

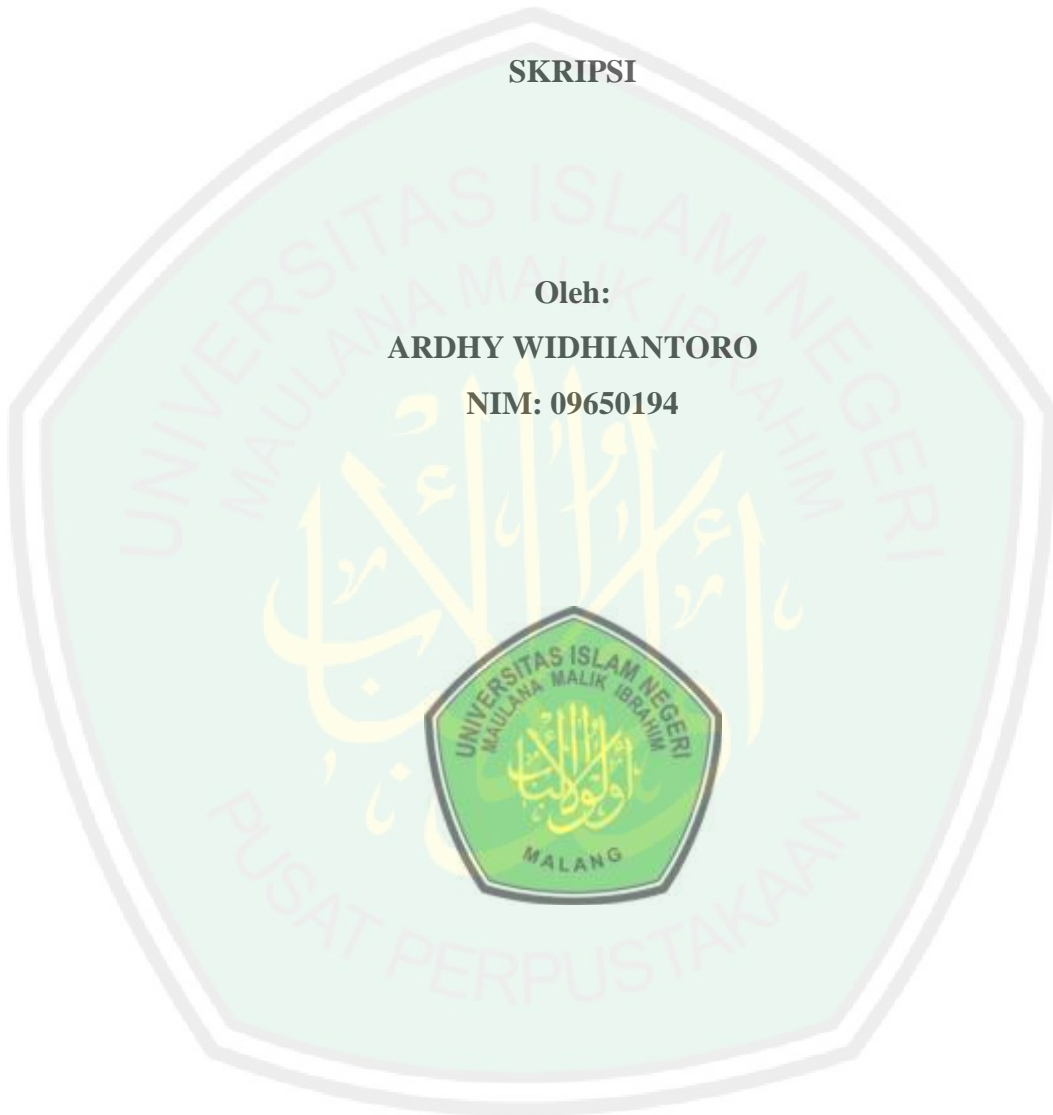
**IMPLEMENTASI FUZZY STATE MACHINE (FuSM) PADA GAME 3D
EDUKASI BAHASA ARAB**

SKRIPSI

Oleh:

ARDHY WIDHIANTORO

NIM: 09650194



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM
MALANG
2013**

**IMPLEMENTASI FUZZY STATE MACHINE (FuSM) PADA GAME 3D
EDUKASI BAHASA ARAB**

SKRIPSI

**Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:
ARDHY WIDHIANTORO
NIM: 09650194**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM
MALANG
2013**

**IMPLEMENTASI FUZZY STATE MACHINE (FuSM) PADA GAME 3D
EDUKASI BAHASA ARAB**

SKRIPSI

Oleh :

Nama : Ardhy Widhiantoro
NIM : 09650194
Jurusan : Teknik Informatika
Fakultas : Sains Dan Teknologi

Telah Disetujui, Juni 2013

Dosen Pembimbing I

Dosen Pembimbing II

Hani Nurhayati, M.T
NIP. 19780625 200801 2 006

M. IMAMMUDDIN, M.A
NIP. 19740602 200901 1 010

Mengetahui,
Ketua Jurusan Teknik Informatika

Ririen Kusumawati, S.Si, M.Kom
NIP. 19720309 200501 2 002

IMPLEMENTASI FUZZY STATE MACHINE (FuSM) PADA GAME 3D EDUKASI BAHASA ARAB

SKRIPSI

Oleh:

ARDHY WIDHIANTORO
NIM. 09650194

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima Sebagai Salah Satu Persyaratan Untuk
Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal, 10 Juni 2013

Susunan Dewan Penguji	Tanda Tangan
1. Penguji Utama : <u>Fresy Nugroho, M.T</u> NIP. 19710722 201101 1 001	()
2. Ketua : <u>Dr. Cahyo Crysdian</u> NIP. 19740424 200901 1 008	()
3. Sekretaris : <u>Hani Nurhayati, M.T</u> NIP. 19780625 200801 2 006	()
4. Anggota : <u>M. Imamuddin, Lc., M.A</u> NIP. 19740602 200901 1 010	()

**Mengetahui dan Mengesahkan,
Ketua Jurusan Teknik Informatika**

Ririen Kusumawati, M. Kom
NIP. 19720309 200501 2 002

SURAT PERNYATAAN
ORISINALITAS PENELITIAN

Saya yang bertandatangan di bawah ini:

Nama : Ardhy Widhiantoro
NIM : 09650194
Fakultas/Jurusan : Sains Dan Teknologi / Teknik Informatika
Judul Penelitian : Implementasi Fuzzy State Machine (FuSM) Pada
Game 3D Edukasi Bahasa Arab

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 10 Juni 2013

Yang Membuat Pernyataan,

Ardhy Widhiantoro

09650194

MOTTO

Jangan Terlalu Bingung Memikirkan Masa Depan, Lakukan Apa Yang Terbaik Saat Ini Dan Sekarang Juga, Jika Kita Terbiasa Melakukan Hal Baik Dengan Segera, Maka Masa Depan Juga Pasti Akan Baik.

وَمَا خَلَقْتُ الْجِنَّ وَالْإِنْسَ إِلَّا لِيَعْبُدُونِ ﴿٥٦﴾

dan aku tidak menciptakan jin dan manusia melainkan supaya mereka mengabdikan kepada-Ku. (Q.S Adz-Dzariat : 56)

PERSEMBAHAN



*Dengan menyebut Asma-Mu yang Agung, syukurku akan segala karunia-Mu,
serta shalawat serta salam kepada Muhammad SAW kekasih-Mu,
Ya Allah, semoga setiap langkah selalu Engkau ridhoi dengan segala rahmat-
Mu*

*Saya Persembahkan karya ini kepada semua pihak yang telah membantu dalam
menyelesaikan karya ini*

*Bapak Agung Saronto, S.Pd dan Ibu Sundari, orang tua hebat yang selalu
menyayangi dan mengasihiku dalam setiap langkah hidupku.*

*Kakakku tercinta Alm. Eko Hariadi yang telah mengantarku mencari ilmu di
kota Malang,
Serta adikku Sectio Dwi Febriantoro yang menyejukkan mata*

Keluarga besar Tamsir dan Hiroastro

*Sahabat dan saudara santri PP.Sabilul Huda Ngasem Jatirejo Nganjuk
Yang selalu menjadi kekuatan dalam semangat*

*Teman, rekan dan Sahabatku UIN Malang,
Khususnya teman-teman jurusan Teknik Informatika 2009
Atim Mufarrihah dan Chakim Annubaha sebagai rekan tim pembuatan game
Yoan Kharisma Bunga yang telah membantu menyelesaikan laporan*

*Teman satu atap di Malang,
Mas Amik, Mas Hanafi, Dian dan Aan.*

Kepada setiap orang yang telah membantu

Terima kasih.

KATA PENGANTAR



Segala puji bagi Allah SWT, karena atas rahmat, hidayah serta karuniaNya, penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Fuzzy State Machine (FuSM) Pada Game 3D Edukasi Bahasa Arab” dengan sebaik-baiknya sebagai salah satu syarat untuk memperoleh gelar sarjana pada program studi Teknik Informatika jenjang Strata-1 Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.

Shalawat serta salam semoga senantiasa Allah limpahkan kepada Nabi Muhammad SAW, keluarga, sahabat dan ahlinya yang telah membimbing umat menuju kebahagiaan dunia dan akhirat.

Penulis menyadari adanya banyak keterbatasan yang penulis miliki, sehingga ada banyak pihak yang telah memberikan bantuan baik moril maupun materiil dalam menyelesaikan skripsi ini. Maka dari itu dengan segenap kerendahan hati patutlah penulis menyampaikan doa dan mengucapkan terima kasih kepada:

1. Prof. DR. H. Mudjia Rahardjo, M.Si, selaku rektor UIN Maulana Malik Ibrahim Malang, yang telah banyak memberikan pengetahuan dan pengalaman yang berharga.
2. Dr. Hj. Bayyinatul Muchtaromah., drh., M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Hani Nurhayati, M.T dan M. Imammuddin, M.A selaku dosen pembimbing I dan II yang telah meluangkan waktu untuk membimbing,

memotivasi, mengarahkan dan memberi masukan dalam pengerjaan skripsi ini.

4. Segenap sivitas akademika Jurusan Teknik Informatika, terutama seluruh dosen, terima kasih atas segenap ilmu dan bimbingannya.
5. Bapak dan Ibuku tercinta, adikku dan seluruh keluarga besar yang senantiasa memberikan doa dan restunya kepada penulis dalam menuntut ilmu serta dalam menyelesaikan skripsi ini.
6. Semua pihak yang tidak mungkin penulis sebutkan satu-persatu, atas segala yang telah diberikan, penulis ucapkan terima kasih yang sebesar-besarnya.

Sebagai penutup, penulis menyadari dalam skripsi ini masih banyak kekurangan dan jauh dari sempurna, untuk itu penulis selalu menerima segala kritik dan saran dari pembaca. Harapan penulis, semoga karya ini bermanfaat bagi kita semua.

Malang, 10 Juni 2013

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGAJUAN.....	ii
HALAMAN PERSETUJUAN.....	iii
HALAMAN PENGESAHAN.....	iv
HALAMAN PERNYATAAN.....	v
HALAMAN MOTTO.....	vi
HALAMAN PERSEMBAHAN.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xiv
DAFTAR TABEL.....	xvii
ABSTRAK.....	xviii
ABSTRACT.....	xix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah.....	5
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	6
1.6 Sistematika Penulisan.....	6
BAB II TINJAUAN PUSTAKA.....	8
2.1 Permainan/Game.....	8
2.2 Adventure Game / Game Petualangan.....	9

2.3	Non Player Character (NPC).....	11
2.4	Pembelajaran dalam Pandangan Islam	13
2.5	Bahasa Arab	14
2.6	Test of Arabic as Foreign Language (TOAFL).....	17
2.7	Logika Fuzzy	17
2.8	Finite State Machine (FSM).....	19
2.9	Fuzzy State Machine (FuSM).....	20
2.10	Pemrograman Lite-C	23
2.11	3D Game Studio	24
2.12	Editor	26
BAB III RANCANGAN DESAIN SISTEM.....		29
3.1	Perancangan Sistem.....	29
3.1.1	Keterangan Umum Game.....	29
3.1.2	Storyboard Game	31
3.1.3	Penampilan Umum Game	32
3.1.4	Deskripsi Karakter	32
3.2	Perancangan Aplikasi	34
3.2.1	Perancangan Antarmuka Intro.....	34
3.2.2	Antarmuka Menu Game	35
3.2.3	Perancangan Antarmuka Game.....	36
3.2.4	Perancangan Antarmuka Penutup	37
3.2.5	Perancangan Permainan pada Bagian Awal.....	38
3.2.6	Perancangan Permainan pada Stage Pertama.....	39
3.2.7	Perancangan Permainan pada Stage Kedua	40
3.2.8	Perancangan Permainan pada Stage Ketiga	42

3.3	Perancangan Fuzzy State Machine Perilaku NPC.....	43
3.3.1	Rancangan Alur Sistem.....	43
3.3.2	Skenario Perilaku Menyerang pada NPC Kalajengking dan NPC Pemburu Kalajengking	44
3.3.3	FSM Perilaku NPC Kalajengking dan NPC Pemburu Kalajengking	45
3.3.4	FuSM Perilaku NPC Kalajengking dan NPC Pemburu Kalajengking.....	47
3.3.5	Top Level Finite State.....	48
3.3.6	Desain Fuzzy NPC 1 (Kalajengking).....	49
3.3.7	Desain Fuzzy NPC 2 (Pemburu Kalajengking)	60
3.4	Kebutuhan Sistem.....	65
3.5	Cara Memainkan Game.....	67
3.6	Kebutuhan PC Pemain.....	67
BAB IV HASIL DAN PEMBAHASAN		69
4.1	Implementasi	69
4.2	Implementasi Fuzzy State Machine (FuSM) Perilaku NPC.....	69
4.2.1	FuSM Respon Perilaku NPC 1 (Kalajengking) Pada Stage Kedua	69
4.2.2	FuSM Respon Perilaku NPC 2 (Pemburu Kalajengking) Pada Stage Kedua	76
4.3	Implementasi Aplikasi Game	79
4.3.1	Antarmuka Intro	80
4.3.2	Antarmuka Menu	81
4.3.3	Scene Game pada Bagian Awal	83
4.3.4	Scene Game pada Stage Pertama	84
4.3.5	Scene Game pada Stage Kedua.....	85

4.3.6	Scene Game saat Kalajengking Lari	86
4.3.7	Scene Game saat Kalajengking Mengejar.....	87
4.3.8	Scene Game saat Kalajengking Menyerang.....	88
4.3.9	Scene Game saat Syekh Al Karim-3 Menyerang Kalajengking	88
4.3.10	Scene Game saat Pemain Menunggangi Kuda.....	89
4.3.11	Scene Game pada Stage Ketiga	90
4.3.12	Antarmuka Video Penutup.....	91
4.3.13	Beberapa Scene Sistem Menampilkan Soal.....	92
4.4	Pengujian Game	96
4.4.1	Rekapitulasi Hasil Pengujian Perbandingan FuSM dan FSM	96
4.4.2	Rekapitulasi Hasil Uji Coba Penggunaan Memori dan CPU.....	97
4.4.3	Rekapitulasi Hasil Kuisisioner	100
BAB V PENUTUP.....		103
5.1	Kesimpulan.....	103
5.2	Saran.....	103
DAFTAR PUSTAKA		105
LAMPIRAN.....		108

DAFTAR GAMBAR

Gambar 2.1 Hirarki gerak perilaku	12
Gambar 2.2 Hierarchy Finite State Machine.....	22
Gambar 2.3 Finite State Machine untuk NPC.....	22
Gambar 2.4 FuSM menyerang untuk NPC	23
Gambar 2.5 Contoh skrip untuk menampilkan kata	24
Gambar 2.6 Contoh skrip untuk menampilkan model 3D	24
Gambar 3.1 Storyboard Game	31
Gambar 3.2 Antarmuka Video Intro	35
Gambar 3.3 Antarmuka Menu Game	35
Gambar 3.4 Antarmuka Game	37
Gambar 3.5 Antarmuka Video Penutup	37
Gambar 3.6 Flowchart permainan bagian awal.....	38
Gambar 3.7 Flowchart Permainan Stage Pertama.....	39
Gambar 3.8 Flowchart Permainan Stage Kedua	41
Gambar 3.9 Flowchart Permainan Stage Ketiga	42
Gambar 3.10 Diagram Blok Alur Sistem.....	44
Gambar 3.11 Perilaku NPC.....	45
Gambar 3.12 Finite State Machine untuk NPC 1.....	46
Gambar 3.13 Finite State Machine untuk NPC 2.....	47
Gambar 3.14 FuSM Menyerang untuk NPC 1	48
Gambar 3.15 FuSM Menyerang untuk NPC 2.....	48
Gambar 3.16 Top Level Finite State Machine	49
Gambar 3.17 Logika Fuzzy untuk menghasilkan perilaku NPC 1.....	50

Gambar 3.18 Desain Fuzzy untuk menghasilkan perilaku NPC 1.....	50
Gambar 3.19 Derajat keanggotaan untuk input jarak terhadap musuh	51
Gambar 3.20 Derajat keanggotaan untuk input kesehatan.....	52
Gambar 3.21 Keanggotaan output perilaku NPC 1.....	54
Gambar 3.22 Fungsi Keanggotaan Nilai Jarak 750	56
Gambar 3.23 Fungsi Keanggotaan Nilai Kesehatan 80	57
Gambar 3.24 Respon Fuzzy Perilaku NPC 1 (Kalajengking).....	60
Gambar 3.25 Logika Fuzzy Untuk Menghasilkan Perilaku NPC 2.....	61
Gambar 3.26 Desain Fuzzy Untuk Menghasilkan Perilaku NPC 2.....	61
Gambar 3.27 Derajat Keanggotaan untuk Input Jarak	61
Gambar 3.28 Keanggotaan Output Perilaku NPC 2.....	63
Gambar 3.29 Fungsi Keanggotaan Nilai Jarak 430	64
Gambar 4.1 Implementasi FuSM Perilaku NPC 1 Kalajengking (Stage Kedua) .	74
Gambar 4.2 Implementasi FuSM Perilaku NPC 2 Pemburu Kalajengking (Stage Kedua).....	78
Gambar 4.3 Video Intro 1	80
Gambar 4.4 Video Intro 2	80
Gambar 4.5 Tampilan pilihan menu.....	81
Gambar 4.6 Menu Keluar.....	82
Gambar 4.7 Menu Cara Bermain	82
Gambar 4.8 Menu Developer	82
Gambar 4.9 Menu Pengaturan.....	82
Gambar 4.10 Menu Mulai	82
Gambar 4.11 Scene pada Bagian Awal.....	83
Gambar 4.12 Scene pada Stage Pertama.....	84

Gambar 4.13 Scene pada Stage Kedua	85
Gambar 4.14 Scene saat Kalajengking Lari	86
Gambar 4.15 Scene saat Kalajengking Mengejar	87
Gambar 4.16 Scene saat Kalajengking Menyerang	88
Gambar 4.17 Scene saat Syeikh AL Karim-3 Menyerang Kalajengking	88
Gambar 4.18 Scene saat Pemain Menunggangi Kuda	89
Gambar 4.19 Scene pada Stage Ketiga	90
Gambar 4.20 Video Penutup Game	91
Gambar 4.21 Skor Game	91
Gambar 4.22 Scene Soal 1	92
Gambar 4.23 Scene Soal 2	92
Gambar 4.24 Scene Soal 3	93
Gambar 4.25 Scene Soal 4	93
Gambar 4.26 Scene Soal 5	93
Gambar 4.27 Scene Soal 6	94
Gambar 4.28 Scene Soal 7	94
Gambar 4.29 Scene Soal 8	95
Gambar 4.30 Scene Soal 9	95
Gambar 4.31 Scene Soal 10	95

DAFTAR TABEL

Tabel 3.1 Variabel linguistik input jarak terhadap musuh NPC 1	51
Tabel 3.2 Variabel linguistik input kesehatan.....	53
Tabel 3.3 Variabel linguistik output NPC 1	54
Tabel 3.4 Aturan Fuzzy perilaku NPC 1	54
Tabel 3.5 Rule Evaluation Fuzzy 1	59
Tabel 3.6 Variabel linguistik input jarak terhadap musuh NPC 2	62
Tabel 3.7 Variabel linguistik output NPC 2.....	63
Tabel 3.8 Fungsi Keyboard dan Mouse untuk Memainkan Game.....	67
Tabel 3.9 Kebutuhan PC Pemain	68
Tabel 4.1 Rekapitulasi Hasil Uji Coba Perbandingan Metode	96
Tabel 4.2 Penggunaan Memori dan CPU Metode FSM	98
Tabel 4.3 Penggunaan Memori dan CPU Metode FuSM	99
Tabel 4.4 Rekapitulasi Hasil Kuisisioner	101

ABSTRAK

Widhiantoro, Ardhy. 2013. **Implementasi Fuzzy State Machine (FuSM) Pada Game 3D Edukasi Bahasa Arab**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Hani Nurhayati, M.T (II) M. Imammuddin, M.A

Kata Kunci: *Game, Bahasa Arab, edukasi, AI, NPC, fuzzy state machine, perilaku menyerang.*

Game 3D merupakan media teknologi yang baik untuk media pembelajaran. Dalam game, pengembangan kecerdasan buatan sangat dibutuhkan untuk mendesain perilaku NPC (Non Player Character). Penelitian ini membahas perubahan perilaku menyerang NPC kalajengking dan NPC pemburu kalajengking pada game yang di atur oleh Fuzzy State Machine (FuSM), sehingga ada interaksi antara karakter pemain, kalajengking dan pemburu kalajengking. Game 3D yang dibuat ini bertujuan sebagai media pembelajaran bahasa Arab dengan konten pembelajaran sesuai soal-soal Test of Arabic as a Foreign Language (TOAFL). Game dibuat menggunakan game engine 3D Game Studio, dan desain FuSM perilaku NPC dibuat menggunakan software MATLAB. Hasil uji coba implementasi FuSM pada perilaku menyerang NPC kalajengking dan pemburu kalajengking sesuai dengan desain output perilaku yang didesain sebelumnya, yaitu melarikan diri, mengejar dan menyerang untuk NPC kalajengking dan berjalan mendekat, berlari mendekat serta menyerang untuk NPC pemburu kalajengking.

ABSTRACT

Widhiantoro, Ardhy. 2013. **Implementation of Fuzzy State Machine (FuSM) In Arabic Educational 3D Games**. Thesis. Informatics Department of Faculty of Science and Technology. Maulana Malik Ibrahim State Islamic University, Malang.

Adviser: (I) Hani Nurhayati, M.T (II) M. Imammuddin, M.A

Keywords: *Game, Arabic, Education, AI, NPC, Fuzzy State Machine, attack behavior*

3D game technology is a good medium for learning media. In the game, the development of artificial intelligence is needed to design the behavior of NPC (Non-Player Character). This research discusses about behavioral change of scorpions NPC and scorpion hunter NPC by Fuzzy State Machine (FuSM), so that there is interaction between the player character, scorpion and scorpion hunter. This game is intended as a medium of Arabic learning with learning content appropriate with questions of Test of Arabic as a Foreign Language (TOAFL). Games created using 3D Game Studio, and FuSM design of NPC behavior made using MATLAB software. Result of FuSM implementation on behavior of scorpion NPC and scorpion hunters accordance with the design output behavior is designed previously, there are run, chasing and attacking for scorpion NPC and walking, run and attacking for scorpion hunter NPC.

BAB I PENDAHULUAN

1.1 Latar Belakang

Bahasa adalah sarana komunikasi yang penting dalam kehidupan sehari-hari. Bagi umat Islam, bahasa Arab menjadi bahasa yang wajib dimengerti karena semua yang berhubungan dengan Islam secara umum menggunakan bahasa Arab, termasuk dalam beribadah.

Al-Quran dan Al-Hadist adalah pedoman hidup yang wajib dipahami oleh seluruh umat Islam. Keduanya menggunakan bahasa Arab dalam penyampaian, sehingga jelas bahwa dalam pemahaman Al-Quran dan Al-Hadist diperlukan pengetahuan bahasa Arab yang baik. Hal ini juga dijelaskan dalam firman Allah :

وَإِنَّهُ لَتَنْزِيلُ رَبِّ الْعَالَمِينَ ﴿١٩٣﴾ نَزَلَ بِهِ الرُّوحُ الْأَمِينُ ﴿١٩٣﴾ عَلَى قَلْبِكَ لِتَكُونَ
مِنَ الْمُنذِرِينَ ﴿١٩٤﴾ بِلِسَانٍ عَرَبِيٍّ مُبِينٍ ﴿١٩٥﴾

Artinya :

Dan Sesungguhnya Al Quran ini benar-benar diturunkan oleh Tuhan semesta alam. Dia dibawa turun oleh Ar-Ruh Al-Amin (Jibril). Ke dalam hatimu (Muhammad) agar kamu menjadi salah seorang di antara orang-orang yang memberi peringatan. Dengan bahasa Arab yang jelas. (QS. Asy-Syu'ara': 192 - 195)

Syekh Al-Ghulayaini (*Jami'ud Durus Al-Arabiyah*, 1980:4) menyimpulkan bahwa “Bahasa itu adalah lafal-lafal yang dengannya setiap bangsa mengungkapkan maksud-maksud mereka”.

Banyak kendala dalam mempelajari bahasa Arab karena kompleksitas bahasa Arab yang cukup rumit dan terbatasnya media untuk memudahkan dalam mempelajarinya. Akan tetapi hal itu menjadi tantangan baru dalam penelitian ini, bagaimana bahasa Arab bisa dipelajari dengan mudah dengan memanfaatkan kemajuan teknologi. Bahasa Arab yang terkenal dengan kerumitan tata bahasa dan kekayaan kosa katanya menjadi tantangan tersendiri bagi masyarakat untuk menguasainya. Tidak hanya kalangan orang awam, bagi santri yang belajar bahasa Arab di pondok pesantren pun mengalami kesulitan (Umi, Rosyidi, 2008 : 8).

Dalam belajar bahasa Arab, para pelajar sering memanfaatkan tes kemampuan bahasa Arab atau *Test Of Arabic As a Foreign Language* (TOAFL) untuk mengetahui kemampuannya dalam penguasaan bahasa Arab. TOAFL merupakan tes untuk menguji kemampuan bahasa Arab dari berbagai sisi, baik kemampuan membaca, mendengar, maupun analisis gramatikal Arab dan tata bahasanya. Tentunya, untuk mengerjakannya membutuhkan kecakapan bahasa Arab yang memadai. Jika tidak memiliki kemampuan tersebut maka dapat dipastikan kemampuan menganalisis dan mengerjakannya tidak maksimal. Oleh karenanya, sangat dianjurkan sebelum mengerjakan TOAFL, untuk memahami gramatikal bahasa Arab dan sering melatih ketangkasan berfikir dengan cara membaca teks-teks Arab (Barmawi : 2011).

Penelitian ini mencoba mewujudkan media pembelajaran tes bahasa Arab dalam media yang menarik dan menyenangkan dengan memanfaatkan teknologi. *Game* 3D bergenre *Adventure* menjadi bentuk media yang dipilih untuk memudahkan dalam pembelajaran bahasa Arab dengan memasukkan soal-soal

TOAFL sebagai konten pembelajaran. Seperti pernyataan Nario Baba bahwa perkembangan teknologi telah banyak mengubah struktur dasar dari perindustrian kita dalam kehidupan sehari-hari. Perindustrian beberapa dekade yang lalu telah membuat kemajuan luar biasa dan berkembang pesat. Salah satu contoh yang paling terlihat adalah industri *game* komputer (Nario Baba : 2007).

Game dalam penelitian ini bercerita tentang persahabatan dua pelajar yang berbeda kewarganegaraan. Pada suatu hari salah satu dari mereka menghilang, akhirnya salah satu yang lain mencarinya ke negara asalnya. Dia menemukan sahabatnya itu berdiri di samping makam, dan ternyata dia menghilang karena ayahnya meninggal terbunuh saat berperang membela negaranya. Dalam pencariannya itu, pemain akan dihadapkan dengan beberapa pertanyaan yang harus dijawab, dan di akhir permainan, pemain akan mengetahui jumlah soal yang dijawab benar dan nilai yang diperoleh. *Game* ini tidak hanya menjawab soal yang terkesan *monotone*, tetapi dalam *game* akan ada beberapa aksi variatif *Non Player Character* (NPC) yang mempengaruhi pemain sehingga permainan bisa lebih menarik dan menyenangkan.

Penelitian sebelumnya pernah dilakukan oleh Miftakhur Rohman (Rohman : 2011) yang membuat *game* 3D edukasi dengan mengimplementasikan *Finite State Machine* (FSM) sebagai aksi perubahan wajah NPC, edukasi yang ditanamkan adalah pembelajaran khusus sekolah dasar. FSM ini ternyata terlalu kaku dan mudah ditebak sehingga permainan terkesan membosankan. Periode berikutnya penelitian lain yang dilakukan oleh Rahma Rizania Nasol (Nasol : 2013) yang mengembangkan *game* 3D edukasi untuk sekolah dasar dengan

implementasi *Fuzzy State Machine* sebagai perilaku NPC musuh menyerang dalam peperangan melawan musuh.

Penelitian kali ini juga akan memakai *Fuzzy State Machine* sebagai kecerdasan *game*, akan tetapi dengan konten edukasi berbeda yaitu pembelajaran bahasa Arab dengan mengacu soal TOAFL. *Fuzzy State Machine* tidak diterapkan pada peperangan yang nantinya cenderung adanya aksi kekerasan, akan tetapi diterapkan pada perilaku NPC kalajengking dan NPC pemburu kalajengking. Perilaku lebih variatif NPC didapatkan dari algoritma *Fuzzy* yang digunakan. *Fuzzy State Machine* sendiri adalah gabungan dari metode *Fuzzy* dan *Finite State Machine* (FSM). *Fuzzy* adalah algoritma yang mengidentifikasi nilai samar atau daerah ketidakpastian dan FSM adalah alur sistem *state per state* yang ada pada *game*. Kelebihan *Fuzzy State Machine* adalah dihasilkannya output perilaku pada NPC yang lebih variatif daripada FSM biasa, jika FSM memiliki output ON dan OFF, maka *Fuzzy State Machine* memiliki *state* diantara ON dan OFF.

Penulis berharap *game* ini bisa digunakan oleh pemain sebagai media pembelajaran bahasa Arab dengan konten pembelajaran mengacu pada soal TOAFL, sehingga bisa lebih mudah dalam memahami bahasa Arab dengan baik yang merupakan bahasa kitab suci Al Quran dan Al-Hadist, hal ini adalah tujuan utama dalam penelitian ini.

1.2 Rumusan Masalah

Bagaimana implementasi *Fuzzy State Machine* (FuSM) sebagai kecerdasan perilaku NPC kalajengking dan NPC pemburu kalajengking sehingga bisa lebih

berperilaku natural dengan cara membangun sebuah *game* 3D edukasi bahasa Arab yang berisi soal-soal TOAFL.

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah:

- a) Permainan ini dimainkan oleh satu pemain/*single player*
- b) Permainan bergenre *Adventure Game*
- c) Bahasa pemrograman yang digunakan adalah *Lite-C*
- d) 3D Engine yang digunakan adalah *3D Game Studio*
- e) *Fuzzy State Machine* (FuSM) diimplementasikan pada NPC kalajengking dan NPC pemburu kalajengking untuk menghasilkan variasi output perilaku terhadap pemain
- f) Permainan ini ada 3 level sesuai dengan tingkat kesulitan
- g) Permainan berisi soal-soal TOAFL dengan aspek ketrampilan memahami kosa kata, memahami gramatika dan struktur kalimat, serta memahami teks atau naskah cerita, dan ditujukan untuk level mahasiswa.
- h) Permainan diimplementasikan pada Komputer Desktop

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah membuktikan kelayakan metode *Fuzzy State Machine* (FuSM) sebagai kecerdasan output perilaku NPC kalajengking dan NPC pemburu kalajengking dengan cara membangun media pembelajaran *game* 3D edukasi bahasa Arab yang menghibur dan menarik.

1.5 Manfaat Penelitian

Manfaat penelitian ini adalah memberikan kontribusi bagi para mahasiswa dalam mempermudah pembelajaran bahasa Arab sesuai soal TOAFL melalui permainan yang lebih menghibur dan menarik. Para pelajar bisa memanfaatkan *game* ini sebagai media latihan tes TOAFL.

1.6 Sistematika Penulisan

Dalam penulisan skripsi ini, secara keseluruhan terdiri dari lima bab yang masing-masing bab disusun dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini merupakan bab pendahuluan, yang di dalamnya memuat latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Pada bab tinjauan pustaka menjelaskan teori yang berhubungan dengan permasalahan penelitian yang meliputi: (1) Permainan / *Game* (2) Pembelajaran dalam Pandangan Islam (3) Bahasa Arab (4) *Finite State Machine* (FSM) (5) *Logika Fuzzy* (6) *Fuzzy State Machine* (FuSM) (7) Pemrograman Lite-C (8) 3D *Game Studio*

BAB III ANALISIS DAN PERANCANGAN SISTEM

Bab ini menjelaskan tentang pembuatan desain dan perancangan *game* 3D edukasi bahasa Arab dengan implementasi *Fuzzy State Machine* sebagai kecerdasan pada NPC yang meliputi metode penelitian yang digunakan, perancangan aplikasi dan desain aplikasi yang akan digunakan.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan hasil yang dicapai dari perancangan sistem dan implementasi program, yaitu dengan melakukan pengujian sehingga dapat ditarik kesimpulan.

BAB V PENUTUP

Pada bab terakhir berisi kesimpulan berdasarkan hasil yang telah dicapai dari pembahasan. Serta berisi saran yang diharapkan sebagai bahan pertimbangan oleh pihak-pihak yang akan melakukan pengembangan terhadap program ini.



BAB II

TINJAUAN PUSTAKA

2.1 Permainan/*Game*

Dalam bahasa Indonesia “*Game*” berarti “Permainan”. Permainan yang dimaksud dalam *game* juga merujuk pada pengertian sebagai “kelincahan intelektual” (*intellectual playbility*). Sementara kata “*game*” bisa diartikan sebagai arena keputusan dan aksi permainannya. Ada target-target yang ingin dicapai pemainnya. Kelincahan intelektual, pada tingkat tertentu, merupakan ukuran sejauh mana *game* itu menarik untuk dimainkan secara maksimal. (Arix, 2011:5)

Teori permainan pertama kali ditemukan oleh sekelompok ahli Matematika pada tahun 1944. Teori itu dikemukakan oleh John von Neumann and Oskar Morgenstern yang berisi :

“Permainan terdiri atas sekumpulan peraturan yang membangun situasi bersaing dari dua sampai beberapa orang atau kelompok dengan memilih strategi yang dibangun untuk memaksimalkan kemenangan sendiri atau pun untuk meminimalkan kemenangan lawan. Peraturan-peraturan menentukan kemungkinan tindakan untuk setiap pemain, sejumlah keterangan diterima setiap pemain sebagai kemajuan bermain, dan sejumlah kemenangan atau kekalahan dalam berbagai situasi.” (Neumann dkk, 1953).

Menurut Salen dan Zimmerman (2004:80) dalam *Rules of Play: Game Design Fundamentals* bahwa *game* adalah sebuah sistem di mana pemain terlibat dalam konflik buatan, didefinisikan oleh aturan, yang menghasilkan hasil kuantitatif. Dalam permainan muncul dari hubungan antara tindakan pemain dan sistem hasil, itu adalah proses dimana seorang pemain mengambil tindakan dalam sistem yang dirancang dari sebuah permainan dan sistem merespon tindakan.

Artinya dari suatu tindakan dalam permainan berada dalam hubungan antara tindakan dan hasil. *Game* edukasi merupakan bidang akademik baru dan bidang interdisipliner pembelajaran, yang berfokus pada *game*, bermain dan fenomena terkait.

2.2 *Adventure Game / Game* Petualangan

Menurut David Fox dan Roman Verhosek (2002:9), *game* dapat diklasifikasikan antara lain *Action Game*, *Combat Game*, *Adventure Game*, *Puzzle Game*, *Strategy Game*, dan *Card Game*. Pada penelitian ini, *game* dibangun dengan jenis *Adventure Game* atau *game* petualangan.

Adventure game sering dianggap sebagai bentuk interaksi fiksi. interaksi fiksi merupakan sebuah istilah yang mengacu pada media dimana pemain dapat mempengaruhi hasil cerita.

Karakter kunci dari *game* petualangan sebagai berikut:

1. Narasi sebagai daya tarik dimana pergerakan pemain sebagai hasil kemajuan permainan
2. Narasi seringkali diambil dari film, komik dan lain sebagainya.
3. Pemain umumnya mengontrol karakter utama
4. Permainan sering didasarkan pertanyaan atau teka teki, yang harus diselesaikan dengan berinteraksi dengan lingkungan permainan dan objek objek yang menghasilkan pengalaman bagi pemain.
5. Penekanan dalam *game* petulangan adalah pada eksplorasi, berpikir dan kemampuan pemecahan masalah secara cepat sebagai gaya aksi permainan

6. Unsur – unsur mendasar diantaranya:

- a. *Games Rules*: dijelaskan oleh penulis (pembuat *game*), yang mengatur operasi dan fungsi objek maupun karakter dalam permainan. Peraturan juga bisa didapatkan ketika pemain telah bermain.
- b. *Game World* : objek yang ada dalam permainan gua, pulau, penduduk dan lain sebagainya.
- c. *Plot* : Berisi informasi apa yang terjadi sebelum pemain bermain dan tujuan menyelesaikan permainan dan beberapa cara untuk menyelesaikan permainan.
- d. *Theme* : Tema moral yang biasanya mendasari permainan. seperti pemain diharuskan menjawab pertanyaan jika berhasil maka dapat memulihkan keseimbangan lingkungan.
- e. *Characters*: Player maupun NPC cenderung untuk memproses sifat tertentu atau atribut (kekuatan sihir, fitur- fitur khusus, wajah dsb)
- f. *Object / Item* : mempunyai peran penting dan biasanya dikumpulkan dan digunakan oleh pemain untuk memecahkan masalah. Seringkali pemain harus memiliki keahlian tertentu atau pengetahuan untuk menggunakannya.
- g. *Text, Graphics* dan *Sound* : *game* melibatkan kombinasi dari teks, grafis dan suara.
- h. *Animation*: animasi biasanya telah diprogram dan ditanamkan dalam permainan, yang berfungsi membantu dan memberi kesenangan pada pemain.

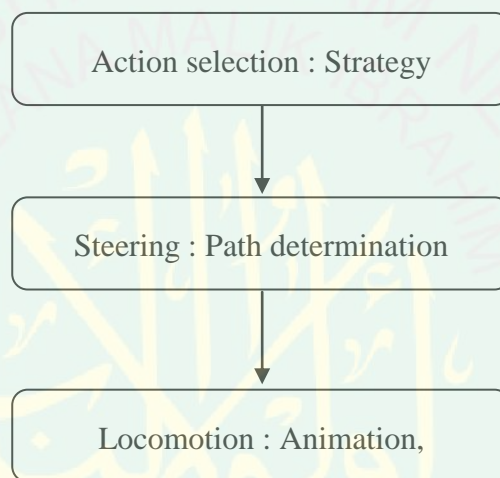
- i. *User Interface* : tampilan pada layar memungkinkan pengguna untuk berkomunikasi dengan mudah melalui pemilihan teks, grafis, suara dan animasi. (Dillon, 2005)

2.3 *Non Player Character* (NPC)

Autonomous character adalah jenis *otonom agent* yang ditujukan untuk penggunaan komputer animasi dan media interaktif seperti *games* dan *virtual reality*. Agen ini mewakili tokoh dalam cerita atau permainan dan memiliki kemampuan untuk improvisasi tindakan mereka. Dalam permainan, karakter otonom biasanya disebut *Non-Player Character* (NPC).

Penelitian sebelumnya yang juga meneliti tentang NPC antara lain penelitian yang dilakukan oleh Supeno Mardi dkk (2011), yang berjudul “Perilaku Taktis Untuk Non-Player Characters Di Game Peperangan Meniru Strategi Manusia Menggunakan Fuzzy Logic Dan Hierarchical Finite State Machine”. Penelitian tersebut membahas taktik menyerang yang dilakukan oleh NPC sehingga NPC bisa bersifat lebih natural seperti manusia. Selain itu ada penelitian lain yang dilakukan oleh Yunifa Miftachul Arif dkk (2011) yang meneliti pergantian senjata NPC pada game FPS menggunakan *fuzzy sugeno*. Penelitian lain yang dilakukan oleh Rahma Rizania Nasol (2013) berjudul “*Fuzzy State Machine* (FuSM) sebagai pengontrol player dalam game Assyuhada” yang merupakan pengembangan *game* Assyuhada yang dibangun oleh Miftakhur Rohman (2011) yang juga membahas tentang respon perilaku NPC musuh sehingga musuh bisa menentukan perilakunya sendiri dengan input variabel kesehatan sebagai faktor penentu keputusan.

Perilaku karakter yang otonom dapat lebih baik dipahami dengan membaginya menjadi beberapa lapisan. Lapisan ini dimaksudkan hanya untuk kejelasan dan kekhususan dalam diskusi yang akan mengikuti. Gambar 2.1 menunjukkan sebuah divisi gerak perilaku otonom hirarki karakter menjadi tiga lapisan: seleksi tindakan, steering, dan penggerak.



Gambar 2.1 Hirarki gerak perilaku

Tiga lapisan hirarki tersebut, adalah motivasi, tugas, dan motor (Reynold). Pada penelitian ini lebih di fokuskan pada bagian *action selection* yang di dalamnya berisi strategi pergerakan NPC.

2.4 Pembelajaran dalam Pandangan Islam

Belajar adalah salah satu kewajiban setiap umat Islam. Terlebih untuk mempelajari Al-Quran dan Al-Hadist sebagai pedoman hidup umat Islam. Sehingga perlu bagi umat Islam untuk mempelajari bahasa Arab yang merupakan bahasa Al-Quran dan Al-Hadist. Seperti yang dijelaskan dalam Al Hadist tentang menuntut ilmu

طَلَبُ الْعِلْمِ فَرِيضَةٌ عَلَى كُلِّ مُسْلِمٍ وَ مُسْلِمَةٍ - رواه ابن عبد البر

Artinya : “menuntut ilmu wajib atas setiap muslim dan muslimah” (HR. Ibnu Abdil Bari).

Belajar memiliki tiga arti penting menurut Al Quran. Pertama, bahwa orang yang belajar akan mendapatkan ilmu yang dapat digunakan untuk memecahkan segala masalah yang dihadapinya di kehidupan dunia. Kedua, manusia dapat mengetahui dan memahami apa yang dilakukan karena setiap apa yang yang diperbuat akan dimintai pertanggung jawabannya. Ketiga dengan ilmu yang dimilikinya mampu mengangkat derajatnya di mata Allah (Muhibbin,2004:68).

Dalam surat Az Zumar ayat 27 – 28 dijelaskan :

وَلَقَدْ ضَرَبْنَا لِلنَّاسِ فِي هَذَا الْقُرْآنِ مِنْ كُلِّ مَثَلٍ لَعَلَّهُمْ يَتَذَكَّرُونَ ﴿٢٧﴾ قُرْءَانًا
عَرَبِيًّا غَيْرَ ذِي عِوَجٍ لَعَلَّهُمْ يَتَّقُونَ ﴿٢٨﴾

Artinya :

Sesungguhnya telah Kami buat bagi manusia dalam Al Quran ini Setiap macam perumpamaan supaya mereka dapat pelajaran. (ialah) Al Quran dalam bahasa Arab yang tidak ada kebengkokan (di dalamnya) supaya mereka bertakwa.(QS. AzZumar : 27 – 28)

Penjelasan menurut tafsir Jalalain, bahwa Allah swt. telah menjadikan Al-Quran setiap macam perumpamaan agar manusia mendapat pelajaran dan mau menerima nasihat yang ada di dalamnya. Pada awal ayat 28 menjadi kata keterangan yang bersifat mengukuhkan atau disebut *hal mu-akkidah* bahwa Al-Quran diturunkan dalam bahasa Arab dan tidak ada kekeliruan atau pertentangan dengan maksud untuk menghindarkan manusia dari kekafiran. (Tafsir Jalalain jilid 2 : 2008)

Disini jelas bahwa Al-Quran pun menggunakan metode pembelajaran dalam memahami dan mengkaji isinya yang begitu luas. Berdasar dari hal itu dirasa perlu untuk membuat sebuah misal media pembelajaran bahasa Arab agar bisa digunakan untuk belajar memahami bahasa Arab dengan baik dan benar.

2.5 Bahasa Arab

Bahasa Arab merupakan bahasa yang istimewa di dunia ini seperti yang kita ketahui, bahwasannya bahasa Arab tidak hanya merupakan bahasa peradaban, melainkan juga sebagai bahasa persatuan umat Islam di dunia. Bahasa Arab adalah selain merupakan bahasa Al-Qur'an (firman Allah atau kitab pedoman umat Islam) yang memiliki uslub yang bermutu juga memiliki sastra yang sangat mengagumkan manusia dan manusia tidak mampu untuk menandingi. Bahwa

bahasa Arab merupakan bahasa orang Arab dan sekaligus juga merupakan bahasa Islam. (Bambang Kiswanto, 2009:1)

Di dalam kitab suci Al Quran, Allah SWT secara khusus meletakkan keutamaan bahasa Arab :

إِنَّا أَنْزَلْنَاهُ قُرْآنًا عَرَبِيًّا لَعَلَّكُمْ تَعْقِلُونَ

Artinya:

Sesungguhnya kami menurunkannya berupa Al Quran dengan berbahasa Arab, agar kamu memahaminya.(QS. Yusuf : 2)

﴿قُرْآنًا عَرَبِيًّا﴾ “*Sesungguhnya Kami menurunkan...*” yaitu Al-Qur’an, *إِنَّا أَنْزَلْنَاهُ*

“...Yang berbahasa Arab...” dengan menggunakan lisan Arab agar dapat dipahami dan dimengerti makna-maknanya, agar dapat diambil petunjukdarinya dan mendapatkan kesempurnaan serta kebahagiaan.

Firman-Nya ﴿لَعَلَّكُمْ تَعْقِلُونَ﴾ “...Agar kamu memahaminya” agar mantap dalam memahaminya dan dapat mengetahui apa yang terdapat padanya berupa petunjuk dan cahaya.

Sesuai penjelasan pada tafsir Al-Qur’an Al-Aisar, *Qur’anan Arabiyyan* adalah hal dari dhamir pada *anzalnaahu* sedangkan ‘*arabiyyan* adalah sifatnya yang tidak beraturan seperti sya’ir. Cerita-cerita yang diceritakan tidak lain adalah kitab yang teratur dengan rapi dibaca dan dijaga dan dapat diketahui isinya serta dapat diamalkan untuk mendapatkan kebahagiaan dunia akhirat. Al-Qur’an diturunkan dengan bahasa Arab yang merupakan bahasa sehari-hari, harapannya agar kita benar-benar mampu memahaminya dan dapat mengetahui apa yang diseru kepada kebenaran dan kepada jalan yang lurus. (Tafsir Al-Aisar,2007:766-767)

Firman Allah swt dalam surat Az-Zukhruf ayat 3 :

إِنَّا جَعَلْنَاهُ قُرْءَانًا عَرَبِيًّا لَعَلَّكُمْ تَعْقِلُونَ ﴿٣﴾

Artinya :

Sesungguhnya Kami menjadikan Al Quran dalam bahasa Arab supaya kamu memahaminya. (QS. Az-Zukhruf : 3)

Menurut Quraish Shihab dalam buku tafsir Al-Mishbah volume 12 menjelaskan ayat di atas bahwa penyifatan Al-Quran dengan kata ‘*arabiyyan*’ disamping sebagai penjelasan yang mengandung pujian terhadap kitab suci ini, juga sebagai kecaman terhadap masyarakat Arab yang menolaknya. Rujuk juga penjelasan ayat kedua surat Yusuf untuk memahami mengapa Al-Quran diturunkan dalam bahasa Arab.

Thabatha’i berpendapat bahwa pernyataan ayat di atas yang menjadikan tujuan dari dijadikannya Al-Quran dalam bahasa Arab *la’allakum ta’qilun / agar mereka memahaminya*, mengisyaratkan bahwa sebelum kitab suci ini “dijadikan bahasa Arab,” *kalam* Allah itu tidak terjangkau oleh akal manusia, karena akal manusia berpotensi untuk mengetahui segala sesuatu yang dapat dipikirkan – betapapun rumitnya. Dengan demikian, kitab suci ini – dari segi hakikat keberadaannya – merupakan sesuatu yang tidak terjangkau oleh nalar manusia, dan karena itulah maka Allah “menjadikannya” dalam bahasa Arab. (Al Misbah, 2003 : 538)

Bahasa Arab termasuk rumpun bahasa Semit yaitu bahasa yang dipakai bangsa-bangsa yang tinggal di sekitar Sungai Tigris dan Furat, dataran Syria dan

Jazirah Arabia (Timur Tengah) seperti bahasa Finisia, Assyria, Ibrania, Arabia, Suryania, dan Babilonia. Dari sekian banyak bahasa tersebut yang bertahan sampai sekarang adalah Ibrani. Sebenarnya bahasa Arab timbul sejak beberapa abad sebelum Islam, karena bukti peninggalan sastra Arab baru dapat dicatat hanya mulai sejak dua abad sebelum Islam, sehingga pencatatan bahasa Arab dimulai saat ini. (Umi, Rosyidi, 2008 : 7)

2.6 Test of Arabic as Foreign Language (TOAFL)

Test of Arabic as Foreign Language (TOAFL) merupakan tes untuk mengukur kemampuan seseorang dalam penguasaan bahasa Arab, baik secara lisan maupun tulisan. Bahasa Arab merupakan bidang studi yang saat ini mulai dikembangkan dalam wadah lembaga pendidikan formal ataupun nonformal. Namun penguasaan bahasa Arab terkesan kurang memadai, dengan melihat output yang dihasilkan masih sangat minim. Untuk menekan pengembangan bahasa Arab maka dikeluarkanlah standarisasi bahasa Arab yaitu *Test of Arabic as Foreign Language* (TOAFL).

Ketrampilan bahasa yang diujikan dalam *Test of Arabic as Foreign Language* (TOAFL) adalah ketrampilan mendengar, berbicara dan membaca. Materi *Test of Arabic as Foreign Language* (TOAFL) meliputi kebahasaan (Arab), ilmu- ilmu keIslaman, dan pengetahuan umum. (Bharmawi, 2011:9-14)

2.7 Logika Fuzzy

Logika *fuzzy* merupakan pengembangan dari logika klasik, dimana nilai kebenarannya berada pada interval [0,1]. Logika *Fuzzy* adalah algoritma yang

mengidentifikasi nilai samar atau daerah ketidakpastian. Logika ini diperkenalkan pertama kali pada tahun 1965 oleh Lotfi A. Zadeh yang merupakan seorang professor dari University of California di Berkley melalui makalahnya yang berjudul “*Fuzzy Sets*”. Pada saat itu logika *fuzzy* diperkenalkan bukan sebagai metodologi untuk mengatur, tetapi merupakan suatu cara untuk memproses data dimana himpunan keanggotaan parsial diperbolehkan, sebagai pengganti himpunan keanggotaan atau bukan keanggotaan yang *crisp*.

Definisi formal logika *fuzzy*, logika *fuzzy* merupakan sebuah logika yang dipresentasikan oleh ekspresi *fuzzy* (rumus) yang memenuhi kriteria sebagai berikut:

- i. Nilai kebenaran, 0 dan 1, dan variabel $x_i (\in [0,1], i = 1,2, \dots, n)$ merupakan ekspresi *fuzzy*.
- ii. Jika f merupakan ekspresi *fuzzy*, $\sim f$ juga merupakan ekspresi *fuzzy*.
- iii. Jika f dan g merupakan ekspresi *fuzzy*, $f \wedge g$ dan $f \vee g$ juga merupakan ekspresi *fuzzy* (Lee,2005:201).

Terdapat banyak model aturan *Fuzzy* yang bisa digunakan dalam proses *inference* akan tetapi ada dua model aturan yang paling sering digunakan yaitu :

1. Model Mamdani

Bentuk aturan yang digunakan pada model Mamdani adalah sebagai berikut : IF x_1 is A_1 AND ... AND x_n is A_n THEN y is B Dimana A_1, \dots, A_n, B adalah nilai nilai linguistik, sedangkan “ x_1 is A_1 “ menyatakan bahwa nilai dari variabel x_1 adalah anggota himpunan fuzzy A.

2. Model Sugeno

Model Sugeno merupakan varian dari model Mamdani dan memiliki bentuk aturan sebagai berikut : IF x_1 is A_1 AND ... AND x_n is A_n THEN $y = f(x_1, \dots, x_n)$

Dimana f bisa berupa sembarang fungsi dari variabel-variabel masukan yang nilainya berada dalam interval variabel keluaran. Dari penjelasan tentang logika *Fuzzy* dapat diketahui bahwa suatu sistem yang menggunakan logika *Fuzzy* mampu menangani suatu masalah ketidakpastian dimana masukan yang diperoleh merupakan suatu nilai yang kebenarannya bersifat sebagian (Dewi, 2003).

Atas dasar itulah pada penelitian ini, logika *fuzzy* digunakan dengan tujuan untuk mendapatkan respon perilaku NPC berdasarkan variabel input yang dimiliki. Selanjutnya dalam menentukan perubahannya digunakan model *Fuzzy* Sugeno, dimana setiap output senjata NPC diwakili oleh konstanta tetap yang sudah ditentukan sebelumnya.

2.8 *Finite State Machine* (FSM)

Menurut Ian Millington (2006) dalam bukunya yang berjudul *Artificial Intelligence for Games* menyebutkan bahwa *Finite State Machines* (FSM) masuk dalam ranah *Decision Making* (pembuat keputusan) pada *Artificial Intelligence* (AI).

Dalam FSM masing-masing karakter menempati satu *state*. Biasanya, tindakan atau perilaku yang terkait dengan masing-masing *state*. Jadi selama

karakter tetap dalam keadaan itu, ia akan terus melakukan tindakan yang sama. *State* terhubung bersama oleh *transition*. Setiap *transition* mengarah dari satu *state* ke *state* lain yang biasanya *state* tujuan *state target* ini disebut dengan *action* dan masing-masing memiliki seperangkat kondisi yang terkait. Jika permainan menentukan bahwa kondisi *transition* terpenuhi, maka karakter berubah dari *state* ke *state target (action)* melalui *transition* itu.

2.9 Fuzzy State Machine (FuSM)

Untuk mengurangi tingkat prediksi dalam perilaku *Artificial Intelligent* (AI) adalah dengan cara memungkinkan agen AI untuk menggabungkan beberapa perilaku pada waktu yang sama. Hal ini bisa dicapai melalui penggunaan *fuzzy logic* untuk mengimplementasikan *Fuzzy State Machine* (FuSM). FuSM menyatukan logika *fuzzy* dan FSM. FuSM memberikan derajat yang berbeda keanggotaan masing-masing *state*. Oleh karena itu, bukan hanya *state on / off* atau hitam / putih, FuSM dapat di *state 'on'* atau hampir *'off'* (Sweetser, Penelope & Wiles, Janet : 2002). FuSM mirip dengan FSM biasa tetapi menggunakan logika *fuzzy* untuk menangani transisi antara *state* dan / atau tindakan di dalam *state-state*.

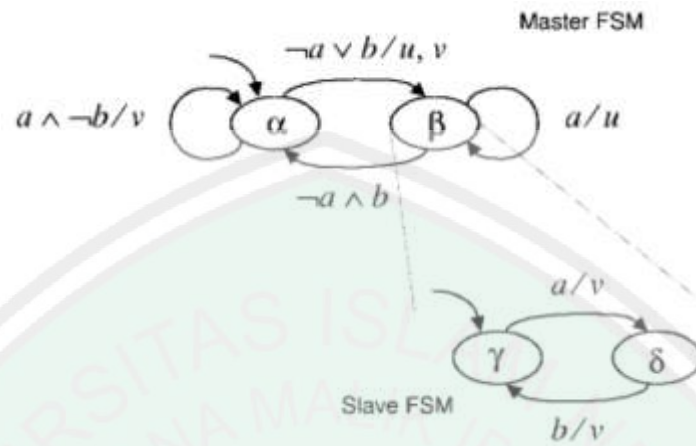
Generalisasi dari *finite state machine* menjadi *fuzzy state machine* adalah cukup mudah. Sebuah *state fuzzy* adalah sebuah *state* dengan derajat kemungkinan keanggotaan $0 \leq \mu \leq 1$, sebagai lawan dari *state (crisp)* dengan derajat implisit dari keanggotaan nol atau satu. Artinya, *fuzzy state machine* memungkinkan sistem untuk menjadi bagian dalam kondisi saat ini. (David L, 1990)

FuSM punya kelebihan dibanding dengan FSM biasa, yaitu *statenya* sangat modular, karena kita cukup fokus pada behaviour *state* tersebut, tidak perlu memikirkan *state* lain. Selain itu juga tidak akan tercipta *state oscillation* seperti pada FSM. Transisi diwakili oleh aturan *fuzzy*. Transisi menerima derajat keanggotaan. FuSM bisa berada di lebih dari satu *state* pada saat yang sama waktu. (Adriano Cruz : 2008)

Dan beberapa poin penting yang harus diperhatikan sebelum masuk lebih dalam ke FuSM ini adalah:

1. FuSM sedikit mengambil istilah-istilah dari *Fuzzy Logic*, misalnya: istilah *fuzzy* itu sendiri & istilah *Degree of Membership* (DOM) dari *fuzzy logic* diganti menjadi *Degree of Activation* (DOA).
2. Terdapat *Degree of Activation* (DOA) yang akan menentukan apakah suatu *state* dapat diaktifkan atau tidak.

FSM dasar yang tersusun lebih sederhana dan berurutan banyak memiliki kelemahan karena sistem yang paling praktis memiliki j *state* dan transisi yang banyak sehingga representasi dan analisis sulit. Salah satu solusinya adalah dengan menggunakan hirarki FSM. Pada Hirarki FSM, *state* dapat lebih disempurnakan ke dalam bentuk FSM lain. FSM lain ini merupakan pengembangan dari FSM dasar, dan bisa jadi di dalamnya diimplementasikan *Fuzzy State Machine*. Pada gambar di bawah, FSM lain yang dimaksud disebut FSM *slave* dan FSM di luar disebut FSM *master* dalam suatu komposisi seperti yang diilustrasikan pada Gambar 2.2.



Gambar 2.2 Hierarchy Finite State Machine

Contoh lain seperti pada penelitian yang dilakukan oleh Ady Wicaksono (2009) yang mengatur strategi menyerang pada NPC dengan metode *Fuzzy State Machine* (FuSM). FSM digambarkan pada Gambar 2.3.



Gambar 2.3 Finite State Machine untuk NPC

Gambar di atas menjelaskan perubahan *state* yang terjadi. Pada *state* menyerang, akan ada *state* lain yang akan diatur oleh logika *fuzzy* sehingga *state*

akan lebih variatif. Gambar 2.4 menjelaskan *state* baru yang diaktifkan dalam *state* menyerang.



Gambar 2.4 FuSM menyerang untuk NPC

Dengan menyertakan logika *fuzzy* dalam perancangan perilaku, maka aplikasi *multiple state* menjadi mungkin. Penerapan logika *fuzzy* dalam *finite state machine* disebut dengan *fuzzy state machine*.

2.10 Pemrograman Lite-C

Lite-C adalah bahasa pemrograman untuk aplikasi multimedia dan permainan komputer, menggunakan sintaks subset dari bahasa C dengan beberapa elemen dari bahasa C++. Perbedaan utamanya ke C adalah implementasi asli objek *game* komputer terkait seperti suara, gambar, film, elemen GUI, 2D dan 3D model. Lite-C berjalan pada 32-bit dan 64-bit Windows XP atau Vista. Lite-C mendukung API Windows dan Component Object Model (COM), sehingga OpenGL dan DirectX program langsung dapat ditulis dalam lite-C. Dan Lite-C Telah terintegrasi mesin 3D *Game Studio A7* dan *A8* secara gratis.

Contoh program Lite-C :

- Program lite-C berikut mencetak "Hello World", kemudian memutar file film dan keluar.

```
void
main()
{
printf("Hello, World!"); // message box
screen_size.x = 400;
screen_size.y = 400; // resize the window
int handle = media_play("greetings.mpg",NULL,50); // start
a movie in the whole window at volume 50.
while(media_playing(handle)) wait(1); // wait until movie
was finished
```

Gambar 2.5 Contoh skrip untuk menampilkan kata

- Program lite-C berikut membuka jendela 3D dan menampilkan bola berputar

```
void main()
{
level_load(""); // open an empty level. you can use NULL
instead of ""
ENTITY* sphere = ent_create("sphere.mdl",vector(0,0,0),NULL);
// create sphere model at position (0,0,0)
while(1) {
sphere->pan += 1; // rotate the sphere with 1 degree per frame
wait(1); // wait one frame
}
}
```

Gambar 2.6 Contoh skrip untuk menampilkan model 3D

2.11 3D Game Studio

Game Studio, sering dikenal sebagai 3D *Game Studio* atau 3DGS adalah software yang digunakan untuk pengembangan *game* yang memungkinkan pengguna untuk membuat *game* 3D dan aplikasi lain yang bersifat virtual-reality. Di dalamnya terdapat Model Editor (MED), World Editor (WED), Script Editor (SED) / debugger dan dilengkapi dengan koleksi tekstur, model dan karya

seni, serta sistem permainan template yang memungkinkan penciptaan dasar permainan penembak atau RPG. *Gamestudio* dipasarkan pada pengguna berbagai tingkat keahlian, menyediakan tiga tingkat penggunaan yang berbeda ("pemula", "maju", dan "profesional") untuk penggemar, seniman, serta programmer. Selain itu, Conitec juga pasar sistem untuk penggunaan seperti adver *game* pengembangan dan penciptaan presentasi real-time, simulasi, dan pameran virtual.

Perkembangan

- 1993 ACK 3D (Animation Construction Kit) by Lary Myers (Wolfenstein-like open source engine).
- 1994 ACK NEXT GENERATION by Johann Christian Lotter / oP Group (improved version of ACK 3D, open source).
- 1995 ACKNEX-2 written for the German TV show X-BASE (engine comparable to Doom).
- 1995 ACKNEX-2 becomes Conitec property and is released as '3D Game Studio'.
- 1997 ACKNEX-3 released.
- 1999 A4 released (Windows Based Quake-like Engine).
- 2000 A5 released (Terrain Engine).
- 2003 A6 released (Physics and Shaders).
- 2007 A7 released (new ABT renderer and Lite-C).
- 2010 A8 released (Enet Newtork Library, Nvidia PhysX, PSSM).

2.12 Editor

a. Editor Dunia (WED - World Editor)

Editor Dunia (WED), adalah editor utama. Dengan itu, pengguna dapat mengatur posisi berbagai objek, menetapkan tindakan untuk model (juga dikenal sebagai entitas) yang didefinisikan melalui script, memberikan tekstur untuk tingkat geometri, dan membangun tingkat menggunakan teknik pohon *Binary Space Partitioning* (BSP).

WED adalah program utama *Gamestudio*, pengguna dapat memulai permainan dari sini, dan memasang script. WED adalah lokasi dimana pengguna dapat menggabungkan semua bagian dari permainan mereka (pemrograman, 3d grafis).

Tata letak untuk WED cukup sederhana. Bagian utama, bagian kanan tengah, di mana sebagian besar editing dilakukan. Ada tiga grafik dan tampilan 3D. Melalui *re-arrangeable*, jendela kiri atas adalah tampak atas, yang memiliki X dan Y koordinat. Bagian bawah kiri adalah sisi tampilan, atau koordinat X dan Z. Bagian kanan atas adalah tampilan 3D, yang memberikan preview cepat. Grafik dibagi menjadi kelipatan 128 dan membagi lebih lanjut menjadi kelipatan 16 untuk membantu dengan gertakan dan spasi.

b. Editor Model (MED – Model Editor)

Meskipun banyak pengguna lebih memilih untuk menggunakan program pemodelan eksternal, model editor (MED), memberi kemampuan untuk merancang model, dan kadang-kadang digunakan untuk membuat

level. Model dapat dibuat dari apa pun mulai dari kotak sederhana, atau model manusia atau lingkungan yang kompleks seperti seluruh kota. Model terbuat dari *meshes*, sekelompok simpul dan segitiga (sering disebut "poligon") disatukan untuk membentuk sebuah bentuk, sebuah kerangka tulang untuk animasi; satu atau beberapa tekstur untuk kulit, dan efek file untuk shader (fx.).

Seperti WED, tata letak umum MED adalah hampir sama. Memiliki tiga grafik dan tampilan 3D. MED mirip seperti WED dalam tata letak umum, tetapi MED juga memiliki editor kulit, yang memungkinkan model yang akan bertekstur. Tekstur akan ditampilkan di sisi kiri dan model akan ditampilkan di sebelah kanan dengan tool bar sekitarnya. Tekstur sering dibuat dalam editor grafis eksternal dan diimpor dari sebuah file gambar PCX, BMP atau TGA. MED menggunakan pemetaan UV *vertex* dari *mesh* model ditempatkan pada tekstur-tekstur model. Model bentuk diciptakan baik melalui primitif (seperti kubus dan piramida) atau dengan menciptakan simpul dan wajah bangunan.

c. Editor untuk Bahasa (SED – Script Editor)

Script Editor (SED), adalah editor teks biasa dengan compiler dan debugger. Namun, kata kunci seperti "fungsi", "alpha", tipe variabel, atau nomor bisa berwarna berbeda untuk membantu, nomor baris membantu untuk menemukan kesalahan sintaks yang diberikan oleh editor, jumper kode memungkinkan melompat ke fungsi yang berbeda, dan fungsi lainnya

lebih membantu dalam pemrograman dan proyek pengorganisasian. SED digunakan untuk program Lite-C atau C-Script.



BAB III

RANCANGAN DESAIN SISTEM

3.1 Perancangan Sistem

Game yang dibangun adalah *game single player* yang berjenis *adventure game* atau *game* petualangan. Dalam permainan ini terdapat sebuah karakter (*avatar*) sebagai pemain utama, dan terdapat beberapa karakter *Non Player Character* (NPC) yang cerdas yang di mainkan oleh komputer. Objek penelitian dalam permainan ini adalah desain animasi pergerakan NPC kalajengking dan NPC pemburu kalajengking yang dikendalikan komputer.

3.1.1 Keterangan Umum *Game*

Lingkungan dalam *game* ini berbentuk dunia fantasi. Latar pada awal permainan adalah perbukitan dan terdapat NPC manusia yang diberi nama Syekh Al Karim-1 yang akan memberi petunjuk kepada pemain untuk menuju *stage* pertama.

Latar pada *stage* pertama pemain berada disebuah bangunan yang didalamnya terdapat 6 ruangan pada setiap ruangan terdapat 2 pertanyaan yang harus dijawab oleh pemain, pada *stage* ini pemain harus menemukan pintu ajaib untuk mendapatkan petunjuk permainan berikutnya pintu ajaib ini juga sebagai jalan menuju *stage* kedua, jika pemain lupa petunjuk pada *stage* ini maka pemain bisa bertanya pada NPC Syekh Al Karim-2 yang berada disalah satu ruangan pada bangunan tersebut.

Pada *stage* kedua latar yang digunakan adalah perbukitan, pada *stage* kedua ini pemain harus menjawab 8 pertanyaan, mengumpulkan rumput,

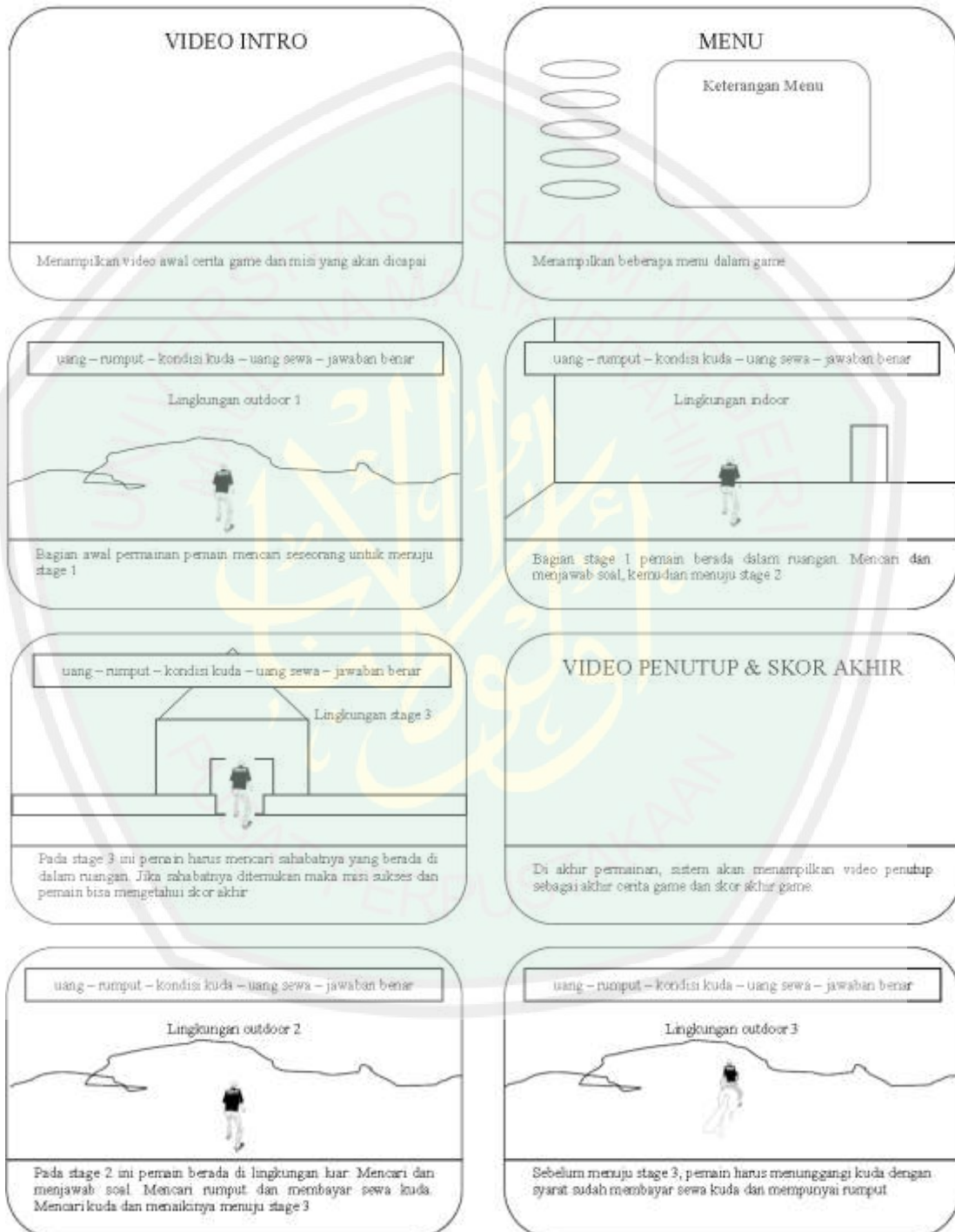
menyewa kuda kepada NPC Khan Al Jundi-2 kemudian menunggangi kuda menuju gerbang sebagai jalan menuju *stage* ketiga. Pada *stage* kedua ini juga terdapat musuh yaitu kalajengking yang dapat mengurangi kesehatan pemain, ketika pemain membutuhkan pertolongan NPC Syekh Al Karim-3 akan menolong pemain dengan membunuh kalajengking tersebut, jika pemain ingin menambah kesehatan bisa menghampiri dan membeli pada NPC Khan Al Jundi-1 dan jika pemain lupa petunjuk pada *stage* ini bisa bertanya pada beberapa NPC seperti tentara dan putri. Penelitian ini fokus pada aksi menyerang NPC kalajengking dan NPC pemburu kalajengking, kalajengking akan mengejar pemain pada jarak tertentu, berlari menjauh, atau menyerang pemain. NPC Syekh Al Karim-3 sebagai NPC yang akan membunuh kalajengking, NPC ini juga dikendalikan oleh komputer dengan aksi mengejar kalajengking dengan cara berjalan, mengejar berlari, dan menyerang kalajengking.

Stage ketiga terdapat 2 NPC penjaga yang menjaga makam dan NPC yang dicari yaitu Zaid Annas.

Unsur edukasi pada *game* ini terdapat pada pertanyaan-pertanyaan yang diberikan yang isinya berupa soal-soal TOAFL (*Test of Arab as A Foreign Language*). Objek penelitiannya adalah pada desain animasi pergerakan NPC (*Non Player Character*) yaitu kalajengking yang bisa mengurangi kesehatan pemain serta desain animasi pergerakan NPC pemburu kalajengking yang bisa membunuh kalajengking.

3.1.2 Storyboard Game

Berikut ini ada gambar dari storyboard dari game.



Gambar 3.1 Storyboard Game

3.1.3 Penampilan Umum *Game*

Game yang dibangun secara umum dibangun dengan 3Dimensi yang menarik, terdapat juga beberapa hal yang berbentuk 2Dimensi seperti munculnya pertanyaan berupa teks, dan panel untuk memuat jumlah uang yang dikumpulkan pemain, jumlah rumput yang dikumpulkan pemain, kondisi kuda, jumlah uang yang diberikan kepada NPC, jumlah pertanyaan yang telah dijawab dan panel-panel petunjuk. Objek-objek 3Dimensi dalam *game* yang dibangun diaplikasikan antara lain dalam bentuk pemain, pohon, rumput, ruangan, perbukitan, NPC dan lainnya.

3.1.4 Deskripsi Karakter

a) Karakter Utama (Pemain)

Karakter Ahmad Alfatih (pemain utama) pada *game* ini dimodelkan berperan sebagai pelajar Indonesia. Pemain merupakan *Player Character* yang mempunyai misi untuk menemukan keberadaan temannya yaitu Zaid Annas, untuk menemui temannya dia harus melewati beberapa tantangan, diantaranya dia harus menjawab pertanyaan – pertanyaan yang diberikan oleh komputer, mengumpulkan rumput dan membayar sewa kuda, serta harus berhati-hati dengan keberadaan kalajengking.

b) Karakter Syekh Alkarim

Karakter Syekh Alkarim adalah sosok laki- laki tua berambut dan berjenggot panjang yang sudah berwarna putih serta memakai jubah. Karakter ini bersifat *Non Player Character* (NPC). Animasi Syekh Alkarim terdapat pada saat

memberi bantuan petunjuk, bantuan perlindungan dari kalajengking dan ketika menjaga makam. Jadi, ada empat NPC Syekh AlKarim dalam *game* ini.

c) Karakter Khan Al Jundi

Karakter Khan Aljundi adalah seorang pria yang berbadan kekar dan membawa senjata laras panjang. Karakter Zaid Anas bersifat *Non Player Character* (NPC) dan animasinya terdapat pada saat meminta dan menerima uang sewa kuda serta ketika pemain membutuhkan kesehatan. Jadi, ada dua NPC Khan Aljundi.

d) Karakter Tentara

Karakter tentara dimodelkan seperti tentara- tentara biasanya menggunakan seragam doreng ala tentara, rompi tebal dan masker penutup sebagian muka. Karakter tentara bersifat *Non Player Character* (NPC). Animasi tentara yaitu ketika memberi bantuan petunjuk kepada pemain.

e) Karakter Putri

Karakter putri dimodelkan sebagai perempuan mengenakan gaun panjang. Karakter putri bersifat *Non Player Character* (NPC). Animasi putri yaitu ketika memberi bantuan petunjuk kepada pemain.

f) Karakter Zaid Annas

Karakter Zaid Annas dimodelkan sebagai pelajar dari Palestina dengan mengenakan jubah. Karakter Zaid Annas bersifat *Non Player Character* (NPC). Animasi Zaid Annas yaitu ketika bertemu dengan Ahmad Al Fatih (pemain utama).

g) Karakter Kalajengking (Aqrobun)

Karakter Kalajengking dimodelkan seperti kalajengking pada umumnya. Karakter ini bersifat *Non Player Character* (NPC). Animasi aksi kalajengking bisa mempengaruhi pemain (Ahmad Alfatih) yaitu mengurangi kesehatan pemain. *Fuzzy State Machine* diimplementasikan pada kalajengking ini, yaitu desain perilaku menyerang kalajengking sehingga kalajengking bisa berperilaku seperti kalajengking sesungguhnya.

h) Karakter Kuda (Jafir)

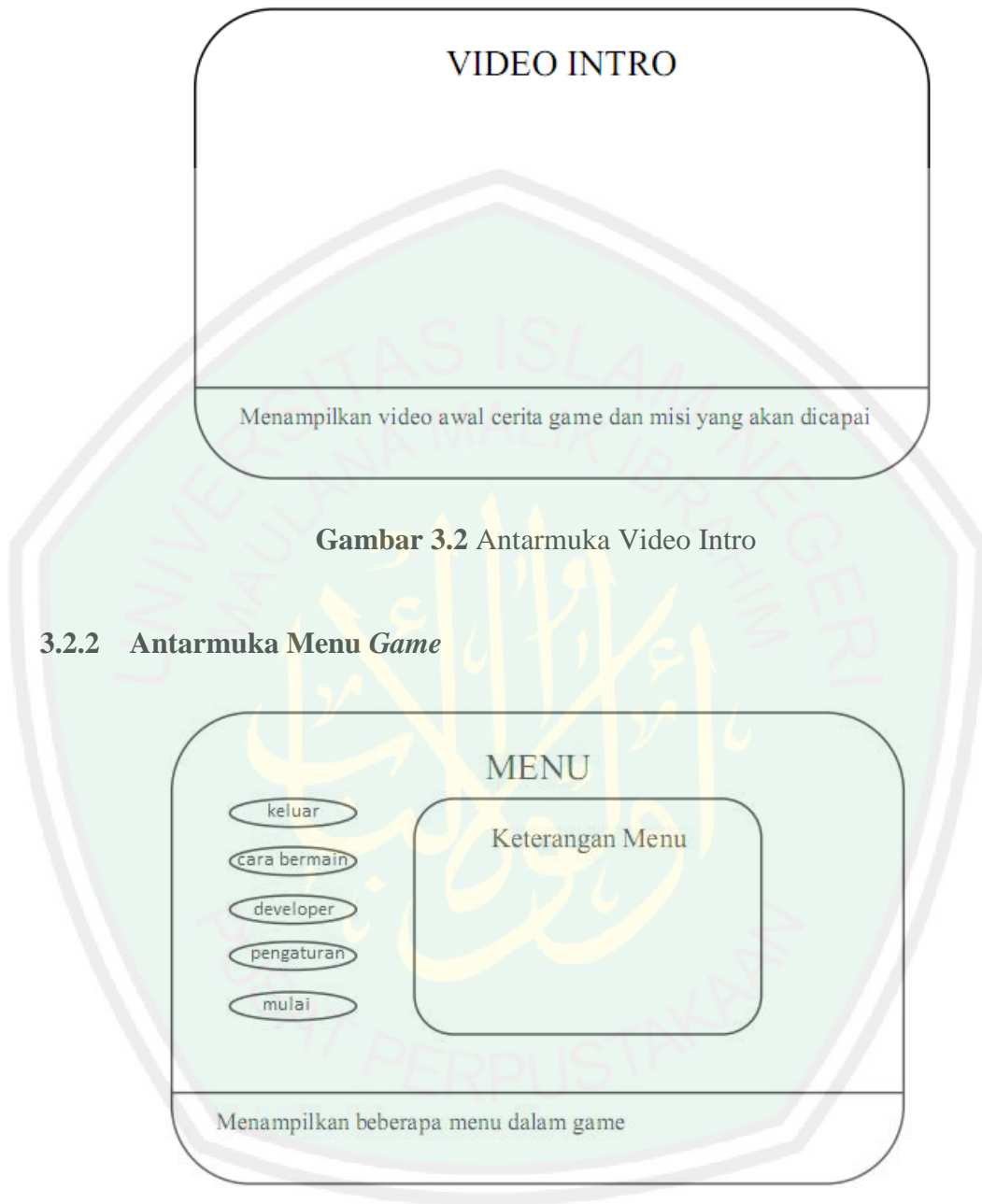
Karakter kuda dimodelkan seperti kuda pada umumnya. Karakter kuda bersifat *Non Player Character* (NPC) animasinya ketika terlihat ketika memasuki *stage* kedua, pergerakan kuda dipengaruhi oleh jumlah rumput yang dimiliki pemain, sewa kuda dan kondisi kuda.

3.2 Perancangan Aplikasi

Berikut ini penjelasan tentang perancangan aplikasi *game* berupa *Flowchart* dan keterangannya.

3.2.1 Perancangan Antarmuka Intro

Pada saat pertama *game* di jalankan akan muncul video intro yang menampilkan profil pemain utama dan NPC yang dicari dan menampilkan cerita dan misi permainan dengan mode layar *fullsrceen*, video ini dapat dihentikan (stop) dengan cara menekan tombol *Enter* pada *keyboard*. Di bawah ini digambarkan desain video *intro game*.



Gambar 3.3 Antarmuka Menu *Game*

Keterangan dari tombol menu yang ada antara lain:

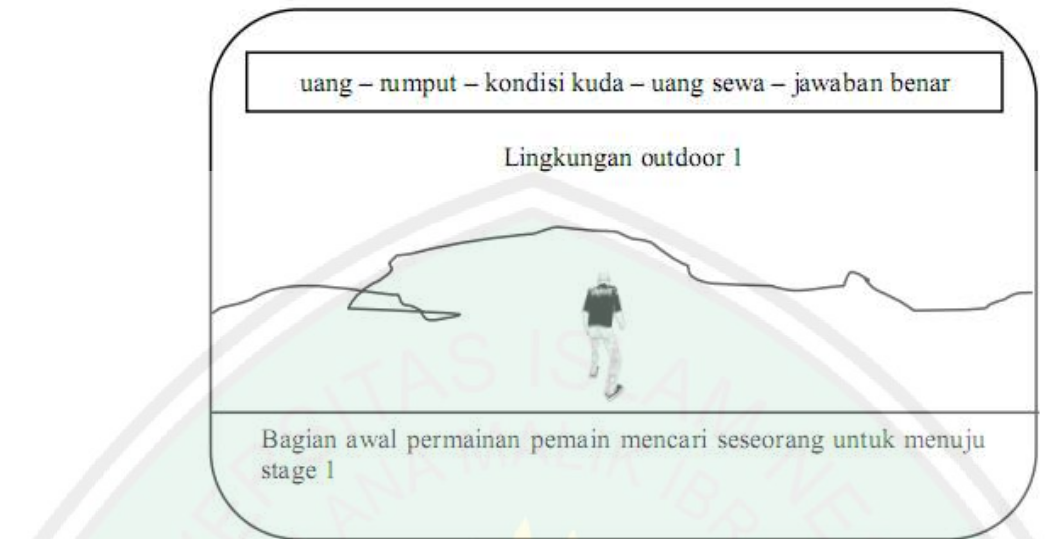
1. Tombol Keluar, berfungsi untuk keluar dari permainan.

2. Tombol Cara Bermain, berfungsi untuk memunculkan keterangan bantuan. Bantuan di sini berupa bagaimana cara menjalankan permainan.
3. Tombol Developer, berfungsi untuk memunculkan kredit yang berisi tentang ucapan terimakasih dan keterangan pembuat *game*.
4. Tombol Pengaturan, berfungsi untuk memunculkan pengaturan besar-kecilnya volume suara *background* musik dan suara efek.
5. Tombol Mulai, berfungsi untuk memulai permainan.

Keterangan akan muncul ketika tombol *game* tersebut diarahkan (*over*) pada tombol-tombol menu yang tersedia. Pada saat *cursor* diarahkan pada tombol, akan mengeluarkan bunyi suara efek. Khusus untuk menu pengaturan, terdapat *slider* yang digunakan untuk menambah atau mengurangi volume efek dan *background* musik pada *game* serta memilih tingkat kesulitan *game* berdasarkan tingkatan level yang disediakan.

3.2.3 Perancangan Antarmuka *Game*

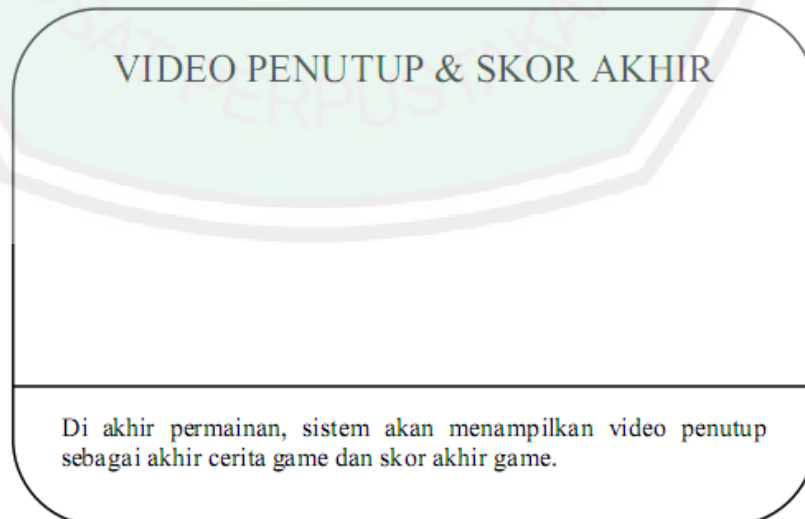
Perancangan antar muka *game* ini menggambarkan keadaan layar ketika pemain sudah pada posisi memulai permainan.



Gambar 3.4 Antarmuka *Game*

3.2.4 Perancangan Antarmuka Penutup

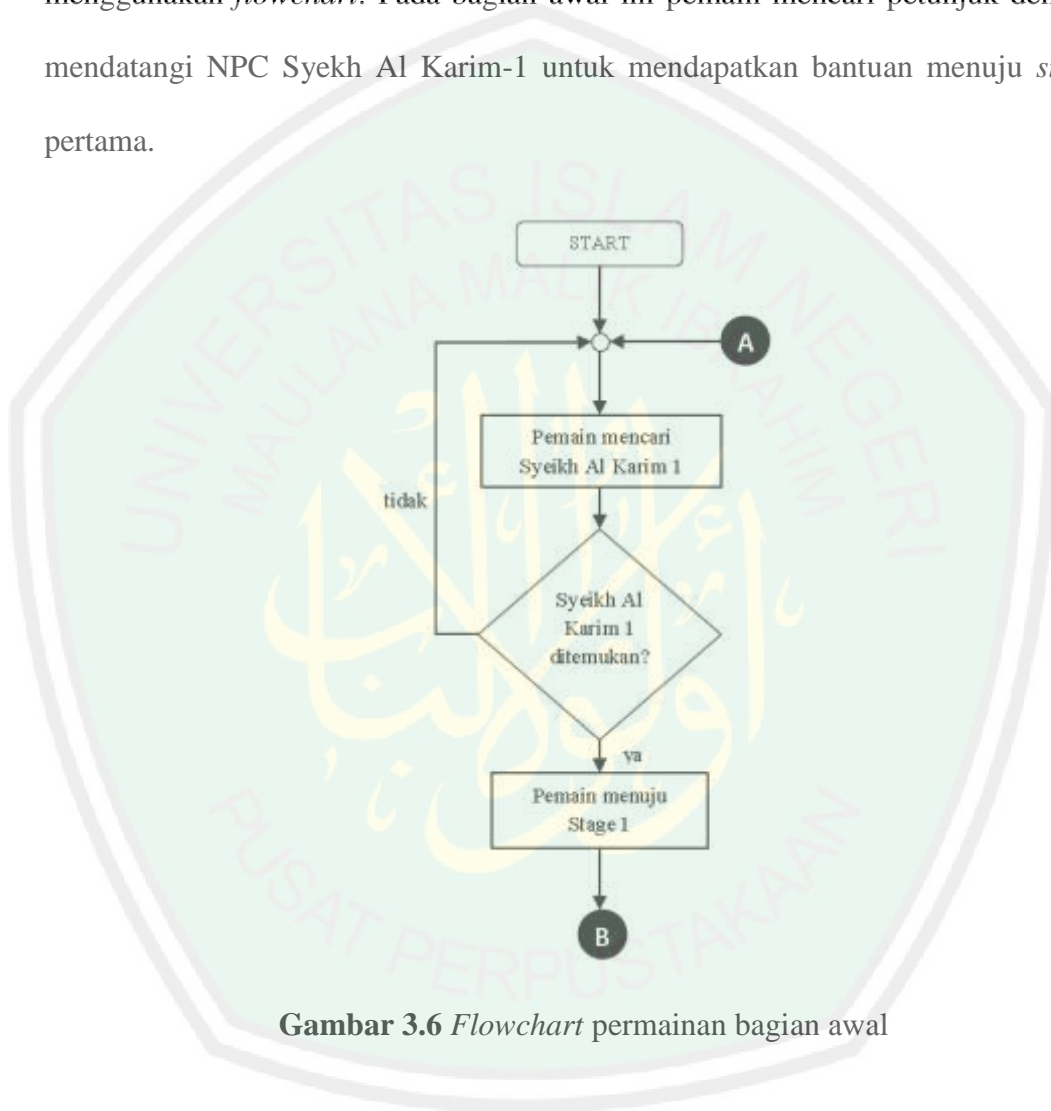
Pada saat permainan selesai akan muncul video penutup yang menampilkan perbincangan antara pemain utama dan NPC yang dicari dan menampilkan skor yang didapat oleh pemain dengan mode layar *fullsrceen*. Di bawah ini digambarkan desain video penutup *game*.



Gambar 3.5 Antarmuka Video Penutup

3.2.5 Perancangan Alur Permainan pada Bagian Awal

Perancangan permainan pada bagian awal permainan dijelaskan menggunakan *flowchart*. Pada bagian awal ini pemain mencari petunjuk dengan mendatangi NPC Syekh Al Karim-1 untuk mendapatkan bantuan menuju *stage* pertama.

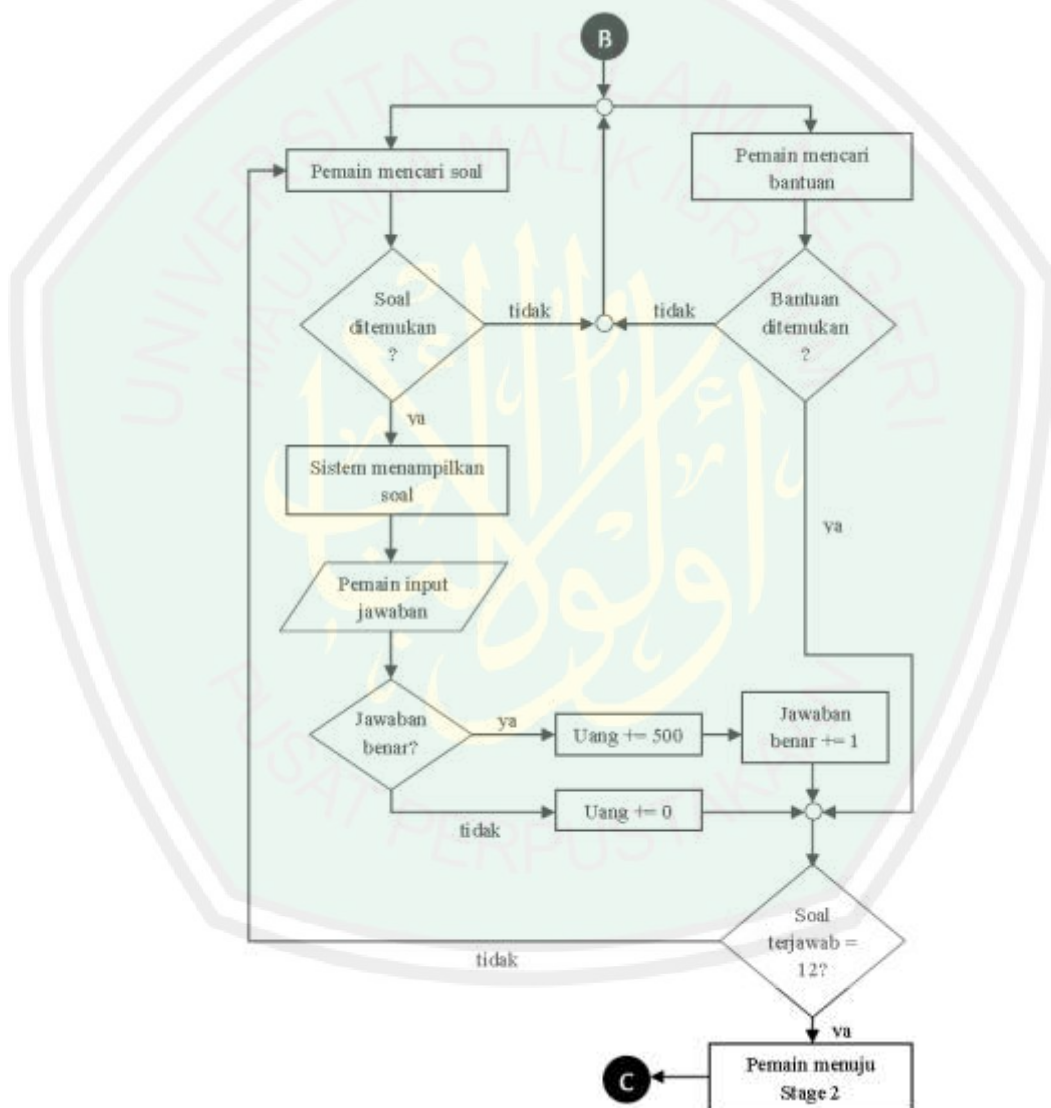


Gambar 3.6 *Flowchart* permainan bagian awal

Flowchart diatas menjelaskan, ketika pemain memulai permainan maka pemain harus mencari pemberi bantuan yang diwakili oleh NPC Syekh Al Karim-1. Pemain harus menemui Syekh Al Karim-1 agar bisa melanjutkan permainan menuju *stage* pertama.

3.2.6 Perancangan Alur Permainan pada Stage Pertama

Perancangan permainan pada *Stage* pertama ini dijelaskan menggunakan *flowchart*. Pada *Stage* pertama ini pemain harus mencari pertanyaan dan menjawab pertanyaan. Ada 12 pertanyaan di *stage* pertama ini.



Gambar 3.7 *Flowchart* Permainan *Stage* Pertama

Flowchart diatas menjelaskan *stage* pertama. Pemain akan berada di dalam ruangan, di dalam ruangan ini ada 12 soal yang harus dicari, soal ada di dalam

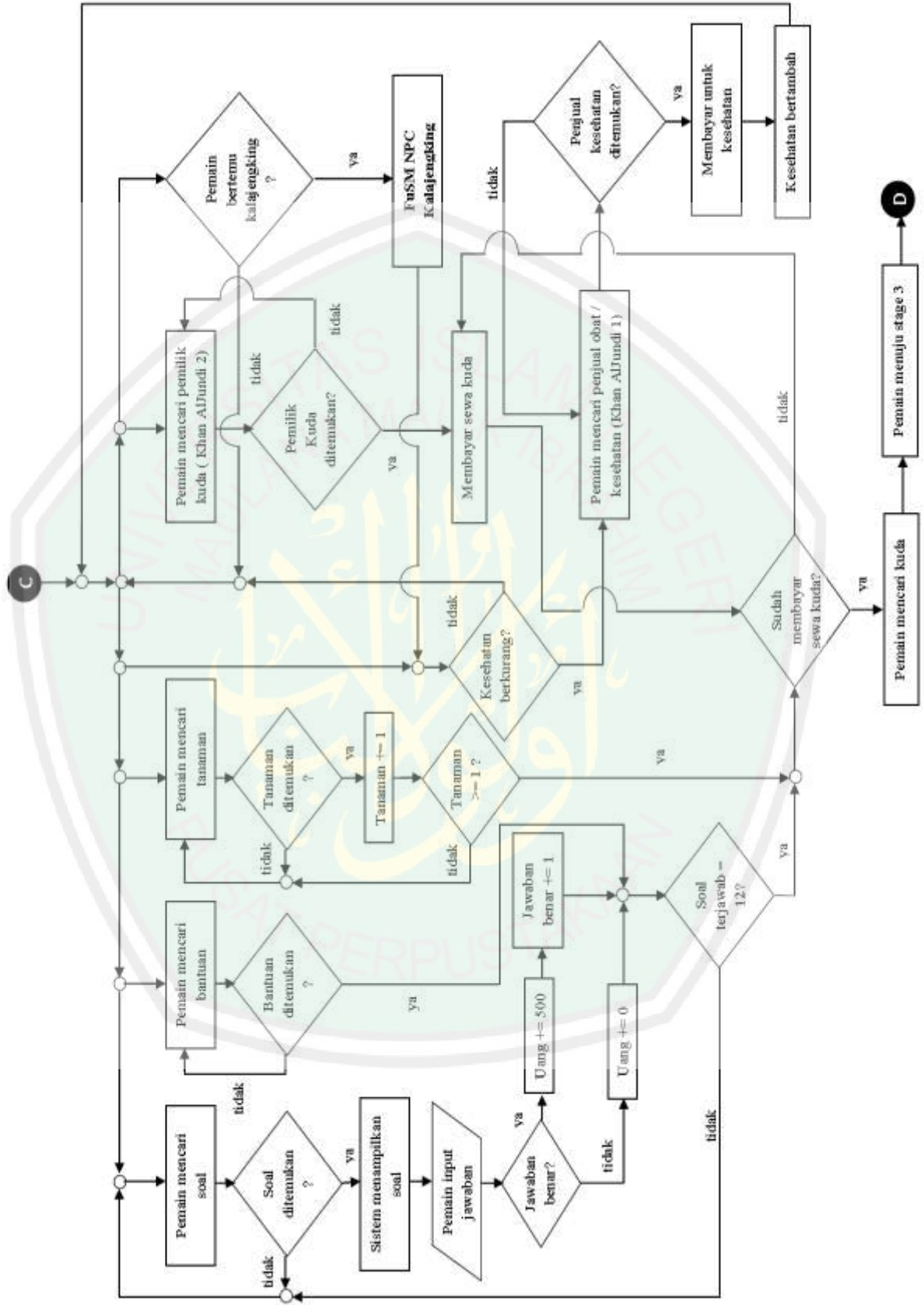
tanda tanya yang bertebaran di setiap lorong ruangan. Setiap soal yang dijawab dengan benar, maka akan menambah uang sebanyak 500, dan jumlah jawaban benar bertambah 1. Tapi jika soal dijawab salah, maka tidak akan menambah uang.

Jika pemain membutuhkan bantuan, maka bisa mencari seseorang yang bisa membantunya yaitu NPC Syekh Al Karim-2. Jika NPC ini ditemui maka sistem akan menampilkan panel bantuan tentang *stage* pertama. Di akhir *stage* pertama, pemain harus mencari pintu keluar untuk menuju *stage* kedua.

3.2.7 Perancangan Alur Permainan pada *Stage* Kedua

Pada *Stage* kedua ini pemain berada di luar ruangan. Pada *stage* kedua pemain juga harus mencari pertanyaan dan menjawabnya dengan benar. Ada 8 pertanyaan yang harus dijawab. Pada *stage* kedua ini ada beberapa syarat dan ketentuan yang harus dipenuhi agar bisa menuju ke *stage* ketiga.

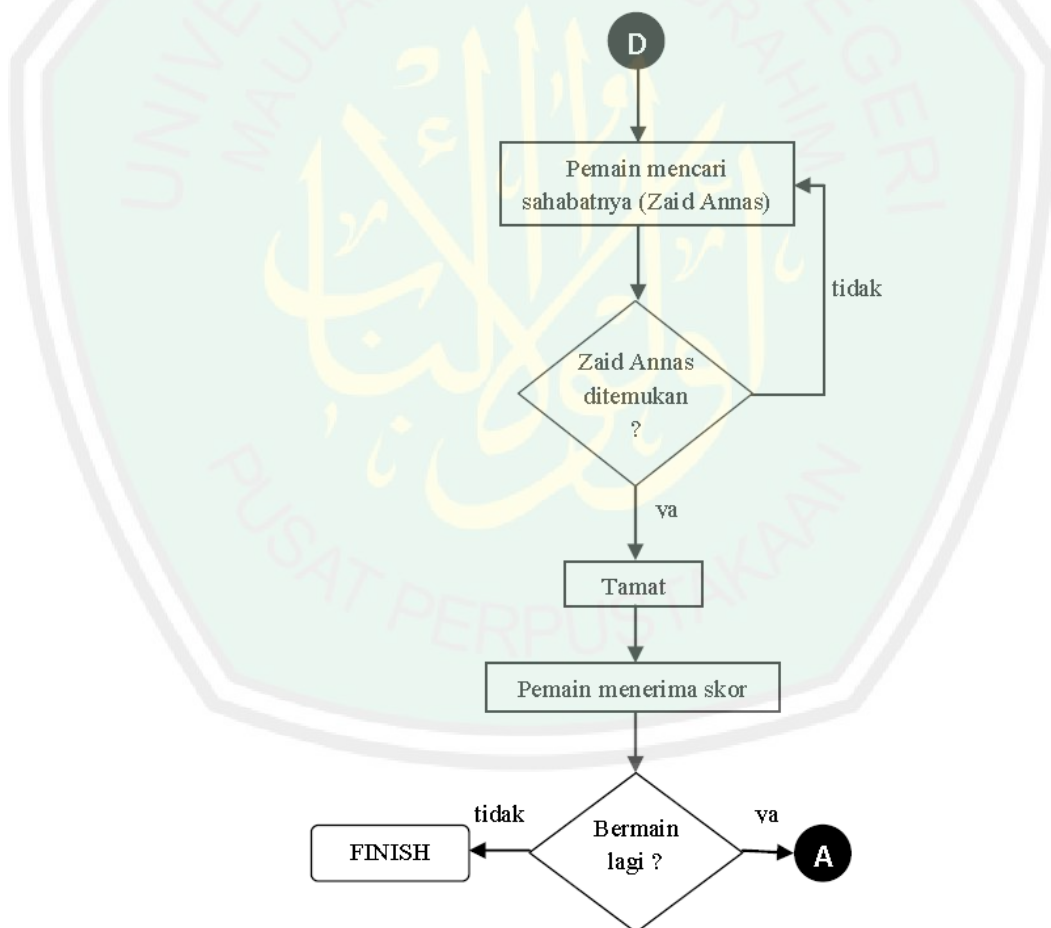
Pada petualangan di *stage* kedua ini, selain menjawab soal, pemain harus mengumpulkan tanaman atau rumput untuk makanan kuda yang akan mengantar pemain menuju *stage* ketiga, sebelum menunggangi kuda, pemain harus membayar sewa kuda pada pemilik kuda (Khan Al Jundi-2). Pemain perlu berhati-hati dengan kalajengking yang berada di lingkungan bebas, kalajengking ini bisa menyerang pemain dan bisa mengurangi kesehatan dengan cepat. Pengaturan perilaku kalajengking menyerang, mengejar dan lari diatur oleh *Fuzzy State Machine*.



Gambar 3.8 Flowchart Permainan Stage Kedua

3.2.8 Perancangan Alur Permainan pada Stage Ketiga

Pada *stage* ketiga ini, pemain berada di lingkungan lain dimana ada sebuah rumah yang di dalamnya ada sebuah makam yang dijaga oleh 2 orang penjaga. Pemain harus mencari seseorang yang berdiri di samping makam (Zaid Annas). Inilah misi permainan ini, pemain bertemu dengan sahabatnya yang kemudian akan ada akhir dari cerita *game* ini yang akhirnya pemain mengetahui skor permainan, nilai tes, dan jumlah jawaban benar.



Gambar 3.9 Flowchart Permainan Stage Ketiga

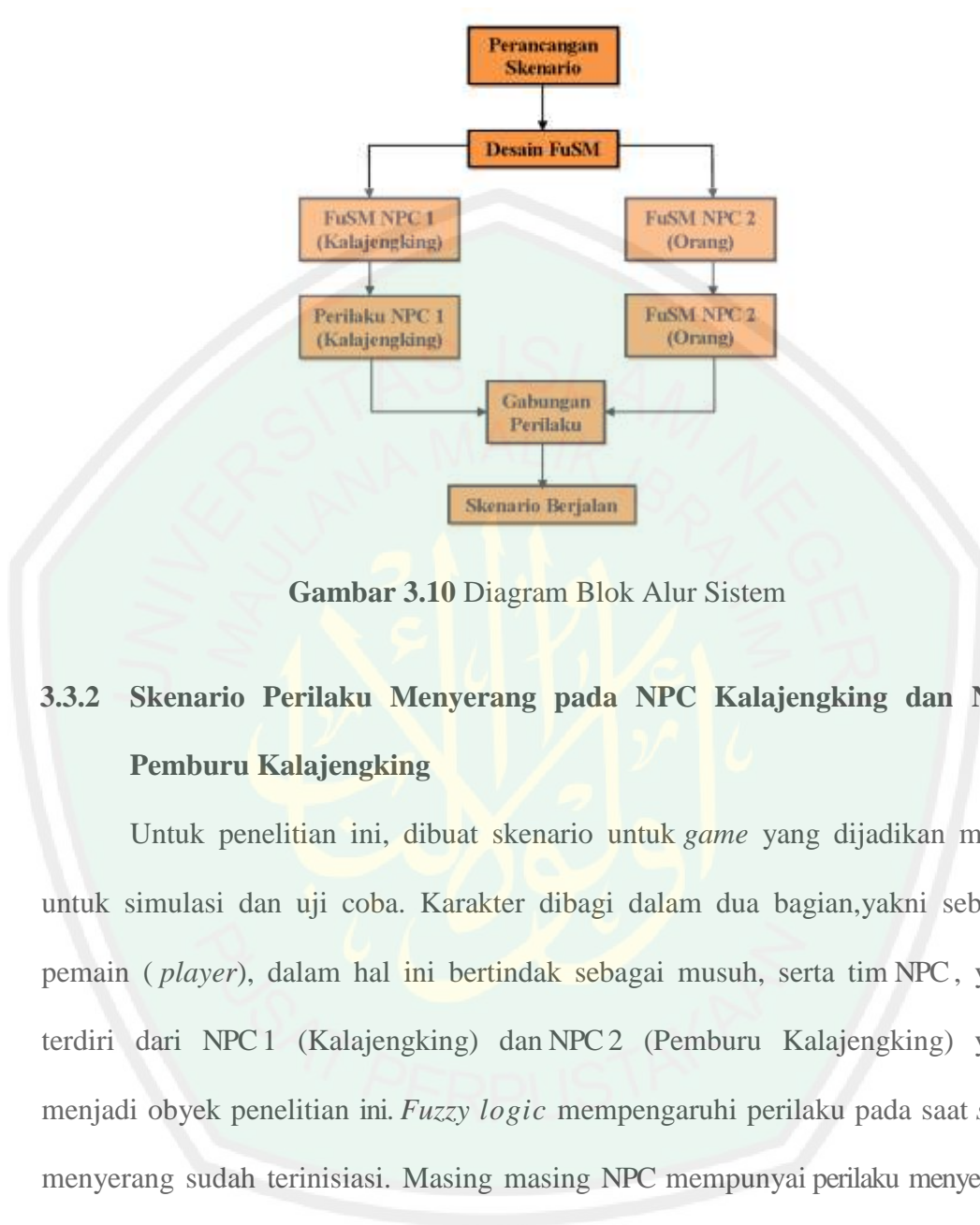
3.3 Perancangan *Fuzzy State Machine* Perilaku NPC

Fuzzy State Machine (FuSM) adalah salah satu kecerdasan yang diimplementasikan pada *game*, *Fuzzy State Machine* merupakan metode *Non-Deterministic* FSM pada sebuah *game*. Pengembangan dari *Finite State Machine* (FSM) dengan menanamkan logika *fuzzy* di dalamnya sehingga didapatkan variasi respon perilaku NPC terhadap player.

Dalam penelitian ini kecerdasan buatan diterapkan pada perilaku NPC kalajengking terhadap player sehingga kalajengking bisa bertindak lebih alami seperti di dunia nyata. Selain itu juga diterapkan pada NPC orang sebagai pemburu kalajengking, sehingga antara player, kalajengking dan NPC orang tersebut bisa saling mempengaruhi satu sama lain. Untuk dapat mengetahui hasil perancangan, maka diperlukan perangkat untuk mensimulasikannya yaitu melalui *game* ini.

3.3.1 Rancangan Alur Sistem

Untuk dapat mengaplikasikan rancangan yang dimaksud, penelitian ini disusun dalam tahapan yang ditunjukkan, seperti dalam gambar 3.10. Diagram tersebut menunjukkan tahapan yang dilakukan dalam melaksanakan penelitian ini, meliputi perancangan skenario *game*, *Fuzzy State Machine* untuk NPC 1 dan NPC 2.



Gambar 3.10 Diagram Blok Alur Sistem

3.3.2 Skenario Perilaku Menyerang pada NPC Kalajengking dan NPC Pemburu Kalajengking

Untuk penelitian ini, dibuat skenario untuk *game* yang dijadikan media untuk simulasi dan uji coba. Karakter dibagi dalam dua bagian, yakni sebagai pemain (*player*), dalam hal ini bertindak sebagai musuh, serta tim NPC, yang terdiri dari NPC 1 (Kalajengking) dan NPC 2 (Pemburu Kalajengking) yang menjadi obyek penelitian ini. *Fuzzy logic* mempengaruhi perilaku pada saat *state* menyerang sudah terinisiasi. Masing masing NPC mempunyai perilaku menyerang, sekaligus variabel yang mempengaruhi perubahan perilaku seperti ditunjukkan pada gambar 3.2 dibawah.

Skenario ini tentang respon perilaku NPC 1 yang akan mengejar pemain dan menyerangnya, kemudian jika pemain mendekati NPC 2 dengan pemicunya adalah jarak NPC 2 ke NPC 1, maka NPC 2 akan bergerak mengejar NPC 1 dan membunuhnya.

NPC	variabel input perilaku	output perilaku
NPC 1	Jarak terhadap player, kesehatan	Lari menjauh, mengejar player, menyerang
NPC 2	Jarak terhadap NPC 1	Berjalan, berlari, menyerang

Gambar 3.11 Perilaku NPC

3.3.3 FSM Perilaku NPC Kalajengking dan NPC Pemburu Kalajengking

Gambar 3.12 menampilkan desain FSM untuk NPC 1 atau kalajengking.

State utama yang tersusun dalam FSM dapat dijelaskan sebagai berikut :

1. *Spawn/ start*

Merupakan *state* posisi awal NPC

2. *Walk/ patrol State* berjalan/ patroli,

NPC bergerak menuju area *player* (musuh)

3. Menyerang

State NPC terlibat pertempuran, dipicu jangkauan NPC terhadap *player*.

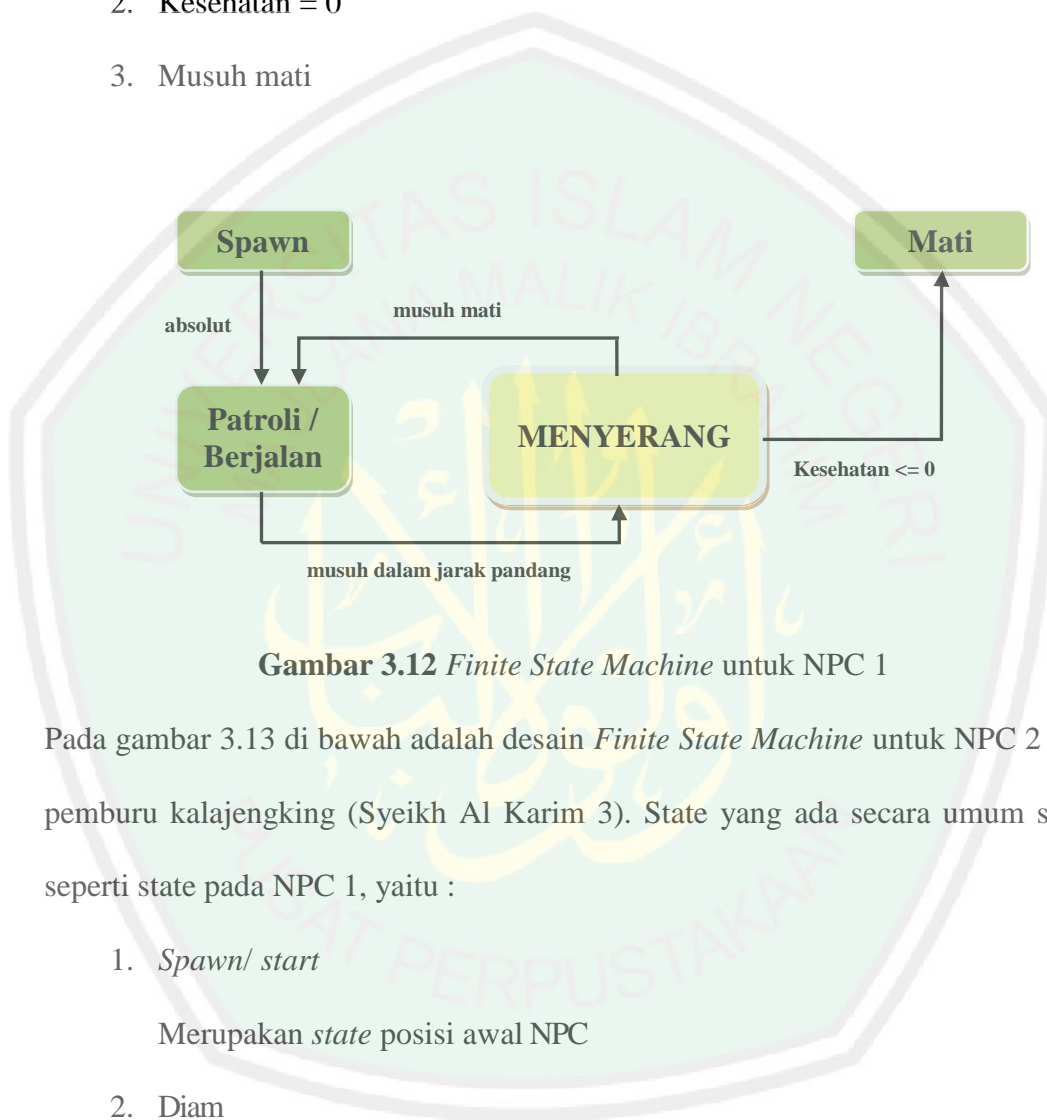
Terdapat *state* menyerang, mengejar dan melarikan diri dalam *state* menyerang ini.

4. Mati

Nilai kesehatan NPC = 0

Daftar *state transition* pada *FSM* dapat disusun dalam *list* sebagai berikut:

1. Musuh dalam jarak pandang
2. Kesehatan = 0
3. Musuh mati



Gambar 3.12 *Finite State Machine* untuk NPC 1

Pada gambar 3.13 di bawah adalah desain *Finite State Machine* untuk NPC 2 atau pemburu kalajengking (Syeikh Al Karim 3). State yang ada secara umum sama seperti state pada NPC 1, yaitu :

1. *Spawn/ start*

Merupakan *state* posisi awal NPC

2. Diam

NPC diam menunggu pemicu untuk bergerak

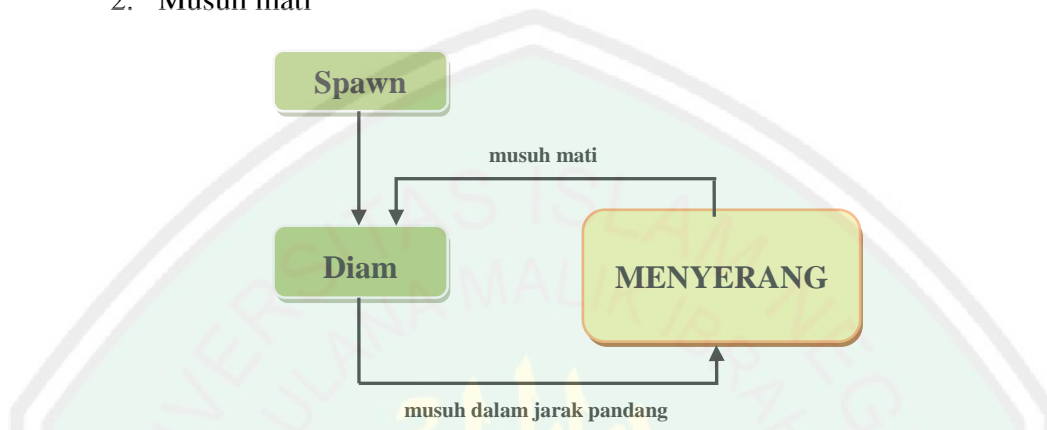
3. Menyerang

State NPC 2 terlibat pertempuran, dipicu jangkauan NPC 2 terhadap NPC

1 atau kalajengking. Terdapat *state* menyerang, berlari dan berjalan dalam *state* menyerang ini.

Daftar *state transition* pada *FSM* dapat disusun dalam *list* sebagai berikut:

1. Musuh dalam jarak pandang
2. Musuh mati



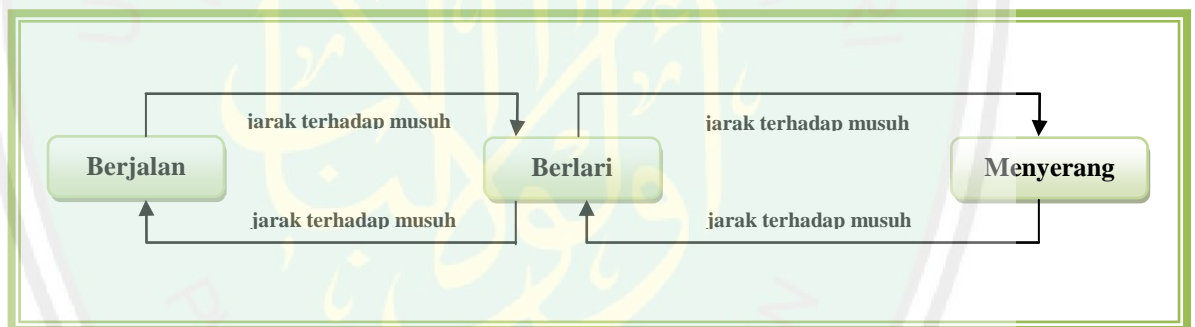
Gambar 3.13 *Finite State Machine* untuk NPC 2

3.3.4 FuSM Perilaku NPC Kalajengking dan NPC Pemburu Kalajengking

Dengan menyertakan logika *fuzzy* dalam perancangan perilaku, maka aplikasi *multiple state* menjadi mungkin. Dalam penelitian ini, logika *fuzzy* diaplikasikan pada perilaku dua NPC pada saat strategi menyerang dilakukan. Perilaku ini muncul saat *state* menyerang aktif. Tahapan *fuzzy* untuk menentukan output sebagai keputusan perilaku NPC yaitu fuzzifikasi, implikasi dan komposisi aturan, serta defuzzifikasi atau penegasan. Rancangan *state machine* untuk *state* menyerang dijabarkan dalam gambar. *Fuzzy State Machine* untuk kedua NPC digambarkan pada Gambar 3.14 dan 3.15



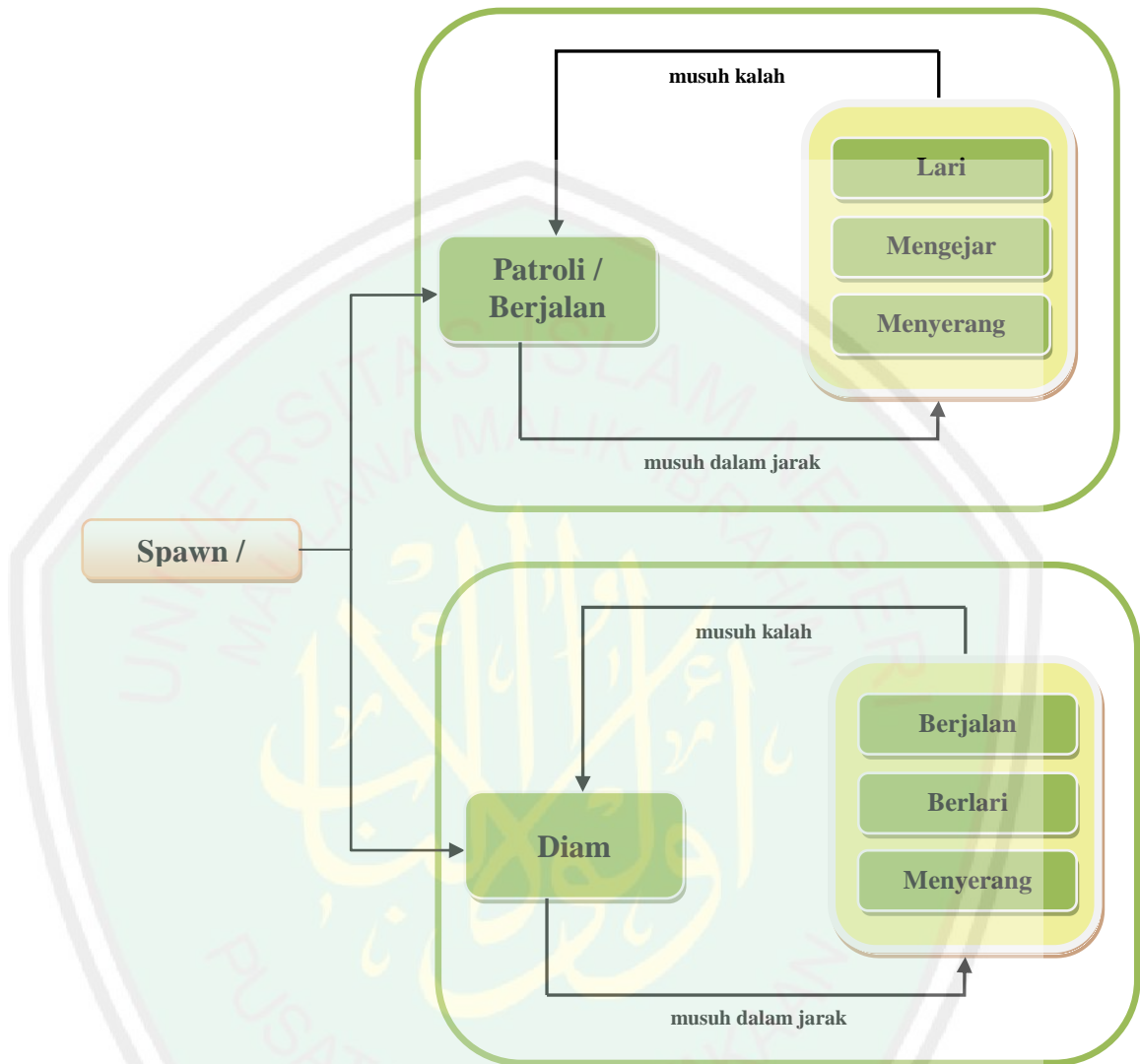
Gambar 3.14 *FuSM* Menyerang untuk NPC 1



Gambar 3.15 *FuSM* Menyerang untuk NPC 2

3.3.5 Top Level Finite State

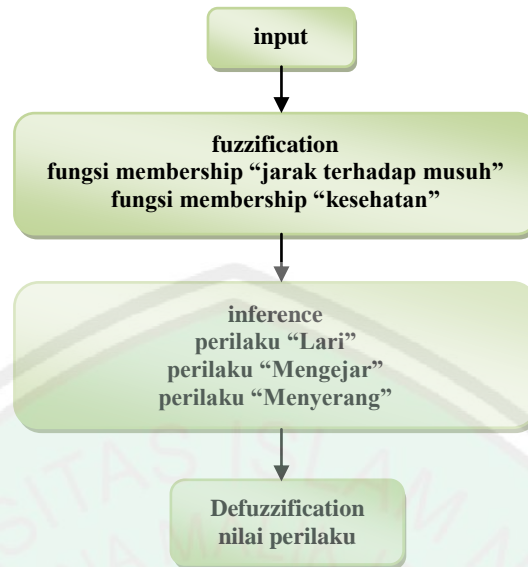
Untuk penelitian strategi menyerang ini, *top level Finite State Machine* merupakan gabungan dari *state machine* untuk dua NPC. *Top level Finite State Machine* ditunjukkan dalam Gambar 3.16



Gambar 3.16 Top Level Finite State Machine

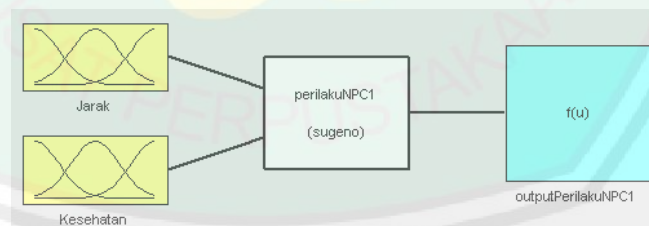
3.3.6 Desain Fuzzy NPC 1 (Kalajengking)

Dua variabel digunakan untuk merancang perilaku NPC 1, yaitu variabel “jarak terhadap musuh” dan variabel “kesehatan”. Logika *fuzzy* untuk menghasilkan perilaku NPC 1 ditunjukkan dalam Gambar 3.17



Gambar 3.17 Logika *Fuzzy* untuk menghasilkan perilaku NPC 1

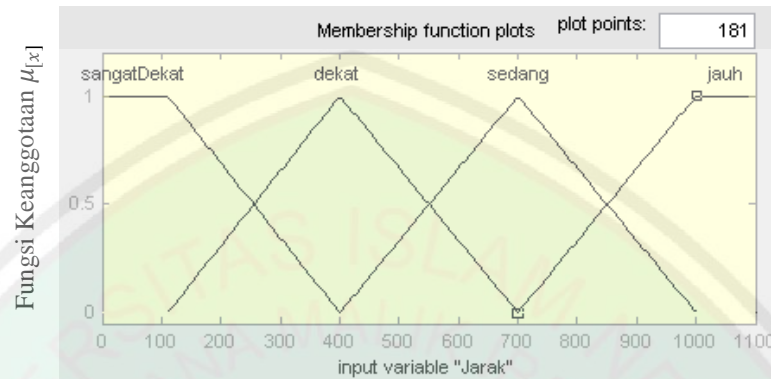
Desain *fuzzy* untuk menghasilkan perilaku NPC 1 dapat dilihat pada Gambar 3.18. Atribut yang diberikan untuk NPC 1 adalah jarak terhadap musuh dan kesehatan. Dengan masing-masing atribut menggunakan gabungan fungsi keanggotaan segitiga dan trapesium, Gambar 3.18 menunjukkan desain perilaku untuk menghasilkan perilaku NPC 1



Gambar 3.18 Desain *Fuzzy* untuk menghasilkan perilaku NPC 1

Seperti dijelaskan pada jbaran terdahulu, variabel linguistik mempunyai nilai-nilai linguistik dalam interval numerik yang semantiknya didefinisi oleh fungsi keanggotaannya. Gambar 3.19 menunjukkan derajat keanggotaan

(membership degree) untuk *input* jarak terhadap musuh yang mempunyai nilai dalam interval antara 0 sampai 1100.



Gambar 3.19 Derajat keanggotaan untuk *input* jarak terhadap musuh

Untuk fungsi keanggotaan “jarak terhadap musuh” mempunyai beberapa variabel linguistik serta notasinya dengan interval nilai beragam, dalam interval 0 sampai 1100, yang ditentukan seperti dalam Tabel 3.1

Tabel 3.1 Variabel linguistik *input* jarak terhadap musuh NPC 1

Input jarak		
Variabel	Notasi	Nilai
SD	Sangat Dekat	0 – 400
D	Dekat	100 – 700
S	Sedang	400 – 1000
J	Jauh	700 - 1100

Dengan fungsi keanggotaan jarak sebagai berikut :

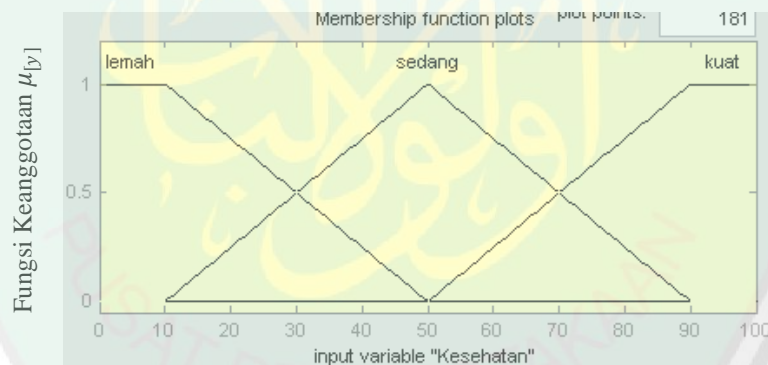
$$\mu_{\text{sangatdekat}} [x] = \begin{cases} 1; & x \leq 100 \\ \frac{400-x}{300}; & 100 \leq x \leq 400 \dots\dots\dots (1) \\ 0; & x \geq 400 \end{cases}$$

$$\mu_{jdekati} [x] = \begin{cases} \frac{x-100}{300}; & 100 \leq x \leq 400 \\ \frac{700-x}{300}; & 400 \leq x \leq 700 \\ 0; & x \geq 700 \text{ atau } x \leq 100 \end{cases} \dots\dots\dots(2)$$

$$\mu_{jkadang} [x] = \begin{cases} \frac{x-400}{300}; & 400 \leq x \leq 700 \\ \frac{1000-x}{300}; & 700 \leq x \leq 1000 \\ 0; & x \geq 1000 \text{ atau } x \leq 400 \end{cases} \dots\dots\dots (3)$$

$$\mu_{jjauh} [x] = \begin{cases} 0; & x \leq 700 \\ \frac{x-700}{300}; & 700 \leq x \leq 1000 \\ 1; & x \geq 1000 \end{cases} \dots\dots\dots(4)$$

Untuk *input* “kesehatan”, derajat keanggotaannya mempunyai nilai dalam interval 0 sampai 100, ditunjukkan dalam Gambar 3.20 dan Tabel 3.2



Gambar 3.20 Derajat keanggotaan untuk *input* kesehatan

Untuk *input* “kesehatan” mempunyai beberapa variabel linguistik serta notasinya dengan interval nilai beragam, yang ditentukan seperti disusun dalam Tabel 3.2

Tabel 3.2 Variabel linguistik *input* kesehatan

Input kesehatan		
Variabel	Notasi	Nilai
L	Lemah	0 – 50
S	Sedang	10 – 90
K	Kuat	50 – 100

Dengan fungsi keanggotaan kekuatan sebagai berikut :

$$\mu_{klema} h[x] = \begin{cases} 1; x \leq 10 \\ \frac{50-x}{40}; 10 \leq x \leq 50 \dots\dots\dots(5) \\ 0; x \geq 50 \end{cases}$$

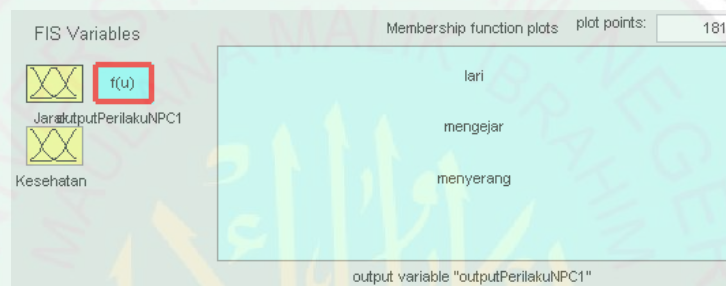
$$\mu_{ksedang} [x] = \begin{cases} \frac{x-10}{40}; 10 \leq x \leq 50 \\ \frac{90-x}{40}; 50 \leq x \leq 90 \dots\dots\dots(6) \\ 0; x \geq 90 \text{ atau } x \leq 10 \end{cases}$$

$$\mu_{kkuat} [x] = \begin{cases} 0; x \leq 50 \\ \frac{x-50}{40}; 50 \leq x \leq 90 \dots\dots\dots(7) \\ 1; x \geq 90 \end{cases}$$

Untuk *output* perilaku NPC 1 dijelaskan melalui Tabel 3.3, nilai linguistiknya dibagi menjadi 3, dengan notasi Lari (L), mengejar (MJ) dan menyerang (MY). Gambar 3.21 menunjukkan keanggotaan untuk *output* pada NPC 1

Tabel 3.3 Variabel linguistik *output* NPC 1

Output NPC 1	
Variabel	Notasi
L	Lari
MJ	Mengejar
MY	Menyerang

**Gambar 3.21** Keanggotaan *output* perilaku NPC 1

Selanjutnya aturan *fuzzy* (*fuzzy rule*) yang disusun untuk untuk menghasilkan perilaku NPC 1 dijelaskan pada Tabel 3.4. *Fuzzy rule* yang tersusun dalam matrikstabel tersebut merupakan formulasi dari dua input yaitu jarak terhadap musuh dan kesehatan. Tabel 3.4 Aturan *fuzzy* untuk menghasilkan perilaku NPC 1

Tabel 3.4 Aturan *Fuzzy* perilaku NPC 1

		Kesehatan		
Jarak Terhadap Musuh	Variabel	L	S	K
	SD	MJ	MY	MY
	D	MJ	MJ	MJ
	S	MJ	MJ	MJ
	J	L	L	MJ

Dari table *fuzzy rule* pada tabel 3.4, diperoleh *fuzzy rule IF/THEN* yang menjelaskan hubungan antara *input* dan *output* variabel linguistik, yang tersusun sebagai berikut :

1. **IF** Jarak Musuh = sangat dekat **AND** Kesehatan = lemah **THEN** Perilaku =MJ.
2. **IF** Jarak Musuh = sangat dekat **AND** Kesehatan = sedang **THEN** Perilaku =MY
3. **IF** Jarak Musuh = sangat dekat **AND** Kesehatan = kuat **THEN** Perilaku = MY
4. **IF** Jarak Musuh =dekat **AND** Kesehatan = lemah **THEN** Perilaku =MJ
5. **IF** Jarak Musuh =dekat **AND** Kesehatan = sedang **THEN** Perilaku =MJ
6. **IF** Jarak Musuh =dekat **AND** Kesehatan = kuat **THEN** Perilaku =MJ
7. **IF** Jarak Musuh =sedang **AND** Kesehatan = lemah **THEN** Perilaku =MJ
8. **IF** Jarak Musuh =sedang **AND** Kesehatan = sedang **THEN** Perilaku =MJ
9. **IF** Jarak Musuh =sedang **AND** Kesehatan = kuat **THEN** Perilaku =MJ
10. **IF** Jarak Musuh =jauh **AND** Kesehatan = lemah **THEN** Perilaku =L
11. **IF** Jarak Musuh =jauh **AND** Kesehatan = sedang **THEN** Perilaku =L
12. **IF** Jarak Musuh =jauh **AND** Kesehatan = kuat **THEN** Perilaku =MJ

Sehingga dalam menentukan output atau keputusan dapat dilakukan dengan menggunakan rumus sebagai berikut :

1. Fuzzifikasi

Menghitung nilai $\mu_{[x]}$ sebagai nilai derajat keanggotaan fuzzy untuk setiap variabel input, menggunakan rumus fungsi keanggotaan 1 – 7.

2. Implikasi

$$\alpha_n = \mu_{predikatRn} = \min(\mu_{[x]}; \mu_{[y]}) \dots \dots \dots (8)$$

$(\mu_{[x]}; \mu_{[y]})$ = variabel input yang saling berhubungan, lihat tabel 3.4

3. Defuzzifikasi

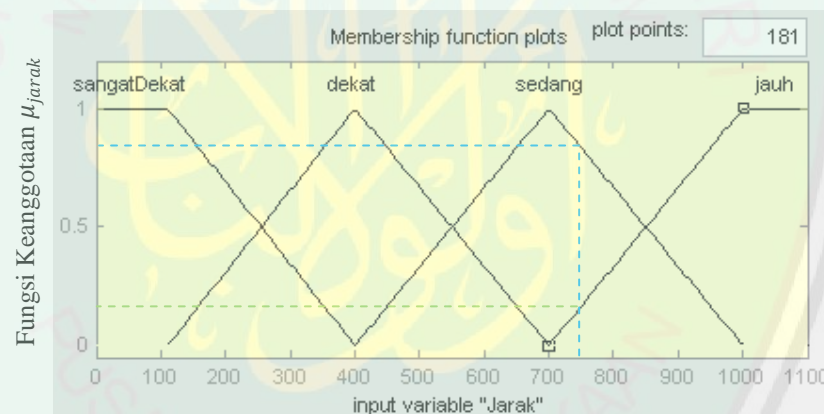
$$k = \max(\alpha_1, \alpha_2, \dots, \alpha_n) \dots \dots \dots (9)$$

Berikut ini adalah studi kasus FuSM perilaku Kalajengking pada *Stage 2* :
 Contoh kasus, misal pemain berada pada jarak 750 dari kalajengking dan kekuatan kalajengking saat itu 80, maka untuk mengetahui perilaku pergerakan kalajengking sebagai berikut :

Diketahui :

- Jarak terhadap musuh = 750
- Kesehatan = 80

a. Fuzzifikasi



Gambar 3.22 Fungsi Keanggotaan Nilai Jarak 750

Dengan fungsi keanggotaan sebagai berikut:

$$\mu_{jsangatdekat} [x] = \begin{cases} 1; & x \leq 100 \\ \frac{400 - x}{300}; & 100 \leq x \leq 400 \\ 0; & x \geq 400 \end{cases}$$

$$\mu_{jdekat} [x] = \begin{cases} \frac{x - 100}{300}; & 100 \leq x \leq 400 \\ \frac{700 - x}{300}; & 400 \leq x \leq 700 \\ 0; & x \geq 700 \text{ atau } x \leq 100 \end{cases}$$

$$\mu_{j\text{sedang}} [x] = \begin{cases} \frac{x - 400}{300}; & 400 \leq x \leq 700 \\ \frac{1000 - x}{300}; & 700 \leq x \leq 1000 \\ 0; & x \geq 1000 \text{ atau } x \leq 400 \end{cases}$$

$$\mu_{j\text{jauh}} [x] = \begin{cases} 0; & x \leq 700 \\ \frac{x - 700}{300}; & 700 \leq x \leq 1000 \\ 1; & x \geq 1000 \end{cases}$$

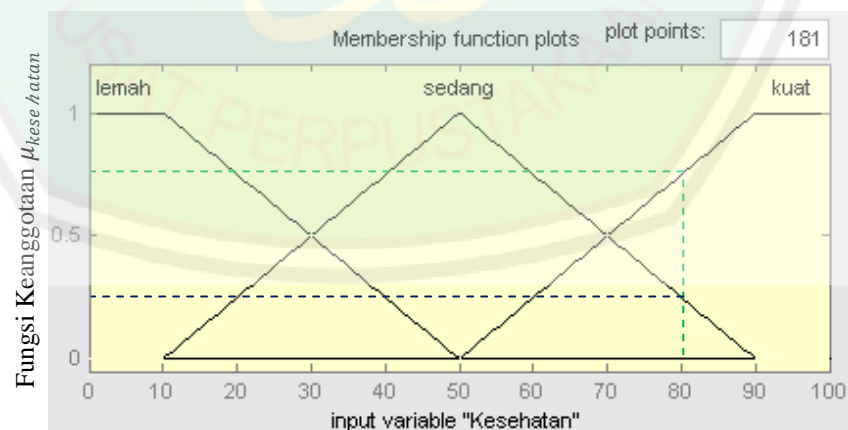
Nilai jarak 750 termasuk kedalam himpunan fuzzy sedang dan jauh dengan tingkat keanggotaan sesuai fungsi sebagai berikut:

$$\mu_{j\text{sangatdekat}} (750) = 0$$

$$\mu_{j\text{dekat}} (750) = 0$$

$$\mu_{j\text{sedang}} (750) = \frac{1000 - 750}{300} = 0,8$$

$$\mu_{j\text{jauh}} (750) = \frac{750 - 700}{300} = 0,2$$



Gambar 3.23 Fungsi Keanggotaan Nilai Kesehatan 80

Dengan fungsi keanggotaan sebagai berikut:

$$\mu_{klemah}[x] = \begin{cases} 1; & x \leq 10 \\ \frac{50-x}{40}; & 10 \leq x \leq 50 \\ 0; & x \geq 50 \end{cases}$$

$$\mu_{ksedang}[x] = \begin{cases} \frac{x-10}{40}; & 10 \leq x \leq 50 \\ \frac{90-x}{40}; & 50 \leq x \leq 90 \\ 0; & x \geq 90 \text{ atau } x \leq 10 \end{cases}$$

$$\mu_{kuat}[x] = \begin{cases} 0; & x \leq 50 \\ \frac{x-50}{40}; & 50 \leq x \leq 90 \\ 1; & x \geq 90 \end{cases}$$

Nilai kesehatan 80 termasuk kedalam himpunan fuzzy sedang dan kuat dengan tingkat keanggotaan sesuai fungsi sebagai berikut:

$$\mu_{klemah}(80) = 0$$

$$\mu_{ksedang}(80) = \frac{90-80}{40} = 0,25$$

$$\mu_{kuat}(80) = \frac{80-50}{40} = 0,75$$

b. Implikasi

Dari perhitungan pada fuzzifikasi di atas diperoleh $\mu_{jsedang}(750) = 0,8$ dan $\mu_{jjauh}(750) = 0,2$ untuk fungsi jarak, dan $\mu_{ksedang}(80) = 0,25$ dan $\mu_{kuat}(80) = 0,75$ untuk fungsi kesehatan. Berdasarkan aturan – aturan yang sesuai dengan kondisi tersebut, dengan menggunakan rumus (8) maka diperoleh :

1. $\alpha_1 = \mu_{predikatR_1} = \min(0; 0) = 0$
2. $\alpha_2 = \mu_{predikatR_2} = \min(0; 0,25) = 0$
3. $\alpha_3 = \mu_{predikatR_3} = \min(0; 0,75) = 0$

4. $\alpha_4 = \mu_{predikatR\ 4} = \min(0; 0) = 0$
5. $\alpha_5 = \mu_{predikatR\ 5} = \min(0; 0,25) = 0$
6. $\alpha_6 = \mu_{predikatR\ 6} = \min(0; 0,75) = 0$
7. $\alpha_7 = \mu_{predikatR\ 7} = \min(0,8; 0) = 0$
8. $\alpha_8 = \mu_{predikatR\ 8} = \min(0,8; 0,25) = 0,25$
9. $\alpha_9 = \mu_{predikatR\ 9} = \min(0,8; 0,75) = 0,75$
10. $\alpha_{10} = \mu_{predikatR\ 10} = \min(0,2; 0) = 0$
11. $\alpha_{11} = \mu_{predikatR\ 11} = \min(0,2; 0,25) = 0,2$
12. $\alpha_{12} = \mu_{predikatR\ 12} = \min(0,2; 0,75) = 0,2$

Jika ditampilkan dalam tabel sebagai berikut :

Tabel 3.5 Rule Evaluation Fuzzy 1

		Kesehatan		
		Variabel	Lemah	Sedang = 0,25
Jarak Terhadap Musuh	Sangat Dekat	<i>MJ</i>	<i>MY</i>	<i>MY</i>
	Dekat	<i>MJ</i>	<i>MJ</i>	<i>MJ</i>
	Sedang = 0,8	<i>MJ</i>	<i>MJ = 0,25</i>	<i>MJ = 0,75</i>
	Jauh = 0,2	<i>L</i>	<i>L = 0,2</i>	<i>MJ = 0,2</i>

Diketahui ada 4 rule yang akan aktif, dengan proses defuzzifikasi maka akan didapatkan 1 output sebagai keputusan respon perilaku NPC kalajengking.

c. Defuzzifikasi

Dengan menggunakan rumus (9) diperoleh nilai output:

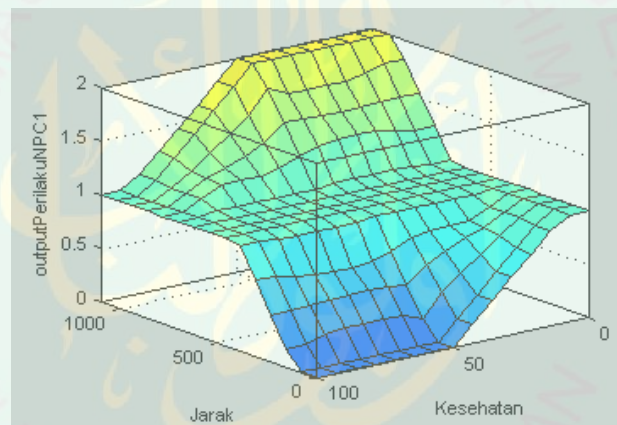
$$k = \max(\alpha_1, \alpha_2, \dots, \alpha_n)$$

$$k = \max(0; 0; 0; 0; 0; 0; 0; 0; 0,25; 0,75; 0; 0,2; 0,2)$$

$$k = 0,75$$

Diketahui index tertinggi adalah 0,75 dengan output perilaku MJ atau Mengejar. Jadi, Mengejar adalah respon perilaku NPC kalajengking pada saat jarak 750 dan kekuatan 80.

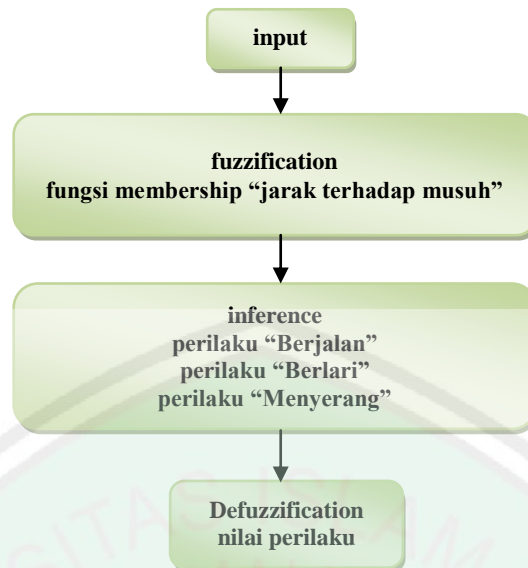
d. Respon *Fuzzy* Perilaku NPC Kalajengking dalam Grafik Permukaan



Gambar 3.24 Respon *fuzzy* perilaku NPC 1 (kalajengking)

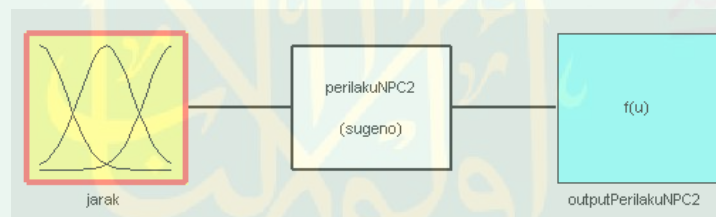
3.3.7 Desain *Fuzzy* NPC 2 (Pemburu Kalajengking)

Variabel yang digunakan untuk merancang perilaku NPC 2 adalah variabel “jarak terhadap musuh”. Logika *fuzzy* untuk menghasilkan perilaku NPC 2 ditunjukkan dalam Gambar 3.22



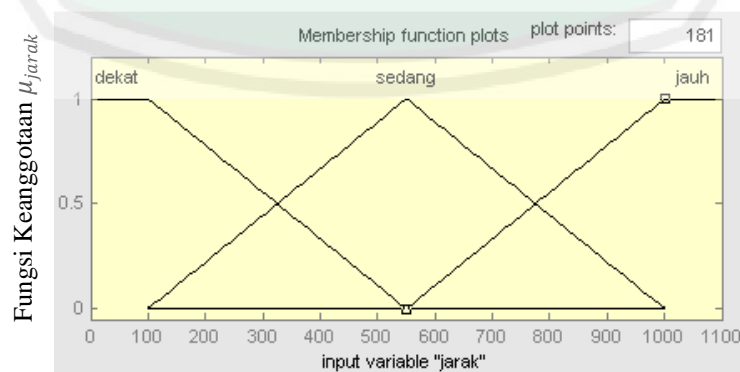
Gambar 3.25 Logika *Fuzzy* untuk menghasilkan perilaku NPC 2

Desain *fuzzy* untuk menghasilkan perilaku NPC 2 dapat dilihat pada Gambar 3.23.



Gambar 3.26 Desain *Fuzzy* untuk menghasilkan perilaku NPC 2

Gambar 3.24 menunjukkan derajat keanggotaan (*membership degree*) untuk *input* jarak terhadap musuh yang mempunyai nilai dalam interval antara 0 sampai 1100.



Gambar 3.27 Derajat keanggotaan untuk *input* jarak

Untuk *input* “jarak” mempunyai beberapa variabel linguistik serta notasinya dengan interval nilai beragam, yang ditentukan seperti disusun dalam Tabel 3.5

Tabel 3.6 Variabel linguistik *input* jarak terhadap musuh NPC 2

Input jarak		
Variabel	Notasi	Nilai
D	Dekat	0 – 550
S	Sedang	100 – 1000
Jauh	Jauh	550 – 1100

Dengan fungsi keanggotaan jarak sebagai berikut :

$$\mu_{jrkdekat} [x] = \begin{cases} 1; & x \leq 100 \\ \frac{550 - x}{450}; & 100 \leq x \leq 550 \\ 0; & x \geq 550 \end{cases}$$

$$\mu_{jrksedang} [x] = \begin{cases} \frac{x - 100}{450}; & 100 \leq x \leq 550 \\ \frac{1000 - x}{450}; & 550 \leq x \leq 1000 \\ 0; & x \geq 1000 \text{ atau } x \leq 100 \end{cases}$$

$$\mu_{jrkjauh} [x] = \begin{cases} 0; & x \leq 550 \\ \frac{x - 550}{450}; & 550 \leq x \leq 1000 \\ 1; & x \geq 1000 \end{cases}$$

Untuk *output* perilaku NPC 2 dijelaskan melalui Tabel 3.6, nilai linguistiknya dibagi menjadi 3, dengan notasi berjalan (J), berlari (L) dan menyerang (MY) dengan nilai 0. Gambar 3.25 menunjukkan keanggotaan untuk *output* pada NPC 2

Tabel 3.7 Variabel linguistik *output* NPC 2

Output NPC 2	
Variabel	Notasi
J	Berjalan
L	Berlari
MY	Menyerang

**Gambar 3.28** Keanggotaan *output* perilaku NPC 2

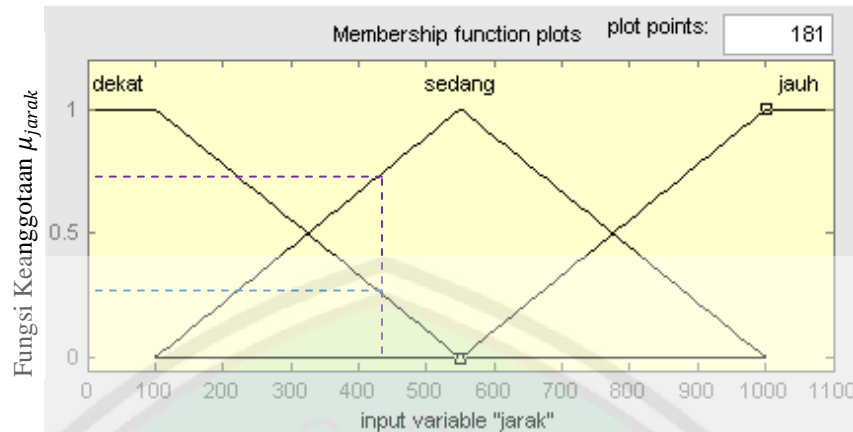
Aturan *fuzzy* untuk menghasilkan perilaku NPC 2 sebagai berikut :

1. **IF** Jarak Musuh =dekat **THEN** Perilaku =MY
2. **IF** Jarak Musuh =sedang **THEN** Perilaku =L
3. **IF** Jarak Musuh =jauh **THEN** Perilaku =J

Contoh kasus, misal kalajengking berada pada jarak 430 dari pemburu kalajengking, maka untuk mengetahui perilaku pergerakan pemburu kalajengking sebagai berikut :

Diketahui :

- Jarak terhadap musuh = 430



Gambar 3.29 Fungsi keanggotaan nilai jarak 430

Dengan fungsi keanggotaan sebagai berikut:

$$\mu_{jrkdekat} [x] = \begin{cases} 1; & x \leq 100 \\ \frac{550 - x}{450}; & 100 \leq x \leq 550 \\ 0; & x \geq 550 \end{cases}$$

$$\mu_{jrksedang} [x] = \begin{cases} \frac{x - 100}{450}; & 100 \leq x \leq 550 \\ \frac{1000 - x}{450}; & 550 \leq x \leq 1000 \\ 0; & x \geq 1000 \text{ atau } x \leq 100 \end{cases}$$

$$\mu_{jrkjauh} h[x] = \begin{cases} 0; & x \leq 550 \\ \frac{x - 550}{450}; & 550 \leq x \leq 1000 \\ 1; & x \geq 1000 \end{cases}$$

Nilai kesehatan 430 termasuk kedalam himpunan fuzzy lemah dan sedang dengan tingkat keanggotaan sesuai fungsi sebagai berikut:

$$\mu_{jrkdekat} (430) = \frac{550-430}{450} = 0,3$$

$$\mu_{jrksedang} (430) = \frac{430-100}{450} = 0,5$$

$$\mu_{jrkjauh} h (430) = 0$$

Sehingga dari perhitungan di atas bisa ditarik kesimpulan bahwa pemburu kalajengking akan mengejar dengan berlari.

3.4 Kebutuhan Sistem

Pada bagian spesifikasi kebutuhan sistem ini, diulas tentang kebutuhan sistem perangkat lunak maupun perangkat keras yang mendukung dalam pembuatan maupun pada saat pengoperasian program aplikasi.

a. Kebutuhan Perangkat Keras (*hardware*)

Kebutuhan perangkat keras yang digunakan untuk mendukung proses pengembangan sistem aplikasi *game* ini adalah:

1. Processor CoreSolo T1350 1,86GHz
2. RAM (Random Acces Memory) 1 GB.
3. VGA 128 MB
4. Hardisk 75 GB
5. Keyboard
6. Mouse
7. LCD/Monitor yang mendukung resolusi 1280x800 pixel.
8. Speaker

b. Kebutuhan Perangkat Lunak (*software*)

Perangkat lunak (*software*) yang digunakan untuk mendukung dalam pembuatan atau pengoperasian aplikasi *game* ini, antara lain:

1. Sistem Operasi yang digunakan adalah Microsoft Windows7. Sistem operasi windows merupakan sistem operasi yang user friendly, serta mendukung 3D *Game Studio* yang akan digunakan dalam membangun *game*.

2. Software 3D *Game Studio*

3D *Game Studio* atau 3DGS juga dikenal sebagai *Gamestudio*, adalah sistem pengembang permainan komputer 3D yang memperbolehkan pengguna membuat permainan 3D dan aplikasi realitas maya. *Gamestudio* meliputi model/terrain editor, level editor, script editor/debugger. Contoh permainan yang dibuat dengan *Gamestudio* adalah *Kabus 22* (2006, Son Isik LTD, Merscom LLC) dan *Citroën C4 Robot* (2008, 2GEN Studio, Citroën Turkey).

3. Adobe Photoshop

Adobe Photoshop adalah program pengolah grafis yang sangat baik dan hampir mendukung semua format gambar. Dalam pembuatan aplikasi *game* ini banyak perannya diantaranya untuk pembuatan tombol, pembuatan background dan pewarnaan *skin* objek 3D

4. 3D Studio Max

3D Studio Max adalah software dimensi tiga yang dapat membuat objek dimensi tiga tampak realistis. Keunggulan yang dimiliki adalah kemampuannya dalam menggabungkan objek image, vektor dan tiga dimensi, serta langsung dapat menganimasikan objek tersebut. Dalam *game* ini 3D Studio Max digunakan untuk pembuatan model selain yang telah dibuat di MED 3DGS.

5. Pinnacle Studio 14

Pinnacle Studio 14 adalah software pengolah video. Dalam pembuatan video intro dan penutup pada *game* ini menggunakan Pinnacle Studio 14 dengan cara menggabungkan beberapa gambar sehingga terlihat seperti video slide show.

3.5 Cara Memainkan *Game*

Cara menjalankan *game* menggunakan kombinasi antara *mouse* dan *keyboard*, dengan kombinasi tersebut *game* bisa dimainkan sesuai kebutuhan untuk mencapai misi *game*. Cara tersebut ditunjukkan dalam tabel 3.1 berikut ini.

Tabel 3.8 Fungsi *Keyboard* dan *Mouse* untuk Memainkan *Game*

Perintah	Fungsi
Tombol A,W, D, S	Berjalan ke kiri, depan, kanan, mundur ke belakang
Tombol anak panah atas, kanan, kiri, bawah	Berjalan ke depan, kanan, kiri, mundur ke belakang
Mouse klik kiri	Memilih jawaban dari kuis
Geser mouse ke kanan, kiri, atas, bawah	Menggeser layar tampilan ke kanan, kiri, atas, bawah

3.6 Kebutuhan PC Pemain

Dalam memainkan *game* ini, komputer yang dipakai harus memenuhi standar spesifikasi sehingga *game* bisa berjalan dengan baik dan lancar. Berikut ini merupakan tabel daftar spesifikasi komputer yang harus dimiliki pemain untuk memainkan *game* ini.

Tabel 3.9 Kebutuhan PC Pemain

Kebutuhan	Spesifikasi Minimum	Spesifikasi Rekomendasi
Operating System	Windows XP, Vista, 7	Windows XP, Vista, 7
CPU	Pentium 4 1,7GHz	Dual Core 2,1 GHz
RAM	512 MB	1 GB
VGA Card	128 MB, 1024 x 768	256 MB, 1024 x 768
Sound Card	Support DirectX 9.0c	Support DirectX 9.0c
Keyboard & Mouse	-	-

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

Bab ini membahas tentang implementasi dari perancangan yang dibuat sebelumnya. Selain itu juga melakukan pengujian terhadap aplikasi yang dibuat untuk mengetahui apakah aplikasi tersebut telah berjalan sesuai dengan tujuan yang ingin dicapai.

4.2 Implementasi *Fuzzy State Machine* (FuSM) Perilaku NPC

Proses Implementasi adalah proses pembangunan komponen-komponen pokok suatu sistem yang didasarkan pada desain dan rancangan yang telah dibuat sebelumnya. Implementasi perancangan *Artificial Intelligence* pada penelitian ini diterapkan pada pengaturan respon perilaku NPC dengan metode *Fuzzy State Machine* (FuSM).

4.2.1 FuSM Respon Perilaku NPC 1 (Kalajengking) Pada *Stage* Kedua

Pada bagian ini membahas mengenai implementasi *Fuzzy State Machines* (FuSM) untuk pengaturan respon perilaku NPC 1 (Kalajengking) sehingga dihasilkan output perilaku yang lebih natural. FuSM bekerja pada saat pemain bertemu dengan NPC kalajengking sehingga kalajengking akan berperilaku seperti kalajengking di dunia nyata yaitu menyerang. Kalajengking akan mempengaruhi kesehatan pemain jika kalajengking sampai menyerang pemain. FuSM diterapkan pada bahasa pemrograman *Lite-C*. Berikut ini adalah *Script* FuSM perilaku Kalajengking pada *Stage* 2.


```

#define ANIMATION skill12
#define STATE skill11
var kuatankalajengking=90;
var animasitenaga=1;
var animA;
var tenagaku;
var animB;
var animtenaga=0;
var animtenaga_tambah=0;
STRING* kolo="";
ENTITY* koloj;

action scorpio()
{
    koloj=me;
    var a=0;
    var d=1;
    my.STATE=1;

    while (1)
    {
        VECTOR temp[3];
        vec_set(temp.x, my.x);
        temp.z -= 1000;
        my.z -= c_trace(my.x, temp.x, IGNORE_PASSABLE) - 17;
    }

    if(my.STATE == 1)
    {
        VECTOR vDirection;
        VECTOR vdir;
        ANGLE vTargetAngle;
        vec_diff(vDirection,vdir,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 3 * time_step;
        ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);

        if(vec_dist (pemain.x, my.x) < 1100){
            my.STATE=2;
        }
    }

    if(my.STATE==2){

        if((vec_dist (pemain.x, my.x) <= 1000) && (vec_dist (pemain.x, my.x)
        >= 700) && (kuatankalajengking > tenagaku))
        {
            VECTOR vDirection;
            ANGLE vTargetAngle;
            vec_diff(vDirection,pemain.x,my.x);
            vec_to_angle(vTargetAngle,vDirection);
            my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
            c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
            my.ANIMATION += 3 * time_step;
            ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
        }
    }
}

```

```

if((vec_dist (pemain.x, my.x) <= 1000) && (vec_dist (pemain.x, my.x)
>= 700) && (kekuatankalajengking == tenagaku))
{
    VECTOR vDirection;
    ANGLE vTargetAngle;
    vec_diff(vDirection,pemain.x,my.x);
    vec_to_angle(vTargetAngle,vDirection);
    my.pan += time_step * sign(ang(vTargetAngle.pan + my.pan));
    c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
    my.ANIMATION += 3 * time_step;
    ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
}

if((vec_dist (pemain.x, my.x) <= 1000) && (vec_dist (pemain.x, my.x)
>= 700) && (kekuatankalajengking < tenagaku))
{
    VECTOR vDirection;
    ANGLE vTargetAngle;
    vec_diff(vDirection,pemain.x,my.x);
    vec_to_angle(vTargetAngle,vDirection);
    my.pan += time_step * sign(ang(vTargetAngle.pan + my.pan));
    c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
    my.ANIMATION += 3 * time_step;
    ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
}

if((vec_dist (pemain.x, my.x) <= 700) && (vec_dist (pemain.x, my.x)
>= 300) && (kekuatankalajengking > tenagaku))
{
    VECTOR vDirection;
    ANGLE vTargetAngle;
    vec_diff(vDirection,pemain.x,my.x);
    vec_to_angle(vTargetAngle,vDirection);
    my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
    c_move (my, vector(2 * time_step, 0, 0), nullvector, GLIDE);
    my.ANIMATION += 5 * time_step;
    ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
}

if((vec_dist (pemain.x, my.x) <= 700) && (vec_dist (pemain.x, my.x)
>= 300) && (kekuatankalajengking == tenagaku))
{
    VECTOR vDirection;
    ANGLE vTargetAngle;
    vec_diff(vDirection,pemain.x,my.x);
    vec_to_angle(vTargetAngle,vDirection);
    my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
    c_move (my, vector(2 * time_step, 0, 0), nullvector, GLIDE);
    my.ANIMATION += 5 * time_step;
    ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
}

if((vec_dist (pemain.x, my.x) <= 700) && (vec_dist (pemain.x, my.x)
>= 300) && (kekuatankalajengking < tenagaku))
{
    VECTOR vDirection;
    ANGLE vTargetAngle;
    vec_diff(vDirection,pemain.x,my.x);
    vec_to_angle(vTargetAngle,vDirection);
    my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
    c_move (my, vector(2 * time_step, 0, 0), nullvector, GLIDE);
    my.ANIMATION += 5 * time_step;
    ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
}

```

```

    }

    if((vec_dist (pemain.x, my.x) <= 300) && (vec_dist (pemain.x, my.x)
    >= 80) && (kekuatankalajengking > tenagaku))
    {
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection,pemain.x,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(3 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 7 * time_step;
        ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
    }

    if((vec_dist (pemain.x, my.x) <= 300) && (vec_dist (pemain.x, my.x)
    >= 80) && (kekuatankalajengking == tenagaku))
    {
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection,pemain.x,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(3 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 7 * time_step;
        ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
    }

    if((vec_dist (pemain.x, my.x) <= 300) && (vec_dist (pemain.x, my.x)
    >= 80) && (kekuatankalajengking < tenagaku))
    {
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection,pemain.x,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(3 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 7 * time_step;
        ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
    }

    if((vec_dist (pemain.x,my.x)<80) && (kekuatankalajengking > tenagaku))
    {
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection,pemain.x,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 3 * time_step;
        ent_animate(me, "scatt", my.ANIMATION , ANM_CYCLE);

        tenagaku-=20;
        animtenaga_tambah-=0.095;
        detikx=0;
        wait(250);
        if(tenagaku<(-15))
        {
            kesehatanplayer=0;
            viewterkunci=1;
            reset (panBgTenaga , SHOW);
            gameplaying=0;
            waktudihentikan=1;
        }
    }

```

```

        mati();
    }
}

if((vec_dist(pemain.x,my.x)<80)&&(kekuatankalajengking == tenagaku))
{
    VECTOR vDirection;
    ANGLE vTargetAngle;
    vec_diff(vDirection,pemain.x,my.x);
    vec_to_angle(vTargetAngle,vDirection);
    my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
    c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
    my.ANIMATION += 3 * time_step;
    ent_animate(me, "scatt", my.ANIMATION , ANM_CYCLE);

    tenagaku-=20;
    animtenaga_tambah-=0.015;
    detikx=0;
    wait(250);
    if(tenagaku<(-15))
    {
        kesehatanplayer=0;
        viewterkunci=1;
        reset(panBgTenaga ,SHOW);
        gameplaying=0;
        waktudihentikan=1;
        mati();
    }
}

if((vec_dist(pemain.x,my.x)<80)&&(kekuatankalajengking < tenagaku))
{
    VECTOR vDirection;
    ANGLE vTargetAngle;
    vec_diff(vDirection,pemain.x,my.x);
    vec_to_angle(vTargetAngle,vDirection);
    my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
    c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
    my.ANIMATION += 3 * time_step;
    ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
}
}

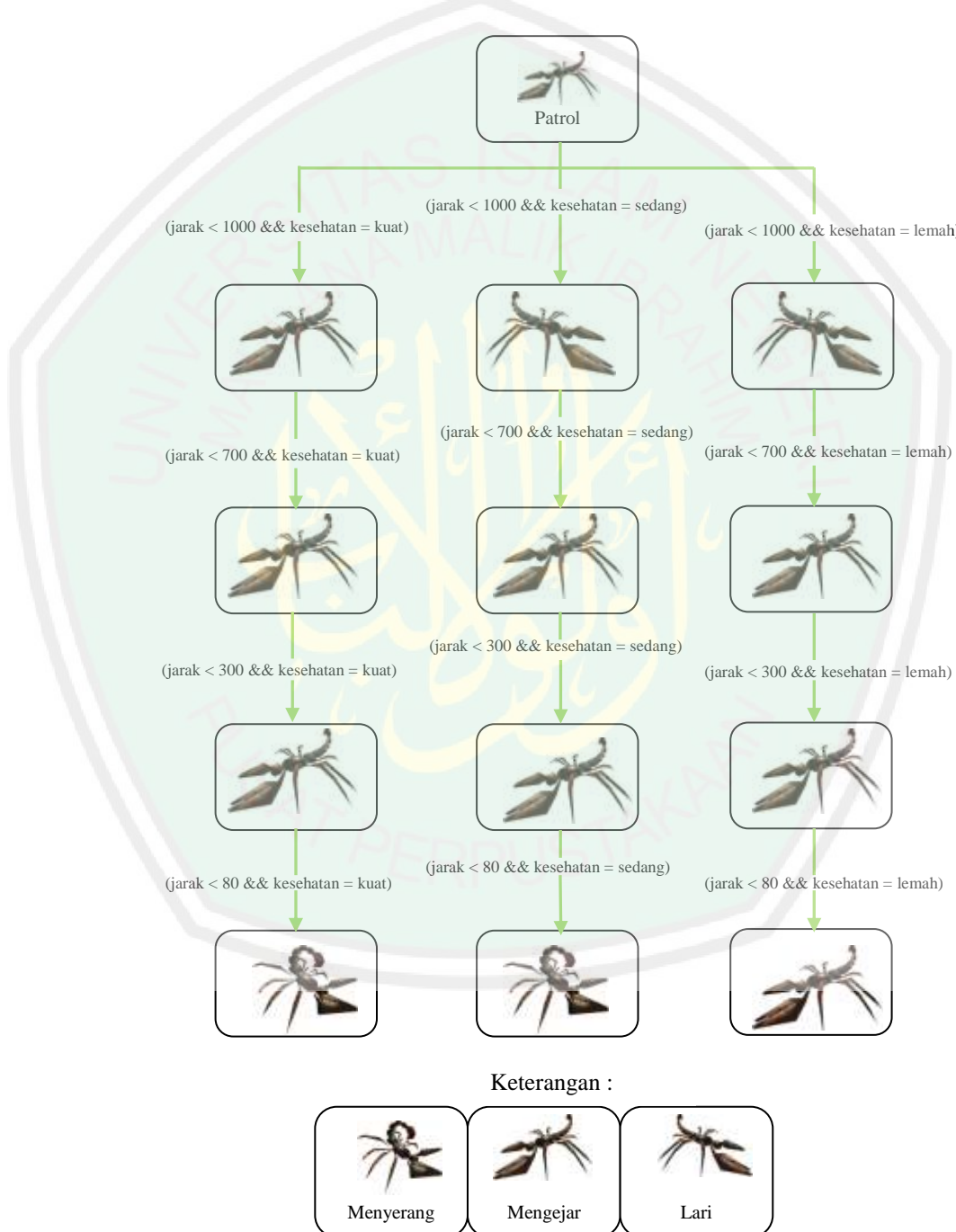
if(vec_dist (pemain.x, my.x) > 1000)
{
    my.STATE=1;
}

if(kekuatankalajengking<=0)
{
    my.STATE=3;
}

if(my.STATE == 3)
{
    my.ANIMATION += 3*time_step;
    ent_animate(me,"scd",my.ANIMATION,0);
    if (my.ANIMATION > 70)
    ent_remove (me);
    return;
}
wait (1);
}}

```

Script di atas mengimplementasikan *Fuzzy State Machine* (FuSM) yaitu mengatur tingkah laku atau prinsip kerja NPC kalajengking. Berikut ini sebagai gambaran dari hasil *run Script* di atas.



Gambar 4.1 Implementasi FuSM Perilaku NPC 1 Kalajengking (*Stage* Kedua)

Ketika pemain tidak berada pada pandangan NPC kalajengking, maka posisi *state* adalah *state* 1, yaitu berupa patrol / berjalan bebas, ketika pemain berada dalam jangkauan NPC kalajengking, maka *state* bertransisi ke *state* 2 ditandai dengan berubahnya perilaku NPC kalajengking menjadi mengejar pemain, yaitu pada *Script* :

```
my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);
```

Script di atas maksudnya adalah `my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));` sebagai fungsi untuk mengarahkan NPC menuju ke keberadaan pemain. Sedangkan `c_move (my, vector(1 * time_step, 0, 0), nullvector, GLIDE);` berfungsi untuk memindahkan NPC bergerak per satuan waktu. Setelah *Script* di atas, diikuti dengan *Script* :

```
my.ANIMATION += 3 * time_step;
ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
```

maksudnya adalah NPC kalajengking yang berupa model 3 dimensi beserta animasinya akan diaktifkan melalui *Script* di atas. `my.ANIMATION += 3 * time_step;` adalah fungsi yang menentukan nilai animasi model. Sedangkan `ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);` adalah fungsi untuk mengaktifkan animasi model sesuai nama pada frame animasinya. Sehingga bisa disimpulkan bahwa *Script* tersebut mengaktifkan animasi "walk" dengan kecepatan animasi `3 * time_step` dan bergerak mendekati karakter pemain. FuSM ditanamkan pada *state* 2. Pada *Script* di atas jelas bahwa ada beberapa kondisi yang menentukan aktifasi *state* baru di dalam *state* 2. Semisal `if((vec_dist(pemain.x, my.x)<=1000) && (vec_dist (pemain.x, my.x) >= 700) && (kekuatankalajengking > tenagaku))` artinya jarak NPC kalajengking dengan

pemain kurang dari sama dengan 1000 dan lebih dari sama dengan 700, serta kekuatan kalajengking pada kondisi kuat, karena kekuatan kalajengking lebih dari kekuatan pemain, sehingga pada blok kondisi tersebut akan diaktifkan. Jika kondisi berubah, maka *state* yang diaktifkan juga berubah. Hal ini diatur sesuai desain *fuzzy* yang dijelaskan pada bab sebelumnya.

Pada intinya, perilaku NPC ini dipengaruhi oleh 2 inputan yaitu jarak terhadap musuh, dalam hal ini adalah pemain, dan kekuatan. Dari kedua inputan tadi akan dihasilkan variasi output respon perilaku NPC, sehingga NPC tidak bersifat statis dan bisa berperilaku lebih natural.

4.2.2 FuSM Respon Perilaku NPC 2 (Pemburu Kalajengking) Pada *Stage*

Kedua

Pada bagian ini membahas mengenai implementasi *Fuzzy State Machines* (FuSM) untuk respon perilaku NPC 2 (Pemburu Kalajengking). FuSM bekerja saat kalajengking berada dalam jangkauan pemburu kalajengking. Berikut ini adalah *Script* dari FuSM respon perilaku NPC 2 :

```
#define ANIMATION skill2
#define STATE skill1
#include "scorpio.c"

ENTITY* wonga;
STRING* anim="";
action wong()
{
    wonga=me;
    my.STATE=1;
    while (1)
    {
        VECTOR temp[3];
        vec_set(temp.x, my.x);
        temp.z -= 1000;
        my.z -= c_trace(my.x, temp.x, IGNORE_PASSABLE) - 80;
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection, pemain.x, my.x);
        vec_to_angle(vTargetAngle, vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
    }
}
```

```

if(my.STATE == 1)
{
    vec_to_angle(vTargetAngle,vDirection);
    my.ANIMATION += 3 * time_step;
    ent_animate(me, "stand", my.ANIMATION , ANM_CYCLE);

    if(vec_dist (koloj.x, my.x) < 1100){
        my.STATE=2;
    }
}

if(my.STATE==2)
{
    if(kekuatankalajengking <=0){
        my.ANIMATION += 3 * time_step;
        ent_animate(me, "stand", my.ANIMATION , ANM_CYCLE);
    }

    if((vec_dist (koloj.x, my.x) < 1000))
    {
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection,koloj.x,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(3 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 3 * time_step;
        ent_animate(me, "walk", my.ANIMATION , ANM_CYCLE);
    }

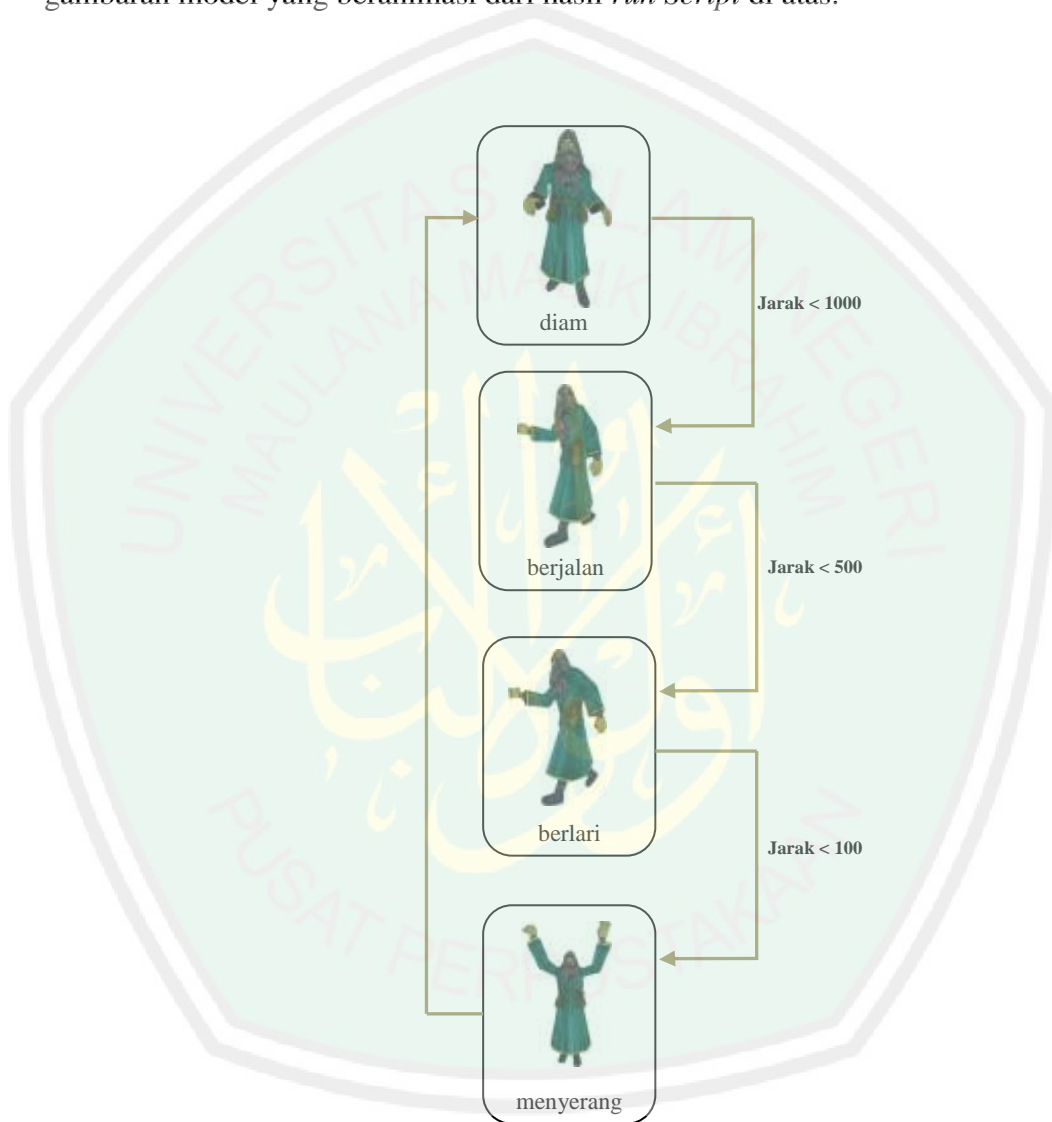
    if((vec_dist (koloj.x, my.x) < 500))
    {
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection,koloj.x,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(3 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 3 * time_step;
        ent_animate(me, "run", my.ANIMATION , ANM_CYCLE);
    }

    if((vec_dist (koloj.x, my.x) < 100))
    {
        VECTOR vDirection;
        ANGLE vTargetAngle;
        vec_diff(vDirection,koloj.x,my.x);
        vec_to_angle(vTargetAngle,vDirection);
        my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
        c_move (my, vector(3 * time_step, 0, 0), nullvector, GLIDE);
        my.ANIMATION += 3 * time_step;
        ent_animate(me, "attack", my.ANIMATION , ANM_CYCLE);
        kekuatankalajengking-=10;
        wait(350);
    }
}

wait (1);
}
}

```


Script di atas mengimplementasikan *Fuzzy State Machines* (FuSM) yaitu mengatur tingkah laku NPC 2 (Pemburu Kalajengking). Berikut ini sebagai gambaran model yang beranimasi dari hasil *run Script* di atas.



Gambar 4.2 Implementasi FuSM Perilaku NPC 2 Pemburu Kalajengking(Stage Kedua)

Ketika pemain dalam kondisi dikejar kalajengking, maka pemain bisa mencari bantuan yaitu dengan mendekati NPC 2 atau pemburu kalajengking yang diberi nama Syekh Al Karim-3. Jika kalajengking mendekati pemain dan pemain

mendekati pemburu kalajengking, maka dalam jarak jangkauan, kalajengking akan dikejar oleh pemburu kalajengking dengan cara berjalan, berlari dan jika jaraknya sangat dekat maka kalajengking akan diserang oleh pemburu kalajengking sehingga bisa mengurangi kekuatan kalajengking dan bahkan bisa membunuh kalajengking jika kekuatan kalajengking = 0.

Script yang diterapkan sama seperti *Script* sebelumnya. Sebelum ada pemicu, NPC 2 pada keadaan *idle*. Jika ada NPC 1 yang berada dalam jangkauannya maka akan mengaktifkan *state* baru sesuai dengan inputan jarak. Sebagai contoh pada *Script* `if((vec_dist (koloj.x, my.x) < 100))` artinya adalah suatu kondisi dimana NPC 1 berada pada jarak kurang 100 dari NPC 2. Sehingga dalam kondisi seperti itu akan ada blok aksi yang diaktifkan

```
my.pan += time_step * sign(ang(vTargetAngle.pan - my.pan));
c_move (my, vector(3 * time_step, 0, 0), nullvector, GLIDE);
my.ANIMATION += 3 * time_step;
ent_animate(me, "attack", my.ANIMATION , ANM_CYCLE);
kekuatankalajengking-=10;
```

Script di atas maksudnya adalah NPC 2 akan bergerak menuju NPC 1, dan ketika jarak diantara keduanya kurang dari 100 maka animasi "attack" diaktifkan yang artinya NPC 2 menyerang NPC 1. Jika kondisi ini terjadi maka kekuatan kalajengking berkurang 10 tiap satuan waktu, ditandai dengan *Script* `kekuatankalajengking-=10;`

4.3 Implementasi Aplikasi *Game*

Implementasi merupakan proses pembangunan komponen-komponen pokok suatu sistem, yang didasarkan pada desain dan rancangan yang telah dibuat sebelumnya.

4.3.1 Antarmuka *Intro*

Pada bagian ini video *intro* pertama menampilkan berisil tentang cerita *game*. Maksud dari *intro* pertama ini dibuat adalah untuk memberikan informasi tentang awal cerita dari *game* ini. Untuk menghentikan video ini pemain dapat melakukannya dengan cara menekan tombol Enter pada keyboard.



Gambar 4.3 Video *Intro* 1



Gambar 4.4 Video *Intro* 2

4.3.2 Antarmuka Menu

Menu dalam *game* ini terdiri dari 5 pilihan, antara lain “keluar”, “cara bermain”, “developer”, “pengaturan” dan “mulai”. Berikut ini fungsi dari masing-masing tombol:

1. Tombol Keluar, berfungsi untuk keluar dari permainan.
2. Tombol Cara Bermain, berfungsi untuk memunculkan keterangan bantuan. Bantuan di sini berupa bagaimana cara menjalankan permainan.
3. Tombol Developer, berfungsi untuk memunculkan kredit yang berisi tentang ucapan terimakasih dan keterangan pembuat *game*.
4. Tombol Pengaturan, berfungsi untuk memunculkan pengaturan besar-kecilnya volume suara *background* musik dan suara efek.
5. Tombol Mulai, berfungsi untuk memulai permainan.



Gambar 4.5 Tampilan Pilihan Menu



Gambar 4.6 Menu Keluar



Gambar 4.7 Menu Cara Bermain



Gambar 4.8 Menu Developer



Gambar 4.9 Menu Pengaturan



Gambar 4.10 Menu Mulai

4. 3. 3 *Scene Game* pada Bagian Awal



Gambar 4.11 *Scene* pada Bagian Awal

Setelah pemain meng-klik tombol mulai, maka pemain akan memasuki dan memulai permainan. Skenarionya yaitu, pemain harus menemui seseorang bernama Syeikh Al Karim-1 yang berjubah merah dan melambaikan tangannya kepada pemain. Setelah pemain menemuinya, maka pemain akan diberi informasi tentang *stage* selanjutnya yaitu *stage* pertama.

4.3.4 Scene Game pada Stage Pertama



Gambar 4.12 Scene pada Stage Pertama

Setelah pemain menemui Syekh AL Karim, maka pemain otomatis akan masuk ke dalam ruangan dimana ruangan ini adalah *stage* pertama permainan. Pada *stage* ini sekenarionya adalah pemain mencari 12 soal yang bertebaran di setiap ruang berbentuk tanda tanya. Jika pemain membutuhkan bantuan bermain pada *stage* pertama, maka pemain bisa bertanya pada Syekh Al Karim-2 yang ada di dalam ruangan tersebut. Permainan dinyatakan selesai pada *stage* pertama apabila pemain berhasil menjawab 12 soal dan berhasil menemukan pintu keluar.

4.3.5 Scene Game pada Stage Kedua



Gambar 4.13 Scene pada Stage Kedua

Pada *stage* kedua ini pemain berada di luar ruangan. Pemain berpetualang mencari 8 soal, selain itu juga mencari tanaman atau rumput untuk makanan kuda yang akan digunakan sebagai kendaraan menuju *stage* ketiga. Jika pemain membutuhkan bantuan tentang *stage* kedua, maka pemain bisa bertanya kepada NPC masyarakat yang ada di lingkungan *stage* kedua. Pemain dinyatakan telah menyelesaikan *stage* kedua ini jika 8 soal terjawab, mempunyai minimal 1 tanaman untuk kuda, dan sudah membayar sewa kuda kepada NPC algojo bernama Khan Al Jundi. Jika semua syarat telah terpenuhi, pemain bisa langsung menunggangi kuda menuju *stage* ketiga.

Pada *stage* kedua ini, pemain harus berhati-hati dengan keberadaan kalajengking yang bisa menyerang pemain secara tiba-tiba. Akan tetapi pemain bisa berlindung dari serangan kalajengking dengan cara mencari dan mendekati

NPC pemburu kalajengking bernama Syeikh Al Karim-3. NPC ini akan menyerang kalajengking sampai kalajengking mati.

4.3.6 *Scene Game* saat Kalajengking Lari



Gambar 4.14 *Scene* saat Kalajengking Lari

NPC kalajengking akan melarikan diri jika kalajengking berada pada jarak jauh dari pemain dan jika kekuatan kalajengking lemah atau sedang. Kalajengking akan mengejar pemain jika pemain berada dalam jarak pandang kalajengking.

4.3.7 Scene Game saat Kalajengking Mengejar



Gambar 4.15 Scene saat Kalajengking Mengejar

NPC kalajengking yang semula berjalan bebas akan segera fokus untuk mengejar pemain jika pemain berada pada jangkauan kalajengking. Kalajengking akan mengejar pemain, dan jika jaraknya mulai dekat dan kekuatan kalajengking kuat atau sedang, maka kalajengking bisa menyerang pemain dan mengurangi kesehatan pemain dengan cepat.

Pada pergerakan NPC kalajengking inilah metode *Fuzzy State Machine* diterapkan. Diharapkan pergerakan perilaku kalajengking bisa sesuai dengan perilaku kalajengking secara nyata.

4.3.8 Scene Game saat Kalajengking Menyerang



Gambar 4.16 Scene saat Kalajengking Menyerang

Jika jarak kalajengking dengan pemain sangat dekat dan kekuatan kalajengking kuat atau sedang, maka secara otomatis kalajengking akan menyerang pemain dan mengurangi kesehatan pemain dengan cepat.

4.3.9 Scene Game saat Syeikh Al Karim-3 Menyerang Kalajengking



Gambar 4.17 Scene saat Syeikh Al Karim-3 Menyerang Kalajengking

NPC orang pemburu kalajengking yang semula diam bisa bergerak berjalan ataupun berlari menuju NPC kalajengking dengan maksud menyerang kalajengking sampai kalajengking mati sehingga tidak bisa mengganggu pemain lagi.

4.3.10 *Scene Game* saat Pemain Menunggangi Kuda



Gambar 4.18 *Scene* saat Pemain Menunggangi Kuda

Setelah pemain memenuhi syarat pada *Stage* kedua, maka pemain bisa langsung menunggangi kuda menuju *Stage* ketiga. Perilaku berjalannya kuda tergantung pada kondisi kuda, jumlah rumput yang dimiliki pemain dan banyaknya sewa kuda yang diberikan kepada NPC algojo Khan Al Jundi.

4.3.11 Scene Game pada Stage Ketiga

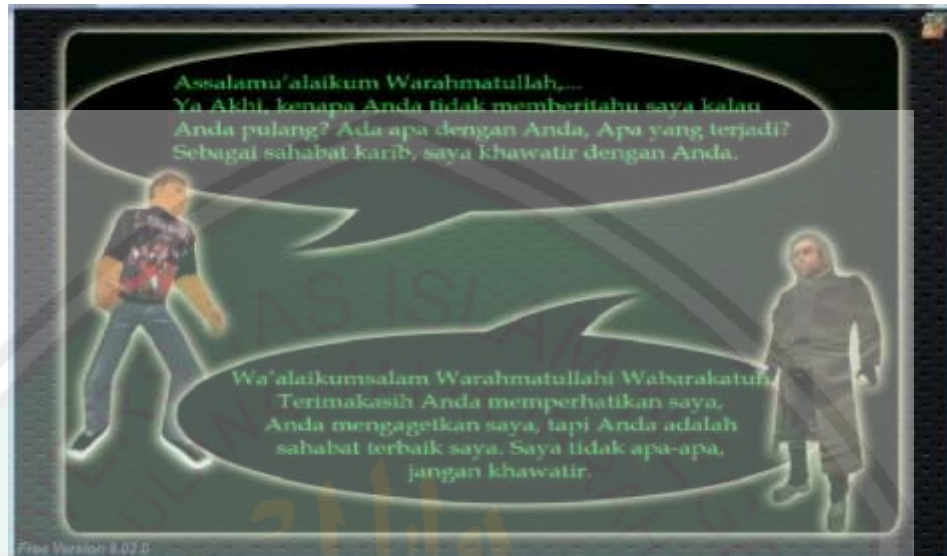


Gambar 4.19 Scene pada Stage Ketiga

Stage ketiga adalah akhir dari game ini. Pada *stage* ketiga ini, pemain berada pada sebuah lingkungan yang di tengahnya ada rumah yang dijaga oleh dua penjaga, dan di dalam rumah ini ada makam seseorang yang sedang ditunggu oleh sahabat pemain yaitu Zaid Annas. Misi terakhir adalah menemukan Zaid Annas dan dilanjutkan dengan video penutup alur cerita pada *game*.

Sesuai dengan cerita di awal bahwa misi permainan ini adalah menemukan sahabat pemain (Ahmad Alfatih) yaitu seseorang yang sedang berdiri di samping makam, sahabat pemain ini bernama Zaid Annas. Jika misi berhasil, maka Zaid Annas akan menceritakan apa yang terjadi padanya kemudian pemain akan diberikan sebuah dokumen rahasia yang berisi jumlah jawaban benar, skor TOAFL dan skor permainan.

4.3.12 Antarmuka Video Penutup



Gambar 4.20 Video Penutup Game



Gambar 4.21 Skor Game

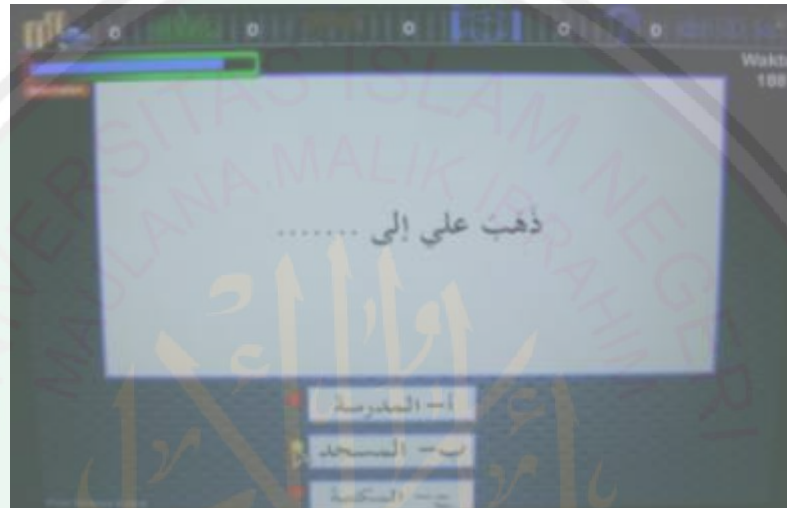
Pada video penutup ini diceritakan tentang dialog antara pemain (Ahmad Alfatih) dengan sahabatnya (Zaid Annas), yang kemudian dilanjutkan oleh

tampilan akhir berisi jumlah jawaban benar, skor TOAFL dan skor permainan.

Selanjutnya pemain bisa memilih untuk keluar dari game atau bermain lagi.

4.3.13 Beberapa Scene Sistem Menampilkan Soal

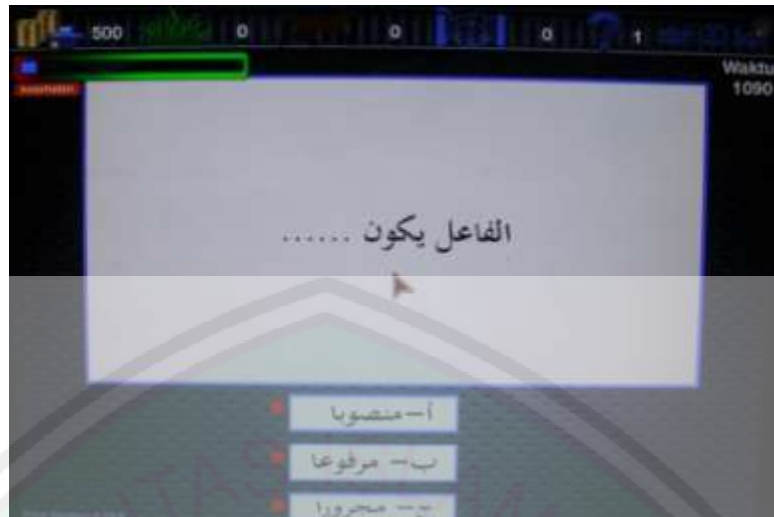
Scene soal pada stage 1 :



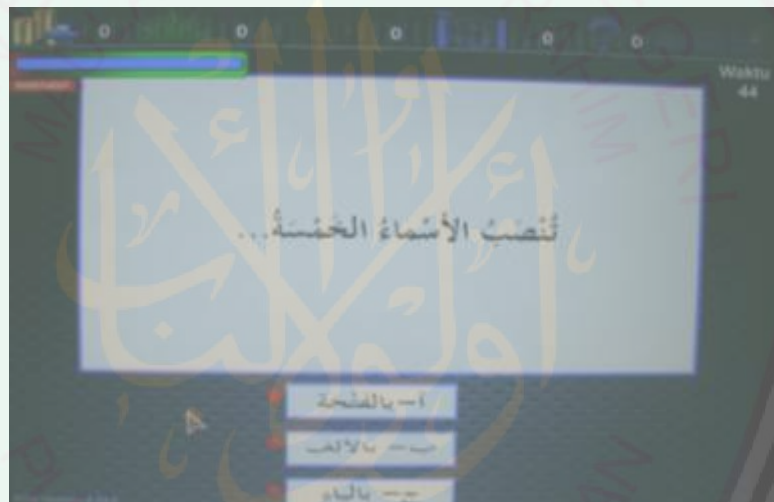
Gambar 4.22 Scene soal 1



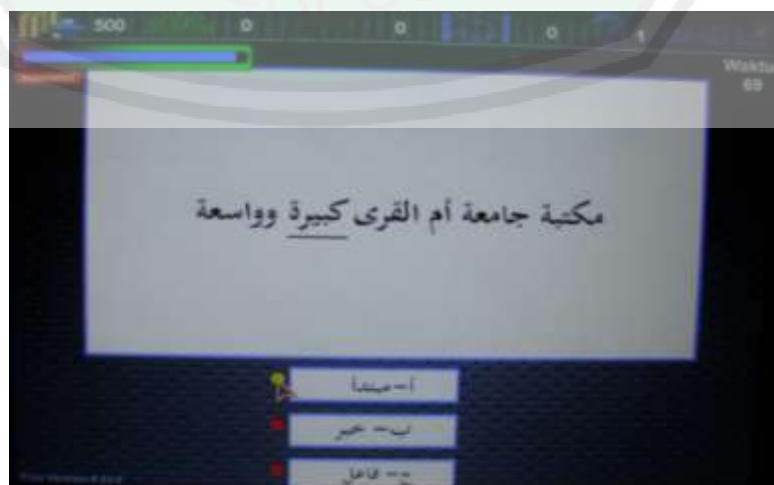
Gambar 4.23 Scene soal 2



Gambar 4.24 Scene soal 3



Gambar 4.25 Scene soal 4

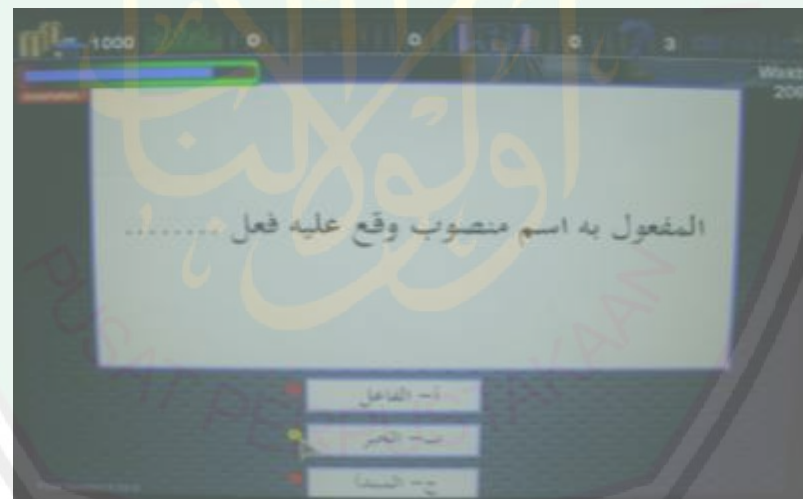


Gambar 4.26 Scene soal 5

Scene soal pada stage 2 :



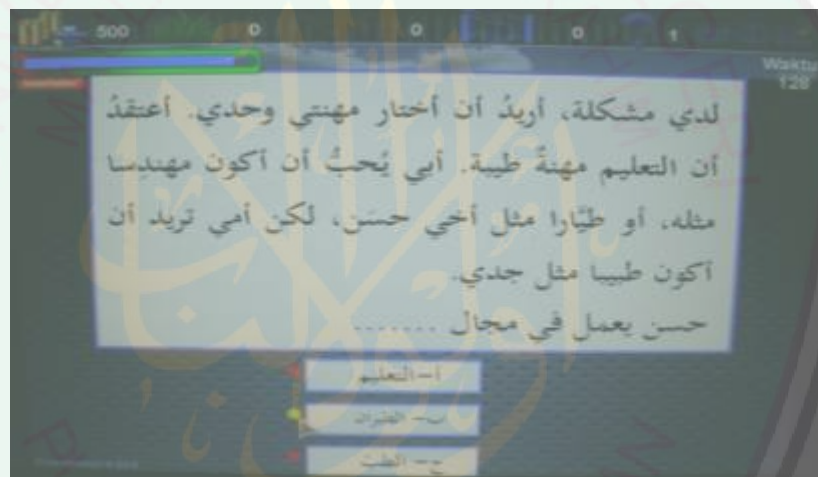
Gambar 4.27 Scene soal 6



Gambar 4.28 Scene soal 7



Gambar 4.29 Scene soal 8



Gambar 4.30 Scene soal 9



Gambar 4.31 Scene soal 10

4. 4 Pengujian *Game*

4.4.1 Rekapitulasi Hasil Pengujian Perbandingan FuSM dan FSM

Tabel 4.1 Rekapitulasi Hasil Uji Coba Perbandingan Metode

No	Nilai Input		Output	Output FSM
	Jarak	Kekuatan	FuSM	
1	761	20	Mengejar	Mengejar
2	859	75	Mengejar	Mengejar
3	148	80	Menyerang	Mengejar
4	79	85	Menyerang	Menyerang
5	950	10	Lari	Mengejar
6	462	25	Mengejar	Mengejar
7	983	95	Mengejar	Mengejar
8	505	95	Mengejar	Mengejar
9	73	90	Menyerang	Menyerang
10	916	25	Lari	Mengejar
11	890	30	Lari	Mengejar
12	221	25	Mengejar	Mengejar
13	408	40	Mengejar	Mengejar
14	59	55	Menyerang	Menyerang
15	73	65	Menyerang	Menyerang
16	846	45	Lari	Mengejar
17	360	50	Mengejar	Mengejar
18	717	60	Mengejar	Mengejar
19	850	75	Mengejar	Mengejar
20	895	80	Mengejar	Mengejar
21	84	85	Menyerang	Menyerang
22	62	15	Mengejar	Menyerang
23	64	25	Mengejar	Menyerang
24	485	70	Mengejar	Mengejar
25	80	85	Menyerang	Menyerang

Metode *Fuzzy State Machine* (FuSM) menghasilkan output lebih variatif pada desain perilaku menyerang NPC kalajengking daripada *Finite State Machine* (FSM). Selain itu, pergerakan NPC terlihat lebih halus pada implementasi FuSM daripada FSM karena alur perubahan *state* FSM lebih kaku dan sederhana, sedangkan FuSM lebih detil dan kompleks dalam penanganan *state* yang dikembangkan.

Dari data di atas dapat disimpulkan bahwa FuSM mempunyai kelebihan daripada FSM, yaitu adanya *state* yang lebih variatif sehingga NPC lebih terlihat natural dan tidak kaku, selain itu FuSM dalam perubahan respon perilaku NPC terlihat lebih halus. FuSM lebih dinamis daripada FSM, jika dalam FSM ada kemungkinan terjadinya perubahan *state* satu ke *state* tertentu dengan alur yang mudah diingat dan ditebak, sehingga mudah membuat pemain bosan, maka FuSM menangani masalah tersebut dengan mengembangkan *state* satu menjadi beberapa *state* lain yang diatur oleh logika *fuzzy*.

Sesuai dengan definisinya, *Finite State Machine* (FSM) adalah alur sistem perubahan *state* pada game, sedangkan *Fuzzy State Machine* (FuSM) adalah FSM yang digabungkan dengan logika *fuzzy* sehingga alur sistem perubahan *state* lebih variatif dan natural.

4.4.2 Rekapitulasi Hasil Uji Coba Penggunaan Memori dan CPU

Setelah melakukan uji coba penggunaan memori dan CPU ketika sistem dijalankan, menghasilkan data sebagai berikut :

Tabel 4.2 Penggunaan Memori dan CPU Metode FSM

No	Waktu	Memory	CPU
1.	00 : 20 : 25	74.280	14.01
2.	00 : 51 : 39	74.292	14.28
3.	01 : 10 : 02	75.080	15.70
4.	01 : 35 : 22	75.656	16.27
5.	01 : 56 : 64	75.712	17.85
6.	02 : 21 : 66	75.916	17.85
7.	02 : 47 : 67	75.892	17.30
8.	03 : 32 : 22	75.892	17.01
9.	04 : 00 : 30	76.212	15.41
10.	04 : 27 : 93	76.284	14.39
11.	04 : 55 : 42	76.288	13.47
12.	05 : 32 : 91	76.288	15.28
13.	06 : 22 : 77	76.300	16.32
14.	07 : 07 : 72	76.292	16.76
15.	08 : 36 : 90	76.292	17.09

$$\text{memori rata - rata FSM} = \frac{\sum_{i=1}^n \text{memori}_i}{\sum \text{pengukuran memori}}$$

$$\text{memori rata - rata FSM} = \frac{1136676}{15}$$

$$\text{memori rata - rata FSM} = 75.778$$

Setelah dihitung rata-rata penggunaan memori, FSM memakan memori rata-rata = 75.778.

$$\text{CPU rata - rata FSM} = \frac{\sum_{i=1}^n \text{CPU}_i}{\sum \text{pengukuran CPU}}$$

$$CPU \text{ rata - rata FSM} = \frac{238,99}{15}$$

$$CPU \text{ rata - rata FSM} = 15,93$$

Sedangkan penggunaan CPU rata-rata pada FSM sebesar 15,93.

Tabel 4.3 Penggunaan Memori dan CPU Metode FuSM

No	Waktu	Memory	CPU
1.	00 : 15 : 93	73.648	13.61
2.	00 : 54 : 13	75.580	14.26
3.	01 : 14 : 16	75.636	14.53
4.	01 : 50 : 61	75.628	14.22
5.	02 : 16 : 62	75.592	13.67
6.	02 : 37 : 86	75.652	13.40
7.	03 : 22 : 36	75.656	14.37
8.	03 : 53 : 55	75.664	16.17
9.	04 : 58 : 17	75.812	14.22
10.	06 : 02 : 70	75.844	13.13
11.	06 : 29 : 02	75.844	14.29
12.	06 : 52 : 06	75.836	16.44
13.	07 : 31 : 93	75.992	17.63
14.	08 : 54 : 46	75.996	16.26
15.	09 : 13 : 90	76.028	16.36

$$memori \text{ rata - rata FuSM} = \frac{\sum_{i=1}^n memori_i}{\sum pengukuran memori}$$

$$memori \text{ rata - rata FuSM} = \frac{1134408}{15}$$

$$memori \text{ rata - rata FuSM} = 75.627$$

Setelah dihitung rata-rata penggunaan memori, FuSM memakan memori rata-rata = 75.627.

$$CPU \text{ rata - rata } FuSM = \frac{\sum_{i=1}^n CPU_i}{\sum \text{pengukuran CPU}}$$

$$CPU \text{ rata - rata } FuSM = \frac{222,56}{15}$$

$$CPU \text{ rata - rata } FuSM = 14,83$$

Sedangkan penggunaan CPU rata-rata pada FSM sebesar 14,83. Dari data tersebut dapat disimpulkan bahwa FuSM lebih efisien dalam penggunaan memori dan CPU daripada FSM, sehingga FuSM lebih tepat digunakan sebagai kontrol respon perilaku NPC pada game.

4.4.3 Rekapitulasi Hasil Kuisisioner

Berikut ini adalah rekapitulasi hasil kuisisioner yang diisi oleh 24 responden dari kelas A2 Program Khusus Pembelajaran Bahasa Arab (PKPBA) UIN Maulana Malik Ibrahim Malang setelah memainkan *game* Alfatih.

Sitem uji coba ini yaitu semua mahasiswa memainkan *game* Alfatih sampai selesai, kemudian masing-masing mahasiswa mengisi kuisisioner yang telah disediakan. Hasil pengisian kuisisioner tersebut menjadi dasar layak atau tidak layak *game* Alfatih ini digunakan oleh semua orang.

Tabel 4.4 Rekapitulasi Hasil Kuisisioner

No	URAIAN	PENILAIAN	
		YA	TIDAK
1	Saya mengetahui fungsi setiap menu	83%	16,6%
2	Saya memahami menu petunjuk permainan	87,5%	12,5%
3	Saya mengetahui cara memilih level	75%	25%
4	Saya mengetahui cara bermain	100%	0%
5	Saya mengetahui cara mendapatkan bantuan	91,6%	8,3%
6	Saya bosan dengan game ini	12,5%	87,5%
7	Game Al Fatih menarik dan menyenangkan	87,5%	12,5%
8	Saya merasa lebih mudah belajar bahasa Arab dengan game Al fatih	91,6%	8,3%
9	Saya ingin memainkan game Al Fatih lagi	91,6%	8,3%

Dari data hasil pengujian *game* di atas dapat disimpulkan secara umum bahwa *game* Alfatih ini mudah dimainkan dengan prosentase responden mengetahui fungsi setiap menu sebesar 83%, memahami petunjuk permainan sebesar 87,5%, mengetahui cara memilih level sebesar 75% dan mengetahui cara bermain sebesar 100%. Game Al Fatih juga dapat disimpulkan sebagai game yang menarik dan dapat meningkatkan kualitas belajar bahasa Arab dengan prosentase responden tidak bosan memainkan game Al Fatih sebesar 87,5%, game Al Fatih menarik dan menyenangkan 87,5%, responden merasa lebih mudah belajar bahasa Arab menggunakan game Al Fatih sebesar 91,6% dan responden

ingin memainkan game Al Fatih lagi sebesar 91,6%. Hasil rekapitulasi kuisisioner ini secara keseluruhan menunjukkan bahwa game ini berhasil dan berjalan dengan baik.



BAB V

PENUTUP

5.1 Kesimpulan

Hasil penelitian yang telah dilakukan menyimpulkan bahwa metode *Fuzzy State Machine* (FuSM) dapat mendesain perilaku NPC kalajengking dan NPC pemburu kalajengking pada *game* Al Fatih dengan hasil yang lebih baik daripada penggunaan metode *Finite State Machine* (FSM), hal ini dibuktikan oleh hasil uji coba tabel 4.1 pada bab IV. Sedangkan untuk penggunaan memori dan CPU, FuSM lebih efisien dan efektif daripada FSM, hal ini dibuktikan dengan hasil uji coba tabel 4.2 dan 4.3 serta perhitungan rata-rata yang menunjukkan bahwa FuSM menggunakan memori dan CPU lebih rendah daripada FSM.

Hasil pengujian *game* pada responden menunjukkan bahwa *game* Alfatih menarik dan menyenangkan dengan prosentase sebanyak 87,5%. Dari aspek edukasi atau pendidikan *game* Alfatih bermanfaat dalam pembelajaran bahasa Arab dengan prosentase sebanyak 91,6% dari responden yang menyatakan bahwa belajar bahasa Arab menggunakan *game* ini lebih mudah.

5.2 Saran

Masih banyak kekurangan dalam *game* ini. Adanya kekurangan yang ada dapat dijadikan sebagai bahan pengembangan *game* selanjutnya. Oleh karena itu penulis menyarankan beberapa hal untuk bahan pengembangan selanjutnya, diantaranya:

1. Mengembangkan aplikasi *game* ini dengan tampilan yang lebih menarik dengan perilaku NPC yang lebih variatif.
2. Mengembangkan *game* ini untuk sistem berbasis mobile dengan sistem operasi terbaru, seperti Android, iOS, Windows Phone , Blackberry atau yang lainnya.



DAFTAR PUSTAKA

- Arif, Yunifa Miftachul. dkk. 2012. *Pergantian Senjata NPC pada Game FPS Menggunakan Fuzzy Sugeno*. Jurnal Unipdu. <http://www.journal.unipdu.ac.id/index.php/seminas/article/download/214/164> diakses tanggal 1 Juni 2013.
- Baba, Nario dkk. 2007. *Advanced Intelligent Paradigms in Computer Games*. New York:Springer-Verlag.
- Barmawi, Mohammad. 2011. *Lulus TOAFL dengan Mudah dan Memuaskan*. Yogyakarta: DIVA Press.
- Craig, W. Reynolds. *Steering Behaviors For Autonomous Characters*. Sony Computer Entertainment America.
- Cruz, Adriano. 2008. *Fuzzy State Machine*. NCE/UFRJ. <http://equipe.nce.ufrj.br/adriano/fuzzy/transparencias/fsm/fsm.pdf> diakses tanggal 10 Nopember 2012.
- Dillon, Teresa. 2004. *Adventure Games for Learning and Storytelling*. Futurelab <http://archive.futurelab.org.uk> diakses 30 April 2013.
- Fox, David dan Roman Verhosek. 2002. *Micro Java™ Game Development*. Indianapolis: Pearson Education.
- Henry, Samuel. 2005. *Panduan Praktis Membuat Game 3D*. Yogyakarta: Graha Ilmu.
- Jazairi, A.B.J. 2007. *Tafsir Al Qur'an. Al-Aisar (Jilid 3)*. Jakarta: Darus Sunnah.
- Kiswanto, Bambang. 2009. *Efektifitas Pembelajaran Bahasa Arab Dengan Kitab Ta'limul Lughotil 'Arobiyyah Di SMP Muhammadiyah 2 Mlati, Mlati, Sleman 2008/2009*. Skripsi Tidak Diterbitkan. Surakarta: UMS Surakarta.
- Kusumadewi, Sri. 2002. *Analisis dan Desain Sistem Fuzzy Menggunakan Tool Box Matlab*. Yogyakarta: Graha Ilmu.
- Lee, K.H. 2005. *First Course on Fuzzy Theory and Applications*. (J. Kacprzyk, Penyunt.) Taejon, Republic of South Korea: Springer.
- Machmudah, Umi dan Rosyidi, Abdul Wahab. 2008. *Active Learning dalam Pembelajaran Bahasa Arab*. Malang: UIN-Malang Press.

- Mahalli, Imam Jalaludin dan Imam Jalaludin As Suyuti. 2008. *Tafsir Jalalain Jilid 2*. Bandung : Sinar Baru Algesindo.
- Millington, Ian. 2006. *Artificial Intelligence for Games*. San Francisco, U.S.A.: Morgan Kaufmann Publishers.
- Muhibbin, Syah. 2004. *Psikologi Belajar*. Jakarta: Raja Grafindo Persada.
- Mustafa Al Ghalayai. 1980. *Jami'ud Durus Al-Arabiyah*. Bairut: Al-Maktabah Al-'Ashriyyah.
- Nasol, Rahma Rizania. 2013. *Fuzzy State Machine Sebagai Pengontrol Player Dalam Game Assyuhada*. Skripsi Tidak Diterbitkan. Malang : UIN.
- Neumann, J. Von and O. Morgenstern. 1953. *Theory of Games and Economic Behavior (3d ed.)* Princeton University Press. <http://tutoriaalkuliah.onsugar.com/Dasar-dasar-Teori-PermainanGame-Pengertian-dan-Macam-macam-Teori-Animasi-13748681> diakses tanggal 23 Mei 2013.
- Nofiantoro, Arif. 2011. *Analisis dan Perancangan Game "Bermain Bersama Dito & Dola"*. Skripsi Tidak Diterbitkan. Yogyakarta: AMIKOM.
- Nugroho, Supeno Mardi Susiki. dkk. 2011. *Perilaku Taktis Untuk Non-Player Characters Di Game Peperangan Meniru Strategi Manusia Menggunakan Fuzzy Logic Dan Hierarchical Finite State Machine*. Surabaya: Jurnal Ilmiah Cursor Vol. 6, No. 1, Januari 2011.
- Rohman, Moh Miftakhur. 2011. *Finite State Machine Sebagai Agen Cerdas Animasi Wajah Pada Game Assyuhada*. Skripsi Tidak Diterbitkan. Malang : UIN.
- Salen, Katie and Eric Zimmerman. 2004. *Rules of Play: Game Design Fundamentals*. Cambridge (MA): The MIT Press.
- Shihab, M. Quraish. 2002. *Tafsir Al-Mishbah Vol. 12*. Jakarta : Lentera Hati.
- Shinta. 2012. *Logika Himpunan Fuzzy*. <http://www.mdp.ac.id/materi/2011-2012-1/TI421/041033/TI421-041033-510-1.ppt> diakses 17 Mei 2013.
- Sweetser, Penelope & Wiles, Janet. 2002. *Current AI in games : a review*. *Australian Journal of Intelligent Information Processing Systems*, 8(1), pp. 24-42. http://eprints.qut.edu.au/45741/1/AJIIPS_paper.pdf diakses 17 Oktober 2012.

Zadeh,L.A.1965. *Fuzzy Sets.Informationand Control*,338-353.

<http://www.arabion.net/indonesian/> diakses 13 mei 2013

<http://en.wikipedia.org/wiki/Gamestudio> diakses 27 April 2013



LAMPIRAN

1. Tabel Rekapitulasi Hasil Kuisisioner

Tabel 1. Rekapitulasi Hasil Kuisisioner

No	URAIAN	JUMLAH	
		YA	TIDAK
1	Saya mengetahui fungsi setiap menu	20	4
2	Saya memahami menu petunjuk permainan	21	3
3	Saya mengetahui cara memilih level	18	16
4	Saya mengetahui cara bermain	24	0
5	Saya mengetahui cara mendapatkan bantuan	22	2
6	Saya bosan dengan game ini	21	3
7	Game Al Fatih menarik dan menyenangkan	21	3
8	Saya merasa lebih mudah belajar bahasa Arab dengan game Al fatih	22	2
9	Saya ingin memainkan game Al Fatih lagi	22	2

2. Gambar Pengujian Kelayakan Game Al Fatih di PKPBA UIN Malang



Gambar 1. Uji Coba Kelayakan Game Al Fatih di PKPBA



مركز اللغة العربية – جامعة مولانا مالك إبراهيم الإسلامية الحكومية بمالانق

اختبار الكفاية اللغوية

القسم الأول :

■ اختر الجواب الصحيح.

- 1 - محمدٌ على الكرسي أ- يجلس ب- يجلس ج- جلست
- 2 - ذهب علي إلى أ- المدرسة ب- المسجد ج- المكتبة
- 3 - عمر في قرية صغيرة. أ- أسكن ب- يسكن ج- تسكن
- 4 - في المدينة كثيرة. أ- شركات ب- الشركة ج- شركة
- 5 - سارة تساعد في البيت. أ- والدته ب- والده ج- والدتها
- 6 - ابني حارث. أ- اسمه ب- اسمك ج- اسمها
- 7 - جميل يعمل في الجامعة. أ- أستاذ ب- الأستاذ ج- أستاذاً
- 8 - زينب ستكون إن شاء الله. أ- ممرض ب- ممرضة ج- ممرضتان
- 9 - عمر له ابنة اسمها هبة. أ- واحد ب- واحدة ج- الواحدة

■ املا الفراغ بالكلمة الصحيحة.

- 10 - رأيت سيارات من المكتبة أ- أربع ب- أربعة ج- رابع
- 11 - أريد الثوب أ- أبيض ب- أبيض ج- الأبيض
- 12 - عدداً إن شاء الله أ- سأسافر ب- سافرت ج- سفر
- 13 - قرأت ثلاثة أ- كتاب ب- كتاب ج- كتاب
- 14 - يا محمد القهوة أ- تشرب ب- أ تشرب ج- اشرب
- 15 - كم ستبقى هنا؟ أ- يوم ب- يوماً ج- أياماً
- 16 - أسكن في بيت أ- الجميل ب- جميلة ج- جميل
- 17 - قابلت أ- طبيب الأسنان ب- الطبيب الأسنان ج- الطبيب أسنان
- 18 - أريد الإسلامية أ- الكتب ب- كُتُب ج- الكتاب

■ اكتبِ الفعلَ الصحيحَ.

- 19 -الطفل أ-
يَبْكِي ب- تَبْكِي ج- بَكَتْ
- 20 -صديقي لا اللحمَ أ-
تَحِبُّ ب- يَحِبُّ ج- يَحَبُّ
- 21 -فاطمة في البيت أ-
تَصَلِي ب- يَصَلِي ج- يَصَلِّي
- 22 -ماذا يا أحمد ؟ أ-
سَتَذْهَبُ ب- سَتَنَامُ ج- سَتَأْكُلُ
- 23 -.....أخي إلى مَكَّةَ قَبْلَ أُسْبُوعٍ. أ- يُسَافِرُ ب- سَيُسَافِرُ ج- سَافَرَ
- 24 -يا بُيَّيْ هذا الدَّوَاءُ. أ- اشْرَبْ ب- شَرِبْ ج- يَشْرَبْ
- 25 -لا اليومَ يا أخي. أ- ذَهَبْ ب- تَذْهَبْ ج- يَذْهَبُ
- 26 -يا مُحَمَّدُ في العُرْفَةِ. أ- نَمَّ ب- نَامَ ج- يَنَامُ
- 27 -تَخْرُجُ الطَّالِبُ وَمَ عَمَلًا مُنَاسِبًا. أ- وَجَدَ ب- يَجِدُ ج- أَجَدُ
- 28 -يَحِبُّ أَنْ الرِّيَاضَةَ كُلَّ يَوْمٍ. أ- تُمَارِسُ ب- تُمَارِسُ ج- تُمَارِسَ
- 29 -يا أَبْنَائِي لَنْ غَدًا إِلَى الحَفْلِ. أ- تَذْهَبَا ب- تَذْهَبُونَ ج- تَذْهَبُوا
- 30 -الطَّالِبَانِ لَمْ إِلَى المَدْرَسَةِ اليَوْمَ أ- يَحْضُرَا ب- يَحْضُرَانِ ج- يَحْضُرُوا
- 31 -الطُّلَابُ اللُّغَةَ العَرَبِيَّةَ فِي الجَامِعَةِ. أ- يَدْرُسُوا ب- يَدْرُسُونَ ج- يَدْرُسْنَ
- 32 -لا الطَّعَامَ الحَارَّ يا مُحَمَّدُ. أ- تَأْكُلُ ب- تَأْكُلَا ج- تَأْكُلُ
- 33 -.....أخي مِنْ مَكَّةَ قَبْلَ يَوْمٍ. أ- حَضَرَ ب- يَحْضُرُ ج- سَيَحْضُرُ
- 34 -يا طَالِبِيَّةُ هَذَا الكِتَابِ. أ- تَنَاوَلْ ب- تَنَاوَلْتِ ج- تَنَاوَلْتِ
- 35 -لا اليَوْمَ يا سَعِيدُ. أ- تَذْهَبْ ب- تَذْهَبِ ج- تَذْهَبُ
- 36 -تَسْتَطِيعُ أَنْ رَسَائِلَكَ بِالرِّبْرِيدِ الإِلِكْتُرُونِيِّ. أ- تُرْسِلْ ب- تُرْسِلِ ج- تُرْسِلِنِ
- 37 -البِنْتُ كَثِيرًا. أ- يُفَكِّرُ ب- تُفَكِّرَانِ ج- تُفَكِّرُ
- 38 -المُهَنْدِسَانِ أ- حَضَرَا ب- حَضَرُوا ج- يَحْضُرُونَ
- 39 -نَحْنُ مِنْ هَذِهِ الدَّرُوسِ قَبْلَ ثَلَاثَةِ أَيَّامٍ. أ- انْتَهَى ب- انْتَهَيْنَا ج- انْتَهَوْا
- 40 -النَّاسُ عَنِ العَوْلَمَةِ. أ- تَحَدَّثَ ب- يَتَحَدَّثُ ج- يَتَحَدَّثُونَ
- 41 -يا فَاطِمَةُ هَذَا الكِتَابِ. أ- خُذِي ب- خُذْ ج- تَأْخُذُ
- 42 -الأبْنَاءُ لَمْ مِنَ المَدْرَسَةِ. أ- يَحْضُرُ ب- يَحْضُرَا ج- يَحْضُرُوا

أولي

■ اقرأ الفقرة، ثم اختر الجواب الصحيح.

- تُفَكِّرُ فاطمة في حلِّ مشكلتها، وزُها الآن سبعون كيلا ويجبُ أن يكون وزُها خمسين كيلا فقط. تركتُ فاطمة السكريات والنشويات، لكن للأسف ظلَّ وزُها نحوَ سبعين كيلا
- 43 -مشكلة فاطمة أنها أ- نحيفة ب- طويلة ج- سمينة
- 44 -الوزن المناسب لفاطمة كيلا أ- سبعون ب- ستون ج- خمسون
- 45 -..... بعد أن تركت فاطمة السكريات أ- ظلَّ وزُها ب- نقص وزُها ج- انخفض وزُها
- لدي مشكلة، أريدُ أن أختار مهنتي وحدي. أعتقدُ أن التعليم مهنةٌ طيبة. أبي يُحِبُّ أن أكون مهندسا مثله، أو طيارا مثل أخي حسن، لكن أُمي تريدُ أن أكون طبيبا مثل جدي.
- 46 -مهنة الوالد هي أ- مهندس ب- مدرس ج- طيار
- 47 -حسن يعمل في مجال أ- التعليم ب- الطيران ج- الطب
- 48 -يريد المتحدث أن يكون أ- مهندسا ب- مدرسا ج- طيارا
- هاجرَ جَعْفَرُ - رضي الله عنه - إلى الحبشة، وعندما قابَلَ مَلِكها قَرَأَ عَلَيْهِ جُزءاً مِنْ سُورَةِ مَرْيَمَ، بَكَى النَّجاشِيُّ مَلِكُ الحبشة، وَبَكَى مَنْ كَانَ عِنْدَهُ فِي الْقَصْرِ مِنَ الْوِزراءِ وَالْأَساقِفَةِ، عِنْدَمَا سَمِعُوا الْقُرْآنَ. ثُمَّ قَالَ النَّجاشِيُّ لَجَعْفَرٍ: "إِنَّ هَذَا وَالَّذِي جَاءَ بِهِ عيسى، لَيُخْرِجُ مِنْ مِشْكَاةٍ وَاحِدَةٍ"، وَدَخَلَ فِي الْإِسْلَامِ.
- 49 -بَكَى النَّجاشِيُّ ... أ- خوفاً على مُلْكِهِ ب- لَأَنَّ مَنْ مَعَهُ كُتْلُهُمْ بَكَوا ج- لِأَنَّهُ تَأَثَّرَ بِالْقُرْآنِ
- 50 -في الْفِقْرَةِ إِشارةٌ إلى رِسالَةِ نَبِيِّنِ كَرِيمَيْنِ، هُمَا...
- أ- مُحَمَّدٌ وَعيسى ب- مُحَمَّدٌ وَموسى ج- عيسى وَموسى
- 51 -حَدَّثَ ذَلِكَ فِي ... أ- الْيَمَنِ ب- مَكَّةَ ج- الْحَبشةَ
- 52 -كَانَ النَّجاشِيُّ ... أ- نَبِيًّا ب- مَلِكاً ج- أميراً
- 53 -أَسْلَمَ النَّجاشِيُّ لِأَنَّهُ ... أ- يَعْرِفُ جَعْفَرًا ب- عَرَفَ الْحَقِيقَةَ ج- خَافَ على مُلْكِهِ
- اِقْرَأِ الْفِقْرَةَ، ثُمَّ اخْتَرِ الْجَوَابَ الصَّحِيحَ.
- في الْوَقْتِ الْحاضِرِ، تَشُقُّ اللُّغَةُ الْعَرَبِيَّةُ طَرِيقَها بِكُلِّ قُوَّةٍ وَثَباتٍ؛ لِكَيْ تَسْتَعِيدَ دَوْرَها التَّاريخيَّ الْعَظِيمَ، الَّذِي أَدَّتْهُ مِنْذُ مُنتَصَفِ الْقَرْنِ السَّابِعِ المِيلادي، وَحَتَّى نِهايةِ الْقَرْنِ الْحادي عَشَرَ مِنْهُ؛ عِنْدَمَا أَصْبَحَتْ لُغَةُ الْعِلْمِ، وَالثَّقافةِ، وَالْفِكْرِ، وَالاتِّصالاتِ الدَّولِيَّةِ الْوَحِيدَةَ فِي الْعالِمِ الْقَدِيمِ؛ أَيُّ أَمَّا الْآنَ فِي طَرِيقِها، لِأَنَّ تَصْبِيحَ مِنْ جَدِيدٍ لُغَةً عَالَمِيَّةً مِثْلَ اللُّغَاتِ الْعَالَمِيَّةِ الْمُعاصِرَةِ. وَلَعَلَّ مِنْ أَمَمِ الْعَوامِلِ الَّتِي سَاعَدَتْ فِي الْماضِي، وَتُساعِدُ فِي الْحاضِرِ وَالْمُسْتَقْبَلِ، على جَعْلِ اللُّغَةِ الْعَرَبِيَّةِ لُغَةً ذاتِ مَكَانَةٍ خَاصَّةٍ عِنْدَ الْمُسْلِمِينَ، أَمَّا لُغَةُ الْقُرْآنِ الْكَرِيمِ، وَالْقُرْآنِ كِتَابِ الْمُسْلِمِينَ؛ يَقْرؤُهُ أبنائُهُمْ مِنْذُ الصَّعْرِ، وَيَحْفَظونَهُ كُلُّهُ أَوْ جُلَّةَ (مُعْظَمَهُ). وَالْعَرَبِيَّةُ هِيَ الَّتِي تُؤَدِّي بِها الصَّلواتِ الْحَمْسُ كُلَّ يَوْمٍ وَلَيْلَةٍ.
- 54 -في الْقَرْنِ الْحادي عَشَرَ المِيلادي كَانَتْ اللُّغَةُ الْعَرَبِيَّةُ...

أ- لُغَةٌ دِينِيَّةٌ ب- اللُّغَةُ الْوَحِيدَةُ فِي الْعَالَمِ ج- لُغَةٌ عَالَمِيَّةٌ

55 - اِكْتَسَبَتِ اللُّغَةُ الْعَرَبِيَّةُ اَهْمِيَّتَهَا فِي نَفُوسِ الْمُسْلِمِيْنَ، لِاَنَّهَا...

أ- اللُّغَةُ الَّتِي نَزَلَ بِهَا الْقُرْآنُ ب- لُغَةُ الْعَرَبِ وَالْمُسْلِمِيْنَ ج- حَافِظَتْ عَلَى الْقُرْآنِ الْكَرِيمِ

56 - ظَلَّتِ اللُّغَةُ الْعَرَبِيَّةُ لُغَةَ الْعِلْمِ وَالثَّقَافَةِ مُدَّةً...

أ- 400 سَنَةٍ ب- 450 سَنَةً ج- 420 سَنَةً

57 - مِمَّا فَهَمَّتْ مِنْ النَّصِّ ؛ يُقْبَلُ الْمُسْلِمُونَ عَلَى قِرَاءَةِ الْعَرَبِيَّةِ وَهُمْ ...

أ- شَبَابٌ ب- صِغَارٌ ج- كِبَاؤُ السِّنِّ

القسم الثاني:

■ اخترِ الجوابَ الصحيحَ.

58 - الفاعل يكون أ- منصوبا ب- مرفوعا ج- مجرورا

59 - المبتدأ والخبر كلاهما أ- منصوب ب- مرفوع ج- مجرور

60 - المفعول به يكون أ- دائما ب- مرفوعا ج- مجرورا

61 - نائب الفاعل أيضا أ- منصوبا ب- مرفوعا ج- مجرورا

62 - تُنْصَبُ الْأَسْمَاءُ الْخَمْسَةُ... أ- بِالْفَتْحَةِ ب- بِالْأَلِفِ ج- بِالْيَاءِ

63 - يُرْفَعُ الْمُثَنَّى... أ- بِالْأَلِفِ ب- بِالضَّمَّةِ ج- بِالْوَاوِ

64 - يُرْفَعُ جَمْعُ الْمُدَكَّرِ السَّلَامِ... أ- بِالضَّمَّةِ ب- بِالْأَلِفِ ج- بِالْوَاوِ

65 - تُنْصَبُ الْأَفْعَالُ الْخَمْسَةُ... أ- بِحَذْفِ التَّوْنِ ب- بِالْفَتْحَةِ ج- بِالْأَلِفِ

66 - يُجْرَمُ الْفِعْلُ الْمَضَارِعُ الْمَعْتَلُ الْآخِر... أ- بِالسُّكُونِ ب- بِحَذْفِ التَّوْنِ ج- بِحَذْفِ حَرْفِ الْعِلَّةِ

67 - يُنْصَبُ الْفِعْلُ الْمَضَارِعُ إِذَا سَبَقَ بِ... أ- أَنْ وَإِنَّ ب- لَمْ وَلَا ج- أَنْ وَلَنْ

68 - الْحَالُ اسْمٌ... أ- مَعْرِفَةٌ مَنْصُوبٌ ب- نَكْرَةٌ مَجْرُورٌ ج- نَكْرَةٌ مَنْصُوبٌ

69 - تَمَيِّزُ الْأَعْدَادِ مِنْ 11 إِلَى 19 يَكُونُ... أ- مُفْرَدًا مَنْصُوبًا ب- مُفْرَدًا مَجْرُورًا ج- جَمْعًا مَجْرُورًا

70 - "إِنَّ" و "أَنَّ" حَرْفَا تَوْكِيدٍ...

أ- يَرْفَعَانِ الْمُبْتَدَأَ وَيَنْصَبَانِ الْحَبَرَ ب- يَنْصَبَانِ الْمُبْتَدَأَ وَيَرْفَعَانِ الْحَبَرَ ج- يَنْصَبَانِ الْمُبْتَدَأَ وَالْحَبَرَ

71 - يُنْصَبُ جَمْعُ الْمُدَكَّرِ السَّلَامِ بِ... أ- الْفَتْحَةِ ب- الْأَلِفِ ج- الْيَاءِ

72 - تُجْرَى الْأَسْمَاءُ الْخَمْسَةُ... أ- بِالْفَتْحَةِ ب- بِالْأَلِفِ ج- بِالْيَاءِ

73 - يُنْصَبُ الْمُثَنَّى... أ- بِالْأَلِفِ ب- بِالْيَاءِ ج- بِالْفَتْحَةِ

74 - تُجْرَمُ الْأَفْعَالُ الْخَمْسَةُ... أ- بِحَذْفِ التَّوْنِ ب- بِالسُّكُونِ ج- بِحَذْفِ حَرْفِ الْعِلَّةِ

75 - يُرْفَعُ جَمْعُ الْمُؤنَّثِ السَّلَامِ... أ- بِالْوَاوِ ب- بِالضَّمَّةِ ج- بِالْكَسْرِ

- 76 - الحَالُ اسْمٌ نَكْرَةٌ أ- مَنْصُوبٌ ب- مَجْرُورٌ ج- مَرْفُوعٌ
 77 - يُجْرَى جَمْعُ الْمَذْكَرِ السَّلَامِ أ- بِالْكَسْرِ ب- بِالْيَاءِ ج- بِحَذْفِ النُّونِ
 78 - الْفِعْلُ الْمُضَارِعُ إِذَا سَبِقَ بِأَنْ أَوْ لَنْ أ- يُجْزَمُ ب- يُرْفَعُ ج- يُنْصَبُ

■ اختر الإجابة الصحيحة، المناسبة لما تحته خط في كل آية.

- 79 - هذه مَيْسَرَةٌ وهي طيبة أ- اسم الآلة ب- اسم الموصول ج- اسم الإشارة
 80 - مكتبة جامعة أم القرى كبيرة وواسعة أ- مبتدأ ب- خبر ج- فاعل
 81 - احفظ بعض الأحاديث النبوية أ- صفة ب- بدل ج- حال
 82 - قَالَ تَعَالَى : (سبحان الذي أسرى بعبده ليلاً).
 كَلِمَةُ (لَيْلًا) ... أ- تَمَيِّزُ ب- حَالٌ ج- ظَرْفٌ
 83 - قَالَ تَعَالَى : (قال رب إني وهن العظم مني واشتعل الرأس شيباً).
 كَلِمَةُ (شَيْبًا) ... أ- تَمَيِّزُ ب- حَالٌ ج- صِفَةٌ
 84 - قَالَ تَعَالَى : (لو كان فيهما آلهة إلا الله لفسدنا).
 كَلِمَةُ (آلهة) ... أ- خَبَرٌ كَانَ ب- مُبْتَدَأٌ مُؤَخَّرٌ ج- اسْمٌ كَانَ
 85 - قَالَ تَعَالَى : (إن الإنسان خلق هلوعاً).
 كَلِمَةُ (هلوعاً) ... أ- مَفْعُولٌ بِهِ ب- حَالٌ ج- اسْمٌ إِنَّ
 86 - قَالَ تَعَالَى : (من المؤمنين رجال صدقوا الله).
 كَلِمَةُ (رجال) ... أ- خَبَرٌ ب- نَائِبُ فَاعِلٍ ج- مُبْتَدَأٌ
 87 - قَالَ تَعَالَى : (يقول الإنسان يومئذ أين المفر).
 كَلِمَةُ (أَيْنَ) ... أ- حَرْفٌ اسْتِفْهَامٍ ب- مُبْتَدَأٌ مُؤَخَّرٌ ج- خَبَرٌ مُقَدَّمٌ
 88 - قَالَ تَعَالَى : (إحم يكيدون كيداً).
 كَلِمَةُ (كَيْدًا) ... أ- مَفْعُولٌ مُطْلَقٌ ب- تَمَيِّزٌ ج- حَالٌ
 89 - قَالَ تَعَالَى : (فسبح بحمد ربك واستغفره إنه كان تواباً).
 كَلِمَةُ (تواباً) ... أ- اسْمٌ إِنَّ مُؤَخَّرٌ ب- خَبَرٌ كَانَ ج- حَالٌ
 90 - قَالَ تَعَالَى : (قُتِلَ أَصْحَابُ الْأَحْدُودِ).
 كَلِمَةُ (أَصْحَابُ) ... أ- مُبْتَدَأٌ مُؤَخَّرٌ ب- فَاعِلٌ ج- نَائِبُ فَاعِلٍ
 91 - قَالَ تَعَالَى : (إنا أعطيناك الكوثر).
 كَلِمَةُ (الكوثر) ... أ- مَفْعُولٌ بِهِ ثَانٍ ب- مَفْعُولٌ بِهِ أَوَّلٌ ج- خَبَرٌ إِنَّ

92 - قَالَ تَعَالَى : (شهر رمضان الذي أنزل فيه القرآن).

كَلِمَةٌ (رَمَضَانَ) مُضَافٌ إِلَيْهِ...

أ- مَجْرُورٌ بِالْكَسْرِ نِيَابَةً عَنِ الْفَتْحِ ب- مَجْرُورٌ بِالْكَسْرِ ج- مَجْرُورٌ بِالْفَتْحِ نِيَابَةً عَنِ الْكَسْرِ

93 - قَالَ تَعَالَى : (يا أيها المزمّل * قم الليل إلا قليلاً).

كَلِمَةٌ (قَلِيلًا) ... أ- مُسْتَثْنَى مُنْصُوبٌ ب- حَالٌ مُنْصُوبٌ ج- تَمَيِّزٌ مُنْصُوبٌ

94 - قَالَ تَعَالَى : (وَمَنْ يَعْمَلْ مِثْقَالَ ذَرَّةٍ شَرًّا يَرَهُ).

كَلِمَةٌ (مَنْ) ... أ- اسْمٌ اسْتِفْهَامٍ ب- أَدَاةٌ شَرْطٌ جَارِمَةٌ ج- حَرْفٌ حَرٌّ

95 - قَالَ تَعَالَى : (لا خَيْرَ فِي كَثِيرٍ مِنْ نَجْوَاهُمْ).

كَلِمَةٌ (لا) ... أ- حَرْفٌ نَفْيٍ ب- لا النَّافِيَةُ لِلْجِنْسِ ج- لا النَّاهِيَةُ

96 - قَالَ تَعَالَى : (إِذْ قَالَ لَهُمْ أَخُوهُمْ نُوحٌ أَلَا تَتَّقُونَ).

كَلِمَةٌ (نُوحٌ) ... أ- بَدَلٌ ب- صِفَةٌ ج- تَوْكِيدٌ

97 - قَالَ تَعَالَى : (قل إن الأمر كله لله).

كَلِمَةٌ (كُلُّهُ) ... أ- صِفَةٌ ب- بَدَلٌ ج- تَوْكِيدٌ

98 - قَالَ تَعَالَى : (ومن آياته يريكم البرق خوفاً وطمعاً).

كَلِمَةٌ (خَوْفًا) ... أ- صِفَةٌ ب- تَمَيِّزٌ ج- مَفْعُولٌ لِأَجْلِهِ

99 - قَالَ تَعَالَى : (أَنَا أَكْثَرُ مِنْكَ مَالًا وَوَلَدًا).

كَلِمَةٌ (أَكْثَرُ) ... أ- اسْمٌ فَاعِلٍ ب- اسْمٌ تَفْضِيلٍ ج- اسْمٌ مَفْعُولٍ

■ اخْتَرِ الْعِبَارَةَ الصَّحِيحَةَ الَّتِي تُكْمِلُ الْمَعْنَى.

100 - المفعول به اسم منصوب وقع عليه فعل
أ- الفاعل ب- الخبر ج- المبتدأ

101 - ينصب الاسم المفرد
أ- بثبوت النون ب- بحذف النون ج- بالحركة

102 - تُنَوِّنُ الْوَقَايَةَ وَاجِبَةٌ مَعَ ...
أ- الأفعالِ وَمِنْ وَعَنْ ب- إِنَّ وَأَخَوَاتِهَا ج- كُلِّ حُرُوفِ الْجُرِّ

103 - يُصَاغُ اسْمُ الْفَاعِلِ مِنْ غَيْرِ الثَّلَاثِيَّ بِإِبْدَالِ حَرْفِ الْمُضَارَعَةِ ...
أ- مِيمًا مَكْسُورَةً ب- مِيمًا مَفْتُوحَةً ج- مِيمًا مَضْمُومَةً

104 - يُبْنَى الْفِعْلُ الْمَاضِي لِلْمَجْهُولِ ...
أ- بِضَمِّ أَوَّلِهِ وَفَتْحِ مَا قَبْلَ آخِرِهِ ب- بِضَمِّ أَوَّلِهِ وَكَسْرِ مَا قَبْلَ آخِرِهِ ج- بِفَتْحِ أَوَّلِهِ وَضَمِّ مَا قَبْلَ آخِرِهِ

105 - يُجْرُ الْمُنْعُوعُ مِنَ الصَّرْفِ بِالْفَتْحَةِ بَدَلًا عَنِ الْكُسْرَةِ، إِذَا...

أ- لَمْ يَكُنْ مُضَافًا أَوْ مُحَلَّى بِأَلْ - ب- كَانَ مُضَافًا - ج- كَانَ عَلَى وَزْنِ أَعْمَالٍ

القسم الثالث :

■ أَقْرَأِ الْفَقْرَةَ، ثُمَّ اخْتَرِ الْجَوَابَ الصَّحِيحَ.

يَسْتَيْقِظُ طَاهِرٌ عِنْدَ الْفَجْرِ، وَيَتَوَضَّؤُ فِي الْبَيْتِ، ثُمَّ يَذْهَبُ إِلَى الْمَسْجِدِ. يَصَلِّي طَاهِرُ الْفَجْرِ فِي الْمَسْجِدِ. طَاهِرٌ يَقْرَأُ الْقُرْآنَ، وَلَا يَنَامُ بَعْدَ الصَّلَاةِ.

106 - متى يذهب طاهر الى المسجد؟ أ- بعد الفجر ب- عند الصلاة ج- عند الفجر

107 - ماذا يفعل طاهر بعد الصلاة؟ أ- الصلاة النافلة ب- قراءة القرآن ج- قراءة الحديث

من أهمّ المشكلات التي يواجهها سكان الريف في الحاضر، أنّ فرص العمل فيه قليلة، يعمل معظمهم في الزراعة، أو الرعي. ولأنّ فرص العمل في المدن أكثر، يهاجر الشباب إليها، ويتركون أعمال الزراعة للشيوخ والنساء وأعمال الرعي للأطفال.

108 - فرص العمل للشباب في الريف

أ- طيبة ب- كثيرة ج- قليلة

109 - ما الأعمال التي يقوم بها الأطفال في الريف ؟

أ- الدراسة ب- الرعي ج- التسوّل

110 - إذا هاجر الشباب إلى المدن ، فمن يعمل الزراعة ؟

أ- النساء و الشيوخ ب- الأطفال و النساء ج- الشيوخ والأطفال

يذهب أحمد إلى السوق مع أسرته في يوم العطلة، يشتري أحمد الملابس بعشرين ديناراً في قسم القمصان، ويشتري الحذاء الأبيض بثلاثين ديناراً، ثم يرجع أحمد وأسرته إلى البيت ساعدين.

111 - يشتري أحمد في قسم القمصان أ- القميص ب- الحذاء ج- الملابس

112 - كم المطلوب من الملابس والحذاء ؟ أ- خمسون ديناراً ب- أربعون ديناراً ج- ثلاثون ديناراً

كَانَ عَمْرُو بْنُ الْعَاصِ وَالْيَأْ عَلَى مِصْرَ، وَكَانَ ابْنُهُ يُجْرِي الْحَيْلَ فِي مَيْدَانِ السَّبَاقِ، فَتَارَعَهُ بَعْضُ الْمِصْرِيِّينَ السَّبَقَ، وَاخْتَلَفَا بَيْنَهُمَا لِمَنْ يَكُونُ الْفَرَسُ السَّابِقُ. وَعَضِبَ ابْنُ الْوَالِي، فَضَرَبَ الْمِصْرِيَّ وَهُوَ يَقُولُ: أَنَا ابْنُ الْأَكْرَمِينَ. فَاسْتَدْعَى عُمَرُ الْوَالِي وَابْنَهُ، حِينَ رَفَعَ إِلَيْهِ الْمِصْرِيُّ أَمْرَهُ. وَنَادَى بِالْمِصْرِيِّ فِي جَمْعٍ مِنَ النَّاسِ، أَنْ يَضْرِبَ خَصْمَهُ قَائِلًا لَهُ: اضْرِبِ ابْنَ الْأَكْرَمِينَ، ثُمَّ

أَمْرُهُ أَنْ يَضْرِبَ الْوَالِي، لِأَنَّ ابْنَهُ لَمْ يَجْرُؤْ عَلَى ضَرْبِ النَّاسِ إِلَّا بِسُلْطَانِهِ. وَصَاحَ بِالْوَالِي مُغْضَبًا: بِمِ اسْتَعْبَدْتُمْ النَّاسَ، وَقَدْ
وَلَدْتَهُمْ أُمَّهَاتُهُمْ أَحْرَارًا؟ فَمَا نَحَا مِنْ يَدِهِ إِلَّا يَرْضَى مِنْ صَاحِبِ الشُّكُوى وَاعْتِنَادٍ مَقْبُولٍ.

113 - كَانَ عَمْرُو بْنُ الْعَاصِ وَالِيًّا عَلَى... أ- الْمَدِينَةِ ب- مِصْرَ ج- الشَّامِ

114 - كَانَتْ الْمِبَارَاهُ فِي... أ- السَّبَاحَةِ ب- الْجَزْيِ ج- سِبَاقِ الْحَيْلِ

115 - اسْتَدْعَى عَمْرُو بْنُ مِصْرَ... أ- الْوَالِي ب- الْوَالِي وَابْنَهُ ج- ابْنَ الْوَالِي

116 - فِي قَوْلِهِ أَمْرُهُ أَنْ يَضْرِبَ الْوَالِي، الْوَالِي هُنَا هُوَ...

أ- عَمْرُو بْنُ الْوَالِي ب- عَمْرُو بْنُ الْعَاصِ ج- الْمِصْرِي

117 - فِي الْقِصَّةِ...

أ- ضَرَبَ عَمْرُو بْنُ الْعَاصِ ابْنَ الْمِصْرِي ب- ضَرَبَ ابْنَ عَمْرُو ابْنَ الْمِصْرِي ج- ضَرَبَ ابْنَ الْمِصْرِي ابْنَ عَمْرُو

■ أَقْرَأَ الْفَقْرَةَ، ثُمَّ اخْتَرِ الْجَوَابَ الصَّحِيحَ.

لَمَّا خَطَبَ عَمْرُو بْنُ حُجْرٍ الْكِنْدِيِّ، إِلَى عَوْفِ بْنِ مُحَمَّدِ الشَّيْبَانِيِّ ابْنَتَهُ أُمَّ إِيَّاسٍ، وَأَجَابَهُ إِلَى ذَلِكَ، أَقْبَلَتْ عَلَيْهَا أُمَّهَا لَيْلَةً
دُخُولِهِ بِهَا ثَوْبِيهَا. فَكَانَ مِمَّا أَوْصَتْهَا بِهِ أَنْ قَالَتْ: أَيُّ بَنِيَّةٍ، إِنَّكَ مُفَارِقَةٌ بَيْتِكَ الَّذِي مِنْهُ خَرَجْتَ، وَعُشَّكَ الَّذِي فِيهِ دَرَجَتْ
إِلَى رَجُلٍ لَمْ تَعْرِفِيهِ، وَقَرِينٍ لَمْ تَأْلَفِيهِ، فَكُونِي لَهُ أُمَّةً يَكُنْ لَكَ عَبْدًا؛ وَاحْفَظِي لَهُ خِصَالًا عَشْرًا يَكُنْ لَكَ دُخْرًا. فَأَمَّا الْأُولَى
وَالثَّانِيَةُ فَالرِّضَا وَالقَّنَاعَةُ، وَحَسَنُ السَّمْعِ لَهُ وَالطَّاعَةُ. وَأَمَّا الثَّلَاثَةُ وَالرَّابِعَةُ فَالتَّفَقُّدُ لِمَوَاقِعِ عَيْنِيهِ وَأَنْفِيهِ؛ فَلَا تَفْعُ عَيْنُهُ مِنْكَ عَلَى
قَبِيحٍ، وَلَا يَشْمُ أَنْفُهُ مِنْكَ إِلَّا أَطْيَبَ الرِّيحِ. وَأَمَّا الْخَامِسَةُ وَالسَّادِسَةُ فَالتَّفَقُّدُ لَوْقَتِ طَعَامِهِ وَمَنَامِهِ؛ فَإِنَّ شِدَّةَ الْجُوعِ مُلْهَبَةٌ
وَتَنْغِيصَ النَّوْمِ مَغْضَبَةٌ. وَأَمَّا السَّابِعَةُ وَالثَّامِنَةُ فَالْإِحْرَارُ لِمَالِهِ، وَالْإِرْعَاءُ عَلَى حَشْمِهِ وَعِيَالِهِ. وَأَمَّا التَّاسِعَةُ وَالْعَاشِرَةُ فَلَا تَعْصِي
لَهُ أَمْرًا، وَلَا تُفْشِي لَهُ سِرًّا، فَإِنَّكَ إِنْ خَالَفْتَ أَمْرَهُ أَوْغَرْتَ صَدْرَهُ، وَإِنْ أَفْشَيْتَ سِرَّهُ لَمْ تَأْمَنِ عَدْرَهُ. وَإِيَّاكَ ثُمَّ إِيَّاكَ وَالْفَرَحَ بَيْنَ
يَدَيْهِ إِذَا كَانَ مُهْتَمًّا، وَالكَآبَةَ لَدَيْهِ إِذَا كَانَ فَرِحًا. فَاقْبَلْتُ وَصِيَّةَ أُمَّهَا، فَأَجْبَبْتُ لَهُ الْحَارِثَ بْنَ عَمْرٍو جَدَّ امْرِئِ الْقَيْسِ الْمَلِكِ
الشَّاعِرِ.

118 - أُمُّ إِيَّاسٍ ... أ- ابْنَةُ عَمْرُو ب - ابْنَةُ عَوْفِ ج - زَوْجَةُ عَمْرُو

119 - أَوْصَتْ الْأُمَّ ابْنَتَهَا بِ... أ - 10 خِصَالٍ ب - 8 خِصَالٍ ج - 9 خِصَالٍ

120 - الْخِصْلَةُ السَّادِسَةُ ... أ - تَفَقَّدَ وَقْتِ طَعَامِهِ ب - تَفَقَّدَ وَقْتِ نَوْمِهِ ج - تَفَقَّدَ مَالَهُ