BAB 2

TINJAUAN PUSTAKA

2.1 Permainan Video

2.1.1 Definisi

Permainan video atau dalam istilah asing sering disebut ‘video game’ merupakan permainan elektronik yang melibatkan interaksi antara manusia dengan *interface* tertentu yang kemudian hasilnya ditampilkan secara langsung pada perangkat video [TGA72]. Kata video dalam istilah permainan video sebenarnya merujuk secara tradisional pada perangkat display berbasis raster, namun kini istilah itu pun berlaku pada perangkat display apa saja, bahkan yang tidak berbasis raster sekalipun.

Sistem perangkat elektronis yang digunakan untuk menjalankan permainan video biasa disebut dengan *platform.* Secara garis besar, *platform* terbagi menjadi dua, yaitu komputer personal dan konsol permainan video. *Platform* tersebut sangatlah bervariasi, mulai dari yang berukuran sangat besar seperti komputer *mainframe* sampai dengan yang berukuran kecil seperti *handheld device* ‘perangkat genggam’.

Alat *input* ‘masukan’ yang digunakan untuk memanipulasi permainan video disebut *game controller* dan jenisnya pun bermacam-macam dari setiap *platform.* Jenis kontroler tersebut mulai dari *joystick* sederhana, *keyboard* atau bahkan sampai dengan sensor gerak. Pada masa-masa awal, alat *input* untuk permainan

video yang berjalan pada komputer personal hanyalah *keyboard* saja, tapi kini *mouse* pun bisa digunakan. Lebih lagi, kini berbagai jenis *game controller* pun banyak bermunculan seperti *joystick* nirkabel, dan perangkat yang dapat menerima sensor gerak.

Permainan video memberdayakan berbagai hal pendukung untuk menyediakan interaktifitas dan informasi bagi para pemainnya. Hal tersebut diantaranya adalah audio yang tentunya dihasilkan oleh bantuan perangkat lain seperti *speaker* atau *headphones.* Selain itu, penggunaan *vibrator* 'penggetar' juga diterapkan dalam beberapa jenis *platform.* Fungsi *vibrator* tersebut adalah untuk merangsang interaktifitas pemainnya sehingga *game experience* yang mereka rasakan pun meningkat.

2.1.2 Sejarah dan Perkembangan

Sejarah permainan video jauh dimulai pada sekitar tahun 1940-an, tepatnya pada tahun 1947. Ketika itu, Thomas T. Goldsmith, Jr. dan Estle Ray Mann mengajukan permohonan kepada *United States Patent* untuk mempatenkan pemuan mereka yang diberi nama "*cathode ray tube amusement device*". Setelah itu, permainan video tidak mencapai ketenaran hingga tahun 1970 dan 1980-an, yaitu masa dimana permainan *arcade*, permainan komputer, dan konsol permainan mulai diperkenalkan pada khalayak luas. Sejak saat itulah permainan video akhirnya menjadi sarana hiburan dan bagian dari budaya modern.

2.1.2.1 Era Awal

Pada masa ini permainan video baru diciptakan, dan *platform* yang digunakan untuk menjalankannya pun tidak terlalu beragam. Sebagian besar

permainan video yang diciptakan merupakan proyek “iseng” para ilmuwan. Sejak dulu manusia memang ingin *having fun* dengan teknologi dan itu terbukti dengan proyek itu. *Platform* yang digunakan kebanyakan adalah komputer *mainframe* karena pada masa tersebut hanya itulah alat yang memungkinkan untuk digunakan dalam proses pengembangan maupun pembuatannya.

Berdasarkan literatur yang ada, didapatkan bahwa sejarah permainan video dimulai pada tanggal 25 Januari 1947. Pada hari itu Thomas T. Goldsmith Jr. dan Eastle Ray Mann mendaftarkan hasil penemuan mereka yang diberi nama “*chatode ray amusement device*” ke United States Pattent Office. Paten tersebut kemudian diresmikan pada tanggal 14 Desember 1948. Adapun isinya mendeskripsikan tentang penemuan tersebut yaitu sebuah mesin yang difungsikan melalui sejumlah kenop dan tombol untuk memanipulasi sinar katoda dan mensimulasikan penembakan terhadap objek yang sedang terbang di angkasa.

Pada tahun 1945-1950, Charley Adama manciptakan “*Bouncing Ball”* pada komputer *Whirlwind*-nya MIT. Pada mulanya, program tersebut belum begitu interaktif, namun program inilah yang merupakan cikal bakal bagi program-permainan-video selanjutnya pada masa ini.

Di bulan februari 1951, Christopher Strachey mencoba menjalankan program permainan *draughts* ‘dam-das’ yang ia bangun untuk NPL Pilot ACE. Program ini ternyata membutuhkan memori yang melampaui kapasitas memori mesin yang akan digunakannya. Pada akhirnya, Strachey menyimpan programnya untuk dijalankan kemudian pada mesin lain yang berada di Manchester dan mesin ini tentunya memiliki kapasitas memori yang lebih besar dari mesin sebelumnya.

Penambahan kapasitas memori pada mesin tersebut dilakukan hingga bulan Oktober.

Pada tahun 1951 juga seorang penemu, Ralph Baer, mengemukakan ide cerdasnya. Ketika itu ia sedang mengembangkan teknologi televisi untuk perusahaan elektronik New York, Loral. Idenya adalah untuk memanfaatkan sekumpulan cahaya dan pola-pola yang diberdayakan pada pekerjaannya itu bukan sekedar untuk peralatan kalibrasi. Ia menyadari bahwa cahaya dan pola-pola yang ada tersebut jika dimanipulasi dengan sedemikian rupa, maka akan dapat menghasilkan suatu interaktifitas dengan penontonnya, sehingga nantinya cahaya dan pola-pola tersebut tidak menjadi sekedar tontonan pasif saja. Sayangnya saat ia mengajukan ide tersebut kepada petinggi perusahaan, idenya ditolak karena proyek tersebut sudah melampaui jadwal.

OXO yaitu versi grafis dari permainan *tic-tac-toe* diciptakan oleh A.S. Douglas pada tahun 1952 di Universitas Cambridge. Adapun aplikasi tersebut merupakan hasil dari tesisnya mengenai interaksi manusia dengan komputer. Aplikasi tersebut dikembangkan dengan komputer EDSAC yang menggunakan tabung sinar katoda sebagai layar visualnya untuk manampilkan konten memory. Saat itu penggunanya bermain melawan komputer.

Di tahun 1958, William Highbotham menciptakan permainan dengan menggunakan osiloskop dan komputer analog. Permainan tersebut diberi nama ”*Tennis for Two*” dan berhasil menghibur para pengunjung di Laboratorium Nasional Brookheaven ketika itu. Permainan ini menampilkan penyederhanaan dari lapangan tenis yang terlihat dari samping. Adapun untuk memainkannya yaitu

dengan memantulkan bola yang terpengaruhi grafitasi sehingga ia harus melewati jaring. Alat input permainan ini berbentuk kotak yang padanya terdapat sebuah kenop yang berfungsi untuk memukul bola tenis. Permainan ini dipamerkan selama dua musim hingga tahun 1959.

1. Akhir 1950-1969

Sebagian besar permainan komputer pada era awal dibangun dan dijalankan pada komputer *mainframe* milik universitas dan dikembangkan oleh individu sebagai hobi. Keterbatasan akses pada perangkat komputer pada saat itu mengakibatkan banyak program permainan yang terlupakan bahkan tidak dikenal sama sekali pada masa kini.

Pada rentan waktu 1959-1961, sekumpulan program interaktif berbasis grafis diciptakan pada mesin TX-0-nya MIT, yang diantaranya adalah:

a. *Mouse in the Maze*: pada permainan ini, pemaian dapat menyimpan dinding, potongan keju, dan bahkan martini pada layar dengan menggunakan *light pen*. Setelah itu, pemain dapat melepaskan seekor tikus yang kemudian akan berusaha melewati diding yang ditempatkan untuk mencari barang-barang yang disimpan sebelumnya pada layar.

b. HAX: Pemain berinteraksi dengan sebuah tombol yang kemudian program akan bereaksi dengan mengeluarkan input berupa suara-suara dan warna layar yang berubah-ubah.

c. *Tic-Tac-Toe*: Dengan menggunakan *light pen*, pengguna dapat memainkan permainan, *tic-tac-toe*, yang sederhana ini melawan komputer.

Pada tahun 1961, sekelompok mahasiswa MIT yang salah satu diantaranya adalah Steve Russel memprogram permainan yang diberi nama *Spacewar!* pada komputer DEC PDP-1— komputer terbaru pada saat itu. Dalam program ini, pemain yang terlibat terdiri dari dua orang. Masing-masing dari mereka mengendalikan pesawat antariksa yang dapat menembakkan misil, sementara itu ditengah layar terdapat satu rintangan besar sebagai pembatas yang berbentuk bintang. Permainan ini didistribusikan dengan komputer DEC terbaru dan dipublikasikan melalui internet primitif saat itu. Pada akhirnya, *spacewar!* diakui sebagai permainan-komputer pertama yang paling berpengaruh.

Di tahun 1966, Ralph Baer merekrut Bill Harrison dalam proyeknya yang didanai oleh kontraktor peralatan militer *Sanders Associates* di Nashua, New Hemisphere. Mereka menciptakan permainan video sederhana berjudul *Chase* yang merupakan permainan pertama yang dapat dijalankan melalui layar televisi standar. Dengan bantuan Baer, Bill pun berhasil menciptakan *light gun*. Baer dan Harrison dipertemukan oleh Bill Rusch pada 1967, seorang lulusan MIT bergelar M.S.EE yang kemudian berhasil mempatenkan alat-alat temuan mereka yaitu untuk peralatan permainan televisi. Pengembangan pun terus berlanjut hingga mereka berhasil menciptakan purwarupa yang dapat menjalankan beberapa permainan seperti tenis meja dan *target shooting*.

Pada tahun 1969, pemrogram komputer AT&T Ken Thompson menciptakan permainan video yang berjudul *Space Travel* untuk sistem operasi *Multics*. Permainan ini mensimulasikan berbagai macam lingkungan sistem tata surya dan kemudian pemain harus bisa mendaratkan pesawat padanya.

Sayangnya, AT&T menghentikan proyek ini sehingga Thompson menulis ulang program ini dalam FORTRAN untuk supaya bisa dijalankan dalam Sistem Operasi GECOS. Sistem operasi tersebut merupakan sistem operasi bagi komputer *mainframe*-nya *General Electric* GE 635. Saat program berjalan pada komputer tersebut ternyata menghabiskan biaya yang tinggi yaitu sekitar $75.

Dikarenakan biaya operasional yang relatif tinggi pada saat itu akhirnya Thompson beralih untuk menggunakan komputer lain. Ia kemudian memilih komputer *mainframe* yang tidak digunakan secara optimal yaitu PDP-7. Dibantu oleh Dennis Ritchie kemudian ia mengkonversi permainan *Space Travel* ke bahasa assembler untuk PDP-7. Dalam perjalanannya saat itu ketika mempelajari proses pegembangan perangkat lunak untuk mesin, proses pengembangan sistem operasi UNIX pun dimulai. Pada akhirnya *Space Travel* pun disebut-sebut sebagai aplikasi UNIX pertama.

2.1.2.2 Era 1970-an

Pada era ini, sejarah perkembangan permainan video terbagi menjadi tujuh bagian berdasarkan fenomena yang terjadi.

1. Era Awal Permainan *Arcade* (1971-1977)

Pada bulan September 1971, sebuah program permainan berjudul *Galaxy Game* dipamerkan saat Pertemuan Mahasiswa di Universitas Stanford. Berbasis pada permainan sebelumnya yaitu *Spacewar!* permainan ini merupakan permainan video pertama yang dioperasikan dengan koin. Sistem yang dibuat hanya satu yaitu menggunakan mesin DEC PDP-11 dan terminal vektor display.

Kemudian pada 1972, program permainan beserta sistemnya ini dikembangkan lebih lanjut sehingga dapat menangani empat hingga delapan konsol.

Selain itu juga, pada tahun 1971 ini Nolan Bushnell dan Ted Dabney (pendiri Atari) menciptakan *Spacewar!* dalam versi *arcade*. Versi tersebut tentunya diopersikan dengan koin dan diberi judul *Computer Space*. *Nutting Associates* kemudian membelinya dan membangun 1500 mesin *Computer Space* yang dirilis pada bulan November tahun 1971. Ternyata permainan ini tidak sukses sesuai yang diharapakan dikarenakan sulitnya memainaknnya karena sistem input yang sulit. Walaupun demikian, permainan ini menjadi monumental karena merupakan permainan video pertama yang diproduksi secara masal dan ditawarkan kepada khalayak umum.

Adapun kejadian lainnya yang memengaruhi sejarah perkembangan permainan video pada era ini diantaranya adalah:

2. Konsol Generasi I (1972-1977)

Sistem konsol “rumahan” pertama dikembangkan oleh Ralph Baer yang dibantu dengan kolega-koleganya. Pengembangannya dimulai sekitar tahun 1966 yang kemudian menghasilkan *prototype* ‘purwarupa’ yang dinamai *Brown Box* dan sudah dapat dioperasikan pada tahun 1968. Kemudian purwarupa itu pun didemonstrasikan kepada beberapa calon investor potensial seperti GE, Sylvania, RCA, Philco dan Sears, namun pada akhirnya Magnavox lah yang menjadi pilihan.

Dengan kerjasamanya bersama Magnavox, konsol tersebut pun dipasarkan dengan nama *Magnavox Odyssey*. *Odyssey* menggunakan media *catridge* untuk

menyimpan permainan-permainan yang dapat dijalankannya. Hal tersebut merupakan inovasi pada saat itu karena satu konsol dapat menjalankan bermacam-macam permainan dengan sebuah sistem yang sama. Adapun untuk menjalankan konsol ini dibutuhkan media televisi untuk menampilkan grafisnya. Pada tahun pertama saja, konsol ini terjual sebanyak 100.000 unit dan salah satu penyebabnya adalah karena pemasaran melalui iklan televisi yang dibintangi oleh Frank Sinatra. Konsol tersebut dapat dilihat pada gambar di bawah ini:



Gambar 2.1 Konsol *Magnavox Odyssey*

Pada tahun 1974, Philips membeli Magnavox dan merilis bermacam-macam permainan untuk konsumen eropa. Di Eropa merk dagang konsol ini masih *Odyssey*, walaupun pemilik sahamnya telah berganti. Hal itu dikarenakan ketenaran nama *Odyssey* yang bahkan sampai ke benua Eropa. Jumlah penjualan *Odyssey* diperkirakan mencapai 2 juta unit dari awal sampai akhir masanya.

3. Komputer *Mainframe*

Pengembangan permainan video dengan komputer *mainframe*, yang sebagian besar merupakan milik universitas, tumbuh subur pada awal tahun 1970-an. Berdasarkan hasil studi literatur, hanya sedikit informasi yang didapat

mengenai permainan yang tenar pada saat itu. Hal itu mungkin disebabkan karena pengembangan permainan video yang dilakukan hanya untuk tujuan penelitian atau hobi, bukan untuk tujuan komersil. Kebanyakan orang yang terlibatnya pun adalah para mahasiswa dan mereka melakukannya secara sembunyi-sembunyi. Itu dikarenakan adanya peraturan keras mengenai aturan penggunaan komputer *mainframe* yang notabene membutuhkan sumber daya besar dan mahal, pada saat itu, dalam hal pengopersiannya.

Pada saat itu ada dua jalur distribusi dalam pengembangan permainan video pada *mainframe* komputer:

a. PLATO *Systems*

   Sistem ini merupakan lingkungan komputasi bersifat edukasional hasil perancangan University of Illinois yang berjalan pada komputer *mainframe* buatan *Control Data Corporation*. Permainan video yang dikembangkan dapat dijalankan pada berbagai PLATO *systems* yang berbeda.

b. DECUS

   DECUS digunakan oleh kelompok pengguna komputer yang diproduksi oleh DEC (*Digital Equipment Corporation*). Sistem ini mendistribusikan program, termasuk permainan video, yang dapat dijalankan pada berbagai jenis komputer produksi DEC.

Pada masa ini pun beberapa permainan video dikembangkan pada *minicomputer* keluaran Hewlett-Packard seperti HP2000.

Berikut ini merupakan kejadian penting yang disusun secara kronologis berkaitan dengan pengembangan permainan video pada komputer *mainframe*:

a. 1971: Don Daglow membangun permainan kasti berbasis komputer pertama pada komputer *mainframe* milik Pomona College, DEC PDP-10. Pemain dapat mengatur permainan secara individu atau mensimulasikan satu musim penuh liga kasti. Kemudian, Daglow membangun permainan berjudul *Earl Wave Baseball* dibantu oleh sebuah tim yang salah satu personilnya adalah Eddi Dombower sebagai pemrogramnya. Permainan ini dirilis oleh Electronic Arts pada tahun 1987.
b. 1971: *Star Trek* dibangun pada Sigma 7 yang merupakan *minicomputer* milik University of California. Dari seluruh seri *Star Trek* yang ada, seri inilah yang paling terkenal dan paling banyak dimainkan. Permainan ini merupakan permainan video pertama yang bisa berjalan pada *platform* perangkat keras apapun. Daglow juga membangun permainan *Star Trek* ini untuk PDP-10 pada sekitar tahun 1970-1972 yang menyajikan percakapan dari suara asli karakter film serinya. Beberapa versi lain permainan ini pun dikembangkan untuk PLATO dan DECUS sepanjang dekade ini.
c. 1972: Gregory Yob membangun permainan “petak umpet” berbasis komputer yang berjudul *Hunt the Wumpus* pada PDP-10 yang diyakini sebagai permainan berjenis *adventure* ‘petualangan’ berbasis teks pertama. Yob membangun permainan ini sebagai reaksi terhadap

bermunculannya permainan “petak umpet” pada masa itu seperti *Hurkle*, *Mugwump*, dan *Snark*.

d. 1974: Permainan *multi-player* 3D berjenis *first-person shooter* pertama diciptakan. Adapun permainan tersebut berjudul *Maze War* yang berjalan pada Imlac PDS-1 milik NASA Ames Research Center di Kalifornia. Selain itu ada pula permainan jenis ini yang berjalan pada PLATO dan dikenal dengan judul *Spasim.*

e. 1974: Brand Fortner bersama rekan-rekannya mengembangkan *Airflight* yaitu sebuah simulator pelatihan pesawat terbang. Untuk membuat simulasi ini menarik, pemain diberi pilihan untuk mengendalikan beberapa jenis pesawat mulai dari pesawat antariksa sampai pesawat jet militer. Pemain dapat mengkustomisasi jenis senjata, jumlah amunisi, dan muatan bahan bakarnya. Ketika permainan dimulai, pemain berada pada sebuah area tertentu yang juga ditempati pemain lainnya, sehingga para pemain bisa saling menembak. Walaupun grafisnya masih *mediocore* dan screen refreshnya lambat, permainan ini tetap menjadi primadona di masanya. Permainan ini berjalan pada PLATO, dan permainan ini merupakan inspirasi bagi *Microsoft Flight Simulator*.

f. 1975: William Crowther membangun permainan berjenis *adventure text* pertama (pada awal mulanya disebut ADVENT, lalu kemudian menjadi *Colossal Cave*). Permainan tersebut dibangun dengan menggunakan Fortran untuk PDP-10. Pemain mengendalikan permainan melalui pemilihan serangkaian teks sebagai perintah yang nantinya dimasukkan

ke dalam sistem. Kemudian sistem akan mengeluarkan *feedback* berupa teks deskriptif. Pada pengembangan selanjutnya, permainan ini kemudian dibangun ulang dengan PLATO.

g. 1975: Sebagian besar universitas menerapkan CRT sebagai pengganti *print terminals*. CRT digunakan karena dapat menampilkan 30 garis teks hanya dalam waktu beberapa detik. Sedangkan *teleprinters* atau *line printers* hanya dapat menghasilkan keluaran yang dicetak dalam media kertas yang tentunya memakan waktu lebih lama. Adapun rata-rata kecepatan cetak alat-alat cetak tersebut adalah antara 10-30 karakter per detik.

h. 1975: Pada tahun ini pula permainan berjenis RPG pertama diciptakan. Judulnya adalah *dnd* yang alur permainannya berbasis pada permainan papan *Dungeons and Dragons*. Adapun sistem yang digunakan untuk menjalankannya adalah PLATO dan CDC.

i. 1976: Permainan berjenis RPG pada masa-masa awal yang murni berbasis *Dungeons and Dragons* berjudul *Telengrad* dikembangkan. Disamping itu permainan lain yang sejenis juga dikembangkan dan diberi judul *Zork* (yang kemudian berganti judul menjadi *Dungeon* pada tahun 1977).

j. 1977: Kelton Finn dan John Taylor membangun versi pertama dari permainan berjudul *Air*. Permainan tersebut merupakan permainan pertempuran angkasa berbasis teks pertama. Permainan ini kemudian dikembangkan menjadi berbasis grafis dan dapat dimainkan oleh lebih

dari seorang pemain secara *online*. Adapun permainan ini kemudian berganti judul menjadi *Air Warrior*, dan permainan tersebut merupakan permainan *online multi-player* pertama pada jenis ini.

k. 1977: Pengembangan permainan berjudul *Zork* dimulai pada tahun ini oleh Dave Lebling Marc Blank, Tim Anderson, dan Bruce Daniels. Tim Zork kemudian mengembangkan permainan ini agar dapat berjalan pada komputer personal.

l. 1978: Roy Trubshaw dan Richard Bartle mengembangkan MUD (*Multi-User Dugeon*) yang merupakan cikal bakal dari permainan MMORPG modern.

m. 1980: Michael Toy, Glenn Wichman, dan Ken Arnold merilis hasil kerjanya selama dua tahun yaitu *Rogue* pada BSD Unix. Permainan tersebut kemudian menginspirasi permainan lainnya yang bertipe serupa. Tidak jauh berbeda dengan *Dungeon* pada PDP-10 dan *dnd* pada PLATO, *Rogue* menampilkan peta ruang penjara bawah tanah dengan karakter teks. Adapun perbedaannya adalah pada *Rogue*, baik area permainan, harta karun dan musuhnya di tentukan secara acak oleh sistem setiap sesi permainan sehingga posisi mereka tak dapat ditebak. Kemudian *Rogue* dikembangkan untuk versi personal komputernya dan menjadi produk komersil sejak saat itu.

4. Krisis Permainan Video (1977)

Pada tahun 1977 sejumlah pabrikan konsol permainan video tipe lama, yang sebagian besar memasarkan permainan klon dari *Pong*, mengobral saham

perusahaan mereka kepada publik. Hal itu dilakukan untuk menutupi kerugian yang mereka dapat karena banyaknya konsol yang tidak terjual sementara jumlah stoknya masih melimpah. Dua perusahaan konsol besar saat itu yaitu Fairchild dan RCA bahkan meninggalkan pasar konsol permainan video. Hanya Atari dan Magnavox yang bertahan dalam pasar tersebut walaupun mereka menelan kerugian besar pada tahun 1977 dan 1978.

Penyebab terjadinya krisis adalah banyaknya permainan video yang merupakan klon dari permainan *Pong* baik itu pada konsol rumahan ataupun *arcade*. Krisis ini akhirnya berakhir pada tahun 1978 ketika dirilisnya permainan produksi Taito yaitu *Space Invaders* dan permainan ini berhasil merebut kembali hati para pemain permainan video. Keberhasilan permainan ini menandai era kebangkitan permainan video dan bahkan membawanya ke era keemasan permainan *arcade*.

Beberapa saat setelah itu, *Space Invaders* dibeli oleh Atari dan dilisensikan pada mesin keluaran terbaru mereka Atari VCS. Adapun gambar dari Atari VCS yang lebih dikenal dengan Atari 2600 dapat dilihat pada gambar di bawah ini:



Gambar 2.2 Konsol Atari 2600

Space Invaders pada konsol ini telah menjadi "*killer app"* bagi Atari sendiri karena mampu melipatgandakan keuntungan dari hasil penjualannya hingga empat kali lipat. Keuntungan tersebutlah yang akhirnya menutupi kerugian yang Atari alami saat era krisis permainan video I. Keberhasilan Atari 2600 pun menghidupkan kembali pasar permainan video di Amerika Utara hingga terjadi krisis permainan video II.

5. Konsol Generasi II (1977-1983)

Pada konsol generasi sebelumnya, kode komputer untuk satu atau beberapa permainan ditanamkan ke dalam *microchips* menggunakan *discrete logic*. Hal itu menjadikan sebuah konsol tidak dapat ditambahi permainan lain selain yang tertanam padanya. Pada pertengahan tahun 1977 hal itu berubah dengan dikembangkannya sistem *catridge* 'kaset' untuk konsol permainan video. Pelopor dari penggunaan sistem tersebut adalah Fairchild yang merilis produk terbarunya yaitu *Video Entertainment System* (VES) pada tahun 1976.

Pada sistem 'kaset' ini, program-program dibakar pada chip ROM yang kemudian dikemas di dalam *casing* plastik. *Catridge* ini kemudian bisa dibongkar pasang pada *slot* konsol yang kompatibel dengannya. Pada saat 'kaset' terpasang pada konsol, sebuah mikroprosesor *general-purpose* yang terdapat pada konsol kemudian membaca memori 'kaset' tersebut dan mengeksekusi program apapun yang berada padanya. Hal itu menjadikan pemain dapat memainkan lebih banyak permaianan bukan hanya permainan bawaan dari konsol saja.

Pada saat itu, untuk memproduksi suatu permainan video dibutuhkan *skill* 'keterampilan' individu yang sangat tinggi. Hal itu dikarenakan dalam proses

pengembanganya hanya melibatkan satu orang saja. Seorang pemrogram senior permainan *adventure* terkenal, Warren Robbinet, menyatakan bahwa saat itu untuk memgembangkan sebuah permainan video, satu persatu permainannya diselesaikan oleh hanya seoranag pemrogram. Adalah seorang pemrogram yang menciptakan konsep, menulis program, membuat grafis – menggambar pada kertas lalu kemudian mengkonversinya ke heksadesimal, dan mengolah suaranya.

Tiga mesin yang mendominasi konsol generasi II di Amerika Utara adalah sebagai berikut:

a. *The Video Computer System* (VCS)

   Konsol ini kemudian dikenal dengan nama Atari 2600, sebuah konsol berbasis *catridge* 'kaset' yang dirilis oleh Atari pada tahun 1977. Sembilan judul permainan dikembangkan khusus untuk edisi musim liburan saat itu. Adapun "*killer app"* dari konsol ini adalah *Space Invaders* yang melipatgandakan keuntungan penjualan konsol ini hingga empat kali. Konsol ini kemudian menjadi konsol yang paling populer di Amerika Utara hingga era krisis permainan video II.

b. *The Intellivision*

   Diperkenalkan oleh Mattel pada tahun 1980. Walaupun masih merupakan konsol generasi 8-bit, konsol ini memiliki prosesor unik yang dapat memproses instruksi yang lebarnya hingga 10 bits dan memliki *register* hingga . Hal itu tentunya memungkinkan konsol ini untuk dapat mengeksekusi instruksi lebih banyak dan lebih cepat dari konsol lainnya. Selain itu, konsol ini memiliki grafis yang lebih baik

dari Atari 2600. Beberapa hal tersebutlah yang menjadikan konsol ini populer.

c. *The ColecoVision*

Sebuah konsol yang bahkan lebih canggih dari yang lainnya dan muncul pada tahun 1982. Pada mulanya penjualannya pun meroket, namun disebabkan kurangnya pilihan permainan pada konsol ini, lambat laun akhirnya membuat *The ColecoVision* ditinggalkan.

Pada tahun 1979, Activison didirikan oleh para mantan pemrogram Atari yang sadar bahwa mereka telah dicurangi oleh perusahaannya. Program permainan video yang mereka kembangkan ternyata menghasilkan keuntungan sebesar $100 juta bagi perusahaan hanya dalam penjualan *catridge* ‘kaset’ saja. Hal itu dirasakan tidak sebanding dengan jumlah upah yang mereka terima yaitu hanya sebesar $20 ribu. Activision kemudian menjadi perusahaan pengembang permainan video pihak ketiga.

Pada tahun 1982, diperkirakan ada 8 juta rumah di Amerika yang memiliki konsol permainan video. Industri permainan-video rumahan mendapat keuntungan sekitar $3,8 miliar setiap tahunnya. Jumlah tersebut sebanding dengan kurang lebih seperempat dari keuntungan yang dihasilkan oleh penukaran koin mesin *arcade*. Pada industri permainan video *arcade* saat itu, keuntungan dari penukaran koin saja diperkirakan sebesar $8 milliar.

6. Era Keemasan Permainan *Arcade* (1978-1986)

Industri permainan *arcade* memasuki masa keemasannya dimulai pada tahun 1978 dengan dirilisnya *Space Invaders* oleh Taito sebagai pelopornya. Hal

tersebut menginspirasi puluhan rumah produksi untuk memasuki pasar ini. Pada masa ini mesin *arcade* banyak ditemui di lokasi keramaian seperti pusat perbelanjaan, pasar swalayan, rumah makan, dan bahkan di mini *market*. Mesin *arcade Space Invaders* terjual sebanyak 360.000 buah di seluruh dunia dan pada tahun 1982 meraih keuntungan sebesar $2 milliar dari penukaran koinnya saja. Sementara itu pada tahun 1979, Namco berhasil menjual 40.000 mesin *arcade*-nya yang permainannya berjudul *Galaxian* dan Atari merilis *Asteroids* yang kemudian terjual sebanyak lebih dari 70.000 mesin *arcade*.

Total penjualan mesin *arcade* di Amerika Utara meningkat drastis pada masa ini yang semula $50 juta pada tahun 1978 hingga $900 juta pada tahun 1981. Pada tahun 1970-an industri permainan video *arcade* meraih keuntungan sebesar $1 milliar pada penukaran koinnya saja dan jumlah tersebut meningkat hampir tiga kali lipatnya menjadi $2,8 milliar pada tahun 1980. Terlebih lagi dengan kemunculannya mesin *arcade* berwarna pada tahun 1978. Dirilisnya permainan *Pac-Man* pada tahun 1980 pun turut memengaruhi era keemasan permainan *arcade* pula. Mesin *arcade Pac-Man* terjual sebanyak lebih dari 350.000 kabinet dan mendapat keuntungan dari penukaran koinnya lebih dari $1 milliar hanya dalam waktu satu tahun.

Pada tahun 1981, industri permainan video *arcade* mendapat total keuntungan tahunan sebesar $5 milliar di Amerika Utara, yang jumlah tersebut setara dengan $12.3 milliar pada tahun 2011. Pada tahun 1982, industri ini mencapai puncaknya yang ditandai dengan pencapaian keuntungan sebesar $8 milliar dari keuntungan penukaran koinnya. Keuntungan tersebut melampaui

keuntungan dari industri musik pop ($4 miliar) dan industri perfilman Hollywood ($3 milliar) walaupun keuntungan keduanya dibabungkan sekalipun. Keuntungan ini juga masih dua kali lipatnya dari keuntungan industri permainan video rumahan yaitu pada sebesar $3.8 miliar di tahun yang sama. Keuntungan industri permainan video jika digabungkan, *arcade* dan rumahan, yaitu berjumlah sebesar $11.8 miliar pada tahun 1982 yang setara dengan lebih dari $27,3 miliar jika pada tahun 2011. Industri permainan video *arcade* terus berlanjut meraih keuntungan pertahunnya hingga sebesar $5 miliar sampai dengan tahun 1985 dari penukaran koinnya.

7. Permainan Komputer Rumahan (akhir 1970 – awal 1980)

Ketika mendominasinya permainan video *arcade* dan konsol pada kemajuan industri permainan video, *home computer* ‘komputer rumahan’ mulai muncul. Kemunculannya ini berawal dari sekitar akhir tahun 1970-an yang kemudian berkembang sangat pesat pada dekade 80-an. Komputer rumahan ini memungkinkan pemiliknya memainkan program permainan yang terpasang padanya. Perangkat lunak permainan komputer untuk komputer rumahanj kemudian mulai banyak dikembangkan. Pada awalnya pengembangannya itu bukan dilakukan oleh perusahaan pengembang perangkat lunak, melainkan oleh para penghobi permainan video.

Kebanyakan dari permainan video yang dikembangkan untuk komputer rumahan pada masa ini merupakan klon dari permainan video yang sudah ada pada *platform* lain. Beberapa permainan ini diantaranya merupakan klon dari beberapa permainan tenar seperti *Star Trek*, *Space Invaders*, *Froggers*, *Pac-Man*,

dan *Donkey Kong*. Adapun pendistibusiannya pada masa ini sangatlah menarik yang salah satu caranya yaitun dengan diedarkannya kode sumber untuk permainan tertentu. Peredarannya tersebut melalui media majalah, koran, maupun buku sehingga memungkinkan seseorang untuk menulis dan mengembangkan aplikasi permainannya sendiri. Kebanyakan kode sumbernya ditulis dalam bahasa BASIC.

Permainan video untuk komputer rumahan ini juga didistribusikan melalui media fisik seperti disket *floppy*, kaset atau katrij ROM yang dikirim lewat pos langsung ke alamat rumah. Pada masa ini pula banyak perusahaan kecil pengembang permainan video untuk komputer rumahan yang dibentuk oleh para pemrogram amatir yang tentunya merupakan para penghobi permainan video pada awalnya. Oleh mereka, disket yang berisi permainan video dijual dengan bungkusan plastik dan dititipkan di toko-toko atau *minimarket* di sekitar mereka.

2.1.2.3 Era 1980-an

Industri permainan komputer mengalami keterpurukan pada awal dekade ini. Disamping itu, pada era inilah bermunculan perusahaan-perusahaan permainan video yang masih bertahan hingga saat ini seperti Electronics Arts. Kebanyakan permainan pada dekade ini merupakan klon dari permainan *arcade best seller* yang telah ada sebelumnya. Adapun, beberapa kejadian penting yang berkaitan dengan perkembangan sejarah permainan video pada era ini diantaranya adalah:

1. Inovasi *Genre*

Era keemasan permainan video *arcade* mencapai puncaknya pada tahun 1980. Era tersebut memunculkan banyak inovasi dari hal *genre* permainan sehingga berbagai *genre* pun mulai bermunculan. Berikut ini merupakan sekumpulan *genre* permainan yang disusun secara alfabetis yaitu sebagai berikut:

a. *Action Adventure Game*
b. *Action Role-Playing Game*
c. *Adventure Game*
d. *Beat ‘em Up*
e. *Cinematic Platformer*
f. *Computer Role-Playing Vidoe Games*
g. *Console Role-Playing Video Games*
h. *Fighting Games*
i. *Hack and Slash*
j. *Interractive Movies*
k. *Platform Games*
l. *Scrolling Platformers*
m. *Scrolling Shooters*
n. *Isometric Platformer*
o. *Isometric Shooter*
p. *Light Gun Shooter*
q. *Maze Games*
r. *Platform-Adventure Games*

s. *Racing Games*

t. *Rail Shooter*

u. *Real Time Strategy*

v. *Run & Gun Shooters*

w. *Rhytm Game*

x. *Stealth Game*

y. *Survival Horror*

z. *Vehicle Simulation Games*

aa. *Visual Novels*

2. Komputer *Gaming*

Mengikuti kesuksesan Apple II dan Commodore PET pada akhir tahun 1970-an, serangkaian produk dengan harga lebih murah muncul dipasaran diawal 1980-an. Produk tersebut diantaranya adalah Commodeore Vic 20 dan 64, Sinclair ZX80, ZX81 dan X68000, Atari keluarga 8-bit, BBC Micro, Acorn Electron, Amstard CPC, dan seri MSX. Produk yang beredar dengan harga lebih murah ini mengkatalisasi pasar komputer rumahan dan juga pasar permainan video. Commodore 64 dapat dilihat pada gambar berikut ini:



Gambar 2.3 Commodore 64

3. Permainan Online pertama
4. Permainan pada Perangkat Genggam dengan LCD
5. Krisis Permainan Video II (1983)

   Pada akhir tahun 1983, industri permainan video mengalami krisis kembali dan bahkan lebih parah dari krisis permainan video I. Atari bahkan membangun kuburan masal bagi katrij-katrij permainannya yang tidak terjual di daerah New Mexico.
6. Konsol Generasi III (1983-1995 [8-bit])

   Pada tahun 1985, pasar konsol permainan video bangkit kembali berkat dirilisnya konsol terbaru berbasis 8-bit oleh Nintendo. Konsol tersebut dberi nama Famicom pada mulanya, namun kemudian lebih dikenal dengan NES (Nintendo Entertainment System). Berikut ini merupakan gambar dari NES:



Gambar 2.4 Nintendo Entertainment System

2.1.2.4 Era 1990-an

Di tahun 90-an ini ditandai oleh inovasi besar dalam dunia permainan video. Masa ini merupakan masa transisi grafis dari permainan video yang semula berbasis raster ke 3D. Tambahan pula, di dekade ini terjadi peningkataan

permintaan terhadap beberapa genre permaianan seperti *first-person shooter*, *real-time strategy* dan MMO. Perangkat permainan genggam juga menjadi tenar di masa ini karena menyebarnya wabah Game Boy setelah pertama kali rilisnya sejak 1989. Permainan *arcade* masih relatif tenar di masa ini walaupun penggemarnya sudah mulai berkurang.

Industri permainan video telah berkembang menjadi lebih matang pada era ini. Perkembangan besar pada masa ini ditandai dengan peningkatan dalam beberapa hal seperti jumlah pengembang, dana proyek, tim pengembang, dan intensitas kolaborasi antara pengembang dengan industri musik ataupun perfilman. Sebagai contoh adalah keterlibatan Mark Hamill dalam proyek permainan video berjudul *Wing Commander III* atau pengenalan QSound.

Peningkatan performa komputer dan penurunan harga prosesor juga turut andil dalam perkembangan permainan video di dekade ini. Beberapa prosesor yang mengalami penurunan harga diantaranya adalah Intel 80386, Intel 80486, dan Motorola 68030. Hal tersebut berakibat pada kebangkitan grafis 3D serta peningkatan kemampuan multimedia berkat diberdayakannya *sound card* ‘kartu suara’ dan CD-ROM. Walaupun grafis 3D pada saat itu hanya masih berupa gambar datar yang diberi efek *shaded* seperti pada *Elite, Starglider 2, dan Alpha Waves,* ataupun *terxture mapping* sederhana seperti pada *Wolfstein 3D.*

Pada tahun 1991, *Sonic the Hedgehog* diperkenalkan. Permainan ini menjadi primadona sehingga menjadikan konsol *Sega Mega Drive* populer. Persaingan pun terjadi karena karakter Sonic ini menyaingi ketenaran maskot dari *Nintendo*

yaitu Mario. Kedua karakter tersebut sampai saat ini pun masih memiliki tempat di dunia industri permainan.

Pada tahun 1992 permainan berjudul *Dune* dirilis. Permainan ini memang bukan yang pertama dalam *genre*-nya, tetapi ia menjadi *trendsetter* sehingga menjadikan permainan ini kiblat bagi permainan ber-*genre* sejenis setelahnya. Adapun beberapa permainan RTS yang berkiblat pada *Dune* ini bahkan merupakan permainan yang tergolong *blockbuster* seperti *Warcraft: Orcs and Humans, Command and Conquer,* dan *StarCraft.* Permainan ber-*genre* RTS ini ditandai dengan sudut pandang pemain dari atas, sebuah peta kecil, dan kontrol terhadap aspek ekonomi dan militer terutama angkatan bersenjatanya. Persaingan antara dua jenis permainan RTS seperti yang bergaya *Warcraft* dan *C&C* pun berlanjut hingga milenium berikutnya.

Adapun beberapa isu yang fenomenal pada dekade ini yang berkaitan dengan perkembangan sejarah permainan video diantaranya adalah:

1. Kemunduran *Arcade*
2. Kejenuhan terhadap perangkat permainan genggam
3. Konsol generasi IV (1989-1999 [16-bit])
4. Konsol generasi V (1993-2006 [32 dan 64-bit])
5. Transisi ke 3D dan CD
6. Permainan ponsel

2.1.2.5 Era 2000-an

Dekade awal pada milenium baru ini menunjukkan berbagai inovasi baik pada permainan kompouter maupun konsol. Lebih lagi, area permainan ponsel menjadi pasar baru yang kompetitif pada dekade ini.

Fenomena *mods* yang merupakan istilah modifikasi game yang dibuat sendiri oleh pemain pun mulai mewabah. Ketenaran *mods* dikenal pertama kali saat era *Wolfstein 3D* dan *Doom* yang kemudian berlanjut sampai ke awal milenium baru. Di samping itu, *mods* yang paling terkenal adalah yang diterapkan pada *Counter Strike* yang dirilis pada tahun 1999. Hingga saat ini pun permainan tersebut masih terkenal dan mempunyai kelompok penggemar tersendiri. Pada kenyataanya, *mods* pada permainan ini dibuat untuk *Half Life* oleh dua pemrogram berbeda. Akhirnya, para perancang permainan menyadari bahwa *mods* ini ternyata dapat menambah nilai dari permainan yang mereka rancang. Hal itu mengakibatkan bermunculanlah banyak permainan yang memfasilitasi *mods*.

Pada dekade awal milenium baru ini terdapat beberapa isu yang memengaruhi sejarah permainan video seperti:

1. Konsol generasi VI (1998-2011)
2. Kembali tenarnya alat kontrol alternative
3. Permainan *online* bangkit ke permukaan
4. Permainan ponsel
5. Konsol generasi VII (2004-saat ini)
6. Peningkatan dana pengembangan
7. *Nintendo* beralih ke permainan *casual*

8. Sensor gerakan merevolusi alat kontrol permainan
9. Kebangkitan permainan komputer *casual*
10. *Cloud computing* ‘komputasi awan’ mulai menjamah dunia permainan video

2.1.2.6 Era 2010-an

Pada dekade baru ini terlihat kemajuan di berbagai aspek permainan video. Kemungkinan munculnya konsol generasi terbaru pun mulai dinanti oleh para penikmat permainan video. Namun faktanya, industri permainan video sepertinya tidak begitu antusias dalam pengembangan konsol generasi terbaru ini. Salah satu alasanya adalah biaya yang dibutuhkan untuk pengembangan yang pastinya akan sangat tinggi. Hal itu dikarenakan perangkat keras yang dibutuhkan haruslah lebih canggih dari generasi sebelumnya untuk menampilkan grafis yang lebih mutakhir. Pada era ini, pengembang lebih jatuh hati pada pengembangan permainan ponsel dan perangkat genggam.

Pada tanggal 14 Juni 2010 di acara E3, *Microsoft* mengungkap konsol terbaru mereka Xbox 360 S atau Slim. *Microsoft* membuat konsol ini lebih ramping, tak bising dari yang sebelumnya, disertai dengan *harddrive* dengan kapasitas 250 GB, dan 802.11n WiFi yang terintegrasi. Konsol ini rencanaya akan menerapkan Onlive, yaitu permaian video berbasis *cloud computing* ‘komputasi awan’ pertama di dunia.

Berikut ini merupakan beberapa isu seputar perkembangan permainan video yang sedang hangat di awal dekade ini:

1. Bermain tanpa alat kontrol

Pada tanggal 27 November 2010, *Microsoft* merilis *Kinect* sebagai pelengkap pada paket penjualan Xbox 360. Alat tersebut menggunakan sensor dan perangkat kamera-ganda untuk mendeteksi pergerakan pemain. Hal tersebut menjadikannya sebagai sistem sensor gerak pertama yang tidak membutuhkan alat kontrol. Rata-rata dari jumlah unit yang terjual selama 60 hari pertama sejak rilisnya adalah 133,333 unit. Angka tersebut kemudian menjadikannya tercatat pada *Guiness World Record* dalam kategori alat elektronik dengan penjualan tercepat.

2. Konsol generasi VIII (2011-saat ini)

Berikut ini merupakan konsol yang masuk dalam kategori generasi ke-8:

a. PlayStation Vita

Pada tanggal 27 Januari 2011, PlayStation Vita dipublikasikan ke media. Konsol tersebut memiliki layar sentuh di bagian depan dan bantalan sentuh di bagian belakang, dua stik analog, koneksi 3G dan WiFi, serta Sixaxis control. Kemunculannya ini merupakan kompetisi terhadap Nintendo 3DS.

b. 3D tanpa layar

Pada tanggal 25 Maret di Eropa dan 27 Maret di Amerika, Nintendo merilis konsol genggam terbarunya sebagai penerus dari versi sebelumnya yaitu Nintendo 3DS. Perangkat genggam ini tidak memanfaatkan kaca untuk menampilkan grafis 3D, tetapi memanfaatkan teknologi *parallax barrier*. Fitur lainnya adalah

perangkat ini dilengkapi tiga kamera, sensor gerak, sensor *gyro* dan bantalan *slide* yang bisa menerima input analog sampai 360 derajat.

c. Alat kontrol layar sentuh

Pada tanggal 25 april 2011, Nintendo resmi mengumumkan konsol terbarunya sebagai penerus Wii. Konsol baru ini secara resmi diperkenalkan ke publik pada tanggal 7 Juni 2011 dengan nama Wii U. Menurut pernyataan resmi dari Nintendo, konsol ini akan dirilis sekitar tahun 2012 dan akan dipamerkan sebelumnya pada acara E3 tahun 2011. Fitur pada konsol ini diantaranya adalah alat kontrol dengan 6.2 inchi layar sentuh yang dengannya pemain dapat menjalankan konsol bahkan tanpa dukungan perangkat televisi. Selain itu, layar tersebut juga difungsikan sebagai layar kedua yang menyediakan informasi dan interaktifitas bagi pemain. Perangkat keras tambahan sebelumnya seperti remot Wii dan Wii *balance board* pun masih dapat digunakan bersama konsol ini.

2.2 Permainan Komputer

Berdasarkan hasil studi literatur dapat didefinisikan bahwa permainan komputer merupakan permainan video yang berjalan pada *platform* sistem komputer [VID11].

2.2.1 Perkembangan

Peran komputer sebagai “mesin game” telah menjadi mode tersendiri di dunia [LUK03]. Hal tersebut ditandai dengan munculnya para penggemar khusus yang hanya memainkan aplikasi permainan komputer (PC game) saja. Mereka

lebih memilih komputer sebagai *platform* di karenakan berbagai alasan seperti biaya dan kinerja. Biaya pengadaan komputer jauh lebih rendah dibanding dengan konsol. Terlebih lagi, komputer juga dapat beralih fungsi kembali ke fungsi "normal"-nya. Kinerja dari komputer sebagai *game platform* tentunya sangat jauh lebih tinggi dibanding konsol terutama dalam hal penampilan grafisnya.

Pada awal penciptaaannya di Amerika Serikat, aplikasi permainan pertama memang diciptakan dan berjalan dengan komputer. Permainan komputer pertama kali dibuat pada tahun 1966 oleh Ralph Baer bersama 500 insinyur dan terknisi [LUK03]. Permainan yang diproduksinya hanya dapat dimainkan dengan komputer seharga US$ 40.000 dan yang mendanainya adalah Pentagon. Proyek ini berhasil membuat simulasi komputer yang menampilkan dua titik putih yang berkejar-kejaran di layar. Sayangnya, proyek Baer selanjutnya yaitu permainan papan dan bola tidak menarik minat Pentagon. Baer pun kemudian berusaha memperoleh izin agar dapat memproduksi mesin permainan secara komersial dan inilah yang menjadi cikal bakal dari industri permainan komputer.

Industri permainan komputer telah berjaya selama lebih dari 30 tahun. Hal tersebut ditandai dengan banyaknya perusahaan pengembang permainan komputer diantaranya seperti Atari, Namco, Squaresoft, Blizzard Entertainment dan Infinity Ward. Industri ini merupakan industri yang unik karena pemeran utama dalam kejayaannya bukanlah para eksekutif bisnis dan pemasaran perusahaan melainkan oleh para pemainnya (gamers) itu sendiri [HAR08]. Hal tersebut menjadikan pengembang permainan komputer harus menyesuaikan dengan apa yang diinginkan oleh para *gamers* untuk dapat bertahan di industri ini.

Pada empat tahun belakangan ini muncul *trend* terbaru dalam industri permainan yaitu *Casual Game* [HAR08]. Sebenarnya *trend* tersebut merupakan bentuk baru dari *arcade-game* hanya saja untuk dapat memainkannya tidak diperlukan suatu mesin *arcade* tertentu. Selain itu, jenis (genre) permainan ini tidak mengharuskan pemainnya menghabiskan waktu berjam-jam juga tidak membutuhkan keterampilan khusus. Buku manual atau *user instruction* yang rumit pun sepertinya tidak dibutuhkan sama sekali. Hal ini menjadikan penggemar permainan komputer semakin banyak saja karena jenis permainan ini dapat dimainkan oleh hampir siapa pun bukan hanya para penggila permainan (*Hardcore Gamers*) saja.

2.2.2 Kepopuleran

Permainan komputer saat ini telah menjadi bagian yang tidak terpisahkan dari pengguna komputer. Sejak semakin menyusutnya ukuran dan bertambah canggihnya kemampuan komputer, kecenderungan komputer pun beralih dari hanya sekedar mesin yang membantu manusia untuk menyelesaikan tugas-tugas perkantoran menjadi mesin hiburan yang mumpuni. Tambahan pula, mayoritas pengguna komputer yang penulis temui memanfaatkan komputer lebih ke sisi mesin hiburannya terutama untuk menjalankan permainan komputer. Hanya dengan alasan itu saja, mereka pun sampai berlomba-lomba meninggikan spesifikasi komputernya.

Permainan komputer pada dasarnya merupakan sebuah program komputer sama seperti aplikasi atau perangkat lunak lainnya. Suatu aplikasi komputer berjalan pada sistem komputer yang mana sistem komputer ini tersusun oleh

berbagai komponen yang saling berkerja sama untuk mencapai tujuan komputasi. Dalam lingkup permainan komputer, aplikasi permainan ini berjalan pada lingkungan sistem operasi yang dibantu sekumpulan program utilitas untuk mengatur kerja perangkat keras sehingga dapat menjalankan aplikasi permainan tersebut. Dalam hal demikian, tujuan komputasi yang dicapai sistem komputer adalah menjalankan aplikasi permainan secara normal.

Aplikasi permainan komputer merupakan perangkat lunak, namun pernyataan tersebut tidak semata-mata menjadikan hanya mesin komputerlah yang bisa menjalankannya. Saat ini banyak konsol diciptakan khusus untuk menjalankan permainan saja. Konsol-konsol yang beredar pun sekarang sudah banyak jenisnya. Mereka telah melewati berbagai tahap pengembangan sejak generasi pertama, sehingga kinerjanya semakin meningkat dari generasi ke generasi. Walaupun demikian, kinerja yang terbaik masihlah dihasilkan oleh mesin komputer, sehingga sebagian besar penikmat aplikasi permainan memilih menggunakan mesin komputer saja.

2.3 AI (*Artificial Intelligence*)

Sebagian kalangan menerjemahkan *Artificial Intelligence* sebagai kecerdasan buatan, kecerdasan artifisial, atau intelejensia buatan. Pada karya tulis ini, istilah *Artificial Intelligence* sengaja tidak diterjemahkan ke bahasa Indonesia karena istilah tersebut sudah sangat akrab bagi orang Indonesia. Begitu juga dengan singkatan istilah tersebut, yaitu AI, sudah sangat melekat diberbagai media ilmiah maupun non ilmiah.

Pada [RUS95], Stuart Russel dan Peter Norvig mengelompokkan definisi AI, yang diperoleh dari beberapa *textbook* berbeda, ke dalam empat kategori, yaitu:

1. *Thinking humanly: the cognitive modeling approach*
2. *Acting humanly: the Turing test approach*
3. *Thinking rationally: the laws of thought approach*
4. *Acting rationally: the rational agent approach*

Para ilmuwan memiliki dua cara pandang yang berbeda tentang AI. yang pertama adalah memadang AI sebagai bidang ilmu yang hanya fokus pada proses berpikir. Sedangkan yang ke dua adalah memandang AI sebagai bidang ilmu yang fokus pada tingkah laku. Cara pandang ke dua memandang AI secara lebih luas karena suatu tingkah laku selalu didahului dengan proses berpikir.

2.3.1 Definisi

Definisi AI yang paling tepat untuk saat ini adalah *acting rationally* dengan pendekata *human agent* [SUY09]. Hal ini berdasarkan pemikiran bahwa komputer bisa melakukan penalaran secara logis dan bisa melakukan aksi secara rasional bedasarkan hasil penalaran tersebut.

2.3.2 Perkembangan

Sejak pertama kali dikemukakan istilah AI pada tahun 1965 di konferensi Darthmouth, AI terus dikembangkan melalui berbagai penelitian mengenai teori-teori dan prinsip-prinsipnya. Perkembangan AI mengalami pasang surut mengikuti antusias para peneliti dan dana penelitian yang tersedia. Pada periode 1966 sampai 1947, perkembangan AI melambat. Tetapi sejak 1980, AI menjadi

sebuah industri yang besar dengan perkembangan yanga sangat pesat. Banyak industri skala besar yang melakukan investasi besar-besaran dalam bidang AI.

Saat ini, dengan semakin cepatnya perkembangan perangkat keras dan perangkat lunak, berbagai produk AI telah berhasil dibangun dan digunakan dalam kehidupan sehari-hari. Dengan teknologi perangkat keras yang performansinya semakin tinggi dan berukuran kecil serta didukung teknologi perangkat lunak yang semakin beragam dan kuat, produk-produk berbasis AI semakin dekat dengan kehidupan manusia [SUY09].

Pada masa mendatang, AI ditantang untuk membuat suatu kecerdasan yang hampir menyamai kecerdasan manusia. Ray Kuzwell memprediksi bahwa hal itu akan mungkin terwujud melalui tahapan-tahapan prediksi yang dibuatnya secara bertahap sampai sekitar 90 tahun ke depan (tahun 2099).

2.3.3 Teknik Pemecahan Masalah AI

Terdapat empat teknik dasar pemecahan masalah yang terdapat pada bidang *academic/traditional artificial intelligence* (AI) [SUY07] diantaranya adalah :

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 diketahi dapat ditemukan.

2.3.4 *Game Artificial Intelligence*

Dalam ruang lingkup permainan AI yang diterapkan sedikit berbeda dengan *traditional* AI. AI pada permainan dikenal juga dengan sebutan *game* AI. menurut penerapannya pada entitas permainan AI dibedakan menjadi tiga, yaitu :

a. Combat AI

   AI ini diterapkan pada musuh dalam area permainan. Penerapannya dimaksudkan agar musuh memiliki kemampuan untuk dapat melawan pemain lain. Sebagai contoh pada permainan *arcade-shooter* sang musuh harus mengetahui kapan ia harus menembak, menghindar, melemparkan granat dan bersembunyi.

b. Non-combat AI

   AI ini diterapkan pada entitas lain dalam arena permainan selain pemain (*human players*) baik itu yang berperan sebagai musuh ataupun bukan. AI ini dapat kita lihat pada permainan berjenis RPG. Dimana dalam permainan jenis tersebut banyak ditemui entitas-entitas yang memiliki kegiatannya masing-masing dalam suatu arena permainan.

c. Analitycal AI

   AI ini diterapkan dalam permainan yang berjenis simulasi. Hal tersebut dikarenakan AI ini memiliki tingkat kerumitan yang tinggi. Sebagai contoh dalam simulasi pesawat terbang, presisi, pergerakan pesawat, dan *state* membutuhkan AI tersendiri.

Sedangkan Brian Schwab di dalam bukunya [SCH04] membagi AI ke beberapa jenis diantaranya:

a. FSM (*Finite State Machine*)

b. FuSM (*Fuzzy State Machine*)

c. *Message Based System*

d. *Scripting System*

e. *Location Based Information System*

f. *Genetic Algorithm*

g. *Neural Network*

Adapun berdasarkan hasil studi literatur terhadap buku Dustin Clingman [CLI04], AI pada permainan komputer terdiri dari:

a. *Deterministic* AI

b. *Tracking*

c. *Evasion*

Dari keterangan yang telah dibahas diatas dapat disimpulkan bahwa untuk meningkatkan *gaming experience*, AI yang cocok untuk diterapkan pada aplikasi permainan *arcade-shooter* adalah *combat* AI. Teknik *reasoning*-lah yang cocok dalam penerapannya. Hal tersebut dikarenakan teknik ini relatif mudah untuk diterapkan. Dan *game* AI yang dapat diterapkan adalah messaging system dan FSM (*finite state* machines). Suatu *agent* (sang musuh) akan diberi basis pengetahuan yang berisi aksi-aksi yang akan ia lakukan jika terjadi suatu *event* tertentu, sehingga sang musuh akan terlihat lebih cerdas. Penerapan teknik AI pada aplikasi permainan *arcade-shooter* diharapkan bisa menjadikan *gameplay*-nya menjadi lebih menarik.

2.4 Permainan *Arcade-Shooter*

Arcade-shooter merupakan salah satu dari sekian banyak jenis dari *arcade-game.* Membedakan jenis-jenis *arcade-game* merupakan hal yang sulit mengingat banyaknya *arcade-game* yang beredar sejak awal kemunculannya. Ari Feldman (2001) membaginya ke dalam lima jenis sub *genre* sebagai berikut :

a. *Maze/Chase games*
b. *Pong games*
c. *Shooters*
d. *Puzzlers*
e. *Platformers*

Jenis game *arcade-shooters* sering juga disebut “shoot-em-ups” atau “tembak-tembakan” dalam bahasa Indonesia. Kemunculan pertama dari jenis ini tak lama setelah kemunculan game jenis Pong. Jenis *game* ini telah menjadi idola semenjak pertengahan tahun 80 hingga awal tahun 90an. *Shooter* terbagi menjadi dua tipe utama yaitu statis dan *scrolling*. Contoh dari jenis *game* ini yang terkenal adalah Berserk (statis) dan 1943 (*scrolling*).

2.4.1 Skenario Permainan

Permainan *arcade-shooter* pada umumnya ditampilkan dalam 2D pada resolusi layar VGA (640 X 480) dengan latar statis. Pemain yang dikendalikan oleh manusia terdiri dari sebuah objek yang bentuknya dapat berupa pesawat ruang angkasa seperti pada permainan Asteroids atau Tank seperti pada permaian *Battle Tank*. Entitas lain yang terdapat pada permainan ini diantaranya adalah musuh yang dapat berupa sekumpulan asteroid, pesawat alien ataupun tank.

Navigasi dari objek yang dikendalikan manusia dilakukan menggunakan alat input berupa *joystick* pada *platform arcade* maupun konsol atau menggunakan *keyboard* pada platform komputer personal.

Pada permainan Asteroid objek pemain manusia akan ditempatkan di tengah layar diawal permainan. Hal itu tentunya berbeda dengan permainan Battle Tank yang menempatkan objek pemain manusia di bagian bawah layar. Objek musuh akan ditempatkan secara acak oleh komputer dengan jumlah tertentu yang bervariasi tergantung pada tingkat permainannya. Objek musuh akan menyerang pemain dengan menembakan peluru secara brutal atau menabrakkan dirinya kepada mengenai pemain. Dalam Asteroid objek musuh bergerak secara acak dengan menggunakan algoritma random pattern navigation.

Gambar-gambar dibawah ini menunjukkan tampilan pada permainan *arcade-shooter* yaitu Asteroids dan Battle Tanks.



Gambar 2.5 Tampilan pada permainan *arcade-shooter* Asteroids



Gambar 2.6 Tampilan pada permainan *arcade-shooter* Battle Tanks

Pada awal permainan jumlah nyawa yang diberikan pada pemain adalah tiga buah. Objek pemain manusia akan hancur jika terkena peluru musuh atau tertabrak asteroid. Hal tersebut menyebabkan nyawa pemain berkurang satu. Musuh akan hancur jika terkena satu peluru atau bertabrakan deangan objek pemain. Jumlah nyawa yang tersisa dan *score* yang didapatkan pemain akan ditampilkan di layar bagian kiri atas secara *default*. Objek dalam permainan dapat bergerak secara bebas dalam lingkungannya pada pemainan Asteroids. Pada permainan Battle Tank hal tersebut tidak dapat sepenuhnya dilakukan mengingat ada rintangan-rintangan pada lingkungan permainannya.

Alur permainan atau *gameplay*-nya adalah objek yang dikendalikan manusia harus menembaki musuh-musuh yang datang menyerangnya untuk mendapatkan *score*. Objek yang dikendalikan manusia memiliki kemampuan tempur dengan dibekali suatu senjata yang dapat menembakkan peluru dengan jumlah tak terbatas untuk membasmi musuh. Permainan akan berakhir jika semua musuh

telah musnah atau pemain kehabisan nyawa. Jika semua musuh telah musnah maka permainan akan berlanjut ke tingkat selanjutnya. Sedangkan jika pemain kehabisan nyawa maka akan ditampilkan menu untuk melanjutkan permainan atau mengakhirinya.

2.4.2 Visi dan Misi

Visi dari permainan *arcade-shooter* secara umum sama dengan visi dari jenis permainan lainnya. Visi nya adalah untuk menghibur pemain dengan *gameplay* yang disediakan secara menarik sehingga dapat meningkatkan *game experience* para pemainnya.

Adapun misi dari permainan ini adalah untuk menyediakan lingkungan permainan dimana didalamnya terdapat objek pemain yang dapat meraih *score* dengan menghancurkan rintangan dan musuh-musuhnya.

2.4.3 Elemen Permainan

Dalam permainan *arcade-shooter* elemen permainan terdiri dari :

1. Pemain (*Human Player*)

   Objek ini merupakan tokoh utama dalam permainan dan dikendalikan secara langsung oleh pemain manusia. Objek ini dapat melakukan navigasi dan menembakkan peluru untuk menghancurkan musuh ataupun rintangan.

2. Musuh (*CPU controlled Player*)

   Objek ini dapat dikatakan sebagai objek terpenting kedua setelah objek pemain. Musuh dikendalikan oleh komputer dengan menggunakan

algoritma tertentu seperti random movement navigation pada permainan Asteroids.

3. Rintangan (*Obstacles*)

   Objek ini ditempatkan secara acak dalam lingkungan permainan. Walaupun sepertinya tidak begitu penting, tetapi objek ini akan cukup menjengkelkan karena dapat membatasi pergerakan pemain.

4. Peluru (*Bullet*)

   Objek ini merupakan senjata utama dari pemain manusia dan musuh. Objek inilah yang menjadikan permainan *arcade-shooter* menjadi hidup.

2.4.4 Metode dan Algoritma pada Permainan

Setiap permainan komputer pasti memiliki algoritma yang khas tergantung dari jenis permainannya. Secara umum pada jenis permainan *arcade-shooter* terdapat beberapa algoritma atau metode yang sering digunakan. Algoritma tersebut terdiri dari *collision detection, 8 angle movement, random pattern navigation,* dan *ship rotation* [1]. Berikut ini penjelasannya :

a. Collision detection

Algoritma ini dibedakan lagi menjadi tiga berdasarkan metode pencapaian event tabrakannya yaitu *bounding box*, perbandingan jarak dan target indeks [LUK03]. Permaianan dengan jenis *arcade-shooter* pada umunya menggunakan *bounding box collision detection*. Pada metode ini suatu objek (gambar) diberi garis batas virtual pada sisi-sisinya. Suatu objek dikatakan bertabrakan jika antara garis batas virtualnya berpotongan. Hal itulah yang

memicu event berikutnya seperti ledakan atau pengurangan nyawa dari objeknya.

b. *8-angle movement*

Algoritma ini pada intinya menentukan arah pergerakan pesawat yang terdiri dari delapan arah mata angin. Algoritma ini sangat erat kaitannya dengan input dari pemain. Pemain akan menekan suatu tombol arah yang kemudian diikuti aksi pergerakan pesawat sesuai dari tombol yang ditekan.

c. *Random pattern navigation*

Algoritma ini sering kali diterapkan pada musuh. Penerapannya pada musuh akan mengakibatkannya bergerak secara tak menentu (*random*). Pergerakan ini akan kacau jika tidak diberi *statement* kontrol. *Statement* tersebut contohnya dengan mengeset suatu aturan jika musuh berada pada tepi layar maka ia akan memantul atau bahkan menembus layar dan mucul lagi pada arah tertentu. Dewasa ini penerapan algoritma ini menjadi lebih mudah dengan adanya pustaka fungsi random pada beberapa bahasa pemrograman seperti Java dan C++.

d. *Ship rotation*

Algoritma inilah yang menentukan rotasi dari pesawat. Dengan melibatkan perhitungan trigonometri, sehingga perputaran pesawat ataupun entitas lainnya pada game akan terlihat lebih realistis.

2.5 Program Aplikasi

2.5.1 Definisi Program

Program adalah ekspresi pernyataan atau kombinasi yang disusun dan dirangkai menjadi satu kesatuan prosedur yang berupa urutan langkah untuk menyelesaikan masalah. Ekspresi tersebut kemudian diimplementasikan menggunakan bahasa pemrograman, sehingga dapat dieksekusi komputer [MAR00].

2.5.2 Aplikasi

Aplikasi merupakan penerapan, proses menyimpan suatu hal, data, permasalahan pekerjaan ke dalam suatu sarana atau media yang digunakan untuk menerapkan atau mengimplementasikan hal atau permasalahan tersebut, sehingga berubah menjadi bentuk yang baru tanpa menghilangkan nilai-nilai dasar dari hal, data, permasalahan atau pekerjaan. Jadi, dalam hal ini hanya bentuk dari tampilan data yang berubah, sedangkan isi yang memuat dalam data tersebut tidak mengalami perubahan.

Sebagai simpulan, program aplikasi adalah sederetan kode yang digunakan untuk mengatur komputer supaya dapat melakukan pekerjaan sesuai dengan keinginan permrogram atau pengguna [MAR00].

2.5.3 Aplikasi Permainan

Berdasarkan beberapa definisi yang didapat dari hasil studi literatur, maka dapat disimpulkan bahwa aplikasi permainan adalah aplikasi komputer yang berfungsi untuk menjalankan suatu permainan video pada *platform* komputer.

2.6 *Object Oriented Programming* (OOP)

Object-Oriented Programming (OOP) adalah sebuah pendekatan untuk pengembangan suatu perangkat lunak dimana dalam strukturnya didasarkan kepada interaksi objek dalam penyelesaian suatu proses/tugas. Interaksi tersebut berupa pengiriman pesan antara objek-objek yang telibat. Objek akan merespon pesan tersebut menjadi sebuah *action* ‘tidakan’ atau metode. Jika seseorang mencoba melihat bagaimana tugas disekitarnya diselesaikan, maka ia akan mengetahui bahwa ia sedang berinteraksi dalam sebuah *object-oriented world*. Jika akan bepergian seseorang pasti berinteraksi dengan objek mobil. Sebagai sebuah objek, mobil juga ternyata terdiri dari objek- objek lain yang saling berinteraksi dan bekerja bersama-sama untuk melakukan tugasnya membawa seseorang ke tujuannya.

Program-program berorientasi objek terdiri dari objek-objek yang berinteraksi satu sama lainnya untuk menyelesaikan sebuah tugas. Seperti dunia nyata, pengguna dari program tersebut dilibatkan dari logika proses untuk menyelesaikan tugas. Contoh, ketika pengguna mencetak sebuah halaman pada aplikasi pemroses kata, pengguna tersebut berarti melakukan inisialisasi tindakan dengan mengklik tombol cetak. Kemudian ia hanya menunggu respon apakah *job* tersebut sukses atau gagal, sedangkan prosesnya terjadi secara internal tanpa diketahui oleh pengguna tersebut. Tentunya setelah ia menekan tombol cetak, maka secara simultan obejk tombol tersebut berinteraksi dengan objek mesin pencetak untuk menyelesaikan *job* tersebut.

Adapun karakteristik yang dimiliki OOP yaitu :

a. Class

Class atau kelas merupakan cetak biru dari objek-objek yang akan dilibatkan.

b. Objects

Objek adalah instansiasi dari kelas dan merupakan sebuah *structure* yang menggabungkan data dan metode untuk bekerja bersama-sama.

c. Abstraction

Ketika seseoran berinteraksi dengan objek-objek di dunia nya, ia sering hanya konsentrasi dengan sebuah bagian dari propertiesnya. Ia hanya melihat suatu objek dari tampak luarnya saja dengan tidak mempedulikan kompleksitas struktur internal yang ada didalamnya.

Ketika membangun objects dalam aplikasi OOP, penting sekali untuk menggabungkan konsep *abstraction* ini. Jika seseorang membangun aplikasi pengapalan, ia harus membangun objek produk dengan atribut seperti ukuran dan berat. Warna adalah contoh informasi yang tidak ada hubungannya dan harus dibuang. Tetapi ketika ia membangun aplikasi pemesanan, warna menjadi penting dan harus menjadi atribut dari objek produk.

d. Encapsulation

Ciri penting lainnya dari OOP adalah *encapsulation*. *Encapsulation* adalah sebuah proses dimana tidak ada akses langsung ke data yang diberikan, bahkan tersembunyi. Jika seseorang ingin mendapat data, maka ia harus berinteraksi dengan objek perantara yang bertanggung jawab atas data tersebut.

e. Polymorphism

Polymorphisms adalah kemampuan 2 buah objek berbeda untuk merespon pesan permintaan yang sama dalam suatu cara yang unik. Contohnya, fungsi cetak akan menghasilkan keluaran berbeda ketika dijalankan pada mesin pencetak dan layar monitor. Dalam OOP, penerapan tipe *polymorphism* melalui proses yang disebut *overloading*. Hal tersebut mengizinkan pengimplementasian metode yang berbeda pada sebuah objek yang mempunyai nama sama.

f. Inheritance

Penggunaan *inheritance* dalam OOP untuk mengklasifikasikan objek dalam program sesuai karakteristik umum dan fungsinya. Hal ini sama dengan konsep pewarisan sifat pada manusia. Jika manusia memiliki keturunan, maka beberapa sifat dari orang tuanya terkadang menurun pada anaknya. Analogi tersebut kemudian berlaku pada OOP dalam *inheritance* ini. Konsep ini kemudian menjadikan pemrograman lebih mudah karena memungkinkan pemrogram untuk mengkombinasikan karakteristik umum kedalam objek orang tua (*superclass*) dan mewariskan karakteristik ini ke objek anaknya (*subclass*).

2.7 *Unified Modelling Language* (UML)

UML *(Unified Modeling Language)* adalah sebuah bahasa untuk menetukan, visualisasi, kontruksi, dan mendokumentasikan artifact (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak, terutama yang dikembangkan paradigma berbasis objek

[FOW04]. Artifact dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan system non perangkat lunak lainnya.

UML merupakan suatu kumpulan teknik terbaik yang telah terbukti keberhsilannya dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

2.7.1 Bagian-bagian UML

Bagian-bagian utama dari UML adalah *view*, diagram, model *element*, dan *general mechanism.*

1. *View*

View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. *View* bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram. Beberapa jenis view dalam UML antara lain: *use case view*, *logical view*, *component view*, *concurrency view*, dan *deployment view*.

   a. *Use case view*

      Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan *external actors*. Aktor yang berinteraksi dengan sistem dapat berupa pengguna atau sistem lainnya. *View* ini digambarkan dalam *use case diagrams* dan kadang-kadang dengan *activity diagrams*. *View* ini digunakan terutama untuk klien, perancang (*designer*), pengembang (*developer*), dan penguji sistem (*tester*).

b. *Logical view*

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (*class, object,*dan *relationship*) dan kolaborasi dinamis yang terjadi ketika objek mengirim pesan ke objek lain dalam suatu fungsi tertentu. *View* ini digambarkan dalam *class diagrams* untuk struktur statis dan dalam *state, sequence, collaboration,* dan *activity* diagram untuk model dinamisnya. *View* ini digunakan untuk perancang (*designer*) dan pengembang (*developer*).

c. *Component view*

Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari *code module* diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi administratif lainnya. *View* ini digambarkan dalam *component view* dan digunakan untuk pengembang (*developer*).

d. *Concurrency view*

Membagi sistem ke dalam proses dan prosesor. *View* ini digambarkan dalam diagram dinamis (*state, sequence, collaboration,* dan *activity diagrams*) dan diagram implementasi (*component* dan *deployment diagrams*) serta digunakan untuk pengembang (*developer*), pengintegrasi (*integrator*), dan penguji (*tester*).

e. *Deployment view*

Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (*nodes*) dan bagaimana hubungannya dengan lainnya. *View* ini

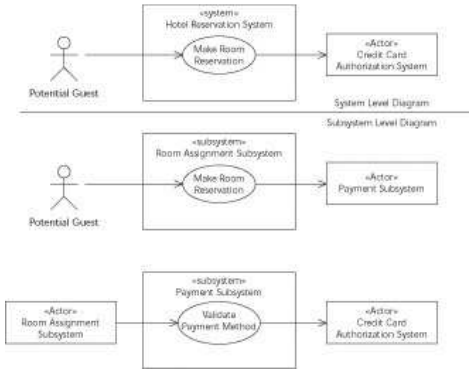
digambarkan dalam *deployment diagrams* dan digunakan untuk pengembang (*developer*), pengintegrasi (*integrator*), dan penguji (*tester*).

2. Diagram

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis diagram antara lain :

a. Use Case Diagram

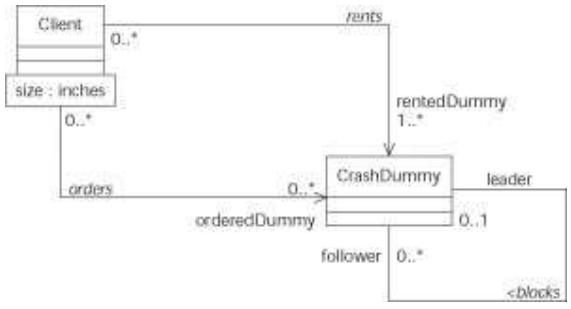
Use case adalah abstraksi dari interaksi antara sistem dan aktor. *Use case* bekerja dengan cara mendeskripsikan tipe interaksi antara pengguna sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. *Use case* merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata user. Sedangkan *use case diagram* memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan client. Gambar berikut ini merupakan gambar *use case diagram*.



Gambar 2.7 *Use Case Diagram* [CHO03]

b. Class Diagram

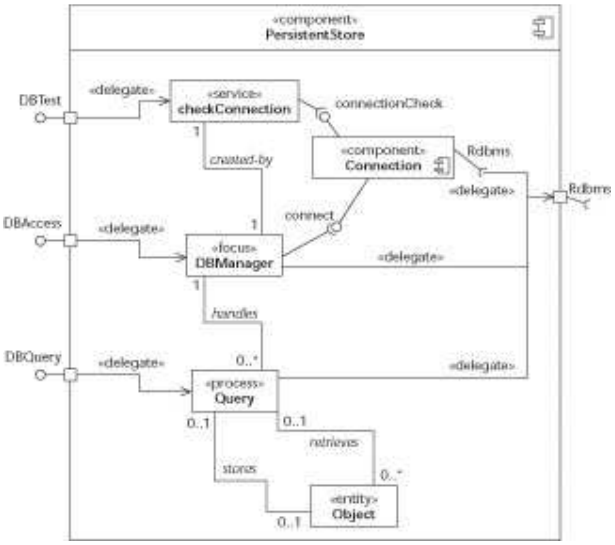
Class adalah dekripsi kelompok obyek-obyek dengan properti, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *class diagram* dapat memberikan pandangan global atas sebuah system. Hal tersebut tercermin dari *class-class* yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa *class diagram. Class diagram* sangat membantu dalam visualisasi struktur kelas dari suatu sistem. Class diagram dapat dilihat pada gambar berikut ini.



Gambar 2.8 *Class Diagram* [CHO03]

c. *Component Diagram*

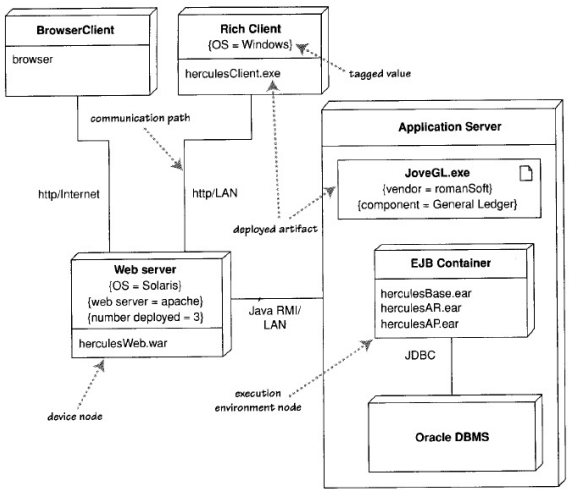
Component software merupakan bagian fisik dari sebuah sistem, karena menetap di komputer tidak berada di benak para analis. Komponen merupakan implementasi *software* dari satu atau lebih *class*. Komponen dapat berupa *source code,* komponen biner, atau *executable component.* Sebuah komponen berisi informasi tentang logika *class* atau *class* yang diimplementasikan sehingga membuat pemetaan dari *logical view* ke *component view*. Sehingga *component diagram* merepresentasikan dunia nyata yaitu *component software* yang mengandung *component, interface* dan *relationship*. *Component diagram* dapat dilihat pada gambar berikut ini.



Gambar 2.9 *Component Diagram* [CHO03]

d. Deployment Diagram

Menggambarkan tata letak sebuah system secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain dan jenis hubungannya. Di dalam *nodes*, *executable component* dan objek yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh *node* tertentu dan ketergantungan komponen. *Deployment diagram* dapat dilihat pada gambar berikut ini.

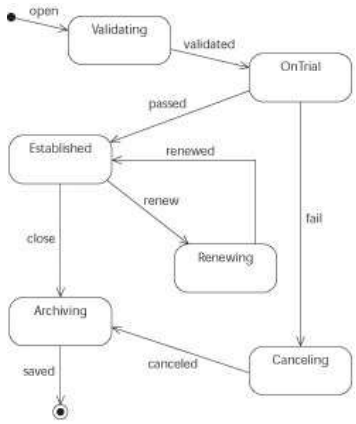


Gambar 2.10 *Deployment Diagram* [FOW04]

e. State Diagram

Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu objek dari suatu *class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa objek lain yang mengirim pesan. *State class* tidak digambarkan untuk semua *class*, hanya yang mempunyai sejumlah *state* yang terdefinisi dengan baik dan kondisi *class* berubah

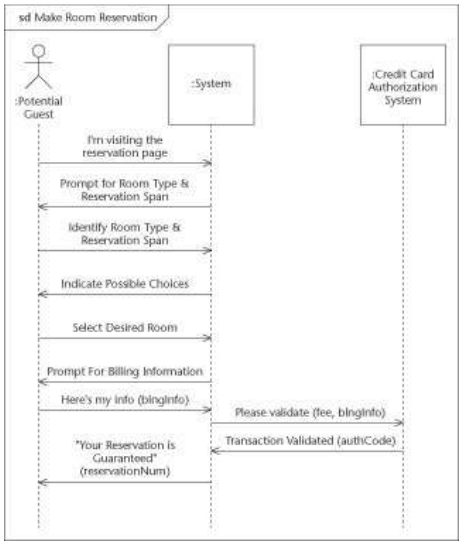
oleh *state* yang berbeda. *State diagram* dapat dilihat pada gambar berikut ini.



Gambar 2.11 *State Diagram* [CHO03]

f. Sequence Diagram

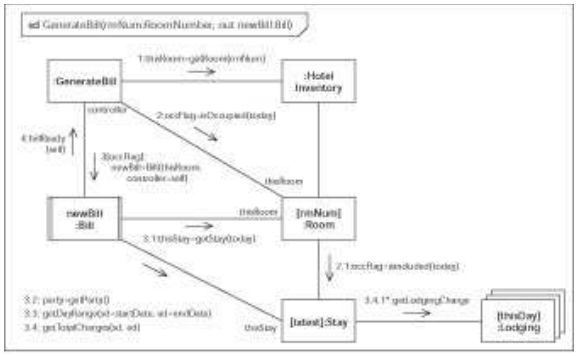
Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. *Sequence diagram* dapat dilihat pada gambar berikut ini.



Gambar 2.12 *Sequence Diagram* [CHO03]

g. Collaboration Diagram

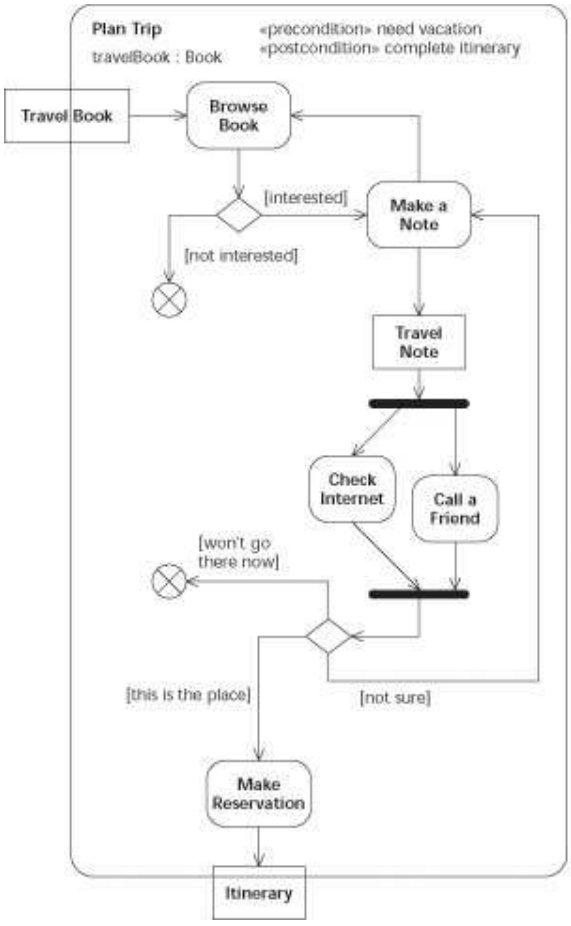
Menggambarkan kolaborasi dinamis seperti *sequence diagrams*. Dalam menunjukkan pertukaran pesan, *collaboration diagrams* menggambarkan objek dan hubungannya. Jika penekannya pada waktu atau urutan, maka lebih baik menggunakan *sequence diagrams*, tetapi jika penekanannya pada konteks gunakan *collaboration diagram. Collaboration diagra*m dapat dilihat pada berikut ini.



Gambar 2.13 *Collaboration Diagram* [CHO03]

h. Activity Diagram

Menggambarkan rangkaian aliran dari aktifitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.



Gambar 2.14 *Activity Diagram* [CHO03]

2.8 Java

Bahasa pemrograman Java (awalnya bernama bahasa pemrograman Oak) menurut literatur yang penulis pelajari [NUG09], diperkenalkan ke hadapan komunitas pemrograman di seluruh dunia oleh James Gosling dari Sun Microsystem Inc. pada sekitar tahun 1990-an. Tujuan pembuatan bahasa pemrograman Java adalah untuk meningkatkan kemampuan bahasa pemrograman C++ yang sebelumnya telah ada sehingga aplikasi-aplikasi (program komputer) yang dikembangkan dengan bahasa pemrograman tertersebut mampu berjalan di atas berbagai *platform* perangkat keras dan perangkat lunak (sistem operasi) yang berbeda.

Hal tersebut merupakan terobosan yang cukup besar sebab sebelumnya aplikasi-aplikasi yang dikembangkan untuk sistem operasi (dan perangkat keras) tertentu hanya akan bisa berjalan dengan baik di atas sistem operasi (dan perangkat keras) tertentu pula. Dengan kata lain, di masa lalu, aplikasi-aplikasi yang ditulis di atas sebuah sistem operasi (misalnya, Windows) hampir mustahil bisa berjalan di atas sistem operasi lain (misalnya Linux/UNIX). Adapun logo Java dapat dilihat pada gambar berikut ini:



Gambar 2.15 Logo Java [RAH09]

Upaya yang dilakukan oleh James Gosling dari Sun Microsytem ini cukup berhasil. Bahasa pemrograman Java saat ini bisa digunakan untuk mengembangkan aplikasi-aplikasi *desktop* (menggunakan JSE [Java Standard Edition]), aplikasi-aplikasi yang berjalan di lingkup *enterprise* dan internet (menggunakan JEE [Java Enterprise Edition]), serta aplikasi-aplikasi yang berjalan di sarana-sarana kompoutasi yang terbatas sumberdaya pemrosesan dan terbatas sumber daya penggunaan memorinya (menggunakan JME [Java Micro Edition]), misalnya telepon seluler, PDA (*Personal Digital Assistant*), dan sebagainya. Selain itu, kecanggihan bahasa pemrograman Java serta kemudahan untuk memperolehnya (dapat diunduh dari jaringan internet secara cuma-cuma) membuat penggunaan bahasa pemrograman Java meluas di berbagai saran komputasi dan tersebar penggunaannya di seluruh dunia.

Seperti yang telah kita ketahui bersama, perkembangan komputasi internet yang berjalan sejak sekitar tahun 1980-an memicu penggunaan bahasa-bahasa pemrograman lintas *platform* sebab aplikasi-aplikasi yang berjalan di jaringan internet harus bisa berjalan dengan baik pada berbagai platform yang berbeda-beda – internet tersusun oleh teknologi yang memilliki *platform* yang beragam. Dengan demikian, bahasa pemrograman Java merupakan bahasa pemrograman berorientasi masa depan atau bisa jadi, Java adalah masa depan itu sendiri.

Dalam hal pengembangan program-program aplikasi, sebenarnya aplikasi-aplikasi yang dikembangkan dengan bahasa-bahasa pemrograman sudah ada sejak tahun 1950-an, namun aplikasi-aplikasi itu pada umumnya dikembangkan tanpa metodologi apapun, semuanya mengandalkan akal sehat dan kreativitas

pemrogram/pengembang aplikasi belaka. Perkembangan berikutnya terjadi sekitar tahun 1970-an pada saat aplikasi-aplikasi yang dikembangkan dengan bahasa pemrograman terstruktur (*structured programming language*), misalnya Pascal dan C, mulai dikembangkan berdasarkan metodologi-metodologi dan perkakas-perkakas (*tools*) yang memang sesuai, misalnya dikembangkan menggunakan DFD (*Data Flow Diagram*) untuk perancangan operasi serta ERD (*Entity Relationship Diagram*) untuk perancangan basis data. Dalam beberapa kasusu umum, pengembangan tersebut masih digunakan oleh para perancang perangkat lunak hingga saat ini.

Dalam aplikasi-aplikasi bisnis, metodologi-metodologi pengembangan aplikasi yang menggunakan bahasa pemrograman terstruktur (*structured programming language*), DFD, dan ERD kurang dapat beradaptasi dnegan kebutuhan dan harapan pengguna (*user's need expectations*). Dengan alasan itu, para pakar di bidang perancangan perangkat lunak pada sekitar tahun 1980-1990 mulai bekerja dengan bahasa pemrograman berorientasi objek (OOP [*Object Oriented Programming*]) seperti C++ dan Java. Dengan demikian, diperlukan metodologi dan perkakas-perkakas (*tools*) yang lebih sesuai. Dalam hal ini, UML (*Unified Modeling Language*) yang merupakan metodologi kolaborasi antara metoda-metoda Booch, OMT (*Object Modeling Technique*), serta OOSE (*Object Oriented Software Engineering*) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk mengadaptasi maraknya penggunaan bahasa berorientasi objek (OOP).

2.8.1 Program di Dalam Java

Dalam sebagian besar bahasa pemrograman, sebuah kode program harus diimplementasi (*compile*) maupun diterjemahkan (*interpret*) [RAH09]. Perangkat lunak yang digunakan untuk melakukan kompilasi kode program disebut dengan *compiler*, sedangkan yang digunakan untuk melakukan interpretasi disebut dengan *interpreter*. Perbedaan antara *compiler* dan *interpreter* adalah terletak pada proses eksekusi kodenya. *Compiler* akan menerjemahkan isi kode program secara keseluruhan, sedangkan *interpreter* akan menerjemahkan dengan cara baris demi baris. Karena proses eksekusinya dilakukan per baris, maka *interpreter* lebih lambat dibandingkan *compiler*.

Java menerapkan dua proses yang telah dijelaskan di atas. Ini berarti bahwa Java berperan sebagai *compiler* dan juga sebagai *interpreter*. Konsepnya sederhana, mula-mula kode program yang ditulis dengan bahasa Java (berekstensi .java) akan dikompilasi menjadi suatu kode objek. Dalam terminologi java, kode objek ini disebut dengan istilah *bytecode*. Jadi, dalam Java, hasil akhir programnya akan berupa *bytecode* (berekstensi .class), bukan *file* .exe. selanjutnya *bytecode* akan dieksekusi baris demi baris oleh *interpreter*. Dengan demikian, proses kompilasi hanya dilakukan sekali, tetapi proses *interpter* akan dilakukan setiap program dieksekusi.

2.8.2 *Bytecode*

Bytecode dapat dianggap sebagai sekumpulan perintah dalam bahasa mesin untuk sebuah JVM (*Java Virtual Machine*). Setiap *interpter* Java, baik yang berupa *development tool* maupun sebuah perambah *web*, merupakan implementasi

dari JVM. Program tidak mungkin dapat dijalankan tanpa di dalam komputer maupun alat lain yang tidak memiliki JVM.

Dengan adanya konsep *bytecode* ini, dalam terminologi Java dikenal adanya istilah "*write once, run anywhere*". Ini berarti bahwa sekali kita menulis program Java dan melakukan kompilasi terhadapnya, maka *bytecode*-nya dapat dijalankandi dalam *platform* manapun, selama *platform* tersebut memilik JVM.

2.8.3 *Applet*

Applet adalah program yang dieksekusi di lingkungan perambah *web* [RAH09]. Dengan kata lain, *applet* adalah program untuk internet. Dalam *swing*, *applet* diimplementasikan dengan kelas JApplet (bukan Applet). Dalam kelas JApplet terdapat sebuah *container* yang berguna untuk menempatkan komponen-komponen lain di dalam *applet* yang bersangkutan. Untuk menjalankan suatu program *applet* perlu digunakan *file* html yang dibuat pada *file* terpisah. Di dalam *file* tersebut kemudian disisipkan *tag* untuk memanggil *applet*.