

BAB II

TINJAUAN PUSTAKA

II.1. Perancangan

Perancangan merupakan spesifikasi umum dan terinci dari pemecahan masalah berbasis komputer yang telah dipilih selama tahap analisis. Perancangan adalah menyeleksi dan menghubungkan pengetahuan, fakta, imajinasi, asumsi untuk masa yang akan datang dengan tujuan memvisualisasi dan memformulasi hasil yang diinginkan, dan juga urutan kegiatan yang diperlukan. Perancangan dalam pengertian ini menitik beratkan kepada usaha untuk menyeleksi dan menghubungkan sesuatu dengan kepentingan masa yang akan datang serta usaha untuk mencapainya. (Al-Bahar 2005 : 51).

II.1.1. Multimedia Interaktif

Pembelajaran yang menggunakan teknologi informasi dan komunikasi atau menggunakan multimedia disebut dengan media pembelajaran berbasis multimedia interaktif. Penggunaan media pembelajaran ini dimaksudkan untuk membantu dosen dalam penyampaian materi dan juga membantu mahasiswa dalam memahami materi yang diajarkan.

Selain itu muatan materi pelajaran dapat dimodifikasi menjadi lebih menarik dan mudah dipahami, tujuan materi yang sulit akan menjadi mudah, suasana belajar yang menegangkan menjadi menyenangkan. Dengan menggunakan media pembelajaran berbasis multimedia dapat memadukan media-

media dalam proses pembelajaran, maka proses pembelajaran akan berkembang dengan baik, sehingga membantu dosen menciptakan pola penyajian yang interaktif. (Dwi Sarwiko;2010).

II.1.2. Animasi

Animasi adalah suatu teknik yang banyak sekali digunakan dalam dunia film dewasa ini, baik sebagai suatu kesatuan yang utuh, bagian dalam suatu film maupun bersatu dalam film live.

Dunia film sebenarnya berakar dari fotografi, sedangkan animasi berakar dari dunia gambar, yaitu ilustrasi design grafis (desain komunikasi visual).

Dapat dikatakan bahwa animasi merupakan suatu media yang lahir dari dua konversi atau disiplin, yaitu film dan gambar. Untuk dapat mengerti dan memakai teknik animasi, dua konversi tersebut harus dipahami dan dimengerti.

Film biasanya dipakai untuk merekam suatu keadaan atau mengemukakan sesuatu. Film digunakan untuk memenuhi suatu kebutuhan umum yaitu mengkomunikasikan suatu gagasan, pesan atau kenyataan. Karena keunikan dimensinya dan karena sifat hiburanannya, film telah diterima sebagai salah satu media audio visual yang paling populer dan paling digemari. Karena itu juga dianggap sebagai media yang paling efektif.

Keinginan manusia untuk membuat gambar atau santiran (*image*) yang hidup dan bergerak sebagai perantara dari pengungkapan (*expression*) mereka, merupakan perwujudan dari bentuk dasar animasi yang hidup dan berkembang. (Yunita Syahfitri;2011).

II.1.3. Jenis-jenis Animasi

Ada beberapa jenis animasi, diantaranya adalah :

1. *Stop Motion* disebut juga *frame by frame*. Teknik animasi ini akan membuat objek seakan bergerak. Objek bisa bergerak karena mempunyai banyak *frame* yang dijalankan secara berurutan.
2. *Cell Animation* merupakan bentuk animasi tertua dan merupakan bentuk animasi yang paling populer.
3. *Time-Lapse*. Setiap *frame* akan di-*capture* dengan kecepatan yang lebih rendah daripada kecepatan ketika *frame* dimainkan.
4. *Claymation* dulunya disebut dengan *Clay Animation* dan merupakan salah satu bentuk dari *stop motion animation*.
5. *Cut-Out Animation*. Teknik ini digunakan untuk memproduksi animasi menggunakan karakter, properti, dan *background* dari potongan material seperti kertas, karton atau foto. Saat ini, *cut-out animation* diproduksi menggunakan komputer dengan gambar dari hasil pemindai atau grafik vektor untuk menggantikan potongan material yang digunakan.
6. *Puppet Animation*. Dalam *puppet animation*, boneka akan menjadi aktor utamanya sehingga animasi jenis ini membutuhkan banyak boneka. Animasi jenis ini dibuat dengan teknik *frame by frame*, yaitu setiap gerakan boneka di-*capture* satu per satu dengan *camera*.

Kata animasi itu sendiri sebenarnya penyesuaian dari kata *animation* yang berasal dari *animation* yang berasal dari kata dasar *to animate* dalam kamus umum Inggris-Indonesia berarti menghidupkan (Wojowasito, 1997). Secara umum animasi adalah suatu kegiatan menghidupkan, menggerakkan benda mati; Suatu benda mati, diberikan dorongan kekuatan, semangat dan emosi untuk menjadi hidup dan bergerak atau hanya berkesan hidup. (Yunita Syahfitri;2011).

II.2. Pengenalan 3Ds Max

3D Studio Max (3Ds Max) adalah sebuah *software* yang dikhususkan dalam pemodelan 3 dimensi ataupun untuk membangun animasi 3 dimensi. Selain terbukti handal untuk digunakan dalam pembuatan objek 3 dimensi, 3Ds Max juga banyak digunakan dalam pembuatan desain *furniture*, konstruksi, maupun desain interior. Selain itu, 3Ds Max juga sering digunakan dalam pembuatan animasi atau film kartun, (Galih Pranowo; 2010: 1).

II.2.1. Area Kerja 3Ds Max10

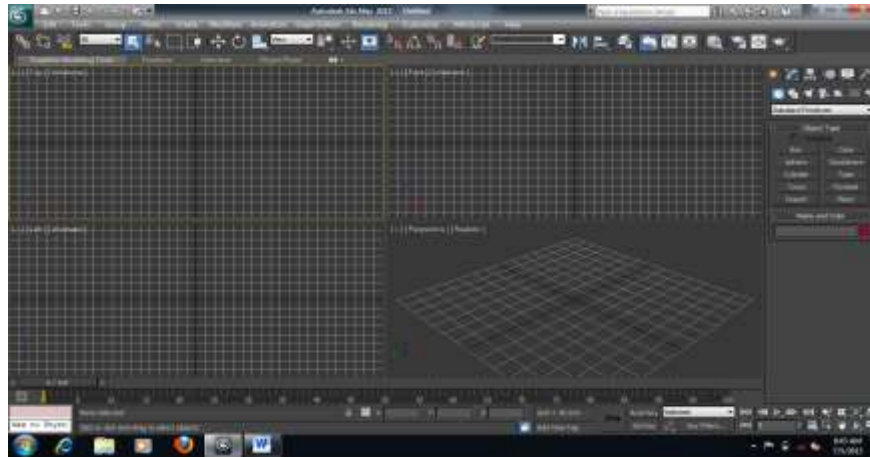
Saat pertama kali mengerjakan program 3D Studia Max12 (3Ds Max12), maka kita akan mendapati tampilan halaman pembuatan 3Ds Max12 seperti yang dilihat pada Gambar II.1.



Gambar II.1. Tampilan Halaman Pembuka 3Ds Max

Sumber : (Galih Pramowo: 2010; 2)

Setelah proses *loading* program 3Ds Max 2010 selesai, maka akan tampil bagian antarmuka dari 3Ds Max 2010. Area kerja 3Ds Max 2010 dapat dilihat pada Gambar II.2.



Gambar II.2. Tampilan Area Kerja 3Ds Max

Sumber : (Galih Pramowo: 2010; 2)

II.3. Pengenalan Adobe Flash CS5

AdobeFlash (dahulu bernama *MacromediaFlash*) adalah salah satu perangkat lunak komputer yang merupakan produk unggulan *AdobeSystems*. *Adobe* digunakan untuk membuat gambar vektor maupun animasi gambar tersebut. Berkas yang dihasilkan dari perangkat lunak ini mempunyai file *extension* .swf dan dapat diputar di penjelajah web yang telah dipasang *AdobeFlashPlayer*. *Flash* menggunakan bahasa pemrograman bernama *ActionScript* yang muncul pertama kalinya pada *Flash 5*. (Kriston Radion, S.ST:2012;2)

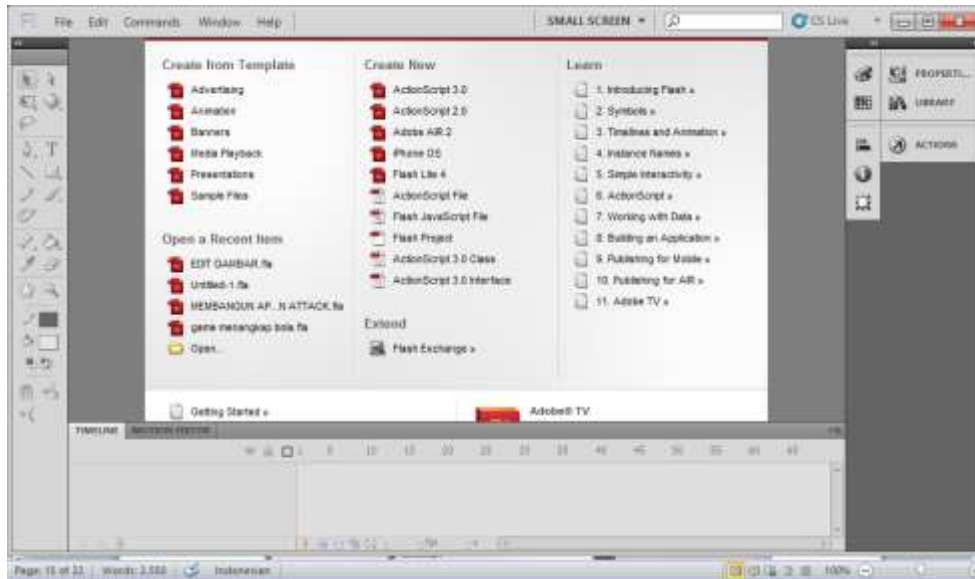
Sebelum tahun 2005, *Flash* dirilis oleh *Macromedia*. *Flash 1.0* diluncurkan pada tahun 1996 setelah *Macromedia* membeli program animasi vektor bernama *FutureSplash*. Versi terakhir yang diluncurkan di pasaran dengan menggunakan nama '*Macromedia*' adalah *MacromediaFlash 8*. Pada tanggal 3 Desember 2005 *AdobeSystems* mengakuisisi *Macromedia* dan seluruh produknya, sehingga nama *MacromediaFlash* berubah menjadi *AdobeFlash*.(Kriston Radion, S.ST:2012;2)

AdobeFlash merupakan sebuah program yang didesain khusus oleh *Adobe* dan program aplikasi standar *authoringtool* profesional yang digunakan untuk membuat animasi dan bitmap yang sangat menarik untuk keperluan pembangunan situs *web* yang interaktif dan dinamis. *Flash* didesain dengan kemampuan untuk membuat animasi 2 dimensi yang andal dan ringan sehingga *Flash* banyak digunakan untuk membangun dan memberikan efek animasi pada *website*, CD interaktif dan yang lainnya. Selain itu, aplikasi ini juga dapat digunakan untuk membuat animasi logo, *movie*, *game*, pembuatan navigasi pada situs *web*, tombol animasi, banner, *menuinteraktif*, *interaktiform* isian, *e-card*, *screensaver*, dan pembuatan aplikasi-aplikasi *web* lainnya. Dalam *Flash*, terdapat teknik-teknik membuat animasi, fasilitas *actionsript*, *filter*, *customeasing* dan dapat memasukkan *video* lengkap dengan fasilitas *playback* FLV. Keunggulan yang dimiliki oleh *Flash* ini adalah ia mampu diberikan sedikit code pemrograman baik yang berjalan sendiri untuk mengatur animasi yang ada di dalamnya atau digunakan untuk berkomunikasi dengan program lain seperti HTML, PHP, dan Database dengan pendekatan XML, dapat dikolaborasikan dengan *web*, karena mempunyai keunggulan antara lain kecil dalam ukuran file outputnya. (Kriston Radion, S.ST:2012;2-3).

Sebuah program grafis animasi standard profesional untuk menghasilkan produk-produk multi media seperti *Multimedia Persentation*, *wabsite*, *Computer Game*, dan *Animation*. Perogram ini mampu menghasilkan animasi yang begitu canggih, sehingga besar aplikasi tutorial yang interaktif, *game*, presentasi, dan lain-lain dibuat dengan program ini. Flase CS5.5 merupakan pengembangan dari penyempurnaan dari virsi sebelumnya (*Flash profesional* 2004). Ada beberapa panel pada plash yang harus diketahui sebagai dasar pembuatan animasi.

II.3.1. Area kerja macromedia Flash Player

Langkah untuk menjalankan program Adobe Flash Pro CS5.5, tekan tombol **Start ► All Programs ► Adobe ► Adobe Flash CS5.5** sehingga tampil **Welcome Screen** seperti tampak pada gambar berikut, (Madcoms Madium; 2012: 4).



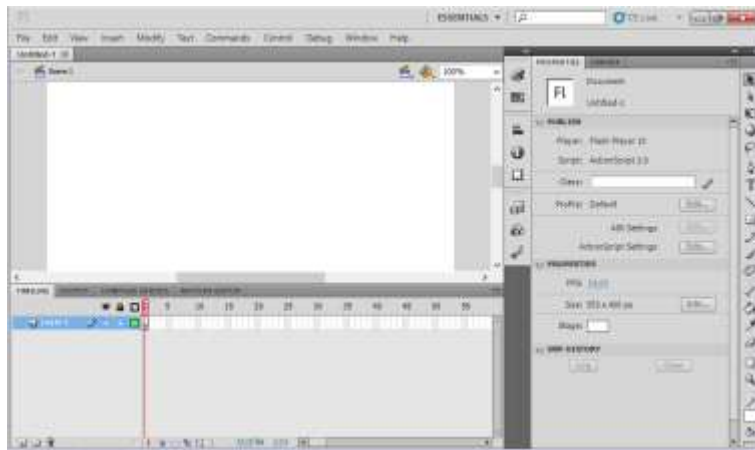
Gambar II.3. Tampilan layar pertama program Adobe Flash Pro CS5.5

Sumber : (Madcoms Madium; 2012: 4)

Welcome Screen menampilkan empat pilihan perintah untuk memulai Adobe Flash Pro CS5.5, yaitu:

1. **Create from Template**, berguna untuk membuka lembar kerja dengan template yang tersedia dalam program Adobe Flash Pro CS5.5.
2. **Open a Recent Item**, berguna untuk membuka kembali file yang pernah Anda simpan atau pernah Anda buka sebelumnya.
3. **Create New**, berguna untuk membuka lembar kerja baru dengan beberapa pilihan script yang tersedia.
4. **Learn**, berguna untuk membuka jendela Help yang berguna untuk mempelajari suatu perintah, (Madcoms Madium; 2012: 4-5).

Jika Anda tidak ingin menampilkan jendela Welcome Screen lagi saat membuka program, aktifkan kotak periksa **Don't Show again** yang terdapat pada sisi bawah dari jendela Welcome Screen. Sebagai contoh klik perintah **ActionScript 2.0** pada bagian Create New sehingga tampil lembar kerja seperti Gambar II.4.



Gambar II.4. Tampilan Jendela program Adobe Flash Pro CS5.5

Sumber : (Madcoms Madium; 2012: 5)

II.4. Gereja Katedral Graha Bunda Maria Annai Velangkani

Sebuah bangunan kuno di medan yang dijadikan tempat beribadah bagi umat katolik, sekaligus objek wisata adalah Gereja Graha Maria Annai Velangkani. Gereja ini terletak di Kecamatan Tuntungan, Kelurahan Tanjung Selamat, di Jalan Sakura III, dekat Perumahan Taman Sakura indah. Dilihat sepintas, bangunan gereja ini tidak mirip dengan gereja, justru mirip dengan kuil yang ada di India. Orang yang membangun gereja ini adalah Pastor James Bharataputra. Selain bangunan ini dari luar nampak seperti kuil, di dalamnya pun ternyata terdapat relief-relief dan patung yang mirip dengan patung yang ada di kuil India.

Desain interior merupakan suatu ilmu yang membahas tentang perancangan karya seni pada suatu bangunan. Perancangan itu sendiri meliputi interior tetap, bergerak, dan juga dekoratif yang sifatnya hanyalah sementara.

Diawali dengan prinsip ‘menghias’ bukan ‘menciptakan’. Hal ini tercermin ketika sampai dengan abad 20, di seluruh Eropa dan Amerika yang berperan merancang interior rumah adalah pembuat mebel, atau para pedagang bahan salut mebel, pedagang tirai bahkan pedagang perabot rumah tangga. Sambil berjualan mereka sekaligus memberi saran gratis cara menata isi dan bagian dalam rumah (memilih bahan tekstil, bahan lantai, mebel lampu dan asesoris ruangan, serta usulan harmonisasi warna ruang). Pada awal abad 20, perkembangan pesat di bidang sosial-ekonomi mendorong munculnya lebih banyak dekorator interior. Penggunaan dekorator interior terutama terjadi di kalangan masyarakat tingkat atas karena dianggap mencerminkan status tertentu, prestis dan berpengaruh .

Arsitektur kolonial datang ke Indonesia terjadi percampuran dari arsitektur Eropa, dengan proses adaptasi dan membentuk Belanda pada masa penjajahan di Indonesia. Arsitektur ini telah beradaptasi dengan iklim setempat. Untuk membuat bangunan yang cocok dengan lingkungan serta nyaman untuk hidup.

Gedung gereja atau bangunan gereja adalah bangunan atau struktur yang tujuan utamanya adalah untuk memfasilitasi pertemuan sebuah gereja. Awalnya, orang Kristen Yahudi bertemu di rumah-rumah ibadat, seperti senakel, dan di rumah-rumah, yang dikenal sebagai gereja rumah. Ketika Kekristenan tumbuh dan diterima oleh pemerintah, khususnya dengan Edik Milano, ruangan dan akhirnya, seluruh bangunan dikhususkan untuk tujuan ibadah Kristen, misalnya Gereja Katedral Graha Bunda Maria Annai Velangkani.

Bangunan gereka tradisional seringkali dalam bentuk salib dan sering memiliki sebuah menar dan kubah. Gedung-gedung gereja yang lebih modern memiliki berbagai variasi arsitektur dan tata letak, banyak bangunan yang sebelumnya dirancang untuk tujuan lain kini telah

dikonversi untuk digunakan sebagai gereja, demikian pula, banyak bangunan gereja yang asli telah dijadikan untuk kegunaan lainnya.

II.4.1. Rumah Ibadah

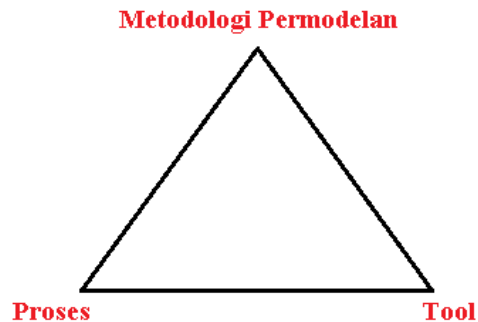
Tempat ibadah, rumah ibadah, tempat peribadatan adalah sebuah tempat yang digunakan oleh umat beragama untuk beribadah menurut ajaran agama atau kepercayaan mereka masing-masing. Paul M. Taylor menjelaskan hak untuk beribadah dalam hubungannya dengan rumah ibadah, tidak hanya mencakup hak mendirikan rumah ibadah (*to establish*), tetapi juga bagaimana hak untuk menjalankan/menjaga rumah ibadah tersebut (*to maintain*).

Menurut Nowak, aktivitas-aktivitas penyebaran ajaran agama seperti dakwah dan *missionary* juga dapat dikategorikan praktek-praktek keagamaan. Hak untuk mendirikan dan menjalankan rumah ibadah merupakan bagian dari kebebasan memmanifestasikan agama/kepercayaan.

II.5. UML (*Unified Modelling Language*)

Pemodelan (*modeling*) adalah proses merancang piranti lunak sebelum melakukan pengkodean (*coding*). Model piranti lunak dapat dianalogikan seperti pembuatan blueprint pada pembangunan gedung. Membuat model dari sebuah sistem yang kompleks sangatlah penting karena kita tidak dapat memahami sistem semacam itu secara menyeluruh. Semakin kompleks sebuah sistem, semakin penting pula penggunaan teknik pemodelan yang baik. Dengan menggunakan model, diharapkan pengembangan piranti lunak dapat memenuhi semua kebutuhan pengguna dengan lengkap dan tepat, termasuk faktor-faktor seperti *scalability*, *robustness*, *security*, dan sebagainya. Kesuksesan suatu pemodelan piranti lunak ditentukan oleh

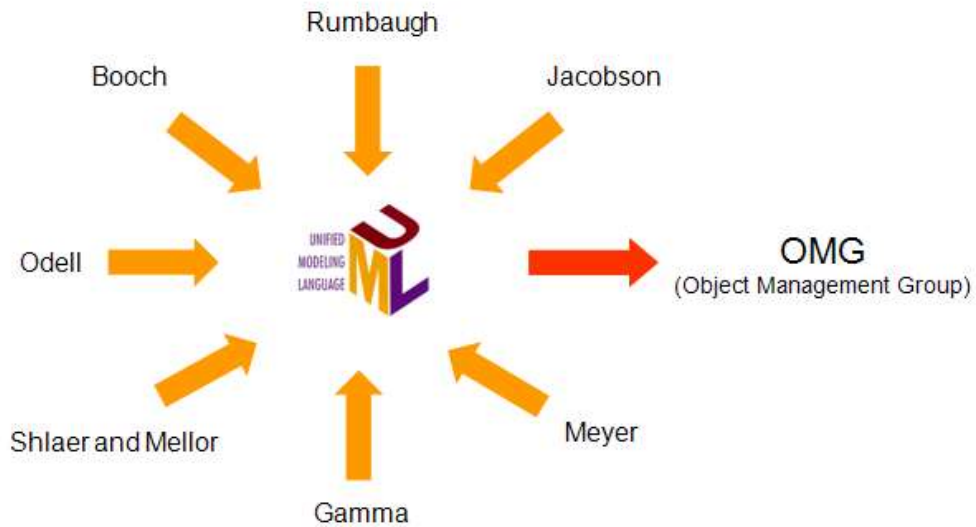
tiga unsur, yang kemudian terkenal dengan sebuah segitiga sukses (*the triangle for success*). Ketiga unsur tersebut adalah metode pemodelan (*notation*), proses (*process*) dan *tool* yang digunakan. Memahami notasi pemodelan tanpa mengetahui cara pemakaian yang sebenarnya (proses) akan membuat proyek gagal. Dan pemahaman terhadap metode pemodelan dan proses disempurnakan dengan penggunaan tool yang tepat.



Gambar II.5. The Triangle For Success
Sumber : (Yuni Sugiarti ; 2013 : 33)

Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C. Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu,

dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (*Object-Oriented Design*), Jim Rumbaugh OMT (*Object Modeling Technique*), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*). Sejarah UML sendiri cukup panjang. Sampai era tahun 1990 seperti kita ketahui puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: *metodologi booch*, *metodologi coad*, *metodologi OOSE*, *metodologi OMT*, *metodologi shlaer-mellor*, *metodologi wirfs-brock*, dsb. Masa itu terkenal dengan masa perang metodologi (*method war*) dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan group/perusahaan lain yang menggunakan metodologi yang berlainan. Dimulai pada bulan Oktober 1994 *Booch, Rumbaugh dan Jacobson*, yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan memelopori usaha untuk penyatuan metodologi pendesainan berorientasi objek. Pada tahun 1995 direlease draft pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh Object Management Group (OMG – <http://www.omg.org>). Tahun 1997 UML versi 1.1 muncul, dan saat ini versi terbaru adalah versi 1.5 yang dirilis bulan Maret 2003. Booch, Rumbaugh dan Jacobson menyusun tiga buku serial tentang UML pada tahun 1999. Sejak saat itulah UML telah menjelma menjadi standar bahasa pemodelan untuk aplikasi berorientasi objek.



Gambar II.6. Metodologi Pemodelan Berorientasi Objek

Sumber : (Yuni Sugiarti ; 2013 : 33)

Dalam hal ini, pemodelan perangkat lunak bekerja dengan cara yang cukup serupa layaknya seorang arsitek atau insinyur teknik sipil yang akan membuat sebuah bangunan/dedung berskala besar. Saat seorang arsitek atau insinyur teknik sipil akan membuat sebuah bangunan gedung berskala besar, ia biasanya membuat denah-denang atau maket-maket yang menggambarkan bentuk jadi dari bangunan/gedung.

Dari berbagai penjelasan rumit yang terdapat di dokumen dan buku-buku UML. Sebenarnya konsepsi dasar UML bisa kita rangkumkan dalam gambar dibawah.

I

Tabel II.1. Konsep Dasar UML

<i>Major Area</i>	<i>View</i>	<i>Diagram</i>	<i>Main Concept</i>
<i>Structural</i>	<i>Static View</i>	<i>Class Diagram</i>	<i>Class, Association, Generalization, Dependency, Realization, Interface</i>
	<i>Use Case View</i>	<i>Use Case Diagram</i>	<i>Use Case, Actor, Association, Extend, Include, Use Case Generalization</i>
	<i>Implementation view</i>	<i>Component Diagram</i>	<i>Component, Interface, Dependency, Location</i>
	<i>Deployment view</i>	<i>Deployment Diagram</i>	<i>Node, Component, Depedency, Location</i>
	<i>Interaction View</i>	<i>Sequence Diagram</i>	<i>Interaction, Object, Message, Activation</i>
		<i>Class Diagram</i>	<i>Collaboration, interaction, Collaboration Role, Message</i>
<i>Model Management</i>	<i>Model Management View</i>	<i>Class Diagram</i>	<i>Package, Subsystem, Model</i>
<i>Extensibility</i>	<i>All</i>	<i>All</i>	<i>Constraint, Stereotype, Tagged Values</i>

Sumber : (Yuni Sugiarti ; 2013 : 35)

Seperti juga tercantum pada gambar diatas UML mendefinisikan diagram-diagram sebagai berikut:

1. *Use case diagram*
2. *Class diagram*
3. *Statechart diagram*
4. *Activity diagram*
5. *Sequence diagram*
6. *Collaboration diagram*
7. *Component diagram*
8. *Deployment diagram*

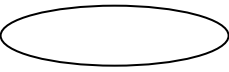
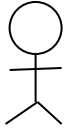

Dalam pembuatan skripsi ini penulis menggunakan diagram Use Case yang terdapat di dalam UML. Adapun maksud dari Use Case Diagram diterangkan dibawah ini.

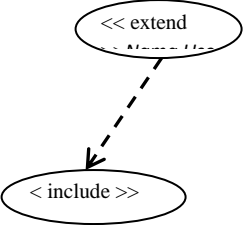
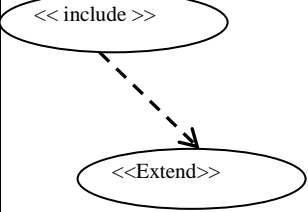
II.5.1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case diagram* dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara

umum diasumsikan bahwa *use case* yang di-include akan dipanggil setiap kali *use case* yang meng-include dieksekusi secara normal. Sebuah *use case* dapat di-include oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-extend *use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

Tabel II.2. Simbol Use Case

Simbol	Deskripsi
<p data-bbox="277 783 399 814"><i>Use Case</i></p> 	<p data-bbox="513 783 1243 961">Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit dan aktor, biasanya menggunakan kata kerja di awal frase nama <i>use case</i></p>
<p data-bbox="302 1003 375 1035">Aktor</p>  <p data-bbox="285 1367 391 1398"><i>Nama Aktor</i></p>	<p data-bbox="513 1003 1243 1398">Orang atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat sistem itu sendiri. Jadi walaupun simbol aktor adalah gambar orang, tapi aktor belum tentu merupakan orang. Biasanya menggunakan kata benda di awal frase nama aktor</p>
<p data-bbox="196 1444 483 1476"><u>Asosiasi / Association</u></p> 	<p data-bbox="513 1444 1243 1549">Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi <i>use case</i> atau <i>use case</i> berinteraksi dengan aktor.</p>

<p style="text-align: center;">Extend</p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walupun tanpa <i>use case</i> tambahan itu, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambhan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, arah panah menunjukan pada <i>use case</i> yang dituju. Contoh :</p>
<p style="text-align: center;">Include</p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini. ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>, <i>include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, contoh;</p>

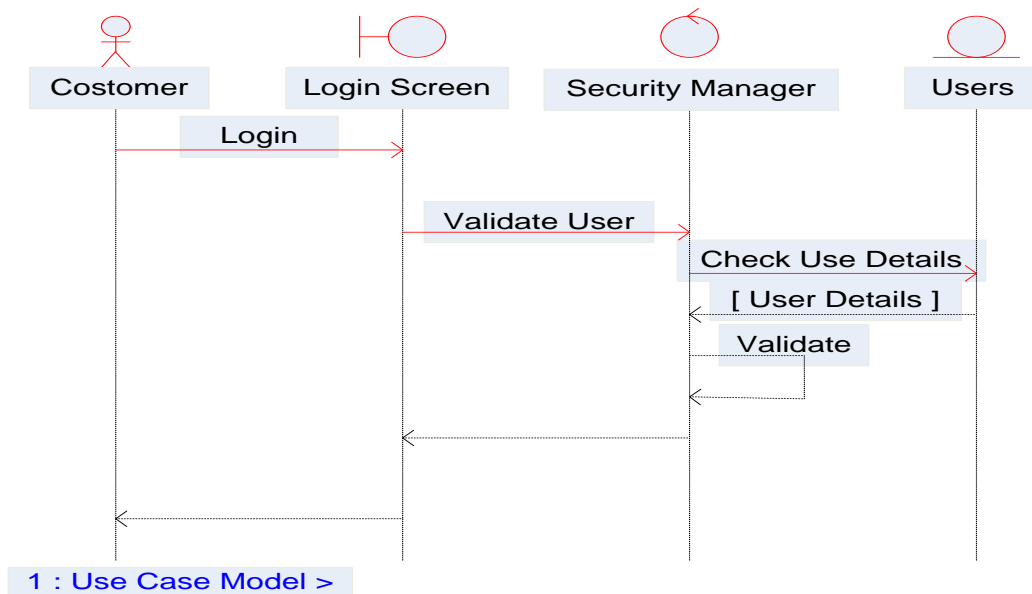
Sumber : (Yuni Sugiarti ; 2013 : 42)

II.5.2. Sequence Diagram

Diagram *Sequence* menggambarkan kelakuan/prilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram *sequence* maka harus diketahui objek-objek

yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Banyaknya diagram *sequence* yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram *sequence* sehingga semakin banyak *use case* yang didefinisikan maka diagram *sequence* yang harus dibuat juga semakin banyak.



Gambar II.7. Contoh Sequence Diagram
Sumber : (Yuni Sugiarti ; 2013 : 63)

II.5.3. Activity Diagram

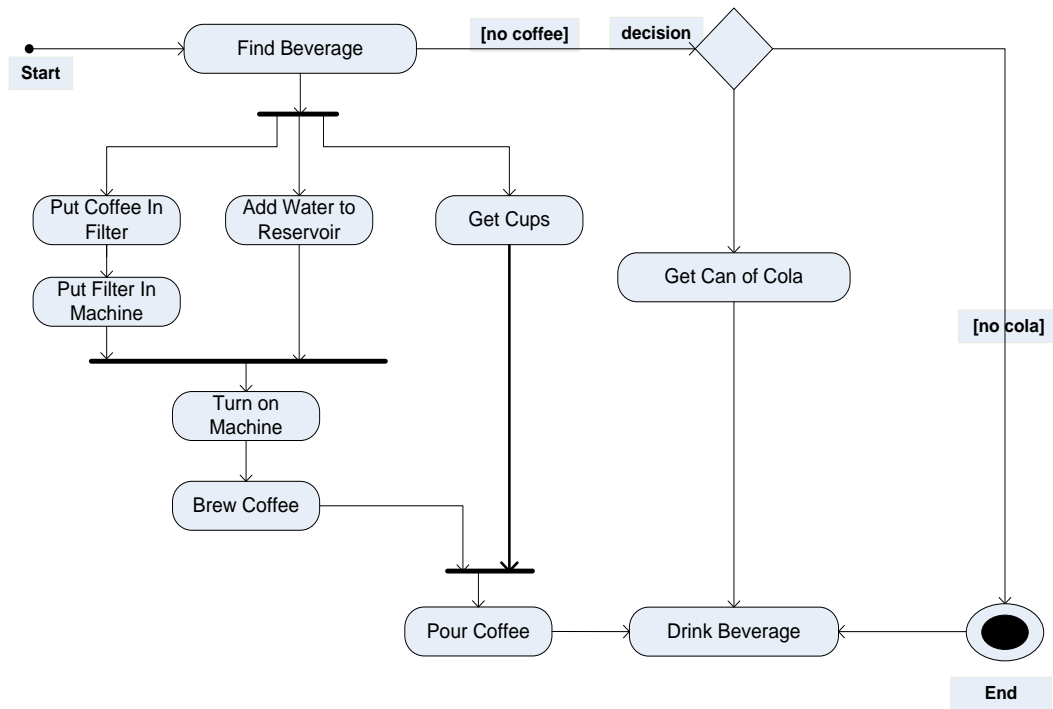
Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal.

Activity diagram dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.



Gambar II.8. Activity Diagram
Sumber : (Yuni Sugiarti ; 2013 : 76)

II.6. 3D Max

3ds Max adalah sebuah software yang dikhususkan dalam pemodelan 3 dimensi ataupun untuk pembuatan animasi 3 dimensi. Selain terbukti andal untuk digunakan dalam pembuatan objek 3 dimensi, 3ds Max juga banyak digunakan dalam pembuatan desain furniture, konstruksi, maupun desain interior. Selain itu, 3ds Max juga sering digunakan dalam pembuatan animasi atau film kartun. (Galih Pranowo, 2010:6).

3D Studio Max adalah software visualisasi (modeling dan animasi Tiga Dimensi yang populer dan serbaguna. Hasil yang dibuat di 3D Studio Max sering digunakan di pertelevisian, Media Cetak, Games, Web dll. (Hendi Hendratman, 2014).

Sekilas tentang aplikasi 3ds Max yaitu suatu aplikasi yang sangat membantu Anda dalam penggambaran objek 3 dimensi, sehingga hasil gambar Anda akan terlihat seperti nyata. (Anditya, 2015: 105).

3D Studio Max 2010 adalah versi terbaru dari serial 3D Studio Max yang merupakan salah satu piranti lunak grafis 3D yang dapat menghasilkan desain objek 3D dengan kualitas dengan kemampuan yang profesional dan realistik. Software ini banyak digunakan oleh para animator dalam mewujudkan gagasan menjadi hasil karya seperti objek 3D untuk arsitektur, desain grafis, dan juga film-film animasi. (Wahana Komputer, 2010: 2).

II.7. Adobe Flash

Adobe Flash merupakan software multifungsi yang mempermudah pembuatan animasi, web, game, dan aplikasi multimedia lainnya. Versi terbaru dari Flash yang dimulai dari versi CS5.5 Profesional hingga yang terbaru saat ini dilengkapi dengan AIR for Android extension. Tidak ada instalasi tambahan yang diperlukan. (Wahana Komputer, 2014: 3, 4).

Adobe Flash CS6 merupakan versi terbaru dari versi sebelumnya, Adobe Flash CS5. Program ini memiliki banyak fungsi, seperti pembuatan animasi objek, membuat presentasi, animasi iklan, game, pendukung animasi halaman web, hingga dapat digunakan untuk pembuatan film animasi. (Wahana Komputer, 2012: 2).

Adobe Flash adalah program yang sangat populer untuk membuat animasi 2D berbasis vector. Kecanggihan dan kelengkapan fitur yang dimiliki Flash, membuat program ini banyak diminati oleh para animator dalam membuat berbagai macam karya animasi 2D. (Madcoms Madiun, 2013: 2).

Adobe Flash merupakan program animasi berbasis vector, yang telah banyak digunakan oleh para animator untuk membuat berbagai animasi. Sekarang ini program Adobe Flash Pro CS6 telah mampu mengolah teks maupun objek dengan efek tiga dimensi sehingga tampak lebih menarik. (Madcoms Madiun, 2013: 2).

Adobe Flash CS3 Professional adalah sebuah program animasi yang telah banyak digunakan oleh para Animator untuk menghasilkan animasi yang professional. Di antara program-program animasi, program Adobe Flash CS3 Professional merupakan program yang paling fleksibel dalam pembuatan animasi, seperti Animasi Interaktif Game, Company Profile, Presentasi, Movie, e-card dan animasi yang digunakan dalam situs web. (Madcoms Madiun, 2008: 1).

Animasi iklan layanan masyarakat pada bab ini bertemakan tentang keselamatan berlalu lintas. Animasi ini akan menggunakan Create Classic Tween dengan ActionScript 2.0. Sebelum Anda membuat animasi pada Adobe Flash, terlebih dahulu Anda mengolah gambar yang akan digunakan di *background* dengan menggunakan Adobe Photoshop. Gambar tersebut disambung dengan gambar hasil duplikasi dari gambar aslinya agar memiliki ukuran yang lebih panjang. (Wahana