

**IMPLEMENTASI *FINITE STATE MACHINE* PADA *GAME*
“*SAVE THE ANIMALS*”**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV
Politeknik Negeri Malang

Oleh:

MOCHAMMAD IRFAN RIZALDI

NIM. 1341180087



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2017**

**IMPLEMENTASI *FINITE STATE MACHINE* PADA *GAME*
“*SAVE THE ANIMALS*”**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV
Politeknik Negeri Malang

Oleh:

MOCHAMMAD IRFAN RIZALDI

NIM. 1341180087



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2017**

HALAMAN PENGESAHAN

IMPLEMENTASI FINITE STATE MACHINE PADA *GAME*
“SAVE THE ANIMALS”

Disusun oleh:
MOCHAMMAD IRFAN RIZALDI NIM. 1341180087

Skripsi ini telah diuji pada tanggal 29 September 2017

Disetujui oleh:

- | | | | |
|------------------|---|---|-------|
| 1. Penguji I | : | <u>Dr. Eng. Rosa Andrie A., ST., MT.</u> | |
| | | NIP. 19801010 200501 1 001 | |
| 2. Penguji II | : | <u>Yan Watequlis Syaifudin, ST., MMT.</u> | |
| | | NIP. 19810105 200501 1 005 | |
| 3. Pembimbing I | : | <u>Ridwan Rismanto, S.ST., M.Kom.</u> | |
| | | NIP. 19860318 201212 1 001 | |
| 4. Pembimbing II | : | <u>Ekojono, ST., M.Kom.</u> | |
| | | NIP. 19591208 198503 1 004 | |

Mengetahui,

Ketua Jurusan
Teknologi Informasi

Ketua Program Studi
Teknik Informatika

Rudy Ariyanto, ST., MCs.
NIP. 19711110 199903 1 002

Ir. Deddy Kusbianto P., M.MKom.
NIP. 19621128 198811 1 001

PERNYATAAN

Dengan ini saya menyatakan bahwa Skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar Ahli Madya/kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Malang, 15 Agustus 2017

Mochammad Irfan Rizaldi

ABSTRAK

Rizaldi, Mochammad Irfan. “Implementasi *Finite State Machine* Pada Game “*Save The Animals*”. **Pembimbing: (1) Ridwan Rismanto., S.ST., M.Kom., (2) Ekojono, ST.,M.Kom.**

Skripsi, Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang, 2017.

Game merupakan salah satu industri yang besar saat ini dan dapat menjadi alternatif hiburan untuk mengisi liburan maupun waktu luang dan menjadi kegiatan yang sangat menyenangkan bagi anak – anak maupun orang dewasa. *Game* dapat dikembangkan menjadi salah satu media pembelajaran yang menyenangkan bagi anak. Salah satunya adalah yang bertema hewan yang bertujuan untuk memperkenalkan macam – macam hewan.

Salah satu pengembangan metode yang dapat mendukung pembuatan *game platformer* ini adalah *Finite State Machine Method* yang merupakan salah satu perangkat komputasi yang memiliki input berupa *string* dan *output* yang merupakan satu dari dua nilai yang dapat di-*accept* dan *reject*. Metode ini digunakan untuk mengatur perilaku pada objek hewan.

Hasil dari penelitian ini yaitu *Game “Save The Animals”* Berbasis *Desktop*, memiliki tampilan *gameplay* yang menarik dan membuat proses pengetahuan tentang pengenalan hewan dengan sangat menyenangkan sehingga *player* dapat mengetahui informasi tentang hewan dengan sangat jelas, Hal ini dapat dilihat dari penilaian setiap pengguna atau responden terhadap pertanyaan kuesioner yang menyatakan bahwa game ini menarik dan membuat proses pengetahuan tentang pengenalan hewan.

Kata Kunci: Pengenalan Hewan, Permainan 3D *Platformer, Desktop, Unity Game Engine, Finite State Machine Method.*

ABSTRACT

Rizaldi, Mochammad Irfan. *“The Implementation of Finite State Machine on Game "Save The Animals”.* **Advisors: (1) Ridwan Rismanto., S.ST., M.Kom., (2) Ekojono, ST.,M.Kom.**

Thesis, Informatics Engineering Study Program, Department of Information Technology, State Polytechnic of Malang, 2017.

Game is one of the big industries today and can be an alternative entertainment to spend holidays and leisure time and be a very fun activity for children and adults. Games can be developed into a fun learning media for children. One of them is a themed animal.

One of the development methods that can support the making of this game platformer is Finite State Machine Method, a computing device that has input and output which one of two things can be accepted and rejected. This method is used to regulate behavior on animal objects.

The results of this research is the Desktop based Game "Save The Animals" that has an interesting gameplay view and makes the process of knowledge about the introduction of animals so fun that players can find information about animals very clearly. This can be seen from the questionnaire given to respondents that state that this game is interesting and makes the process of knowledge about the introduction of animals well.

Keywords: *animal Introduction, 3D game platformer, Unity Game Engine, Finite State Machine.*

KATA PENGANTAR

Puji Syukur kami panjatkan kehadiran Allah AWT atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul “IMPLEMENTASI *FINITE STATE MACHINE* PADA GAME “*SAVE THE ANIMALS*”. Skripsi ini penulis susun sebagai persyaratan untuk menyelesaikan studi program Diploma IV Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang.

Kami menyadari tanpa adanya dukungan dan kerjasama dari berbagai pihak, kegiatan skripsi ini tidak dapat berjalan baik. Untuk itu, kami ingin menyampaikan rasa terima kasih kepada:

1. Allah SWT yang telah memberikan petunjuk dan hidayah-Nya dalam penyusunan Skripsi ini sehingga dapat berjalan dengan baik.
2. Kedua Orang tua dan Keluarga penulis yang telah banyak memberikan dukungan baik secara moral maupun material sehingga dalam penyusunan Skripsi ini dapat berjalan dengan lancar.
3. Bapak Rudy Ariyanto, ST., M.Cs., selaku ketua jurusan Teknologi Informasi.
4. Bapak Ir. Deddy Kusbianto P., M.MKom., selaku ketua program studi Teknik Informatika
5. Bapak Ridwan Rismanto., S.ST., M.Kom dan Ekojono, ST.,M.Kom selaku Dosen Pembimbing Skripsi yang telah memberikan waktu, dukungan dan bimbingan selama penyusunan Skripsi.
6. Dosen-dosen pengajar Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang yang telah memberikan bimbingan dan ilmunya.
7. Teman-teman Program Studi Teknik Informatika angkatan 2013 dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan Laporan Skripsi dari awal hingga akhir yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan skripsi ini, masih banyak terdapat kekurangan dan kelemahan yang dimiliki penulis baik itu sistematika penulisan maupun penggunaan bahasa. Untuk itu penulis mengharapkan saran dan kritik dari berbagai pihak yang bersifat membangun demi penyempurnaan laporan

ini. Semoga laporan ini berguna bagi pembaca secara umum dan penulis secara khusus. Akhir kata, penulis ucapkan banyak terima kasih.

Malang, 1 Juni 2017

Penulis

DAFTAR ISI

	Halaman
SAMPUL DEPAN	i
HALAMAN JUDUL.....	ii
HALAMAN PENGESAHAN.....	iii
PERNYATAAN.....	iv
ABSTRAK	v
<i>ABSTRACT</i>	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan.....	2
1.4 Batasan Masalah.....	2
1.5 Sistematika Penulisan.....	3
BAB II. LANDASAN TEORI.....	5
2.1 Finite State Machine.....	5
2.2 C# (C-Sharp)	6
2.3 Unity <i>Game Engine</i>	6
2.4 Blender 3D	8
2.5 Artifical Intelligence (AI).....	8
2.6 <i>Work Breakdown Structure</i>	9
2.7 Flowchart.....	10
2.8 Storyboard	12
2.9 <i>Use Case</i> Diagram.....	13
BAB III. METODOLOGI PENELITIAN	16
3.1 Metode Pengumpulan Data	16
3.1.1 <i>Survey Game</i> Sejenis.....	16
3.1.2 Studi Artikel dan Jurnal	16
3.2 Metodologi Pengembangan Sistem	16
3.2.1 Konsep (<i>Concept</i>).....	17

3.2.2	Desain/Perancangan (<i>Design</i>)	17
3.2.3	Pengumpulan Bahan (<i>Material Collecting</i>)	17
3.2.4	Pembuatan (<i>Assembly</i>)	17
3.2.5	Pengujian (<i>Testing</i>)	18
3.2.6	Distribusi (<i>Distribution</i>).....	18
3.3	Hasil Survey <i>Game</i> Sejenis	18
BAB IV. ANALISIS DAN PERANCANGAN		19
4.1	Analisis	19
4.1.1	Gambaran Umum Sistem	19
4.1.2	Analisis Kebutuhan	19
4.2	Perancangan.....	20
4.2.1	Perancangan Karakter	20
4.2.2	Perancangan Gameplay	20
4.2.3	Perancangan Flowchart <i>Gameplay</i>	21
4.2.4	Perancangan <i>Storyboard</i>	22
4.2.5	Perancangan Metode	24
4.2.6	Perancangan <i>Use Case</i>	25
BAB V. IMPLEMENTASI.....		28
5.1	Implementasi Pada <i>Photoshop</i>	28
5.1.1	Pembuatan Logo.....	28
5.1.2	Pembuatan Splash	29
5.1.3	Pembuatan Menu.....	29
5.1.4	Pembuatan <i>Frame</i>	30
5.2	Implementasi Pada Blender	30
5.2.1	Pembuatan Karakter Hewan.....	30
5.3	Implementasi Pada Unity	31
5.3.1	Pengaturan Menu	32
5.3.2	Pengaturan Kamera	34
5.3.3	Pengaturan <i>Player</i>	35
5.3.4	Pengaturan Hewan	35
5.3.5	Pengaturan Menu <i>Pause</i>	37
5.3.6	Pembuatan Peta Permainan	38
5.3.7	Implementasi Metode <i>Finite State Machine</i> pada <i>Game</i>	39
BAB VI. PENGUJIAN DAN PEMBAHASAN		44
6.1	Pengujian Sistem	44
6.1.1	Rencana Pengujian	44

6.1.2	Pengujian Alpha	45
6.1.3	Pengujian <i>Beta</i>	46
6.1.4	Hasil Pengujian Metode <i>Finite State Machine</i>	47
6.2	Pembahasan	48
6.2.1	Hasil Pengujian Alpha.....	48
6.2.2	Hasil Pengujian <i>Beta</i>	48
BAB VII. PENUTUP.....		51
7.1	Kesimpulan.....	51
7.2	Saran	51
DAFTAR PUSTAKA		52

DAFTAR GAMBAR

	Halaman
Gambar 2. 1 Diagram State FSM Kategori Musuh A.....	5
Gambar 2. 2 Diagram State FSM Kategori Musuh B.	5
Gambar 2. 3 Diagram State FSM Kategori Musuh C.	6
Gambar 2. 4 Simbol Flowchart	10
Gambar 2. 5 Simbol Proses Dalam Flowchart	12
Gambar 2. 6 Storyboard	13
Gambar 2. 7 Komponen Actor Pada Use Case Diagram	14
Gambar 2. 8 Komponen Proses Pada Use Case Diagram	14
Gambar 2. 9 Contoh Gambar Use Case Diagram	15
Gambar 3. 1 Perancangan Multimedia Development Life Cycle (MDLC)	16
Gambar 3. 2 Tampilan Game Pokemon Go.	18
Gambar 4. 1 Desain Karakter Hewan	20
Gambar 4. 2 Perancangan Flowchart Pada Gameplay	21
Gambar 4. 3 Use Case Diagram Pada Game "Save the Animals"	25
Gambar 5. 1 Pembuatan Logo Game	28
Gambar 5. 2 Pembuatan Splash	29
Gambar 5. 3 Pembuatan Background Untuk Menu Utama.....	29
Gambar 5. 4 Pembuatan Frame	30
Gambar 5. 5 Pembuatan Karakter Hewan.....	31
Gambar 5. 6 Tampilan Menu Unity v5.6.3f1	31
Gambar 5. 7 Tampilan Pengaturan Inspector Halaman Splash.....	32
Gambar 5. 8 Tampilan Pengaturan Inspector Menu	32
Gambar 5. 9 Tampilan Pengaturan Inspector Button Pada Menu.....	33
Gambar 5. 10 Tampilan Hasil dari Splash	33
Gambar 5. 11 Tampilan Hasil dari Menu.....	34
Gambar 5. 12 Tampilan Pengaturan Inspector Kamera	34
Gambar 5. 13 Tampilan Pengaturan Hasil dari Kamera	35
Gambar 5. 14 Tampilan Pengaturan Inspector Pada Pemain	35
Gambar 5. 15 Tampilan Animasi Hewan Saat Berjalan	36
Gambar 5. 16 Tampilan Animasi Hewan Saat Melompat	36
Gambar 5. 17 Tampilan Animasi Hewan Saat Berlari.....	37
Gambar 5. 18 Tampilan Inspector Pada Menu Pause	37
Gambar 5. 19 Tampilan Hasil Dari Panel Menu Pause	38
Gambar 5. 20 Tampilan Inspector Pada Pembuatan Map.....	38
Gambar 5. 21 Tampilan Hasil Dari Pembuatan Map	39
Gambar 5. 22 Tampilan Diagram FSM Pada Hewan	40
Gambar 6. 1 Saat Hewan Masuk Ke Dalam Radius	47
Gambar 6. 2 Pengujian Objek Pada Console	48

DAFTAR TABEL

	Halaman
Tabel 4. 1 Tabel Pembeda Hewan Pada Setiap Level.....	21
Tabel 4. 2 Storyboard Game "Save the Animals"	22
Tabel 4. 3 Rancangan alur Pada Finite State Machine.....	25
Tabel 4. 4 Use Case Play.....	26
Tabel 4. 5 Use Case Setting	26
Tabel 4. 6 Use Case Exit.....	27
Tabel 5. 1 Tabel Transisi Diagram State Throw	41
Tabel 6. 1 Daftar Rencana Pengujian.....	44
Tabel 6. 2 Daftar Pengujian Alpha Pada Game	45
Tabel 6. 3 Hasil Kuisisioner Penilaian Game	48
Tabel 6. 4 Hasil Pengujian Pertanyaan Pertama	49
Tabel 6. 5 Hasil Pengujian Pertanyaan Kedua	49
Tabel 6. 6 Hasil Pengujian Pertanyaan Ketiga.....	49
Tabel 6. 7 Hasil Pengujian Pertanyaan Keempat	50
Tabel 6. 8 Hasil Pengujian Pertanyaan Kelima.....	50

DAFTAR LAMPIRAN

Lampiran 1 Listing Program

Lampiran 2 Kuisisioner

Lampiran 3 Profil Penulis

Lampiran 4 Lembar Revisi Penguji I

Lampiran 5 Lembar Revisi Penguji II

Lampiran 6 Lembar Verifikasi Abstrak dan Tata Tulis

BAB I PENDAHULUAN

Pada bagian ini menjelaskan tentang latar belakang permasalahan dari skripsi yang akan menjadi rumusan masalah. Selanjutnya, ditentukan batasan-batasan masalah yang akan diselesaikan, tujuan penelitian, manfaat dari dilakukannya penelitian, dan sistematika penulisan buku laporan skripsi ini.

1.1 Latar Belakang

Game merupakan salah satu industri yang besar saat ini dan *Game* dapat menjadi alternatif hiburan untuk mengisi liburan maupun waktu luang dan menjadi kegiatan yang sangat menyenangkan bagi anak-anak maupun orang dewasa. Berbagai macam *game* yang berkembang pada saat ini berbasis Desktop/PC (Windows, Linux, Mac, dll), *Mobile* (Symbian, Java, Android, iOS, BlackBerry OS), dan berbasis Web (HTML5, Flash).

Adapun perkembangan *game* di dunia semakin pesat, tidak terkecuali di Indonesia. *Game* saat ini sudah menjadi sebagai media pembelajaran. Industri dan bisnis pengembangan *game* juga sudah menjadi suatu hal yang menjanjikan, terbukti dengan banyaknya perusahaan pengembang game di Amerika, Eropa dan Asia. Adanya konsol – konsol atau alat yang ditambahkan untuk menambahkan inovasi pada dunia game seperti *Kinect*, *Augmented Reality*, 2D, 3D, bahkan yang terbaru ini adalah *Virtual Reality*.

Game adventure dapat dikembangkan menjadi salah satu media pembelajaran yang menyenangkan bagi anak. *Game adventure* bertema hewan dapat menjadi media yang menarik untuk memperkenalkan macam-macam hewan sehingga anak dapat mengenal hewan melalui media *game* yang interaktif.

Game bergenre *adventure* kemunculan NPC/Non Playable Character selama ini masih *static* dengan menggunakan pola pengeluaran sehingga *game* masih terkesan monoton dan tidak dinamis tapi saat ini sudah banyak *game* yang telah menggunakan pola metode pengacakan untuk memunculkan posisi dari musuh agar tidak statik dan tidak monoton.

Game ini merupakan sebuah *game* berbasis desktop dengan genre *adventure*. Dengan menggunakan metode FSM (*Finite State Machine*) kita dapat memberikan kecerdasan pada NPC agar dapat memiliki kecerdasan yang sama dengan pemain

sehingga *game* menjadi lebih menantang dan karakter menjadi memiliki berbagai macam kondisi.

Berdasarkan hal di atas penulis tertarik untuk melakukan penelitian dengan mencoba membangun *game* dengan jenis “*Adventure*” pada *game* “*Save The Animals*”, dimana pada penelitian ini penulis mencoba untuk mengoptimalkan kinerja AI pada karakter hewan dengan menerapkan FSM (*Finite State Machine*). Ditambah desain tampilan yang diterapkan pada *game* nantinya dengan berbasis 3D.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, maka dapat dirumuskan beberapa permasalahan sebagai berikut :

- a) Bagaimana cara mengenalkan macam - macam hewan melalui media *game*?
- b) Bagaimana menerapkan metode *Finite State Machine* ke dalam sebuah *game* yang bergenre “*Adventure*”?

1.3 Tujuan

Tujuan dari pengembangan *game* “*Save the Animals*” ini adalah :

- a) Tujuan yang ingin dicapai dalam pembuatan *game* ini adalah memperkenalkan macam – macam hewan melalui media *game*.
- b) Dapat mengimplementasikan *Finite State Machine* ke dalam *game* untuk mengatur perilaku pada objek.

1.4 Batasan Masalah

Batasan masalah yang diangkat dalam proposal skripsi ini dapat dipaparkan sebagai berikut :

- a) *Game* hanya bisa dijalankan pada Personal Komputer atau berbasis *Desktop*.
- b) *Game* ini hanya dapat dimainkan oleh Satu Pemain atau *Single Player* (pemain tunggal).
- c) *Game* ini bersifat *offline* tidak terhubung ke jaringan internet.
- d) Objek binatang dalam *game* ini dibatasi pada jenis binatang jinak.
- e) *Game* ini berbasis 3D.

- f) *Game* ini bergenre “*Adventure*” yang mempunyai misi dan tujuan yang harus diselesaikan.
- g) *Game* ini hanya memiliki 3 level
- h) Menggunakan FSM (*Finite State Machine*) untuk merancang kondisi.
- i) *Software* pembangun aplikasi *game* ini menggunakan *Unity* v5.6.
- j) *Software* desain untuk aplikasi *game* ini menggunakan *Adobe Photoshop* dan *Blender 3D*.
- k) Bahasa pemrograman yang digunakan adalah *Cs script C#*.
- l) Sasaran usia pengguna 10 – 11 Tahun.

1.5 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini berisi penjelasan mengenai latar belakang mengapa melakukan penelitian ini, rumusan masalah, tujuan, serta batasan masalah dari penelitian ini.

BAB II LANDASAN TEORI

Bab ini berisi penjelasan mengenai algoritma, bahasa pemrograman, dan aplikasi yang digunakan untuk membangun *game* ini. Bab ini juga berisi penjelasan mengenai istilah-istilah asing yang digunakan dalam laporan ini.

BAB III METODOLOGI PENELITIAN

Bab ini berisi penjelasan mengenai cara pengumpulan data yang digunakan serta metode yang digunakan dalam membangun *game* ini.

BAB IV ANALISIS DAN PERANCANGAN

Bab ini berisi penjelasan mengenai analisa dan bagian-bagian yang perlu dipersiapkan terlebih dahulu sebelum melakukan implementasi, seperti *flowchart*, *storyboard*, serta desain.

BAB V IMPLEMENTASI

Bab ini berisi penjelasan mengenai implementasi pada aplikasi yang digunakan dalam pembuatan *game* ini, seperti pembuatan *sprites* yang akan digunakan dalam *game* serta implementasi metode pada *game*.

BAB VI PENGUJIAN DAN PEMBAHASAN

Bab ini berisi mengenai hasil pengujian dari *game* yang dikembangkan serta pembahasan dari algoritma yang digunakan.

BAB VII KESIMPULAN

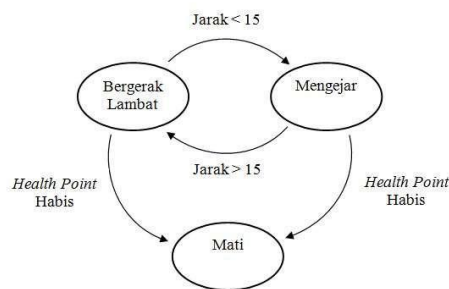
Bab ini berisi kesimpulan dari penelitian yang dilakukan serta saran yang bisa dilakukan apabila ingin mengembangkan penelitian ini.

BAB II. LANDASAN TEORI

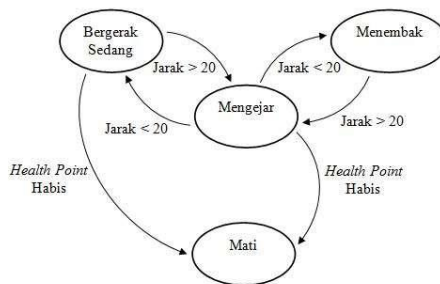
Pada bab ini berisi tentang teori-teori yang mendasari dan berkaitan dengan masalah perencanaan dan pembuatan *game* yang digunakan untuk memudahkan pemahaman serta pemecahan terhadap masalah yang ada.

2.1 Finite State Machine

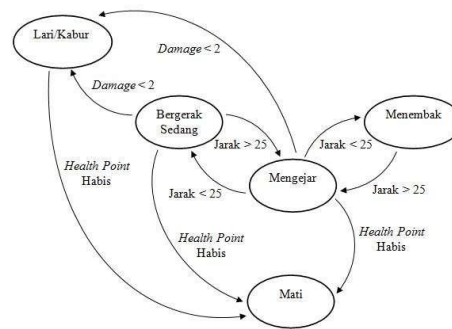
Metode yang akan digunakan adalah *Finite State Machine* (FSM). Selain untuk sistem kontrol, FSM adalah model yang umum digunakan untuk merancang perilaku agen cerdas di *game* yang mempunyai kelebihan pada kesederhanaan komputasinya dan kemudahan dalam pemahaman dan implementasinya. Penentuan diagram state sesuai dengan jenis perilaku yang sudah ditentukan, dengan kondisi-kondisi tertentu yang dapat memperlancar alur permainan. Gambar berikut adalah diagram state sesuai perancangan kategori musuh. [1]



Gambar 2. 1 Diagram State FSM Kategori Musuh A.



Gambar 2. 2 Diagram State FSM Kategori Musuh B.



Gambar 2. 3 Diagram State FSM Kategori Musuh C.

2.2 C# (C-Sharp)

C# atau yang dibaca C sharp adalah bahasa pemrograman sederhana yang digunakan untuk tujuan umum, dalam artian bahasa pemrograman ini dapat digunakan untuk berbagai fungsi misalnya untuk pemrograman server-side pada website, membangun aplikasi desktop ataupun *mobile*, pemrograman *game* dan sebagainya. Selain itu C# juga bahasa pemrograman yang berorientasi objek, jadi C# juga mengukung konsep objek seperti inheritance, class, polymorphism dan encapsulation.

Dalam prakteknya C# sangat bergantung dengan framework yang disebut .NET Framework, framework inilah yang nanti digunakan untuk mengcompile dan menjalankan kode C#. C# dikembangkan oleh Microsoft dengan merekrut Anders Helsing. Tujuan dibangunnya C# adalah sebagai bahasa pemrograman utama dalam lingkungan .NET Framework (lihat C#).

Banyak pihak juga yang menganggap bahwa Java dengan C# saling bersaing, bahkan ada juga yang menyatakan jika pernah belajar Java maka belajar C# akan sangat mudah dan begitu juga sebaliknya. Anggapan tersebut sebenarnya tidak salah karena perlu diketahui sebelum adanya C# Microsoft mengembangkan J++ dengan maksud mencoba membuat Java agar berjalan pada platform Windows, karena adanya masalah dari pihak luar maka Microsoft menghentikan proyek J++ dan beralih untuk mengembangkan bahasa baru yaitu C#. [9]

2.3 Unity Game Engine

Pada penelitian ini penulis memilih Unity 3D sebagai *game engine*. Unity 3D adalah sebuah *game engine* yang berbasis cross-platform. Unity dapat digunakan

untuk membuat sebuah *game* yang dapat digunakan pada perangkat komputer Windows dan Mac, iPhone, iPad dan bahkan Wii. Unity juga dapat digunakan untuk membuat web browser *game* yang memakai unity web player plug-in. Fitur scripting yang disediakan, mendukung 2 bahasa pemrograman, JavaScript dan C#. Unity 3D memiliki cerita yang menarik karena terus mengalami kemajuan sejak awal kemunculannya. Berawal saat Unity Technologies dibangun pada tahun 2004.

Kemudian pada tahun 2008, Unity melihat kebangkitan iPhone dan menjadi *game engine* pertama yang melakukan dukungan penuh pada platform tersebut. Pada tahun 2009, Unity mulai dapat diunduh secara gratis. Pada April 2012, Unity mencapai popularitas yang sangat tinggi dengan lebih dari 1 juta developer. Pada pembuatan *game*, terdapat beberapa istilah umum yang ada pada Unity, diantaranya:

- *Scenes*, merupakan istilah pada *Unity* yang digunakan untuk menyebutkan sebuah *level* atau *menu*.
- *Game Object*, merupakan sebuah object yang ada pada *Unity*. *Game object* memiliki beberapa komponen yang dapat ditambahkan atau dikurangi dengan cara *drag and drop*.
- *Components*, merupakan reaksi baru, bagi objek seperti *collison* memunculkan partikel, dan sebagainya.
- *Asset Store*, merupakan sebuah *website* yang disediakan oleh *Unity*, yang menyediakan *desain*, *plug in*, hingga *sounds* baik yang *open source* maupun berbayar.
- *Prefabs*, merupakan tempat untuk menyimpan satu jenis *game objects*, sehingga mudah untuk diperbanyak.
- *Collider*, berasal dari kata *collide* yang berarti bertabrakan. *Collider* memiliki fungsi untuk memudahkan *game developer* dalam melakukan test objek saat objek mengalami tabrakan atau sentuhan dengan objek lain.
- *Camera*
Camera merupakan salah satu fungsi dalam *Unity* yang memudahkan *game developer* membuat sudut pandang *user* pada *game* itu. Pada *Unity* dapat dengan mudah *developer* memindahkan posisi camera sehingga tidak terlalu sulit untuk membuat *game* bertipe *first person* atau *third person*.

Dalam *game engine* Unity terdapat berbagai macam bahasa pemrograman seperti javascript, C#, dan boo. Di Unity tidak dapat melakukan desain atau modeling, dikarenakan *game engine* ini dikembangkan bukan untuk mendesain asset-asset untuk *game*. Ada beberapa fitur yang terdapat pada Unity seperti *audio reverb zone*, *sky box*, *particle effect* dan masih banyak yang lain. Pada Unity juga dapat edit texture dari editor seperti photoshop, illustrator, dll.

Features (Scripting) di dalam Unity adalah sebagai berikut:

- Mendukung 3 bahasa pemrograman, JavaScript, C#, dan Boo.
- *Flexible and EasyMoving*, *rotating*, dan *objects* hanya perlu sebaris kode. Begitu juga dengan duplicating, removing, dan changing properties.
- *Multiplatform game* bisa di *deploy* di PC, Mac, Wii, iPhone, iPad, dan Browser, Android.
- *Visual Properties Variables* yang di definisikan dengan scripts ditampilkan pada editor. Bisa digeser, di *drag and drop*, bisa memilih warna dengan *color picker*.
- Berbasis .NET, penjalanan program dilakukan dengan *Open Source .NET platform, Mono*. [2]

2.4 Blender 3D

Blender adalah perangkat lunak sumber terbuka grafika komputer 3D. Perangkat lunak ini digunakan untuk membuat film animasi, efek visual, model cetak 3D, aplikasi 3D interaktif dan permainan video. Blender memiliki beberapa fitur termasuk pemodelan 3D, penteksturan, penyunting gambar bitmap, penulangan, simulasi cairan dan asap, simulasi partikel, animasi, penyunting video, pemahat digital, dan rendering. [6]

2.5 Artificial Intelligence (AI)

Artificial Intelligence atau biasa disebut kecerdasan buatan adalah suatu cara untuk membuat atau menciptakan sebuah mesin cerdas dengan menggunakan alat bantu, sehingga mesin cerdas itu dapat melakukan hal-hal yang biasa dilakukan oleh manusia. Ada beberapa kecerdasan buatan yang biasanya ditanamkan ke dalam sebuah mesin seperti pengenalan suara, pengenalan wajah, belajar, perencanaan,

pemecahan masalah, penalaran, persepsi, serta kemampuan untuk memanipulasi dan memindahkan objek. [5]

2.6 *Work Breakdown Structure*

WBS adalah suatu metode pengorganisasian proyek menjadi struktur pelaporan hierarkis. WBS digunakan untuk melakukan Breakdown atau memecahkan tiap proses pekerjaan menjadi lebih detail. Hal ini dimaksudkan agar proses perencanaan proyek memiliki tingkat yang lebih baik.

WBS disusun berdasarkan dasar pembelajaran seluuuh dokumen proyek yang meliputi kontrak, gambar-gambar, dan spesifikasi. Proyek kemudian diuraikan menjadi bagian-bagian dengan mengikuti pola struktur dan hirarki tertentu menjadi item-item pekerjaan yang cukup terperinci, yang disebut sebagai *Work Breakdown Structure*.

Pada dasarnya WBS merupakan suatu daftar yang bersifat top down dan secara hirarkis menerangkan komponen-komponen yang harus dibangun dan pekerjaan yang berkaitan dengannya.

Model WBS memberikan beberapa keuntungan, antara lain :

1. Memberikan daftar pekerjaan yang harus diselesaikan
2. Memberikan dasar untuk mengestimasi, mengalokasikan sumber daya, menyusun jadwal, dan menghitung biaya
3. Mendorong untuk mempertimbangkan secara lebih serius sebelum membangun suatu proyek .

Manfaat dari WBS :

1. Untuk mempercepat proses penyelesaian suatu proyek
2. Mengetahui pencapaian apa saja yang diinginkan suatu proyek
3. Dapat merencanakan proyek kedepannya

Tujuan dari WBS :

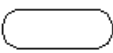








1. Melengkapi komunikasi antar personal proyek
2. Menjaga konsistensi dalam pengendalian dan pelaporan proyek
3. Cara efektif untuk melengkapi tugas manajemen

Setelah WBS berhasil disusun dan perkiraan lama waktu pelaksanaan telah dihitung, selanjutnya dilakukan penyusunan jadwal kerja. Pada dasarnya ada dua jenis model deskripsi penjadwalan, yaitu :

1. Bar Chart : Yang hanya menerangkan flow time dari setiap pekerjaan dan tanpa keterkaitan antar pekerjaan. Deskripsi ini paling baik digunakan pada presentasi
2. Network diagram : Yang menunjukkan keterkaitan antar tugas dan mengidentifikasi saat kritis pada jadwal. [9]

2.7 Flowchart

Flowchart adalah adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program. Berikut ini adalah beberapa simbol yang digunakan dalam menggambar suatu flowchart :

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan / akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/pemberian harga awal
	PROSES	Proses perhitungan/proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/proses menjalankan sub program
	DECISION	Pertandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

Gambar 2. 4 Simbol Flowchart

1. Flowchart Sistem (System Flowchart)

Flowchart Sistem merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Dengan kata lain, flowchart ini merupakan dekripsi secara grafik dari urutan prosedur-prosedur yang terkombinasi yang membentuk suatu sistem.

Flowchart Sistem terdiri dari data yang mengalir melalui sistem dan proses yang mentransformasikan data itu. Data dan proses dalam flowchart sistem dapat digambarkan secara online (dihubungkan langsung dengan komputer) atau offline (tidak dihubungkan langsung dengan komputer, misalnya mesin tik, cash register atau kalkulator).

2. Flowchart Dokumen (Document Flowchart)

Bagan alir dokumen (document flowchart) atau disebut juga bagan alir formulir (form flowchart) atau paperwork flowchart merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan di dalam bagan alir sistem.

3. Flowchart Skematik (Schematic Flowchart)

Bagan alir skematik (schematic flowchart) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Perbedaannya adalah, bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem, juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol bagan alir. Penggunaan gambar-gambar ini memudahkan untuk dipahami, tetapi sulit dan lama menggambarinya.


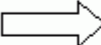



4. Flowchart Program (Program Flowchart)

Bagan alir program (program flowchart) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derivikasi bagan alir sistem.

Bagan alir program dapat terdiri dari dua macam, yaitu bagan alir logika program (program logic flowchart) dan bagan alir program komputer terinci (detailed computer program flowchart). Bagan alir logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Bagan alir logika program ini dipersiapkan oleh analis sistem.

5. *Flowchart Proses (Process Flowchart)*

Flowchart Proses merupakan teknik penggambaran rekayasa industrial yang memecah dan menganalisis langkah-langkah selanjutnya dalam suatu prosedur atau sistem. Bagan alir proses menggunakan lima buah simbol tersendiri seperti terlihat pada tabel di bawah ini.

Simbol	Keterangan
	Menunjukkan suatu operasi (operation)
	Menunjukkan suatu pemindahan (movement)
	Menunjukkan suatu simpanan (storage)
	Menunjukkan suatu inspeksi (inspection)
	Menunjukkan suatu penundaan (delay)

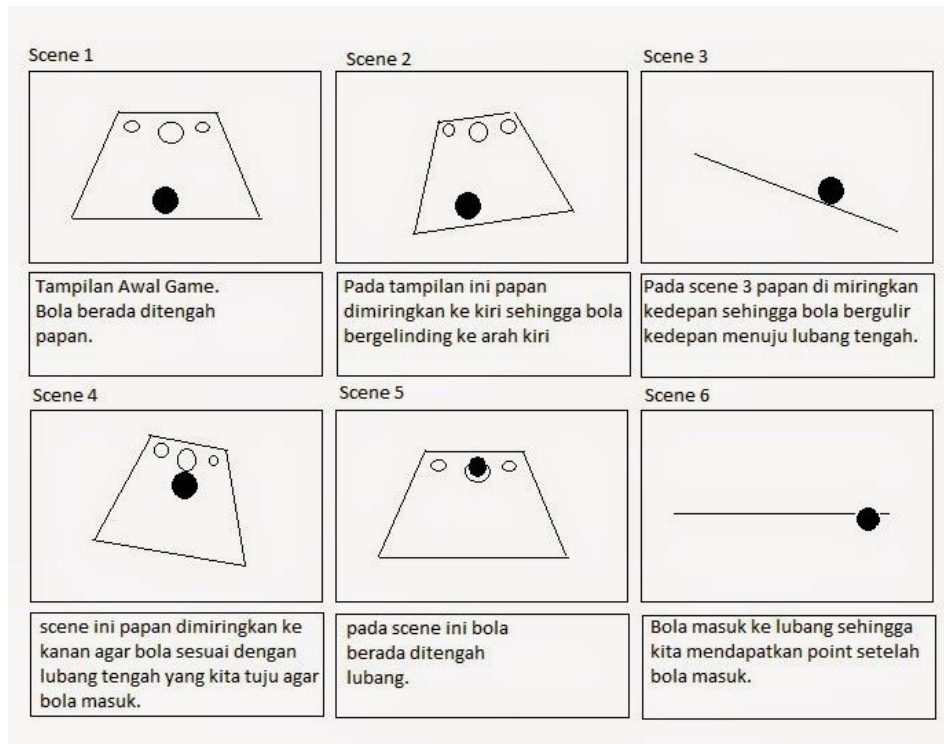
Gambar 2. 5 Simbol Proses Dalam *Flowchart*

Flowchart Proses digunakan oleh perekayasa industrial dalam mempelajari dan mengembangkan proses-proses manufacturing. Dalam analisis sistem, *flowchart* ini digunakan secara efektif untuk menelusuri alur suatu laporan atau *form*. [10]

2.8 Storyboard

Sebelum pembuatan *game*, maka dibutuhkan sebuah desain *storyboard* agar pembuatan *game* menjadi lebih mudah. *Storyboard* adalah visualisasi ide dari aplikasi yang akan dibangun, sehingga dapat memberikan gambaran dari aplikasi yang akan dihasilkan. *Storyboard* dapat dikatakan juga *visual script* yang akan

dijadikan outline dari sebuah proyek, ditampilkan *shot by shot* yang biasa disebut dengan istilah *scene*. *Storyboard* sekarang lebih banyak digunakan untuk membuat kerangka pembuatan website dan proyek media interaktif lainnya seperti iklan, film pendek, *games*, media pembelajaran interaktif ketika dalam tahap perancangan /desain. [7]

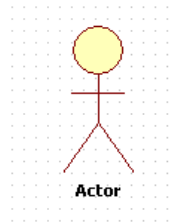


Gambar 2. 6 Storyboard

2.9 Use Case Diagram

Use Case merupakan sebuah teknik yang digunakan dalam pengembangan sebuah *software* atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan, *Use Case* menjelaskan interaksi yang terjadi antara ‘aktor’—inisiator dari interaksi sistem itu sendiri dengan sistem yang ada, sebuah *Use Case* direpresentasikan dengan urutan langkah yang sederhana. *Use Case* memiliki beberapa komponen yaitu : [8]

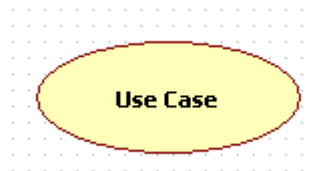
a. *Actor*



Gambar 2. 7 Komponen *Actor* Pada *Use Case* Diagram

Menggambarkan seseorang yang berinteraksi dengan sistem, di mana hanya bisa menginputkan informasi dan menerima informasi dari sistem dan tidak memegang kendali pada *Use Case*. Dan biasa actor di gambarkan dengan stickman.

b. Proses



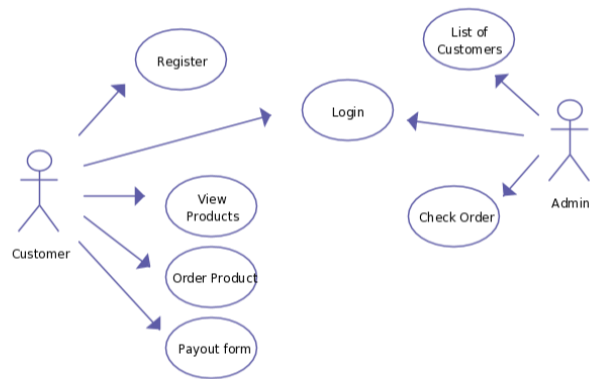
Gambar 2. 8 Komponen Proses Pada *Use Case* Diagram

Gambaran fungsional sistem yang akan di buat, agar pengguna lebih mengerti penggunaan system.

c. Relasi Dalam *Use Case*

Ada beberapa relasi yang terdapat pada *Use Case* Diagram:

1. **Association**, menghubungkan link antar element.
2. **Generalization**, sebuah elemen yang menjadi spesialisasi dari elemen yang lain.
3. **Dependency**, sebuah elemen yang bergantung beberapa cara kepada elemen lainnya.
4. **Aggregation**, bentuk association yang di mana sebuah elemen berisi elemen lainnya.



Gambar 2. 9 Contoh Gambar *Use Case* Diagram

BAB III. METODOLOGI PENELITIAN

Pada bab ini menjelaskan tentang metodologi dan perancangan karya dalam proses pembuatan *Game “Save The Animals”* dengan Berbasis *Desktop*.

3.1 Metode Pengumpulan Data

Tahap pengumpulan data yang digunakan dalam pembuatan *game* ini adalah melalui *survey game - game* yang sejenis dan studi literatur.

3.1.1 *Survey Game* Sejenis

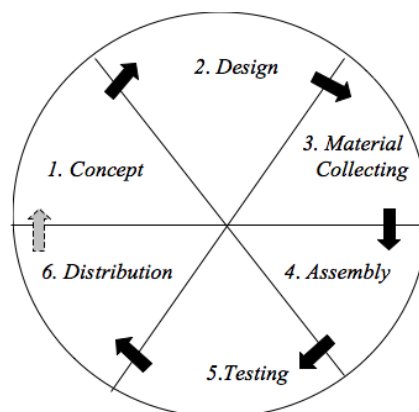
Mencoba *game - game* yang sejenis dengan genre *game* yang akan dibuat untuk memahami cara kerja dari *game* tersebut.

3.1.2 Studi Artikel dan Jurnal

Studi artikel dan jurnal merupakan cara menyelesaikan persoalan dengan menelusuri sumber-sumber tulisan yang pernah dibuat sebelumnya.

3.2 Metodologi Pengembangan Sistem

Dalam metode penelitian ini akan menjelaskan langkah-langkah yang dilakukan untuk merancang *game “Save the Animals”* dengan menggunakan metode *Multimedia Development Life Cycle (MDLC)* dan terdapat 6 sebagai berikut :



Gambar 3. 1 Perancangan *Multimedia Development Life Cycle (MDLC)*

3.2.1 Konsep (*Concept*)

Game Adventure ini berlatar belakang mencari dan mengkoleksi sebuah hewan, sehingga anak-anak mampu mengerti jenis - jenis hewan jinak serta terdapat informasi yang bermanfaat.

3.2.2 Desain/Perancangan (*Design*)

Desain adalah penjelasan untuk setiap unsur – unsur yang akan dibuat seperti desain sistem, alur permainan, serta peran apa saja yang dapat dilakukan oleh *user* atau pengguna.

3.2.3 Pengumpulan Bahan (*Material Collecting*)

Material Collecting atau pengumpulan data adalah tahap dimana pengumpulan bahan yang sesuai dengan kebutuhan dilakukan. Adapun bahan-bahan yang dibutuhkan adalah sebagai berikut :

- Teks : Jenis font yang digunakan pada *game*.
- Gambar : Semua gambar yang digunakan bertipe .jpg, png kemudian dijadikan objek, symbol, graphics, serta background pada *game*.
- *Audio* : File audio yang digunakan dalam *game* ini bertipe .ogg, .wav dan .mp3. Meliputi *background music* dan *sound fx*.
- *Animasi* : Objek animasi yang digunakan adalah animasi pada backgroud dan karakter yang ada pada *game*.
- *Interface* : Tampilan yang dibuat dan digunakan sebagai petunjuk navigasi pada keseluruhan *game*.
- Studi pustaka, jurnal-jurnal, e-book, serta buku-buku yang di perlukan sebagai referensi.

3.2.4 Pembuatan (*Assembly*)

Tahap *assembly* (pembuatan) adalah tahap dimana semua objek atau bahan multimedia dibuat. Pembuatan aplikasi berdasarkan *storyboard*, bagan alir (*flowchart*), dan struktur navigasi yang berasal pada tahap *design*. Proses ini dimulai dengan pemodelan karakter dan lingkungan *game*, pembuatan animasi pada karakter, dan pembuatan *game* serta *source code*. [3]

3.2.5 Pengujian (*Testing*)

Tahap *testing* (pengujian) dilakukan setelah menyelesaikan tahap pembuatan (*assembly*) dengan menjalankan aplikasi atau program dan dilihat apakah ada kesalahan atau tidak. Pengujian melibatkan pengguna akhir yaitu para pengguna atau *player*. [3]

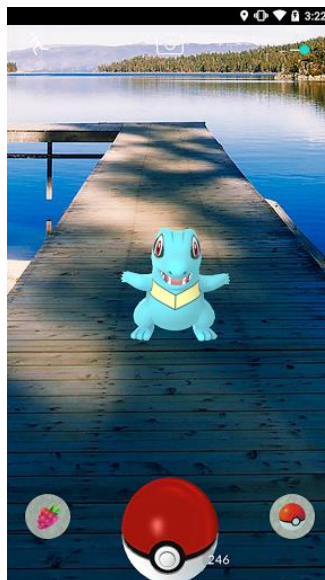
3.2.6 Distribusi (*Distribution*)

Tahap ini aplikasi akan disimpan dalam suatu media penyimpanan. Tahap ini juga dapat disebut tahap evaluasi untuk pengembangan produk yang sudah jadi supaya menjadi lebih baik. Hasil evaluasi ini dapat digunakan sebagai masukan untuk tahap concept pada produk selanjutnya. [4]

3.3 Hasil Survey *Game* Sejenis

Game yang di *survey* adalah *game* “*Pokemon Go*”. Setelah mencoba *game* “*Pokemon Go*” diketahui cara kerja dan aturan-aturan yang terdapat di dalam *game* tersebut. Cara kerja *game* tersebut adalah sebagai berikut :

- Saat melempar sebuah bola jika tidak terjangkau oleh pemain maka bola tidak akan mengenai sebuah objek atau karakter yang ingin di dapatkan.
- Karakter di dalam *game* yang bergerak secara acak. Dimana kemunculan sebuah objek atau karakter yang tidak dapat diketahui oleh pemain.



Gambar 3. 2 Tampilan *Game* *Pokemon Go*.

BAB IV. ANALISIS DAN PERANCANGAN

Pada bab ini, akan dibahas secara detail dan terperinci mengenai analisis dan perancangan *game* “*Save the Animals*” berbasis *Desktop* yang akan diimplementasikan dengan menerapkan kerangka konsep dan metode penelitian yang telah dijelaskan pada Bab III.

4.1 Analisis

Sebelum perancangan dilakukan analisis mengenai gambaran umum sistem yang akan dibangun.

4.1.1 Gambaran Umum Sistem

Sistem *video game* “*Save the Animals*” menggunakan metode *Finite State Machine*. *Finite State Machine* digunakan untuk memberikan perilaku pada sebuah objek karakter.

4.1.2 Analisis Kebutuhan

Analisis kebutuhan adalah sebuah proses untuk mendapatkan informasi, model, spesifikasi tentang kebutuhan perangkat lunak yang diinginkan klien/pengguna. Kebutuhan *software* yang digunakan dalam membangun *video game* “*Save the Animals*” adalah :

- Windows 10
- Blender 3D
- Adobe Photoshop CC
- UNITY *Game Engine* v.5.6.3f1

Kebutuhan *hardware* yang digunakan dalam membangun *video game* “*Save the Animals*” adalah :

- Processor Intel Core i5, 2.60 GHz
- RAM 4 GB
- *Mouse*
- *Keyboard*
- *Speaker*

Kebutuhan minimum *software* dan *hardware* pada *desktop* agar dapat memainkan *video game* “*Save the Animals*” adalah:

- Monitor minimal 800 x 600
- *Mouse*
- *Keyboard*
- *Space* HDD 2GB atau lebih
- *Speaker*

4.2 Perancangan

Ada beberapa bagian yang harus dirancang sebelum melakukan implementasi, seperti karakter, *gameplay*, *flowchart*. *usecase* perilaku npc, *flowchart* kondisi menang, serta *storyboard*.

4.2.1 Perancangan Karakter

Karakter dalam *game* ini hanya untuk pembuatan pada karakter hewan. Karakter NPC adalah hewan yang muncul secara acak. Berikut adalah desain karakter NPC. Berikut adalah contoh salah satu pembuatan karakter untuk hewan.



Gambar 4. 1 Desain Karakter Hewan

4.2.2 Perancangan Gameplay

Dalam *game* ini pemain akan mencari sebuah hewan jinak yang harus di jinakan lalu di koleksi selengkap – lengkapnya. Pada awal permainan pemain akan mencari sebuah makanan dan akan di berikan kepada hewan yang harus di

dapatkan pada tiap – tiap level sehingga hewan yang telah di berikan makanan akan menjadi jinak dan menjadi koleksi bagi pemain.

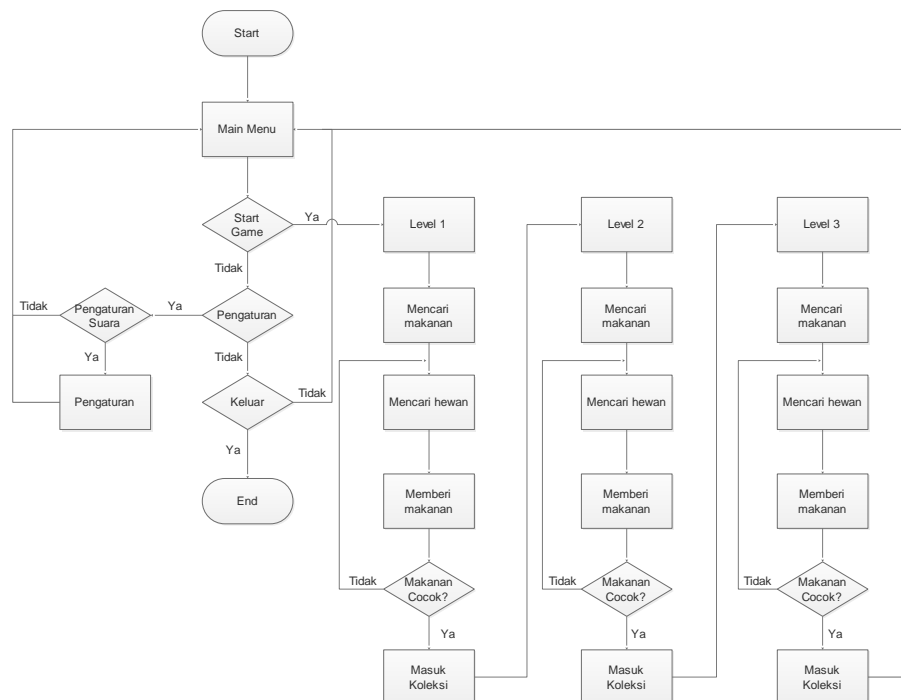
Jika menemukan hewan dan saat memberikan makanan tidak cocok maka hewan tersebut akan kabur/lari dan jika makanan itu cocok maka hewan tersebut akan mendekat dan masuk ke dalam koleksi.

Tabel 4. 1 Tabel Pembeda Hewan Pada Setiap Level

Level	Jumlah Hewan	Jenis Hewan	Jumlah Hewan Yang Harus Di Koleksi
Level 1	2	Ayam & Sapi	2
Level 2	4	Kambing, Kelinci, Babi & Zebra	4
Level 3	6	Ayam, Sapi, Kambing, Kelinci, Babi, & Zebra	6
Jumlah seluruh hewan yang telah di koleksi dari setiap level			12

4.2.3 Perancangan Flowchart *Gameplay*

Dalam game “*Save the Animals*” ada beberapa kondisi yang harus dipenuhi sebelum pemain dapat menyelesaikan pada setiap level tersebut dan menang.



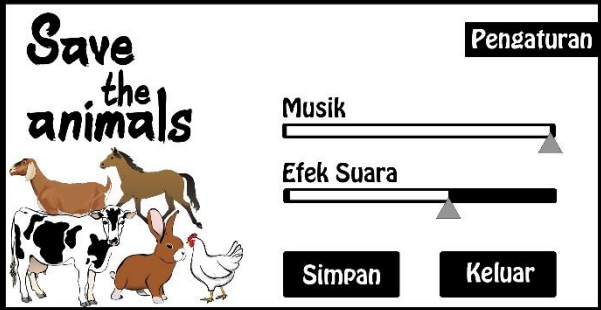




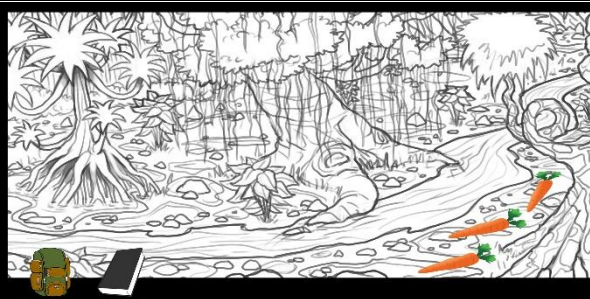
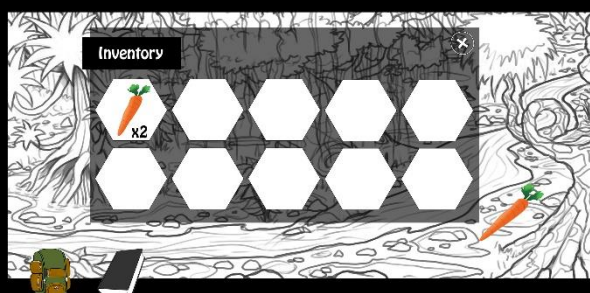

Gambar 4. 2 Perancangan Flowchart Pada *Gameplay*


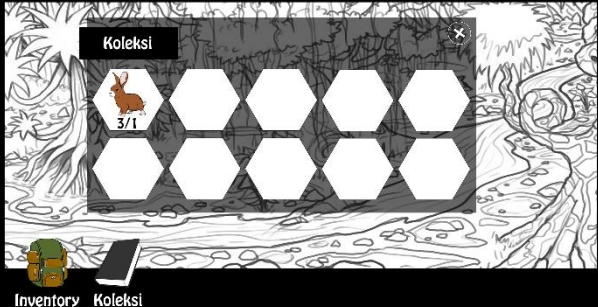
4.2.4 Perancangan *Storyboard*

Storyboard adalah visualisasi ide dari *game* yang akan dibangun sehingga dapat memberikan gambaran dari aplikasi yang akan dihasilkan. *Storyboard* dapat dikatakan juga *visual script* yang akan dijadikan outline dari sebuah proyek, ditampilkan *shot by shot* yang biasa disebut dengan istilah *scene*. *Storyboard* sekarang lebih banyak digunakan untuk membuat kerangka pembuatan website dan proyek media interaktif seperti iklan, film pendek, *games*, ketika dalam tahap perancangan. Berikut adalah perancangan *storyboard* dalam *game* “*Save the Animals*”.

Tabel 4. 2 *Storyboard Game "Save the Animals"*

No.	Gambar	Keterangan
1		Ini adalah tampilan awal dari <i>Game</i> “ <i>Save The Animals</i> ”
2		Ini adalah tampilan selanjutnya dari <i>Game</i> “ <i>Save The Animals</i> ”, pada main menu terdapat : <ul style="list-style-type: none"> • <i>Mulai</i> : Menu Memulai permainan • <i>Pengaturan</i> : Menu pengaturan musik dan efek suara • <i>Keluar</i> : Menu Keluar Dari <i>Game</i>
3		Ini adalah tampilan dalam menu pengaturan disana terdapat pengaturan untuk musik dan efek suara

4		Ini adalah tampilan level saat pemain memulai permainan
5		Ini adalah tampilan informasi tentang hewan yang ada dalam game
6		Ini adalah tampilan pemandangan yang ada dalam game
7		Ini adalah tampilan saat makanan masuk ke dalam sebuah inventori
8		Ini adalah tampilan saat pemain menemukan hewan yang akan di koleksi

9		Ini adalah tampilan saat pemain memberikan makanan kepada hewan yang akan di koleksi
10		Ini adalah tampilan hewan yang telah di koleksi

4.2.5 Perancangan Metode

Pada aplikasi ini menggunakan metode FSM diimplementasikan pada rangkaian tahap atau state dengan masukan *event* dan *action* sehingga menghasilkan *output* yang diharapkan sesuai kondisi yang ada. Dari kondisi – kondisi tersebut maka membentuk suatu pola *state* yang saling berhubungan. Perpindahan *state* dapat terjadi apabila mendapat masukan berupa *event* tertentu yang disertai aksi yang dilakukan oleh sistem. *Output state* sebelumnya menjadi inputan untuk *state* berikutnya.

Dalam aplikasi ini metode FSM diterapkan pada hewan dalam permainan. Penerapan FSM pada aplikasi ini dilakukan agar *user* dapat mengenali perilaku hewan dalam sistem. Rancangan alur state FSM “*Save The Animals*” dibagi menjadi beberapa bagian dengan mencantumkan kebutuhan permainan dan aksi yang perlu dilakukan oleh pengguna untuk dapat mengenali perilaku hewan dengan baik. Diantaranya perilaku *state* (Q), *action* (a) dan *event* (e). Penentuan perilaku hewan tersebut diterapkan dengan menggunakan kecerdasan buatan FSM. Berikut merupakan penerapan FSM untuk kondisi hewan pada *state game* :

Tabel 4. 3 Rancangan alur Pada *Finite State Machine*

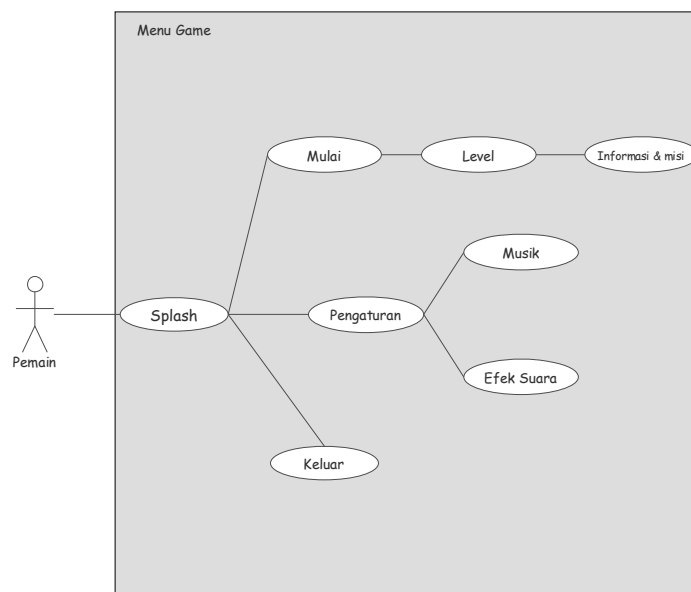
<i>State (Q)</i>	<i>Event (e)</i>	<i>Action (a)</i>	<i>Next State</i>
<i>Idle</i>	Jika tidak ada kondisi yang memenuhi	<i>Looping</i>	<i>Idle</i>
<i>Idle</i>	Jika jarak pemain mendekati makanan	Perilaku pemain <i>Grab</i>	<i>Idle</i>
<i>Idle</i>	Jika jarak antara makanan dan hewan tidak dalam radius	Perilaku objek hewan <i>Idle</i>	<i>Idle</i>
<i>Idle</i>	Jika jarak antara makanan dan hewan masuk ke dalam radius	Perilaku objek hewan <i>Eat</i>	<i>Stop</i>
<i>Idle</i>	Jika makanan tidak cocok	Perilaku objek hewan <i>Run</i>	<i>Run</i>
<i>Stop</i>	Makanan berkurang dan masuk ke dalam buku koleksi	Perilaku objek hewan <i>Stop</i>	<i>Stop</i>

4.2.6 Perancangan *Use Case*

Berikut adalah perancangan dari *use case* untuk game dari “*Save the Animals*”. Hak akses yang dapat dilakukan oleh pemain :

1. Menu *Play*, pemain dapat langsung bermain game *Save The Animals*.
2. Menu *Setting*, pemain dapat *on/off Back sound* musik
3. Menu *Exit*, pemain dapat keluar dari game.

Berikut merupakan *use case diagram* dari game “*Save the Animals*”:

Gambar 4. 3 *Use Case Diagram* Pada Game "*Save the Animals*"

4.2.6.1 Deskripsi *Use Case*Tabel 4. 4 *Use Case Play*

<i>Use case id</i>	1
<i>Use case name</i>	<i>Play</i>
<i>Created by</i>	
<i>Actor</i>	Pemain
<i>Description</i>	<i>Use case ini berfungsi untuk memulai permainan</i>
<i>Trigger</i>	Pemain klik tombol <i>play</i>
<i>Pre-Condition</i>	Pemain memulai permainan
<i>Exception</i>	
<i>Includes</i>	
<i>Priority</i>	<i>High</i>
<i>Frequency of use</i>	<i>High</i>
<i>Bussiness rule</i>	
<i>Special requirement</i>	
<i>Assumption</i>	
<i>Note issues</i>	

Tabel 4. 5 *Use Case Setting*

<i>Use case id</i>	2
<i>Use case name</i>	<i>Setting</i>
<i>Created by</i>	
<i>Actor</i>	Pemain
<i>Description</i>	<i>Use case ini berfungsi untuk on/off Back sound musik game</i>
<i>Trigger</i>	Pemain klik tombol pengaturan
<i>Pre-Condition</i>	Pemain memilih klik musik
<i>Post-Condition</i>	<i>On/off musik</i>
<i>Normal Flow</i>	1. Pemain klik tombol <i>setting</i> 2. Pemain memilih <i>on/off</i> musik
<i>Exception</i>	
<i>Includes</i>	
<i>Priority</i>	<i>Low</i>
<i>Frequency of use</i>	<i>Low</i>
<i>Bussiness rule</i>	
<i>Special requirement</i>	
<i>Assumption</i>	
<i>Note issues</i>	

Tabel 4. 6 *Use Case Exit*

<i>Use case id</i>	3
<i>Use case name</i>	Exit
<i>Created by</i>	
<i>Actor</i>	Pemain
<i>Description</i>	<i>Use case</i> ini berfungsi untuk keluar dari aplikasi
<i>Trigger</i>	Pemain klik tombol <i>exit</i>
<i>Pre-Condition</i>	Pemain keluar dari aplikasi
<i>Post-Condition</i>	keluar dari aplikasi
<i>Normal Flow</i>	1. Pemain klik tombol keluar 2. Keluar dari aplikasi
<i>Exception</i>	
<i>Includes</i>	
<i>Priority</i>	<i>Low</i>
<i>Frequency of use</i>	<i>Low</i>
<i>Bussiness rule</i>	
<i>Special requirement</i>	
<i>Assumption</i>	
<i>Note issues</i>	

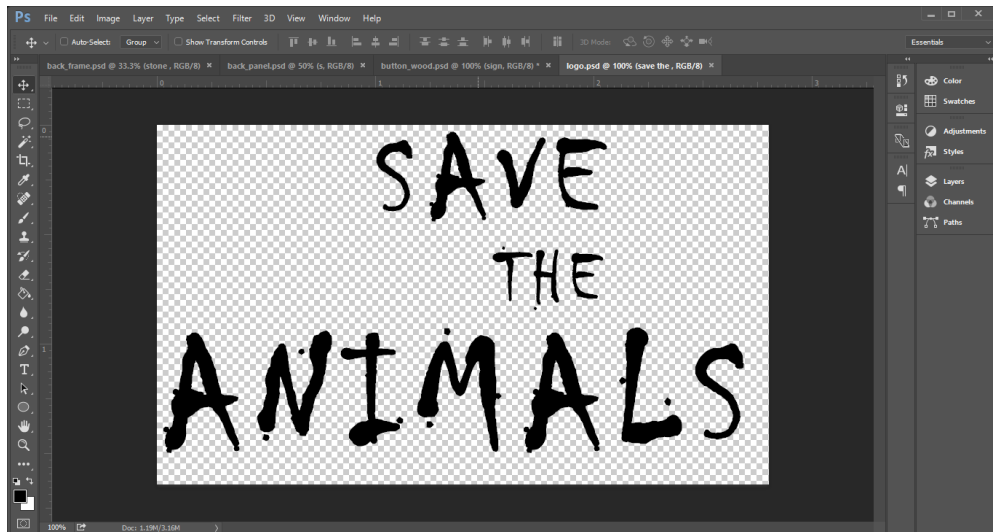
BAB V. IMPLEMENTASI

Setelah melakukan tahapan perancangan aplikasi, maka pada bab ini akan dibahas tentang proses implementasi, yaitu realisasi perancangan menjadi nyata. Bagian pada bab ini meliputi implementasi pembuatan *asset* gambar, implementasi sistem, dan implementasi metode.

5.1 Implementasi Pada *Photoshop*

Berikut ini merupakan langkah-langkah pembuatan *asset* gambar yang terdapat pada *game* menggunakan *Software Photoshop*:

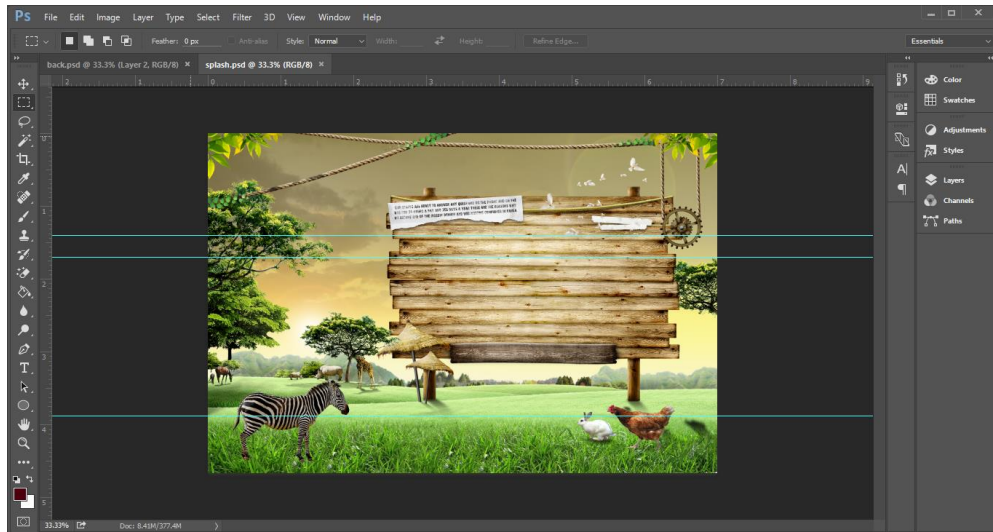
5.1.1 Pembuatan Logo



Gambar 5. 1 Pembuatan Logo *Game*

Pada gambar 5. 1 pembuatan logo *game* menggunakan *software photoshop* dan memilih jenis *font* yang akan di gunakan untuk membuat *text* yang telah ditentukan.

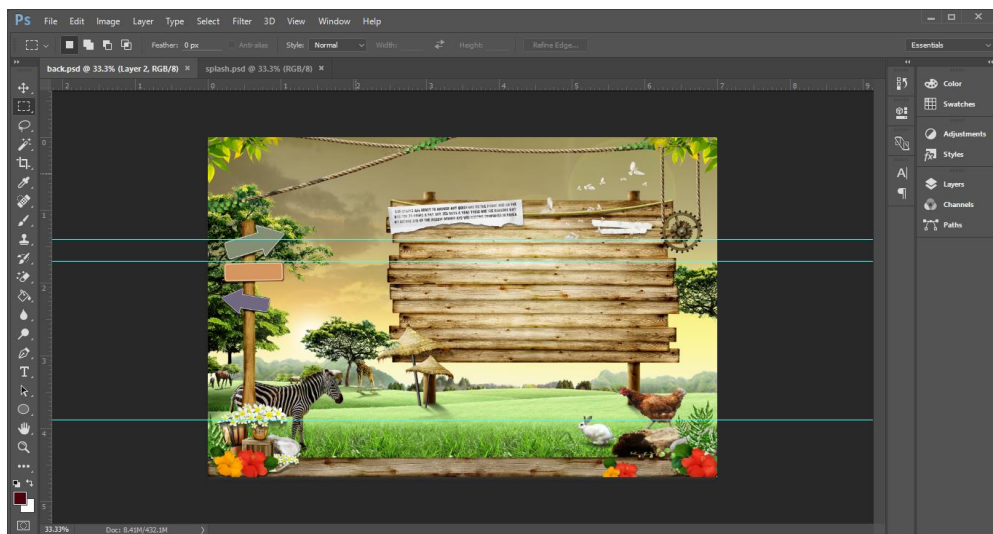
5.1.2 Pembuatan Splash



Gambar 5. 2 Pembuatan *Splash*

Pada gambar 5. 2. merupakan tampilan utama saat *game* dibuka serta terdapat tombol *start game* untuk masuk ke dalam *scene* selanjutnya.

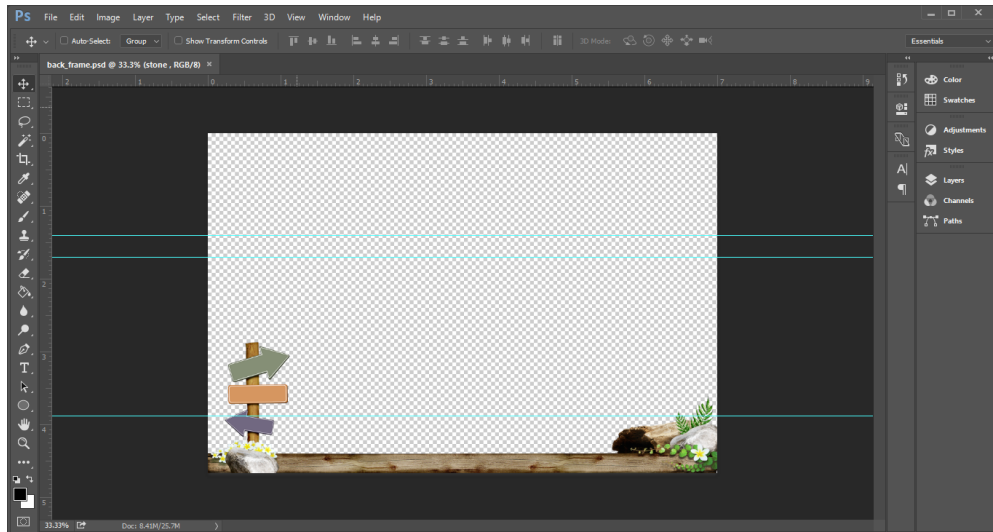
5.1.3 Pembuatan Menu



Gambar 5. 3 Pembuatan *Background* Untuk Menu Utama

Pada gambar 5. 3 terdapat pembuatan dari menu utama yang terdapat logo, tombol mulai, tombol pengaturan, dan tombol keluar.

5.1.4 Pembuatan *Frame*



Gambar 5. 4 Pembuatan *Frame*

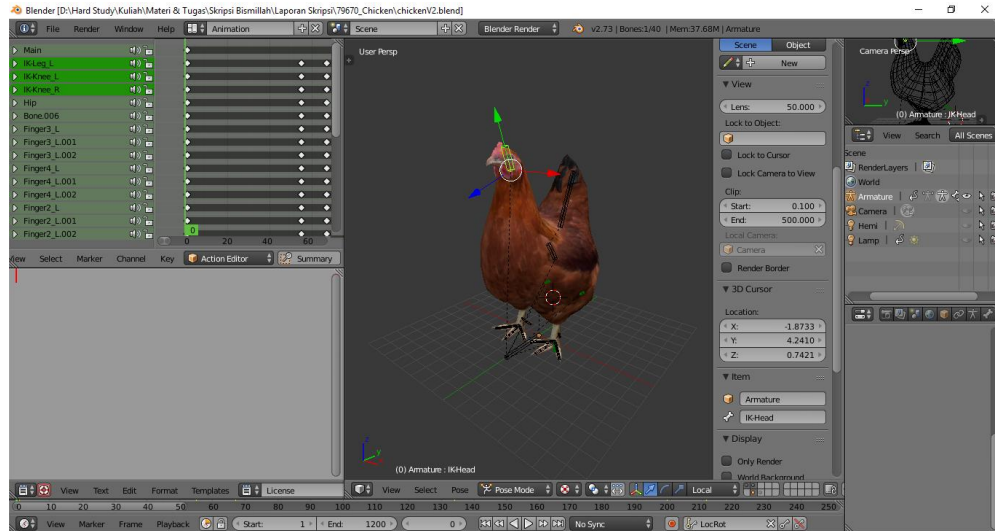
Pada gambar 5. 4 *frame* disini digunakan untuk membuat tombol bagian kiri adalah tombol koleksi, tombol inventori, dan tombol keluar. Untuk bagian bawah terdapat skor dan waktu.

5.2 Implementasi Pada Blender

Berikut ini merupakan langkah-langkah pembuatan objek hewan 3D dengan menggunakan *Software Blender 3D*:

5.2.1 Pembuatan Karakter Hewan

Berikut adalah salah satu contoh untuk pembuatan objek hewan pada *Software Blender* yang ditunjukkan pada gambar 5. 5.

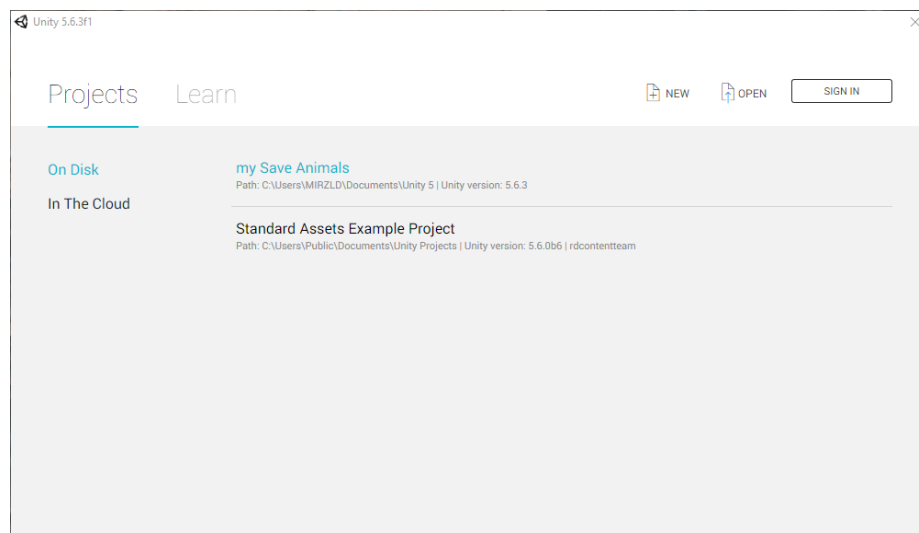


Gambar 5. 5 Pembuatan Karakter Hewan

Pada gambar 5. 5 untuk pembuatan objek hewan pengguna menggunakan *software blender* beserta animasi yang akan diterapkan pada *game*

5.3 Implementasi Pada Unity

Berikut ini merupakan langkah-langkah implementasi untuk pembuatan *game* pada *Unity Game Engine* yang ditunjukkan pada gambar 5. 6:



Gambar 5. 6 Tampilan Menu *Unity v5.6.3f1*

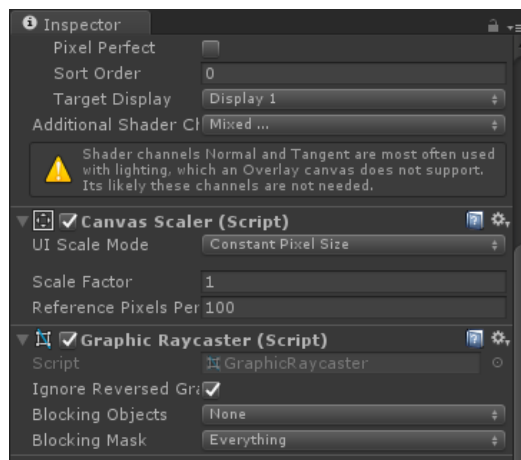
Pada gambar 5. 6 menjelaskan tampilan awal *unity* untuk membuat suatu proyek baru atau yang sudah pernah dibuat.

5.3.1 Pengaturan Menu

Menu utama memiliki 5 tombol yang masing-masing dari tombol berfungsi sebagai berikut:

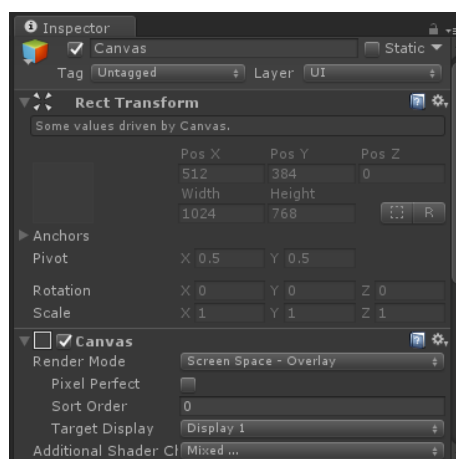
- Tombol Mulai : Untuk memulai permainan.
- Tombol Pengaturan : Berisi pengaturan *background* musik
- Tombol Keluar : Untuk keluar dari aplikasi *game*.

Untuk dapat membuat menu utama berjalan dengan yang diinginkan, maka diperlukan pengaturan pada unity sebagai berikut:



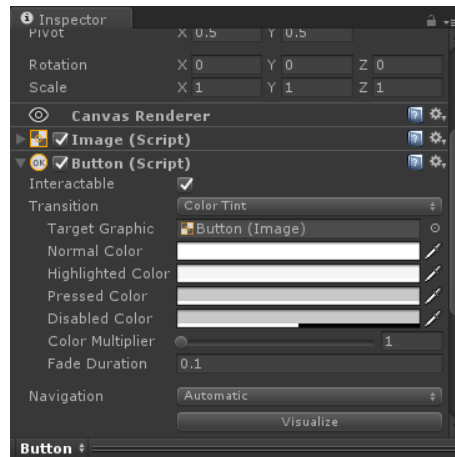
Gambar 5. 7 Tampilan Pengaturan *Inspector* Halaman Splash

Pada gambar 5. 7 membuat tampilan *Inspector* untuk bagian *scene splash* pada game “*Save the Animals*” agar *game* menjadi lebih menarik yang ditunjukkan.



Gambar 5. 8 Tampilan Pengaturan *Inspector* Menu

Pada gambar 5. 8 dibuat pengaturan tampilan *Inspector* untuk *scene menu* pada game “*Save the Animals*” sebuah pengaturan untuk menu yang dapat digunakan oleh pemain.



Gambar 5. 9 Tampilan Pengaturan *Inspector* Button Pada Menu

Pada gambar 5. 9 membuat animasi pada setting, agar hasil sesuai dengan apa yang diinginkan



Gambar 5. 10 Tampilan Hasil dari *Splash*

Berikut hasil dari pembuatan tombol start game pada *scene splash* tombol tersebut dapat menuju ke *scene* selanjutnya yang ditunjukkan pada gambar 5. 11.

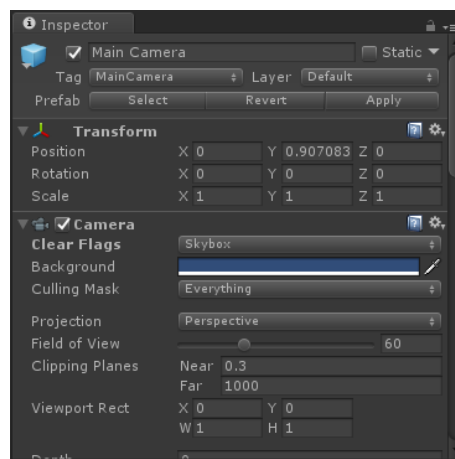


Gambar 5. 11 Tampilan Hasil dari Menu

Pada gambar 5. 11 menampilkan *scene* menu utama yang terdapat 3 tombol yaitu mulai, pengaturan dan keluar yang ditunjukkan

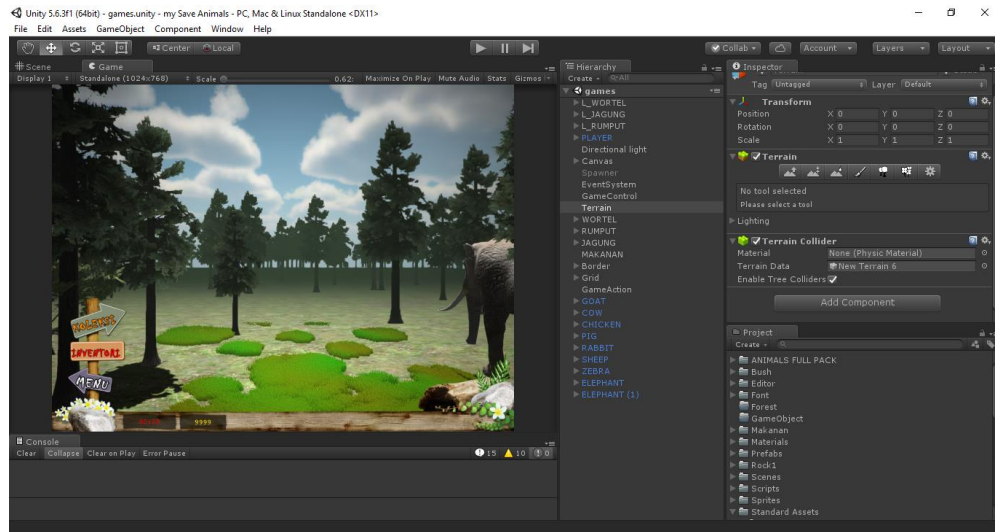
5.3.2 Pengaturan Kamera

Kamera yang akan di gunakan adalah *First Person Controller* kamera akan selalu mengikuti kemanapun *player* tersebut pergi. Agar dapat berjalan sesuai dengan yang diinginkan, maka diperlukan perngaturan sebagai berikut yang ditunjukkan pada gambar 5. 11 :



Gambar 5. 12 Tampilan Pengaturan *Inspector* Kamera

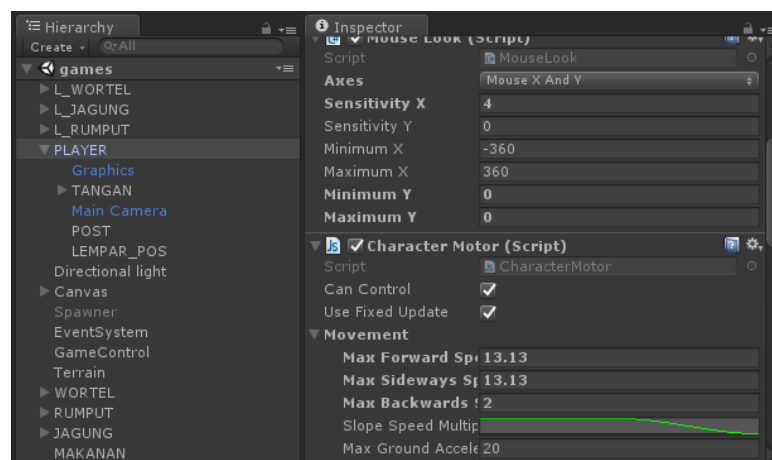
The Player pada *camera Script* digunakan agar agar kamera mengikuti letak *player*, seperti yang di tunjukan pada gambar 5.13.



Gambar 5. 13 Tampilan Pengaturan Hasil dari Kamera

5.3.3 Pengaturan *Player*

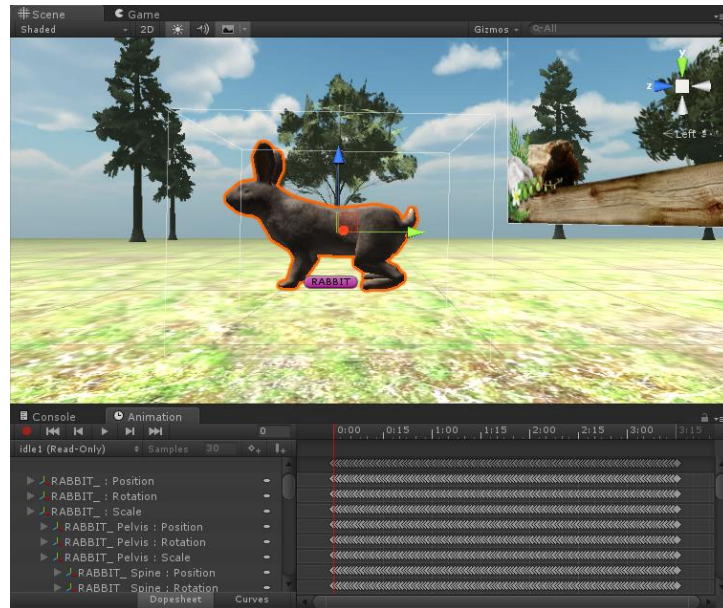
Player memiliki animasi yang digunakan pada *player* hanya meliputi *idle* agar jalannya *player* sesuai dengan yang diinginkan, maka diperlukan pengaturan pada unity sebagai berikut:



Gambar 5. 14 Tampilan Pengaturan Inspector Pada Pemain

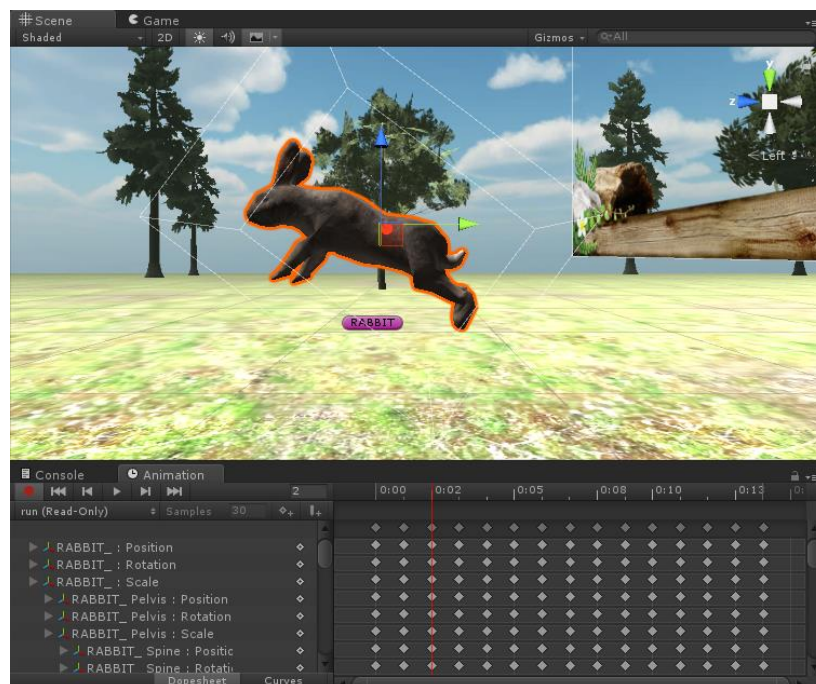
5.3.4 Pengaturan Hewan

Animasi yang dibutuhkan oleh Hewan yaitu *idle*, berlari, melompat dan makan. Berikut *Animation* yang ada pada hewan :



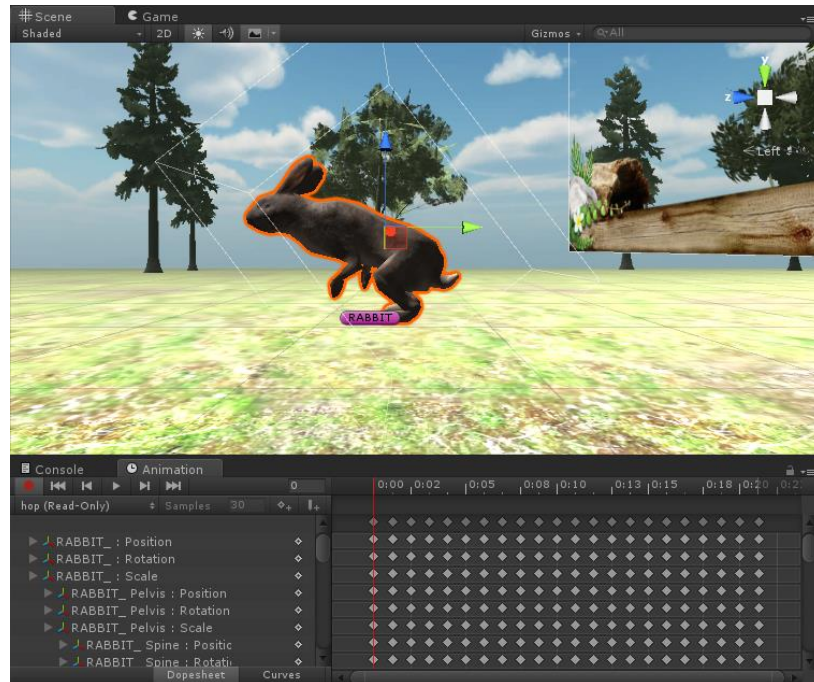
Gambar 5. 15 Tampilan Animasi Hewan Saat Berjalan

Berikut adalah animasi pada hewan kelinci saat berjalan yang ditunjukkan pada gambar 5. 15.



Gambar 5. 16 Tampilan Animasi Hewan Saat Melompat

Berikut adalah animasi pada hewan kelinci saat melompat yang ditunjukkan pada gambar 5. 16.



Gambar 5. 17 Tampilan Animasi Hewan Saat Berlari

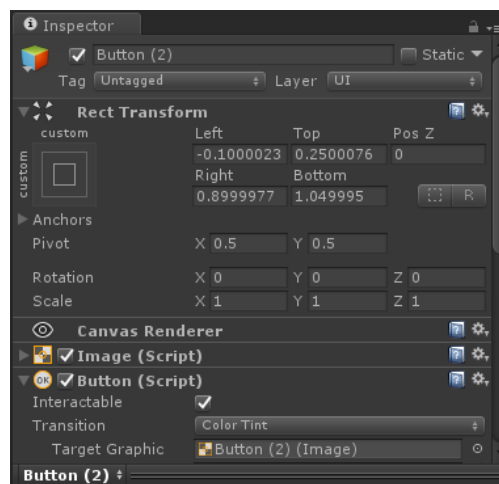
Berikut adalah animasi pada hewan kelinci berlari yang ditunjukkan pada gambar 5. 15

5.3.5 Pengaturan Menu *Pause*

Apabila user menekan tombol pause yang terletak di kanan atas, maka *game* akan berhenti sementara, kemudian panel pause akan keluar. Ada beberapa menu pada panel pause yaitu :

1. Tombol Lanjut : Untuk *player* melanjutkan permainan
2. Tombol Akhiri : Untuk *player* kembali ke menu awal

Berikut adalah pengaturan inspector *pause menu* pada unity :



Gambar 5. 18 Tampilan *Inspector* Pada Menu Pause

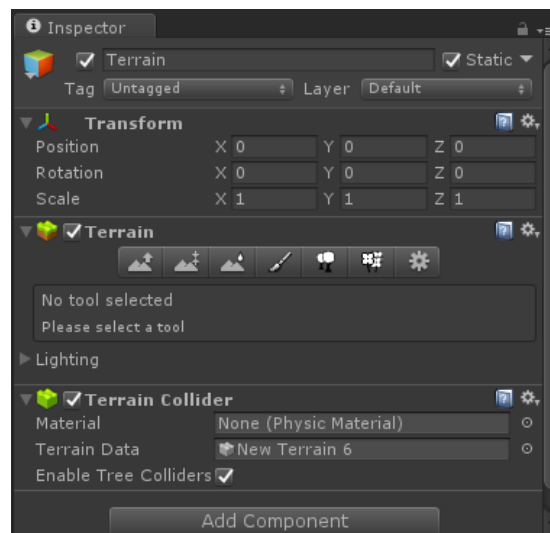


Gambar 5. 19 Tampilan Hasil Dari Panel Menu Pause

Pada gambar 5. 19 *Player* dapat melakukan *pause* saat permainan sedang dimulai dan Apabila *player* kehabisan waktu maka *game* akan berhenti dengan otomatis.

5.3.6 Pembuatan Peta Permainan

Map atau arena adalah istilah umum yang merujuk kepada suatu tempat yang digunakan untuk *player* melakukan kegiatan bermain dalam sebuah *game*.



Gambar 5. 20 Tampilan *Inspector* Pada Pembuatan *Map*

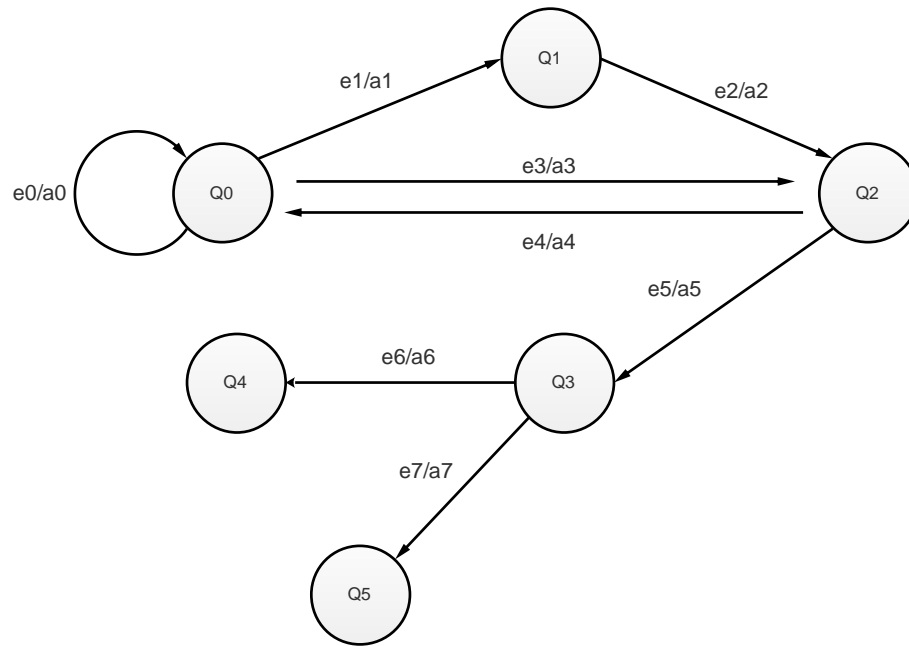


Gambar 5. 21 Tampilan Hasil Dari Pembuatan *Map*

5.3.7 Implementasi Metode *Finite State Machine* pada *Game*

a. Diagram State Hewan

Gambar 5. 22 berikut merupakan diagram *state* permainan “*Save The Animals*” dengan penerapan kecerdasan buatan *Finite State Machine* (FSM). Pada diagram *state* ini menjelaskan alur yang terjadi saat hewan memakan sebuah makanan yang berada dalam permainan. *State* dalam diagram ini meliputi ketentuan yang berlaku dan kegiatan yang dapat terjadi yang dipengaruhi oleh *state* (*Q*) yang telah dijabarkan pada tabel perancangan alur untuk mengatur jalannya FSM pada hewan. Adapun *state* tersebut meliputi *Idle* (*Q0*), *Grab* (*Q1*), *Throw* (*Q2*), *Eat* (*Q3*), *Run* (*Q4*), *Get the Animals* (*Q5*),



Gambar 5. 22 Tampilan Diagram FSM Pada Hewan

b. Fungsi Transisi Diagram State Menuju Makanan

Tabel transisi dapat dijabarkan kedalam bentuk fungsi dengan. Lambang δ merupakan lambang dari fungsi dari transisi. *State* merupakan *state* awal dengan input parameter *event* dan *action* akan menghasilkan transisi menuju ke *state* yang dituju atau *state* hasil transisi. Berikut penjelasan diagram transisi,

a. Transisi *State* Q0

Transisi pada *state* Q0 dituliskan dalam bentuk fungsi sebagai berikut :

- $\delta(Q0, a0) = Q0$

b. Transisi *State* Q1

Transisi pada *state* Q1 dituliskan dalam bentuk fungsi sebagai berikut,

- $\delta(Q0, a0) = Q0$
- $\delta(Q0, a1) = Q1$

c. Transisi *State* Q2

Transisi pada *state* Q2 dituliskan dalam bentuk fungsi sebagai berikut,

- $\delta(Q0, a0) = Q0$
- $\delta(Q0, a1) = Q1$
- $\delta(Q1, a2) = Q2$

Keterangan :

- *State* (Q) adalah keadaan
- *Event* (e) adalah kondisi
- *Action* (a) adalah aksi
- $\delta (State, event/action) = State$ transisi

c. Tabel Transisi Diagram State *Throw*

Tabel transisi dapat dijabarkan kedalam bentuk fungsi dengan $\delta (State, event/action) = State$ transisi. Lambang δ merupakan lambang dari fungsi transisi. *State* meupakan state awal dengan input parameter *event* (e) dan *action* (a) akan menghasilkan transisi menuju ke *state* yang dituju atau state hasil transisi. Berikut adalah penjabaran dari tabel transisi

Tabel 5. 1 Tabel Transisi Diagram State *Throw*

Event (e) / Action (a)	Tabel Transisi						
	State	Q0	Q1	Q2	Q3	Q4	Q5
	0	Q0	Q1	Q2	Q3	Q4	Q5
	1	Q2	Q0	Q2	Q3	Q4	Q5
	2	Q2	Q1	Q2	Q3	Q4	Q5
	3	Q1	Q0	Q2	Q3	Q4	Q5
	4	Q2	Q1	Q2	Q3	Q4	Q5
	5	Q2	Q1	Q2	Q3	Q4	Q5
	6	Q5	Q1	Q2	Q3	Q4	Q5
	7	Q2	Q1	Q2	Q3	Q4	Q5
	8	Q1	Q0	Q2	Q3	Q4	Q5
	9	Q2	Q3	Q2	Q3	Q4	Q5
	10	Q2	Q1	Q2	Q3	Q4	Q5

Keterangan :

- d. *State* (Q) adalah keadaan
- e. *Event* (e) adalah kondisi
- f. *Action* (a) adalah aksi

Berikut adalah potongan script dari FSM

Source code:

```
public float speed = 10f;
    float limits = 10f;
    string status;
    void Update () {
        if (GameAction.CURRENT_MAKANAN != null && !CEK_OBJECT.activeSelf)
        {
            float Jarak = Vector3.Distance(GameAction.CURRENT_MAKANAN.transform.position, transform.position);
            if (!isTemuMakanan) {
                if (Jarak < 80) {
                    GameAction.CURRENT_MAKANAN.GetComponent<CekMakanan>().isMakan = true;
                    gameObject.GetComponent<Hewan>().isTemuMakanan = true;
                    gameObject.GetComponent<Hewan>().RADAR_OBJECT.SetActive(true);
                    gameObject.GetComponent<pathFinder>().enabled = false;
                    isTemuMakanan = true;

                    status = ("states-Start");
                }
            }
        }
    }
}
```

State pada source code ini digunakan apabila jarak hewan < 80 maka hewan akan mendekati sebuah makanan yang telah dilemparkan oleh pemain.

Source code:

```
}else if (isTemuMakanan) {
    Debug.Log("Jarak " + Jarak);
    if (Jarak >= limits && !inRadius) {
        status = ("states-1");
        UpdateStatus("walk");
        Vector3 targetPos = GameAction.CURRENT_MAKANAN.transform.position;
        Quaternion targetRotation = Quaternion.LookRotation(new Vector3(targetPos.x, transform.position.y, targetPos.z) - transform.position);
        transform.rotation = Quaternion.Slerp(transform.rotation, targetRotation, Time.deltaTime * 2f);
        transform.Translate(Vector3.forward * speed * Time.deltaTime);
        nedTime = Time.fixedTime;
    }
}
```



```

    }else if(Jarak<limits){
        status = ("states-2");
        inRadius = true;
    }

```

State pada *source code* ini digunakan apabila hewan menemukan sebuah makanan dan hewan dapat juga memeriksa apakah makanan cocok atau tidak.

Source code:

```

if (isDestroy) {
    status = ("states-Finish");
}

if (WRONG_OBJECT.activeSelf)
{

    UpdateStatus("run");
    Vector3 targetPos = transform.position;
    targetPos.x += 40;
    targetPos.z += 40;
    Quaternion targetRotation = Quaternion.LookRotation(
new Vector3(targetPos.x, transform.position.y, targetPos.z) - transform.position);
    transform.rotation = Quaternion.Slerp(transform.rotation, targetRotation, Time.deltaTime * 2f);
    transform.Translate(Vector3.forward * speed * Time.deltaTime * 3 );
}
if (isTemuMakanan || CEK_OBJECT.activeSelf || WRONG_OBJECT.activeSelf )
{
    if ( Time.fixedTime - nedTime > 3.0f)
    {
        isDestroy = true;
    }
    Debug.Log(status);
}

}
}

```

State pada *source Code* ini digunakan apabila makanan tidak cocok pada hewan maka hewan akan lari dan menjauh dari pemain.

BAB VI. PENGUJIAN DAN PEMBAHASAN

Pengujian sistem merupakan tahap selanjutnya setelah aplikasi selesai dalam pembuatannya. Pengujian sistem yang dilakukan pada *game* ini meliputi dua tahapan, yaitu pengujian *Alpha* dan *Beta*. Pengujian tersebut dilakukan untuk mengevaluasi hasil sistem yang dibuat.

6.1 Pengujian Sistem

Pengujian sistem merupakan tahap selanjutnya setelah aplikasi selesai dalam pembuatannya. Pengujian sistem yang dilakukan pada *game* “*Save The Animals*” meliputi dua tahapan, yaitu pengujian *Alpha* dan *Beta*. Pengujian tersebut dilakukan untuk mengevaluasi hasil sistem yang dibuat.

6.1.1 Rencana Pengujian

Adapun rancangan pengujian menu *game* yang akan diuji dengan teknik pengujian *Black Box*. Pengujian *Black Box* terfokus pada persyaratan fungsional perangkat lunak. Daftar rencana pengujian dapat dilihat pada tabel 6.1.

Tabel 6. 1 Daftar Rencana Pengujian

No.	Komponen yang di Uji	Skenario Butir Uji	Jenis Uji
1	Tampilan Splash	Terdapat tampilan splash	<i>Black Box</i>
2	Tampilan Main Menu	Terdapat tampilan menu : Mulai, Pengaturan, Keluar	<i>Black Box</i>
3	Menu Mulai	Memilih tombol <i>play</i> untuk memulai permainan	<i>Black Box</i>
4	Menu Pengaturan	Memilih tombol pengaturan untuk mengatur musik	<i>Black Box</i>
5	Menu Keluar	Memilih tombol <i>exit</i> untuk keluar dari aplikasi	<i>Black Box</i>
6	Menu Informasi	Menampilkan informasi hewan	<i>Black Box</i>
7	Tombol Musik	Memilih tombol musik untuk mematikan/menghidupkan musik	<i>Black Box</i>
8	Tampilan <i>Stage</i>	Tampilan <i>stage</i> permainan dari awal hingga akhir	<i>Black Box</i>
9	Tombol <i>Pause</i>	Memilih tombol <i>pause</i> untuk menghentikan permainan sementara	<i>Black Box</i>
10	Tombol Lanjut	Memilih tombol lanjut untuk melanjutkan permainan	<i>Black Box</i>
11	Tombol Akhiri	Memilih tombol akhiri untuk kembali ke menu utama	<i>Black Box</i>

6.1.2 Pengujian Alpha

Pengujian *Alpha* merupakan pengujian fungsional yang digunakan untuk menguji sistem yang telah dibuat dengan metode pengujian *BlackBox*. Daftar pengujian alpha *Game Save The Animals* dapat dilihat pada tabel 6.2.

Tabel 6. 2 Daftar Pengujian Alpha Pada *Game*

No.	Aktifitas yang di Uji	Skenario Uji	Hasil yang Diharapkan	Hasil Uji
1	Tampilan Splash	Terdapat tampilan splash	Menampilkan sebuah intro dari <i>game</i>	Berhasil
2	Tampilan Main Menu	Terdapat tampilan menu : Mulai, Pengaturan, Keluar	Menu utama terdiri dari Menu mulai, pengaturan, dan keluar.	Berhasil
3	Menu Mulai	Memilih tombol mulai untuk memulai permainan	Ketika memilih tombol mulai pengaturan, akan ada animasi yang berupa tombol musik	Berhasil
4	Menu Pengaturan	Memilih tombol pengaturan untuk mengatur musik	Ketika memilih tombol kredit akan keluar panel yang berisi <i>high score player</i>	Berhasil
5	Menu Keluar	Memilih tombol keluar untuk keluar dari aplikasi	Ketika memilih tombol keluar maka pemain akan keluar dari aplikasi	Berhasil
6	Menu Informasi	Menampilkan informasi hewan	Ketika level dimulai pemain dapat mengetahui informasi hewan yang haru di koleksi	Berhasil
7	Tombol Musik	Memilih tombol musik untuk mematikan/menghidupkan musik	Ketika memilih tombol <i>pause</i> permainan akan berhenti sementara dan akan keluar panel <i>pause</i>	Berhasil
8	Tampilan Level	Tampilan level permainan dari awal hingga akhir	Ketika memilih tombol <i>resume</i> ,	Berhasil

			<i>player</i> akan melanjutkan permainan di tempat terakhir pada saat memilih tombol <i>pause</i>	
9	Tombol Menu	Memilih tombol menu untuk menghentikan permainan sementara	Ketika memilih tombol menu, akan muncul sebuah layar sebuah menu	Berhasil
10	Tombol Lanjut	Memilih tombol lanjut untuk melanjutkan permainan	Ketika pemain memilih tombol lanjut maka akan kembali ke dalam permainan	Berhasil
11	Tombol Akhiri	Memilih tombol akhiri untuk kembali ke menu utama	Ketika pemain memilih tombol akhiri maka akan kembali ke menu utama	Berhasil

6.1.3 Pengujian *Beta*

Pengujian *beta* merupakan pengujian yang dilakukan dengan tujuan untuk mengetahui bagaimana kualitas sistem yang telah dibuat. Dalam pengujian *beta* dilakukan terhadap responden atau calon pengguna sistem dengan menggunakan kuesioner atau angket. Kuesioner diberikan kepada 10 orang yang berusia 10 – 11 tahun. Kuesioner dibuat menggunakan skala jawaban 1 sampai 5. Daftar pertanyaan kuesioner akan dijelaskan pada Tabel 6.3.

Tabel 6.3 Daftar Pertanyaan Kuesioner

No	Pertanyaan	Nilai				
		5	4	3	2	1
1	Bagaimana tampilan <i>game</i> “ <i>Save the Animals</i> ”?					
2	Bagaimana tingkat kesulitan permainan dalam <i>game</i> “ <i>Save the Animals</i> ”?					
3	Bagaimana dengan desain karakter hewan					

	dalam game “ <i>Save the Animals</i> ”?					
4	Apakah game “ <i>Save the Animals</i> ” menarik untuk dimainkan?					
5	Apakah game ini dapat memberikan edukasi yang bermanfaat?					
Skor						
1	2	3	4	5		
Sangat kurang	Kurang	Cukup	Baik	Sangat baik		

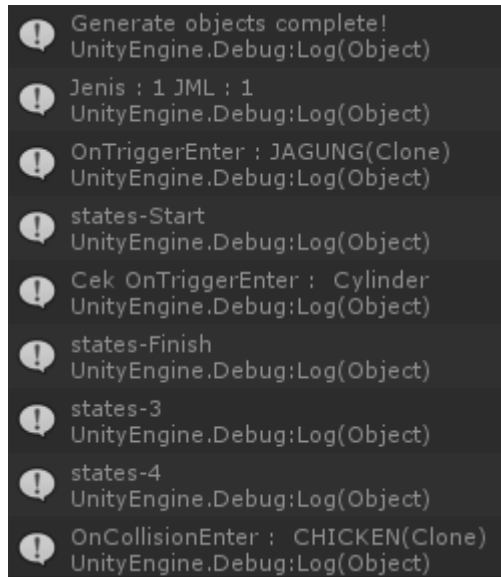
6.1.4 Hasil Pengujian Metode *Finite State Machine*

Untuk menguji *Finite State Machine* ini berjalan sesuai yang diharapkan, maka dilakukan pengujian dengan melakukan melempar benda dengan radius yang telah ditentukan ke arah objek maka objek akan mendekat jika radius telah tercapai, jika jarak antara benda yang di lempar dengan objek tidak dalam radius yang telah di tentukan maka objek tetap berjalan sesuai titik yang sudah di tetapkan.



Gambar 6. 1 Saat Hewan Masuk Ke Dalam Radius

Pada gambar 6. 1 dijelaskan bahwa metode *Finite State Machine* berhasil diterapkan pada objek hewan “Ayam” dengan melemparkan sebuah makanan yaitu “Jagung” dan berjalan sesuai yang diharapkan.

Gambar 6. 2 Pengujian Objek Pada *Console*

6.2 Pembahasan

6.2.1 Hasil Pengujian Alpha

Berdasarkan hasil pengujian *Alpha* di tabel 6.2, diperoleh bahwa *Game* “Save The Animals” ini secara fungsional mengeluarkan hasil sesuai dengan yang diharapkan.

6.2.2 Hasil Pengujian *Beta*

Hasil kuesioner yang telah dilakukan kepada 4 orang pada umur 10 – 11 tahun dapat dilihat pada tabel 6. 3.

Tabel 6. 3 Hasil Kuisisioner Penilaian *Game*

No	Nama	Umur	JK	#1	#2	#3	#4	#5
1.	Dimaz Eka Putra	11	L	5	4	5	4	5
2.	Catur Joko Tetuko	11	L	5	3	4	5	3
3.	Bifadllika	11	L	3	5	4	5	3
4.	Zyahra Zulfa Nabila	11	P	3	4	4	5	5

Berdasarkan hasil kuesioner tersebut, dicari prosentase masing-masing jawaban dengan menggunakan rumus : $H = J/U * 100\%$

Keterangan:

H = Hasil perhitungan

J = Banyaknya jawaban (skala jawaban) oleh Responden

U = Jumlah User

Berikut ini hasil perhitungan prosentase dari jawaban hasil kuesioner yang telah dilakukan terhadap 4 *user* yang dijadikan contoh penelitian *Game “Save The Animals”*.

b. Pertanyaan pertama, “Bagaimana tampilan *game “Save the Animals”*?”

Tabel 6. 4 Hasil Pengujian Pertanyaan Pertama

Skala Jawaban	1	2	3	4	5
Responden	0	0	2	0	2
Hasil	0%	0%	50%	0%	50%

Berdasarkan hasil tabel 6. 4, dapat dilihat bahwa dari 4 responden sebanyak 2 responden menjawab sangat baik dengan prosentase 50%, 2 responden menjawab cukup dengan prosentase 50% dan tidak ada responden menjawab kurang dengan prosentase 0%. Dengan hasil tersebut dapat disimpulkan bahwa tampilan *Game “Save the Animals”* bagus.

c. Pertanyaan kedua, “Bagaimana tingkat kesulitan permainan dalam *Game “Save The Animals”*?”

Tabel 6. 5 Hasil Pengujian Pertanyaan Kedua

Skala Jawaban	1	2	3	4	5
Responden	0	0	1	2	1
Hasil	0%	0%	25%	50%	25%

Berdasarkan hasil tabel 6. 5, dapat dilihat bahwa dari 4 responden sebanyak 1 responden menjawab sangat mudah dengan prosentase 25%, 2 responden menjawab cukup mudah dengan prosentase 50% dan 1 responden menjawab mudah dengan prosentase 25%. Dengan hasil tersebut dapat disimpulkan bahwa tingkat kesulitan permainan dalam *game “Save The Animals”* adalah cukup sulit.

d. Pertanyaan ketiga, “Bagaimana dengan desain karakter hewan dalam *game “Save the Animals”*?”

Tabel 6. 6 Hasil Pengujian Pertanyaan Ketiga

Skala Jawaban	1	2	3	4	5
Responden	0	0	0	3	1

Hasil	0%	0%	0%	75%	25%
-------	----	----	----	-----	-----

Berdasarkan hasil tabel 6. 6, dapat dilihat bahwa dari 4 responden sebanyak 1 responden menjawab baik dengan prosentase 25%, 1 responden menjawab cukup dengan prosentase 75%, dan 0 responden menjawab kurang dengan prosentase 0%. Dengan hasil tersebut dapat disimpulkan bahwa desain karakter pada game “*Save The Animals*” baik.

- e. Pertanyaan keempat, “Apakah *game* “*Save the Animals*” menarik untuk dimainkan?”

Tabel 6. 7 Hasil Pengujian Pertanyaan Keempat

Skala Jawaban	1	2	3	4	5
Responden	0	0	0	1	3
Hasil	0%	0%	0%	25%	75%

Berdasarkan hasil tabel 6. 7, dapat dilihat bahwa dari 4 responden sebanyak 3 responden menjawab sangat menarik dengan prosentase 75%, 1 responden menjawab baik dengan prosentase 25%, dan tidak ada responden menjawab kurang menarik dengan prosentase 0%. Dengan hasil tersebut dapat disimpulkan bahwa *game* “*Save The Animals*” sangat menarik untuk dimainkan.

- f. Pertanyaan kelima, “Apakah *game* ini dapat memberikan edukasi yang bermanfaat?”

Tabel 6. 8 Hasil Pengujian Pertanyaan Kelima

Skala Jawaban	1	2	3	4	5
Responden	0	0	2	0	2
Hasil	0%	0%	50%	0%	50%

Berdasarkan hasil tabel 6. 8, dapat dilihat bahwa dari 4 responden sebanyak 2 responden menjawab sangat menarik dengan prosentase 50%, 2 responden menjawab cukup dengan prosentase 50%, dan tidak ada responden menjawab kurang menarik dengan prosentase 0%. Dengan hasil tersebut dapat disimpulkan bahwa *game* “*Save The Animals*” bermanfaat untuk hal edukasi.

BAB VII. PENUTUP

Bab ini akan menjelaskan tentang kesimpulan yang didapat pada saat pengerjaan skripsi melalui uji coba yang dilakukan dan analisa yang dilakukan dalam penelitiannya. Bab ini juga berisi saran yang bisa dilakukan untuk penelitian di masa yang akan datang.

7.1 Kesimpulan

Berdasarkan penelitian yang dilaksana oleh penulis pada Implementasi Finite State Machine Pada *Game* “Save The Animals” dengan melakukan penerapan pada sebuah objek yang ada di dalam game, maka dapat ditarik kesimpulan sebagai berikut :

1. *Game* ini telah berhasil menerapkan metode *Finite State Machine (FSM)* pada objek hewan ke dalam sebuah *game* yang bergenre *adventure*.
2. Hasil pengujian bahwa *game* ini memiliki tampilan *gameplay* yang menarik dan dapat memberikan sebuah pengetahuan informasi tentang hewan.

7.2 Saran

Video game “Save The Animals” Ini masih bisa dikembangkan lebih jauh dari banyak sisi, *game* ini dapat di kembangkan dari sisi :

1. Level bisa di tingkat menjadi lebih banyak agar permainan lebih menantang.
2. Dari sisi tampilan dan desain visual pada objek dapat dibuat lebih nyata.
3. Dari sisi *gameplay* dapat ditambahkan mode bonus *game* agar permainan lebih menarik.

DAFTAR PUSTAKA

- [1] Bimantoro, Tito. Haryanto, Hanny. (2016). Pemodelan Perilaku Musuh Menggunakan *Finite State Machine (FSM)* Pada *Game* Pengenalan Unsur Kimia. *Journal of Applied Intelligent System*, Vol.1, No. 3, Oktober 2016 : 210-219.
- [2] Tjahjono, Harvey. Liliana. Gunadi, Kartika. (2015). Pembuatan *Game* Cerita Rakyat Dengan Bentuk *Adventure Game*. Vol 3, No 2 (2015).
- [3] Terttiaavini. Kaspriyanto, Ari. Hartini. (2016). Aplikasi Cak Ingkling Berbasis Android. *Jurnal Ilmiah Informatika Global* Volume 7 No.1 Juli 2016.
- [4] Dyta, Eka, Septian. (2013). Rancang Bangun Aplikasi Game Edukasi Anak Untuk Mengenal Bentuk Dan Warna. Fakultas Ilmu Komputer Program Studi Teknik Informatika-S1, Universitas Dian Nuswantoro, Semarang.
- [5] Arif, Miftachul, Yunifa. Hariadi, Mochamad. S.N, Mardi, Supeno. (2012). Integrasi *Hierarchy Finite State Machine* Dan Logika *Fuzzy* Untuk Desain Strategi NPC Game. Jurusan Teknik Elektro. ITS Surabaya.
- [6] Blender Org. About Blender 3D. (Online) Tersedia : <https://www.blender.org/about/>, [08 Maret 2017]
- [7] Prasetyo, David. 2015. “*Pengertian Storyboard*”, (Online) Tersedia : <http://www.davidprasetyo.com/2015/11/pengertian-storyboard.html>, [14 Maret 2017]
- [8] Wicaksana, Arif. 2016. “*Pengertian Use Case*”, (Online) Tersedia : <https://medium.com/@arifwicaksanaa/pengertian-use-case-a7e576e1b6bf>, [01 April 2017]
- [9] Filus, Teo. 2017. “*Pengenalan Bahasa Pemrograman C#*”, (Online) Tersedia : <https://www.codepolitan.com/pengenalan-bahasa-pemrograman-c-587effa1cb95b>, [05 Mei 2017]
- [10] Andika, Dwiky. 2017. “*Pengertian Flowchart / IT-Jurnal*”, (Online) Tersedia : <https://www.it-jurnal.com/pengertian-flowchart/>, [10 Juni 2017]
- [11] Spelman, Dr. Lucy. “*National Geographic Animal Encyclopedia/Fauna Pedia*”. Jakarta : KPG Kepustakaan Populer Gramedia. 2012