BAB 2

LANDASAN TEORI

2.1. Permainan (*Game*)

Permainan merupakan sebuah aktivitas rekreasi dengan tujuan bersenang-senang ataupun untuk mengisi waktu luang. Permainan biasanya dilakukan secara sendiri-sendiri ataupun secara kelompok. Permainan sendiri terbagi menjadi 2 bagian yaitu [5]:

1. Permainan Anak-Anak

Banyak permainan yang sering dilakukan oleh anak-anak bersama teman-teman mereka di sekitar rumah mereka. Mereka biasanya berkelompok, senang berlarian, ataupun kadang duduk melingkar untuk memainkan salah satu permainan, dan dari situlah biasanya akan tercipta suatu keakraban diantara mereka. Beberapa permainan dari masa lalu disebut dengan permainan tradisional, sedangkan untuk permainan yang muncul di jaman sekarang ini disebut dengan permainan modern dan biasanya menggunakan peralatan canggih seperti halnya *video game*. Dan berikut beberapa jenis permainan yang bisa dimainkan oleh anak-anak [5]:

a. Congklak
b. Layangan
c. Petak umpet
d. Boneka
e. Adu kelereng
f. Bentengan.

2. Permainan Dewasa

Untuk permainan orang dewasa biasanya memerlukan pemikiran untuk memainkannya. Dan berikut beberapa jenis permainan yang bisa dimainkan oleh orang dewasa [5]:

a. Kartu remi
b. Catur
c. Domino
d. Karambol.

2.1.1 Permainan Video (*Video Game*)

Permainan video (*video game*) adalah permainan yang menggunakan interaksi antarmuka pengguna melalui gambar yang dihasilkan oleh piranti video. Permainan video umumnya menyediakan sistem penghargaan, misalnya *score* yang dihitung berdasarkan tingkat keberhasilan yang dicapai dalam menyelesaikan tugas-tugas yang ada di dalam permainan video tersebut. Sistem elektronik yang digunakan untuk menjalankan permainan video disebut *platform*, contohnya adalah komputer pribadi (*personal computer*) ataupun konsol permainan (*game console*) [5].

1. Permainan komputer (*Computer Game*)

Permainan komputer adalah permainan video yang di mainkan pada komputer pribadi, dan bukan pada konsol permainan. Permainan komputer telah berevolusi dari sistem grafis sederhana menjadi sangat kompleks dan mutakhir.

Permainan *onlin*e (*online game*) adalah jenis permainan video atau permainan komputer dengan menggunakan jaringan internet sebagai media interaksinya.

Permainan *online* terdiri dari dua unsur utama, yaitu *server* dan *client*. *Server* adalah penyedia layanan gaming yang merupakan basis agar *client –client* yang

terhubung dapat memainkan permainan dapat melakukan komunikasi dengan baik. Suatu *server* pada prinsipnya hanya melakukan administrasi permainan dan menghubungkan *client-client*. Sedangkan *client* adalah pengguna permainan dan memakai kemampuan *server*. Contoh permainan online adalah *Ragnarok Online, Risk Your Life, Xian* [5].

2. Konsol Permainan (*Game Console*)

Konsol permainan adalah sebuah mesin elektronik yang dirancang khusus untuk memainkan permainan video. Perangkat keluarannya biasa berupa monitor komputer atau televisi dan alat masukan utamanya berupa sebuah pengendali. Konsol permainan yang pertama kali dibuat adalah *Atari*, kemudian dilanjutkan dengan *Nintendo* yang sukses pada tahun 1985-1989. Konsol permainan modern sekarang ini antara lain adalah *Playstation* buatan perusahaan *Sony*. Untuk *Console* yang berbentuk kecil dan dapat dibawah kemana-mana biasa disebut *Portable Console* [5].

2.1.2 Klasifikasi Permainan

Sejarah *game* tidak lepas dengan namanya *platform*, dari yang awalnya hanya mesin *Arcade* (Mesin ding dong) hingga sampai sekarang ini yang sudah mencapai era *Mobile Games*. Untuk lebih jelasnya, berikut ini adalah klasifikasi *game* berdasarkan jenis *Platform* yang di gunakan [12]:

1. *Arcade Games* yaitu yang sering disebut ding-dong di Indonesia, biasanya berada di daerah / tempat khusus dan memiliki box atau mesin yang memang khusus didesain untuk jenis video games tertentu dan tidak jarang bahkan memiliki fitur yang dapat membuat pemainnya lebih merasa “masuk” dan menikmati. Seperti pistol, kursi khusus, sensor gerakan, sensor injakkan dan stir mobil (beserta transmisinya).
2. *PC Games* (*personal computer*) yaitu video game yang dimainkan menggunakan *Personal Computers*.

3. *Console Games* yaitu video game yang dimainkan menggunakan *console* tertentu seperti *playstation XBOX*, *Nitendo*, dan lan-lain.
4. *Handheld Game* yaitu *game* yang dimainkan di console khusus video game yang dapat dibawa kemana-mana, contoh *nitendo DS dan Sony PSP*.
5. *Mobile Games* yaitu game yang biasa dimainkan khusus untuk *handphone atau PDA*.

Klasifikasi permainan berdasarkan pembatasan usia dari pemainnya, menurut ***ESRB*** (***Entertainment Software Rating Board***) adalah sebagai berikut [9]:

1. ***Early Childhood (3+)*** memiliki isi yang mungkin cocok untuk usia 3 keatas. Tidak mengandung materi yang tidak pantas seperti adegan kekerasan.
2. ***Everyone (6+)*** memiliki konten yang mungkin cocok untuk usia 6 keatas. Judul-judul dalam kategori ini mungkin berisi sedikit kartun, fantasi atau kekerasan ringan dan tidak jarang menggunakan bahasa yang ringan.
3. ***Everyone (10+)*** memiliki konten yang mungkin cocok untuk usia 10 keatas. Judul-judul dalam kategori ini mungkin berisi lebih banyak kartun, fantasi atau kekerasan ringan, bahasa ringan dan juga minimal tema sugestif.
4. ***Teen (13+)*** memiliki konten yang mungkin cocok untuk usia 13 keatas. Judul-judul dalam kategori ini mungkin mengandung kekerasan, tema sugestif, humor kasar, sedikit darah, simulasi perjudian, dan tidak jarang menggunakan bahasa yang kuat.
5. ***Mature (17+)*** memiliki konten yang mungkin cocok untuk orang usia 17 dan lebih tua. Judul-judul dalam kategori ini mungkin berisi intens kekerasan, darah, konten seksual dan bahasa yang kuat. *Mature* (17+) memiliki konten yang mungkin cocok untuk orang usia 17 dan

lebih tua. Judul judul dalam kategori ini mungkin berisi intens kekerasan, darah, konten seksual dan bahasa yang kuat.

6. ***Adults Only (18+)*** memiliki konten yang hanya boleh dimainkan oleh orang-orang 18 tahun keatas. Judul-judul dalam kategori ini mungkin termasuk adegan yang intens, kekerasan dan juga mengandung konten seksual.
7. ***RP* (*Rating Pending*)** telah disampaikan kepada ESRB dan sedang menunggu penilaian akhir. (Simbol ini hanya muncul dalam iklan sebelum permainan dibebaskan).

2.1.3 Sudut Pandang Permainan (*Game View Point*)

Sebuah *game* biasanya mempunyai sudut pandang permainan tersendiri disesuaikan berdasarkan *genre game* yang diambil. Berikut beberapa macam sudut pandang permainan yang biasa digunakan [17]:

1. Side Scrolling

Adalah sudut pandang permainan yang terlihat dari samping dan memungkinkan karakter utama untuk bergerak dari kiri ke kanan serta memungkinkan background pada *game* seolah-olah bergeser mengikuti pergerakan karakter utama.

2. Top Down

Adalah sudut pandang permainan yang memungkinkan karakter utama bermanuver ke empat arah namun cara permainannya sendiri bergeser dari bawah ke atas, dan biasanya *game* yang menggunakan sudut pandang permainan jenis ini adalah *shooter game*.

3. Isometric

Adalah sudut pandang permainan yang memungkinkan permainan terlihat diantara sisi *Side Scrolling* dan juga *Top Down*, dan biasanya diterapkan pada *game* dengan *genre RTS* (*Real Time Strategy*).

2.1.4 Jenis Permainan (*Genre Game*)

Game dari tahun-tahun semakin banyak saja jenisnya, teknologi yang dipakaipun semakin canggih. Dari ratusan atau bahkan ribuan *game* yang telah bermunculan di dunia, *game-game* tersebut dikelompokkan menjadi beberapa *genre*. Pada dasarnya, *genre-genre game* yang beredar di pasaran dapat dikelompokkan sebagai berikut [17]:

a. Advergames

Permainan yang dikembangkan untuk tujuan periklanan.

b. Action Game

Adalah tipe *game* dengan fitur utama berupa banyaknya aksi di mana pemain harus memiliki keterampilan reaksi yang cepat untuk menghindari musuh atau menghindari rintangan. Pengembang game tipe ini perlu memastikan *game* yang dibuat dioptimasi sehingga pemain memiliki pengalaman bermain yang baik, yang tidak terganggu oleh *delay* proses yang lama, contohnya : *Metal Gear Solid*.

c. Adventure Game

Adalah tipe *game* yang umumnya membuat pemain harus berjalan mengelilingi suatu tempat yang terkondisi, seperti sebuah istana, gua yang berkelok, dan planet yang jauh. Pemain melakukan navigasi suatu area mencari pesan-pesan rahasia, memperoleh obyek yang memiliki kemampuan yang bervariasi, bertempur dengan musuh, dan lain-lain. Untuk membuat *game* ini, diperlukan perencanaan yang akurat sehingga memiliki alur cerita yang menarik bagi pemain, contohnya : *Tomb Raider, Prince of Persia, Hitman*, dan lain-lain.

d. Fighting Game

Adalah tipe *game* yang intinya harus menjatuhkan lawan tandingnya, entah itu dengan pukulan, tendangan, combo, maupun dengan jurus special, contohnya : *Tekken, Mortal Kombat, Naruto*, dan lain-lain.

e. FPS (First Person Shooter)

Adalah tipe *game* yang mengunakan sudut pandang orang pertama untuk membidik atau membunuh musuh, sehingga kita hanya melihat tangannya saja dan tidak melihat tubuh karakter yang dimainkan, contohnya : *Call of Duty, Half Life, Crysis*, dan lain-lain.

f. RPG (Role Playing Game)

Adalah tipe *game* yang seringkali berupa multi karakter *game* di mana setiap pemain memiliki karakter dengan kemampuan, kekuatan, dan kelemahan yang spesifik. Para pemain saling berkompetisi, berinteraksi, dan bertempur satu sama lain. Tampilan grafis yang khas untuk setiap karakter pemain ditambah dengan *storyline* yang mendebarkan akan sangat menarik dan memberikan pengalaman yang berbeda di dalam bermain, contohnya : *Final Fantasy, Warcraft*, dan lain-lain.

g. Platform Game

Adalah tipe *game* yang mengharuskan pemain mengarahkan suatu obyek dengan melalui berbagai tahap atau tingkatan area untuk menyerang musuh dan menghindar terhadap serangan. Tipe *game* ini sedikit serupa dengan *action game*, tetapi aksinya tidak secepat *action game*. Teknik *collision detection* sangat sering dimanfaatkan pada tipe *game* ini, contohnya : *Sonic The Hedgehog, Mario Bros, Metal Slug*.

h. Puzzle Game

Adalah tipe *game* yang umumnya membuat pemain menggunakan kemampuan berpikirnya sebagai pengganti keterampilan reaksi yang cepat karena terdapat rahasia yang perlu dipecahkan. *Game* ini lebih bersifat statis dibanding *action game*. Pembuatan *game* tipe ini seringkali ditunjang dengan AI (*Artificial Intelligence*), contohnya : *7 Wonders of The Ancient World.*

i. Racing Game

Adalah tipe *game* yang tujuannya adalah mencapai garis finish dari suatu *race*, dalam *game* ini biasanya pemain dapat memilih dan membeli kendaraan, mendandani, melakukan upgrade mesin , contohnya : *Need For Speed.*

j. Simulation Game

Adalah tipe *game* yang memberikan pengalaman atau interaksi sedekat mungkin dengan keadaan yang aslinya, meskipun terkadang keadaan tersebut masih eksperimen atau bahkan fiktif, tapi ada penekanan khusus pada detail dan pengalaman realistik menggunakan keadaan tersebut, contohnya : *Gran Turismo.*

k. Sandbox Game

Adalah tipe *game* yang umumnya ditujukan untuk menjelajahi suatu kota atau tempat dan bebas berinteraksi dengan objek disekitarnya. Mungkin *game* ini bisa di bilang mirip *RPG* tapi *Sandbox* sangat berbeda dengan *RPGgame*, perbedaannya adalah tidak adanya peningkatan level karakternya, contohnya : *GTA dan Bully*.

l. RTS (Real Time Strategy)

Adalah tipe *game* yang mengharuskan pemainnya menggunakan taktik dan strategi untuk jeli dalam melihat setiap peluang, kelemahan musuh dan bijaksana dalam menggunakan sumber daya yang ada, contohnya : *Command and Conquer.*

m. Sport Game

Adalah tipe *game* yang berupa kompetisi antara dua pemain atau lebih, di mana pemain dapat berupa individual atau tim. Contoh *game* tipe ini antara lain sepakbola, bola basket, tenis, dan bilyard. Tergantung seberapa cepat permainan yang terjadi, aplikasi *game* perlu dioptimalkan, contohnya : *Pro Evolution Soccer.*

2.2 Makanan Tradisional

Kue tradisional atau makanan tradisional telah menjadi bagian dari budaya masyarakat Indonesia hampir semua orang menyukai penganan khas negeri ini, sebagian besar makanan tradisional dapat dijumpai di pasar-pasar, namun tidak semua dapat dijumpai dengan mudah. Pulau jawa adalah salah satu pulau yang memiliki banyak ragam makanan tradisional, mulai dari Jawa Barat, Jawa Tengah, sampai Jawa Timur[10].

2.2.1. Makanan Tradisional Jawa Barat

Jawa Barat adalah sebuah propinsi di tanah Jawa yang memiliki kekayaan kuliner yang sangat beragam. Hampir di setiap kabupaten di Jawa Barat memiliki makanan , minuman atau jajanan khas. Kita pasti sudah mengenal Bandung sebagai surga kuliner, begitu juga denganbogor dan daerh lainnya. Contoh makanan khas Jawa Barat adalah [10] :

1. Ali Agrem

   Ali agrem merupakan makanan sejenis kue tradisional yang terbuat dari tepung beras. Rasanya manis dan enak, bentuknya menyerupai donat tapi lebih kecil. Dinamai Ali Agrem karena bentuknya seperti aliatau cincin yang melingkar. Ali agrem ini merupakan makanan tradisional yang sangat unik.

2. Bandros

   Bandros adalah makanan ringan (cemilan) yang mirip kue pancong (kue pukis) yang terbuat dari terigu campuran tepung beras, kelapa parut dengan bumbu garam dan gula.

3. Ladu

   Makanan atau jajanan ladu ini merupakan suatu penganan yang terbuat dari ketan, makanan tradisional ini berasal dari garut. Untuk pembuatan ladu, bahan dasarnya terbuat dari tepung ketan putih sangria, gula putih, gula aren merah, serta kepala yang telah di parut.

4. Dodol jawadah

   Dodol jawadah adalah makanan khas asal kecamatan Gununghalu kabupaten Bandung Barat, yang terbuat dari bahan beras ketan, gula merah, dan parutan kelapa. Berbeda dengan jenis dodol dari daerah lain, bentuk dodol jawadah menyerupai makanan rollade, bedanya adalah pada jawadah yang di gulung adalah kelapa parut dengan dodol berbahan beras ketan dan gula merah, rasanya pun tidak terlalu manis, tetapi tetap legit.

5. Gula Cakar

   Gula cakar mungkin tidak dikenal di daerah lain dan hanya di kenal di Kabupaten Majalengka. Gulanya berwarna merah berbentuk persegi ukurannya sekitar 3 cm X 4 Cm. Gula tersebut terbuat dari gula putih dicampur soda atau jaman dulu dengan sabun beko yag berwarna kehitaman agar bisa mengembang, tak heran kalau gula ini berlubang namun lubang-lubangnya kecil, selain itu menggunakan zat pewarna yang pada umumya berwarna merah.

2.2.2. Makanan Tradisional Jawa Tengah

Provinsi Jawa Tengah adalah salah satu provinsi yang ada di tanah Jawa dengan ibu kota Semarang. Berbatasan dengan Jawa Barat dari sebelah barat, dengan Jogja dari sebelah selatan dan dengan Jawa Timur di bagian timur, membuat Jawa Tengah menjadi pusatnya Jawa. Di Jawa Tengah pun terdapat banyak makanan tradisional seperti [10] :

1. Getuk

   Getuk adalah makanan ringan yang terbuat dengan bahan utama ketela pohon atau singkong. Getuk merupakan makanan yang mudah ditemukan di Jawa Tengah. Pembuatan getuk dimulai dari singkong di kupas kemudian kukus atau perebusan, setelah matang kemudian ditumbuk atau dihaluskan dengan cara digiling lalu diberi pemanis gula dan pewarna

makanan. Untuk penghidangan biasanya ditaburi dengan parutan buah kelapa.

2. Lanting

   Lanting adalah makanan renyah asli dari Kabupaten Kebumen, terbuat dari singkong, rasanya gurih dan renyah. Bentuk lingkaran, ada juga yang seperti angka delapan. Lanting dibuat dari bahan singkong yang diparut. Singkong tersebut kemudian dibuat menjadi kepalan, lalu dikukus hingga setengah matang, Kemudian digoreng.

3. Nopia

   Nopia dan mino (mini nopia) adalah makanan khas banyumas yang terbuat dari terigu, gula jawa, dan beberapa rempah-rempah. Cara membuat nopia adalah terigu dibuat adonan kemudian dibagi dua, untuk bagian kulit di pisah, bagian isi diberi gula jawa dan rempah-rempah serta perasa misalnya rasa bawang, nangka, coklat, dan durian.

4. Horok-horok

   Sajian kuliner jepara sangat di pengaruhi oleh budaya Jawa yang gemar menyantap makanan yang mengandung santan, salah satu kuliner Jepara yang paling khas adalah Horok-horok. Horok-horok terbuat dari tepung pohon aren, cara pembuatan horok-horok cukup unik yaitu dengan menggunakan sisir rambut dan bentuknya menjadi mirip dengan gabus atau sterofom yang kenyal dan sedikit asin.

5. Carang Madu

   Carang madu atau di kenal juga dengan nama Sarang madu adalah makan khas jepara.

2.2.3. Makanan Tradisional Jawa Timur

Provinsi yang berbatasan dengan Jawa Tengah di bagian barat ini adalah salah satu provinsi yang memiliki banyak sekali kuliner khas, hampir tiap kabupaten atau kota memiliki kuliner khas. Contohnya sebagai berikut [10]:

1. Kue Cucur

   Serabi kocor atau lebih dikenal dengan nama kue cucur, makanan khas jogja ini berbeda dengan serabi lain. Kue cucur dalam pembuatannya tidak ada pemanis sama sekali. Bahan-bahan yang digunakan adalah tepung beras, kelapa muda dan garam.

2. Brem

   Brem adalah makanan yang berasal dari sari ketan yang dimasak dan dikeringkan, merupakan hasil fermentasi ketan hitam yang di ambil sarinya saja yang kemudian di endapkan dalam waktu sekitar sehari semalam.

3. Jerangking

   Jerangking adalah makanan ringan atau camilan yang terbuat dari beras ketan yang dipanggang. Bentuknya segiempat menyerupai bentuk lempeng (kerupuk yang berasal dari beras), namun jerangking mempunyai ukuran yang kecil seperti biskuit *crackers*. Rasanya enak, gurih, dan aroma gosong - gosongnya sangat menggugah selera.

4. Ledre

   Ledre berbentuk gapit (seperti emping gulung) dengan aroma khas pisang yang manis.

5. Pudak

   Pudak dibuat dari tepung beras dibungkus daun aren, santan, variasinya makanan ini ada yang diberi campuran gula merah atu dicampur daun pandan wangi.

6. Bongko

   Bongko kopyor merupakan makanan tradisional yang dibungkus daun pisang. Namanya merupakan singkatan dari bubur nangka dan kelapa kopyor. Selain menyegarkan bongko dipercaya berkhasiat untuk memulihkan stamina. Bongko menggunakan bahan baku yang terbuat dari tepung terigu, buah kelapa, pisang, nangka, santan kelapa dan roti tawar.

2.3 Kabayan

Kabayan merupakan tokoh imajinatif dari budaya sunda yang juga telah menjadi tokoh imajinatif masyarakat umum di Indonesia. Polahnya yang di anggap lucu, polos, tetapi sekaligus cerdas. Cerita-cerita lucu mengenai kabayan di masyarakat Sunda di tuturkan turun temurun secara lisan sejak abad ke-19 sampai sekarang. Seluruh cerita kabayan juga menggambarkan kehidupan sehari-hari masyarakat sunda yang terus berkembang sesuai zaman.

Rassers [1941] menilai, si kabayan adalah tokoh ambivalen. Selain sebagai penghubung dan pewarta dari sang pencipta semesta, juga dinilai sebagai tokoh yang mewakili totalitas dan kekuatan masyarakat yang bersifat membangun, tetapi juga menghambat. Di dalam dirinya sifat ketuhanan dan demonis mewujud menjadi satu. Oleh karena itu, Rassers menganggap si kabayan adalah pahlawan budaya sekaligus tukang tipu [6].

2.4 Kecerdasan Buatan / AI (*Artificial Intelligence*)

Kecerdasan buatan atau dalam bahasa Inggrisnya *Artificial Intelligence* sering disingkat AI merupakan cabang terpenting dalam dunia komputer. Akhir-akhir ini teknologi Ai telah begitu banyak mempengaruhi kehidupan manusia.

AI ialah ilmu dan rekayasa, yang membuat mesin mempunyai intelegensi tertentu khususnya program komputer yang ‘cerdas’ [Jhon McCarthy, 1956]. Intelegensi merupakan bagian kemampuan komputasi untuk mencapai tujuan di

dalam dunia. Ada beberapa macam jenis dan derajat intelegensi untuk manusia, binatang dan beberapa mesin.

AI berhubungan dengan penggunaan komputer untuk melaksanakan tugas atau menyelesaikan masalah yang memerluakan kecerdasan atau kepintaran. Pada umumnya komputer memerlukan aplikasi atau software untuk melaksanakan tugas atau menyelesaikan masalah, maka aplikasi atau software yang berdasarkan teknologi AI diperlukan. Intelegensi perlu dilihat dari berbagai sudut pandang, termasuk pemahaman, inferensi, daya penjelasan, pembelajaran, komunikasi dan sebagainya [14].

2.4.1. Tujuan akhir AI

Menurut Lenat dan Feigenbaum (1992), terdapat sembilan tujuan AI yaitu [14]:

1. Memahami kognisi manusia. Bagaimana manusia menyelesaikan masalah? Coba untuk mendapatkan pengetahuan ingatan manusia yang mendalam, kemampuan *problem-solving*, belajar, membuat keputusan, dan lain-lain.
2. Otomatisai biaya-efektif. Menggantikan manusia dalam tugas-tugas intelegensi.
3. Penguatan intelegensi biaya-efektif. Membangun sistem untuk membantu manusia berpikir lebih baik, lebih cepat, lebih dalam.
4. Intelegensi manusia super. Membangun program yang mempunyai kemampuan untuk melebihi intelegensi manusia.
5. *Problem-solving* umum. Sistem penyelesaian berbagai masalah yang luas.
6. Wacana koheren. Komunikasi dengan manusia mengguanakan bahasa, alami.
7. Otonomi. Mempunyai sistem intelegensi yang beraksi atas inisiatif sendiri.
8. Belajar (induksi). Sistem sebaiknya dapat untuk memperoleh data sendiri, dan tau bagaimana memperolehnya.

9. Informasi. Simpan informasi dan mengetahui bagaimana untuk mengambil informasi.

2.4.2 Karakteristik AI

Karakteristik ini didasarkan pada pandangan bahwa AI ada 4 kategori yaitu [14]:

a. Sistem yang dapat berpikir seperti manusia.
b. Sistem yang dapat beraksi seperti manusia.
c. Sistem yang dapat berpikir secara rasional.
d. Sistem yang dapat beraksi secara rasional.

2.4.3 Perkembangan AI dan Aplikasinya

Ai merujuk pada satu bidang komputer yang berupaya melakukan tugas sama seperti manusia. Dalam keadaan ini, komputer tersebut boleh dikatakan cerdas. Banyak bidang yang melibatkan AI antara lain. Penyelesaian masalah, logika kabur (*fuzzy*), pengenalan bahasa alamiah, robotik, dan sistem pakar. Ai boleh di identifikasikan sebagai kajian yang menyeluruh bagaimana sebuah komputer berupaya melakukan tugas dengan terapan yang lebih baik jika di bandingkan manusia. Kajian tersebut meliputi kebolehan komouter belajar, memahami, membuat keputusan, dan meningkatkan prestasi atas hasil kesalahan yang telah dilakukan sebelumnya [14].

2.4.4 Metode Pencarian (*Searching*)

Pada dasarnya teknik pencarian (*searching*) dapat dibagi menjadi 2 kelompok besar, yaitu pencarian buta (*blind search*) dan pencarian terbimbing (*heuristic search*). Untuk mengukur performa metode pencarian, terdapat empat kriteria yang dapat dilakukan [20] yaitu :

a. *Completeness* : apakah metode tersebut menjamin adanya solusi jika solusinya ada?
b. *Time Complexity* : berapa lama waktu yang diberikan untuk menemukan solusi tersebut.
c. *Space complexity* : berapa banyak memori yang dibutuhkan untuk menemukan solusi tersebut.
d. *Optimaly* : apakah metode tersebut menjamin menemukan solusi terbaik jika terdapat beberapa solusi yang berbeda.

2.4.5 Pencarian *Heuristic* (terbimbing)

Pencarian buta biasanya tidak efisien karena waktu akses dan memori yang di butuhkan cukup besar. Untuk mengatasi hal ini maka perlu ditambahkan suatu informasi pada domain yang bersangkutan sehingga proses pencarian yang baru akan dihasilkan. Pencarian seperti ini disebut sebagai *informed search* atau pencarian *heuristic* (terbimbing) yaitu pencarian berdasarkan panduan.

Teknik pencarian *heuristic* (*heuristic searching*) merupakan suatu strategi untuk melakukan proses pencarian secara selektif dan dapat memandu proses pencarian yang memiliki kemungkinan sukses paling besar, namun dengan kemungkinan mengorbankan kelengkapan (*completeness*).

Untuk menerapkan pencarian *heuristic* diperlukan suatu fungsi *heuristic*. Fungsi *heuristic* adalah aturan-aturan yang digunakan untuk mendapatkan solusi yang diinginkan. Pencarian *Heuristic* terdiri dari : *Generate And Tes* (Pembangkitan dan Pengujian), *Hill Climbing* (Pendakian Bukit) : *Simple Hill Climbing, Steepest-Ascent Hill Climbing, Best First Search*, Algoritma A* (A *Star*), *Simulated Anneliang* [3].

2.4.6 AI yang digunakan

Algoritma A* (A *Star*)Algoritma ini merupakan algoritma *Best First Search* yang menggabungkan *Uniform Cost Search* dan *Greedy Best First Search*. Biaya yang diperhitungkan didapat dari biaya sebenarnya ditambah dengan biaya perkiraan. Dalam notasi matematika dituliskan sebagai : f(n) = g(n) + h(n). Dengan perhitungan biaya seperti ini, algoritma A* (A *Star*) adalah *complete* dan optimal.

Sama dengan algoritma dasar *Best First Search*, algoritma A* (A *Star*) ini juga menggunakan *OPEN* dan *CLOSED*. Terdapat tiga kondisi bagi setiap suksesor yang dibangkitkan, yaitu : sudah berada di-*OPEN*, sudah berada di-*CLOSED*, dan tidak berada di-*OPEN* maupun *CLOSED*.

Pada ketiga kondisi tersebut diberikan penanganan yang berbeda-beda. Jika suksesor sudah berada di-*OPEN*, maka dilakukan pengecekan apakah perlu pengubahan *parent* atau tidak tergantung pada nilai g nya melalui *parent* lama atau *parent* baru. Jika melalui *parent* baru memberikan nilai g yang lebih kecil, maka dilakukan pengubahan *parent*. Jika pengubahan *parent* dilakukan, maka dilakukan pula perbaruan (*update*) nilai g dan f pada suksesor tersebut. Dengan perbaruan ini, suksesor tersebut memiliki kesempatan yang lebih besar untuk terpilih sebagai simpul terbaik (*best node*).

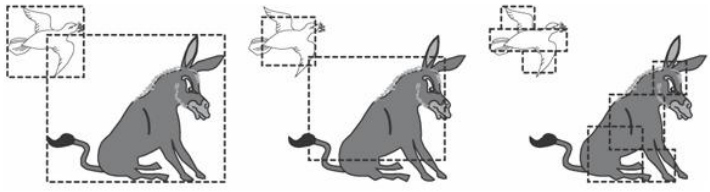
Jika suksesor sudah pernah berada di-*CLOSED*, maka akan dilakukan pengecekan apakah perlu pengubahan parent atau tidak. Jika ya, maka dilakukan perbaruan nilai g dan f pada suksesor tersebut serta pada semua “anak cucunya” yang sudah pernah berada di-*OPEN*. Dengan perbaruan ini, maka anak cucunya tersebut memiliki kesempatan lebih besar untuk terpilih sebagai simpul terbaik (*best node*).

Jika suksesor tidak berada di-*OPEN* maupun *CLOSED*, maka suksesor tersebut dimasukkan ke dalam *OPEN*. Tambahkan suksesor tersebut sebagai suksesornya *best node*. Hitung biaya suksesor tersebut dengan rumus f = g + h[7].

2.4.7 Metode *Collision Detection*

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

Proses *collision* dapat dibagi menjadi dua kategori dasar, yaitu *collision detection* dan *collision response*, dengan jarak respon yang telah diaplikasikan secara spesifik. Dalam *collision detection*, terdapat banyak sekali jenis dari *collision detection* itu sendiri, pada gambar 2.1 merupakan tiga tipe dari *collision detection*.



Gambar 2.1 Tiga jenis *collision detection*

Setiap objek diberi sebuah kotak sebagai acuan jika *collision* itu terjadi. Untuk gambar yang pertama, *collision detection* terjadi ketika objek buruh bertabrakan dengan kotak yang memuat objek keledai, pada jenis *collision detection* ini rentan terhadap ketidaktelitian, untuk gambar yang kedua, *collision detection* terjadi ketika kotak dari kedua objek tersebut bersinggungan, pada jenis *collision detection* ini sudah lebih akurat dibanding jenis yang pertama, namun untuk memilih factor reduksi yang cocok itu sulit, dan untuk gambar yang ketiga, *collision detection* akan terjadi ketika objek saling bersinggungan tanpa memperdulikan kotak secara keseluruhan, pada jenis *collision detection* ini adalah yang paling akurat dibandingkan dengan jenis yang lainnya.

Selain itu pada sumber lain disebutkan bahwa *collision detection* terbagi menjadi dua jenis, yaitu *collision detection object to object* dan *collision detection*

object to world. Pada *object to object* yang terjadi adalah adanya tubrukan antara satu objek dengan objek lainnya yang berada di dalam satu *frame*, baik itu objek tersebut bertubrukan secara langsung ataupun bertubrukan antara kotak maupun lingkaran yang mengandung objek tersebut, sedangkan pada *object to world* yang terjadi adalah adanya tubrukan antara objek dengan *frame* itu sendiri, contohnya adalah tubrukan antara *sprite* dengan dinding pada aplikasi permainan itu sendiri sehingga apabila terjadi, maka *sprite* tersebut tidak dapat melewati dinding tersebut [19].

2.5 Tools yang digunakan

2.5.1. OOP (*Object Oriented Programming*)

Objek adalah kesatuan entitas yang memiliki sifat dan tingkah laku. Dalam kehidupan sehari-hari, objek adalah benda, baik benda berwujud nyata seperti manusia, hewan, mobil, komputer, handphone, pena, ataupun benda yang tidak nyata atau konsep, seperti halnya tabungan bank, sistem antrian, sistem internet banking, dan sebagainya.

Jadi OOP adalah konsep yang membagi program menjadi objek objek yang saling berinteraksi satu sama lain. Objek adalah benda, baik benda yang berwujud nyata maupun benda yang tidak nyata (konsep).

Enam keuntungan menggunakan OOP : Alami (Natural), dapat diandalkan (*Reliable*), dapat digunakan kembali (*Reusable*), mudah untuk dalam perawatan (*Maintainable*), dapat diperluas (*Extendable*), efisiensi waktu.

Berikut ini beberapa bahasa pemrograman yang sudah menggunakan konsep OOP, adalah :

1. *C++*.
2. *Visual C++*.
3. *Visual Basic*.
4. *Java*.

Pemrograman berorientasi objek menekankan pada konsep-konsep berikut :

a. Objek (*Object*)

   Menurut [douglas, 2004: bab 2.1] objek adalah entitas yang memiliki atribut, karakter dan kadang kala disertai kondisi. Objek merepresentasikan sesuatu *system real* seperti siswa, sensor, atau mesin. Objek juga mempresentasikan sesuatu dalam bentuk konsep.

b. Kelas (*class*)

   Kelas adalah penggambaran satu set objek yang memiliki atribut dan *behavior* yang sama. Kelas mirip tipe data pada pemrograman non objek, tetapi lebih komprehensif karena terdapat struktur sekaligus karakteristknya.

c. Pembungkusan (*Encapsulation*)

   [nugroho, 2005: 6] mengartikan pembungkusan sebagai penggabungan potongan-potongan informasi dan perilaku-perilaku spesifik yang bekerja pada informasi tersebut, kemudian mengemasnya menjadi apa yang disebut objek.

d. Pewarisan (*Inheritance*) dan Generalisasi/Spesialisasi

   Merurut [Whitten, 2004: 411], konsep dimana metode dan atau atribut yang ditentukan di dalam sebuah objek kelas dapat diwariskan atau digunakan lagi oleh objek kelas yang lainnya.Sedangkan Generalisai/spesialisasi merupakan teknik dimana atribut dan perilaku yang umum pada beberapa tipe kelas objek dikelompokan ke dalam kelasnya sendiri (dinamakan *supertype*).

e. *Polimorfisme*

   Polimorfisme bererti suatu fungsionalitas yang diimplementasikan dengan berbagai cara yang berbeda. Pada terminologi berorientasi objek, ini berarti dapat memiliki berbagai implementasi untuk sebagian fungsionalitas tertentu [18].

2.5.2 Java

Java adalah bahasa pemrograman yang sederhana, berorientasi-objek, bertipe statik, terkompilasi, *multi-threaded*, *garbage collected*, *netral arsitektur*, *robust*, aman, dan dapat di perluas.

a. Sederhana

   Pengembangan java banyak membuang fitur-fitur yang tidak di perlukan seperti yang dimiliki oleh bahasa pemrograman tingkat tinggi lainnya.

b. Berorientasi-objek

   Seperti halnya C++, java menggunakan kelas-kelas untuk mengelola kode ke dalam modul-modul logika.

c. Bertipe statik.

   Semua objek yang digunakan dalam program harus di deklarasikan sebalum digunakan.

d. Terkompilasi

   Sebelum dapat menjalankan program yang di tulis dengan bahasa pemrograman java, program harus dikompilasi oleh kompilator java.

e. *Multi-treaded*.

   Program java dapat berisi eksekusi *multiple treadeds*, yang memungkinkan program mengulas citra pada layar pada satu *thread*, sementara tetap menerima masukan dari *keyboard* pada *thread* utamanya.

f. *Garbage collected* (pembuang sampah)

   Program java menyingkirkan sendiri ‘sampah-sampah’ yang tak berguna.

g. Bebas arsitektur

   Pada dasarnya, java tidak dirancang untuk prosesor ataupun sistem operasi tertentu.

h. Tangguh (*robust*)

   Karena interpreter java memeriksa seluruh akses sistem yang dilakukan program, maka program java tidak akan membuat sistem menjadi crash.

i. Aman

Sistem java hanya memverifikasi seluruh akses memori tetapi juga menjamin bahwa tidak ada virus yang ikut pada applet yang sedang berjalan.

j. Dapat diperluas

Program java mendukung metode native yakni fungsi-fungsi yang dituliskan dalam bahasa lain, biasanya C++ [13].

2.5.3 UML (*Unified Modeling Language*)

Unified Modelling Language (UML) yang berarti bahasa pemodelan standar. [Chonoles, 2003: bab 1] mengatakan sebagai bahasa, berarti UML memiliki sintak dan semantik. UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

a. Merancang perangkat lunak.
b. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
c. Menjabarkan system secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
d. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

Blok pembangun utama UML adalah diagram. Beberapa diagram ada yang rinci, dan ada yang bersifat umum. Dengan demikian UML mrupakan alat komunikasi yang konsisten dalam mensuport para pengembang sistem saat ini [18].

A. Evolusi UML

Pada bulan oktober 1994, Jim Rumbaugh, penemu *Object Modelling Technique (OMT)* dan Grady Booch, penemu *Booch Methode* (Metode Booch) bersama-sama menyamakan notasi mereka. Di tahun yang sama Ivar Jacobson (penemu Objectory Method) ikut bergabung. Ketiga orang itu bersama-sama membangun notasi standar OOP untuk software Rational IBM. Jim Rumbaugh banyak member masukan dalam pembuatan notasi dan analisa

UML. Grady Booch merancang secara detail kapsitas UML, sedang Ivar Jacobson berusaha membuat UML cocok dengan model bisinis dan mencoba mengembangkan use case diagram lebih lanjut [18].

B. Diagram-diagram UML

Jenis-jenis Diagram UML yaitu :

a. Diagram kelas. Bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antar muka, kolaborasi, serta relasi-relasi. Pemodelan ini umum dijumpai pada pemodelan sistem berorientasi objek.
b. Diagram Paket. Bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.
c. Diagram *Use-case*. Bersifat statis. Diagram ini memperlihatkan himpunan *use case* dan actor-aktor. Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu system yang dibutuhkan serta diharapkan pengguna.
d. Diagram interaksi dan *Sequence*. Bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu.
e. Diagram komunikasi. Bersifat dinamis. Diagram yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.
f. Diagram *Statechart*. Bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi, kejadian sera aktifitas.
g. Diagram aktivitas. Bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainya dalam suatu sistem.

h. Diagram komponen. Bersifat statis. Diagram komponen ini memperlihatkan organisani serata kebergantungan system / perangkat lunak pada komponen-komponen yang telah ada sebelumnya.
i. Diagram *Deployment*. Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi di jalankan [18].

2.5.4 Greenfoot

Greenfoot merupakan *software* untuk pemrograman OOP (*Object Oriented Programming*), didukung dengan aplikasi grafis berbasiskan pemrograman Java. *Greenfoot* diprakarsai oleh Michael Kolling pada 2003. Tujuan dibuatnya *Greenfoot* ini awalmulanya adalah menarik minat anak-anak untuk belajar memprogram.

Menggunakan *Greenfoot* sama seperti memprogram biasa berbentuk teks. Namun, *software* ini lebih mudah digunakan karena ada perangkat bantu seperti *class browser*, editor, dan *compiler*. *Greenfoot* juga mendukung bahasa Java secara penuh dan perangkat tersebut membuat agar lebih mudah dalam memahami konsep OOP. Dalam *Greenfoot* terdapat proyek yang dinamakan *scenario*. Dalam proyek tersebut, kita bisa membuat dua macam tipe *class* dari *library Greenfoot*, yaitu *class world dan actor*. *World* merupakan latar dari *game* yang akan dibuat. *Class actor* adalah kelas yang berfungsi sebagai karakter pada *game* yang akan kita buat [15].

2.5.5 Adobe Photoshop

Adobe Photoshop adalah *software* yang dibuat oleh perusahaan *Adobe System*, yang di khususkan untuk pengeditan foto atau gambar dan pembuatan effect. Perangkat lunak ini banyak digunakan oleh Fotografer Digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (*market leader*) untuk perangkat lunak pengolah gambar. Meskipun pada awalnya *Photoshop* dirancang untuk menyunting gambar untuk cetakan berbasis-kertas, *Photoshop* yang ada saat ini juga dapat

digunakan untuk memproduksi gambar untuk *World Wide Web*. Beberapa versi terakhir juga menyertakan aplikasi tambahan, *Adobe ImageReady*, untuk keperluan tersebut.

Photoshop juga memiliki hubungan erat dengan beberapa perangkat lunak penyunting media, animasi, dan *authoring* buatan-*Adobe* lainnya. File format asli *Photoshop*, PSD, dapat diekspor ke dan dari *Adobe ImageReady*. *Adobe Illustrator, Adobe Premiere Pro, After Effects dan Adobe Encore DVD* untuk membuat DVD profesional, menyediakan penyuntingan gambar non-linear dan layanan special *effect* seperti *background*, tekstur, dan lain-lain untuk keperluan televisi, film, dan situs web [16].