

BAB II

TINJAUAN PUSTAKA

II.1. Kapal Selam

Kapal selam adalah kapal yang bergerak di bawah permukaan air, umumnya digunakan untuk tujuan dan kepentingan militer. Sebagian besar angkatan laut memiliki dan mengoperasikan kapal selam sekalipun jumlah dan populasinya masing-masing di negara berbeda. Selain digunakan untuk kepentingan militer, kapal selam juga digunakan untuk ilmu pengetahuan dan bertugas di kedalaman yang tidak sesuai untuk penyelam manusia.(Aji Sokaroni,FT UI,2012: 6)

Dengan menggunakan Cavitation Tunnel, penelitian mengenai sinyal akustik yang dihasilkan oleh fenomena kavitasi yang berasal dari obyek bawah air dapat dilakukan. Sistem pengukuran sinyal akustik yang disebabkan oleh propeller dengan menggunakan fasilitas cavitation tunnel di UPT BPPH - BPPT telah dilakukan dan dianalisa pada penelitian ini. Di sini dibuktikan bahwa terjadinya kavitasi dapat disimulasikan untuk berbagai nilai kecepatan aliran air, kecepatan rotasi propeller dan tekanan air. Dengan proses sinkronisasi antara pengukuran sinyal akustik dan perekaman video, mekanisme fisik dan karakteristik dari sinyal akustik yang diukur dapat dipelajari dalam kaitannya dengan fenomena yang dihasilkan seperti yang terdapat pada hasil rekaman sinyal akustik dan gambar (video). Penyelidikan lebih lanjut dapat dilakukan dengan menggunakan berbagai jenis baling-baling. (Endang Widjiati, 2012)

II.2. Perancangan

Perancangan adalah spesifikasi umum dan terinci dari pemecahan masalah berbasis komputer yang telah dipilih selama tahap analisis. Berdasarkan definisi tersebut dapat disimpulkan bahwa perancangan adalah kemampuan untuk membuat alternatif pemecahan masalah berbasis komputer selama tahap analisis. (Azhar Susanto; 2004:332)

II.3. Multimedia

Definisi Multimedia Interaktif Sistem yang menggunakan lebih dari satu media presentasi (Teks, Suara, Citra, Animasi dan Video) secara bersamaan dan melibatkan keikutsertaan pemakai untuk memberi perintah, mengendalikan dan memanipulasi. (Ilham Eka Putra; 2013: 1)

II.4. Media Animasi

Media animasi merupakan pengembangan dari penggunaan komputer yang dimanfaatkan dalam bidang pendidikan. Menurut Wojowasito (dalam Djalle, 2007) animasi merupakan suatu kegiatan menghidupkan benda mati yang diberikan dorongan kekuatan, semangat, dan emosi untuk menjadi hidup dan bergerak, atau hanya berkesan hidup. Media animasi berisi kumpulan gambar yang diolah sedemikian rupa dan menghasilkan gerakan sehingga berkesan hidup serta menyimpan pesan-pesan pembelajaran. Maryanto (2010) menyatakan bahwa, keistimewaan dari media animasi adalah memvisualisasikan konsep abstrak yang tidak dapat diamati indera penglihatan secara langsung. (Marta Maria Dona; 2013 : 3)

II.5. Sejarah Animasi

Animasi 3D (Tiga Dimensi) Perkembangan teknologi dan dunia computer membuat teknik pembuatan animasi 3D semakin berkembang dan maju pesat. Animasi 3D adalah perkembangan dari animasi 2D. Dengan animasi 3D, karakter yang diperlihatkan semakin hidup dan nyata, mendekati wujud aslinya. (Yunita Syahfitri; 2011 : 3)

Perkembangan film animasi 3D tidak lepas dari sejarah perkembangan komputer. Pada awal tahun 1940-an, percobaan komputer grafis dimulai, dan dengan berjalannya waktu perkembangan inovatif komputer grafis berjalan. Pada mulanya, penggunaan diutamakan untuk tujuan penelitian ilmiah dan teknik. Pada pertengahan tahun 1960-an mulai muncul eksperimentasi artistik. Di sekitar tahun 1970-an banyak pengembangan pada komputer animasi, pengembangan menuju pada realistis dalam citra 3D, dan perancangan efek-efek untuk film. 21 Volume 3 No. 1 Desember 2011 Berbagai upaya tersebut mulai diperkenalkan pada media publik. Film yang sudah menggunakan teknologi komputer adalah Star Wars: A New Hope (1977), tetapi masih dalam bentuk garis dan belum ada bayangan (wireframe). Pemakaian komputer dicoba pada formasi penerbangan pesawat ruang angkasa X-wing fighters, tetapi menggunakan model tradisional kembali ketika teknologi saat itu kurang begitu mengesankan. Pada akhir1980-an, foto-realistis 3D mulai muncul dalam film bioskop, dan seiring daya komputer semakin meningkat, upaya untuk mencapai realistis 3D menjadi hal yang penting. Pada pertengahan 1990-an telah berkembang pada animasi 3D secara menyeluruh. (Yasuki Rahmad; 2011 : 3)




II.6. Unified Modeling Language (UML)

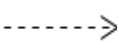






UML yang merupakan singkatan dari *Unified Modelling Language* adalah sekumpulan pemodelan konvensi yang digunakan untuk menentukan atau menggambarkan sebuah sistem

perangkat lunak dalam kaitannya dengan objek. UML dapat juga diartikan sebuah bahasa grafik standar yang digunakan untuk memodelkan perangkat lunak berbasis objek. UML pertama kali dikembangkan pada pertengahan tahun 1990an dengan kerjasama antara James Rumbaugh, Grady Booch dan Ivar Jacobson, yang masing-masing telah mengembangkan notasi mereka sendiri di awal tahun 1990an. (Lethbride dan Leganiere; 2009:11)

II.6.1. Use Case Diagram

Use case diagram, adalah sebuah gambaran dari fungsi sistem yang dipandang dari sudut pandang pemakai. *Actor* adalah segala sesuatu yang perlu berinteraksi dengan sistem untuk pertukaran informasi. *System boundary* menunjukkan cakupan dari sistem yang dibuat dan fungsi dari sistem tersebut. (Lethbride dan Leganiere; 2009:11)

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).

4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi






Gambar II.1. Simbol-simbol yang ada pada Use Case Diagram

(Sumber :Lethbride dan Leganiere; 2009:11)

II.6.2. 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. Menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses. Dipakai pada *business modeling* untuk memperlihatkan urutan aktifitas proses bisnis. Struktur diagram ini mirip *flowchart* atau *Data Flow Diagram* pada perancangan terstruktur. Activity diagram dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram*. (Lethbride dan Leganiere; 2009:13)

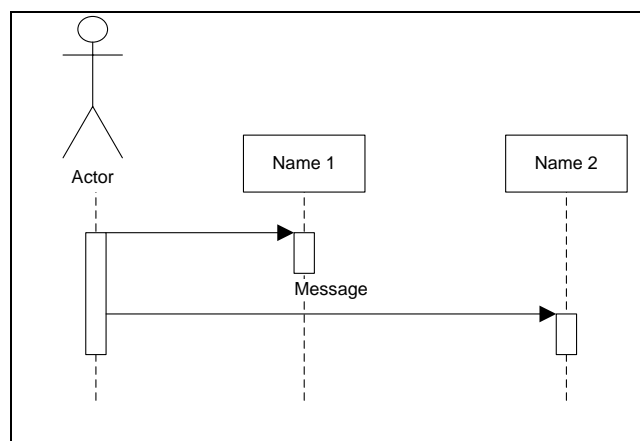
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actifity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Actifity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Gambar II.2. Simbol-simbol yang ada pada Activity Diagram

(Sumber :Lethbride dan Leganiere; 2009:15)

II.6.3. *Sequence diagram*

Sequence diagram menambahkan dimensi waktu pada interaksi diantara obyek. Pada diagram ini participant diletakkan di atas dan waktu ditunjukkan dari atas ke bawah. *Life line participant* diurutkan dari setiap participant. Kotak kecil pada life line menyatakan *activation* : yaitu menjalankan salah satu operation dari participant. Satate bisa ditambahkan dengan menempatkannya sepanjang life line. *Message* (sederhana, synchronous atau *asynchronous*) adalah tanda panah yang menghubungkan suatu *life line* ke *life line* yang lain. Lokasi *life line* dalam dimensi vertikal mewakili urutan waktu dalam *sequence diagram*. Message yang pertama terjadi adalah yang paling dekat dengan bagian atas diagram dan yang terjadi belakangan adalah yang dekat dengan bagian bawah. Pada beberapa sistem, operasi bisa dilakukan kepada dirinya sendiri. Hal ini disebut dengan rekursif. Untuk melukiskannya digunakan anak panah dari *activation* kembali ke dirinya sendiri, dan sebuah kotak kecil diletakkan pada bagian atas dari *activation*.



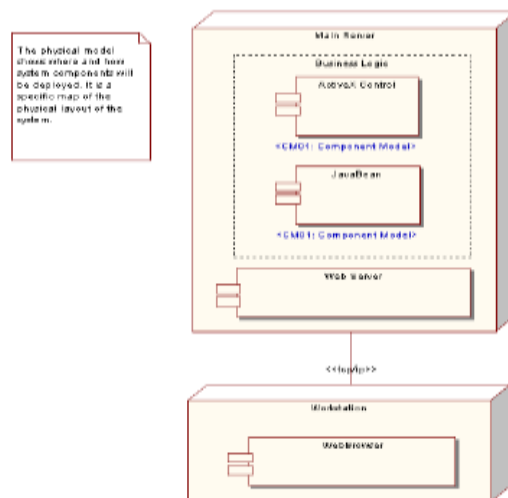
Gambar II.3. Simbol-simbol yang ada pada *sequence diagram*

(Sumber :Agus Putranto; 2009:14)

II.6.4. *Deployment diagram*

Deployment diagram menggambarkan sumber fisik dalam sistem, termasuk node, komponen dan koneksi (model implementasi sistem yang statistik). Dalam hal ini meliputi topologi *hardware* yang dipakai sistem.

Deployment/physical diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisik. Sebuah *node* adalah server, *workstation*, atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya. Hubungan antar *node* (misalnya TCP/IP) dan *requirement* dapat juga didefinisikan dalam diagram ini.



Gambar II.4. Simbol-simbol yang ada pada *Deployment Diagram*

(Sumber :Agus Putranto;2009:9)

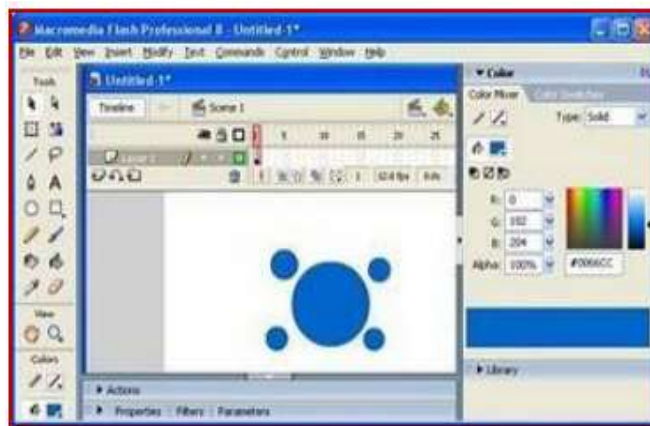
II.7. Pengenalan Macromedia Flash Player 8

Sebuah program grafis animasi standard professional untuk menghasilkan produk-produk multimedia seperti *Courseware*, *Multimedia Presentation*, *Website*, *Computer Game*, dan *Animation*. Program ini mampu menghasilkan animasi yang demikian canggih, sehingga besar

aplikasi tutorial yang interaktif, *game*, presentasi, dan lain-lain dibuat dengan program ini. *Flash professional 8* merupakan pengembangan dan penyempurnaan dari versi sebelumnya (Flash 5, Flash 6/MX, Flash MX professional 2004). Ada beberapa *panel* pada *flash* yang harus diketahui sebagai dasar pembuatan animasi :

II.7.1. Area Kerja Macromedia Flash Player

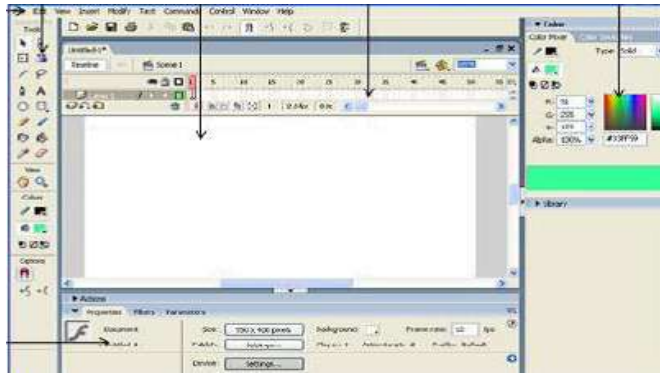
Saat pertama kali menjalankan program Macromedia Flash Player, maka kita akan mendapati tampilan halaman pembuka Macromedia Flash Player seperti yang terlihat pada Gambar II.5 di bawah ini :



Gambar II.5. Tampilan Halaman Macromedia Flash 8

(Sumber :Amal Jamaludin; 2010:11)

Setelah proses *loading* program *Macromedia Flash Player* selesai, maka akan tampil bagian antarmuka dari *Macromedia Flash Player*. Area kerja *Macromedia Flash Player* dapat dilihat pada Gambar II.6

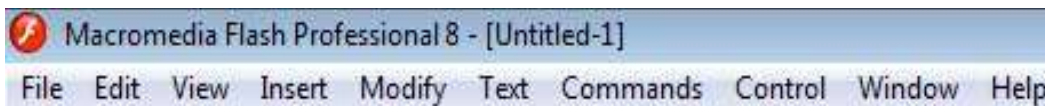


Gambar II.6. Tampilan Macromedia Flash Player

(Sumber : Amal Jamaludin; 2010:12)

II.7.2. Menu Bar

Menu, berisi kumpulan instruksi atau perintah-perintah yang digunakan dalam *Flash*, misalnya, klik menu File > Save berfungsi untuk menyimpan dokumen,. Menu terletak di bagian area Flash. Berikut ini merupakan gambar dari *Menu Bar*.

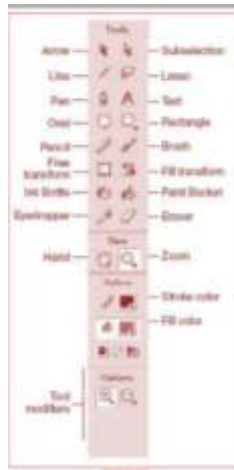


Gambar II.7. Tampilan Menu Bar

(Sumber : Amal Jamaludin; 2010 : 16)

II.7.3. Toolbox

Toolbox, berisi tool-tool yang berfungsi untuk membuat, menggambar, memilih dan memanipulasi obyek atau isi yang terdapat di layar dan *timeline*. *Toolbox* dibagi menjadi 4 bagian, yaitu *tools*, *view*, *colors*, dan *options*. Berikut ini merupakan gambar dari Main *Toolbox*.



Gambar II.8. Tampilan Toolbox

(Sumber :Amal Jamaludin; 2010:13)

II.7.4. Panel

Panel, berisi kontrol fungsi yang dipakai dalam *flash*, yang berfungsi untuk mengganti danmemodifikasi berbagai atribut dari objek atau animasi secara cepat dan mudah. *Panels* biasanya terletak di bagian kanan area *Flash*. Untuk menampilkan panel tertentu, klik menu Window > (nama *panel*). Berikut ini merupakan gambar dari *panel*.

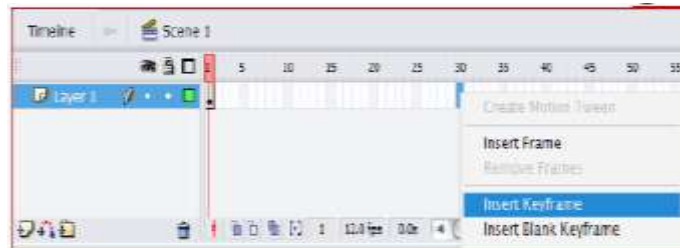


Gamba. Tampilan Panel

(Sumber : Amal Jamaludin; 2010:21)

II.7.5. Timeline

Timeline, berisi *layer* dan *frame-frame* yang berfungsi untuk mengontrol *object* yang akan dianimasikan. *Timeline* terletak dibawah menu. Berikut ini merupakan gambar dari *timeline*.

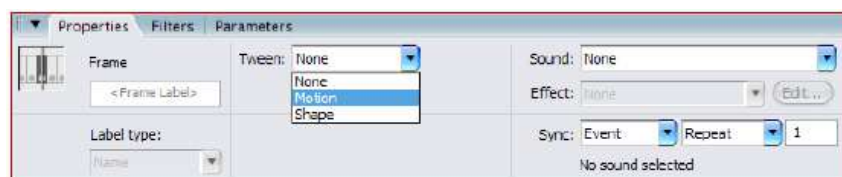


Gambar II.10. Tampilan Timeline Animation

(Sumber : Amal Jamaludin; 2010:19)

II.7.6. Properties

Properties, berfungsi hampir sama dengan panels, hanya saja *properties* merupakan penggabungan atau penyederhanaan dari *panel*. Jadi dapat lebih mempercepat dalam mengganti dan memodifikasi berbagai atribut dari objek, animasi, *frame* dan komponen secara langsung. Berikut ini merupakan gambar dari *properties*.



Gambar II.11. Tampilan Properties

(Sumber : Amal Jamaludin; 2010:19)

II.7.7. Stage

Stage adalah dokumen atau layar yang akan digunakan untuk meletakkan obyek-obyek dalam *flash*. *Stage* terletak pada bagian tengah area *flash*. Berikut ini merupakan gambar dari *stage*.

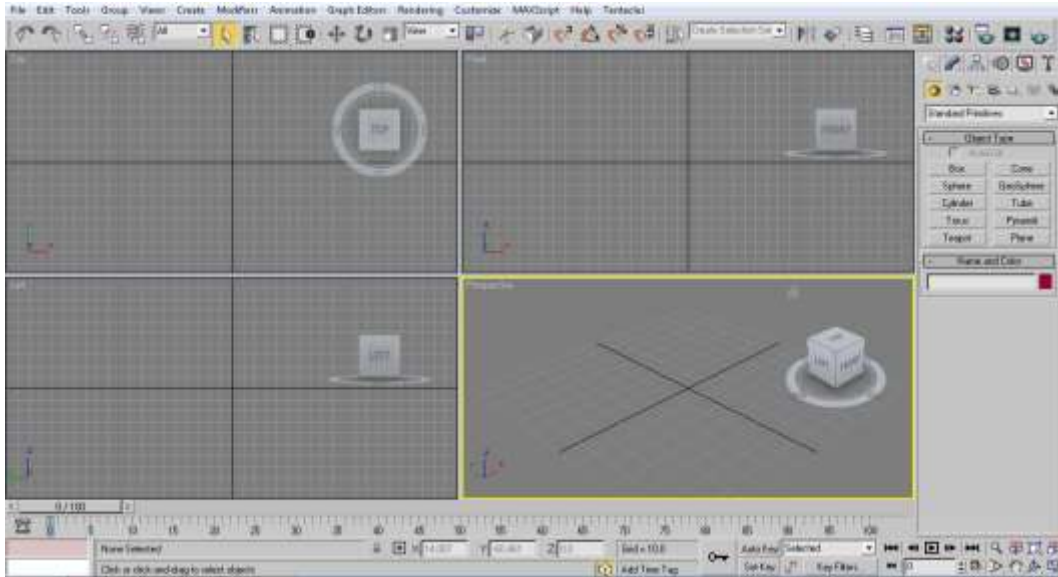


Gambar II.12. Tampilan Stage

(Sumber : Amal Jamaludin;2010:17)

II.8. Autodesk 3Ds Max 2009

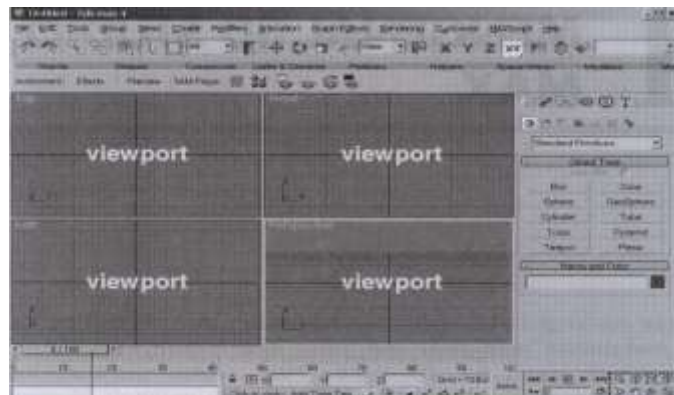
Pembuatan modeling 3 dimensi serta animasi yang mendekati keadaan sebenarnya atau yang sering disebut *Prototype* sangat dimungkinkan dengan bantuan *software design graphic*. Perkembangannya pada saat ini sangatlah pesat. 3D Studio Max merupakan salah satu software yang ada untuk membantu para designer modeling 3 dimensi membuat karyanya dengan mengembangkan ide dan imajinasinya kedalam bentuk visual. Adapun perancangan situs-situs web, advertising, kios, *broadcasting*, film, pendidikan game dan entertainment pada saat ini sering mengikiut sertakan animasi dinamis. (Adhi Dharma Suriyanto, 2010 : 2).



Gambar II.13. Interface 3ds Max 2009
 (Sumber : Adhi Dharma Suriyanto; 2010 : 2)

II.8.1. Interface

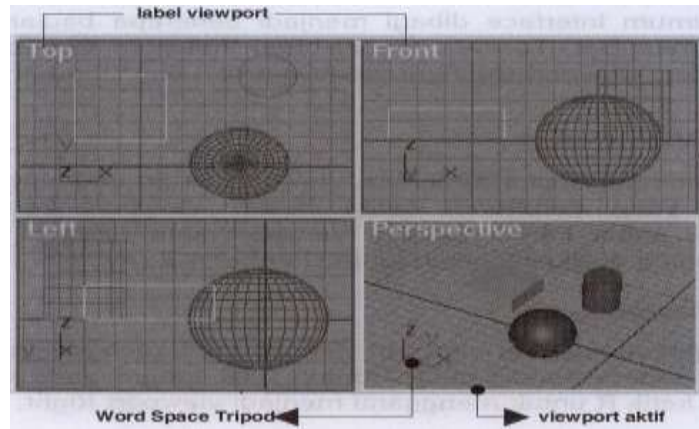
Dibagi beberapa bagian, yaitu; *Title Bar*, *Menu Bar*, *Tool Bar*, empat buah *Viewport*, *Control Panel*, *Time Slider*, *Viewport Configuration Control*.



Gambar II. 14. Tampilan Interface
 (Sumber : Adhi Dharma Suriyanto; 2005: 4)

II.8.2. Viewport

Viewport terdiri dari empat buah viewport yang berukuran sama besar. *Viewport Perspektif* berada di sudut kanan bawah, *Viewport* aktif ditandai dengan warna kuning disekelilingnya.



Gambar II. 15. Tampilan Viewport
(Sumber : Adhi Dharma Suriyanto; 2005: 4)

II.8.3. Layout Viewport

Klik kanan pada *viewport left*, lalu ketik R untuk mengganti *viewport Right*

Viewport Tunggal

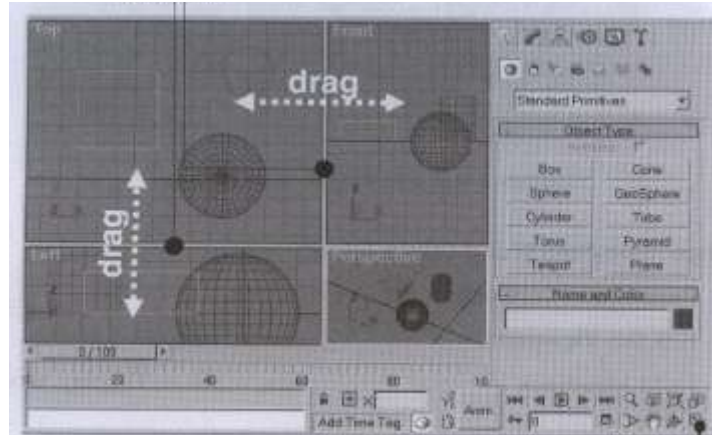
Viewport aktif dapat anda maksimalkan dengan cara :

1. Tekan W pada *keyboard*
2. Klik tombol *Min/Max Full Screen Toggle* yang berada di sudut kanan bawah

jendela *3DS Max*

Mengubah ukuran Viewport

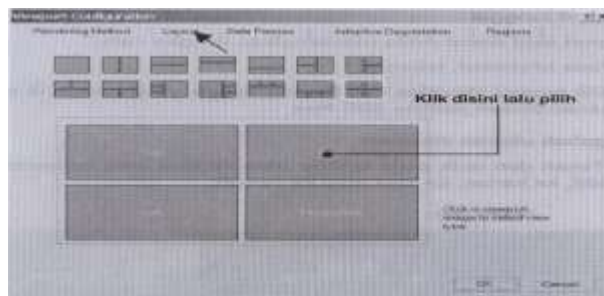
1. Tekan dan tarik garis splitter bars Vertikal atau Horizontal ke kiri, ke kanan, ke atas atau ke bawah



Gambar II. 16. Tampilan Layout
 (Sumber : Adhi Dharma Suriyanto; 2005: 5)

II.8.4. Mengganti jumlah Viewport

1. Klik kanan pada label *Viewport*, pilih *configure*
2. Pada kotak dialog *Viewport Configuration*, klik *tab layout*
3. Pilih salah satu dari 14 pilihan yang berbeda, lalu klik ok



Gambar II. 17. Tampilan konfigurasi Viewport
 (Sumber : Adhi Dharma Suriyanto; 2005: 6)

II.8.5. Viewport Control

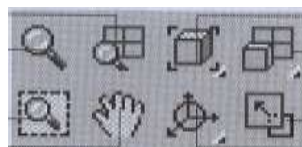
Terdapat sekelompok tombol di sudut kanan bawah jendela *3DS Max* yang berfungsi untuk mengatur tampilan pada *Viewport*. Beberapa tombol akan berubah secara otomatis ketika *Viewport* diganti menjadi *Viewport Camera* dan *Light*.



Gambar II. 18. Tampilan Viewport Control
 (Sumber : Adhi Dharma Suriyanto; 2005: 7)

II.8.6. Viewport Navigation Control

Navigation Control tergantung pada *Viewport* yang aktif. *Viewport* *Perspektive*, *Orthographic*, *Camera*, dan *Light* mempunyai pengontrol yang berbeda



Gambar II. 19. Tampilan Viewport Navigation Control
 (Sumber : Adhi Dharma Suriyanto; 2005: 7)

II.8.7. Menu Bar

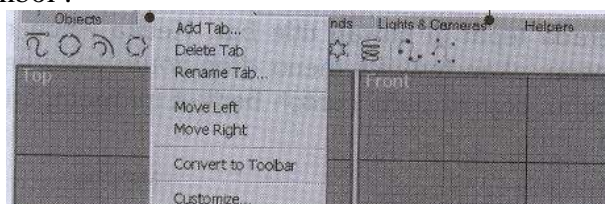
Tempatnya adalah tepat di bawah title bar, akan terbuka menu *Drop-down* yang berisi beberapa sub menu bila kita memilih salah satu menu .



Gambar II. 20. Tampilan Menu Bar
 (Sumber : Adhi Dharma Suriyanto; 2005: 8)

II.8.8. Main Toolbar

Terdiri dari beberapa tombol :



Gambar II. 21. Tampilan Main Tollbar
(Sumber : Adhi Dharma Suriyanto; 2005: 8)

II.8.9.Tab Panel

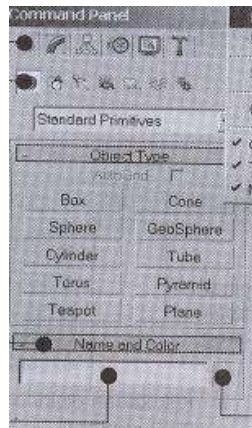
Cara untuk menampilkannya yaitu klik kanan pada bagian tepi/kosong *main toolbar*, lalu pilih *Tab Panel*.



Gambar II. 22. Tampilan Tab Panel
(Sumber : Adhi Dharma Suriyanto; 2005: 8)

II.8.10. Panel Command

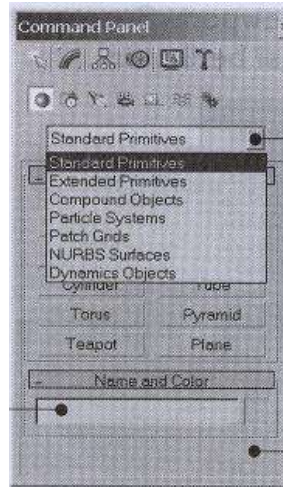
Terdiri dari beberapa panel, seperti *Panel Create*, *Modify*, *Hierarchy*, *Motion*, *Display* dan *utilites*.



Gambar II. 23. Tampilan Panel Command
(Sumber : Adhi Dharma Suriyanto; 2005: 9)

II.8.11. Geometry

Model *3D Max* dapat dibuat dari objek primitif atau objek 2D. Perintah tersebut merupakan sub perintah *Geometry*.



Gambar II. 24. Tampilan Geometry
(Sumber : Adhi Dharma Suriyanto; 2005: 9)

II.8.12. Panel Modify

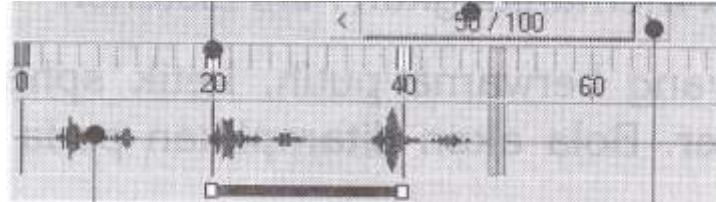
Digunakan untuk mengedit atau memodifikasi objek terpilih. Jika objek tidak terpilih maka panel ini tidak berisi perintah.



Gambar II. 25. Tampilan Panel Modify
(Sumber : Adhi Dharma Suriyanto; 2005: 10)

II.8.16. Track Bar

Posisinya ditempatkan di antara *Time Slider* dan *Status Bar*



Gambar II. 29. Tampilan Track Bar
(Sumber : Adhi Dharma Suriyanto; 2005: 12)