

SHOOTER GAME UNDERWATER MUTATION MENGGUNAKAN METODE FSM (FINITE STATE MACHINE)

Fachri Kurnia Utama

Teknik Informatika, Institut Teknologi Nasional Malang
fachri.utama@gmail.com

ABSTRAK

Perkembangan teknologi pada saat ini sangat pesat, salah satunya di bidang Game. Game merupakan software yang kebanyakan tujuannya adalah untuk menghibur usernya. Terdapat banyak sekali game mulai dari yang sederhana sampai yang rumit. Mulai dari yang gratis sampai yang mahal.

Game memiliki banyak macam dan jenis genre, diantaranya adalah genre Shooter. Genre Shooter adalah jenis game genre yang menggunakan aksi menembak sebagai fiturnya. Video game shooter mengharuskan pemainnya untuk bisa bereaksi cepat dan tepat dalam memberikan respon terhadap rangsangan penglihatan dan pendengaran pada permainan tersebut.

Setiap game pasti memiliki AI (Artificial Intelligence). Artificial Intelligence adalah cabang dari ilmu komputer yang membuat sebuah sistem agar dapat bertindak layaknya seperti seorang manusia. Di dalam game, AI digunakan untuk membuat musuh seolah – olah dapat bertindak seperti makhluk hidup. Salah satu metode AI adalah FSM (Finite State Machine), yang memiliki prinsip kerja sistem dengan menggunakan tiga hal berikut: State (Keadaan), Event (kejadian) dan action (aksi). Pada game Underwater Mutation, metode ini diterapkan pada musuh agar musuh dapat bereaksi dengan aksi dari player.

Setelah menyelesaikan game Underwater Mutation, dapat disimpulkan bahwa implementasi FSM (Finite State Machine) dapat diterapkan pada game 2 dimensi dengan indikasi musuh dapat berinteraksi terhadap player dalam kondisi tertentu. Hasilnya pergerakan dan respon musuh dalam game Underwater Mutation berjalan sesuai apa yang telah direncanakan. Game ini hanya dapat dimainkan di platform desktop. Untuk pengembangan selanjutnya disarankan agar game ini dapat dimainkan di platform lain selain desktop seperti platform android.

Kata kunci : *game, Artificial Intelligence, Finite State Machine, Shooter Game.*

1. PENDAHULUAN

1.1 Latar Belakang

Saat ini perkembangan teknologi semakin maju , terutama perkembangan dalam bidang komputer. Salah satunya *Game* atau permainan di dalam komputer, sebagian juga merupakan simulasi dari bentuk-bentuk nyata dalam kehidupan manusia. Perkembangan *game* saat ini sangat pesat, terdapat banyak sekali *game* yang dapat dimainkan di dalam komputer, baik dari *game* yang sederhana sampai *game* yang cukup rumit dalam pembuatannya. Game juga memiliki banyak jenis , diantaranya adalah jenis *Shooter Game*. *Shooter* adalah jenis game genre yang menggunakan aksi menembak sebagai fiturnya. Video game shooter mengharuskan pemainnya untuk bisa bereaksi cepat dan tepat dalam memberikan respon terhadap rangsangan penglihatan dan pendengaran pada permainan tersebut.

Oleh dari itu penulis ingin membuat *Shooter Game* dengan judul *Underwater Mutation*, dengan menggunakan metode FSM (*Finite State Machine*). *Finite State Machine* adalah metode kecerdasan buatan yang prinsip kerjanya menggunakan 3 hal yaitu *state* (keadaan), *event* (kejadian), *action* (aksi) . Pada periode waktu tertentu, sistem akan berada pada salah satu *state* yang aktif. Sistem dapat beralih atau bertransisi menuju *state* lain jika mendapatkan ada *event* tertentu, baik yang berasal dari perangkat luar atau komponen dalam sistemnya itu sendiri

Konsep game yang akan dibuat adalah game tembak – menembak dan melewati beberapa rintangan dan musuh di setiap level. Dengan dirancangnya game ini , penulis berharap game ini dapat bermanfaat dan menghibur bagi pemainnya.

1.2 Rumusan Masalah

1. Bagaimana merancang *Game* jenis *Shooter* dan *Adventure* menggunakan Unity 5.4 ?
2. Bagaimana implementasi metode FSM pada game ?

1.3 Batasan Masalah

1. *Game* yang dibuat menggunakan *Software Unity3D*
2. *Object* pada *game* dibuat menggunakan *Software Corel Draw*
3. *Game* ini memiliki 3 level
4. *Game* ini hanya dapat dimainkan di *PC* atau berbasis *Desktop*

1.4 Tujuan

1. Membangun Bagaimana merancang *Game* jenis *Shooter* dan *Adventure* menggunakan Unity 5.4
2. Mengimplementasikan metode FSM pada game.

2. TINJAUAN PUSTAKA

2.1 Game

Game adalah sebarang karya seni di mana peserta, yang disebut pemain, membuat keputusan untuk mengelola sumber daya yang dimilikinya melalui benda di dalam game demi mencapai tujuan. Sebuah game yang baik akan memberikan pengalaman bermain yang optimal, sebuah proses pengembangan game yang baik juga akan memberikan pengalaman yang istimewa serta proses pembelajaran yang luar biasa bagi mereka yang terlibat di dalamnya. Menurut Rahadian (2016) pada penelitian yang berjudul Penerapan Metode *Finite State Machine* Pada Game “*The Relationship*”, proses pengembangan sebuah game secara baik secara garis besar terbagi menjadi beberapa tahapan berikut:

- Merancang *Game*
Hal yang dilakukan pertama kali adalah menentukan jenis game yang akan dikembangkan.
- Perumusan *Gameplay*
Gameplay adalah pola, aturan, atau mekanisme yang mengatur bagaimana proses interaksi pemain dengan game yang diciptakan.
- Penyusunan *Asset* dan *Level Design*
Tahapan ini berfokus pada penyusunan konsep dari semua karakter serta *asset* (termasuk suara atau musik) yang diperlukan.
- *Test Play (Prototyping)*
Pada tahapan ini dilakukan tahapan pengujian game yang masih dalam bentuk prototype
- Development
Pada tahap ini seluruh konsep (karakter dan *asset*) yang sebelumnya telah tersusun mulai dikembangkan secara penuh.
- Percobaan
Tahap ini dilakukan untuk mengetahui apakah semua komponen utama dari game telah mampu memberikan pemain pengalaman seperti yang diharapkan.
- Rilis
Pada tahap ini game sudah siap untuk dirilis dan diperkenalkan pada target pemainnya.

2.2 Kecerdasan Buatan

Definisi Kecerdasan Buatan (AI) merupakan cabang dari ilmu komputer yang membuat sebuah sistem agar dapat bertindak layaknya seperti seorang manusia.

Ada tiga tujuan kecerdasan buatan, yaitu, membuat komputer lebih cerdas, mengerti tentang kecerdasan, dan membuat mesin yang lebih. Kecerdasan adalah kemampuan untuk belajar dan mengerti dari pengalaman, memahami pesan yang kontradiktif dan ambigu, menanggapi dengan cepat dan baik atas situasi yang baru, menggunakan penalaran dalam memecahkan masalah serta

menyelesaikannya dengan efektif. Menurut Ian (2009), adapun lingkup utama dalam kecerdasan buatan adalah :

1. Sistem Pakar
2. Pengolahan Bahasa Alami (Natural Language Processing)
3. Pengenalan Ucapan (Speech Recognition)
4. Robotika dan Sistem Sensor
5. Computer Vision
6. Intelligent Computer-aided Instruction
7. Game.

2.3 Finite State Machine

FSM adalah sebuah metodologi perancangan sistem kontrol yang menggambarkan tingkah laku atau prinsip kerja sistem dengan menggunakan tiga hal berikut: *State* (Keadaan), *Event* (kejadian) dan *action* (aksi). Pada satu saat dalam periode waktu yang cukup signifikan, sistem akan berada pada salah satu *state* yang aktif. Sistem dapat beralih atau bertransisi menuju *state* lain jika mendapatkan masukan atau *event* tertentu, baik yang berasal dari perangkat luar atau komponen dalam sistemnya itu sendiri. (Rahadian, 2016)

Menurut Suharian (2008) pada penelitian yang berjudul Pembuatan Game 3D *Fighting* Dengan Menggunakan *Finite State Machine* Sebagai Strategi Karakter, pada game ini *Finite State Machine* digunakan untuk menentukan bagaimana karakter bereaksi atau memutuskan tindakan berdasarkan kejadian atau kondisi yang sedang berlangsung. Pada *Finite State Machine* ini, penulis menggunakan tujuh buah *state* yang biasanya harus ada pada suatu game dengan jenis *fighting*. Hasil dari penerapan *Finite State Machine* pada game ini adalah terciptanya karakter musuh dimana dalam keadaan tertentu dapat memutuskan gerakan berupa maju, serang, bertahan, diam, jurus 1, jurus 2, tangkis dan mundur. Dengan demikian player dapat melawan musuh berupa NPC yang dapat memutuskan gerakannya sendiri.

Menurut Sutaya (2013) sebuah *state* menangkap aspek-aspek yang relevan dari *history* yang dimiliki oleh sistem secara efisien. Sebuah *state* bisa mengabstrakkan urutan *event* dan menangkap satu *event* yang relevan. Konsep dari *state* ini mengurangi masalah pengidentifikasian bagian yang dieksekusi didalam kode untuk memastikan hanya variabel *state* yang akan diproses dibandingkan banyak variabel, sehingga bisa mengeliminasi banyak kondisional. Pada umumnya sebagian besar teknik pengimplementasian *state machine* menggunakan “*nested-switch*”

Menurut Shaleh (2009) pada penelitian yang berjudul Rancang Bangun Game Edukasi Ular Tangga Pada Aplikasi Mobile, pada game ini penerapan metode *Finite State Machine* terletak pada karakter NPC. Penerapan *Finite State Machine* pada karakter NPC ini bertujuan agar karakter NPC dapat memutuskan aksi yang akan dia lakukan saat bermain.

Dengan demikian player utama dapat memainkan *game* ular tangga dengan melawan *NPC* namun terasa seperti melawan manusia nyata. *State* yang terdapat pada karakter *NPC* diantaranya mengacak dadu, jalan, naik dan turun.

2.4 Unity 3D

Unity 3D dibangun pada tahun 2004 oleh David Helgason, Nicholas Francis dan Joachim Ante. Unity 3D merupakan *game engine* yang memungkinkan seseorang atau sebuah tim untuk membuat *game* dengan cepat. Unity diluncurkan secara gratis namun ada juga versi berbayar dengan beberapa tambahan fitur. (Roedavan, 2015)

3. Analisa Dan Perancangan

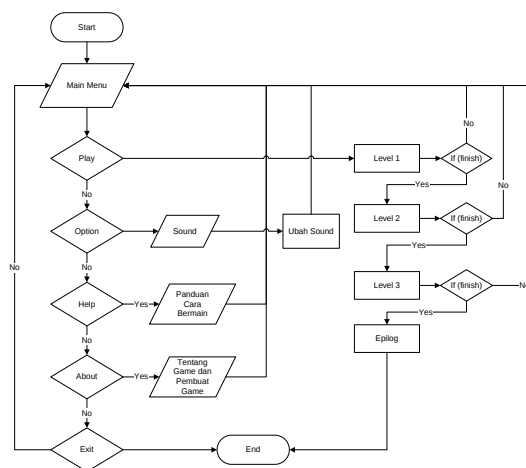
3.1 Analisis Target User

Analisis pengguna digunakan untuk mengetahui spesifikasi pengguna untuk dapat memainkan *game Underwater Mutation*, diantaranya:

1. Pengguna diprioritaskan untuk semua umur.
2. Berlatar belakang pendidikan mulai dari sd, smp, sma, kuliah hingga kerja.
3. Pengguna mengerti cara pengoperasian komputer.

3.2 Alur Game (Flowchart)

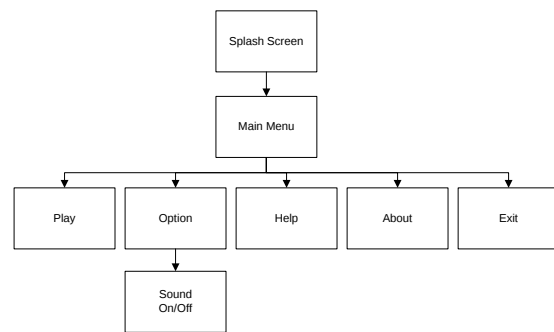
Perancangan Alur *Game* berfungsi untuk mengetahui alur proses awal program dijalankan sampai pada akhir program. Alur dari *game Underwater Mutation* ditunjukkan dengan *flowchart* pada gambar 1



Gambar 1 Desain system

3.3 Struktur Menu

Perancangan struktur menu adalah perancangan struktur urutan menu dari *game Underwater Mutation*. Struktur menu pada *game Underwater Mutation* ditunjukkan pada gambar 2



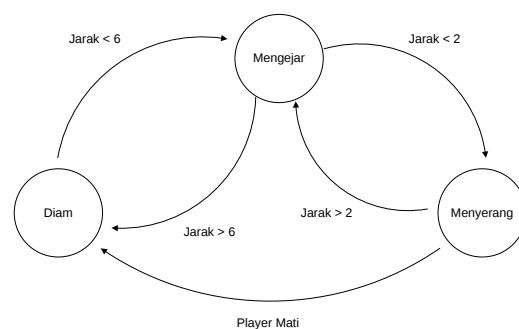
Gambar 2 Struktur menu

Gambar stuktur menu adalah tampilan dari struktur menu pada *game Underwater Mutation*. terdapat 5 menu utama pada *main menu* diantaranya adalah menu *play*, *option*, *help*, *about* dan *exit*. Untuk penjelasan tiap menu yang adalah:

1. *Play* adalah menu yang digunakan jika *player* ingin memulai bermain *game*.
2. *Option* adalah menu untuk mengatur setelan pada *game*.
3. *Help* adalah menu yang digunakan untuk menampilkan panduan bagaimana cara bermain *game Underwater Mutation*.
4. *About* adalah menu yang digunakan untuk menampilkan profil pembuat *game*.
5. *Exit* adalah menu untuk keluar atau menutup *game*.

3.4 Diagram Finite State Machine

Rancangan alur diagram *Finite State Machine* pada musuh 1 dapat dilihat pada gambar 3



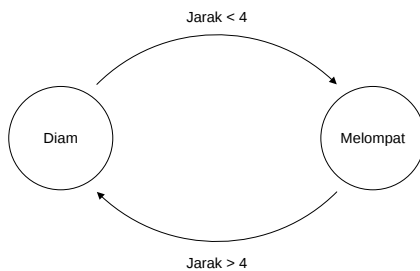
Gambar 3 Diagram FSM Musuh 1

Diagram FSM diatas adalah diagram yang menunjukkan AI musuh yang menggunakan metode *Finite State Machine*. Ada 3 *state* dari musuh yaitu *diam*, *mengejar* dan *menyerang*. *State* ini akan bergantian tergantung dengan kondisi yang sedang ada. Kondisi-kondisi yang mengubah *state* dari musuh 1 antara lain:

1. Musuh 1 akan berpindah dari *state* *diam* ke *state* *mengejar* saat jarak antara musuh dengan *player* kurang dari 6. Sebaliknya musuh 1 akan berpindah dari *state* *mengejar* ke *state* *diam* saat jarak antara keduanya lebih dari 6.

- Musuh 1 akan berpindah dari state mengejar ke state menyerang saat jarak antara musuh dengan player kurang dari 2. Sebaliknya musuh 1 akan berpindah dari state menyerang ke state mengejar saat jarak antara keduanya lebih dari 2..
- Musuh 1 akan berpindah dari state menyerang ke state diam saat player mati.

Sedangkan untuk rancangan alur diagram *Finite State Machine* pada musuh 2 dapat dilihat pada gambar 4



Gambar 4 Diagram FSM Musuh 2

Kondisi-kondisi yang mengubah state dari musuh 2 antara lain:

- Musuh 2 akan berpindah dari state diam ke state melompat saat jarak antara musuh dengan player kurang dari 4. Sebaliknya musuh 1 akan berpindah dari state melompat ke state diam saat jarak antara keduanya lebih dari 4.

3.5 Perancangan Karakter dan Lingkungan

Terdapat beberapa karakter dalam *game Underwater Mutation* diantaranya dapat dilihat pada tabel 1.

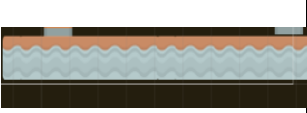
Tabel 1 Tabel karakter

Karakter	Gambar	Keterangan
Robot		Karakter yang dikendalikan oleh pemain
Ikan Mutan (Musuh 1)		Karakter yang akan mengejar dan menyerang pemain
Monster Duri (Musuh 2)		Sebagai jebakan

selanjutnya adalah pengenalan *environment* atau lingkungan. Pengenalan *level environment* adalah pembahasan mengenai lingkungan yang terdapat pada

game Underwater Mutation. Berikut penjabaran *environment* dapat dilihat pada tabel 2

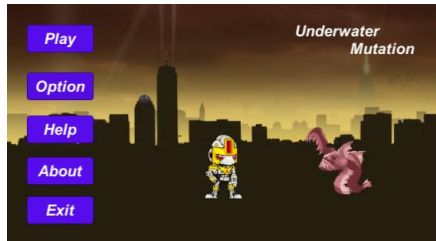
Tabel 2 Lingkungan

Nama	Tampilan	Keterangan
Terrain		Tempat Game pada level 1.
Platform		Lantai tempat berpijaknya pemain
Barrel Kimia 1		Barrel berisi bahan kimia
Barrel Kimia 2		Barrel berisi bahan kimia
Barrel Kimia 3		Barrel berisi bahan kimia
Barrel Kimia 4		Barrel berisi bahan kimia
Cairan Kimia		Cairan kimia dari barrel
Arrow		Penunjuk Arah

4. HASIL DAN PEMBAHASAN

4.1 Tampilan Menu Utama

Tampilan menu utama adalah tampilan awal yang akan muncul pada saat game *Underwater Mutation* dijalankan. Tampilan menu utama ditunjukkan pada Gambar 5

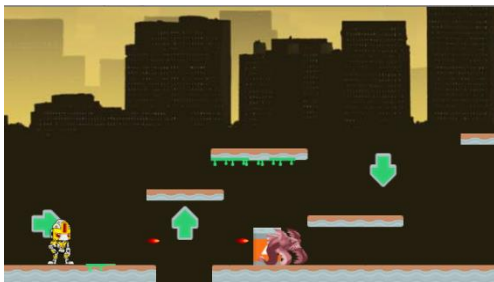


Gambar 5 Tampilan Menu Utama

Didalam menu utama terdapat tombol *play*, *option*, *help*, *about* dan *exit*.

4.2 Tampilan Game

Tampilan ini adalah tampilan saat game *Underwater Mutation* dijalankan. Tampilan game ditunjukkan pada gambar 6



Gambar 6 Tampilan Game

4.3 Pengujian Metode (Finite State Machine)

Pengujian metode adalah pengujian untuk mengetahui apakah metode yang telah diterapkan pada musuh apakah sudah berjalan dengan sesuai. Hasil pengujian FSM musuh 1 dapat dilihat pada Tabel 3

Tabel 3 Pengujian FSM musuh 1

State	Event	Action	Ket.
Diam	Player terlihat dekat	Mengejar	Sesuai
Mengejar	Player terlihat jauh	Diam	Sesuai
Mengejar	Posisi player dekat	Menyerang	Sesuai
Menyerang	Posisi player jauh	Mengejar	Sesuai
Menyerang	Player mati	Diam	Sesuai

Untuk pengujian FSM pada musuh 2 dapat dilihat pada tabel 4

Tabel 4 Pengujian FSM musuh 2

State	Event	Action	Ket.
Diam	Player terlihat dekat	Mengejar	Sesuai
Mengejar	Player terlihat jauh	Diam	Sesuai
Mengejar	Posisi player dekat	Menyerang	Sesuai
Menyerang	Posisi player jauh	Mengejar	Sesuai
Menyerang	Player mati	Diam	Sesuai

4.4 Pengujian Fungsionalitas

Pengujian fungsionalitas adalah pengujian setiap fungsi atau fitur yang diterapkan pada game *Underwater Mutation*. Hasil pengujian dapat dilihat pada Tabel 5

Tabel 5 Tabel Pengujian fungsionalitas

No.	Item Uji	Berhasil	Gagal
1.	Splash Screen	√	-
2.	Menu utama	√	-
3.	Button Play	√	-
4.	Button Option	-	√
5.	Button Help	√	-
6.	Button About	√	-
7.	Masuk Level 1	√	-
8.	Health Point player	√	-
9.	Health Point musuh	√	-
10.	Player dapat menyerang	√	-
11.	Musuh dapat mengejar	√	-
12.	Musuh dapat menyerang	√	-
13.	Health Point Player berkurang ketika diserang musuh	√	-
14.	Health Point musuh mati ketika di serang	√	-
15.	Player mati ketika Health Point Player = 0	√	-
16.	Animasi musuh	√	-
17.	Background sound	√	-
18.	Sound Effect	-	√

4.5 Pengujian Control Player

Pengujian *control player* adalah pengujian setiap fungsi dari tombol yang sudah diterapkan untuk menggerakkan player utama. Hasil pengujian *player* pada Table 6

Tabel 6 Pengujian Control Player

Tombol	Fungsi	Hasil
D	Menjalankan <i>player</i> ke arah kanan	Sesuai
A	Menjalankan <i>player</i> ke arah kiri	Sesuai
V	Membuat <i>player</i> menembak	Sesuai
Shift	Membuat <i>player</i> melakukan serangan jarak dekat	Sesuai
Space	Membuat <i>player</i> berlompat	Sesuai

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah menyelesaikan pembuatan *Game Underwater Mutation*, maka penulis dapat mengambil kesimpulan :

1. Implementasi FSM (*Finite State Machine*) dapat diterapkan pada *game* 2 dimensi dengan indikasi musuh dapat berinteraksi terhadap *player* dalam kondisi tertentu.
2. Gameplay pada *game* ini mengharuskan pemain untuk menembak musuh untuk melewati musuh sehingga *game* ini memiliki genre *Shooter*.
3. Fungsi dari menu belum berjalan sesuai apa yang telah direncanakan, terutama pada tombol option.
4. Pergerakan *unit* dan respon musuh dalam *game Underwater Mutation* berjalan sesuai apa yang telah direncanakan.

5.2 Saran

Setelah dilakukan pengujian terhadap *Game Underwater Mutation*, maka masih terdapat kekurangan sehingga untuk pengembangan lebih lanjut disarankan:

1. Menambah fitur *save game*, agar pemain dapat melanjutkan progress permainan yang sebelumnya telah disimpan

2. Menambah fitur *pause*, agar pemain dapat menghentikan sejenak jalannya *game*.
3. Penambahan level pada *game Underwater Mutation*, Karena pada *game* ini hanya terdapat 3 level.
4. Penambahan jenis variasi musuh agar *game* menjadi lebih menarik.

DAFTAR PUSTAKA

- [1] Ian Millington dan John Funge. 2009. *ARTIFICIAL INTELEGENCE FOR GAMES Second Edition*. Burlington: Elsevier Inc.
- [2] Jellyanto, F.A., Kuswardayan, I. and Suciati, N., 2016. Rancang Bangun Pembangkit World Dinamis menggunakan Algoritma Recursive Backtracking pada *Game 2D Platformer*" Mine Meander". *Jurnal Teknik ITS*, 5(2), pp.A591-A595.
- [3] Miftah Fauzan Rahadian, Addy Suyatno dan Septya Maharani. 2016. Penerapan Metode Finite State Machine Pada *Game "The Relationship"*. Skripsi. Samarinda: Universitas Mullawarman
- [4] Munawar, Shaleh. 2009. RANCANG BANGUN GAME EDUKASI ULAR TANGGA PADA APLIKASI MOBILE, Institut Teknologi Sepuluh Nopember Surabaya
- [5] Roedavan, Rickman, 2015. *Unity Tutorial Game Engine*. Bandung: Penerbit Informatika
- [6] Sutaya, I.W., 2013. PERANCANGAN DAN PEMBUATAN GENERATOR FRAMEWORK SISTEM EMBEDDED BERBASIS FSM (FINITE STATE MACHINE) DENGAN MEMANFAATKAN OPEN SOURCE ArgoUML. *Jurnal Pendidikan Teknologi dan Kejuruan*, 10(1).
- [7] Tjahjono, H., Liliana, L. and Gunadi, K., 2015. Pembuatan *Game Cerita Rakyat Dengan Bentuk Adventure Game*. *Jurnal Infra*, 3(2), pp.pp-277.