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 [12]. Permainan sendiri terbagi menjadi 2 bagian yaitu:

1. Permainan Anak-Anak

Banyak permainan yang sering dilakukan oleh anak-anak bersama temanteman mereka di sekitar rumah mereka. Mereka biasanya berkelompok, senang berlarian, ataupun kadang duduk melingkar untuk memainkan salah satu permainan. Dan berikut beberapa jenis permainan yang bisa dimainkan oleh anak-anak :

a. Bentengan.
b. Boneka.
c. Congklak.
d. Goncang Kaleng.
e. Kasti.
f. Layangan.
g. Main Hadang.
h. Petak Umpet.
i. Yoyo.

2. Permainan Dewasa

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

a. Kartu Remi atau Bridge.
b. Catur.
c. Domino.
d. Kuis.
e. Sudoku.

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*) [16].

1. Permainan Komputer (*Computer Game*)

Permainan komputer adalah permainan video yang dimainkan pada komputer pribadi, dan bukan pada konsol permainan maupun mesin ding-dong. Permainan komputer telah berevolusi dari sistem grafis sederhana sampai menjadi sangat kompleks dan mutakhir.

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.

2.1.2 (*Genre Game*)

Game dibagi atas beberapa *genre*, diantaranya yaitu [11]:

a. *Action Shooting* (tembak – menembak). Menembak , memukul , bisa juga tusuktusukan, tergantung cerita dan tokoh di dalamnya.
   Contoh : CS (*Counter Strike*) dan Crysis.

b. *Fighting* (pertarungan). Ada yang mengelompokan video *game fighting* di bagian Aksi, jenis ini memang memerlukan kecepatan *refleks* dan koordinasi mata-tangan.
   Contoh : Mortal Kombat dan Tekken.

c. *Adventure* (Petualangan). Memasuki gua bawah tanah, melompati bebatuan di antara lahar, bergelayutan dari pohon satu ke pohon lain.
   Contoh : Kings Quest, dan Space Quest.

d. *Strategy* (strategi). Video *game* strategi biasanya memberikan pemain atas kendali tidak hanya satu orang tapi minimal sekelompok orang dengan berbagai jenis tipe kemampuan.

e. *Simulation* (Simulasi). Video *game* jenis ini seringkali menggambarkan dunia di dalamnya sedekat mungkin dengan dunia nyata dan memperhatikan dengan detil berbagai faktor.

Puzzle (teka-teki). Video *game* jenis ini sesuai namanya berintikan mengenai pemecahan teka-teki, baik itu menyusun balok, menyamakan warna bola.

f. *Sport game* (Olahraga). *Game* ini merupakan adaptasi dari kenyataan, membutuhkan kelincahan dan juga strategi dalam memainkannya.

g. *RPG (Role Playing Game).* Video *game* jenis ini sesuai dengan terjemahannya, bermain peran, memiliki penekanan pada tokoh/peran perwakilan pemain di dalam *game,* yang biasanya adalah tokoh utamanya, dimana seiring kita memainkannya.

2.1.3 Klasifikasi Permainan (*Game Classification*)

Berikut ini adalah klasifikasi permainan berdasarkan pembatasan usia dari pemainnya, menurut ESRB (*Entertainment Software Rating Board)* adalah sebagai berikut : [10]

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.

6 *Only (18+)* orang-orang 18 tahun keatas. Judul-judul dalam kategori ini mungkin termasuk adegan yang *intens*, kekerasan dan juga mengandung konten seksual.

2.2 *First Person Shooter* (FPS)

First-person shooter (FPS) adalah permainan *video* yang berpusat pada permainan di sekitar senjata-senjata dan peluru tempur berbasis melalui sudut pandang orang pertama yaitu: pengalaman pemain aksi melalui mata seorang protagonis. FPS ini dikembangkan pada tahun 1973 Permainan ruang yang belum sempurna flight simulator, yang menampilkan sudut pandang orang pertama [13].

Pada tahun 2000-sekarang merupakan tonggak dari FPS itu sendiri. Pada E3 permainan di tahun 1999, Bungie mengumumkan sebuah real-time strategy permainan yang disebut Halo; di E3 berikut, yang dibongkar-menembak orang-ketiga versi ditampilkan.

Ini juga menerima pujian untuk karakter, baik protagonis, Master Chief dan antagonis asing. Kelanjutannya, Halo 2, membawa popularitas online- ke konsol pasar melalui media Xbox Live, di mana itu permainan sering dimainkan selama hampir dua tahun. Deus Ex, dirilis oleh Ion Storm pada tahun 2000, menampilkan sebuah sistem levelling mirip dengan yang ditemukan pada permainan peran, tetapi juga memiliki banyak cerita, tergantung pada bagaimana pemain menyelesaikan misi dan memperoleh pujian untuk yang serius, gaya artistik. Metroid Prime, dirilis pada tahun 2002 untuk nintendo Cube, konsol lain pertama sangat memuji orang jujur, yang tergabung petualangan tindakan unsur-unsur seperti melompat teka-teki dan dibangun di atas serangkaian Metroid 2D *platformers*. Contoh Fps pada gambar 2.1 fps monster[13].



Gambar 2. 1 Wolfenstein[20].

2.3 Sejarah Bandung Lautan api

Pada awal tahun 1946, Inggris menjanjikan penarikan pasukannya dari Jawa Barat dan menyerahkan Jawa Barat kepada Belanda, yang selanjutnya akan menggunakan sebagai basis militer untuk menghadapi Republik Indonesia. Kesepakatn dua sekutu Inggris dan NICA (Nederlands Indie Civil Administration) Belanda itu memunculkan perlawanan heroic dari masyarakat dan pemuda pejuang di Bandung, ketika tentara Inggris dan NICA melakukan serangan militer ke Bandung. Tentara sekutu berusaha untuk menguasai Bandung, meskipun harus melanggar hasil perundingan dengan Republik Indonesia. Agresi militer Inggris dan NICA Belanda pun memicu tindakan pembumihangusan kota oleh para pejuang dan masyarakat Bandung, peristiwa Bandung Lautan Api bisa di lihat pada gambar 2.2.



Gambar 2. 2 Peristiwa Sejarah Bandung Lautan Api [16].

Tentara Sekutu dan NICA Belanda, yang menguasai wilayah Bandung Utara (wilayah di utara jalan kereta api yang membelah kota Bandung dari timur ke baratt), memberikan ultimatum (23` Maret 1946) supaya Tentara Republik Indonesia (TRI) mundur sejauh 11 km dari pusat kota (wilayah di selatan jalan kereta api dikuasai TRI) paling lambat pada tengah malam tanggal 24 Maret 1946. Tuntutan itu disetujui Pemerintah Republik Indonesia di Jakarta, padahal Markas Besar di Yogyakarta telah memerintahkan TRI untuk mempertahankan setiap jengkal tanah Bandung. TRI dan masyarakat Bandung memutuskan untuk mundur ke selatan, tetapi sambil membumihanguskan Kota Bandung agar pihak musuh tidak dapat memanfaatkannya.

Untuk menghindari jatuhnya korban, maka pukul 16.00 tanggal 24 Maret 1946 Komandan Divisi III mengirimkan pesan supaya pemerintah kota Bandung jam 22.00 harus meninggalkan kota sebab seluruh kota akan dibakar dihancurkan. Perintah pengosongan kota Bandung sebelum jam 22.00 dari Komandan Divisi III, menyebabkan seluruh instansi dan jawatan Republik Indonesia serta penduduk kota Bandung sibuk mempersiapkan diri untuk mengungsi keluar kota Bandung, berat memang yang dirasakan oleh Masyarakat Bandung harus rela meninggalkan rumah, harta benda yang dimilikinya untuk kepentingan perjuangan dan sebagai bentuk loyalitas terhadap Keputusan Pemerintah RI. Maka berbondong-bodonglah

penduduk kota Bandung menuju kearah selatan kota Bandung. Selain dari pemerintah pusat, ternyata datang juga perintah dari Markas Tertinggi TRI yang berkedudukan di Yogyakarta kepada pasukan TRI di Jawa Barat yang pada intinya berisi perintah jangan menyerahkan Bandung begitu saja kepada Sekutu/Inggris.

Tak ayal lagi pada jam 20.00 tanggal 24 Maret 1946, terdengarlah ledakan pertama kemudian disusul ledakan demi ledakan menggelegar di seluruh penjuru kota Bandung. Langit kota Bandung memerah sehingga dinginya udara Bandung tidak terasa lagi. Seluruh kota terasa panas oleh api yang sedang mengamuk membakar kota. Api yang berasal dari tempat yang dibakar itu semula hanya merupakan kobaran api kecil kemudian menjalar ketempat-tempat lainnya, sehingga merupakan riak gelombang dilautan, malam itu Kota Bandung merupakan lautan Api. Selanjutnya Pasukan TRI dan laskar, pemuda, penduduk Kota Bandung mundur kearah Bandung Selatan dan bermarkas di Daerah Kulalet seberang sungai Citarum dan juga Ciparay. Ada pula yang mundur kearah barat sejauh 11 km sampai ke daerah Cililin dan juga kearah Timur Kota Bandung. Dari Bandung Selatan disusun kembali taktik dan Strategi untuk merebut kembali Kota Bandung dengan melakukan serangan-serangan secara seporadis seperti yang dilakukan oleh pemuda Toha dan Moch Ramdan yang berjibaku menghancurkan gudang senjata/munisi di Dayeuh Kolot. Strategi yang diterapkan adalah gerilya di pegunungan disekitar Bandung[8].

2.4 *Game* Edukasi

Sesuai dengan arti bahasa Indonesia, *Game* berarti permainan, sedangkan edukasi adalah pendidikan [1].

Game edukasi adalah salah satu *genre game* yang digunakan untuk memberikan pengajaran / menambah pengetahuan penggunanya melalui suatu media unik dan menarik. Ada beberapa kriteria *game* edukasi yang ideal yaitu:

1. Rasa ingin tahu, fantasi dan kontrol pengguna
    a. Motivasi *intrinsik*, menyenangkan untuk digunakan. Pengguna memiliki kontrol terhadap permainan.
    b. Mendorong keingintahuan, pengguna dapat melakukan eksplorasi

bebas. Permainan mengandung rahasia tersembunyi.

c. Keberhasilan ditentukan oleh pengetahuan, bukan kebetulan.

d. Simulasi realistik dunia.

e. Materi edukasi disesuaikan dengan materi dunia nyata.

2. Tantangan

a. Tantangan diberikan secara terus menerus tapi disesuaikan dengan tingkat pemahaman pemain.

b. Menyediakan hint dan instruksi untuk membantu pengguna.

3. Pengguna anak-anak dan yang berkebutuhan khusus

a. Gambar, objek, layar tertata rapi.

b. Permainannya sendiri merupakan aktivitas yang penting (*play for the sake of play*).

c. Menginspirasi anak, bahkan setelah komputer dimatikan.

2.5 Kecerdasan Buatan (*Artificial Intelligence*)

1. Definisi AI *(Artificial Intelligence)*

Sebagian kalangan menerjemahkan *Artificial Intelligence* sebagai kecerdasan buatan, kecerdasan artifisial, intelijensia artifisial, atau intelejensia buatan. istilah *Artificial Intelligence* ke bahasa Indonesia karena istilah tersebut sudah sangat akrab bagi orang Indonesia. Begitu juga dengan singkatan istilah tersebut, yaitu AI, sudah sangat melekat di berbagai media ilmiah maupun non ilmiah [2].

Para ahli mendefinisikan AI secara berbeda-beda tergantung pada sudut pandang mereka masing-masing. Ada yang focus pada logika berpikir manusia saja, tetapi ada juga yang mendefinisikan AI secara lebih luas pada tingkah laku manusia. 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*

Pendekatan ini dilakukan dengan dua cara sebagai berikut :

a. Melalui introspeksi : mencoba menangkap pemikiran-pemikiran kita sendiri pada saat kita berpikir. Tetapi, seorang psikolog Barat

mengatakan "*how do you know that you understand*?" Bagaimana anda tahu bahwa anda mengerti? Karena pada saat anda menyadari pemikiran anda. Sehingga definisi ini terkesan mengada-ngada dan tidak mungkin dilakukan.

b. Melalui eksperimen-eksperimen psikologi.

2. *Acting humanly : the Turing test approach* ‘

Pada tahun 1950, Alan Turing merancang suatu ujian bagi komputer berintelijensia untuk menguji apakah komputer tersebut mampu mengelabui seorang manusia yang menginterogasinya melalui *teletype* (komunikasi berbasis teks jarak jauh). Jika *interrogator* tidak dapat membedakan yang d2nterogasi adalah manusia atau komputer, maka komputer berintelijensia tersebut lolos dari *Turning test*. Komputer tersebut perlu memiliki kemampuan : Natural *Language Processing, Knowledge Representation, Automated Reasoning, Machine Learning, Computer Vision, Robotics. Turing test* sengaja menghindari interaksi fisik antara *interrogator* dan komputer karena simulasi fisik manusia tidak memerlukan intelijensia.

A. *Thinking rationally : the laws of thought approach*

Terdapat dua masalah dalam pendekatan ini, yaitu :

1. Tidak mudah untuk membuat pengetahuan informal dan menyatakan pengetahuan tersebut ke dalam *formal term* yang diperlukan oleh notasi logika, khususnya ketika pengetahuan tersebut memiliki kepastian kurang dari 100%.
2. Terdapat perbedaan besar antara dapat memecahkan masalah "dalam prinsip" dan memecahkannya "dalam dunia nyata".

B. *Acting rationally : the rational agent approach*

Membuat inferensi yang logis merupakan bagian dari suatu rational agent. Hal ini disebabkan satu-satunya cara untuk melakukan aksi secara rasional adalah dengan menalar secara logis. Dengan menalar secara logis, maka bisa didapatkan kesimpulan bahwa aksi yang diberikan akan mencapai tujuan atau tidak. Jika mencapai tujuan, maka agent dapat melakukan aksi berdasarkan kesimpulan tersebut.

Thinking humanly dan *acting humanly* adalah dua definisi dalam arti yang sangat luas. Sampai saat ini, pemikiran manusia yang diluar rasio, yakni *reflex* dan *intuitif* (berhubungan dengan perasaan), belum dapat ditirukan sepenuhnya oleh komputer. Dengan demikian, kedua definisi ini dirasa kurang tepat untuk saat ini. Jika kita menggunakan definisi ini, maka banyak produk komputasi cerdas saat ini yang tidak layak disebut sebagai produk AI.

Definisi *thinking rationally* terasa lebih sempit daripada *acting rationally.* Oleh karena itu, definisi AI yang paling tepat untuk saat ini adalah acting rationally dengan pendekatan rational agent. Hal ini berdasarkan pemikiran bahwa komputer bisa melakukan penalaran secara logis dan juga bisa melakukan aksi secara rasional berdasarkan hasil penalaran tersebut.

Kecerdasan buatan interaktif semakin berkembang dari waktu ke waktu, hal ini menyebabkan harapan untuk yang berkualitas semakin tinggi. Kemampuan developer untuk membuat yang menarik perlu ditunjang oleh kemampuan untuk menciptakan karakter-karakter dalam yang mampu merespon lingkungannya.

AI (*Artificial Intelligence*) atau kecerdasan buatan merupakan cabang ilmu dari ilmu komputer yang konsern dengan pengautomatisasi tingkah laku cerdas. Pernyataan tersebut juga dapat dijadikan definisi dari AI. Definisi ini menunjukkan bahwa AI adalah bagian dari komputer sehingga harus didasarkan pada *sound theoretical* (teori suara) dan prinsip-prinsip aplikasi dari bidangnya. Prinsip-prinsip ini meliputi struktur data yang digunakan dalam representasi pengetahuan, algoritma yang diperlukan untuk mengaplikasikan pengetahuan tersebut, serta bahasa dan teknik pemrograman yang digunakan dalam mengimplementasikannya.

Namun, definisi di atas belum memadai karena pada kenyataannya istilah kecerdasan itu sendiri tidak didefinisikan atau dipahami dengan sangat baik. Kerap kali kita meyakini telah mengetahui tingkah laku cerdas saat melihatnya sendiri. Akan tetapi, keraguan juga akan muncul ketika orang lain menyatakan kecerdasan dalam cara khusus menurut cara mereka yang cukup membantu dalam evaluasi kecerdasan program komputer. Dengan demikian, belum ada suatu definisi yang cocok untuk *Artificial Intelligence* itu sendiri.

2.6 Metode Logika *Fuzzy Logic*

Algoritma Logika *fuzzy* merupakan salah satu komponen pembentuk *soft computing*. Logika *fuzzy* pertama kali diperkenalkan oleh Prof.Ltfi A. Zadeh pada tahun 1965. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Pada teori himpunan *fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai keanggotaan atau derajat keanggotaan atau membership *function* menjadi ciri utama dari penalaran dengan logika *fuzzy* tersebut [4].

Orang yang belum pernah mengenal logika *fuzzy* pasti akan mengira bahwa logika *fuzzy* adalah sesuatu yang amat rumit dan tidak menyenangkan. Namun, sekali seseorang mulai mengenalnya, ia pasti akan sangat tertarik dan akan menjadi pendatang baru untuk ikut serta mempelajari logika *fuzzy*. Logika *fuzzy* dikatakan sebagai logika baru yang lama, sebab ilmu tentang logika *fuzzy* modern dan metodis baru ditemukan beberapa tahun yang lalu, padahal sebenarnya konsep tentang logika *fuzzy* itu sendiri sudah ada pada diri kita sejak lama.

2. 6. 1. Pengertian Logika *Fuzzy Logic*

Logika *fuzzy* adalah suatu metode yang mengadopsi penilaian yang dilakukan manusia terhadap suatu kebenaran. Yang diekspresikan dalam fungsi *continue* dari 0 sampai 1. Berbeda dengan logika klasik yang menyatakan segala hal dalam istilah binary (0 atau 1, ya atau tidak).

1. *Fuzzifikasi*

Merupakan suatu proses untuk mengubah suatu masukan dari bentuk tegas (crisp) menjadi peubah *fuzzy* (variable *linguistik*) yang biasanya disajikan dalam bentuk himpunan-himpunan *fuzzy* dengan fungsi keanggotaannya masing-masing.

2. Evaluasi Kaidah

Merupakan proses pengambilan keputusan (*inference*) yang berdasarkan aturan aturan yang ditetapkan pada basis aturan (*rules base*) untuk menghubungkan antara peubah-peubah *fuzzy* masukan dan peubah *fuzzy* keluaran. Aturan-aturan ini berbentuk IF…THEN. Proses ini berfungsi untuk mencari suatu nilai *fuzzy* output dari *fuzzy* input.

3. *Defuzzifikasi*

Merupakan proses pengubahan besar *fuzzy* yang disajikan dalam bentuk himpunan-himpunan *fuzzy* keluaran dengan fungsi keanggotaannya untuk mendapatkan kembali bentuk tegasnya. Hal ini diperlukan karena *plant* hanya mengenal nilai tegas sebagai besaran sebenarnya untuk regulasi prosesnya. Proses ini berfungsi untuk menentukan suatu nilai *crisp output.*

Secara keseluruhan istilah yang digunakan dalam sistem *fuzzy* adalah sebagaiberikut :

a. *Degree of Membership*

Fungsi dari derajat keanggotaan adalah untuk memberikan bobot pada suatu input yang kita berikan, sehingga input tadi dapat dinyatakan dengan nilai.

Batas dari derajat keanggotaan adalah dari 0 sampai 1.

b. *Scope* (Domain)

Merupakan suatu batas dari kumpulan input tertentu. Misalnya suhu dingin adalah 10-50 derajat, sangat cepat adalah dari 200-500 rpm.

c. Label (Himpunan)

Adalah kata-kata untuk memberikan suatu keterangan pada domain. Contohnya : panas, dingin, cepat, sangat cepat.

d. *Membership Function* (Fungsi Keanggotaan)

Suatu bentuk bangun yang merepresentasikan suatu batas dari domain.

e. *Crisp Input*

Nilai input analog yang kita berikan untuk mencari derajat keanggotaan.

f. *Universe of discourse* (Semesta Pembicaraan)

Batas input yang telah kita berikan dalam merancang suatu *fuzzy* sistem. Batas ini berbeda dengan batas *scope*/domain. *Universe of discourse* adalah batas semua input yang akan diberikan sedangkan *scope*/domain adalah suatu batas yang menentukan bahwa input tersebut dinyatakan panas, dingin, cepat.

2. 6. 2. Himpunan *Fuzzy Logic*

Pada himpunan tegas (*crisp*) nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan, memiliki dua kemungkina [4] , yaitu Himpunan *fuzzy* memiliki 2 atribut, yaitu:

1 *Linguistik*, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA, TUA.

2 Numeris, yaitu suatu nilai (angka) yang menunjukan ukuran dari variabel seperti: 40,25,50,dsb.

Ada beberapa hal yang perlu diketahui dalam memahami system *fuzzy*, yaitu:

a. Variabel *fuzzy*

Variabal *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*. Contoh; umur, temperature, permintaan,dsb

b. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*.

2. 6. 3. Fungsi Keanggotaan

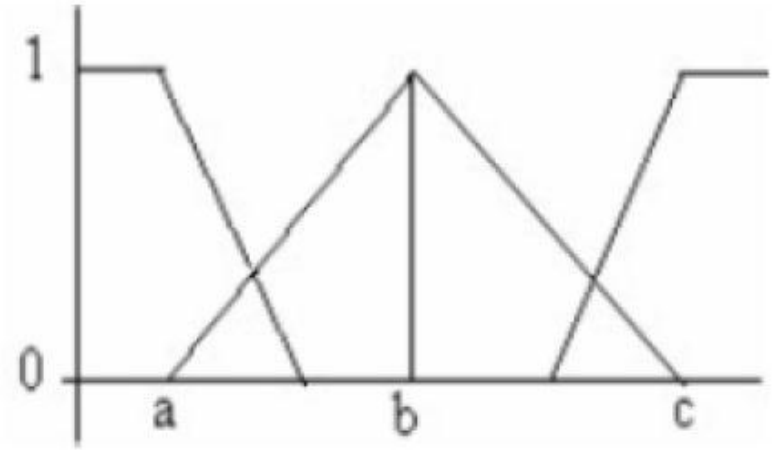
Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering juga disebut derajat keanggotaa) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. beberapa fungsi yang biasa digunakan representasi kurva segitiga, yaitu :

1. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linear). Fungsi keanggotaan . Fungsi keanggotaan. dapat dirumuskan sebagai berikut :

$$\mu(x,a,b,c) = \begin{cases} 0; & x < a \\ (x - a) / (b - a); & a \leq x \leq b \\ (c - x) / (c - b); & b \leq x \leq c \\ 1 & x > c \end{cases}$$

Persamaan tersebut direprsentasikan dalam bentuk grafik dalam gambar di bawah ini :



Gambar 2. 3 Grafik Fungsi Keanggotaan Segitiga [17].

2. 6. 4. Fungsi *Implikasi*

Tiap-tiap aturan (proposisi) pada basis pengetahuan *fuzzy* akan berhubungan dengan suatu relasi *fuzzy*. Bentuk umum dari aturan yang digunakan dalam fungsi *implikasi* adalah :

IF x is A THEN y is B

dengan x adalah skalar, A dan B adalah himpunan *fuzzy*. Proposisi yang mengikuti IF disebut sebagai antiseden sedangkan proposisi yang mengikuti THEN disebut sebagai konsekuen. Secara umum ada 2 implikasi yang dapat digunakan, yaitu :

1. Min (minimum). Fungsi ini akan memotong output himpunan *fuzzy*.
2. Dot (product). Fungsi ini akan menskala output himpunan

2. 6. 5. *Inferensi System* Metode Tsukamoto

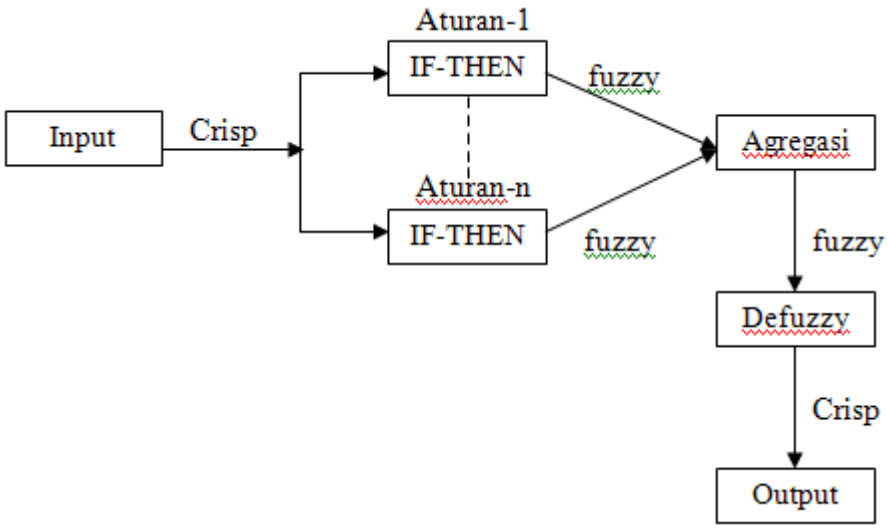
Fuzzy inference system (FIS) atau *fuzzy inferensi engine* adalah sistem yang dapat melakukan penalaran dengan prisip serupa seperti manusia melakukan penalaran dengan nalurinya. Terdapat beberapa jenis FIS yang dikenal yaitu Mamdani, sugeno dan Tsukamoto [4].

Proses dalam kendali logika fuzzy i*nput* yang diberikan adalah berupa bilangan tertentu dan *output* yang dihasilkan juga harus berupa bilangan tertentu. Aturan –aturan dalam bahasa *linguistic* dapat digunakan sebagai *input* yang bersifat teliti harus dikonversikan terlebih dahulu, lalu melakukan penalaran

berdasarkan aturan-aturan dan mengkonversikan hasil penalaran tersebut menjadi *output.*

Pada metode Tsukamoto, setiap konsekuen pada aturan yang berbentuk If-Then harus direpresentasikan dengan suatu himpunan fuzzy dengan fungsi keanggotaan yang monoton. Sebagai hasilnya, *output* hasil inferensi dari tiap-tiap aturan diberikan secara tegas (*crips*) berdasarkan a-predikat (*fire strength*). Hasil akhimya diperoleh dengan menggunakan rata-rata terbobot.

Menurut [18] sistem inferensi fuzzy merupakan suatu kerangka komputasi yang didasarkan pada teori himpunan fuzzy, aturan fuzzy yang berbentuk IF-THEN, dan penalaran fuzzy. Secara garis besar, diagram blok proses inferensi fuzzy terlihat pada Gambar 2.3



Gambar 2. 4 Diagram Blok Sistem *Inferensi Fuzzy* [18].

Sistem inferensi fuzzy menerima input crisp. Input ini kemudian dikirim ke basis pengetahuan yang berisi n aturan fuzzy dalam bentuk IF-THEN. Fire strength (nilai keanggotaan anteseden atau α) akan dicari pada setiap aturan. Apabila aturan lebih dari satu, maka akan dilakukan agregasi semua aturan. Selanjutnya pada hasil agregasi akan dilakukan defuzzy untuk mendapatkan nilai crisp sebagai output sistem. Salah satu metode FIS yang dapat digunakan untuk pengambilan keputusan adalah metode Tsukamoto. Berikut ini adalah penjelasan mengenai metode FIS Tsukamoto.

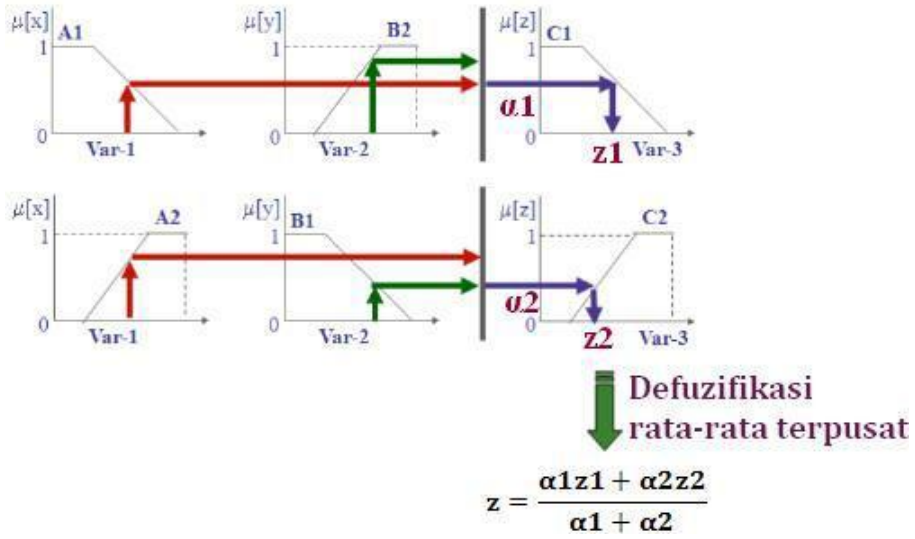
Pada metode Tsukamoto, implikasi setiap aturan berbentuk implikasi "Sebab-Akibat"/Implikasi "Input-Output" dimana antara anteseden dan konsekuen harus ada hubungannya. Setiap aturan direpresentasikan menggunakan himpunan-himpunan fuzzy, dengan fungsi keanggotaan yang monoton. Kemudian untuk menentukan hasil tegas (Crisp Solution) digunakan rumus penegasan (defuzifikasi) yang disebut "Metode rata-rata terpusat" atau "Metode defuzifikasi rata-rata terpusat [18].

Misalkan ada 2 variabel input, Var-1 (x) dan Var-2(x), serta variabel output, Var-3(z), dimana Var-1 terbagi atas 2 himpunan yaitu A1 dan A2. Var-2 terbagi atas 2 himpunan B1 dan B2, Var-3 juga terbagi atas 2 himpunan yaitu C1 dan C2 (C1 dan C2 harus monoton). Ada 2 aturan yang digunakan, yaitu:

[R1] IF (x is A1) and (y is B2) THEN (z is C1)

[R2] IF (x is A2) and (y is B1) THEN (z is C2)

Pertama-tama dicari fungsi keanggotaan dari masing-masing himpunan fuzzy dari setiap aturan, yaitu himpunan A1, B2 dan C1 dari aturan fuzzy [R1], dan himpunan A2, B1 dan C2 dari aturan fuzzy [R2]. Aturan fuzzy R1 dan R2 dapat direpresentasikan dalam Gambar 2.4 untuk mendapatkan suatu nilai crisp Z.



Gambar 2. 5 Inferensi dengan menggunakan Metode Tsukamoto [17].

Karena pada metode Tsukamoto operasi himpunan yang digunakan adalah konjungsi (AND), maka nilai keanggotaan anteseden dari aturan fuzzy [R1]

adalah irisan dari nilai keanggotaan A1 dari Var-1 dengan nilai keanggotaan B1 dari Var- 2. Menurut teori operasi himpunan pada persamaan, maka nilai keanggotaan anteseden dari operasi konjungsi (And) dari aturan fuzzy [R1] adalah nilai minimum antara nilai keanggotaan A1 dari Var-1 dan nilai keanggotaan B2 dari Var-2. Demikian pula nilai keanggotaan anteseden dari aturan fuzzy [R2] adalah nilai minimum antara nilai keanggotaan A2 dari Var-1 dengan nilai keanggotaan B1 dari Var-2. Selanjutnya, nilai keanggotaan anteseden dari aturan fuzzy [R1] dan [R2] masing-masing disebut dengan α1 dan α2. Nilai α1 dan α2 kemudian disubstitusikan pada fungsi keanggotaan himpunan C1 dan C2 sesuai aturan fuzzy [R1] dan [R2] untuk memperoleh nilai z1 dan z2, yaitu nilai z (nilai perkiraan produksi) untuk aturan fuzzy [R1] dan [R2]. Untuk memperoleh nilai output. crisp/nilai tegas Z, dicari dengan cara mengubah input (berupa himpunan fuzzy yang diperoleh dari komposisi aturan-aturan fuzzy) menjadi suatu bilangan pada domain himpunan fuzzy tersebut. Cara ini disebut dengan metode defuzifikasi (penegasan). Metode defuzifikasi yang digunakan dalam metode Tsukamoto adalah metode defuzifikasi rata-rata terpusat yang dirumuskan *Defuzifikasi* rata rata dibawah ini.

$$Z = \frac{a1z1 + a2z2}{a1 + a2}$$

2.7 UNITY

*Unity*3D adalah suatu *tools game engine* untuk membuat *game* 3D atau konten interaktif lainnnya seperti visualisasi arsitektur atau *real time* 3D *animation*. *Game engine* ini *support* pada *Microsoft windows* dan MAC OS X, dan *gamenya* sendiri dapat dijalankan pada *windows, mac, xbox 360, playstation 3, wii, ipad, iphone, dan android*. Unity juga dapat membuat *game* berbasis web browser, dengan menggunakan *plugin* pada browser yaitu *unity* web *player*, yang *support* pada mac dan *windows* saja. *Unity* juga mempunyai kemampuan untuk mengekspor *game* ke *adobe map* 3D dalam *flash*, tetapi beberapa fitur pada web *player* tidak bisa digunakan karena ada beberapa batasan pada *flash*.

Unity sudah termasuk sebuah editor untuk *developing* konten dan *game engine*. *Unity* hampir sama dengan *software game engine* lainnya seperti *director*,

*blender game engin*e, *virtools, torque game builder*, dan *game* studio yang bisa sama-sama terintegrasi antara grafik sebagai *primary method. Unity* mendapatkan penghargaan *the Wall Street Journal 2010 Technology Innovation Award* pada *software category.*

Game engine ini dapat mensuport tiga jenis bahasa pemrograman yaitu Java, C#, dan *Boo*. Banyak *game* yang terkenal dan sudah terbit dipasaran yang menggunakan *game engine i*ni, contohnya adalah : Tiger Woods PGA TOUR Online, *BattleStar Galactica Online, A.I Invation, Sucker Punch,* dan lain-lain[6]



Gambar 2. 6 Logo Unity [6].

2.8 C# (C sharp)

C# (dibaca: C sharp) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka .NET Framework. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain) dengan beberapa penyederhanaan.

Menurut standar ECMA-334 C# Language Specification, nama C# terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan angka # (U+0023). Tanda pagar # yang digunakan memang bukan tanda kres dalam seni musik (U+266F), dan tanda pagar # (U+0023) tersebut digunakan karena karakter kres dalam seni musik tidak terdapat di dalam keyboard standar Bahasa pemrograman C# akan diterapkan pada engine unity yang memiliki compiler MonoDevelop yang sudah termasuk salah satu bagian dari unity [3].

2.9 UML (Unified Modeling Language)

Pada perkembangan teknik pemrograman berorientasi objek, muncul sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu Unified Modelling Language (UML). Adapun pengertian dari UML adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. Dalam hal ini UML merupakan suatu bahasa visual untuk melakukan pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Bahasa pemrograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbangan yang besar pada developer pengembang bahasa pemrograman berorientasi objek selanjutnya.

Perkembangan aktif dari pemrograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemrograman Smalltalk pada awal 1980-an yang kemudian d2kuti dengan perkembangan bahasa pemrograman berorientasi objek yang lainnya seperti C objek, C++, Eiffel, dan CLOS. Secara actual, penggunaan bahasa pemrograman berorientasi objek pada saat itu masih terbatas, namun telah banyak menarik perhatian di saat itu. Sekitar lima tahun setelah Smalltalk berkembang, maka berkembang puka metode pengembangan berorientasi objek. Metode yang pertama diperkenalkan oleh Sally Shalaer dan Edward Yourdon, d2kuti oleh Grady Booch, James R. Rumbaugh, Michael R. Blaha, William Lorensen, Frederick Eddy, William Premerlani, dan masih banyak lagi.

Karena banyaknya metodologi-metodologi yang berkembang pesat saat itu, maka muncullah ide untuk membuat sebuah bahasa yang dapat dimengerti semua

orang. Usaha penyatuan ini banyak mengambil dari metodologi-metodologi yang berkembang pada saat itu. Maka dibuat bahasa yang merupakan gabungan dari beberapa konsep seperti konsep Object Modelling Technique (OMT) dari Rumbaugh dan Booch, konsep The Classes, Responsibilities, Collaborators (CRC) dari Rebecca Wirfs-Brock, konsep pemikiran Ivar Jacobson, dan beberapa konsep lainnya dimana James R. Rumbaugh, Grady Booch, dan Ivan Jacobson bergabung dalam sebuah perusahaan yang bernama Rational Software Corporation menghasilkan bahasa yang disebut dengan Unified Modeling Language (UML) [5].



Gambar 2. 7 Logo UML [16].

1. Diagram

   Diagram adalah representasi grafis dari himpunan elemen. Diagram UML terdapat 9 diagram yaitu :

   a. *Diagram Kelas*. Bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek.

   b. Use case diagram. Bersifat statis. Diagram ini memperlihatkan himpunan use case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan.

c. *Sequence Diagram.* Bersifat dinamis. Diagram urutan adalah interaksi yang menekankan pada pengiriman pesan (message) dalam suatu waktu tertentu.
d. *Colaboration Diagram*. Bersifat dinamis. Diagram kolaborasi adalah diagram interaksi yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.
e. *Statechart Diagram*. Bersifat dinamis. Diagram state ini memperlihatkan *statestate* pada *system*, memuat *state*, transisi, *event*, serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem yang reaktif.
f. *Activity Diagram.* Bersifat dinamis. Diagram aktivitas ini adalah tipe khusus dari digram state yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu system. Diagram ini terutama penting dalam pemodelan fungsi-fungsi.
g. *Component Diagram*. Bersifat statis. Diagram komponen ini memperlihatkakan organisasi serta kebergantungan system/perangkat lunak pada komponen-komponen yang telah ada sebelummnya.
h. *Diployment Diagram.* Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan saat run time. Diagram ini membuat simpul-simpul (*node*) beserta komponen-komponen yang ada di dalamnya.

UML adalah suatu bahasa yang digunakan untuk menentukan. memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. Ini merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan system.

2.10 BLENDER

Pada tahun 1988 Ton Roosendaal mendanai perusahaan yang bergerak dibidang animasi yang dinamakan NeoGeo. NeoGeo adalah berkembang pesat sehingga menjadi perusahaan animasi terbesar di Belanda dan salah satu perusahaan animasi terdepan di Eropa. Ton Roosendaal selain bertanggung jawab sebagai art director juga bertanggung jawab atas pengembangan software internal [7].



Gambar 2. 8 Logo Blender [15].

Karena tidak ingin Blender hilang ditelan waktu begitu saja, Ton Roosendaal mendirikan organisasi non profit yang bernama Blender Foundation. Tujuan utama Blender Foundation adalah tersu mempromosikan dan mengembangkan Blender sebagai proyek open source. Pada tahun 2002 Blender dirilis ulang dibawah syarat – syarat GNU General Public License.

Blender merupakan OSS (*Open Source Software*) atau istilah lainnya software yang dapat di gunakan di berbagai macam OS (*Operating System*). Ini digunakan untuk dikembangkan secara komersial, tetapi sekarang dirilis di bawah GPL.

Blender adalah perangkat lunak untuk grafis 3 dimensi yang gratis dan populer di kalangan desainer. Blender dapat digunakan untuk membuat animasi 3 dimensi. Perangkat lunak ini juga memiliki fitur untuk membuat permainan. Blender tersedia untuk berbagai sistem operasi.

Blender memiliki fitur sama kuat mengatur dalam lingkup dan kedalaman ke ujung lain tinggi 3D software seperti Softimage | XSI, Cinema 4D, 3ds Max dan Maya.

1. Fitur – fitur Blender :

Model: Obyek 3D tipe, termasuk jerat poligon, permukaan NURBS, Bezier dan kurva B-spline; multiresolusi patung kemampuan; *Modifier stack deformers; model Mesh; Python Scripting*

a) Rigging: Skeleton kode ciptaan; *Skinning; lapisan Bone; B-splines* interpolated tulang
b) Animasi: animasi *editor non-linear*; *Vertex framing* kunci untuk morphing, animasi Karakter berpose editor; deformers animasi, pemutaran Audio; sistem kendala animasi
c) *Rendering: raytracer inbuilt; oversampling*, blor gerak, efek pasca produksi, ladang, non-square pixel, lapisan Render dan melewati; Render baking ke peta UV, Efek termasuk halo, suar lensa, kabut, *vektor motion-blur* proses pasca- , dan proses pasca-defocus; Ekspor naskah untuk penyaji eksternal
d) UV *unwrapping*: Laurent dan metode Berdasarkan Sudut unwrapping; unwreapping berdasarkan jahitan; *falloff proporsional* mengedit peta UV
e) *Shading*: membaur *dan shader specular*; Node editor; hamburan Bawah, shading Tangent; peta Refleksi
f) Fisika dan *Partike*l: sistem *Partikel* dapat dilampirkan ke *mesh* objek; simulator *Fluida; solver Realtime* tubuh lembut.