

SKRIPSI

PERMAINAN PAC-MAN BERBASIS SISTEM MULTIAGEN



Harseto Pandityo

NPM: 2013730060

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2019**

UNDERGRADUATE THESIS

MULTIAGENTS SYSTEM BASED PAC-MAN GAME



Harseto Pandityo

NPM: 2013730060

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2019**

LEMBAR PENGESAHAN

PERMAINAN PAC-MAN BERBASIS SISTEM MULTIAGEN

Harseto Pandityo

NPM: 2013730060

Bandung, 30 Juli 2019

Menyetujui,

Pembimbing

Dr.rer.nat. Cecilia Esti Nugraheni

Ketua Tim Penguji

Anggota Tim Penguji

Kristopher David Harjono, M.T

Husnul Hakim, M.T.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PERMAINAN PAC-MAN BERBASIS SISTEM MULTIAGEN

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 30 Juli 2019

Meterai Rp. 6000

Harseto Pandityo
NPM: 2013730060

ABSTRAK

Pac-Man adalah sebuah permainan klasik yang dibuat tahun 1980 oleh Namco, sangat populer pada jamannya. Prinsip permainan ini adalah permainan dalam sebuah labirin, dan di dalam labirin tersebut terdapat makanan, *power pellet*, buah ceri, Pac-Man, dan empat agen hantu yang mengejarnya. Untuk memenangkan permainan, pemain yang mengendalikan Pac-Man harus memakan semua makanan dan *power pellet* serta menghindari hantu.

Algoritma yang digunakan pada permainan Pac-Man orisinal tahun 1980 adalah sistem yang pasif dimana hantu tidak bisa bertindak independen berdasarkan data yang diberikan dan hanya dapat bertindak berdasarkan instruksi dari sistem utama. Gerakan yang bisa diambil hantu sangat terbatas. Oleh karenanya, tujuan dari penelitian ini adalah untuk merancang dan membuat permainan Pac-Man berbasis Sistem Multiagen, sehingga jalannya permainan menjadi lebih tidak terduga dan akan lebih mudah jika ingin diperbarui karena sistem multiagen tidak terhubung langsung pada sistem utama permainan. Agen adalah sistem yang dapat mendapatkan informasi, memproses, dan melakukan sebuah aksi secara independen tanpa instruksi langsung dari sistem lainnya.

Permainan Pac-Man berbasis Sistem Multi Agen dibuat menggunakan *framework* Jason dengan bahasa pemrograman AgentSpeak untuk bagian pemrograman agen dan Java untuk bagian pemrograman sistem lainnya. Perubahan untuk Pac-Man berbasis Sistem Multiagen dari game Pac-Man tahun 1980 adalah dengan mengubah algoritma cara hantu mendapatkan informasi. Pada Pac-Man tahun 1980, hantu mendapatkan informasi langsung melalui sistem utama setelah beberapa waktu tertentu. Sementara pada Pac-Man berbasis Sistem Multiagen yang akan dibuat, agen hantu akan mendapatkan informasi berdasarkan sensor yang dimiliki agen yang berupa jarak pandangan di depan agen yang dapat diatur jauhnya oleh pengguna. Ditambahkannya juga mekanisme jarak komunikasi antara agen untuk membatasi jarak mengirim pesan agen hantu ke agen hantu lainnya. Selain dari itu yang berubah lainnya adalah pada permainan Pac-Man tahun 1980, Pac-Man dikendalikan oleh pengguna. Pada permainan Pac-Man berbasis Sistem Multiagen yang akan dibuat, Pac-Man adalah sebuah agen yang akan bergerak secara otomatis berdasarkan langkah yang dipilih oleh agen Pac-Man.

Pengujian dilakukan dengan cara menjalankan program dan mengubah-ubah *parameter* yang bisa diubah oleh pengguna seperti kecepatan, poin yang didapatkan, jarak pandang agen hantu, dan jarak komunikasi agen hantu. Pengujian tersebut bertujuan untuk memastikan program sudah dapat berfungsi dengan baik dan juga untuk mencari kombinasi *parameter* terbaik yang dapat membuat agen hantu dan agen Pac-Man seimbang dalam kemampuannya untuk menyelesaikan tugas mereka.

Kata-kata kunci: Sistem Multi Agen, Pac-man, Jason, AgentSpeak, Java

ABSTRACT

Pac-man is a video game made in 1980 by Namco. The game mechanics for Pac-man is that it is set in a labyrinth which includes food, power pellets, cherries, four ghost agents as enemies, and the pac-man. To win the game, the player who controlled pac-man needs to eat all the food and power pellets while avoiding the ghosts.

The algorithm used in the Pac-man game that was made in 1980 includes a passive system where the ghosts can't act independently based on the data given. They can only act according to instructions from the main system. Therefore, the purpose of this research is to design and create a Pac-man game based on a Multi Agent System. The Agent is a system that can collect and process information and then act independently without direct instruction from other systems.

The Pac-man game with a Multi Agent system is made by using the Jason framework with the programming language AgentSpeak for the agent programming part and Java for the other systems's programming. The changes of the Pac-man Multi Agent System based compared to the 1980 original Pac-man game is by changing the ghost's mechanics of getting information. In the 1980's Pac-man, the ghost received direct information through the main system after a certain time. However, on the Multi Agent System of Pac-man that will be created, the ghost agent will get information based on the sensors that the ghosts have that consists of the visibility distance in front of the agent. The visibility distance can be regulated by the user. Communication distance mechanism between agent will also be added to limit the distance of sending messages between one ghost agent to another. Besides that, other changes between both version is the control of the Pac-man. In the original 1980's Pac-man, the pac-man is controlled by the user while in the Multi Agent System, the Pac-man will be an agent that will move automatically based on the best move that is chosen by the Pac-man agent.

The test is done by running the program and changing the parameters that can be changed by the user like speed, the overall points, the visibility distance of the ghost, and the communication distance between ghost agents. The test is also used to confirm that the program can function well and also to find the best parameter combination to make a balanced of ghost agent and Pac-man agent's ability to accomplish its tasks.

Keywords: Multi Agent System, Pac-man, Jason, AgentSpeak, Java

Dipersembahkan untuk Orang tua saya yang selalu mendukung saya pada setiap keputusan yang saya pilih, dan pembimbing saya Dr.rer.nat. Cecilia Esti Nugraheni yang sudah membimbing saya dalam menyelesaikan tugas akhir ini.

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT atas segala karunia-Nya selama ini sehingga penulis dapat menyelesaikan tugas akhir dan penulisan skripsi yang berjudul Permainan Pac-man Berbasis SMA. Selama pengerjaan tugas akhir, penulis menemui banyak masalah yang menyebabkan proses pengerjaan tugas akhir ini menjadi panjang. Penulis tidak akan bisa menyelesaikan skripsi ini tanpa bantuan orang lain. Oleh karena itu pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Keluarga penulis yang selalu memberikan nasihat dan dukungan kepada penulis.
2. Dr.rer.nat. Cecilia Esti Nugraheni selaku dosen pembimbing yang dengan sabar memberikan saran dan bimbingan yang penulis butuhkan untuk menyelesaikan tugas akhir ini.
3. Kristopher David Harjono, M.T dan Husnul Hakim, M.T. selaku dosen penguji yang dengan sabar menguji dan memberikan kritik membangun yang berguna dalam revisi tugas akhir ini.
4. Seluruh dosen pengajar Teknik Informatika UNPAR, yang telah memberikan ilmu penting dan membuat penulis tetap cinta dengan pemrograman.
5. Teman-teman terbaik penulis pada grup LINE "Keluarga Banyak Bicara" yang membantu penulis melupakan kesulitan penulis dalam menulis tugas akhir ini.

Bandung, Juli 2019

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Tujuan	1
1.4 Batasan Masalah	2
1.5 Metodologi	2
1.6 Sistematika Pembahasan	2
2 LANDASAN TEORI	5
2.1 Pac-Man	5
2.1.1 Kontrol dan Mekanisme	5
2.1.2 Cara Kerja Hantu pada Pac-Man versi 1980	7
2.2 Graf	13
2.3 Breadth First Search (BFS)	14
2.4 Agen dan Lingkungannya	16
2.4.1 Karakteristik dari Lingkungan	16
2.4.2 Jenis Agen	18
2.4.3 Sistem Multiagen	18
2.5 AgentSpeak dan Jason	18
2.5.1 AgentSpeak	18
2.5.2 Jason	20
3 ANALISIS	23
3.1 Analisis Pac-Man	23
3.2 Analisis Agen dan Lingkungan	24
3.2.1 Analisis Agen	24
3.2.2 Analisis BDI	24
3.2.3 Analisis Lingkungan	25
3.3 Analisis Graf	26
3.4 Analisis Algoritma BFS	26
3.5 Arsitektur Program	26
4 PERANCANGAN	27
4.1 Antarmuka Pengguna	27
4.2 Diagram Kelas	28

4.2.1	Agents	28
4.2.2	Component	29
4.2.3	Engine	29
4.2.4	GameObjects	29
4.2.5	Board	30
4.2.6	Level	30
4.2.7	Global	30
4.2.8	iAGhost	30
4.2.9	iAPacman	31
4.3	Diagram Aktifitas	38
5	IMPLEMENTASI DAN PENGUJIAN	41
5.1	Implementasi	41
5.1.1	Implementasi Antarmuka Pengguna	41
5.1.2	Implementasi Sistem Agen	42
5.1.3	Implementasi Lingkungan	44
5.2	Pengujian	45
5.2.1	Pengujian Parameter Standar / Fungsionalitas	45
5.2.2	Pengujian Parameter Eksperimental	46
6	KESIMPULAN DAN SARAN	49
6.1	Kesimpulan	49
6.2	Saran	49
	DAFTAR REFERENSI	51
	A KODE PROGRAM	53

DAFTAR GAMBAR

2.1	Permainan Pac-Man	5
2.2	Papan permainan Pac-man yang dibagi per-kotak posisi	7
2.3	Lokasi <i>tunnel</i> pada papan permainan	7
2.4	Jalur patroli hantu	8
2.5	Hantu mode <i>fright</i>	9
2.6	Blinky	10
2.7	Pinky	10
2.8	Ilustrasi dari cara kinerja Chase mode	10
2.9	Ilustrasi dari cara kinerja chase mode Clyde	11
2.10	Clyde	11
2.11	Inky	11
2.12	Ilustrasi dari cara kinerja <i>Chase</i> mode dari Inky.	12
2.13	Contoh graf, titik hitam adalah simpul dan garis-garis yang menghubungkan simpul adalah sisi.	12
2.14	Contoh Graf berarah, tidak terhubung, dan berbobot	14
2.15	Contoh Graf Pohon	14
2.16	Contoh Graf tidak berarah, terhubung, dan tidak berbobot	14
2.17	Ilustrasi Menelusuri graph menggunakan algoritma BFS	15
2.18	Contoh kode Algoritma BFS	15
2.19	Ilustrasi dari hubungan agen dan lingkungannya	16
2.20	Struktur Sistem Multi Agen	19
2.21	Contoh kode Jason	20
3.1	Ilustrasi jarak pandangan hantu	23
3.2	Ilustrasi zona komunikasi	23
4.1	Halaman <i>settings</i>	27
4.2	Halaman permainan	28
4.3	halaman hasil	28
4.4	Diagram Kelas Paket <i>Agents</i>	32
4.5	Diagram Kelas Paket <i>Component</i>	33
4.6	Diagram Kelas Paket <i>Engine</i>	34
4.7	Diagram Kelas Paket <i>GameObjects</i>	34
4.8	Diagram Kelas Paket <i>Level</i>	35
4.9	Diagram Kelas Paket <i>GUI</i>	35
4.10	Diagram Kelas Paket <i>iAGhost</i>	36
4.11	Diagram Kelas Paket <i>iAPacman</i> . <i>Method</i> yang memiliki * (asterisk) pada namanya adalah <i>parameter</i> lanjutan dari <i>method</i> yang bernama sama	37
4.12	Diagram Aktifitas Agen Hantu	38
4.13	Diagram Aktifitas Agen Pacman	39
5.1	Halaman <i>settings</i>	42
5.2	Halaman permainan	42

5.3 halaman hasil	43
-----------------------------	----

DAFTAR TABEL

2.1	Tabel perubahan mode hantu (Dalam detik)	9
2.2	Tabel kecepatan Pac-man	9
2.3	Tabel kecepatan Hantu	9
5.1	Tabel Pengujian Fungsional	46
5.2	Tabel Pengujian eksperimental ke-1	47
5.3	Tabel Pengujian Eksperimental ke-2	47

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pac-Man adalah sebuah permainan klasik yang dibuat tahun 1980 oleh Namco, sangat populer pada jamannya. Prinsip permainan ini adalah permainan dalam sebuah labirin, dan di dalam labirin tersebut terdapat makanan, *power pellet*, buah ceri, Pac-Man, dan empat agen hantu yang mengejarnya. Untuk memenangkan permainan, pemain yang mengendalikan Pac-Man perlu memakan semua makanan dan *power pellet* serta menghindari hantu.

Algoritma yang digunakan pada permainan Pac-Man orisinal tahun 1980 adalah sistem yang pasif dimana agen tidak bisa bertindak independen berdasarkan data yang diberikan dan hanya dapat bertindak berdasarkan instruksi dari pengendali utama. Pilihan variasi sangat terbatas sehingga mengurangi tantangan permainan bagi si pemain. Hal ini membuat sistem tidak fleksibel dan kemungkinan yang dapat diraih dari sistem lama lebih terbatas jika dibanding dengan sistem agen pada permainan modern. Aktor dalam permainan yang dapat bergerak seperti Pac-man dan hantu merupakan entitas berbeda dari sistem utama yang dapat melakukan aksi secara independen tanpa perintah dari sistem utama.

Maka dari itu, pada skripsi ini yang akan dilakukan adalah memodernisasikan sistem yang ada pada permainan Pac-Man dengan cara merancang dan membuat permainan Pac-Man berbasis Sistem Multiagen. Agen adalah sistem yang dapat mendapatkan informasi, memproses, dan melakukan sebuah aksi secara independen tanpa instruksi langsung dari sistem lainnya.

1.2 Rumusan Masalah

Berdasarkan pada latar belakang masalah yang telah didefinisikan, maka rumusan masalah yang diangkat dalam penelitian ini adalah:

- Bagaimana cara kerja permainan Pac-Man?
- Bagaimana mengkonversi cara kerja AI (*Artificial Intelligence*) yang ada pada Pac-Man versi tahun 1980 menjadi sistem multiagen yang dapat berkomunikasi dengan agen lainnya secara independen untuk mencapai tujuannya?
- Bagaimana mengukur performa dari sistem multiagen tersebut?

1.3 Tujuan

Dengan mengacu pada rumusan masalah, penelitian ini akan membahas permasalahan tersebut beserta analisis solusinya. Tujuan yang hendak dicapai dari penelitian ini adalah sebagai berikut:

- Mempelajari cara kerja permainan Pac-Man yang dirilis pada tahun 1980.

- Mengkonversi AI pada Pac-Man yang dirilis pada tahun 1980 menjadi Pac-Man modern yang memanfaatkan sistem multiagen yang dapat berkomunikasi dengan agen lain secara independen untuk mencapai tujuannya
- Merancang lingkungan permainan dari AI yang sudah diubah menjadi sistem multiagen.

1.4 Batasan Masalah

Dikarenakan adanya keterbatasan waktu penelitian serta terbatasnya kemampuan untuk menyesuaikan perangkat lunak dengan tujuan penelitian, maka didefinisikan batasan sebagai berikut:

- Permainan Pac-Man tidak akan memiliki fitur yang sama persis dengan permainan Pac-man tahun 1980. Hal itu dikarenakan perlunya menyesuaikan fitur untuk sistem multiagen.
- Perubahan dalam desain dari permainan Pac-man ini tidak pasti akan membuat desain permainan lebih bagus. Sesuai dengan tujuan penelitian, yang akan dibuat dalam perangkat lunak adalah agen yang berbasis multiagen. Fokus utama penelitian adalah tercapainya fungsi sistem yang didesain berbasis sistem multiagen dan tidak pada faktor lain dari permainan komputer seperti grafis dan estetika tampilan.

1.5 Metodologi

Metodologi yang akan digunakan pada penelitian ini adalah:

- Melakukan studi literatur pada topik-topik yang akan digunakan pada penelitian ini.
- Merancang perangkat lunak permainan Pac-Man berbasis sistem multiagen.
- Membuat perangkat lunak permainan Pac-Man berbasis sistem multiagen.
- Melakukan pengujian terhadap perangkat lunak yang dibuat, membuat sintesa dan analisa serta menyimpulkan hasil yang diperoleh.
- Mendokumentasikan seluruh penelitian dalam dokumen Skripsi

1.6 Sistematika Pembahasan

Sistematika pembahasan dalam penelitian ini adalah:

- Bab 1: Pendahuluan
Bab 1 berisi latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan dari penelitian ini.
- Bab 2: Dasar Teori
Bab 2 berisi teori-teori yang digunakan dalam penelitian ini seperti teori tentang agen, sistem multiagen, Pac-man, dan algoritma-algoritma yang akan digunakan.
- Bab 3: Analisis
Bab 3 berisi analisis terhadap teori-teori yang sudah dijabarkan di Bab 2 dan juga analisis terhadap perangkat lunak yang akan dibuat.
- Bab 4: Perancangan
Bab 4 berisi perancangan perangkat lunak seperti cara kerja, antarmuka pengguna, dan diagram kelas.

- Bab 5: Implementasi dan Pengujian

Bab 5 berisi pengujian terhadap perangkat lunak yang sudah di implementasikan untuk mengetahui perilaku agen dan menguji perbedaan antara Pac-Man dengan sistem multiagen dan Pac-Man 1980 yang tidak menggunakan sistem multiagen.

- Bab 6. Kesimpulan dan Saran

Bab 6 berisi kesimpulan dari hasil penelitian dan saran-saran untuk mengembangkan perangkat lunak ini lebih lanjut.