

PENGEMBANGAN GAME BATTLE TANK – WHEN FIGHT IS THE LAST CHOICE MENGUNAKAN METODE FSM (*Finite State Machine*)

Stefanus Daniel H P

Teknik Informatika, Institut Teknologi Nasional Malang
stefda20101994@gmail.com

ABSTRAK

Seiring perkembangan waktu, musuh dalam dunia game juga semakin cerdas dan unik. Pembuatan game dengan kecerdasan yang semakin unik, hal ini dimaksud dengan tujuan membuat musuh menjadi lebih menarik dan menantang. Pada awalnya musuh dalam game bersifat pasif dan tentunya tantangannya juga semakin berkurang, oleh karena itu dengan adanya perkembangan teknologi sekarang dapat merubah hal tersebut dengan adanya kecerdasan buatan pada musuh.

Dalam pengembangannya, metode FSM yang diterapkan telah sesuai dengan apa yang telah dirancang. Pada game ini FSM diterapkan pada musuh. state (keadaan) awal musuh adalah diam. kemudian ketika terjadi event (peristiwa) yaitu karakter datang mendekati musuh, maka musuh akan melakukan salah satu dari action (aksi) sesuai dengan yang telah ditentukan yaitu, patrol, mengejar, dan menyerang..

Manfaat yang diperoleh dari pembuatan game ini yaitu dapat memberikan hiburan dan pendidikan pada game adventure. Pengujian metode dan control player mendapatkan hasil yang sesuai dan berjalan 100%.

Kata kunci : *Game, FSM*

1. PENDAHULUAN

Perkembangan *game* di dunia sudah semakin pesat, tidak terkecuali di Indonesia. *Game* haruslah dapat memberikan tantangan kepada *player* agar *player* tidak merasa bosan dan bersemangat untuk melanjutkan *game* hingga tamat. *Game* akan disebut menarik jika *game* tersebut tidak terlalu mudah dan tidak terlalu sulit.

Melihat perkembangan *game* yang semakin pesat, hal ini akan mendatangkan banyak peluang kerja apabila generasi muda dapat membuat *game*. Dengan teknologi yang berkembang pesat menjadikan *game* juga berkembang cepat seiring dengan perkembangan teknologi, begitu juga dengan cara membuat *game*. Saat ini banyak *software game* yang memiliki kualitas dan *feature* yang mudah dipahami serta canggih untuk membuat animasi dalam *game*. Salah satunya adalah unity 3D. Unity adalah sebuah *tool* yang terintegrasi untuk membuat *game*, arsitektur bangunan dan simulasi. Unity tidak hanya merupakan sebuah *game engine*, namun juga merupakan sebuah editor.

Pada kesempatan ini, penulis akan membuat *game* dengan jenis *adventure* dengan judul *Game Battle Tank – When Fight is the Last Choice* dengan Metode FSM (*Finite State Machine*). Adapun alasan dalam pembuatan *game adventure* ini dengan menggunakan metode FSM ini karena musuh dapat patrol, mengejar, dan menyerang *player*. Hal ini memberikan pengalaman seru kepada pemain ketika bermain *game Battle Tank*. Sehingga, dengan membuat *game Battle Tank* ini dapat menarik lebih

banyak peminat *game adventure* khususnya *game Adventure. Game* ini terinspirasi dari *game Battle City* yang sudah dulu dirilis pada *console NES*, karena ide dari *game* tersebut adalah untuk menembaki dan mengalahkan semua musuh.

1.1. Rumusan Masalah

Rumusan masalah pada pembuatan *game battle tank* ini adalah sebagai berikut :

1. Bagaimana membuat *game Battle Tank* ?
2. Bagaimana merancang dan membangun *game Battle Tank* menggunakan Unity 5.0 ?
3. Bagaimana menerapkan FSM dalam *game Battle Tank* ?

1.2. Batasan Masalah

Batasan masalah pada *game battle tank* ini adalah sebagai berikut.

1. *Software* yang digunakan dalam pembuatan *game* ini adalah Unity 3D.
2. AI hanya diterapkan pada karakter musuh di dalam *game*.
3. *Control player* pada *game* menggunakan arah *keyboard* untuk berjalan dan *space* untuk menembak.

1.3. Tujuan

Tujuan dari pembuatan *game battle tank* ini adalah sebagai berikut :

1. Untuk menciptakan sebuah *game adventure* yang menarik.

2. Untuk dapat mempelajari penerapan AI pada game.
3. Untuk menciptakan sebuah grafik 2D pada karakter

2. TINJAUAN PUSTAKA

2.1. Game

Game atau permainan merupakan sesuatu yang dapat dimainkan dengan aturan tertentu sehingga ada yang menang ada yang kalah biasanya tidak dalam konteks serius atau dengan tujuan untuk refreshing. Permainan juga adalah sarana yang dibuat untuk memenuhi kebutuhan manusia akan hiburan, sehingga sampai saat ini, permainan terus berkembang sesuai dengan kebutuhan dan tidak dapat dipisahkan dari manusia. Berevolusi mengikuti perkembangan intelegensi seiring dengan perkembangan jaman

Teori permainan adalah suatu cara belajar yang digunakan dalam menganalisa sejumlah pemain maupun perorangan yang menunjukkan strategi-strategi rasional. Teori permainan pertama kali ditemukan oleh sekelompok ahli matematika pada tahun 1944. Teori itu di kemukakan oleh John Von Neumann and Oscar Morgenstern, menurutnya permainan terdiri atas sekumpulan peraturan yang membangun situasi 5 bersaing dari dua sampai beberapa orang atau kelompok dengan memilih strategi yang dibangun untuk memaksimalkan kemenangan sendiri ataupun untuk meminimalkan kemenangan lawan. Peraturan- peraturan menentukan kemungkinan tindakan untuk setiap pemain, sejumlah keterangan diterima setiap pemain sebagai kemajuan pemain dan sejumlah kemenangan ataupun kekalahan dalam berbagai situasi (Ekasari,2012).

2.2. Kecerdasan Buatan (Artificial Intelligence)

Kecerdasan Buatan (*Artificial Intelligence*) merupakan salah satu bagian dari ilmu komputer yang mempelajari bagaimana membuat mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia bahkan bisa lebih baik daripada yang dilakukan manusia. Menurut John McCarthy, 1956, AI : untuk mengetahui dan memodelkan proses-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 (pandai) dalam menyelesaikan permasalahan karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan diperoleh dari belajar. Semakin banyak bekal pengetahuan yang dimiliki tentu akan lebih mampu menyelesaikan permasalahan. Tapi bekal pengetahuan saja tidak cukup, manusia juga diberi akal untuk melakukan

penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang dimiliki. Tanpa memiliki kemampuan untuk menalar dengan baik, manusia dengan segudang pengalaman dan pengetahuan tidak akan dapat menyelesaikan masalah dengan baik. Demikian juga dengan kemampuan menalar yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik (Dahria,2008).

2.3. FSM (Finite State Machine)

Finite State Machines (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 (misal interupsi timer). Transisi keadaan ini umumnya juga disertai oleh aksi yang dilakukan oleh sistem 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 digunakan sebagai basis perancangan perangkat lunak pengendalian yang bersifat 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 kontrol, 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 prosedural seperti bahasa C, FSM ini umumnya direalisasikan dengan menggunakan statemen kontrol *switch case* atau/dan *if.. then*. Dengan menggunakan statemen-statement kontrol ini, aliran program secara praktis akan mudah dipahami dan dilacak jika terjadi kesalahan logika (Setiawan,2006).

2.4. Unity 3D

Unity Game Engine adalah software yang digunakan untuk membuat video Game berbasis dua atau tiga dimensi dan dapat digunakan secara gratis, selain untuk membuat Game, Unity 3D juga dapat digunakan untuk membuat konten yang interaktif lainnya seperti, visual arsitektur dan real-time 3D animasi, selain sebagai Game engine Unity 3D juga dapat digunakan sebagai sebuah editor bagi Game yang sudah ada.

Unity 3D dibuat dengan menggunakan bahasa program C++, tapi pengguna tidak perlu menggunakan bahasa C++ yang sulit, karena Unity 3D mendukung bahasa program lain seperti JavaScript, C#, dan Boo, Unity memiliki kemiripan dengan Game engine lainnya seperti, Blender Game engine, Virtools, Gamestudio, adapaun kelebihan dari Unity 3D, Unity dapat dioperasikan pada platform Windows dan Mac Os dan dapat menghasilkan Game untuk Windows, Mac, Linux, Wii, iPad, iPhone, google Android dan juga browser. Untuk browser, kita memerlukan sebuah plugin, yaitu Unity Web Player, sama halnya dengan Flash Player pada Browser. Game Unity 3D juga mendukung dalam pembuatan Game untuk console Game Xbox 360 dan PlayStation 3 (Ekasari, 2012).

3. METODE PELAKSANAAN

3.1. Diagram Blok Sistem



Gambar 3.1 Diagram Blok Sistem

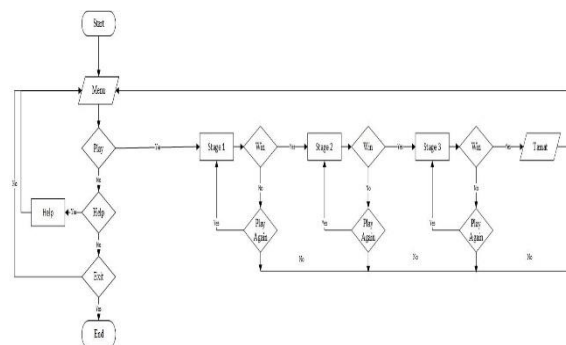
Penjelasan Blok Diagram Sistem:

1. Studi Literatur adalah mencari referensi teori yang relevan dengan penelitian yang dilakukan. Referensi yang digunakan terkait dengan pembuatan *game battle tank – when fight is the*

last coice dari berbagai buku maupun jurnal ilmiah.

2. Pada tahap konsep dan perancangan *game battle tank* terdiri dari beberapa proses yaitu proses analisis konsep dari pengumpulan ide untuk merancang jalannya pembuatan *game battle tank*. Sedangkan pada proses perancangan yaitu tahapan melakukan desain pada *game battle tank* yang akan dibuat. Pembuatan desain *game* didasarkan pada pengumpulan ide dan konsep yang telah dibuat sebelumnya.
3. Dalam pembuatan *game adventure battle tank* menerapkan teori yang sudah dipelajari dari studi literatur. Pembuatan *game adventure* ini menggunakan software Unity3D versi 5.3 dan Bahasa Pemrograman C#. *Game battle tank* ini berbasis desktop.
4. Uji coba *game* dimaksudkan untuk menghasilkan suatu *Game Battle Tank – When Fight Is The Last Coice* guna memberikan kesenangan dan hiburan kepada pemainnya.
5. Dari kegiatan ini diakhiri dengan proses pembuatan laporan. Laporan ini berisi tentang indikator-indikator dari tujuan apakah sudah terpenuhi ataupun belum. Selain itu terdapat dokumentasi dari kegiatan serta kesimpulan dari kegiatan yang telah dilakukan.

3.2. Flowchart Sistem



Gambar 3.2 Flowchart Sistem

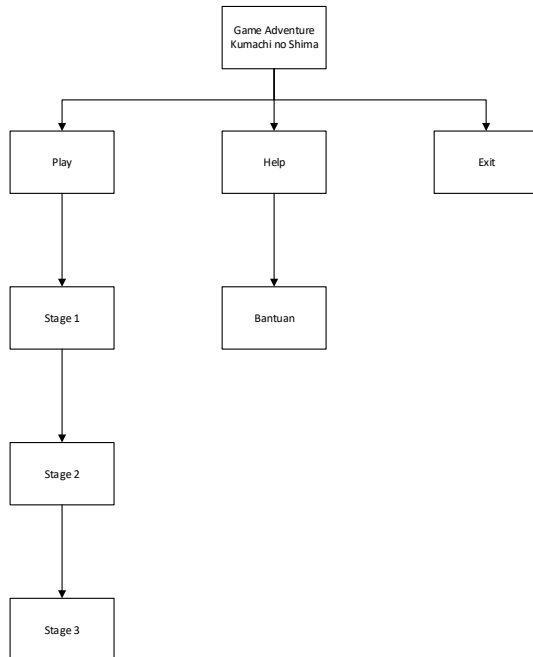
Berikut Penjelasan flowchart sistem dari gambar

- 3.2. 1. Game dimulai.
2. Muncul tampilan utama yang berupa pilihan beberapa menu.
3. Jika memilih menu Play, maka pemain akan langsung masuk kedalam game, lebih tepatnya masuk pada stage 1, dan dari stage 1 itu nantinya pemain harus mengalahkan semua musuh untuk maju ke stage berikutnya.

4. Jika memilih menu Help, maka pemain akan dihadapkan pada jendela bantuan tentang tata cara bermain game ini.
5. Jika memilih menu Exit, maka pemain akan keluar dari game.

3.3. Struktur Menu

Struktur menu pada aplikasi ini ditunjukkan pada Gambar 3.3.



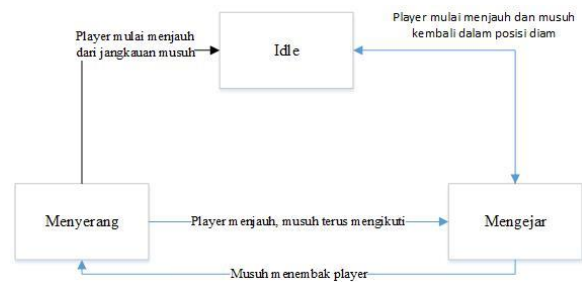
Gambar 3.3. Struktur Menu

Pada “main menu” terdiri dari beberapa pilihan:

1. *Play* berfungsi untuk memulai permainan.
 - a. *Stage 1* dengan permainan yang agak mudah.
 - b. *Stage 2* dengan sistem permainan dengan level yang lebih sulit, tapi masih dikategorikan sedang.
 - c. *Stage 3* dengan permainan yang levelnya paling sulit.
2. *Help* digunakan untuk memunculkan jendela bantuan.
3. *Exit* berfungsi untuk keluar dari game.

3.4. Alur Finite State Machine

Alur metode *finite state machine* yang terdapat pada musuh *game Battle Tank*, seperti yang ditunjukkan pada Gambar 3.4



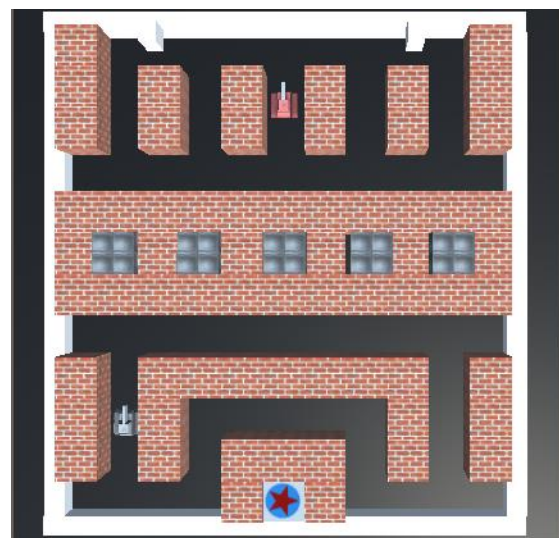
Gambar 3.4. Alur metode pada musuh

Pada Gambar 3.4 Jika jarak pemain diluar jangkauan musuh, musuh akan tetap diam. Jika pemain mulai memasuki jangkauan musuh, musuh akan mengejar, begitu juga ketika pemain semakin mendekat maka musuh akan mulai menembak.

4. HASIL DAN PEMBAHASAN

4.1. Tampilan Game

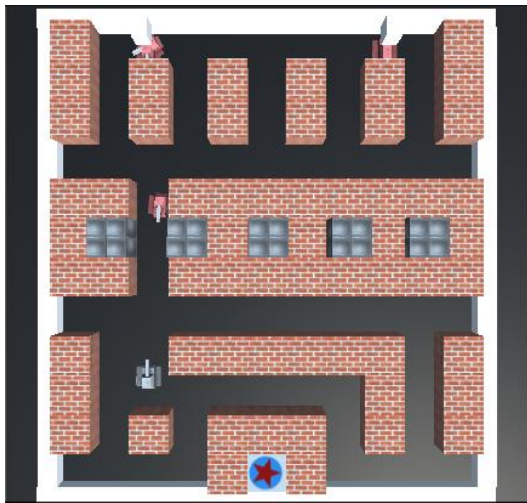
Tampilan *gameplay* dari *game battle tank*. Seperti yang ditunjukkan pada Gambar 4.1.1.



Gambar 4.1.1 Tampilan Game

4.2. Pengujian AI

Pengujian *AI* ditujukan untuk mengetes sistem *AI* apakah sudah berjalan dengan baik atau belum. Seperti yang ditunjukkan pada Gambar 4.2.1.



Gambar 4.2.1 Pengujian AI



Gambar 4.4.1 Halaman List Data

4.3. Pengujian Fitur Shoot

Pengujian yang ditujukan untuk mengecek apakah tank sudah bisa menembakan peluru atau belum beserta untuk melihat sejauh mana fitur *shoot* berfungsi. Seperti ditunjukkan pada Gambar 4.3.1.



Gambar 4.3.1 Pengujian Fitur Shoot

4.4. Pengujian GameOver dan Base

Pengujian GameOver adalah pengujian apakah game akan selesai jika pemain ditembak oleh musuh serta pengujian apakah fitur Base akan hancur setelah ditembak. Halaman List Data ditunjukkan pada Gambar 4.4.1.

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil dari perancangan dan implementasi pada *game adventure* Battle Tank maka diambil beberapa kesimpulan sebagai berikut.

1. *Game Battle Tank* memiliki 3 *stage*. Masing-masing mempunyai tingkat kesulitan berbeda. *Stage 1* mudah, *stage 2* normal dan *stage 3* sulit.
2. Implementasi *Finite State Machine* dapat diterapkan pada *game 2D bergenre adventure* dengan indikasi musuh dapat mengejar dan menyerang player dengan kondisi tertentu.
3. *Game Battle Tank* dapat dijalankan pada komputer dengan RAM 1GB hingga 8GB dan monitor beresolusi 1366 x 768 pixel, juga dengan menggunakan Windows XP, 7, 8, dan Win 10.

5.2. Saran

Saran yang dapat disampaikan ialah sebagai berikut.

1. Grafik karakter musuh, karakter pemain dan objek nya dibuat lebih bagus agar lebih menarik.
2. Penambahan *stage* pada game yang lebih banyak agar permainan bisa lebih lama untuk diselesaikan.
3. Menambahkan fitur *save game* agar pemain bisa melanjutkan *game* nya lagi ketika *game* dikeluarkan.

DAFTAR PUSTAKA

- [1] Barros, G.A., Liapis, A. and Togelius, J., 2015. Data adventures. In *Proceedings of the FDG Workshop on Procedural Content Generation* (pp. 2).
- [2] Norton, T., 2013. *Learning C# by Developing Games with Unity 3D*. Packt Publishing Ltd, 123-128.
- [3] Soon, G.K., Guan, T.T., On, C.K., Alfred, R. and Anthony, P., 2013, November. A comparison on the performance of crossover techniques in video game. In *Control System, Computing and Engineering (ICCSCE), 2013 IEEE International Conference on* (pp. 493-498). IEEE.
- [4] Zamroni Rosidi, Suryaman Nizar, Jalaluddin Ahmad. September 2013. Rancang Bangun Aplikasi Permainan Untuk Pembelajaran Anak Menggunakan HTML5. Volume 5, No.
- [5] Saarelainen, T. and Pakarinen, M., 2013. 2D Game Development With Unity 3D: Case study: Icemare, (pp. 8).
- [6] Creighton, R.H., 2010. *Unity 3D game development by example: A seat-of-your-pants manual for building fun, groovy little games quickly*. Packt Publishing Ltd, 6-7.