

## PENERAPAN METODE FINITE STATE MACHINE UNTUK NON PLAYER CHARACTER PADA GAME LOST IN SPACE

**Muhammad Hanif Alqorni**

Teknik Informatika, Institut Teknologi Nasional Malang  
hanif.alqorni@gmail.com

### ABSTRAK

Terdapat media untuk bermain *game* yaitu konsol *game computer* dan *smartphone*. *Computer* salah satunya media yang paling banyak digunakan para pengembang industri *game* untuk mengeluarkan produknya. Dalam *game*, musuh dan karakter merupakan salah satu elemen yang paling penting dalam *video game*, untuk menjadikan *game* semakin menarik, karakter dan NPC (*Non Playable Character*), salah satunya musuh dalam *game* harus dapat meniru sifat atau perilaku manusia dalam dunia nyata. Untuk melakukan hal tersebut diperlukan kecerdasan buatan pada karakter NPC (*Non Playable Character*) pada musuh yaitu kecerdasan buatan FSM (*Finite State Machine*) yang menggunakan kecerdasan buatan FSM (*Finite State Machine*) pada karakter musuh agar dapat mendeteksi keberadaan *player*.

Dalam perancangan *game* ini, penulis menggunakan aplikasi berupa *Unity3D* dengan bahasa pemrograman C#. Penulis dalam membuat *game* ini menerapkan metode kecerdasan buatan *Finite State Machine* yang menggunakan 3 hal yaitu keadaan, kejadian, dan aksi, untuk menentukan aksi dengan jarak *player* ke musuh.

Hasil pengujian *game Lost in Space* yang telah dilakukan menunjukkan bahwa fungsi yang terdapat pada *game* dari menu utama hingga level 1 dan level 2 dapat berhasil. Metode kecerdasan buatan *Finite State Machine* yang diterapkan pada musuh, dapat mendeteksi keberadaan *player* dan menghasilkan *output* yang dapat menentukan aksi apa yang akan dilakukan oleh musuh.

**Kata kunci :** *Game, Unity3D, Finite State Machine, Non Player Character*

### 1. PENDAHULUAN

#### 1.1 Latar Belakang

Dari sekian banyak jenis genre *video game* di luar sana, *side-scrolling action* bisa dibilang termasuk salah satu yang tertua dan terpopuler. Genre ini menarik karena konsep dasarnya sederhana, namun *fun* dan bisa dikembangkan terus dengan berbagai inovasi sehingga tetap menarik. Sampai sekarang pun banyak *game side-scrolling action* baru terus bermunculan, masing-masing dengan keunikan tersendiri. Beberapa judul *game* dengan genre *side scrolling* antara lain: *Pinstripe*, *Limbo*, dan *This War of Mine*.

*Game-game* tersebut selain bergenre *side-scrolling* juga ada yang berpadu dengan genre fantasi untuk *Pinstripe* dan *Limbo*, sedangkan *This War of Mine* selain *side scrolling* dan fantasi juga terdapat genre *war*. *Game-game* tersebut memiliki keunikan masing-masing sehingga asyik untuk dimainkan. Namun sayangnya pada *game* tersebut hanya terdapat media hiburan dan tidak terdapat informasi tambahan atau ilmu yang bisa menambah pengetahuan dan wawasan dari pemain *game* tersebut.

Dari uraian tersebut penulis kemudian tergagas untuk mengembangkan sebuah *game* yang tidak hanya menjadi sarana hiburan namun juga sebagai sarana menambah wawasan dari pemain yang memainkan *game* yang akan penulis kembangkan. Maka dari itu penulis berusaha mengembangkan sebuah *game* yang berjudul *Lost in Space*, yang akan menggunakan metode *finite State Machine* yang diterapkan pada *non player character* dalam *game*. *Non player character* tersebut akan berperan sebagai

musuh dalam *game* yang harus dikalahkan oleh *player*.

Dalam *game Lost in Space*, selain pemain dapat bermain untuk mengalahkan musuh-musuh, mengambil beberapa item, dan menghadapi jebakan. Terdapat pula beberapa informasi tentang antariksa yang akan diberikan saat bermain, sehingga selain bermain pemain juga bisa belajar tentang dasar dari antariksa. *Game* ini akan dikembangkan dengan genre *side-scrolling* selain sebagai nilai tambah *game* juga agar menjadikan *game* ini lebih menarik untuk dimainkan.

#### 1.2 Rumusan Masalah

Dari latar belakang tersebut maka penulis membuat rumusan masalah untuk memecahkan masalah tersebut yaitu, bagaimana merancang dan membuat sebuah *game* dengan tema *side-scrolling* yang menggunakan metode *finite state Machines* dan berjudul *Lost In Space* yang dapat menghibur dan menambah wawasan bagi pemainnya?

#### 1.3 Batasan Masalah

Batasan masalah dari pembuatan *game* ini adalah :

1. *Game* ini dibuat dengan model 2D.
2. *Game* ini dibuat menggunakan tools dan *game engine* *Unity3D*.
3. *Game* ini dibuat dengan 2 level kesulitan dengan perbedaan pada musuhnya.
4. *Game* ini memiliki beberapa item untuk membantu pemain yang berupa item nyawa, item HP, dan item koin.

5. *Game* ini akan menampilkan beberapa informasi tentang antariksa saat player mengambil item tertentu.
6. Saat *game* dimulai *player* memiliki 2 nyawa.
7. Terdapat 3 jenis musuh yang akan dihadapi oleh pemain, yaitu: Alien A, Alien B, dan Alien C.
8. Saat HP pemain habis pemain harus mengulang dari awal level dimana pemain bermain.
9. Sedangkan saat nyawa habis pemain harus mengulang dari level 1.
10. *Game* ini hanya dapat dimainkan pada PC atau dekstop dengan sistem operasi Windows.
11. *Input game* ini menggunakan *keyboard* yang berupa tombol navigasi untuk menggerakkan karakter dan tombol "K" untuk menembak.

#### 1.4 Tujuan

Tujuan dari pembuatan *game* ini adalah :

1. Menciptakan suatu *game* untuk sarana hiburan dan menambah edukasi tentang antariksa.
2. Menerapkan Metode *Finite State Machine* untuk menentukan gerakan pada NPC (*Non Playable Character*).
3. Menciptakan suatu *game* untuk meningkatkan pengalaman berpetualang pemain

## 2. TINJAUAN PUSTAKA

### 2.1 Penelitian yang Terkait

Metode pembuatan kecerdasan buatan pada *game* dengan menggunakan Metode *Finite State Machine* juga pernah digunakan oleh peneliti sebelumnya. Maka peneliti juga perlu untuk mempelajari penelitian-penelitian sebelumnya sebagai dasar acuan dalam mengerjakan penelitian ini.

Rujukan penelitian pertama yaitu penelitian oleh Tito (2016) dengan judul *Pemodelan Perilaku Musuh Menggunakan Finite State Machine (FSM) Pada Game Pengenalan Unsur Kimia*. Dalam penelitian tersebut peneliti menggunakan metode *finite state machine* pada pemodelan perilaku musuh. Dengan menggunakan metode *Finite State Machine* musuh akan bereaksi pada *player* berdasarkan jarak dari *player* terhadap musuh.

Rujukan penelitian kedua yaitu penelitian oleh Rahadian (2016) dengan judul *Penerapan metode finite state machine pada game "the relationship"*. Dalam penelitian tersebut peneliti menggunakan metode *finite state machine* untuk menentukan respons karakter *non-player character* yang berbeda tergantung dari interaksi yang dilakukan oleh pemain. Saat pertama dijalankan NPC akan berada pada *State idle*, kemudian NPC akan mulai bergerak dan melakukan interaksi setelah diberi perintah oleh *player*.

Rujukan penelitian ketiga yaitu penelitian oleh Rostianingsih (2013) dengan judul *Perancangan dan pembuatan game simulasi pertanian dan peternakan*. Dalam penelitian tersebut peneliti menggunakan metode *finite state machine* untuk menentukan respons karakter *non-player character* yang menjadi pemain lain dalam *game*. NPC tersebut termasuk dalam

karakter utama, yang kemudian akan berinteraksi satu sama lain, dan metode *Finite State Machine* diterapkan pada aktivitas bercocok tanam dari setiap karakter.

### 2.2 Game

*Game* merupakan salah satu hiburan karena mampu mengurangi tingkat kepenatan seseorang dari rutinitas pekerjaan setiap hari. *Game* juga mampu meningkatkan kecerdasan seseorang ketika *game* tersebut memerlukan tingkat ketangkasan dari seorang pemain. Beberapa *game* yang beredar saat ini terdapat unsur mendidik, ketangkasan dan ada unsur kekerasan, maka ketika *game* itu diperjual belikan terdapat batasan umur pemakainya (Ardi, 2012)

### 2.3 AI (Artificial Intelligence)

Kecerdasan buatan (*artificial intelligence*) merupakan salah satu bagian dari ilmu computer yang mempelajari bagaimana membuat mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia bahkan bisa lebih baik dari pada yang dilakukan manusia.

Menurut McCarthy (1956), AI adalah untuk mengetahui dan memodelkan prose-proses berpikir manusia dan mendesain mesin agar dapat menirukan perilaku manusia. Cerdas, berarti memiliki pengetahuan ditambah pengalaman. Penalaran (bagaimana membuat keputusan dan mengambil tindakan), moral yang baik. manusia cerdas dalam menyelesaikan permasalahan karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan diperoleh dari belajar. Semakin banyak bekal pengetahuan yang dimiliki tentu lebih mampu menyelesaikan permasalahan. Tapi bekal pengetahuan tidak cukup, manusia juga diberi akal untuk melakukan penalaran, mengambil keputusan berdasarkan pengetahuan dan pengalaman yang dimiliki. Tanpa memiliki kemampuan untuk menalar dengan baik, manusia dengan segudang pengalaman dan pengetahuan tidak dapat menyelesaikan masalah dengan baik. Demikian juga dengan kemampuan pengalaman yang memadai manusia tidak bisa menyelesaikan masalah dengan baik (Dahria, 2008)

### 2.4 FSM (Finite State Machine)

Finite state machine adalah sebuah metodologi perancangan *system control* yang menggambarkan tingkah laku atau prinsip kerja dengan menggunakan tiga hal yang meliputi: *State* (keadaan), *Event* (Kejadian) dan *Action* (aksi). Pada satu saat dalam periode waktu yang cukup signifikan, *system* akan berada pada salah satu *state* yang aktif. *System* 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 (misal interupsi *timer*).

Transisi keadaan ini umumnya juga disertai aksi oleh aksi yang dilakukan oleh *system* ketika menanggapi masukan yang terjadi. Aksi yang dilakukan tersebut dapat berupa aksi yang sederhana atau melibatkan rangkaian proses yang relative kompleks. Berdasarkan sifatnya metode FSM ini sangat cocok untuk digunakan sebagai basis perancangan perangkat lunak pengendalian yang reaktif dan *real time*. Salah satu keuntungan nyata

penggunaan FSM adalah kemampuannya dalam mendekomposisi aplikasi yang relatif besar dengan hanya menggunakan sejumlah kecil *item state*.

Selain untuk bidang *control*, penggunaan metode ini pada kenyataannya juga umum digunakan sebagai basis untuk perancangan protokol-protokol komunikasi, perancangan perangkat lunak *game*, aplikasi *WEB* dan sebagainya, dalam bahasa pemrograman procedural seperti bahasa C, FSM umumnya direalisasikan menggunakan *statemen control switch case* atau / dan *if .... Then*. Dengan menggunakan statemen-statemen kontrol ini, aliran program secara praktis akan mudah dipahami dan dilacak jika terjadi kesalahan logika (Setiawan, 2006)

**3. ANALISIS DAN PERANCANGAN**

**3.1 Story Line**

*Game* ini menceritakan tentang seorang astronot yang mengalami kerusakan pada roket ruang angkasa miliknya yang akan dia gunakan untuk pulang ke Bumi. Untuk alur cerita *game* ini yaitu :

1. Di ceritakan seorang astronot yang dinamai Henry, dia adalah astronot pertama dari Indonesia yang dikimkan ke luar angkasa untuk misi penelitian.
2. Namun pada saat perjalanan roket luar angkasanya mengalami masalah dan ada bagian dari roketnya yang terlepas sehingga dia tidak dapat melanjutkan perjalanan.
3. Maka untuk dapat kembali pulang Henry harus pergi mengambil mesin dari roketnya.
4. Namun ternyata mesin dari roketnya itu dijaga oleh alien yang jahat.
5. Untuk dapat mengambil mesinnya, Henry harus mengalahkan alien jahat yang menghadang.
6. Setelah mengalahkan alien yang berjaga tersebut, akhirnya Henry dapat kembali ke roket ruang angkasanya, dan kemudian memasangkan bagian dari roketnya yang terlepas tadi.
7. Henry merasa sangat senang dan lega akhirnya dia dapat kembali pulang ke bumi dengan selamat.

**3.2 Desain Level**

*Game* ini terbagi menjadi dua level utama. Masing-masing level memiliki latar, musuh dan tingkat kesulitan yang berbeda-beda. Beberapa aturan dari *game* ini yaitu :

- a. Player memiliki nyawa dalam bentuk hati.
- b. Player memiliki HP (*health point*) dalam bentuk bar.
- c. Player mempunyai tiga aksi utama, yaitu *primary attack* (menembak super), *secondary attack* (menembak), dan *jump*.
- d. Player mati jika darah / HP Bar habis
- e. Terdapat item koin disepanjang perjalanan sebagai poin tambahan.
- f. Terdapat item untuk menambah *health point* (HP)
- g. Terdapat item untuk menambah nyawa.
- h. Terdapat item untuk mendapat senjata baru.
- i. Terdapat 3 musuh yang berupa Alien A, Alien B, dan Alien C.
- j. Setiap membunuh musuh selain mendapat poin juga terdapat item.

k. Poin ditentukan dari jumlah musuh yang dibunuh dan jumlah koin yang dikumpulkan.

- l. Untuk menyelesaikan permainan ini, pemain harus melewati semua level dan mengalahkan boss utama.
- m. Apabila HP habis pemain harus kembali ke awal level.
- n. Apabila nyawa habis pemain harus mengulang ke level 1.

Level 1 adalah level pertama yang harus dimainkan pada *Game Lost In Space*, berikut adalah beberapa aturan pada Level 1:

- a. Player pertama diberikan dua nyawa.
- b. Terdapat 1 item penambah HP.
- c. Terdapat 1 item senjata baru.
- d. Terdapat 20 item koin
- e. Terdapat 7 musuh Alien A.
- f. Terdapat 1 musuh Alien B yang akan menambah poin jika dibunuh.

Level 2 adalah level kedua yang harus dimainkan pada *Game Lost In Space* setelah berhasil melewati level 1, berikut adalah beberapa aturan pada Level 2:

- a. Player pertama diberikan dua nyawa.
- b. Terdapat 1 item penambah nyawa.
- c. Terdapat 2 item penambah HP.
- d. Terdapat 1 item senjata baru.
- e. Terdapat 30 item koin.

Terdapat 3 musuh Alien A dan 4 Alien B terdapat 1 musuh Alien C.

**3.3 Kontrol dalam Game**

Kontrol pada *player* ini digunakan untuk menggerakkan *player* yang nantinya pada gerakan-gerakan tersebut ada pada tiap-tiap tombol pada keyboard yang sudah diatur seperti pada tabel 3.1.

Tabel 3.1 Penjelasan Kontrol Pada *Player*



No	Tombol	Fungsi
1	A / ←	Menggerakkan ke kiri
2	D / →	Menggerakkan ke kanan
3	Spasi	<i>Player</i> meloncat
4	K	<i>Player</i> melakukan aksi berupa serangan (Menembak)
5	L	<i>Player</i> melakukan aksi berupa serangan (Jurus)

**3.4 Desain Karakter**

Perancangan karakter merupakan pembahasan mengenai karakter yang terlibat dalam *Game Lost in Space*. Karakter pada *Game Lost in Space* pada Tabel 3.2.

Tabel 3.2 Penjelasan karakter

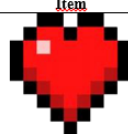

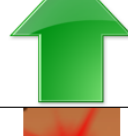



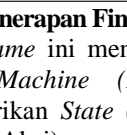
No	Desain	Keterangan
1		Karakter utama pada <i>game</i> .
2		Musuh 1 (Alien A)

3		Musuh 2 (Alien B)
4		Musuh 3 (Alien C)

**3.5 Item Pada Game**

Dalam game *Lost in Space* ini terdapat 7 macam item tambahan untuk menambah daya tarik user untuk memainkan game ini. Berikut adalah *item game* yang akan dijelaskan pada Tabel 3.3 berikut ini.

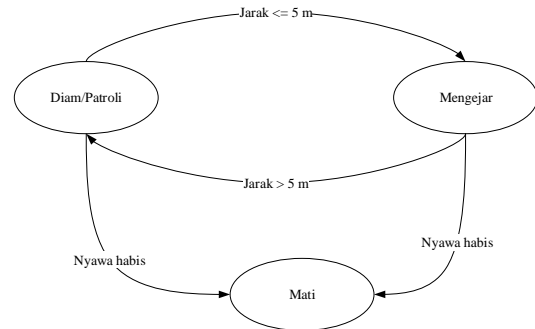
Tabel 3.3. Item-Item Pada Game

No	Item	Keterangan
1		Gambar <u>disamping</u> adalah <u>item</u> penambah <i>Health</i> yang akan menambah <i>Health poin</i> pada <i>player</i> .
2		Gambar <u>disamping</u> adalah <u>item</u> koin yang akan menambah <i>score</i> pada <i>game</i> .
3		Gambar <u>disamping</u> adalah <u>item</u> <i>level up</i> yang akan mengaktifkan senjata spesial pada <i>player</i> .
4		Gambar <u>disamping</u> adalah <u>item</u> <i>star</i> yang akan memunculkan gambar informasi antariksa.
5		Gambar <u>disamping</u> adalah <u>item</u> <i>Lives</i> (Nyawa) yang akan menambah <i>Lives</i> (nyawa) pada <i>player</i> .
6		Gambar <u>disamping</u> adalah <u>item</u> rintangan yang akan <i>health</i> pada <i>player</i> .
7		Gambar <u>disamping</u> adalah <u>item</u> mesin roket yang merupakan syarat yang harus diambil <i>player</i> untuk memenangkan <i>game</i> .

**3.6 Penerapan Finite State Machine pada Game**

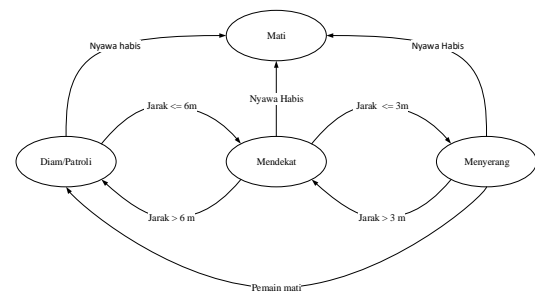
Game ini menggunakan 1 metode yaitu *Finite State Machine (FSM)* yang digunakan untuk memberikan *State* (Keadaan), *Event* (Kejadian) dan *Action* (Aksi).

Sifat perilaku musuh Alien A hanya mengejar karakter. *Health poin player* akan berkurang jika musuh A menyentuh *player*. Musuh akan mengejar *player* jika *player* masuk dalam jangkauan musuh Alien A. *Player* dan musuh A akan mati jika *Health poin* sudah habis, seperti ditunjukkan pada Gambar 3.1.



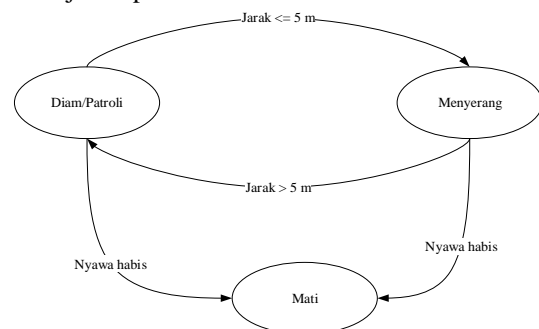
Gambar 3.1 Konsep FSM pada musuh a

Sedangkan sifat perilaku musuh Alien B hanya mengejar dan menyerang karakter. *Health poin player* akan berkurang jika musuh B menyentuh *player* dan terkena serangan dari Musuh Alien B. Musuh akan mengejar *player* jika *player* masuk dalam jangkauan 1 musuh Alien A, dan akan diam jika keluar jangkauan. Kemudian musuh akan menyerang *player* jika *player* masuk dalam jangkauan 2. *Player* dan musuh B akan mati jika *Health poin* sudah habis, seperti ditunjukkan pada Gambar 3.2.



Gambar 3.2 Konsep FSM pada musuh b

Sifat perilaku musuh Alien C hanya menyerang karakter. *Health poin player* akan berkurang jika musuh A menyentuh *player*. Musuh akan langsung menyerang *player* jika *player* masuk dalam jangkauan musuh Alien A. *Player* dan musuh A akan mati jika *Health poin* sudah habis, seperti ditunjukkan pada Gambar 3.3



Gambar 3.3 Konsep FSM pada musuh c

**4. IMPLEMENTASI DAN PENGUJIAN**

**4.1 Implementasi Sistem**

Tahap implementasi merupakan tahap kelanjutan dari kegiatan perancangan sistem. Wujud dari hasil implementasi ini nantinya adalah sebuah sistem yang siap untuk diuji dan digunakan.

**4.2 Implementasi Perangkat Keras**

Spesifikasi perangkat keras (*hardware*) yang digunakan untuk mengimplementasikan aplikasi *game Lost in Space* dapat dilihat pada tabel 4.1.

Tabel 4.1 Perangkat keras yang digunakan

Perangkat Keras	Keterangan
Processor	Intel i3 2.20GHz
RAM	4,00 GB
Harddisk	500 GB
VGA	Onboard

**4.3 Implementasi Perangkat Lunak**

Spesifikasi perangkat lunak (*software*) yang digunakan untuk mengimplementasikan aplikasi *game Lost in Space* dapat dilihat pada tabel 4.2.

Tabel 4.2 Perangkat lunak yang digunakan

Perangkat Lunak	Keterangan
Sistem operasi	Windows 7

**4.4 Implementasi Menu Utama**

Tampilan menu utama adalah tampilan awal yang akan muncul pada saat pemain menjalankan *Game Lost in Space*. Pada tampilan awal ini berisikan tombol *Play*, *Help*, *About*, dan *Exit*. *Play* untuk memulai *game* baru, Tombol *Help* berisi cara *controller player*, tombol *about* berisi tentang pembuat *game* dan tombol *Exit* untuk keluar dari *game*. tampilan menu utama seperti gambar 4.1



Gambar 4.1 tampilan menu utama

**4.5 Implementasi Level 1**

Pada *game level* pertama pemain harus melewati daerah di planet lain yang penuh dengan musuh dengan kecerdasan mengejar dan menyerang *player* supaya tidak bisa menyelesaikan level 1 dari musuh yang telah dilengkapi dengan pertahanan tubuh yang kuat dan halangan berupa bebatuan tajam kecil yang memotong jalan sehingga dapat mengalahkan *player*. Jika *player* dapat menyelesaikan misi pada level pertama maka *player* dapat melanjutkan ke level kedua. Level pertama pada *game* ini bersifat *easy*, artinya tantangan yang diberikan tidak terlalu kuat. Tantangan tersebut terletak pada musuh. Jika pada level pertama musuh akan menyerang dengan mengejar *player* dan bos yang akan mengejar kemudian menyerang *player*. Berikut merupakan desain layout pada level pertama ditunjukkan pada Gambar 4.2.



Gambar 4.2 Level 1 pada *game*

**4.6 Implementasi Level 2**

Pada *game level* kedua pemain harus melewati planet yang penuh dengan musuh dengan kecerdasan mengejar dan menyerang *player* supaya tidak bisa menyelesaikan level. Dan musuh serangan yang kuat dan halangan berupa batu tajam yang dapat mengurangi *Health player* sehingga dapat mengalahkan *player*. Jika *player* dapat menyelesaikan misi pada level kedua maka *player* dapat memenangkan *game* ini. Level kedua pada *game* ini bersifat *hard*, artinya tantangan yang diberikan sangat kuat. Tantangan tersebut terletak pada *damage enemy* dan kecepatan serangan pada musuh. Jika pada level pertama hanya mengejar maka pada level 3 musuh dapat mengejar serta menembak, jika *player* mendekat dan masuk jangkauan musuh. Berikut merupakan desain *layout* pada level kedua ditunjukkan pada Gambar 4.3



Gambar 4.3 Level 2 pada *game*

**4.7 Implementasi Unlock Senjata**

Pada gambar 4.4 menggambarkan ketika karakter *player* menyentuh item *Power up* yang akan membuka kemampuan untuk menggunakan senjata roket dari *player*.

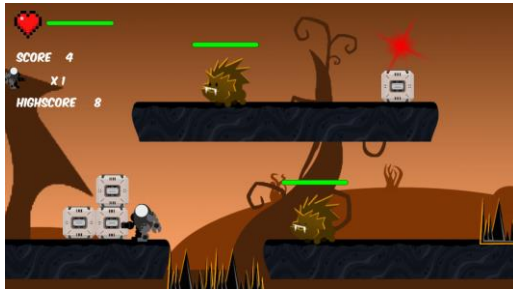


Gambar 4.4 Tampilan *Unlock Senjata*

**4.8 Implementasi Tambah Health**

Pada gambar 4.5 menggambarkan ketika karakter *player* menyentuh item hati yang akan menambah *Health poin* dari *player*.

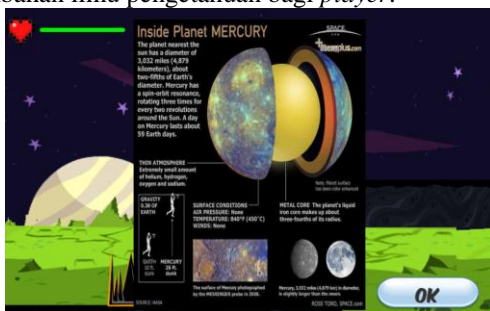




Gambar 4.5 Tampilan Tambah Health

#### 4.9 Implementasi Info Antariksa

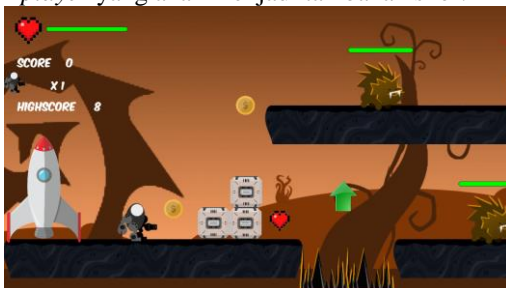
Pada gambar 4.6 menggambarkan ketika karakter *player* menyentuh item batang yang akan menampilkan informasi tentang antariksa sebagai tambahan ilmu pengetahuan bagi *player*.



Gambar 4.6 Tampilan Info Antarik

#### 4.10 Implementasi Item Koin

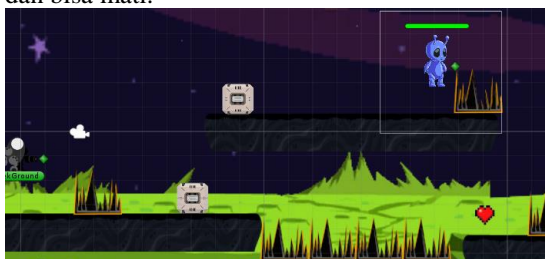
Pada gambar 4.7 menggambarkan adanya beberapa koin yang dapat diambil dan dikumpulkan oleh *player* yang akan menjadi tambahan skor.



Gambar 4.7 Tampilan Koin Dalam Game

#### 4.11 Implementasi Jebakana Pada Game

Pada gambar 4.8 menggambarkan adanya beberapa macam jebakan yang jika *player* menyentuh jebakan tersebut maka nyawa *player* akan berkurang dan bisa mati.



Gambar 4.8 Beberapa Jebakan

#### 4.12 Implementasi Musuh Menyerang Dekat

Pada game *Lost in Space* ini *player* memiliki dua serangan, yaitu serangan pertama yang berupa menabrak yang memiliki *damage* kecil, berikut

adalah tampilan serangan dekat yang dijelaskan pada gambar 4.9 berikut ini.



Gambar 4.9 Tampilan Serangan Dekat

#### 4.13 Implementasi Musuh Menyerang Jauh

Pada game *Lost in Space* ini *player* memiliki dua serangan, yaitu serangan kedua yang berupa menembak yang memiliki *damage* besar, berikut adalah tampilan serangan jauh yang dijelaskan pada gambar 4.10 berikut ini.



Gambar 4.10 Tampilan Serangan Jauh

#### 4.14 Implementasi Tembak Laser

Pada game *Lost in Space* ini *player* memiliki dua serangan, yaitu serangan pertama yang berupa menembak laser yang memiliki *damage* kecil, berikut adalah tampilan menembak dengan laser yang dijelaskan pada gambar 4.11 berikut ini.



Gambar 4.11 Tampilan Tembak Laser

#### 4.15 Implementasi Tembak Roket

Pada game *Lost in Space* ini *player* memiliki dua serangan, yaitu serangan pertama yang berupa menembak laser yang memiliki *damage* kecil, berikut adalah tampilan menembak dengan roket yang merupakan jurus yang dapat diaktifkan saat mengambil item *Power up* yang dijelaskan pada gambar 4.12 berikut ini.



Gambar 4.12 Tampilan Tembak Raket

**4.16 Pengujian AI (Artificial Intelligence)**

Pengujian *artificial intelligence* adalah pengujian mengenai fungsi yang berkaitan dengan *artificial intelligence* yang ada pada *Game Lost in Space*. Hasil pengujian dari *artificial intelligence* pada *game* ini seperti pada tabel 4.3

Tabel 4.3 Pengujian *artificial intelligence*

Keadaan	Kejadian	Tindakan	Hasil
Idle	Jika <i>player</i> berada pada jangkauan 1 musuh	Musuh mengejar <i>player</i>	Sesuai
Idle	Jika <i>player</i> berada pada jangkauan 2 musuh	Musuh menyerang <i>player</i> dengan senjata jarak jauh	Sesuai
Idle	Jika <i>player</i> berada pada jarak sangat dekat dengan musuh	Musuh memberikan damage dengan menabrak <i>player</i>	Sesuai
Mengejar / menyerang	Jika <i>player</i> keluar dari jangkauan musuh	Musuh akan kembali idle	Sesuai
Idle	Jika <i>player</i> berada pada jarak pandang musuh	Musuh tetap diam.	Sesuai

Dari tabel 4.3, menunjukkan bahwa kecerdasan buatan yang diterapkan pada musuh sudah berhasil diimplementasi sesuai dengan rancangan kecerdasan buatan yang sudah dibuat.

**4.17 Pengujian Kontrol Player**

Pengujian kontrol *player* adalah pengujian setiap fungsi dari tombol yang sudah diterapkan untuk menggerakkan karakter utama. Hasil pengujian *player* pada tabel 4.4

Tabel 4.4 Pengujian *control player*

NO	Tombol	Fungsi	Hasil
1	A	Bergerak ke arah kiri	Sesuai
2	D	Bergerak ke arah kanan	Sesuai
3	Spasi	Loncat	Sesuai
4	K	Menyerang	Sesuai
5	L	Menyerang dengan jurus	Sesuai

Dari tabel 4.4, menunjukkan bahwa semua fungsi tombol yang diterapkan untuk melakukan aksi pada *player* sudah berhasil diimplementasikan sesuai dengan rancangan kontrol *game* yang sudah dibuat.

**4.18 Pengujian Menu Utama**

Pengujian menu utama adalah pengujian setiap fungsi dari tombol yang sudah diterapkan pada menu utama, seperti ditunjukkan pada tabel 4.5

Tabel 4.5 Pengujian menu utama

No	Tombol	Fungsi	Hasil
1	<i>Play</i>	Mulai bermain dan masuk ke Level 1	Berhasil
2	<i>Help</i>	Menampilkan bantuan kontrol <i>game</i>	Berhasil
3	<i>About</i>	Menampilkan informasi pembuat <i>game</i>	Berhasil
4	<i>Exit</i>	Keluar dari <i>game</i>	Berhasil

Dari tabel 4.5, menunjukkan bahwa semua fungsi tombol pada menu utama sudah berhasil

semua sesuai desain menu yang sudah dibuat sebelumnya.

**4.19 Pengujian Level 1**

Pengujian Level 1 adalah pengujian setiap fungsi dari aksi yang diterapkan pada karakter *game*, seperti ditunjukkan pada tabel 4.6.

Tabel 4.6 Pengujian level 1

No	Skenario	Keluaran	Hasil
1	Player menyerang Alien 1	HP dari Alien 1 berkurang/mati	Berhasil
2	Player menyerang Boss 1	HP dari Boss 1 berkurang/mati	Berhasil
3	Alien 1 menyerang Player	HP dari Player berkurang/mati	Berhasil
4	Alien 1 menyerang Player	HP dari Player berkurang/mati	Berhasil
5	Player menyentuh jebakan	HP dari Player berkurang/mati	Berhasil
6	Player mengambil roket	Player memenangkan <i>game</i>	Berhasil

Dari tabel 4.6, menunjukkan bahwa seluruh interaksi *player* dengan karakter dan lingkungan dalam *game* pada level 1 sudah berhasil diimplementasikan sesuai dengan rancangan desain level 1 yang sudah dibuat.

**4.20 Pengujian Level 2**

Pengujian Level 2 adalah pengujian setiap fungsi dari aksi yang diterapkan pada karakter *game*, seperti ditunjukkan pada tabel 4.7.

Tabel 4.7 Pengujian level 2

No	Skenario	Keluaran	Hasil
1	Player menyerang Alien 1	HP dari Alien 1 berkurang/mati	Berhasil
2	Player menyerang Alien 2	HP dari Alien 2 berkurang/mati	Berhasil
3	Player menyerang Boss 2	HP dari Boss 1 berkurang/mati	Berhasil
4	Alien 1 menyerang Player	HP dari Player berkurang/mati	Berhasil
5	Alien 2 menyerang Player	HP dari Player berkurang/mati	Berhasil
6	Boss 2 menyerang Player	HP dari Player berkurang/mati	Berhasil
7	Player menyentuh jebakan	HP dari Player berkurang/mati	Berhasil
8	Player mengambil roket	Player memenangkan <i>game</i>	Berhasil

Dari tabel 4.7, menunjukkan bahwa seluruh interaksi *player* dengan karakter dan lingkungan dalam *game* pada level 2 sudah berhasil diimplementasikan sesuai dengan rancangan desain level 2 yang sudah dibuat.

**4.11. Pengujian Fungsionalitas**

Pada fungsionalitas ini *game* dicoba dijalankan pada Windows 7 dan Windows 8. Hasil dari pengujian dapat dilihat pada tabel 4.8.

Tabel 4.8 Pengujian Fungsional.

No	Item Uji	Windows 7		Windows 8	
		Berhasil	Gagal	Berhasil	Gagal
1	Menu Utama	√	-	√	-
2	<i>Button Start</i>	√	-	√	-
3	<i>Button Help</i>	√	-	√	-
4	<i>Button About</i>	√	-	√	-
5	<i>Button Exit</i>	√	-	√	-
6	Masuk Level 1 <i>Game</i>	√	-	√	-
7	Masuk Level 2 <i>Game</i>	√	-	√	-
8	<i>Bar Health Point</i> karakter	√	-	√	-
9	<i>Bar Health Point</i> Musuh	√	-	√	-
10	Loncat dan <i>animation state</i> loncat pada karakter	√	-	√	-
11	<i>Attack</i> dan <i>animation state attack</i> pada karakter	√	-	√	-
12	HP musuh berkurang ketika terkena <i>attack</i>	√	-	√	-
13	<i>Collision Detection</i> pada Musuh	√	-	√	-
14	<i>Item</i> tambah HP pada <i>player</i>	√	-	√	-

15	Item Koin	√	-	√	-
16	Musuh mengikuti pada daerah yang ditentukan	√	-	√	-
17	Musuh menyerang pada jarak yang ditentukan	√	-	√	-
18	Game tamat ketika player menyentuh roket	√	-	√	-

Berdasarkan pengujian tabel 4.8 menunjukkan bahwa pengujian fungsionalitas di Windows 7 dan 8 berhasil dijalankan sesuai dengan yang diuji.

**4.12. Pengujian Terhadap User**

Pengujian dilakukan untuk mengetahui apakah sistem sudah berjalan dengan baik atau belum. Pengujian dilakukan terhadap 10 mahasiswa ITN Malang, 7 dari 10 responden bukan dari pecinta game, dan 3 orang lagi seorang pecinta game atau Gamers. Berikut Hasil pengujian game Lost in Space. Hasil dari pertanyaan terhadap responden dapat dilihat pada Tabel 4.9.

Tabel 4.9. Pengujian Terhadap User

No	Pertanyaan	Penilaian		
		Baik	Cukup	Kurang
1	Desain Karakter Game	8	2	0
2	Desain Animasi pada Game	6	3	1
3	Kontrol pada game	7	2	1
4	Fitur game	8	2	0
5	Informasi game (Cerita, Narasi Game)	5	3	2
6	Game sudah menarik	8	2	0
Jumlah		42	14	4
Persentase		70%	23%	7%

Dari Tabel 4.9 dapat diketahui bahwa secara umum rata-rata dari responden menganggap 70% game baik, 23% menganggap game cukup baik, dan 7% menganggap game masih kurang.

**4.13. Pengujian Cheat**

Pengujian dilakukan untuk mengetahui apakah cheat sudah berjalan dengan baik atau belum. Pengujian dilakukan dengan mencoba shortcut cheat di semua level. Berikut hasil pengujian cheat game Lost in Space. Hasil dari pengujian dapat dilihat pada Tabel 4.10.

Tabel 4.10 Pengujian Fungsi Cheat

No	Tombol Shortcut	Fungsi	Penjelasan	Ket
1	R	Reset highscore	Tombol angka R pada keyboard jika tekan akan otomatis reset highscore menjadi 0	Sesuai
2	1	Menuju ke Level 1	Tombol angka 2 pada keyboard jika tekan akan otomatis menuju ke level 1 jika ingin mengulang level 1	Sesuai
3	2	Menuju ke Level 2	Tombol angka 2 pada keyboard jika tekan akan otomatis menuju ke level 2 tanpa harus mengalahkan Bos musuh	Sesuai
4	3	Menuju ke lokasi bos musuh level	Tombol angka 3 pada keyboard jika ditekan akan otomatis memindahkan karakter player pada posisi di dekat bos musuh	Sesuai

Dari tabel 4.10 dapat diketahui bahwa 100% sesuai dengan fungsi yang sudah diterapkan oleh penulis.

**5. KESIMPULAN DAN SARAN**

**5.1 Kesimpulan**

Setelah pembuatan Game Lost in Space, maka penulis dapat mengambil kesimpulan :

1. Implementasi kecerdasan buatan FSM (Finite State Machine) berhasil diterapkan pada game Lost in Space dengan aksi musuh dapat mengejar dan menyerang player dengan kondisi player masuk dalam jarak serang musuh pada musuh 1 (6m), pada musuh 2 (3m & 6m), dan pada musuh 3 (5m).
2. Semua aksi pada player berhasil diimplementasikan sesuai dengan yang dirancang sebelumnya.
3. Fungsi tombol pada menu utama sudah berhasil berjalan sesuai yang dirancang pada struktur menu.
4. Fungsi yang diterapkan pada karakter game pada level 1 dan level 2 berhasil seluruhnya sesuai dengan desain level yang sudah dibuat.
5. Fungsionalitas pada game berhasil dijalankan pada versi windows 7 maupun windows 8.
6. Sebagian besar responden yang melakukan pengujian pada game menyatakan jika game Lost in Space termasuk baik sebesar 70%.
7. Semua fungsi cheat pada game berhasil diterapkan untuk membantu pembuat dalam uji coba game.
8. Informasi tambahan berupa informasi dasar tentang antariksa berhasil diterpakn yang dapat menambah wawasan tentang antariksa bagi player.

**5.2 Saran**

Setelah dilakukan pengujian terhadap Game Lost in Space maka masih ada kekurangan sehingga untuk pengembangan lebih lanjut disarankan :

1. Dapat dikembangkan sebagai mobile game berbasis platform android dan ios atau versi webnya.
2. Dapat ditambah jumlah Level dan di tingkatkan kualitas grafiknya.

**DAFTAR PUSTAKA**

[1] Ardi, R., 2012. Pembuatan Game First Person Shooter (FPS) "Operation Zygm Force" menggunakan FPS Creator X10". Yogyakarta : STMIK AMIKOM.

[2] Dahria, M., 2008. Kecerdasan Buatan Jurnal SAINTIKOM, p.185

[3] Rahadian, M.F., Suyatno, A. and Maharani, S., 2016. PENERAPAN METODE FINITE STATE MACHINE PADA GAME "THE RELATIONSHIP". Jurnal Informatika Mulawarman (JIM), 11(1), pp.14-22.

[4] Rostianingsih, S., Satiabudhi, G. and Wijaya, H.K., 2013. Game Simulasi Finite State Machine Untuk Pertanian dan



- Peternakan* (Doctoral dissertation, Petra Christian University).
- [5] Setiawan, I., 2006. Perancangan Software Embedded System Berbasis FSM. *Semarang: Universitas Diponegoro*.
- [6] Tanjung, M.A.P. and Indonesia, T.I.U.K., 2011. Analisis Pengaruh Storytelling Terhadap Game Lorong Waktu–Pangeran Dipenogoro Sebagai Media Edukasi Sejarah.
- [7] TITO, B., 2016. Pemodelan Perilaku Musuh Menggunakan Finite State Machine (FSM) Pada Game Pengenalan Unsur Kimia. *Skripsi, Fakultas Ilmu Komputer*.