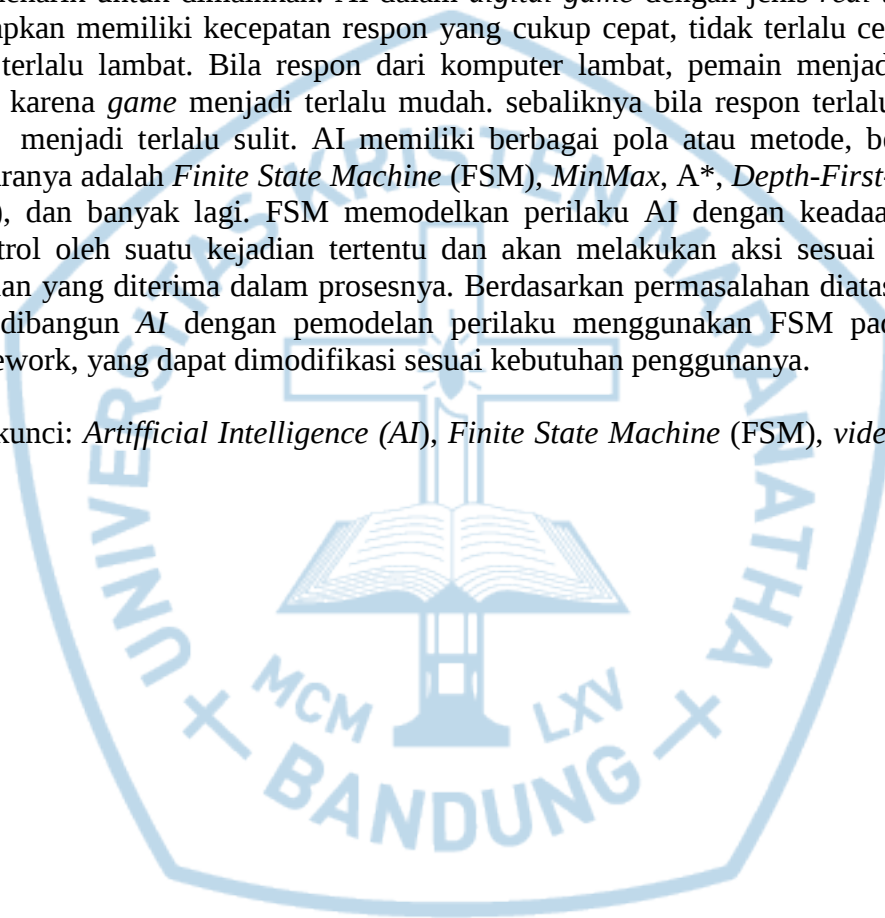


ABSTRAK

Video game adalah jenis kegiatan permainan, dilakukan dalam konteks tiruan realitas. Setiap *video game* memiliki jenis, jenis *video game* digunakan untuk membagi *video game* berdasarkan interaksi permainan mereka. Seiring perkembangan *video game* dengan berbagai jenisnya, dibutuhkan pula kecerdasan buatan dalam *video game*. *Artificial Intelligence* (AI) adalah kecerdasan buatan yang diimplementasikan pada mesin agar mesin seperti dapat berpikir. AI dibutuhkan dalam *video game* untuk merancang dan menghasilkan *video game* yang menyenangkan serta antarmuka antara manusia dan komputer yang cerdas dan menarik untuk dimainkan. AI dalam *digital game* dengan jenis *real-time* ini diharapkan memiliki kecepatan respon yang cukup cepat, tidak terlalu cepat dan tidak terlalu lambat. Bila respon dari komputer lambat, pemain menjadi cepat bosan karena *game* menjadi terlalu mudah. sebaliknya bila respon terlalu cepat, *game* menjadi terlalu sulit. AI memiliki berbagai pola atau metode, beberapa diantaranya adalah *Finite State Machine* (FSM), *MinMax*, *A**, *Depth-First-Search* (DFS), dan banyak lagi. FSM memodelkan perilaku AI dengan keadaan yang dikontrol oleh suatu kejadian tertentu dan akan melakukan aksi sesuai dengan kejadian yang diterima dalam prosesnya. Berdasarkan permasalahan diatas, maka akan dibangun AI dengan pemodelan perilaku menggunakan FSM pada .Net Framework, yang dapat dimodifikasi sesuai kebutuhan penggunaanya.

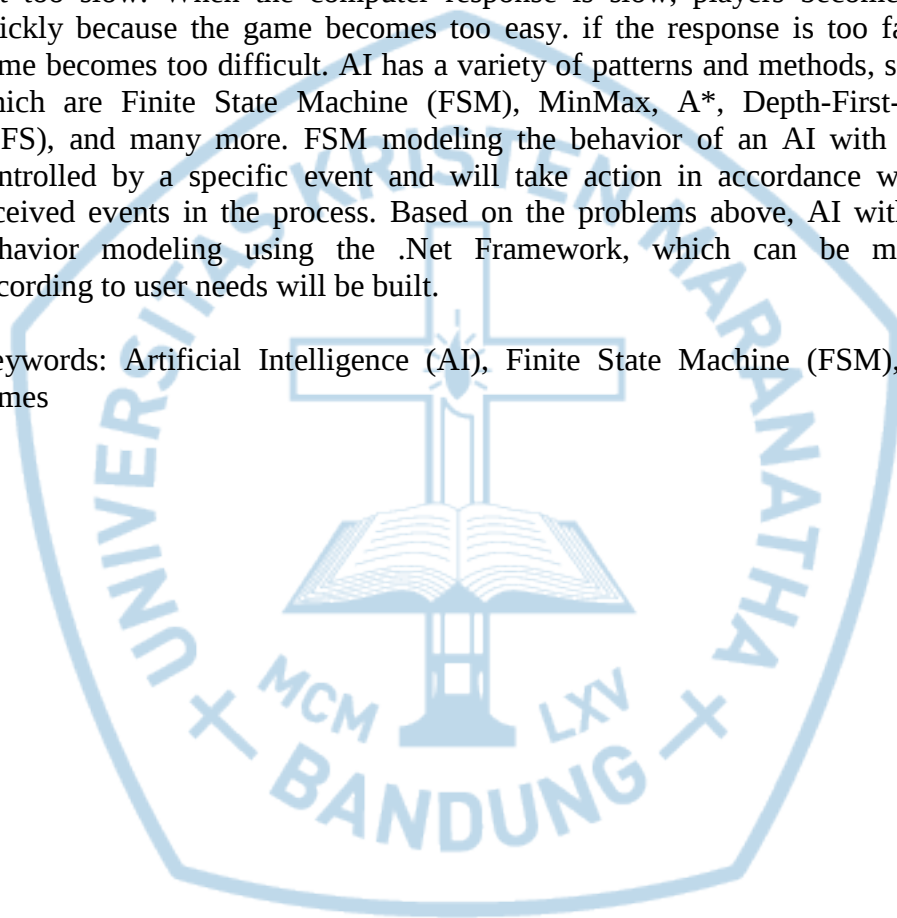
Kata kunci: *Artificial Intelligence* (AI), *Finite State Machine* (FSM), *video game*



ABSTRACT

Video games are kind of activities, carried out in the context of a mock reality. Every video game has a type, the type of video game is used to divide the video game based on the interaction of their game. Along with the development of video games, artificial intelligence (AI) is necessary in video games. AI is implemented on a machine so the machine can think. AI is necessary in video games to design and produce a video game that is fun so the interaction between humans and computers are interesting to play. AI in real-time digital game is expected to have enough response speed, not too fast and not too slow. When the computer response is slow, players become bored quickly because the game becomes too easy. if the response is too fast, the game becomes too difficult. AI has a variety of patterns and methods, some of which are Finite State Machine (FSM), MinMax, A*, Depth-First-Search (DFS), and many more. FSM modeling the behavior of an AI with a state controlled by a specific event and will take action in accordance with the received events in the process. Based on the problems above, AI with FSM behavior modeling using the .Net Framework, which can be modified according to user needs will be built.

Keywords: Artificial Intelligence (AI), Finite State Machine (FSM), video games



DAFTAR ISI

LEMBAR PENGESAHAN	i
PERNYATAAN ORISINALITAS LAPORAN PENELITIAN.....	Error! Bookmark not defined.
PERNYATAAN PUBLIKASI LAPORAN PENELITIAN.....	Error! Bookmark not defined.
PRAKATA	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI.....	1
DAFTAR GAMBAR	2
DAFTAR TABEL.....	4
BAB 1. PENDAHULUAN	5
1.1 Latar Belakang Masalah.....	5
1.2 Rumusan Masalah	6
1.3 Tujuan Pembahasan	7
1.4 Ruang Lingkup Penelitian.....	7
1.5 Metode Penelitian	7
1.6 Sistematika Pembahasan	8
BAB 2. KAJIAN TEORI	9
2.1 Finite State Machine (FSM).....	9
2.2 Dijkstra Algorithm	11
2.3 Monogame dan XNA Game Studio.....	14
BAB 3. ANALISA DAN RANCANGAN SISTEM.....	15
3.1 Pemodelan Jenis AI.....	15
3.2 Pemodelan Perilaku AI	15
3.3 Pemodelan Class	17
BAB 4. IMPLEMENTASI.....	21
4.1 Class Dijkstra.....	21
4.2 Class FSM.....	25
BAB 5. PENGUJIAN.....	28
5.1 Pengujian Pergerakan <i>agent</i>	28
5.2 Pengujian Deteksi <i>Agent</i>	30
5.3 Pengujian Perpindahan <i>State Agent</i>	31
5.4 Pengujian Perpindahan Jalur <i>Agent</i>	36
5.5 Pengujian <i>Custom Behavior</i>	41
BAB 6. SIMPULAN DAN SARAN	45
6.1 Simpulan	45
6.2 Saran	45
DAFTAR PUSTAKA	46

DAFTAR GAMBAR

Gambar 2.1: Ilustrasi Finite State Machine	10
Gambar 2.2: Contoh FSM pada permainan “Quake”	10
Gambar 2.3: Contoh algoritma dijkstra 1	12
Gambar 2.4: Contoh algoritma dijkstra 2	12
Gambar 2.5: Contoh algoritma dijkstra 3	12
Gambar 2.6: Contoh algoritma dijkstra 4	13
Gambar 2.7: Contoh algoritma dijkstra 5	13
Gambar 2.8: Contoh algoritma dijkstra 6	14
Gambar 3.1: Pola Perilaku Agent 1	15
Gambar 3.2: Pola Perilaku Agent 2	16
Gambar 3.3: Class Diagram	20
Gambar 4.1: Ilustrasi Proyek	21
Gambar 4.2: Fungsi <i>execute</i>	22
Gambar 4.3: Fungsi <i>getMinimum</i>	22
Gambar 4.4: Fungsi <i>findMinimalDistances</i>	23
Gambar 4.5: Fungsi <i>getNeighbors</i>	24
Gambar 4.6: Fungsi <i>getPath</i>	24
Gambar 4.7: Fungsi <i>getStateList</i> dan <i>getState</i>	25
Gambar 4.8: Fungsi <i>twoWayMove</i> dan <i>oneWayMove</i>	25
Gambar 4.9: Fungsi <i>movement</i> dan <i>reRoute</i>	26
Gambar 4.10: Fungsi <i>addLane</i> dan <i>createNode</i>	26
Gambar 4.11: Fungsi <i>detectNearestNode</i> dan <i>detectPlayerInRange</i>	27
Gambar 4.12: Fungsi <i>pathMaker</i>	27
Gambar 5.1: Pembuatan Jalur Pada 2 agent	28
Gambar 5.2: Pengujian Pergerakan Agent 1	29
Gambar 5.3: Pengujian Pergerakan Agent 2	29
Gambar 5.4: Pengujian Pergerakan Agent 3	29
Gambar 5.5: Pengujian Deteksi Agent 1	30
Gambar 5.6: Pengujian Deteksi Agent 2	31
Gambar 5.7: Pengujian Perpindahan State Agent 1	32
Gambar 5.8: Pengujian Perpindahan State Agent 2	32

Gambar 5.9: Pengujian Perpindahan <i>State Agent</i> 3	32
Gambar 5.10: Pengujian Perpindahan <i>State Agent</i> 4	33
Gambar 5.11: Pengujian Perpindahan <i>State Agent</i> 5	33
Gambar 5.12: Pengujian Perpindahan <i>State Agent</i> 6	34
Gambar 5.13: Pengujian Perpindahan <i>State Agent</i> 7	34
Gambar 5.14: Pengujian Perpindahan <i>State Agent</i> 8	35
Gambar 5.15: Pengujian Perpindahan <i>State Agent</i> 9	35
Gambar 5.16: Pengujian Perpindahan <i>State Agent</i> 10	36
Gambar 5.17: Pengujian Perpindahan Jalur <i>Agent</i> 1	37
Gambar 5.18: Pengujian Perpindahan Jalur <i>Agent</i> 2	37
Gambar 5.19: Pengujian Perpindahan Jalur <i>Agent</i> 3	37
Gambar 5.20: Pengujian Perpindahan Jalur <i>Agent</i> 4	38
Gambar 5.21: Pengujian Perpindahan Jalur <i>Agent</i> 5	39
Gambar 5.22: Pengujian Perpindahan Jalur <i>Agent</i> 6	40
Gambar 5.23: Pengujian Perpindahan Jalur <i>Agent</i> 7	40
Gambar 5.24: Pembuatan Jalur 2 <i>Agent</i> Yang Telah Dimodifikasi.....	41
Gambar 5.25: Pengujian <i>Custom Behavior</i> 1.....	42
Gambar 5.26: Pengujian <i>Custom Behavior</i> 2.....	42
Gambar 5.27: Pengujian <i>Custom Behavior</i> 3.....	42
Gambar 5.28: Pengujian <i>Custom Behavior</i> 4.....	43

DAFTAR TABEL

Tabel 5.1: Tabel Hasil Pengujian.....	43
--	-----------

