
Perancangan Permainan Komputer Boulder Thinks

Wilianto¹⁾

STMIK IBBI Medan

Jl. Sei Deli No. 18 Medan, Telp. 061-4567111 Fax. 061-4527548

e-mail: wilianto00@yahoo.com

Abstrak

Penelitian ini bertujuan untuk merancang perangkat lunak permainan computer (computer games). Computer games adalah sebuah program permainan dalam komputer yang dikendalikan oleh sang pemain dimana permainan ini disertai dengan tampilan grafik dan dukungan multimedia. Oleh karena itu, dirancang suatu permainan perangkat lunak yang diberi nama Boulder Thinks Games, dimana perancangan dilakukan dengan menggunakan metode tweening dengan teknik collision detection. Hasil dari perancangan adalah menghasilkan suatu perangkat lunak permainan Boulder Thinks yang bertujuan untuk mengasah logika seseorang dengan cara menggerakkan objek orang untuk mendapatkan serangkaian kata kunci untuk menyelesaikan misi. Perancangan tampilan program cukup sederhana sehingga dapat dimainkan dengan baik oleh orang tingkatan dewasa maupun anak-anak.

Kata kunci: Permainan Komputer , Boulder Thinks , Tweening, Collision Detection

Abstract

This study aims to design computer games software. Computer games is a game in the computer program that controlled by the player where the game is accompanied by a graphic display and multimedia support. Therefore, designed a software game called Boulder Thinks Games, where the design is done using tweening method with collision detection techniques. The result of the design is to produce a game software Boulder Thinks aimed to hone one's logic by moving the object to get a series of keywords to complete the mission. The design look fairly simple program that can be played well by the level of adults and children alike.

Keywords: Computer Games , Boulder Thinks , Tweening, Collision Detection

1. Pendahuluan

Sesuai dengan perkembangan teknologi dan ilmu pengetahuan dimasa sekarang ini, komputer merupakan salah satu alat yang didukung oleh teknologi canggih yang cukup banyak dipakai orang dalam kehidupan sehari-hari mulai dari pekerjaan kantor, desain grafik, pendidikan, multimedia sampai dengan tempat untuk bermain *games*. Salah satu kelebihan komputer adalah memiliki kecerdasan tiruan. Kecerdasan tiruan dapat didefinisikan sebagai cabang ilmu komputer yang mempelajari otomatisasi tingkah laku cerdas, yang menggunakan mesin cerdas untuk mendapatkan pengetahuan dan memecahkan masalah-masalah sulit secara cerdas. Masalah yang dipecahkan meliputi dua tipe yaitu masalah yang dapat dipecahkan dengan menggunakan prosedur *deterministik* yaitu yang menjamin keberhasilan yang disebut dengan *computation*, dan masalah yang diselesaikan dengan *non deterministik* yaitu pencarian dalam pemecahannya.

Permainan (*game*) adalah bagian dari penerapan kecerdasan tiruan yang menggunakan proses pencarian dalam menyelesaikan masalah penentuan langkah yang akan diambil. Proses pencarian tersebut dinamakan dengan pencarian ruang keadaan (*state space search*). Dimana dalam proses pencarian digunakan metode *heuristik* untuk mencapai penyelesaian yang diharapkan atau dapat diterima.

2. Landasan Teori

2.1. Grafika Komputer

Grafika komputer merupakan ilmu yang mengelola data dalam bentuk grafik. Grafika komputer berkaitan erat dengan titik grafik (piksel). Grafika komputer banyak melakukan hubungan yang bersifat *sintesis* yang mempunyai citra data masukan berbentuk *deskriptif* dengan keluaran hasil proses yang berbentuk gambar. Sebagai contoh adalah proses penggambaran *perspektif* suatu objek tiga dimensi, yang menggunakan data masukan berupa koordinat titik-titik yang berbentuk objek, kemudian melalui sistem perhitungan transformasi tiga dimensi maka dapat menghasilkan suatu gambar dengan objek yang berbentuk tiga dimensi sebagai hasil keluarannya.

Berikut ini grafika komputer dapat diterapkan pada beberapa bidang berikut :

1. Desain

Dalam proses desain, grafika komputer terutama digunakan pada sistem engineering dan arsitektur. Pada beberapa aplikasi desain, objek ditampilkan dalam bentuk wireframe, dimana diperlihatkan keseluruhan bentuk dengan bentuk internal dari objek tersebut. Secara aktraktif, desainer dapat melihat dengan cepat efek dari penyesuaian yang dilakukan.

2. Hiburan

Pada saat ini grafika komputer digunakan untuk menunjang pembuatan film, video musik, tayangan televisi dan permainan (games). Efek yang diperoleh dari komputer dapat digantikan kedalam efek manual pada pembuatan film secara konvensional. Pengolahan dengan komputer menghasilkan data digital.

3. Permainan (games)

Perancangan dan pembuatan hampir 90% semua dilakukan dengan menggunakan komputer melalui pengolahan grafika komputer untuk menghasilkan gambar dan animasi yang dapat bergerak dan dikendali oleh si pemain dengan peralatan masukan seperti gamespad, keyboard, joystick dan mouse.

4. Film

Pada pembuatan film layar lebar, komputer banyak digunakan untuk menunjang proses pemodelan, visualisasi dan editing. Misalnya film kartun jenis film anak-anak dengan animasi gambar yang dapat bergerak-gerak juga dapat dilakukan dengan baik dengan teknik grafika komputer.

5. Televisi

Televisi secara tetap menggunakan grafika komputer untuk menunjang tayangan yang disajikan. Mulai dari awal sampai dengan akhir siaran, dapat tampil animasi yang dihasilkan dari paket aplikasi grafika komputer. Grafika komputer dalam tayangan televisi dapat berupa film iklan, tampilan daftar acara televisi dan lain-lain. Pembuatan animasi atau iklan menggunakan perangkat lunak bantu mulai dari yang sederhana sampai dengan perangkat lunak yang mempunyai kemampuan baik yang dilengkapi dengan perangkat keras khusus.

6. Video Musik

7. Produksi video musik tidak lepas dari grafika komputer, diantaranya pembuatan promosi, cover atau kemasan video, serta animasi yang mengiringi setiap lagu.

8. Pendidikan

9. Komputer sebagai alat bantu pendidikan (Computer Assisted Instruction) sudah cukup dikenal, terutama di negara maju.

Berikut ini aplikasi bidang pengajaran dengan komputer sebagai alat bantu, diantaranya adalah :

1. Drill and Pratice (latih dan praktek) yang menggantikan pengajar untuk memberikan latihan kepada siswa.
2. Tutorial (penjelasan) yaitu sistem komputer yang digunakan untuk menyampaikan materi ajaran.
3. Simulation yaitu digunakan untuk mengkaji permasalahan yang rumit dan banyak digunakan dibidang teknologi, transportasi, ekonomi dan lain-lain.
4. Games (permainan) yaitu permainan yang sangat banyak digemari oleh anak-anak dan menambah ilmu pengetahuan.
5. Pengolahan Citra
6. Citra adalah representasi dari sebuah objek. Citra merupakan kumpulan dari titik-titik yang mempunyai intensitas tertentu membentuk satu kesatuan perpaduan yang mempunyai arti baik secara artistik maupun intristik.

Pengolahan citra (image citra) merupakan teknik untuk memodifikasi atau menginterpretasi gambar yang ada, seperti foto dan rangkaian gambar film. Ada dua macam prinsip pengolahan citra yaitu :

1. Meningkatkan kualitas gambar

Membersihkan persepsi dari informasi visual seperti pada robotic.

Sebagai contoh dalam proses pembuatan film animasi, objek dan proses animasinya diciptakan dengan teknik Grafika Komputer sedangkan pembuatan latar belakangnya dilakukan dengan teknik Pengolahan Citra. Disini latar belakang gambar dapat dibuat secara sederhana melalui kombinasi proses digitisasi suatu foto dan proses zooming atau scrolling atau dapat pula dengan teknik yang lebih rumit seperti pembuatan fraktal dan tekstur. Pembauran antara teknik Pengolahan Citra dan Grafika Komputer juga dapat dilihat pada proses penggabungan atau pembuatan. Contoh lain, pada proses pengenalan objek

yang terkandung pada suatu citra dimana diperlukannya proses segmentasi yang merupakan bagian dari teknik Pengolahan Citra dan proses pengenalan objeknya merupakan bagian dari teknik Visi Komputer.

2. Antar Muka

Saat ini telah biasa digunakan graphical interface (antar muka grafik) dalam setiap aplikasi. Komponen utama graphical interface adalah window manager, dimana pengguna dapat mengatur tampilan dari window. Graphical yang dibuat merupakan tampilan utama dari suatu perangkat lunak animasi pemrograman untuk mengakses menu tertentu.

2.2. Video Games

Video games terdiri dari kata ‘*video*’ dan ‘*games*’. *Video* dalam bahasa Yunani artinya adalah ‘melihat’. *Games* dalam bahasa Inggris yang artinya ‘permainan’. Jadi *video games* adalah suatu program yang bersifat permainan dengan memanfaatkan media tampilan seperti televisi atau *monitor* dan suatu alat kontrol untuk mengendalikan permainan

Video games merupakan salah satu permainan yang paling populer dimasa sekarang ini dimana perkembangan dari *video games* ini, begitu mengalami peningkatan yang sangat pesat hal ini seiring dengan perkembangan teknologi komputer dan permintaan akan kebutuhan pasar yang ada dimasa sekarang. Dalam era perkembangan permainan perangkat lunak (*video games*) dimulai dari jenis permainan (*games*) yang paling sederhana tahun 50 an sampai dengan jenis permainan (*games*) yang berteknologi sekarang ini yang berbasis multi media dan desain grafik yang sangat halus.

2.3. Pengenalan VGA Mode

Peragaan grafik terdiri atas unsur-unsur berupa titik yang disebut dengan *pixel* (*picture elements*). *Pixel* ini dapat digabung untuk membuat bentuk yang lebih sulit, seperti karakter, simbol, gambar dan lain-lain.

Tingkat rincian (*resolusi*) adalah peragaan grafik yang diukur dalam ukuran jumlah terbesar *pixel* horizontal dan vertikal. Resolusi adalah jumlah piksel dan warna yang terdapat pada layar *monitor*. Jumlah piksel perbaris layar yang terdapat pada layar disebut resolusi vertikal, sedangkan jumlah piksel perkolom layar disebut resolusi horizontal. Jumlah warna keseluruhan yang ditampilkan sekaligus dalam layar *monitor* disebut dengan warna dan secara teknis jumlah warna yang ditampilkan merupakan bagian dari resolusi dan tidak dapat dipisahkan.

Adapun beberapa mode grafik tersebut dapat dilihat pada Tabel 1.

Tabel 1. Tabel Mode Grafik

Resolusi	Jumlah Pilihan Warna
320 x 200	256 warna
640 x 400	256 warna
640 x 480	256 warna
800 x 600	256 warna
1024 x 768	256 warna
640 x 400	16 juta warna
640 x 480	16 juta warna

Resolusi erat kaitannya dengan *mode video* yang ditampilkan oleh *adapter card* yang digunakan. Mode grafik yang dapat ditampilkan oleh *card adapter* yang lebih rendah pasti dapat juga ditampilkan oleh *card adapter* yang lebih tinggi. Pada perancangan tugas akhir ini, mode grafik yang digunakan adalah resolusi mode grafik 800 x 600. 800 x 600 artinya jumlah titik horizontal ialah : 800 titik, sedangkan jumlah titik vertikal ialah 600 titik. Sehingga jumlah titik yang ada di layar 800 x 600 = 480000 titik. Semakin besar resolusi suatu mode grafik, semakin bagus citra yang ditampilkan dilayar karena ukuran titik semakin kecil. Dan semakin besar resolusi dan jumlah titik, semakin besar pula memori pada *VGA card* yang diperlukan.

2.4. Pemrograman Games

Dalam membuat suatu perangkat lunak permainan (*games*), pemrograman atau pengisian kode program merupakan salah satu langkah yang sangat penting karena jika pengisian kode program tidak berhasil maka perangkat lunak permainan (*games*) tidak akan tercipta.

2.4.1. Pemrograman Grafis

Kemampuan melakukan pemrograman grafis di dalam platform Windows merupakan suatu teknik baru. Mengingat platform Windows sendiri sudah menggunakan grafis sebagai tampilan untuk menampilkan objek visual ke layar seperti form, desktop, start, icon dan sebagainya. Lain halnya dengan saat pemrograman DOS yang hanya mengandalkan modus teks dan migrasi pemrograman grafis. Pada platform DOS memerlukan akses memori grafis secara langsung. Grafis dari Windows menggunakan hasil rancangan GUI untuk menampilkan hasil sebenarnya. Di dalam bahasa Visual Basic sudah disediakan berbagai rutin tambahan yang sanggup mengolah rutin untuk menggambar. Rutin untuk menggambar adalah kemampuan untuk menggambar garis, kotak, dan objek – objek grafis lainnya. Kemampuan tersebut tidak lepas dari pustaka Windows yang sudah disediakan.

Namun pemrograman grafis untuk akses rutin yang lebih sulit seperti Bit Bit tidak disediakan oleh bahasa Visual Basic. Tingkat kesulitan didasarkan pada kemampuan untuk merancang sebuah aplikasi animasi yang mulus, sulit dilakukan tanpa bantuan rutin tambahan. Rutin API merupakan salah satu jalan keluar untuk melakukan pemrograman grafis dengan menggunakan rutin yang disediakan. Salah satu pustaka pengolah grafis dari API tak lain adalah GDI.

2.4.2. Pemrograman GDI

Salah satu Windows API yaitu GDI menyediakan fungsi–fungsi dan struktur yang berhubungan dengan sebuah keluaran aplikasi untuk layar, printer, dan peralatan lainnya. Menggunakan fungsi GDI, pengguna dapat menggambar garis, kurva, figur tertutup, jalur, teks, dan gambar bitmap. Warna dan bentuk dari item yang digambar tergantung dari objek gambar yang diciptakan. Pengguna dapat menggunakan pen untuk menggambar garis dan kurva, kuas untuk mengisi interior tertutup dan fonts untuk menulis teks.

Aplikasi akan melakukan keluaran langsung ke peralatan tertentu dengan menciptakan sebuah device context (DC) untuk peralatan tersebut. Device context merupakan sebuah struktur GDI yang berisi informasi tentang alat tersebut, seperti modus operasi dan seleksi aktif. Sebuah aplikasi menciptakan sebuah DC dengan menggunakan fungsi device context. GDI mengembalikan sebuah kendali dari device context, yang dipakai untuk pemanggilan subsekuen untuk mengenal alat. Sebagai contoh, menggunakan handle, sebuah aplikasi dapat mengambil informasi tentang kapabilitas dari alat, seperti tipe teknologi (layar, printer atau alat lainnya) dan dimensi dan resolusi dari permukaan layar. Aplikasi dapat melakukan secara langsung keluaran ke alat fisik, seperti layar atau printer atau ke peralatan “logikal”, seperti memori atau metafile. Peralatan logika memberikan aplikasi maksud untuk menyimpan keluaran ke dalam form yang dengan mudah mengirim subsekuen ke peralatan fisik. Setelah catatan aplikasi keluar ke dalam sebuah metafile, maka metafile tersebut dapat dimainkan sebanyak mungkin, mengirim output ke peralatan fisik yang mana saja.

2.4.3. Pemrograman Musik dan Suara

Rutin – rutin API di dalam pustaka DLL “*Winmm.Dll*” merupakan rutin yang digunakan untuk mengolah citra suara dari *file* yang berekstension MIDI dan WAVE. Format *file* MIDI dan WAVE merupakan format yang paling umum digunakan di dalam *platform Windows* dan kapabilitas dan kompatibilitas dengan menggunakan rutin API telah dari dulu sudah digunakan. Format MIDI merupakan *file* yang berisi suara dengan *channel* sebanyak 16 buah dan menggunakan alat musik sebagai citra. Sedangkan WAVE merupakan kependekan dari *WAVEFORM* yang mampu menyimpan citra dalam format berbagai suara dengan kapasitas maksimum kejernihan suara sampai 44.100 Hz dan mempunyai dua *channel* atau stereo. *Citra Wave* merupakan citra paling umum yang digunakan untuk memainkan efek suara dikarenakan bentuk penyimpanan ke dalam *file* tersebut dapat menyimpan suara apapun.

MIDI merupakan singkatan dari *Musical Instrument Digital Interface* dan merupakan standarisasi *Microsoft* untuk data *audio* dengan ukuran yang kecil. Sebuah *file* MIDI memiliki Peta Saluran, Peta Tutupan dan Peta Kunci. Ketiga peta ini akan digunakan saat sebuah citra MIDI dijalankan dan merupakan bagian dari arsitektur yang ada. Sedangkan format *file* WAVE merupakan standarisasi dari format *RIFF (Resource Interchange File Format)*. Pustaka *DLL Winmm (Windows Multimedia)* menyimpan rutin yang dapat digunakan untuk mengolah citra multimedia WAVE maupun MIDI.

Adapun perintah standar yang terdapat di dalam perintah *string mcisendstring* dapat dilihat pada tabel 2 di bawah ini.

Tabel 2 Perintah *string* yang berlaku di *mcisendstring*

Perintah <i>string</i>	Deskripsi
Break	Menentukan tombol yang digunakan untuk membatalkan penggunaan <i>MCI</i>
<i>Close</i>	Menutup sebuah <i>file</i> multimedia dengan menggunakan alias atau peralatan <i>MCI</i>
<i>Copy</i>	Mengkopi data ke dalam papan klip (<i>clipboard</i>)
<i>Delete</i>	Menghapus segmen data dari <i>file</i>
<i>Info</i>	Mengambil deskripsi perangkat keras dari peralatan <i>MCI</i>
<i>List</i>	Menentukan <i>audio</i> dan <i>video</i> dari sebuah masukan <i>file</i> .
<i>Load</i>	Memuat sebuah <i>file audio</i> atau <i>video</i> .
<i>Mark</i>	Mengendalikan pelepasan atau penancapan dari posisi <i>file</i>
<i>Monitor</i>	Menentukan sumber presentasi
<i>Open</i>	Menginisialisasi sebuah <i>file</i>
<i>Pause</i>	Menghentikan sebuah <i>file</i> dari permainan atau perekaman
<i>Play</i>	Memulai memainkan sebuah <i>file</i>
<i>Put</i>	Menentukan daerah peletakkan sebuah <i>file</i> ke daerah tertentu
<i>Record</i>	Memulai perekaman data
<i>Restore</i>	Merekam sebuah gambar tak bergerak ke data <i>buffer</i>
<i>Resume</i>	Menyambung permainan sebuah <i>file</i> dari posisi berhenti
<i>Save</i>	Menyimpan sebuah <i>file MCI</i>
<i>Seek</i>	Menggerakkan posisi <i>file</i> ke suatu lokasi tertentu
<i>Setaudio</i>	Menentukan nilai asosiasi dari <i>audio</i> dan <i>video</i>
<i>Set</i>	Melakukan <i>setting</i> kendali terhadap peralatan
<i>Settimecode</i>	Melakukan penentuan kode waktu dari peralatan <i>VCR</i>
<i>Setvideo</i>	Menentukan nilai asosiasi dari pengambilan dan rekam <i>video</i>
<i>Status</i>	Meminta informasi status dari peralatan
<i>Spin</i>	Memulai perputaran sebuah <i>disc</i> dari posisi diam
<i>Step</i>	Menggerakkan sebuah <i>frame</i> ke depan atau belakang
<i>Stop</i>	Menghentikan perekaman atau permainan
<i>Sysinfo</i>	Mengambil sistem informasi <i>MCI</i>
<i>Update</i>	Melakukan refresh terhadap sebuah <i>device context</i>
<i>Where</i>	Mengambil informasi dari area sumber dan tujuan
<i>Window</i>	Mengendalikan tampilan jendela (<i>window</i>)

2.4.4. Teknik dalam Pemrograman Games

Perancangan dan pembuatan suatu program aplikasi *games* memerlukan banyak sekali prosedur dan teknik pemrograman. Dalam pemrograman *game* yang harus diperhatikan antara lain manajemen penggunaan memori, pengaturan resolusi layar, pengaturan *pallette* warna, penggunaan *Artificial Intelligensia (AI)*, spesifikasi perangkat keras (*hardware*) dan lain-lain. Maka tidak mengherankan jika dalam pembuatan *game* melibatkan suatu tim kerja (*team work*) yang handal yang biasanya terdiri dari pembuat program (*programmer*), perancang sistem (*system design*) atau perancang alur cerita permainan (*games story*), perancangan gambar (*picture animatian design*), perancang suara (*sound design*) dan penguji (*implementer*) serta pemasaran (*marketing*) *game*. Berikut ini penjelasan dari setiap teknik dalam pemrograman *games* antara lain :

1. Teknik Masking
 - a) Teknik ini merupakan cara meletakkan gambar diatas gambar lain untuk menghasilkan efek tertentu pada titik atau pixel gambar asli dengan menggunakan logika operator AND dan OR.
2. Teknik Scrolling
 - a) Teknik ini berbeda dengan teknik masking. Teknik scrolling membuat gambar latar belakang (*backgroud*) seakan-akan berubah. Misalnya animasi seekor burung yang sedang terbang dengan animasi pengepak sayap di awan dan tampak seakan-akan sedang terbang

dan bergerak, sebenarnya hanya gambar awan saja yang diganti. Ada dua jenis scrolling antara lain :

3. Horizontal scrolling dengan membuat gambar bergerak dari kiri ke kanan.
4. Vertical scrolling dengan membuat gambar bergerak dari atas ke bawah.

Adapun langkah-langkah teknik *scrolling* pada layar adalah sebagai berikut :

1. Pindahkan background page ke proses page
2. Ubah koordinat dan kondisi sprite
3. Letakkan sprite pada proses page
4. Pindahkan proses page ke visual page
5. Ulangi lagi langkah nomor satu sampai ketemu hasil gambar yang diinginkan.

2.4.5. Teknik *Collision Detection* (deteksi tabrakan)

Teknik deteksi tabrakan biasanya dipakai pada saat pemrograman animasi gambar. Teknik deteksi tabrakan adalah proses pemeriksaan apakah suatu *sprite* telah bertabrakan antara objek-objek dengan sesamanya. Pendeteksian dan pemrosesan hasil tabrakan antara objek-objek dalam permainan (*game*) menggunakan tipe '*Collision Bounding*' yang berbeda-beda.

Jenis dari *Collision Bounding* adalah sebagai berikut :

1. Pixel perfect .

Teknik yang diterapkan dengan memeriksa lokasi pixel dalam suatu sprite dengan objek apakah sprite tersebut menimpah objek atau tidak.

2. Circular .

Teknik deteksi yang paling mudah diantara semua teknik yang ada yaitu dengan cara merepresentasikan OffsetX dan OffsetY pusat dari lingkaran.

3. Rectangular .

Teknik yang paling banyak digunakan yaitu dengan memeriksa suatu objek yang mengalami interaksi pada posisi atas, bawah, kiri dan kanan.

4. Teknik Bouncing (pantulan)

Teknik pantulan jika dalam suatu objek yang bergerak mengenai objek lain dan objek tersebut memantul berbalik arah. Kemungkinan yang lain adalah objek mengenai suatu bidang datar yang menyebabkan objek tersebut harus memantul. Teknik pantulan ini memanfaatkan hukum fisika yaitu sudut datang objek sama dengan sudut pantul objek.

2.5. Animasi

Animasi merupakan manipulasi dengan menggunakan teknik-teknik khusus untuk menghasilkan gambar maupun objek yang bergerak dari suatu tempat ketempat yang lain. Animasi atau gambar yang bergerak juga merupakan salah satu bagian dari grafika komputer yang menghasilkan tampilan-tampilan yang kelihatan nyata. Misalnya pencerminan, pengeseran, pemutaran gambar dan lain-lain.

Untuk masa sekarang ini animasi dapat dan banyak dibuat melalui komputer. Melalui komputer suatu animasi dapat dimanfaatkan untuk membuat suatu perangkat lunak permainan (*games*) menjadi lebih menarik dengan cara memvisualisasikan cara kerja suatu alat atau mekanisme tertentu, menampilkan keluaran (*output*) dari program yang dirancang dengan gambar-gambar yang menarik dibanding dengan sederetan angka serta program lainnya. Untuk menghasilkan suatu animasi yang bagus dan menarik diperlukannya persiapan gambar yang harus mendukung. Dukungan gambar yang baik dalam animasi sangat berpengaruh, hal ini dikarenakan semakin banyak gambar yang disertakan dalam perancangan animasi maka gerak atau efek animasi yang dihasilkan juga semakin nyata (halus).

Pelaksanaan animasi dapat dilakukan dengan dua cara yaitu :

1. Pelaksanaan animasi secara interaktif

Pelaksanaan animasi dilakukan dengan menggerakkan objek atau gambar yang mengikuti perintah dan pemakai yang diberikan melalui perangkat interaktif. Misalnya keyboard dan mouse. Animasi ini banyak diterapkan pada perancangan dan pembuatan perangkat lunak permainan (*games*).

2. Pelaksanaan animasi secara non interaktif

Pelaksanaan animasi dilakukan dengan menggerakkan objek yang dikendali oleh prosedur dan function dalam program. Animasi ini banyak diterapkan pada perancangan dan pembuatan perangkat simulasi objek (*simulation*).

Untuk menghasilkan suatu animasi yang baik maka dalam perancangan animasi perlukanya metode-metode atau teknik-teknik tertentu. Adapun beberapa metode animasi terdiri dari :

1. Metode Animasi Tweening

Metode tweening dilakukan dengan menentukan posisi awal serta posisi akhir suatu objek yang akan dianimasi dengan menghitung posisi objek serta menghapus objek pada posisi yang baru sampai objek berada pada posisi akhir.

2. Metode Animasi dengan Menggerakkan Citra

Metode menggerakkan citra dilakukan dengan menggerakkan citra disekiling layar tampilan. Berbeda dengan metode tweening yang pada dasarnya dianimasi adalah titik sedangkan pada metode menggerakkan citra yang dianimasi adalah keseluruhan gambar yang ada. Pada metode animasi ini, menggerakkan citra dengan cara di sekeliling layar dengan mengikuti urutan operasi yang sudah diatur.

3. Animasi dengan Permainan Halaman

Pada animasi permainan halaman memanfaatkan halaman (page) yang dimiliki oleh sejumlah adapter grafik, meskipun tidak semua adapter grafik mendukung lebih dari sebuah halaman.

2.6. Peralatan Kontrol Games (*Input Device Game*)

Peralatan masukan (*input device*) adalah perangkat keras (*hardware*) fisik yang dipakai sebagai interaksi antara permainan (*games*) dengan pemain (*gamer*). Contoh peralatan *input device* yang dipakai dalam komputer adalah papan ketikan (*keyboard*) atau tikus petunjuk (*mouse*). Berikut ini adalah gambar papan ketikan (*keyboard*) dan tikus petunjuk (*mouse*). Semua peralatan *input device* mempunyai fungsi yang sama yaitu mengubah informasi yang disediakan oleh pemakai kedalam bentuk yang dapat dimengerti oleh komputer. Bentuk peralatan masukan penghubung antara pemain *games* dengan gamenya. Peralatan yang dikhususkan untuk bermain *games* saat ini adalah *keypad* dan *joystick* yang juga dapat dihubungkan dengan CPU komputer. Berikut ini contoh gambar *keypad* dan *joystick*. Dalam pemrograman *games* peralatan masukan seperti papan ketikan (*keyboard*) dan tikus petunjuk (*mouse*) sangat penting jika pemrograman *game* yang dirancang menggunakan *keyboard* atau *mouse* sebagai media kontrol permainan *game* yang akan dirancang. Seorang *programmer* harus membuat program untuk mengecek nilai kembalian dari *users* berdasarkan peralatan masukan yang dikirim melalui *keyboard* atau *mouse*. Metoda yang digunakan dalam mengecek nilai kembalian dari papan ketikan (*keyboard*) dan tikus petunjuk (*mouse*) adalah metode *event driven*.

Adapun metode *event driven* dari papan ketikan (*keyboard*) adalah :

1. Keypress : pada saat keyboard ditekan ke bawah lalu dilepas naik keatas.
2. KeyDown : pada saat keyboard ditekan kebawah.
3. KeyUp : pada saat keyboard dilepas naik keatas (setelah ditekan).

2.7. Artificial Intelligence

Artificial Intelligence disebut juga dengan kecerdasan buatan. Banyak dipakai untuk memberikan kecerdasan pada suatu benda sehingga benda tersebut menjadi cerdas seperti manusia sesuai dengan tingkat kecerdasan yang diberikan seperti komputer, robot dan lain-lain.

Artificial Intelligence (kecerdasan buatan) merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti sebaik yang dilakukan oleh manusia. Pada awal diciptakannya, komputer hanya difungsikan sebagai alat hitung saja, namun seiring dengan perkembangan jaman, maka peran komputer semakin mendominasi kehidupan umat manusia, lebih dari itu komputer diharapkan untuk diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan oleh manusia.

Agar komputer dapat bertindak seperti manusia maka komputer harus diberi bekal ilmu pengetahuan dan harus mempunyai kemampuan untuk menalar. Untuk itu pada *Artificial Intelligence*, akan mencoba memberikan beberapa metoda untuk membekali komputer dengan kedua komponen tersebut agar komputer dapat menjadi mesin yang pintar.

Untuk lebih detilnya, pengertian *Artificial Intelligence* dapat dipandang dari berbagai sudut pandang, antara lain :

1. Sudut pandang kecerdasan

Kecerdasan buatan akan membuat mesin menjadi 'cerdas' (mampu berbuat seperti apa yang dibuat oleh manusia)

2. Sudut pandang penelitian

Kecerdasan buatan adalah suatu studi bagaimana membuat agar komputer dapat melakukan sesuatu sebaik yang dilakukan oleh manusia.

3. Sudut pandang bisnis

4. Kecerdasan buatan adalah kumpulan peralatan yang sangat powerfull dan metodologis dalam menyelesaikan masalah-masalah bisnis.

5. Sudut pandang pemrograman

Kecerdasan buatan meliputi studi tentang pemrograman simbolik, penyelesaian masalah (problem solving) dan pencarian (searching).

Artificial Intelligence juga merupakan sub bidang pengetahuan komputer yang khusus ditujukan untuk membuat *software* dan *hardware* komputer sepenuhnya meniru otak manusia. Tujuan utama *Artificial Intelligence* adalah untuk mengubah komputer agar lebih pintar, lebih menalar dan lebih berguna untuk membantu pemakai dalam mengambil keputusan tanpa banyak campur tangan dari manusia. *Intelligence (intelegensia)* merupakan kepandaian manusia, yaitu kepandaian dalam melaksanakan pengetahuan yang dimiliki.

Kumpulan ilmu pengetahuan yang diberikan pada komputer dikenal dengan istilah basis pengetahuan. Basis pengetahuan dapat berupa kejadian, fakta dan prosedur tentang domain tertentu. Kemampuan komputer yang mencerminkan *Artificial Intelligence (AI)* adalah dalam melakukan inferensi (penarikan kesimpulan), dengan basis pengetahuan yang dimilikinya program *Artificial Intelligence (AI)* pada komputer dapat berpikir, menganalisa dan membuat pada fakta, teori dan prosedur dalam basis pengetahuan yang ada.

Dalam *Artificial Intelligence (AI)* digunakan beberapa metode dalam merepresentasikan pengetahuan yang ada antara lain :

[1] Logika (*logic*)

Logika adalah bentuk representasi pengetahuan yang paling tua. Pada dasarnya proses logika membentuk kesimpulan atau menarik suatu *inferensi* berdasarkan fakta yang telah ada. *Input* dari proses logika berupa *premis* atau fakta-fakta yang diakui kebenarannya sehingga akan melakukan penalaran pada proses logika dapat dibentuk suatu *inferensi* atau kesimpulan yang benar.

[2] Pohon (*tree*)

Pohon merupakan struktur penggambaran pohon secara hirarkis. Struktur pohon terdiri dari *node-node* yang menunjukkan objek dan busur yang menunjukkan hubungan antar objek.

[3] Jaringan semantik (*semantic net*)

Jaringan semantik merupakan gambaran pengetahuan grafis yang menunjukkan hubungan antar berbagai objek. Jaringan semantik terdiri-dari lingkaran-lingkaran yang menunjukkan objek dan informasi tentang objek-objek tersebut. Salah satu kelebihan dari jaringan semantik adalah 'bisa mewariskan'. Sistem jaringan semantik ini selalu tergantung pada jenis masalah yang akan dipecahkan, jika masalah itu bersifat umum maka hanya memerlukan sedikit rincian. Jika masalah itu banyak melibatkan hal-hal lain maka didalam jaringan diperlukannya penjelasan yang lebih terperinci lagi.

[4] Skema kerangka (*frame*)

Skema kerangka merupakan kumpulan pengetahuan tentang suatu objek tertentu, peristiwa, lokasi, situasi dan lain-lain. Skema kerangka memiliki slot yang menggambarkan rincian (*atribut*) dan karakteristik objek. Skema kerangka biasanya digunakan untuk merepresentasikan pengetahuan yang biasanya didasarkan pada karakteristik yang sudah dikenal yang merupakan pengalaman-pengalaman. Dengan menggunakan skema kerangka sangatlah mudah untuk membuat inferensi tentang objek, peristiwa atau situasi yang baru.

[5] Naskah (*script*)

Naskah adalah skema representasi pengetahuan yang sama dengan skema kerangka (*frame*) yaitu merepresentasikan ilmu pengetahuan berdasarkan karakteristik yang sudah dikenal sebagai pengalaman-pengalaman. Perbedaannya skema kerangka menggambarkan objek sedangkan naskah menggambarkan urutan peristiwa. Dalam menggambarkan urutan peristiwa naskah menggunakan *slot* yang berisi informasi tentang orang, objek dan tindakan-tindakan yang terjadi dalam suatu peristiwa

3. Pembahasan Dan Perancangan

3.1. Pembahasan

Permainan *Boulder Thinks* adalah suatu permainan logika yang berasal dari negara Inggris. Permainan ini diciptakan oleh seorang mahasiswa asal Inggris bernama David Boulder. Semula hanya berupa permainan logika yang cukup sederhana yaitu permainan menggeser batu untuk mendapatkan huruf-huruf dengan aturan tertentu sehingga menjadi serangkaian kalimat (*sentences keys*). Permainan ini dibuat di atas selembur papan yang berukuran 30 x 30 cm. Permainan ini pun diberi nama oleh si

penciptanya dengan mana *Boulder Thinks Games* yang berarti permainan cara berpikir Boulder. Kemudian permainan ini semakin meluas dan banyak disukai oleh orang, sehingga oleh beberapa orang tertentu permainan *Boulder Thinks* dikembangkan lagi sampai dengan sekarang.

Dalam pemrograman permainan perangkat lunak yang harus diperhatikan antara lain alur permainan atau ide dasar permainan (*base idea*), perancangan tampilan (*graphic*), pengaturan resolusi layar (*pixel colour*), penggunaan memori (*memory access*), dan spesifikasi perangkat keras (*hardware*) yang dibutuhkan. Perancangan permainan *Boulder Thinks* ini sebenarnya adalah sebuah perancangan animasi yang kelihatan benar-benar nyata tetapi sebenarnya hanya sebuah teknik pergerakan dan pemindahan objek. Animasi dapat dimanfaatkan untuk membuat suatu perangkat lunak permainan (*games*) menjadi lebih menarik dengan cara memvisualisasikan cara kerja suatu alat atau mekanisme tertentu, menampilkan keluaran (*output*) dari program yang dirancang dengan gambar-gambar yang menarik dibanding dengan sederetan angka serta program lainnya. Untuk menghasilkan suatu animasi yang bagus dan menarik diperlukannya persiapan gambar yang harus mendukung. Dukungan gambar yang baik dalam animasi sangat berpengaruh, hal ini dikarenakan semakin banyak gambar yang disertakan dalam perancangan animasi maka gerak atau efek animasi yang dihasilkan juga semakin nyata (halus).

Dalam permainan *Boulder Thinks* animasi merupakan salah satu teknik yang cukup berpengaruh agar permainan perangkat lunak nampak seperti hidup. Untuk menghasilkan animasi maka harus menggunakan metode animasi. Metode animasi yang digunakan dalam perancangan permainan *Boulder Thinks* adalah metode *tweening* yang menggerakkan objek-objek tertentu saja (sebagian gambar) mulai dari posisi awal sampai dengan posisi akhir dan menghapus sebagian objek-objek yang dilewati serta memindah beberapa objek tertentu juga sesuai dengan jalannya permainan. Sedangkan teknik pemrograman yang digunakan adalah teknik *collision detection* bertujuan untuk mendeteksi objek-objek yang bertabrakan.

Berdasarkan uraian pelaksanaan animasi diatas maka perancangan permainan *Boulder Thinks* menggunakan pelaksanaan animasi secara atraktif yang animasi gerakan digerakkan dengan menggunakan *input device* yaitu melalui *keyboard*. Cara memainkan permainan *boulder thinks* ini cukup sederhana, pemain tinggal mendorong huruf-huruf (ke kanan dan ke kiri) dengan menekan tombol panah pada keyboard. (tombol panah kiri dan tombol panah kanan). Sedangkan tombol panah atas dan tombol panah bawah hanya berfungsi untuk memindahkan objek orang menuju ke atas dan ke bawah. Adapun ketentuan permainan *Boulder Thinks* adalah terdiri dari :

1. Masukkan nama pemain dan lama waktu permainan (dalam satuan detik)
2. Lalu tekan tombol mulai untuk memulai permainan Boulder Thinks.
3. Pemain harus membuka jalan terlebih dahulu baru bisa menggeser objek.
4. Pemain dapat mengeser objek huruf dan objek batu ke arah kanan dan kiri.
5. Objek huruf dan objek batu tidak dapat digeser ke atas.
6. Objek huruf dan objek batu bisa digeser ke bawah dengan ketentuan pemain harus membuat lubang atau daratan yang lebih rendah agar objek huruf bisa jatuh ke bawah.
7. Pemain tidak dapat menggeser objek jika terdapat dua objek huruf atau dua objek batu sekaligus.
8. Objek huruf dan objek batu yang telah jatuh ke bawah tidak bisa dikembalikan ke atas lagi (seperti semula)
9. Kata yang disusun (dirangkai) pada masing-masing objek huruf harus sesuai dengan kata kunci yang telah ditetapkan baru permainan bisa selesai.
10. Jika lama waktu permainan telah habis dan permainan belum selesai maka permainan tidak dapat dilanjutkan, akan muncul pesan 'Waktu telah habis'.

3.2. Perancangan






3.2.1. Perancangan Objek

Permainan *Boulder Thinks* sebenarnya adalah sebuah permainan yang cukup sederhana yang menuntut logika seseorang bagaimana caranya untuk menggeser sebuah objek tertentu sehingga mendapat kata kunci yang diinginkan. Objek-objek yang digunakan didapat dengan cara merancang melalui program *Paint* pada Sistem *Windows*. Adapun objek yang dirancang terdiri dari :

1. Objek orang

Objek orang merupakan objek utama dalam permainan yang digunakan sebagai objek kontrol untuk mengendalikan permainan perangkat lunak. Objek ini berfungsi untuk membuka jalan, menggeser objek-objek lain sehingga terakhir didapat kata kunci yang diinginkan. Objek orang dirancang secara bertahap mulai dari tahap perancangan kepala (elipse objek), badan (pencil objek), tangan dan kaki (pencil objek)

baru yang terakhir adalah pemberian warna pada program *Paint* di *Windows*. Objek orang ini terdiri dari lima macam yaitu :

-  Objek orang yang dalam keadaan diam (tidak melakukan apa-apa)
-  Objek orang yang sedang bergerak menuju sebelah kanan.
-  Objek orang yang sedang bergerak menuju sebelah kiri.
-  Objek orang menggeser benda ke sebelah kanan (sedang bekerja).
-  Objek orang menggeser benda ke sebelah kiri (sedang bekerja).

2. Objek huruf

Objek yang digunakan sebagai kunci untuk menyusun serangkaian kata dalam permainan, jika huruf yang digeser (dirangkai) sesuai dengan kata kunci yang diinginkan maka permainan pun selesai. Objek huruf dirancang dengan menggunakan objek text pada program *Paint* di *Windows*. Berikut ini contoh salah satu objek huruf :



3. Objek Batu

Objek batu merupakan salah satu objek yang digunakan untuk menghalangi sang pemain dalam menggeser dengan menggunakan objek *text* pada program *Paint* di *Windows* dan mendapatkan kata kunci. Objek batu dirancang dengan menggunakan objek *circle* pada program *Paint* di *Windows*.



4. Objek Permata

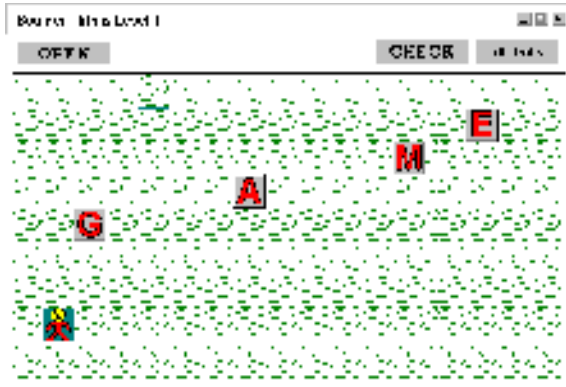
Objek batu merupakan salah satu objek yang digunakan untuk menghalangi sang pemain dalam menggeser dan mendapatkan kata kunci. Bedanya dengan objek batu objek permata

dapat diambil atau dimakan oleh objek orang. menggunakan objek *shape* atau objek *polygon* pada program *Paint* di *Windows*.

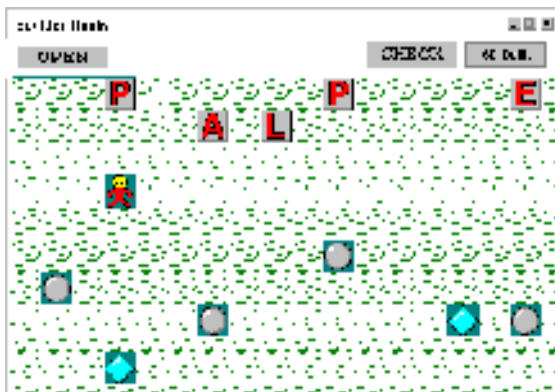


3.2.2. Perancangan Tampilan *Boulder Thinks*

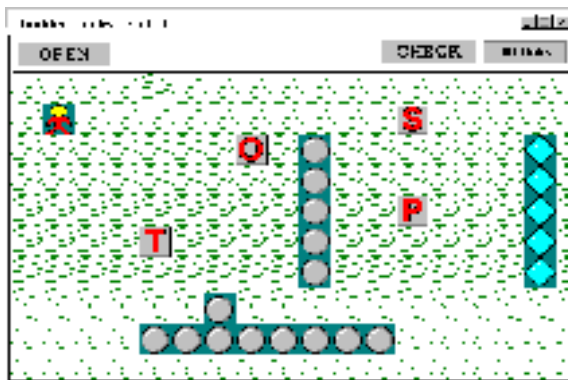
Jenis perancangan antar muka yang dilakukan untuk perangkat lunak adalah jenis antar muka penggunaan grafis (*Graphical User Interface*) memungkinkan adanya manipulasi langsung pada tampilan grafis di layar, yang dapat diselesaikan dengan memasukkan *keyboard* atau *mouse*. Manipulasi langsung ini membutuhkan lebih banyak pengguna orang-orang yang berpengalaman dengan sistem dari pada antar muka. Perancangan *Graphical User Interface* (GUI) dilakukan karena tampilan dari perangkat lunak adalah dalam bentuk grafik bukan tulisan yaitu tampilan objek yang siap untuk dimainkan dengan cara menekan tombol-tombol panah pada *keyboard* sehingga objek orang yang dalam permainan dapat dikendalikan sesuai dengan keinginan pemain. Perancangan antar muka dalam permainan ini terdiri dari lima level yaitu mulai dari level satu, dua, tiga, empat dan lima. Adapun perancangan tampilan dapat dilihat pada gambar berikut :



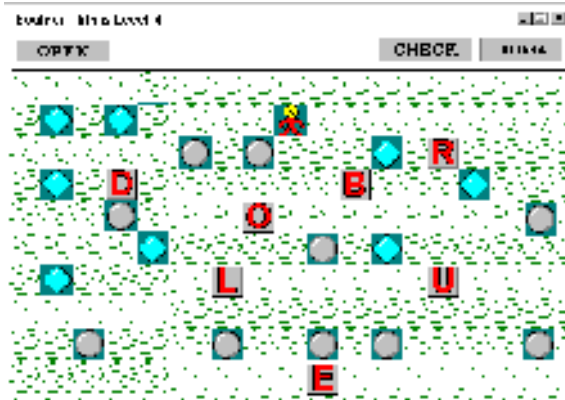
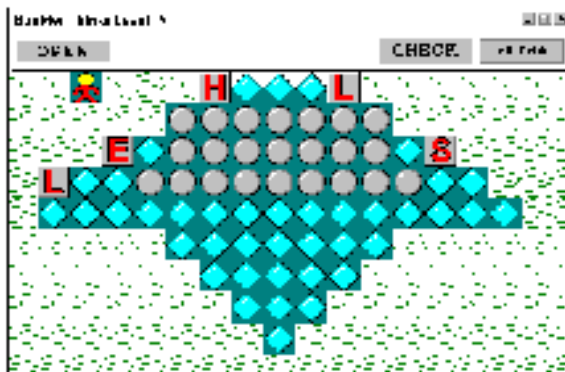
Gambar 1 : Rancangan Tampilan *Boulder Thinks* Level 1



Gambar 2 : Rancangan Tampilan *Boulder Thinks* Level 2



Gambar 3 : Rancangan Tampilan *Boulder Thinks* Level 3

Gambar 4 : Rancangan Tampilan *Boulder Thinks* Level 4Gambar 5 : Rancangan Tampilan *Boulder Thinks* Level 5

Gambar 6 : Rancangan Akhir Tampilan Program

3.2.3. Perancangan Level

Level dalam permainan *Boulder Thinks* juga dapat dibuat secara otomatis tanpa mengetikkan atau mengubah kode program yang ada, sehingga pemain dapat merancang secara langsung tampilan program sesuai dengan level atau tingkat kesulitan yang diinginkan. Hal ini dapat dilakukan karena permainan *Boulder Thinks* dilengkapi fasilitas layar editor. Layar editor adalah program bantuan yang dirancang khusus dengan bertujuan untuk merancang tampilan program secara langsung. Perancangan ini layar editor memanfaatkan objek yang terdiri dari : objek tanah (*lands*), objek huruf (26 objek huruf), objek batu, objek permata (*diamond*).

Berikut ini adalah tampilan perancangan layar editor *Boulder Thinks*.



Gambar 7 : Rancangan Layar Editor

Cara membuat level pertama-tama harus menentukan kata kunci permainan yang berada dibawah layar editor. Tahap selanjutnya adalah merancang tampilan permainan dengan cara dengan memilih objek-objek yang telah disediakan diatas layar editor. Misalnya objek batu, lalu tent objek yang hendak diletakkan pada layar dengan cara mengklik daerah yang akan dijad peletakkan objek batu. Demikian juga untuk objek lainnya berlaku sama. Jika proses peranc selesai maka simpan rancangan dengan cara menekan tombol *save level*.

5. Kesimpulan dan Saran

5.1. Kesimpulan

Kesimpulan yang dapat ditarik dari hasil perancangan perangkat lunak *Boulder Thinks* adalah sebagai berikut permainan *Boulder Thinks* merupakan permainan yang bertujuan untuk mengasah logika seseorang dengan cara menggerakkan objek orang untuk mendapatkan serangkaian kata kunci untuk menyelesaikan misi. Perancangan tampilan program cukup sederhana sehingga dapat dimainkan dengan baik oleh orang tingkatan dewasa maupun anak-anak.

5.2. Saran

Berikut ini saran yang diberikan terhadap perancangan perangkat lunak *Boulder Thinks* adalah Perancangan tampilan grafik *Boulder Thinks* di desain agar lebih bagus dan menarik lagi terutama pada gambar objek-objeknya dengan tampilan 3D (tiga dimensi) dengan menggunakan bantuan perangkat lunak khusus olah grafik seperti *Photo Shop*, *Corel Draw*, *Adobe Page Maker* dan lain-lain. Perancangan perangkat lunak permainan dikembangkan sampai dengan tahap jaringan (*networking*) sehingga dapat dimainkan secara *multiuser*.

Daftar Pustaka

- [1] Halvorson M, *Microsoft Visual Basic 6.0 Professional Step by Step*, PT. Elex Media Komputindo, Jakarta, 2000.
- [2] *Pemrograman Visual Basic 6.0*, Penerbit Andi, Yogyakarta, 2000.
- [3] Sanjaya R, *Membuat Menu Cantik untuk Aplikasi Visual Basic 6.0*, Penerbit Andi, Yogyakarta, 2005.
- [4] Evangelos P. Mastering Visual Basic 6. San Francisco, United States: SYBEX. 2000.
- [5] Kadir A. Pengenalan Sistem Informasi. Edisi-5. Cetakan Ke-5. Jogjakarta: Penerbit ANDI. 2001.
- [6] Kurniadi A. Pemrograman Microsoft Visual Basic 6.0. Jakarta : Penerbit Elex Media Komputindo.2000.
- [7] Hadi R, *Pemrograman Windows API dengan Microsoft Visual Basic*, PT. Elex Media Komputindo, Jakarta, 2001

[4] <http://www.rookscape.com/vbgaming/tutorials.php>, 3 Juni 2007.

[5] <http://www.vbgaming.com>, 3 Juni 2007.
