BAB II

LANDASAN TEORI

2.1 Pengertian Aplikasi

Aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media.

2.2 Pengertian Multimedia

Multimedia tersusun dari 2 unsur kata yaitu Multi dan Medium . Multi berarti banyak atau bermacam – macam sedangkan medium sesuatu yang dipakai untuk menyampaikan atau membawa sesuatu (alat untuk mendistribusikan dan mempresentasikan informasi) . Jadi Multimedia adalah penggunaan beberapa media yang berbeda untuk menggabungkan dan menyampaikan informasi dalam bentuk text, audio, grafik, animasi, dan video. Adapun definisi multimedia menurut para ahli :

1) Kombinasi dari komputer dan video.

2) Kombinasi dari tiga elemen: suara, gambar, dan teks.

3) Kombinasi dari paling sedikit dua media input atau output. Media ini dapat berupa audio (suara, musik), animasi, video, teks, grafik dan gambar.

4) Alat yang dapat menciptakan presentasi yang dinamis dan interaktif yang mengkombinasikan teks, grafik, animasi, audio dan video.

5) Multimedia dalam konteks komputer adalah: pemanfaatan komputer untuk membuat dan menggabungkan teks, grafik, audio, video, dengan menggunakan tool yang memungkinkan pemakai berinteraksi, berkreasi, dan berkomunikasi.

Multimedia memiliki beberapa kemampuan seperti :

1. Mengubah tempat kerja
2. Mengubah cara belanja
3. Mengubah cara bisnis
4. Mengubah cara memperoleh informasi
5. Mengubah cara belajar
6. Internet Multimedia juga mulai bersaing dengan televisi dan radio

2.2.1 *Storyboard*

Pentingnya sebuah *storyboard* dalam sebuah proyek berbasis multimedia. “Para ahli didalam bidangnya, biasanya menuangkan pengetahuan mereka, pengetahuan mereka membentuk suatu kreasi dari sebuah *stroryboard* yang

dimaksudkan untuk persentasi. Storyboard seperti itu seringkali menjadi suatu dasar bagi rencana pembuat dan director dalam pengimplemtasian dari cerita dengan masing – masing media. “

Pembuatan *stotyboard* dalam salah satu langkah – langkah *Interface design workflow*, sebagai representasi penggambaran bagaimana sebuah interface merespon pada interaksi pengguna dan sebagai penggambaran bagaimana sebuah *interface* merespon pada interaksi pengguna dan sebagai penggambaran dan identifikasi dari tiap tiap konten objek yang ada walaupun mereka beum dibuat ataupun dikembangkan. *Storyboard* juga berfungsi sebagai penunjuk fungsionalitas dari sebuah aplikasi.

Secara umum, *storyboard* adalah sekumpulan gambar yang terorganisisr secara berurutan dan dibuat untuk tujuan mem-previsualisasikan sebuah gambar gerak, animasi, ataupun urutan media interaktif, termasuk sebuah interaktif dari sebuah web.

2.3 Pengertian Game

Menurut Avedon dan Smith *games* adalah sebuah latihan system control sukarela, dimana ada sebuah kontes antara kekuatan, dibatasi oleh aturan – aturan dalam rangka memproduksi hasil yang tidak seimbang.

Yang pertama, sebuah latihan system control sukareala: itu adalah games sengaja dimasuki oleh para pemain.

Yang kedua, sebuah kontes antara kekuatan:bagian ini kelihatan menjadi bagian dari kebanyakan games. Dua atau lebih orang berusaha dengan keras untuk mendominasi sebuah game. Tapi tidak semua game mengalami ini, contohnya *single-player game* tidak perlu bersaing dengan kekuatan pemain lain, dan dari kalimat di atas, dapat disimpulkan dua hal, bahwa *games* mempunyai *goals*,dan *games* empunyai konflik.

Yang ketiga,dibatasi oleh aturan-aturan: sebuah poin yang sangat penting. *Games* mempunyai aturan-aturan, mainan tidak mempunyai aturan-aturan. Aturan sebenarnya adalah salah satu aspek yang mendefinisikan games.

Yang keempat, hasil yang tidak seimbang: tidak seimbang atau *disequilibrial* bukan berarti tidak sama, melainkan mengimpikasikan bahwa dalam satu waktu ada keadaan yang tidak seimbang, *games* dimulai dalam keadaan yang sama, lalu ada yang menang da nada yang kalah. Jadi jika kamu bermain *games*, kamu pasti menang atau sebaliknya kamu kalah.

Jadi poin penting dari sebuah *games* adalah *games* dimasuki secara sengaja oleh pemain, *games* mempunyai *goals*, *games* mempunyai konflik, games mempunyai aturan-aturan, dan *games* bisa berarti menang atau kalah.

2.3.1 Pengertian *Casual game*

Casual games adalah sebuah *games* yang dimaksudkan untuk dinikmati oleh setiap orang, baik pria dan wanita, maupun tua dan muda. Dengan ini *casual games* dapat diartikan sebuah *game* yang jauh dengan permainan hentakan,

kekerasan, dan kompleksitas permainan dengan bertujuan untuk menarik pemain seluas mungkin dari berbagai kalangan.

Perbedaan antara *casual gamer* dengan *hardcore gamer* menurut Schell adalah *casual gamer* hanya bermain game sebagai hiburan sesekali, namun *hardcore gamer* bermain *game* sebagai hobi utama mereka.

2.3.2 *Hardcore Game*

Hardcore gamer memperluas game ke dalam gaya hidup mereka dan dapat mewakili stereotip "Pecandu game". Mereka lebih sering memilih untuk menghabiskan waktu yang signifikan pada permainan, dan lebih cenderung untuk mencoba "menguasai" permainan tersebut dengan menyelesaikan sebanyak mungkin *Objective*. Dalam Hardcore sering mengadakan kontes – kontes turnamen. Sehingga penggemar *hardcore* game cenderung menghasbiskan waktu nya untuk menguasai game tersebut. Konten yang terdapat di dalam harcore game juga tidak semudah *casual game*, *hardcore* game lebih rumit untuk dimainkan

2.3.3 Game Genre

Menurut Rogers sebuah *game genre* digunakan untuk mendeskripsikan gaya dari suatu permainan. *Game genre* dibagi menjadi beberapa genre dan subgenre yakni:

a. Action

Action merupakan *game* yang dimainkan memerlukan kelihayan tangan dan mata untuk memainkannya.

b. Shooter

Shooter merupakan game yang berfokus pada penembakan peluru pada musuh. Genre ini termasuk dalam beberapa subgenre yang dibagi berdasarkan dari kamera penglihat.

1) First Person Shooter (FPS)

Merupakan game yang sudut pandangnya sama seperti yang terlihat dari sudut pandang pemain. Tampilan kamera lebih terbatas tapi lebih personal dari *third person shooter*.

2) Shoot em`up

Merupakan permainan dengan gaya arcade yang dimana pemain menembak musuh dalam jumlah yang besar dan menghindari bahaya.

3) Third Person Shooter (TPS)

Merupakan permainan menembak yang dimana kameranya diletakan dibelakang karakter.

c. Adventure

Adventure merupakan permainan yang berfokus pada penyelesaian teka-teki, mengumpulkan barang dan mengatur persedian. *Adventure game* pada awalnya hanya berbasis teks.

d. Contruction/Management

Contruction/Management merupakan permainan yang dimana pemain membangun dan memperluas lokasi dengn sumber yang terbatas. Contohnya SimCity dan ZooTycoon.

e. Life Simulation

Life Simulation merupakan permainan yang seperti Management tetapi lebih ke arah membangun dan memelihara hubungan denga kehidupan buatan. Contohnya The Sim dan Princess Maker

f. Music/Rhytm

Music/Rhytm merupakan permainan yang mencocokan irama atau mencetak nilai yang tertinggi. Contohnya Simon dan Rock Band.

g. Party

Party merupakan game yang didesain untuk permainan banyak orang dan didasarkan dengan persaingan. Sering kali game ini disajikan format game kecil. Contohnya Mario Party dan Buzz.

h. Puzzle

Puzzle merupakan permainan teka-teki yang didasarkan logika dan penyelesaian pola. Contohnya The Incredible machine atatu Tetris.

i. Sport

Sport merupakan permainan yang didasari dari kompetisi atletik, apakah itu tradisional atau extreme.

j. Strategy

Strategy merupakan game yang menekankan pemikiran dan perencanaan sebagai keunggulan game strategy.

1) Real Time Strategy (RTS)

   sama seperti *turn based games*, tapi game ini serba cepat dan berfokus pada " empat X`s" yakni ekspansi, eksporasi, ekploitasi dan pemusnahan.

2) Tower Defense

   Merupakan subgenre yang baru di PC dan handheld systems yang dimana pemain membuat tembakan peluru otomatis dari tower ke musuh yang muncul, dan mempertahankan suatu titik tertentu . Contohnya Defense Grid, The Awakening dan Lock`s Quest.

2.3.4 Gameplay Tower Defense

Tower Defense merupakan sebuah *sub-genre* dari permainan komputer *real-time strategy*[17]. Dalam permainan *tower defense* setiap pemain menggunakan media *mouse* untuk melakukan interaksi terhadap permainan tersebut.

Gameplay dari permainan dengan jenis *tower defense* adalah mencegah setiap musuh yang muncul untuk menuju *goal point*, dimana apabila musuh mencapai *goal point* tersebut maka darah atau nyawa dari pemain akan berkurang dan apabila habis maka pemain dinyatakan kalah. Setiap pemain dapat menempatkan *tower* untuk melakukan pertahanan dan secara otomatis akan menembak musuh yang berada di dalam jangkauannya, dalam beberapa permainan dengan jenis *tower defense* terdapat berbagai macam *tower* dengan harga yang beragam.

Adapun uang yang digunakan untuk membeli *tower* didapatkan dari hasil membunuh setiap musuh yang ada. Permaianan dinyatakan sukses apabila pemain dapat mempertahankan darah atau nyawa nya sampai waktu yang ditentukan habis atau berdasarkan jumlah musuh yang dibunuh.

2.3.5 Karakteristik *Tower Defense*

Karakteristik yang terdapat di dalam aplikasi permainan yang berjenis *tower defense* adalah sebagai berikut :

A. Terdiri dari dua *point*, yaitu *point* awal tempat keluarnya musuh dan *point* akhir yang merupakan tujuan akhir musuh
B. Adanya *shop* yang digunakan untuk membeli *tower*, sehingga *tower* yang bisa digunakan bervariasi
C. Metode interaksi antara pemain dan aplikasi permainan dengan menggunakan *mouse*
D. Terdapat nyawa dan jumlah koin pemain
E. Terdapat jalur yang telah ditentukan sehingga musuh berjalan melewati jalur tersebut

2.4 Dimensi Video Game

Game pasti memiliki dimensi dalam tampilannya adapun jenis dimensi sebagai berikut :

2.4.1 2D (*Two Dimension*)

Game dua dimensi dapat diketahui berdasarkan ruangnya yang hanya memiliki dua sisi (X dan Y). Sedangkan untuk gambarnya sendiri dapat menggunakan Vector maupun Bitmap. Untuk membuat animasi bergerak (berjalan,melompat,berlari,dll) kita harus membuat gambar satu persatu yang disebut dengan(frame). Kerealisasian gerakan ditentukan dari gambar yang dibuat, jumlah gambar(frame) yang digunakan, serta hitungan gambar per detik (frame per second (semakin tinggi hitungan gambar per detik maka semakin mulus gerakan yang akan dihasilkan)). Bagi pemula sebaiknya mencoba untuk membuat game bertipe 2D terlebih dahulu.

2.4.2 3D (*Three Dimension*)

3D ini merupakan model/ object 3 dimensi yang nantinya akan anda jadikan sebagai karakter utama, bangunan, object-object seperti senjata,musuh, permukaan tanah ,pohon, bukit,dll. Object 3d seperti ini bisa anda buat dengan menggunakan program seperti “AutoDesk 3DS.Max” ,”Maya”,”Hash”, “ Blender” .

2d graphic:gambar 2d juga berperan dalam membuat game bertipe 3D fungsi gambar 2d antara lain:

a) sebagai texture untuk object
b) sebagai latar belakang (langit , pemandangan,dll)
c) sebagai meteran untuk nyawa , gambar untuk speedometer (untuk game racing), dll.

Sedangkan untuk membuat animasi dalam 3D dapat digunakan bones(tulang) yang layaknya manusia dapat digerakkan. Selain itu animasi tetap juga dapat menggunakan bitmap (gambar 2d) untuk animasi pada texture air, api,dll.

2.5 Elemen Game

Setiap permainan digital memiliki elemen-elemen dasar, berupa:

1. Grafis

   yakni tampilan *game* tersebut di layar atau display alat elektronik, bisa berupa kumpulan pixel (satuan unit gambar terkecil) atau salah satu bagian gambar tersebut menunjukkan gerakan yang adalah hasil manipulasi si pemain.

2. *Interface*

   alat pengentara dimana pemain memanipulasikan *game* yang mengantarai dan memproses manipulasi pemain menjadi input dalam game, dapat berupa keyboard, mouse, joystick ataupun tampilan-tampilan tertentu dalam *game*

3. Aktivitas pemain

   yakni jantung hati dari *pengalaman* bermain game, tanpa aktivitas pemain maka tidak akan ada suatu permainan; aktivitas bisa bersifat *diegetic* (bagaimana karakter si pemain berubah aktivitas pemain) dan bersifat *extradiegetic* (apa yang terjadi secara fisik pada diri pemain tersebut).

4. Sebuah algoritme

   yakni jantung hati dari *program* game yang merupakan serangkaian program berisi otak dan seperangkat aturan yang mendasari dinamika game tersebut.

2.6 Kecerdasan Buatan *(Artificial Intelligence)*

2.6.1 Definis AI

Sebagian kalangan menerjemahkan *Artificial Intelligence* sebagai kecerdasan buatan, kecerdasan artifisial, intelijensia artifisial, atau intelejensia buatan. Pada bukunya Suyanto sengaja tidak menerjemahkan istilah *Artificial Intelligence* ke bahasa Indonesia karena istilah tersebut sudah sangat akrab bagi orang Indonesia. Begitu juga dengan singkatan istilah tersebut, yaitu AI, sudah sangat melekat di berbagai media ilmiah maupun non ilmiah. Definisi AI yang paling tepat untuk saat ini adalah *acting rationally* dengan pendekatan *rational agent*. Hal ini berdasarkan pemikiran bahwa komputer bisa melakukan penalaran secara logis dan juga bisa melakukan aksi secara rasional berdasarkan hasil penalaran tersebut.

2.6.2 Teknik Pemecahan Masalah AI

Terdapat empat teknik dasar pemecahan masalah yang terdapat pada bidang *academic/traditional artificial intelligence* ,diantaranya adalah sebagai berikut :

A. *Searching*

Pada teknik searching atau pencarian ini terdiri dari beberapa langkah untuk merealisasikannya. Langkah pertama adalah mendefinisikan ruang masalah untuk suatu masalah yang dihadapi. Langkah kedua adalah mendefinisikan aturan produksi yang digunakan untuk mengubah suatu keadaan ke keadaan lainnya. Langkah terakhir adalah memilih metode pencarian yang tepat sehingga dapat menemukan solusi terbaik.

B. *Reasoning*

Teknik reasoning atau penalaran merupakan teknik penyelesaian masalah dengan cara merepresentasikan masalah ke dalam basis pengetahuan (knowledge base) menggunakan logic atau bahasa formal (bahasa yang dipahami komputer). Teknik ini melakukan proses penalaran berdasarkan basis pengetahuannya untuk menemukan solusi.

***C**. Planning*

Planning adalah suatu metode penyelesaian masalah dengan cara memecah masalah ke dalam sub-sub masalah yang lebih kecil, menyelesaikan sub-sub masalah satu demi satu kemudian menggabungkan solusi-solusi dari sub-sub masalah tersebut menjadi sebuah solusi lengkap dengan tetap mengingat dan menangani interaksi yang terdapat pada sub-sub masalah tersebut.

***D**. Learning*

Pada ketiga teknik sebelumnya, seseorang harus mengetahui aturan yang berlaku untuk sistem yang akan dibangunnya. Tetapi, pada masalah tertentu terkadang suatu aturan tidak bisa didefinisikan secara benar ataupun lengkap. Hal tersebut mungkin dikarenakan data-data yang didapat tidak lengkap. Melalui teknik yang disebut learning ini, secara otomatis aturan yang diharapkan bisa berlaku umum untuk data-data yang belum pernah diketahui dapat ditemukan.

2.6.3 Waypoint

Waypoint search merupakan pencarian yang efisien dan menghemat komsumsi pada CPU-time. Hampir semua game sekarang ini menggunakan metode waypoint sebagai search.

Pola gerakan yang paling dasar untuk karakter dalam permainan hanyalah sebuah garis lurus baik dalam bidang horisontal atau vertikal. Biasanya, karakter akan bergerak ke kiri dan kanan antara dua titik dan batas mungkin tepi layar atau sesuatu yang berhubungan dengan latar belakang peta.

2.6.4 Metode Collision Detection

Hampir setiap permainan video yang ada sudah menerapkan collision detection (deteksi tabrakan), baik itu dalam hal tabrakan antara sprite dengan sprite maupun antara sprite dengan peluru dan lain-lain. Dalam aplikasi permainan ada beberapa reaksi yang ditunjukkan dari collision detection, yaitu reaksi dari spirte yang berupa berkurangnya nyawa, berkurangnya darah, meledak, dan lain-lain.

Proses *collision* dapat dibagi menjadi dua kategori dasar, yaitu *collision detection* dan *collision response*, dengan jarak respon yang telah diaplikasikan secara spesifik.

Selain itu pada sumber lain disebutkan bahwa *collision detection* terbagi menjadi dua jenis, yaitu *collision detection object to object* dan *collision detection object to world*. Pada *object to object* yang terjadi adalah adanya tubrukan antara satu objek dengan objek lainnya yang berada di dalam satu *frame*, baik itu objek tersebut bertubrukan secara langsung ataupun bertubrukan antara kotak maupun lingkaran yang mengandung objek tersebut, sedangkan pada *object to world* yang terjadi adalah adanya tubrukan antara objek dengan *frame* itu sendiri, contohnya adalah tubrukan antara *sprite* dengan dinding pada aplikasi permainan itu sendiri

sehingga apabila terjadi, maka *sprite* tersebut tidak dapat melewati dinding tersebut.

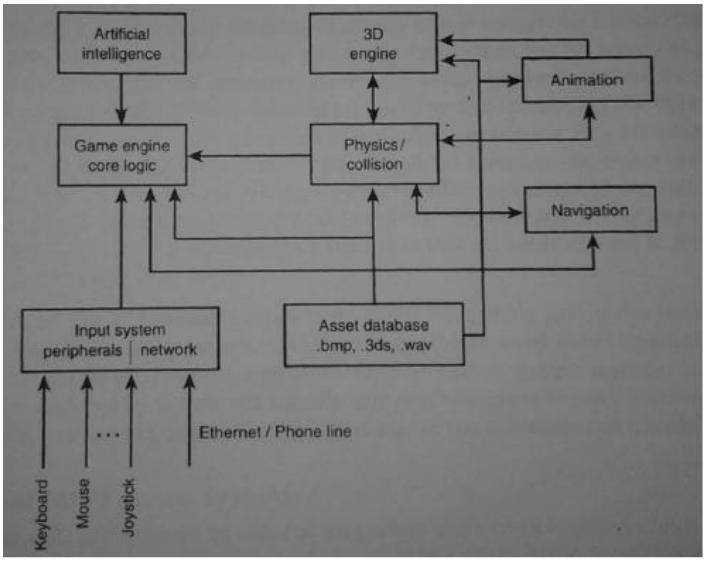
2.7 Game Engine

Menurut Andre LaMothe *Game engine* adalah sistem yang memproses segala sesuatu yang berhubungan dengan game, dan mengendalikan modul – modul yang mengirimkan perintah ke dalam sub-sistem, sehingga *game engine* harus dirancang agar dapat terhubung ke tiap masing-masing *sub*-sistem dalam *game*.

2.7.1 Struktur *Game Engine*

Menurut andre LaMothe tujuan utama dari *game engine* sendiri adalah untuk me-render 3D *game* itu sendiri. Artinya, *game engine* digunakan *game engine* itu sendiri harus terhubung dengan system *physics* dan *artificial intelligence.* Maka dari itu, sebuah *game* harus memiliki :

a. 3D *engine*
b. *Game Engine.*
c. Sistem *input dan networking*
d. Sistem animasi
e. *Collision detection* dan sistem navigasi
f. *Physics engine*
g. Sistem kecerdasan buatan
h. 3D Model dan *database*



Gambar 2.1 Struktur Umum *3D Game Engine*

(**Sumber** : Andre La Mothe, 2003,p495)

2.7.2 Unity Game Engine

Unity *game engine* merupakan sebuah *game engine* dan *editor*, yang memungkinkan pengembang membuat *object*, meng-import *external asset* dan menggabungkan semuanya dengan menggunakan kode secara efisien dan cepat. Prinsip dari Unity sendiri adalah agar pengembang dapat melakukan aktivitasnya dengan menggabungkan *script* dan membuat beberapa variable. Unity juga mendukung kemampuan *networking* dan dapat digunakan pada banyak platform.

Scripting di dalam Unity dapat menggunakan tiga bahasa pemrograman, yaitu JavaScript, C#, dan Boo. Pengembang *game* dapat menggunakan ketiga bahasa pemrograman tersebut dalam satu proyek miliknya. Beberapa konsep penting di dalam *scripting* yang harus pengembang mengerti adalah:

a. Function Update(){}

Merupakan container dari kode yang dieksekusi berkali-kali setiap detik (per – *frame*).

b. Transform

Merupakan *refence* untuk komponen *transform game object*, yang mendefinisikan posisi, rotasi, dan skala dari setiap *game object.* Setiap *object* memiliki *transform.*

c. Rotate()

Merupakan sebuah fungsi yang ada pada komponen *transform.*

d. Angka-angka yang berada di dalam koma merepresentasikan derajat dari rotasi untuk setiap axis di dalam 3D *space : X,Y,Z.*

e. Time.deltaTime

Merupakan anggota dari kelas Time yang berjalan pada setiap detik, sehingga sebuah *game object* dapat berputar dengan kecepatan yang sama seberapa pun *frame* per detik yang dapat dihasilkan setiap detik. Misalnya : 5 * Time.deltaTime artinya 5 derajat perdetik.