BAB 2
LANDASAN TEORI

2.1 Algoritma

Algoritma merupakan salah satu cabang ilmu komputer yang membahas prosedur penyelesaian suatu permasalahan. Algoritma adalah urutan langkah-langkah dalam menentukan suatu masalah. Algoritma juga dapat didefinisikan dengan deretan langkah komputasi yang mentransformasikan masukan (*input*) menjadi keluaran (*output*). Jadi algortima adalah serangkaian urutan langkah-langkah atau prosedur untuk menyelesaikan suatu masalah dengan memproses nilai masukan menjadi nilai keluaran.

Algoritma dapat ditulis dalam bentuk *pseudocode* ataupun *flowchart*. *Pseudocode* berasal dari kata *pseudo* yang artinya mirip atau menyerupai dan *code* berarti kode sehingga *pseudocode* adalah kode yang menyerupai kode program sebenarnya. Sedangkan *flowchart* merupakan bentuk gambar atau diagram yang mempunyai aliran satu atau dua arah secara sekuensial.[14]

2.1.1 Kecerdasan Buatan

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 adalah ilmu dan rekayasa, yang membuat mesin mempunyai intelegensi tertentu khususnya program komputer yang “cerdas”. 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[11].

2.1.2 Kecerdasan Buatan pada Game

Kecerdasan buatan pada *game* memungkinkan sebuah sistem komputer untuk memiliki cara berpikir manusia dalam bermain. Contohnya permainan yang memiliki fasilitas orang melawan komputer. Komputer sudah di program sedemikian rupa agar memiliki cara bermain seperti seorang manusia bahkan bisa melebihi seorang manusia.

2.1.3 Logika Fuzzy

Logika Fuzzy adalah metodologi sistem kontrol pemecahan masalah, yang cocok diimplementasikan pada sistem, mulai dari sistem yang sederahana, sistem kecil, *embedded system*, jaringan PC, *multi-channel* atau *workstation* berbasis akurasi data, dan sistem kontrol. [6]

Konsep himpunan Fuzzy memiliki 2 atribut, yaitu :

a. Linguistik, yaitu nama suatu kelompok yang mewakili suatu keadaan tertentu dengan menggunakan bahasa alami, misalnya dingin, sejuk, panas mewakili variable temperatur.
b. Numeris, yaitu suatu nilai yang menunjukan ukuran dari suatu variabel, misalnya 10, 35, 40 dan sebagainya.

Di dalam Struktur Sistem Inferensi Fuzzy terdapat :

a. Basis Pengetahuan Fuzzymerupakan kumpulan *rules*Fuzzy dalam bentuk pernyataan IF…THEN.
b. Fuzzyfikasi adalah proses untuk mengubah *input* sistem yang mempunyai nilai tegas menjadi variabel linguistik menggunakan fungsi keanggotaan yang disimpan dalam basis pengetahuan Fuzzy.

c. Mesin Inferensi merupakan proses untuk mengubah *input* Fuzzydengan cara mengikutiaturan-aturan (IF-THEN *Rules*) yang telah ditetapkan pada basis pengetahuan Fuzzy.
d. Defuzzyfikasi merupakan proses mengubah *output* Fuzzy yang diperoleh dari mesin inferensi menjadi nilai tegas menggunakan fungsi keanggotaan yang sesuai dengan saat dilakukan fuzzyfikasi.

2.1.4 Operator Dasar Zadeh Untuk Himpunan Fuzzy

Seperti halnya himpunan konvensional, adabeberapa operasi yang didefinisikan secara khususuntuk mengkombinasi dan memodifikasi himpunan Fuzzy. Pada Operator Dasar Zadeh nilai keanggotaan sebagai hasil dari operasi 2 himpunan sering dikenal dengan nama *fire strength* atau α–predikat.

Ada 3 operator dasar yang diciptakan oleh Zadeh, yaitu:

a. Operator AND

   Operator ini berhubungan dengan operasi interseksi pada himpunan. α–predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan.

   Rumus α–predikatnya : α–predikat = μAUB = max(μA[x],μB[y])

b. Operator OR

   Operator ini berhubungan dengan operasi union pada himpunan. α–predikat sebagai hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan.

   Rumus α–predikatnya : α–predikat = μAUB = max(μA[x],μB[y])

c. Operator NOT

   Operator ini berhubungan dengan operasi komplemen pada himpunan. α–predikat sebagai hasil operasi dengan operator NOT diperoleh dengan mengurangkan nilai keanggotaan elemen pada himpunan yang bersangkutan dari 1.

   Rumus α–predikatnya :μA’ = 1- μ A[x] dan μB’ = 1- μ B[Y]

2.1.5 Model Mamdani

Metode yang berdasarkan intuisi, dikenal juga dengan nama Metode Max-Min. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun1975.Dalam metode ini, untuk mendapatkan output diperlukan 4 tahapan:

a. Pada pembentukan himpunan Fuzzy variabel *input* maupun *output* dibagi menjadi satu atau lebih himpunan.
b. Aplikasi fungsi implikasi yang digunakan adalah Min.
c. Terdapat tiga metode yang digunakan dalam melakukan inferensi sistem Fuzzy yaitu Metode Max, Metode Additive (SUM), Metode Probabilistik OR
d. *Input* dari defuzzifikasi adalah suatu himpunan yang diperoleh dari komposisi aturan-aturan Fuzzy, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan Fuzzy tersebut.

2.1.6 Fungsi Keanggotaan

Fungsi Keanggotaan (*membership function*) adalah suatukurva yang menunjukkan pemetaan titik-titik input data kedalam nilai keanggotaannya (sering juga disebut denganderajat keanggotaan) yang memiliki interval antara 0sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi.

Ada beberapa fungsi yang bisa digunakan, yaitu :

a. Representasi Linear.
b. Representasi Kurva Segitiga.
c. Representasi Kurva Trapesium.
d. Representasi Kurva Bentuk Bahu.
e. Representasi Kurva-S.

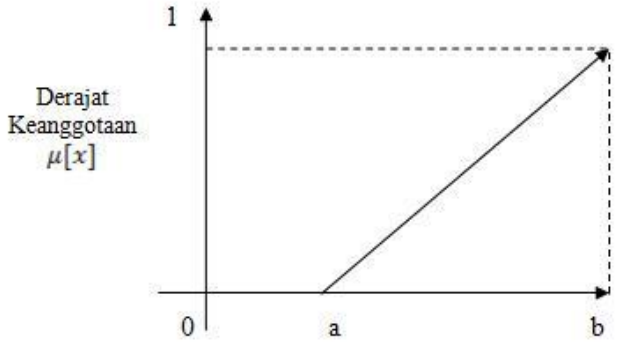
2.1.7 Representasi Linier

Pada representasi linear, pemetaan input ke derajatkeanggotannya digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan menjadi

pilihan yang baik untuk mendekati suatu konsep yang kurang jelas.Ada 2 keadaan himpunan fuzzy linear, yaitu :

a. Representasi Linear Naik

Kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi.



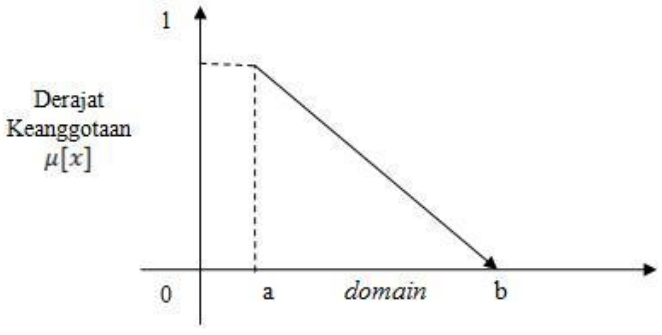
Gambar 2.1 Representasi Linear Naik

Rumus representasi linear naik adalah :

$$\mu[x] = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ 1 & x \geq b \end{cases} \quad \text{.....(1)}$$

b. Representasi Linear Turun

Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah.



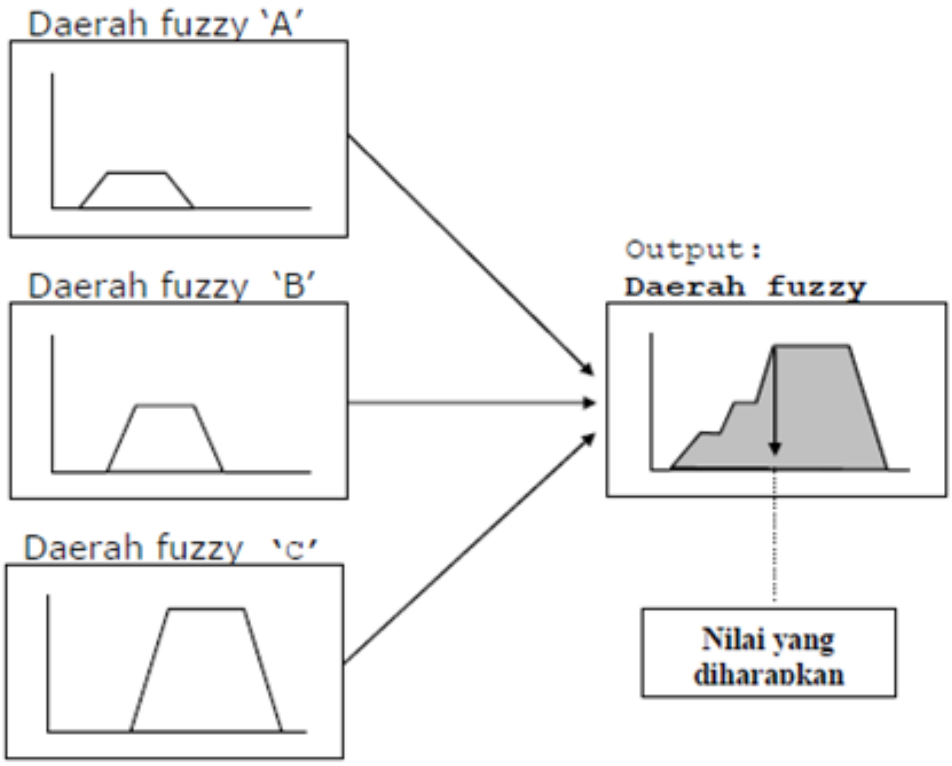
Gambar 2.2 Representasi Linear Turun

Rumus representasi linear turun adalah :

$$\mu[x] = \begin{cases} \frac{b-x}{b-a} if a \leq x \leq b \\ 0 \qquad if\ x \geq b \end{cases} \text{.....(2)}$$

2.1.8 Penegasan (defuzzyfikasi)

Input dari proses *defuzzyfikasi* adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam range tertentu sebagai *output* seperti terlihat pada Gambar berikut :



Gambar 2.3 Proses Komposisi Hasil Implikasi dalam Defuzzyfikasi

2.1.9 Metode Deffuzifikasi Mamdani

Dalam aturan Mamdani terdapat beberapa contoh metode defuzzifikasi seperti [5]:

a. Metode Centroid *(Composite Moment)*

   Metode *Centroid* (Composite Moment). Pada metode centroid solusi *crisp* diperoleh dengan cara mengambil titik pusat daerah *fuzzy*.

b. Metode Bisektor

   Pada metode bisektor solusi *crisp* diperoleh dengan cara mengambil nilai pada domain yang memiliki nilai keanggotaan separuh dari jumlah total nilai keanggotaan pada daerah *fuzzy*.

c. Metode *Mean of Maximun* (MOM)

   Pada metode *mean of maximum* solusi *crisp* diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.

d. Metode *Largest of Maximum* (LOM)

   Pada metode *largest of maximum* solusi *crisp* diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.

e. Metode *Smallest*

   Pada metode *smallest of maximum* solusi *crisp* diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

2.1.10 Metode Centroid

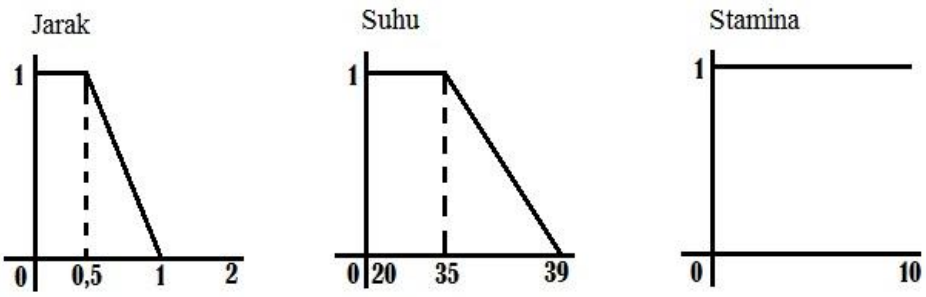
Pada metode ini, solusi crisp diperoleh dengan cara mengambil titik pusat (z*) daerah fuzzy dan secara umum dirumuskan [5]:

$$z* = \frac{total_momen}{total_luas} \quad \text{.....(3)}$$

2.1.11 Gambaran Umum Logika Fuzzy Mamdani Pada Game Menangkap Ular

Berikut ini adalah gambaran Kurva *Membership Function Representation* pada Algoritma yang dibuat dalam *game* ini yang pada himpunannya terdapat 2 atribut [6]:

- Linguistik, yaitu penamaan suatu grup yang mewakili kondisi misalnya jarak jauh, sedang, dan dekat.
- Numeris, yaitu ukuran dari suatu variabel seperti 0,5, 1, 2.



Gambar 2.4 Kurva *Membership Function Representation*

Pada gambar kurva jarak dijelaskan bahwa serangan ular berkisar pada jarak antara 0cm dan 50cm, dan pada saat tangan berada pada jarak 100cm, maka ular sudah merasa waspada terhadap ancaman.

Pada gambar kurva suhu dijelaskan bahwa sensor panas ular dapat mendeteksi keberadaan manusia (ancaman) dengan baik pada suhu 20 hingga 35 derajat celcius. Dan pada suhu 36 derajat celcius keatas maka ular akan merasa bingung dengan keberadaan manusia karena suhu tubuh manusia (36 derajat) tersamarkan oleh suhu lingkungan sekitar.

Dan pada kurva stamina dijelaskan bahwa ular memiliki stamina 10 kali serangan, setelah melakukan 10 kali serangan maka ular akan melarikan diri karena tidak kehabisan stamina.

Ketiga kurva tersebut digabungkan hingga membentuk sebuah *rules* yang berguna sebagai tahap untuk deffuzyfikasi, dan ketiga kurva tersebut berpengaruh pada setiap keputusan ular. *Rules* pada gambar diatas adalah : if jarak & suhu & stamina=1 then attack.

2.2 Game

Di dalam bagian ini akan dibahas tentang *game*, klasifikasi *game* berdasarkan *platform*, genre, serta sudut pandang permainan (*game view point*) dan berdasarkan pembatasan usia dari *user* pada *game* tersebut. [9]

2.2.1 Pengertian Game

Game adalah satu bentuk hiburan yang dijadikan penyegar pikiran dari rasa penat yang disebabkan oleh rutinitas manusia. *Game* sendiri bertujuan untuk menghibur dan banyak disukai oleh bermacam kalangan mulai dari anak-anak hingga orang dewasa sekalipun.

2.2.1.2 Video Game

Game elektronik atau selanjutnya dapat disebut *video game* pertama sekali ditemukan oleh **Thomas T. Goldsmith** dan **Estle Ray Mann**. Penemuan ini dipatenkan pada Januari 1947, yang mendasari perkembangannya saat itu adalah ketika mereka menemukan *Cathode-Ray Tube* sebuah tabung digunakan sebagai media untuk membuat simulasi kecepatan tembakan dan arah tembakan sebuah roket. Pada Februari 1951, **Christopher Strachey** memulai perkembangannya kearah pemrograman yang mulai menggunakan memori dimana aplikasinya diterapkan untuk kebutuhan para pilot. Dan penemuan baru terus berkembang hingga tahun 1959. Namun era perkembangan konsol *game* di mulai pada generasi pertama tahun 1972 sampai tahun 1977. Generasi kedua (1976-1982) sering disebut generasi konsol 8 bit. Di mana pemrograman *video game* dibuat lebih *advance*. Generasi Ketiga (1983-1992), *Nintendo Entertainment System* (NES) mendominasi pasarnya hingga Amerika Utara, *Sega Master System* juga berkembang pesat di Eropa dan Brasil. Generasi Keempat (1987-1996) lebih dikenal dengan generasi 16 Bit dimana pengembangannya dimulai sekitar Oktober 1987. Generasi Kelima (1993-2002), konsol 64 bit mulai muncul pada era ini, grafis 3D mulai menjadi alasan untuk perang konsol ini. Generasi Keenam (1998-2006) merupakan generasi 128 bit, teknologi semakin maju membuat beberapa konsol baru mulai muncul dan semuanya memiliki pasar tersendiri. Generasi Ketujuh (2004) beberapa pengembang konsol mulai fokus dengan pengembangan saja. Mulai tahun 2004 hanya beberapa perusahaan konsol yang masih bertahan dengan pengembangannya *Nintendo, Sony* dan *Microsoft.*

2.2.3 Klasifikasi Game

Game dapat diklasifikasikan kedalam beberapa hal, salah satunya berdasarkan *platform*, *genre*, serta sudut pandang permainan (*game view point*) dan berdasarkan pembatasan usia dari *user* pada *game* tersebut.

2.2.3.1 Platform Game

Terdapat tiga kategori di dalam platform *game*, yaitu *personal computer* (PC), *console*, dan *mobilegame*.

a. *Personal Computer* (PC)

Game PC adalah *game* yang dibangun untuk perangkat komputer yang berbasis Windows, Linux, dan Mac OS. *Game* PC juga disertai kekuatan grafis dan pemrosesan yang kuat dan memungkinkan pengembang dapat membuat *game* yang canggih. Tetapi kelemahannya adalah *game* ini mahal karena pengguna harus memiliki hardware yang mutakhir untuk menjalankan *game* PC dengan baik. Kelemahan *game* PC terletak pada banyaknya varian dari konfigurasi PC yang membuat para pengembang menjadi sulit memastikan *game* itu berjalan dengan baik pada semua setup PC.

b. *Console*

Console adalah perangkat keras yang dibuat oleh pihak ketiga seperti Sony, Nintendo, Microsoft, Sega, Atari. *Console* terhubung ke televisi sebagai output untuk bermain *game*. *Game Console* sangat baik bagi pengembang karena mereka dapat terfokus pada konfigurasi hardware dalam membuat sebuah *software*, sangat berbeda dengan PC yang memiliki banyak opsi konfigurasi.

c. *Mobile Game*

Game ini bersifat *portable* yang meliputi ponsel, PDA, *Handheld game* seperti PSP dan Nintendo Gameboy. *Game mobile* memiliki kontrol yang sederhana jika dibandingkan dengan *game* PC.

2.2.3.2 Genre Game

Menurut buku "Fundamentals of *Game* Design", *genre game* bisa dibagi menjadi 9 jenis, yaitu:

1. *Action*

   Action game adalah *game* dimana kebanyakan dari tantangan yang disajikan merupakan dari tes *physical skill* dan koordinasi pemain. Salah satu sub-*genre action game* adalah *shooters game*, baik yang 2D maupun 3D seperti *First Person Shooters* (FPS).

2. *Strategy*

   Strategy game menantang pemain untuk mencapai kemenangan dengan perencanaan, khususnya melalui perencanaan serangkaian tindakan yang dilakukan melawan satu lawan atau lebih. Kemenangan diraih dengan perencanaan matang dan pengambilan keputusan yang optimal.

3. *Role Playing Game (RPG)*

   RPG adalah *game* dimana pemain mengontrol satu atau lebih karakter yang biasanya di desain oleh pemain itu sendiri, dan memandu mereka melewati berbagai rintangan yang diatur oleh komputer. Perkembangan karakter dalam hal kekuatan dan kemampuannya adalah kunci dari *game* jenis ini.

4. *Sports*

   Sports game mensimulasikan berbagai aspek dari olahraga atletik nyata maupun imajiner, apakah itu memainkan pertandingan, me-manage tim dan karir, atau keduanya. Salah satu contoh *game* jenis ini adalah Pro Evolution Soccer 2012 (PES 2012), dimana pemain bisa memainkan pertandingan, menjadi manajer tim, maupun menjadi pemain dan mengembangkan karirnya sendiri.

5. *Vechicle Simulation*

   Vechicle *simulation* membuat feeling mengendarai kendaraan, baik kendaraan nyata maupun kendaraan imajiner. Performa dan

karakteristik kasar mesin harus menyerupai kenyataan, kecuali jika yang didesain adalah kendaraan imajiner.

6. *Construction and Management Simulation*

   CMS *game* adalah *game* tentang proses. Tujuan pemain bukan untuk mengalahkan musuh, tetapi membangun sesuatu dengan konteks proses yang sedang berjalan. Semakin pemain mengerti dan mengontrol proses, semakin sukses sesuatu yang ia bangun. *Game* seperti ini biasanya menyediakan dua jenis permainan, yaitu mode bebas dimana pemain bebas membangun sesuatu, dan mode misi dimana terdapat skenario hal apa yang harus dibangun oleh pemain.

7. *Adventure/ Platform*

   Adventure atau *platform game* adalah cerita interaktif tentang karakter protagonis yang dimainkan oleh pemain. Penyampaian cerita dan eksplorasi adalah elemen inti dari *game* ini. Penyelesaian teka-teki dan tantangan konseptual adalah bagian besar dari permainan.

8. *Artificial Life and Puzzle Game*

   Artificial Life *game* adalah *game* yang membuat tiruan dari kehidupan sebenarnya. Biasanya ada dua jenis *game* ini, tiruan kehidupan manusia, contohnya The SIMS, dan tiruan kehidupan binatang, contohnya Tamagochi.

9. *Online Game*

   Istilah *online game* disini mengacu kepada *multiplayer game* dimana mesin dari para pemain terhubung dengan jaringan.

2.2.3.3 Game View Point

Sebuah *game* biasanya mempunyai sudut pandang permainan (*game view point*) tersendiri disesuaikan berdasarkan *genre game* yang diambil. Berikut ini adalah beberapa macam sudut pandang permainan yang biasa digunakan :

1. *Side Scrolling*

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*

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 *shootergame*.

3. *Isometric*

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*).

2.2.3.4 Game Berdasarkan Pembatasan Usia dari Pemainnya

Menurut ESRB (Entertainment *Software* Rating Board) ada 7 pembatasan usia dari pemainnya, contohnya sebagai berikut:

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).

2.3 Tools yang digunakan

Tools yang akan mendukung dalam pengimplementasian logika Fuzzy serta pembuatan *Game* Adventure "Menangkap Ular" diantaranya adalah :

2.3.1 Unity3D

Unity 3D adalah sebuah *game engine* yang berbasis *cross-platform*. Unity dapat digunakan untuk membuat sebuah *game* yang bisa digunakan pada perangkat komputer, ponsel pintar android, iPhone, PS3, dan bahkan X-BOX. Unity adalah sebuah sebuah tool yang terintegrasi untuk membuat game, arsitektur bangunan dan simulasi. Unity bisa untuk games PC dan games Online. Untuk games Online diperlukan sebuah plugin, yaitu Unity Web Player. Unity tidak dirancang untuk proses desain atau modelling, dikarenakan unity bukan tool untuk mendesain. Jika ingin mendesain, pergunakan 3D editor lain seperti 3dsmax atau Blender. Banyak hal yang bisa dilakukan dengan unity, ada fitur audio *reverb zone, particle effect*, dan *Sky Box* untuk menambahkan langit.

Fitur *scripting* yang disediakan, mendukung 3 bahasa pemrograman, JavaScript, C#, dan Boo. *Flexible and easy moving, rotating*, dan *scaling objects* hanya perlu sebaris kode. Begitu juga dengan *duplicating, removing*, dan *changing properties*. Visual Properties Variables yang di definisikan dengan *scripts* ditampilkan pada Editor. Bisa digeser, di *drag* and *drop*, bisa memilih warna dengan *color picker*. Berbasis .NET. Artinya penjalanan program dilakukan dengan Open Source .NET platform, Mono.

2.3.2 C#

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.

Standar European Computer Manufacturer Association (ECMA) mendaftarkan beberapa tujuan desain dari bahasa pemrograman C#, sebagai berikut:

1. Bahasa pemrograman C# dibuat sebagai bahasa pemrograman yang bersifat *bahasa pemrograman general-purpose* (untuk tujuan jamak), berorientasi objek, modern, dan sederhana.
2. Bahasa pemrograman C# ditujukan untuk digunakan dalam mengembangkan komponen perangkat lunak yang mampu mengambil keuntungan dari lingkungan terdistribusi.
3. Portabilitas *programmer* sangatlah penting, khususnya bagi *programmer* yang telah lama menggunakan bahasa pemrograman C dan C++.

4. Dukungan untuk internasionalisasi (multi-language) juga sangat penting.
5. C# ditujukan agar cocok digunakan untuk menulis program aplikasi baik dalam sistem klien-server (*hosted system*) maupun sistem embedded (*embedded system*), mulai dari perangkat lunak yang sangat besar yang menggunakan sistem operasi yang canggih hingga kepada perangkat lunak yang sangat kecil yang memiliki fungsi-fungsi terdedikasi.
6. Meskipun aplikasi C# ditujukan agar bersifat 'ekonomis' dalam hal kebutuhan pemrosesan dan memori komputer, bahasa C# tidak ditujukan untuk bersaing secara langsung dengan kinerja dan ukuran perangkat lunak yang dibuat dengan menggunakan bahasa pemrograman C dan bahasa rakitan.
7. Bahasa C# harus mencakup pengecekan jenis (*type checking*) yang kuat, pengecekan larik (*array*), pendeteksian terhadap percobaan terhadap penggunaan Variabel-variabel yang belum diinisialisasikan, portabilitas kode sumber, dan pengumpulan sampah (*garbage collection*) secara otomatis.

2.3.3 UML

Unified Modelling Language (UML) yang berarti bahasa pemodelan standar. UML memiliki sintak dan semantic yang diaplikasikan untuk maksud tertentu, biasanya antara lain untuk [15] :

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.
e. Blok pembangun utama UML adalah diagram. Beberapa diagram ada yang rinci, dan ada yang bersifat umum

2.3.3.1 Diagram-diagram UML

Jenis-jenis Diagram UML yaitu [15]:

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 *use-case* adalah abstraksi dari interaksi antara sistem dan actor, mendeskripsikan bagaimana sistem akan terlihat dimata user.
d. Diagram interaksi dan *Sequence*. Bersifat dinamis. Diagram urutan adalah diagram interaksi yang menampilkan urutan kegiatan dan hubungan kelas.
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.

2.3.4 Adobe Illustrator

Adobe Illustrator adalah program komputer grafis berbasis vektor atau yang lazim disebut *software* ilustrasi. Adobe Illustrator merupakan *software*nomor satu di dunia pilihan pada desainer dan praktisi desain profesional. Aplikasi ini mempunyai banyak kemampuan, diantaranya adalah [10]:

1. Fitur dan fasilitas yang dapat diandalkan untuk mewujudkan pekerjaan kreatif penggunanya.
2. Sangat kompatibel dengan beragam *software* lain untuk berbagai kepentingan akhir, seperti kepentingan cetak, *desktop publishing*, *web* publishing dan lain-lain.
3. Bersifat *user friendly*, yaitu sangat mudah penggunaannya sehingga dapat mengakses beragam fitur yang ada terutama dengan sistem pengelompokan fasilitas melalui menu, *toolbox*, *pallete* dan sebagainya.
4. Mampu menangani beragam pekerjaan desain dari yang sifatnya sederhana hingga amat kompleks, serta mampu mengekspor hasil akhir sebuah desain ke dalam berbagai format sesuai kebutuhan *user* dengan kualitas yang bisa diandalkan.
5. Membantu masalah pewarnaan yang akan diterapkan pada proses cetak.

2.3.5 Corel DRAW

Corel Draw adalah program ilustrasi atau editor grafik vektor yang dikembangkan oleh Corel, sebuah perusahaan perangkat lunak yang bermarkas di Ottawa, Kanada. Versi terbarunya adalah Corel DRAW X6 yg dirilis pada bulan maret 2012. Corel DRAW pada awalnya dikembangkan untuk sistem operasi Windows 2000 dan seterusnya. Versi Corel DRAW untuk Linux dan Mac OS pernah dikembangkan, namun dihentikan karena beberapa alasan tertentu [1].

2.4 PengujianSistem

Pengujian system adalah anomali bagi perekayasa perangkat lunak. Pada proses perangkat lunak, perekayasa berusaha membangun perangkat lunak dari

konsep abstrak ke implementasi yang dapat terlihat, kemudian dilakukan pengujian[9].

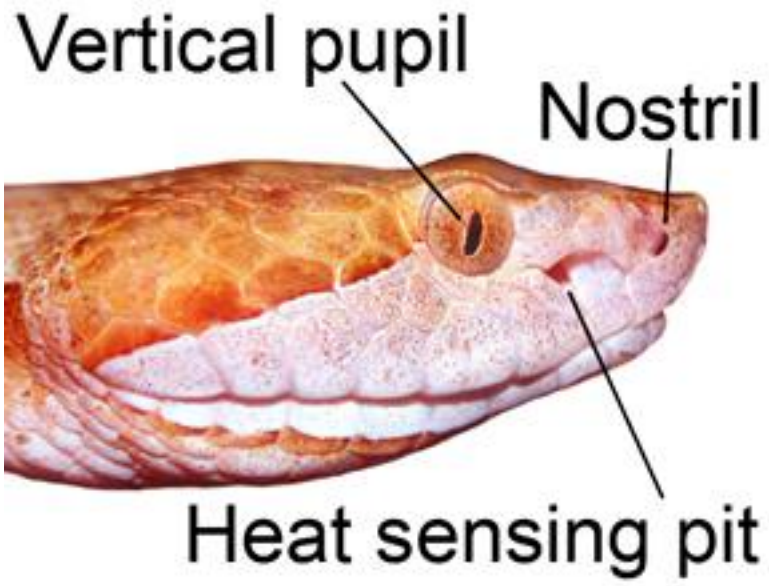
2.4.1 Pengujian Simulasi

Pengujian simulasi adalah suatu pengujian yang ditunjukkan oleh verifikasi dan validasi model, verifikasi adalah proses pemeriksaan apakah logika operasional model (program komputer) sesuai dengan logika kecerdasan buatan dan pemeriksaan apakah program komputer ini berjalan sesuai dengan yang diinginkan (Hoover dan Perry 1989). Proses validasi adalah proses penentuan apakah mode konseptual simulasi adalah representasi akurat dari sistem nyata yang sedang dimodelkan (Law dan Kelton).

Pengujian model simulasi dalam kasus ini ditunjukan untuk menghasilkan kesesuaian nilai *defuzzyfikasi* dari implementasi pada proses *fuzzy* untuk menghasilkan perilaku karakter NPC. Pengujian ini melikupi setiap rule di masing-masing keputusan yang dihasilkan oleh proses defuzzyfikasi seperti menyerang, bertahan, diam, dan melarikan diri.

2.5 Sensor Panas pada Ular

Dikutip dari jurnal karya George S. Bakken dan Aaron R. Krochmal yang berjudul "*The imaging properties and sensitivity of the facial pits of pitvipers as determined by optical and heat-transfer analysis*" [1] dijelaskan bahwa ular viper memiliki sensor panas yang dapat mendeteksi target kecil seperti mangsa pada jarak 0,5 meter dan dapat membedakan serta merespon perbedaan suhu 0,001 derajat celcius pada jarak yang tepat. Membran yang berisi beberapa ribu reseptor tersebut penting dalam membantu penglihatan ular yang kurang dan juga untuk kepentingan berburu ular dalam memilih dan menyergap mangsa serta bertahan dari setiap predatornya di alam. Letak dari sensor panas itu dijelaskan pada gambar 2.5.



Gambar 2.5 Sensor Panas pada Ular