. *Software Development Tools*

Penggunaan *software* yang mendukung dalam implementasi pengembangan aplikasi interaktif media pembelajaran sistem penceraaan manusia dalam bentuk 3 dimensi.

6. Hasil

Implementasi berhasil menciptakan aplikasi interaktif sebagai media pembelajaran sistem penceraaan manusia dalam bentuk 3 dimensi, kemudian dilanjutkan pengujian sistem sebelum di publikasikan.

a. Pengujian Sistem

Pengujian sistem dilakukan bertujuan untuk mengetahui kelayakan sistem pelayanan wisuda apakah sudah memenuhi syarat layak guna atau belum.

b. Publikasi

Pada tahap akhir ini, aplikasi interaktif pembelajaran sistem penceraaan manusia dalam bentuk 3 dimensi siap dipublikasikan dan dibuat dokumentasi dari keseluruhan kegiatan penyusunan Tugas Akhir.

2.3 Media Pembelajaran Interaktif

Menurut Akhmad Sudrajat: 2008 Media Pembelajaran berasal dari dua buah kata yaitu Media dan Pembelajaran. Istilah media berasal dari bahasa latin yang merupakan bentuk jamak dari kata “Medium” yang secara harfiah berarti “Perantara” atau “Pengantar” yaitu perantara atau pengantar sumber pesan dengan penerima pesan. Sedangkan dalam bahasa Arab, media disebut “Wasil” bentuk jamak dari “wasilah” yakni sinonim “Al-wats” yang artinya “Tengah”. Kata “interaktif” secara umum memiliki arti komunikasi dua arah atau lebih dari komponen-komponen komunikasi. Lebih simpelnya, “interaktif” berarti komunikasi aktif antara komunikator dan komunikan. Tidak ada satu pihak yang pasif.

Menurut Yudhi Munadi: 2008 Media Interaktif secara umum mengacu pada pada produk multimedia dan layanan digital pada system IT yang merespon tindakan pengguna dengan menyajikan konten audio, konten visual maupun konten audiovisual. Sehingga dapat disimpulkan bahwa media merupakan pengantar atau

penghubung, yakni yang mengantarkan atau menghubungkan dan menyalurkan sesuatu hal dari satu sisi ke sisi yang lainnya.

Menurut Hujair AH Sanaky: 2013 pengertian Media Pembelajaran Interaktif adalah alat bantu berbasis multimedia yang dapat menjabarkan pesan atau informasi dari guru ke siswa yang dalam prosesnya terjadi komunikasi aktif dua arah antara multimedia dengan pengguna (siswa) yang bertujuan mempermudah proses pembelajaran.

2.4 Proses Sistem Pencernaan

Berdasarkan buku berjudul Ilmu Pengetahuan Alam yang diterbitkan oleh Pusat Kurikulum dan Perbukuan, Balitbang, Kemendikbud untuk SMP/MTs Kelas VIII semester 1. Pada saat makanan masuk ke dalam mulut, proses pencernaan dimulai. Pencernaan merupakan proses memecah makan menjadi molekul kecil sehingga dapat diserap oleh tubuh melalui pembuluh darah. Selanjutnya, molekul makanan dari darah masuk ke dalam sel melintasi membran sel. Molekul yang tidak digunakan dan dibutuhkan oleh tubuh akan dikeluarkan dari tubuh melalui sistem ekskresi seperti keringat dan urine. Makanan yang tidak tercerna berupa feses akan dibuang melalui anus, proses ini disebut defeksasi.

2.5 *Augmented Reality*

Menurut Andre Kurniawan Pamoedji: 2017 *Augmented Reality* atau dalam Bahasa Indonesia diterjemahkan menjadi Realitas Tambahan adalah sebuah teknik yang menggabungkan benda maya dua dimensi maupun tiga dimensi ke dalam sebuah lingkup nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata.

Menurut Andre Kurniawan Pamoedji: 2017 *Augmented Reality* atau yang sering disingkat dengan AR ini berbeda dengan *Virtual Reality* yang kerap disebut VR. *Augmented Reality* tidaklah seperti *Virtual Reality* yang sepenuhnya menggantikan kenyataan, *Augmented Reality* hanya sekedar menambahkan atau melengkapi kenyataan.

2.6 Unity

Menurut Roedavan R: 2016 Unity merupakan suatu aplikasi yang digunakan untuk mengembangkan game multi *platform* yang didesain untuk mudah digunakan. Unity itu bagus dan penuh perpaduan dengan aplikasi yang profesional. Editor pada Unity dibuat dengan *user interface* yang sederhana. Editor ini dibuat setelah ribuan jam dihabiskan untuk membuatnya menjadi nomor satu dalam urutan ranking teratas untuk editor game. Grafis pada unity dibuat dengan grafis tingkat tinggi untuk OpenGL dan DirectX. Unity mendukung semua format file, terutamanya format umum seperti semua format dari *art applications*. Unity cocok dengan versi 64-bit dan dapat beroperasi pada Mac OS x dan windows dan dapat menghasilkan Game untuk Mac, Windows, Wii, iPhone, iPad dan Android. Unity secara rinci dapat digunakan untuk membuat video game 3D, real time animasi 3D dan visualisasi arsitektur dan isi serupa yang interaktif lainnya

Fitur-fitur:

1. *Rendering*

Unity dapat mengambil format desain dari 3ds Max, Maya, Softimage, Blender, modo, ZBrush, Cinema 4D, Cheetah3D, Adobe Photoshop, Adobe Fireworks and Allegorithmic Substance. Asset tersebut dapat ditambahkan ke game project dan diatur melalui *graphical user interface* Unity.

2. *ShaderLab*

Bahasa yang digunakan untuk shaders, dimana mampu memberikan deklaratif “*programming*” dari *fixed-function pipeline* dan program *shader* ditulis dalam GLSL atau Cg. Sebuah shader dapat menyertakan banyak varian dan sebuah spesifikasi *fallback declarative*, dimana membuat Unity dapat mendeteksi berbagai macam video card terbaik saat ini, dan jika tidak ada yang kompatibel, maka akan dilempar menggunakan shader alternatif yang mungkin dapat menurunkan fitur dan performa.

3. *Scripting*

Script Game engine dibuat dengan Mono 2.6, sebuah implementasi *open-source* dari .NET *Framework*. Programmer dapat menggunakan UnityScript (bahasa terkustomisasi yang terinspirasi dari syntax ECMAScript, dalam bentuk

JavaScript), C#, atau Boo (terinspirasi dari sintax bahasa pemrograman python). Dimulai dengan dirilisnya versi 3.0, Unity menyertakan versi MonoDevelop yang terkustomisasi untuk debug script.

4. *Asset Tracking*

Unity juga menyertakan Server Unity Asset. Sebuah solusi terkontrol untuk *developer* Game asset dan script. Server tersebut menggunakan PostgreSQL sebagai *backend*, sistem audio dibuat menggunakan FMOD *library* (dengan kemampuan untuk memutar Ogg Vorbis compressed audio), video playback menggunakan Theora codec, engine daratan dan vegetasi (yang mendukung tree billboard, Occlusion Culling dengan Umbra), built-in lightmapping dan global illumination dengan Beast, multiplayer *networking* menggunakan RakNet, dan navigasi mesh pencari jalur built-in.

5. *Platforms*

Unity *support* pengembangan ke berbagai *platform*. Di dalam *project*, *developer* memiliki kontrol untuk mengirim perangkat *mobile*, web browser, desktop, and *console*. Unity juga mengijinkan spesifikasi kompresi *textur* dan pengaturan resolusi di setiap *platform* yang didukung. Saat ini *platform* yang didukung adalah BlackBerry 10, Windows 8, Windows Phone 8, Windows, Mac, Linux, Android, iOS, Unity Web Player, Adobe Flash, PlayStation 3, Xbox 360, Wii U and Wii. Meskipun tidak semua terkonfirmasi secara resmi, Unity juga mendukung PlayStation Vita yang dapat dilihat pada Game Escape Plan dan Oddworld: New 'n' Tasty.

6. *Asset Store*

Diluncurkan November 2010, *Unity Asset Store* adalah sebuah *resource* yang hadir di Unity editor. *Asset store* terdiri dari koleksi lebih dari 4,400 asset *packages*, beserta 3D models, *textures* dan *materials*, sistem particle, musik dan efek suara, tutorial dan *project*, *scripting package*, editor *extensions* dan servis online.

7. *Physics*

Unity juga memiliki suport built-in untuk PhysX *physics engine* (sejak Unity 3.0) dari Nvidia (sebelumnya Ageia) dengan penambahan kemampuan untuk

simulasi *real-time cloth* pada *arbitrary* dan *skinned meshes*, *thick ray cast*, dan *collision layers*.

2.7 Smartphone

Menurut Williams & Sawyer: 2011:385 yang mengutip dari Steve Lohr penulis New York Times Technology, *smartphone* adalah teknologi terdepan yang memungkinkan menambah alat berkualitas tinggi ke dalam telepon. *Smartphone* adalah telepon dengan prosesor berukuran mikro, memori, tampilan layar, dan alat yang telah terpasang untuk terhubung dengan jaringan.

2.8 Android Studio

Menurut Alfa Satyaputra & Eva Maulina Aritonang: 2015 Android Studio adalah perangkat lunak baru yang dikembangkan oleh Google untuk menghasilkan aplikasi android. Perangkat lunak ini memiliki banyak fitur yang lebih besar tentang google map terkait kegiatan dari *ECLIPSE*. Bahasa Java merupakan dasar *platform* untuk pembuatan aplikasi. Aplikasi ini hanya dibuat untuk perangkat Android.

2.9 Android Software Development Kit (SDK)

Menurut Burnette: 2010:21 *Android Software Development Kit* (SDK) merupakan sebuah kumpulan aplikasi pengembangan yang komprehensif yang meliputi *debugger*, *libraries* dan emulator berbasis QEMU, dokumentasi, tutorial dan contoh kode. Android SDK terbagi dalam dua bagian yaitu Android SDK Starter dan Android SDK Components. Android SDK Starter merupakan aplikasi utama sedangkan Android SDK *Components* berisi tools yang akan digunakan nantinya.

2.10 Blender 3D

Menurut Hendi Hendratman: 2015 *Software* Blender adalah *software* gratis multifungsi: *3D Modelling*, Animasi, Video Edit, *Compositing* bahkan Game! Blender digunakan animator profesional, desainer grafis, arsitek, desainer interior, desainer produk, video editor / *effects*, game programmer dll. Popularitasnya di indonesia sudah mengalahkan seniornya 3D Studio Max dan Maya.

2.11 CorelDraw

Menurut Febian: 2009 CorelDraw merupakan salah satu aplikasi pengolahan gambar berbasis vector yang banyak dipakai oleh pengguna PC. Karena berbagai kemudahan dan keunggulan yang dimiliki oleh coreldraw, maka coreldraw sering dimanfaatkan untuk desktop publishing, percetakan, dan bidang lain yang memerlukan pemrosesan visual. Keunggulan mengolah gambar berbasis vektor adalah ukuran hasil akhir yang dapat ditekan seminimal mungkin namun dengan kualitas yang tidak kalah dengan gambar berbasis raster atau bitmap. Karena itulah desain grafis dan olah gambar berbasis vektor sangat banyak digunakan untuk desktop publishing, percetakan, dan bidang lain yang memerlukan pemrosesan visual.

2.12 Adobe Photoshop

Menurut Musa Almasih: 2014 Adobe Photoshop, atau biasa disebut Photoshop, adalah perangkat lunak editor citra buatan *Adobe Systems* yang dikhususkan untuk pengeditan foto atau gambar dan pembuatan efek. Perangkat lunak ini banyak digunakan oleh fotografer digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (*market leader*) untuk perangkat lunak pengolahan gambar/foto, dan, bersama Adobe Acrobat, dianggap sebagai produk terbaik yang pernah diproduksi oleh Adobe Systems. Versi kedelapan aplikasi ini disebut dengan nama Photoshop CS (*Creative Suite*).

2.13 Analisis dan Perancangan Sistem

Analisis sistem didefinisikan bagaimana memahami dan menspesifikasikan dengan detail apa yang harus dilakukan oleh sistem. Sementara sistem desain diartikan sebagai menjelaskan dengan detail bagaimana bagian-bagian dari sistem informasi diimplementasikan.

2.13.1 Pengertian UML (*Unified Modeling Language*)

Menurut Nugroho: 2010:6 mengatakan *Unified Modeling Language* (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan sesungguhnya digunakan untuk penyederhanaan

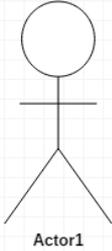
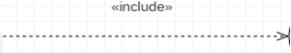
permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. Tujuan pemodelan adalah sebagai sarana analisis pemahaman, visualiasi, dan komunikasi antar anggota tim pengembang, serta sebagai sarana dokumentasi. UML dirancang untuk menyediakan sekumpulan notasi yang dapat dipakai untuk mendokumentasi desain berorientasi objek. UML mendefinisikan beberapa diagram diantaranya *Use case diagram*, *Class Diagram*, dan *Activity Diagram*.

Menurut Windu Gata & Grace: 2013:4 *Unified Modeling Language (UML)* adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

2.13.2 Use Case Diagram

Use case diagram merupakan pemodelan untuk kelakuakn (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case diagram* yaitu:

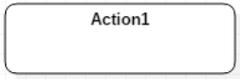
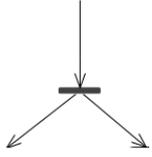
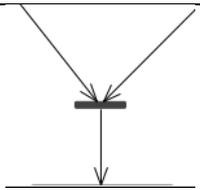
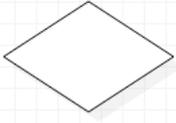
Tabel 2.1 Use Case Diagram

Gambar	Keterangan
	<p><i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja</p>
	<p><i>Actor</i> atau Aktor adalah <i>Abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i>, tetapi tidak memiliki kontrol terhadap <i>use case</i></p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.</p>
	<p>Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem</p>
	<p><i>Include</i>, merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan use case oleh use case lain, contohnya adalah pemanggilan sebuah fungsi program</p>
	<p><i>Extend</i>, merupakan perluasan dari use case lain jika kondisi atau syarat terpenuhi</p>

2.13.3 Activity Diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* yaitu:

Tabel 2.2 *Activity Diagram*

Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas
	<i>End Point</i> , akhir aktivitas
	<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis
	<i>Fork</i> /percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi
	<i>Decision Points</i> , menggambar kan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i>
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa

2.13.4 Class Diagram

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Class Diagram secara khas meliputi: Kelas (*Class*), Relasi *Associations*, *Generalization* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality*.

Tabel 2.3 *Multiplicity Class Diagram*

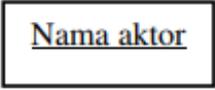
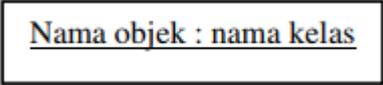
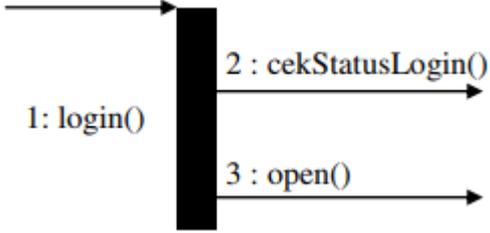
<i>Multiplicity</i>	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4

2.13.5 Sequence Diagram

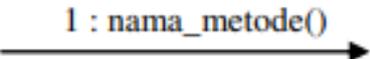
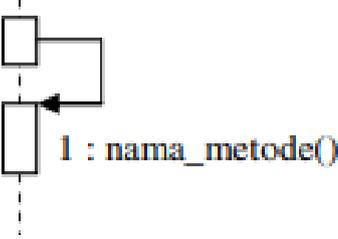
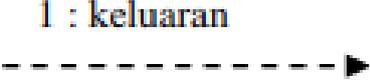
Menurut Rosa & Shalahudin: 2014 *sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan *message* yang dikirimkan dan diterima antar objek. Dalam penggambaran *sequence diagram* harus mengetahui objek yang terlibat dalam sebuah *use case* serta metode yang dimiliki dan melihat skenario yang ada pada *use case*.

Banyaknya *sequence diagram* yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksinya pesan sudah dicakup dalam *sequence diagram* sehingga semakin banyak *use case* yang didefinisikan maka *sequence diagram* yang harus dibuat juga semakin banyak. Berikut adalah simbol-simbol yang ada pada *sequence diagram* dapat dilihat pada tabel 4.2.

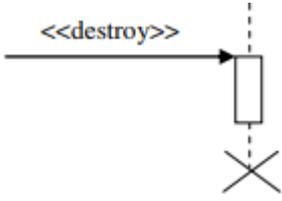
Tabel 2.4 Simbol-simbol *Sequence Diagram*

Gambar	Keterangan
<p>Aktor</p>  <p>Atau</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan dalam menggunakan kata benda diawal frase nama aktor.</p>
<p>Garis hidup/<i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semuanya yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya</p>  <p>Maka <code>cekStatusLogin()</code> dan <code>open()</code> dilakukan didalam metode <code>login()</code>. Aktor tidak memiliki waktu aktif</p>

Lanjutan Tabel 2.4 Simbol-simbol *Sequence Diagram*

Gambar	Keterangan
<p data-bbox="454 443 679 472">Pesan tipe <i>create</i></p> 	<p data-bbox="842 443 1353 584">Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p data-bbox="472 687 662 716">Pesan tipe <i>call</i></p> 	<p data-bbox="842 687 1353 828">Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,</p>  <p data-bbox="842 1122 1353 1429">Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p data-bbox="464 1456 670 1485">Pesan tipe <i>send</i></p> 	<p data-bbox="842 1456 1353 1653">Menyatakan bahwa suatu objek mengirimkan data/masukkan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p data-bbox="456 1680 678 1709">Pesan tipe <i>return</i></p> 	<p data-bbox="842 1680 1353 1928">Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>

Lanjutan Tabel 2.4 Simbol-simbol *Sequence Diagram*

Gambar	Keterangan
<p data-bbox="448 443 687 477">Pesan tipe <i>destroy</i></p> 	<p data-bbox="842 443 1358 696">Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada create maka ada destroy</p>

2.13.6 Pengujian Sistem *White Box* dan *Black Box Testing*

Menurut Nidhra & Dondetti: 2012 *White Box Testing* adalah salah satu cara untuk menguji suatu aplikasi atau *software* dengan cara melihat modul untuk dapat meneliti dan menganalisa kode dari program yang di buat ada yang salah atau tidak. Kalau modul yang telah dan sudah di hasilkan berupa output yang tidak sesuai dengan yang di harapkan maka akan dikompilasi ulang dan di cek kembali kode-kode tersebut hingga sesuai dengan yang diharapkan. Kelebihan *White Box Testing* antara lain (Nidhra & Dondetti, 2012):

1. Kesalahan Logika

Menggunakan syntax 'if' dan syntax pengulangan. Langkah selanjutnya metode *white box testing* ini akan mencari dan mendeteksi segala kondisi yang di percaya tidak sesuai dan mencari kapan suatu proses perulangan di akhiri.

2. Ketidaksesuaian Asumsi

Menampilkan dan memonitor beberapa asumsi yang diyakini tidak sesuai dengan yang diharapkan atau yang akan diwujudkan, untuk selanjutnya akan dianalisa kembali dan kemudian diperbaiki.

3. Kesalahan Pengetikan

Mendeteksi dan menaribahasa-bahasa pemograman yang di anggap bersifat *case sensitif*.

Kelemahan *White Box Testing* adalah pada perangkat lunak yang jenisnya besar, metode *White box testing* ini dianggap boros karena melibatkan banyak sumberdaya untuk melakukannya. (Nidhra & Dondetti, 2012)

Sedangkan *Black Box Testing* berfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. *Black Box Testing* lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*. *Black Box Testing* 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.