

**PEMBANGUNAN *GAME* RPG *TURN-BASED*
MENGUNAKAN ALGORITMA BRUTE FORCE**

TUGAS AKHIR

Disusun Oleh:

David Vincent Turangan

(16013047)



PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS KATOLIK DE LA SALLE

MANADO

2020

**PEMBANGUNAN *GAME* RPG *TURN-BASED*
MENGUNAKAN ALGORITMA BRUTE FORCE**

LAPORAN TUGAS AKHIR

Ditulis untuk memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana
Teknik pada Program Studi Teknik Informatika

Disusun Oleh:

**David Vincent Turangan
(16013047)**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO
2020**

LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : David Vincent Turangan
NIM : 16013047
Tempat /Tanggal Lahir : Manado, 04 Agustus 1999
Fakultas/ Program Studi : Teknik/Teknik Informatika

Menyatakan bahwa Karya Ilmiah/Tugas Akhir dan atau Aplikasi/Program berjudul “**Pembangunan *Game* RPG *Turn-Based* Menggunakan Algoritma *Brute Force***” yang saya buat adalah benar hasil karya saya dan bukan karya tulis orang lain, baik sebagian atau seluruhnya kecuali dalam bentuk kutipan yang telah disebutkan sumbernya.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya dan apabila pernyataan ini tidak benar maka saya bersedia menerima sanksi akademis sesuai dengan yang ditetapkan oleh Fakultas, berupa pembatalan Karya Ilmiah/Tugas Akhir dan hasilnya.

Manado, 30 Juli 2020

Yang Menyatakan,

David Vincent Turangan

Menyetujui,

Pembimbing I,

Pembimbing II,

Debby Paseru, S.T., M.M.S.I., M.Ed Rinaldo T. B. Turang, S.Kom., M.Kom

Mengetahui,

Ketua Program Studi,

Dekan Fakultas Teknik,

Vivie D. Kumenap, S.T., M.Cs

Ronald A. Rachmadi, S.T., M.T



**UNIVERSITAS KATOLIK DE LA SALLE
MANADO – INDONESIA**

Nama : David Vincent Turangan
NIM : 16013047
Fakultas : Teknik
Program Studi : Teknik Informatika
Judul Tugas Akhir : Pembangunan *Game RPG Turn-Based* Menggunakan
Algoritma Brute Force
Pembimbing I : Debby Paseru, S.T., M.M.S.I., M.Ed
Pembimbing II : Rinaldo T. B. Turang, S.Kom., M.Kom

Menyetujui,
Manado, 30 Juli 2020

Pembimbing I,

Pembimbing II,

Debby Paseru, S.T., M.M.S.I., M.Ed Rinaldo T. B. Turang, S.Kom., M.Kom

Mengetahui,

Ketua Program Studi,

Dekan Fakultas Teknik,

Vivie D. Kumenap, S.T., M.Cs

Ronald A. Rachmadi, S.T., M.T

ABSTRACT

Computer games are something that a lot of people like. There are a lot of genre in a computer games, one of them is RPG turn-based. Games with that genre use a turn-based combat which allow player to consider their moves to defeat the opponent. But there is some enemy in a turn-based genre games that don't calculate and only have one type of move or just using a randomized system.

Because of that, an RPG turn-nased genre game is going to be built in this essay with a simple artificial intelligence using Brute Force algorithm to calculate and consider the moves in a battle. Brute Force is an algorithm that have a simple and straightforward way to solve problems.

The method in use for the making of this game is Multimedia Development Life Cycle (MDLC). This game will be programmed using GameMaker Studio 2 Engine.

After finishing and testing is done, this RPG turn-based game is able to have a simple AI that can use the best move after calculating using the Brute Force algorithm.

Keyword : Game, RPG turn-based, AI, Brute Force algorithm.

ABSTRAK

Permainan komputer merupakan sesuatu yang disukai banyak orang. Permainan komputer memiliki banyak *genre*, salah satunya adalah RPG *turn-based*. *Game* dengan *genre* tersebut menggunakan pertarungan *turn-based* yang memungkinkan pemain untuk mempertimbangkan gerakan terbaik untuk melawan musuh. Namun, beberapa musuh dalam *game* dengan *genre* tersebut biasanya tidak melakukan pertimbangan dan hanya memiliki satu jenis gerakan atau menggunakan mekanisme acak.

Untuk itu pada Tugas Akhir ini akan dibuat *game* dengan *genre* RPG *turn-based* dengan kecerdasan buatan sederhana yang menggunakan algoritma *Brute Force* untuk mempertimbangkan gerakan terbaik pada saat pertarungan. Algoritma *Brute Force* merupakan algoritma sederhana yang memiliki cara penyelesaian masalah sederhana, langsung, dan jelas.

Metode yang akan digunakan pada pembuatan *game* ini adalah metode *Multimedia Development Life Cycle* (MDLC). Pemrograman *game* akan dilakukan dengan menggunakan *GameMaker Studio 2*.

Setelah selesai dibangun dan dilakukan pengujian, *game* RPG *turn-based* yang dibuat dapat menghasilkan kecerdasan buatan sederhana yang dapat memilih gerakan terbaik berdasarkan kondisi yang ada dengan melakukan perhitungan menggunakan algoritma *Brute Force*.

Kata Kunci : Permainan, RPG *Turn-Based*, kecerdasan buatan, algoritma *Brute Force*.

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa, karena atas rahmat dan tuntunan-Nya sajalah, penulis dapat menyelesaikan kegiatan tugas akhir dan pembuatan aplikasi beserta dengan penulisan laporan ini.

Dalam pelaksanaan TA dan penulisan laporan ini, penulis mendapatkan banyak sekali bantuan, bimbingan, dan dukungan dari berbagai pihak. Untuk itu pada kesempatan ini penulis hendak menyampaikan rasa terima kasih kepada:

1. Prof. Dr. Johanis Ohoitumur selaku Rektor Universitas Katolik De La Salle Manado.
2. Bapak Ronald Albert Rachmadi, S.T., M.T selaku Dekan Fakultas Teknik Unika De La Salle Manado.
3. Ibu Vivie D. Kumenap, S.T., M.Cs selaku Ketua Program Studi Teknik Informatika Unika De La Salle Manado.
4. Ibu Debby Paseru, S.T., M.M.S.I., M.Ed selaku Pembimbing I yang selalu memberi bimbingan dan masukan, serta waktu yang diberikan untuk membimbing penulis.
5. Bapak Rinaldo Theodorus Bonifasius Turang, S.Kom., M.Kom selaku Pembimbing II yang selalu memberikan bimbingan dan masukan dalam pembuatan aplikasi maupun laporan.
6. Keluarga yang selalu memberikan semangat kepada penulis khususnya selama penulisan laporan ini.
7. Pihak-pihak lainnya yang tidak mungkin penulis sebutkan satu per satu.

Penulis berharap laporan yang telah disusun ini dapat membantu memberikan pengetahuan kepada pembaca. Akhir kata, penulis sadar bahwa masih banyak kekurangan dalam penulisan laporan ini, untuk itu penulis terbuka terhadap saran dan masukan dari semua pihak demi terciptanya laporan yang lebih baik lagi.

Manado, Juli 2020

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERNYATAAN.....	ii
LEMBAR PENGESAHAN	iii
<i>ABSTRACT</i>	iv
ABSTRAK	v
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	2
1.3. Tujuan Kerja Praktek	2
1.4. Manfaat Kerja Praktek	2
1.5. Batasan Masalah.....	2
1.6. Metode Penelitian.....	2
1.7. Sistematika Penulisan.....	3
BAB II LANDASAN TEORI.....	4
2.1. <i>Game</i>	4
2.2. <i>RPG Turn-Based</i>	5
2.3. <i>Algoritma Brute Force</i>	5
2.4. <i>Game Maker Studio 2</i>	6
2.5. Metodologi Pengembangan Aplikasi.....	7
2.6. Kakas Pemodelan	8
2.7. Studi Kasus yang Terkait	10
BAB III ANALISIS	12
3.1. Identifikasi Persyaratan Dasar Aplikasi	12
3.1.1. Analisis <i>Game</i> Serupa	12
3.1.2. Analisis Hasil Perbandingan <i>Game</i>	15
3.1.3. Konsep Dasar <i>Game</i>	16
3.1.4. Mekanik dan <i>Rule Game</i>	16
3.1.5. Analisis Spesifikasi Teknologi	17
3.2. Perhitungan Algoritma <i>Brute Force</i>	18
BAB IV PERANCANGAN	21
4.1. Spesifikas Target Pengguna dan Aplikasi.....	21
4.1.1. Spesifikasi Target Pengguna	21
4.1.2. Spesifikasi Minimum Perangkat.....	21
4.1.3. Fitur <i>Game</i>	22
4.2. Pemodelan Aplikasi yang Dibangun	23
4.2.1. <i>Flowchart</i> Alur <i>Game</i>	23

4.2.2. <i>Flowchart</i> Pengambilan Keputusan AI	24
4.3. <i>Storyboard</i>	25
BAB V IMPLEMENTASI.....	28
5.1. Pengumpulan Aset <i>Game</i>	28
5.1.1. Desain Karakter	28
5.1.2. Desain Latar Belakang <i>Game</i>	32
5.2. Implementasi <i>Storyboard</i>	33
5.3. Pemrograman Aplikasi	35
BAB VI PENGUJIAN	44
6.1. Tujuan Pengujian.....	44
6.2. Kriteria Pengujian	44
6.3. Kasus Pengujian	44
6.4. Pelaksanaan Pengujian	45
6.4.1. Pengujian <i>Whitebox</i>	45
6.4.2. Pengujian <i>Blackbox</i>	53
6.4.3. <i>User Acceptance Test</i> (UAT)	55
BAB VII KESIMPULAN DAN SARAN.....	60
7.1. Kesimpulan.....	60
7.2. Saran.....	60
DAFTAR PUSTAKA	61
LAMPIRAN	

DAFTAR TABEL

Tabel 2.1 Simbol <i>Flowchart</i>	8
Tabel 3.1 Hasil Analisis Perbandingan <i>Game</i>	15
Tabel 3.2 Perangkat Keras yang Digunakan.....	18
Tabel 3.3 Perangkat Lunak yang Digunakan.....	18
Tabel 6.1 Kriteria Pengujian	34
Tabel 6.2 Kasus Pengujian.....	34
Tabel 6.3 Pengujian Algoritma <i>Brute Force</i> Serang	47
Tabel 6.4 Pengujian Algoritma <i>Brute Force Heal</i>	48
Tabel 6.5 Pengujian Algoritma <i>Brute Force Guard</i>	49
Tabel 6.6 Hasil Pengujian <i>Blackbox</i>	53
Tabel 6.7 Hasil UAT	56

DAFTAR GAMBAR

Gambar 2.1	Pencocokan <i>String</i>	6
Gambar 3.1	Tampilan <i>Gameplay Semi-Openworld</i> Pokemon.....	13
Gambar 3.2	Tampilan <i>Gameplay</i> Pokemon <i>Battle</i>	13
Gambar 3.3	Tampilan <i>Gameplay</i> Civilization VI.....	14
Gambar 3.4	Tampilan <i>Gameplay</i> Final Fantasy XIII	15
Gambar 4.1	<i>Flowchart</i> Alur <i>Game</i>	23
Gambar 4.2	<i>Flowchart</i> Pengambilan Keputusan AI.....	24
Gambar 4.3	<i>Storyboard</i> Menu Awal.....	25
Gambar 4.4	<i>Storyboard</i> Tampilan <i>Stage</i>	26
Gambar 4.5	<i>Storyboard</i> Tampilan Pertarungan	27
Gambar 5.1	Desain Karakter Musuh Pertama	28
Gambar 5.2	Desain Karakter Musuh Kedua	29
Gambar 5.3	Desain Karakter Musuh Ketiga.....	29
Gambar 5.4	Desain Karakter Musuh Keempat	29
Gambar 5.5	Desain Karakter Musuh Kelima.....	30
Gambar 5.6	Desain Karakter Musuh Keenam	30
Gambar 5.7	Desain Karakter Musuh Ketujuh.....	30
Gambar 5.8	Desain Karakter Musuh Kedelapan	31
Gambar 5.9	Desain Karakter Musuh Kesembilan	31
Gambar 5.10	Desain Karakter Musuh Kesepuluh	31
Gambar 5.11	Desain Karakter Pemain.....	32
Gambar 5.12	Desain Latar Belakang Pertama	32
Gambar 5.13	Desain Latar Belakang Kedua.....	33
Gambar 5.14	Tampilan Menu Awal	33
Gambar 5.15	Tampilan <i>Stage</i>	34
Gambar 5.16	Tampilan Pertarungan	34
Gambar 6.1	Pengujian Menu <i>Game</i>	45
Gambar 6.2	Pengujian Mulai <i>Game</i> Baru.....	46
Gambar 6.3	Pengujian Melanjutkan <i>Game</i>	46
Gambar 6.4	Pengujian Algoritma <i>Brute Force</i> Serangan Sihir	50
Gambar 6.5	Pengujian Algoritma <i>Brute Force</i> Serangan Gabungan.....	50
Gambar 6.6	Pengujian Algoritma <i>Brute Force</i> Serangan Biasa	50
Gambar 6.7	Pengujian Algoritma <i>Brute Force</i> Gerakan <i>Heal</i>	51
Gambar 6.8	Pengujian Algoritma <i>Brute Force</i> Gerakan <i>Guard</i>	51
Gambar 6.9	<i>Flowchart</i> Kriteria Pengujian Menang/Kalah.....	52
Gambar 6.10	Pengujian Kondisi Menang.....	52
Gambar 6.11	Pengujian Kondisi Kalah	53
Gambar 6.12	<i>Screenshot</i> Menu Awal <i>Game</i>	53
Gambar 6.13	<i>Screenshot</i> Awal <i>Game</i>	54
Gambar 6.14	<i>Screenshot</i> <i>Checkpoint</i> <i>Game</i>	54
Gambar 6.15	<i>Screenshot</i> AI Menyerang.....	54
Gambar 6.16	<i>Screenshot</i> AI Melakukan <i>Heal</i>	54
Gambar 6.17	<i>Screenshot</i> AI Melakukan <i>Guard</i>	55
Gambar 6.18	<i>Screenshot</i> <i>Player</i> Menang	55
Gambar 6.19	<i>Screenshot</i> <i>Player</i> Kalah.....	55

Gambar 6.20	Persentase Pertanyaan Pertama	56
Gambar 6.21	Persentase Pertanyaan Kedua.....	56
Gambar 6.22	Persentase Pertanyaan Ketiga	56
Gambar 6.23	Persentase Pertanyaan Keempat.....	57
Gambar 6.24	Persentase Pertanyaan Kelima	57
Gambar 6.25	Persentase Pertanyaan Keenam.....	57
Gambar 6.26	Persentase Pertanyaan Ketujuh	58
Gambar 6.27	Persentase Pertanyaan Kedelapan	58
Gambar 6.28	Persentase Pertanyaan Kesembilan	58

DAFTAR LAMPIRAN

LAMPIRAN A : Transkrip Survey.....	A-1
------------------------------------	-----

BAB I

PENDAHULUAN

1.1 Latar Belakang

Permainan adalah suatu hal yang dilakukan manusia sebagai bentuk hiburan. Permainan memiliki banyak jenis, seperti permainan tradisional yaitu permainan petak-umpet, congklak, kartu, permainan bola dan lainnya. Pada zaman sekarang ini dengan perkembangan teknologi yang sangat pesat, muncul permainan-permainan yang bersifat digital. Permainan digital tersebut berupa permainan *arcade*, permainan komputer dan permainan *smartphone*. Permainan komputer dan *smartphone* sangatlah populer karena kemudahan akses permainan-permainan tersebut dan juga teknologi yang diperlukan umumnya dimiliki oleh semua orang.

Permainan komputer memiliki banyak *genre*, namun salah satu *genre* yang paling populer adalah *Role Playing Games* (RPG) di mana pemain akan menjadi karakter tertentu dan memainkan permainan tersebut sesuai dengan sudut pandang karakter yang dimainkan. Selain itu, terdapat juga *genre turn-based* di mana permainan dilakukan oleh dua orang secara bergantian. Salah satu contoh permainan tradisional terkenal yang berjenis *turn-based* adalah catur [1].

Permainan komputer berjenis *turn-based* biasanya dapat dimainkan oleh dua orang, atau juga bisa satu orang melawan musuh berupa AI (*Artificial Intelligence*) yang sudah diprogram ke dalam permainan tersebut. Namun, terdapat beberapa *game turn-based* dengan musuh yang hanya dapat menggunakan satu gerakan saja atau menggunakan mekanisme acak yang membuat *game* tersebut terlalu mudah.

Oleh karena itu penulis akan membuat sebuah *game* yang ber-*genre* RPG *turn-based* dengan memiliki cerita tentang seorang karakter yang akan menjadi pahlawan untuk mengalahkan raja kegelapan yang ingin menghancurkan dunia. *Game RPG turn-based* dengan tema tersebut merupakan kombinasi klasik yang sudah lama digunakan para pembuat *game*. Sudah sangat banyak *game* yang menggunakan kombinasi tersebut, namun tidak berarti bahwa orang-orang sudah bosan dengan tema tersebut. Hal ini terbukti dari penggunaan kombinasi RPG dan

tema *hero* tersebut pada sebuah *game* dan sampai sekarang masih banyak digunakan dan diproduksi. Pada *game* ini juga akan dibuat AI sederhana yang dapat mengambil keputusan sebaik mungkin pada saat permainan berjalan untuk mengalahkan lawan sehingga dapat memberikan sensasi bermain yang menyenangkan bagi pemain. AI tersebut akan menggunakan algoritma *Brute Force* yang memiliki cara penyelesaian masalah yang sederhana. Berdasarkan kondisi tertentu, AI akan menghitung gerakan yang paling baik untuk mengalahkan pemain.

1.2 Rumusan Masalah

Bagaimana membuat sebuah *game turn-based* RPG menggunakan algoritma *Brute Force* dengan AI sederhana yang dapat memilih gerakan terbaik?

1.3 Tujuan Tugas Akhir

Membuat sebuah *game turn-based* RPG menggunakan algoritma *Brute Force* dengan AI sederhana yang bisa memilih gerakan terbaik saat pertarungan dan tidak hanya menggunakan satu jenis gerakan atau menggunakan mekanisme acak.

1.4 Manfaat Tugas Akhir

1. Pemain dapat menikmati permainan yang menawarkan tantangan karena memiliki AI yang dapat memilih gerakan terbaik untuk melawan pemain.
2. Penulis mendapat pengalaman dalam membuat sebuah *game turn-based* RPG dan membuat AI sederhana yang bisa memilih gerakan terbaik.

1.5 Batasan Masalah

Batasan masalah dalam pembuatan aplikasi ini adalah:

1. *Game* hanya dapat dimainkan oleh satu orang saja.
2. *Game* hanya dapat dimainkan pada *platform* laptop atau PC.

1.6 Metodologi Penelitian

Metodologi penelitian yang digunakan adalah metode *Research and Development* (R&D). Metode penelitian *Research and Development* adalah penelitian yang digunakan untuk menghasilkan suatu produk dan kemudian

menguji produk tersebut [2]. Metode penelitian *Research and Development* ini akan digunakan sebagai acuan pada metodologi pengembangan sistem yang digunakan, yaitu metodologi pengembangan sistem *Multimedia Development Life Cycle* (MDLC) oleh Luther-Sutopo. Dengan menggunakan metode ini, maka dapat dihasilkan *game* RPG *turn-based* menggunakan algoritma *Brute Force* yang memiliki AI sederhana dengan kemampuan untuk memilih gerakan yang terbaik pada saat melawan pemain.

1.7 Sistematika Penulisan

Sistematika penulisan laporan ini terdiri dari tujuh bab, yaitu:

1. BAB I PENDAHULUAN

Bab ini membahas tentang latar belakang pembuatan aplikasi, rumusan masalah, tujuan dan manfaat, batasan masalah, metode penelitian dan sistematika penulisan.

2. BAB II LANDASAN TEORI

Bab ini membahas tentang teori-teori yang berkaitan dengan pembuatan aplikasi seperti pengertian *game* dan *genre* RPG *turn-based*, algoritma *Brute Force*, pemrograman aplikasi, kaskas pemodelan dan metodologi pengembangan sistem yang digunakan, prosedur pengolahan data dan penelitian-penelitian yang berkaitan dengan pembuatan aplikasi ini.

3. BAB III ANALISIS

Pada bab ini akan dibahas mengenai persyaratan dasar aplikasi, tujuan dan manfaat dari aplikasi, serta konsep dan algoritma dari aplikasi yang akan dibuat.

4. BAB IV PERANCANGAN

Bab ini berisi proses penentuan fitur, target pengguna, desain pemodelan aplikasi, pembuatan desain *storyboard* aplikasi.

5. BAB V IMPLEMENTASI

Bab ini berisi proses pengumpulan aset-aset yang diperlukan dan pembangunan aplikasi berdasarkan hasil perancangan yang sudah dilakukan serta sesuai dengan *storyboard* dan desain yang sudah ada dan telah dikumpulkan.

6. BAB VI PENGUJIAN

Bab ini berisi pengujian dari aplikasi yang telah dibuat dan memperbaiki hal-hal yang masih belum berfungsi sesuai hasil pengujian.

7. BAB VII KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari pembuatan aplikasi ini, dan saran terhadap hal-hal yang harus diperhatikan serta diperbaiki pada pembuatan aplikasi selanjutnya.

BAB II

LANDASAN TEORI

Pada bagian ini akan berisi teori pendukung yang berperan sebagai penunjang pada saat proses pembuatan *game*. Teori-teori tersebut didasarkan dari referensi-referensi yang diambil melalui jurnal atau buku yang berkaitan.

2.1 Game

Secara umum, *game* adalah sesuatu yang dapat dimainkan atau sesuatu yang dapat menghibur karena permainan bersifat menghibur bagi pemainnya [3]. Namun, jika dikaji secara lebih dalam lagi, *game* memiliki pengertian yang lebih kompleks. Secara lebih rinci, *Game* memiliki komponen-komponen utama yang mendefinisikan sebuah *game* itu sendiri. Ketiga komponen tersebut adalah interaktif, tujuan, dan aturan [4].

Komponen yang pertama yaitu interaktif berarti sebuah *game* harus memiliki sifat interaktif, yang adalah harus ada interaksi antara pemain dan *game* itu sendiri. Kedua, *game* memiliki tujuan. Sebuah *game* harus memiliki tujuan akhir yang dapat dicapai oleh pemain. Tujuan tersebut dapat berupa menang melawan seseorang, ataupun menyelesaikan cerita yang ada. Terakhir, sebuah *game* harus memiliki aturan. *Game* yang tidak memiliki aturan akan membuat alur dari *game* tersebut menjadi kacau. Tidak ada yang dapat menentukan mana jalan yang benar atau tidak ada yang menentukan siapa yang menang atau kalah. Oleh karena itu aturan merupakan komponen yang penting pada sebuah *game* [4].

Game dapat diklasifikasikan menurut *genre*-nya masing-masing. Secara umum, *game* dibedakan terhadap *genre* dari cara bermainnya. Terdapat banyak *genre* dan *subgenre* yang ada pada *game*. Beberapa *genre* utama dari *game* adalah *action*, *shooter*, *adventure*, *simulation*, dan *strategy*. Sebuah *game* tidak dibatasi pada satu *genre* saja, tapi dapat memiliki beberapa *genre* yang berbeda-beda [1].

2.2 RPG Turn-Based

Genre dari *game* yang akan dibuat adalah *RPG turn-based*. *RPG* dan *turn-based* termasuk ke dalam *subgenre* dan bukan *genre* utama dalam *game*. *RPG*

merupakan *subgenre* dari *genre action*, dan *turn-based* merupakan *subgenre* dari *genre strategy* [1]. RPG merupakan singkatan dari *Role Playing Game*, dimana pada *game* RPG, pemain memainkan *game* pada segi pandang salah satu karakter utama dalam *game* [5]. *turn-based* adalah permainan dengan pertarungan yang bergantian antara beberapa pemain atau AI. Pemain dapat menggunakan waktu pada saat gilirannya untuk memilih dengan teliti cara mengalahkan lawannya [6].

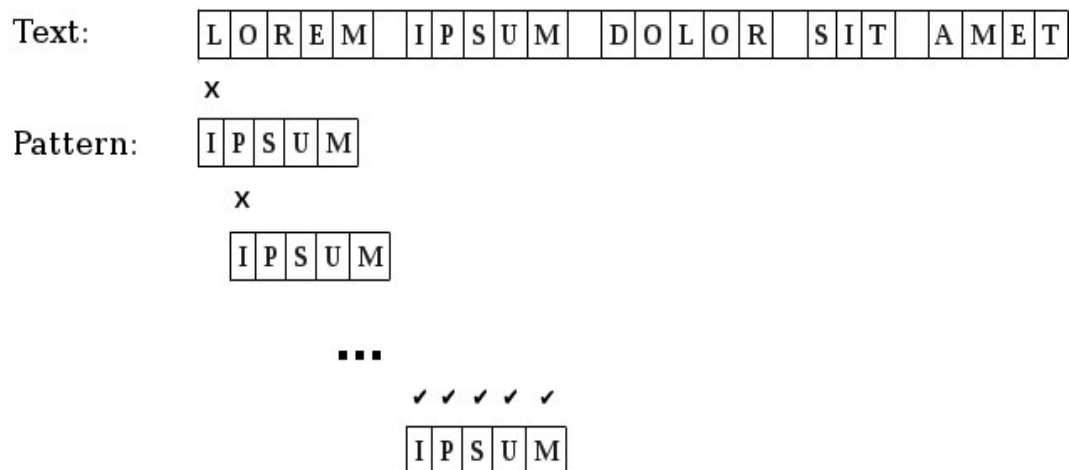
Dengan demikian, jika *genre* RPG digabungkan dengan *genre turn-based*, maka akan menjadi *game* yang memiliki *gameplay* dimana pemain akan berperan sebagai seorang karakter pada sebuah *game* dalam petualangannya dan pada saat melakukan pertarungan, pemain akan melawan musuh dengan cara bermain *game genre turn-based* secara bergantian [1].

2.3 Algoritma *Brute Force*

Algoritma ini adalah algoritma yang memiliki cara kerja yang bersifat langsung atau biasa disebut *straightforward*. Maksudnya adalah dalam suatu proses pemecahan masalah, algoritma *Brute Force* akan memecahkan masalah tersebut menggunakan cara yang paling sederhana, langsung dan jelas. Algoritma *Brute Force* adalah salah satu algoritma yang digunakan pada implementasi strategi algoritma. Strategi algoritma adalah sebuah pendekatan dengan menggunakan algoritma untuk memecahkan suatu masalah [7].

Algoritma *Brute Force* memiliki beberapa kelebihan dan kekurangannya. Kelebihan dari algoritma ini adalah, hampir semua masalah dapat diselesaikan dengan algoritma ini. Algoritma ini juga bersifat sederhana sehingga dapat dengan mudah dimengerti. Sedangkan kekurangan dari algoritma *Brute Force* adalah dari kecepatan pemrosesannya. Algoritma ini memiliki kecepatan pemrosesan yang lambat jika dibandingkan dengan algoritma lainnya sehingga menjadi kekurangan besar dari algoritma ini [8].

Cara penyelesaian algoritma *Brute Force* sangatlah sederhana, tidak sama seperti algoritma lain yang memiliki tahapan-tahapan yang harus diikuti. Algoritma *Brute Force* memecahkan masalah secara langsung sesuai dengan permasalahan yang ada. Contoh kasus yang dapat diselesaikan dengan Algoritma *Brute Force* adalah pencarian atau pencocokan string dan pencarian nilai tertinggi atau terendah.



Gambar 2.1 Pencocokan *String*

Cara kerja algoritma *Brute Force* dalam penyelesaian pencarian *string* di atas adalah dengan membandingkan *string*, dalam hal ini secara satu persatu sampai didapatkan hasil yang diinginkan. Kata 'IPSUM' dicek dari awal teks, jika tidak cocok, maka digeser 1 kata untuk dicek kembali, lalu diulang sampai didapatkan kecocokan [8].

2.4 **Game Maker Studio 2**

Game Maker Studio 2 adalah sebuah *engine* yang dapat digunakan untuk membuat *game*. *Game Maker* memiliki beberapa fitur menarik yang membuat *game engine* ini digunakan banyak orang, yaitu memiliki fitur *drag and drop*, maupun juga memiliki bahasa pemrogramannya sendiri. Berdasarkan fitur-fitur tersebut *Game Maker Studio* menawarkan kemudahan dalam membuat *game*, baik untuk orang-orang yang sudah berpengalaman, orang-orang yang masih pemula, dan juga untuk orang yang tidak tahu bahasa pemrograman sekalipun [9]. *Game Maker* memiliki bahasa pemrogramannya sendiri yaitu *Game Maker Language* atau biasa disebut GML. Perbedaan bahasa GML dengan bahasa pemrograman lainnya ada pada kemudahan penggunaannya, misalnya bahasa pemrograman GML tidak terlalu memperhatikan sintaks pada pemrograman secara umum. Namun GML juga memiliki kelemahan terutama dalam pemrograman pada proyek dengan skala besar, salah satunya adalah batasan maksimum *array* yang hanya berjumlah 32 ribu elemen [10].

2.5 Metodologi Pengembangan Sistem

Metodologi pengembangan sistem yang digunakan pada pembuatan aplikasi ini adalah *Multimedia Development Life Cycle* (MDLC) dengan acuan berdasarkan metode penelitian *Research and Development* (R&D). Berdasarkan Luther-Sutopo, metode ini memiliki 6 tahapan namun hanya 5 saja yang akan digunakan dalam pembuatan *game* ini. Berikut adalah tahapan-tahapan tersebut [11]:

1. *Concept* (Konsep)

Pada tahapan ini akan ditentukan tujuan dan manfaat dari aplikasi yang akan dibuat, target pengguna aplikasi, serta mendeskripsikan secara rinci konsep dan *rule* dari aplikasi yang akan dibuat.

2. *Design* (Desain)

Pada tahapan ini akan dibuat rancangan spesifikasi dan fitur-fitur yang akan digunakan pada aplikasi serta diagram alur dan *storyboard* aplikasi.

3. *Material Collecting* (Pengumpulan Material)

Pada tahapan ini akan dilakukan pengumpulan bahan-bahan yang diperlukan dalam pembangunan aplikasi multimedia *file audio* dan desain yang akan digunakan. Tahapan ini biasanya dilakukan bersamaan dengan tahapan *assembly*.

4. *Assembly* (Pembangunan)

Pada tahapan ini dimulai pembuatan aplikasi sesuai dengan desain aplikasi dan objek yang sudah dibuat dan dikumpulkan.

5. *Testing* (Pengujian)

Tahapan ini akan dilakukan proses uji coba terhadap aplikasi yang sudah selesai dibuat.

6. *Distribution* (Distribusi)

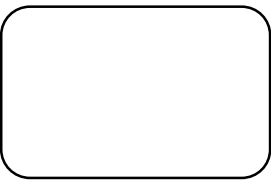
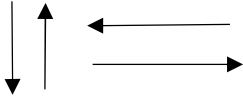

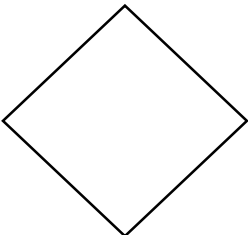
Pada tahapan ini, aplikasi yang sudah diuji akan didistribusikan ke dalam bentuk *file* sesuai platform aplikasi.


2.6 Kakas Pemodelan

Kakas pemodelan yang digunakan pada pembuatan aplikasi ini adalah *flowchart* atau diagram alur. *Flowchart* adalah sebuah diagram yang menunjukkan

alur dari sebuah proses atau aplikasi sesuai langkah-langkah yang ada untuk menghasilkan suatu *output*. Sebuah *flowchart* dimulai dengan *input* sebuah data kemudian menunjukkan semua proses serta langkah-langkah apa saja yang dilakukan sampai dihasilkan *output*. *Flowchart* juga menggunakan simbol-simbol yang memiliki artinya masing-masing untuk menggambarkan langkah-langkah yang dilakukan. Terdapat 5 simbol utama yang ada pada sebuah *flowchart*, simbol-simbol tersebut beserta dengan penjelasannya dapat dilihat pada Tabel 2.1 [12].

Tabel 2.1 Simbol *Flowchart* [12].

Nama	Simbol	Fungsi
<i>Terminal Point</i>		Menunjukkan mulai serta berakhirnya aktivitas.
<i>Flow lines</i>		Garis alur proses, menghubungkan suatu langkah, ke langkah lainnya.
Aktivitas/Proses		Menunjukkan suatu proses yang dilakukan.
Keputusan		Menunjukkan suatu keputusan yang akan diambil dari suatu proses.

Nama	Simbol	Fungsi
<i>Input/Output</i>		Menunjukkan tempat terjadinya proses <i>input</i> dan <i>output</i> .

Dalam pembuatan sebuah *flowchart*, terdapat beberapa langkah utama yang harus diperhatikan. Pertama, definisikan proses-proses yang ada dan mengidentifikasi batasan dari *flowchart*. Kemudian, buat langkah-langkah dan proses yang ada pada *flowchart* beserta dengan relasi dari langkah-langkah tersebut. Selanjutnya, gabungkan semua langkah-langkah yang telah ditentukan sesuai dengan alur dari sistem atau aplikasi yang dibuat, serta lakukan pemeriksaan kembali untuk memastikan *flowchart* tersebut benar [12].

2.7 Studi Kasus yang Terkait

Terdapat beberapa studi kasus yang terkait mengenai pembuatan *game* RPG *turn-based* dan penggunaan algoritma *Brute Force*. Studi kasus tersebut digunakan sebagai inspirasi dan alasan pembuatan *game* ini. Studi kasus yang diambil adalah mengenai pembangunan *game* RPG *turn-based* dan penggunaan algoritma *Brute Force* dalam membandingkan solusi-solusi yang terbaik.

Pada studi kasus yang pertama [13] *game* tersebut menggunakan *genre* RPG dengan pertarungan secara *turn-based* melawan musuh menggunakan sistem permainan batu-gunting-kertas. Dengan menggunakan sistem tersebut, maka sifat dari pertarungan tersebut menjadi acak dan tidak memiliki perhitungan.

Pada studi kasus kedua [14], terdapat pertarungan yang dilakukan oleh pemain dengan monster-monster yang ditemui dengan menggunakan pertarungan *turn-based*. Namun musuh yang dilawan hanya memiliki satu gerakan saja yaitu menyerang.

Pada studi kasus ketiga [15], terdapat dua algoritma yang digunakan dalam proses pemfaktoran RSA. Algoritma *Brute Force* dipilih sebagai algoritma yang

lebih cocok dalam masalah tersebut karena hasil yang didapatkan lebih akurat, walaupun memiliki proses yang lebih lama.

Dalam pembuatan *game* RPG *turn-based*, salah satu algoritma yang dapat digunakan adalah algoritma *Brute Force*. Namun pada *game turn-based* yang berskala kecil, musuh yang dilawan cenderung hanya memiliki satu gerakan saja, atau memiliki pemilihan gerakan yang acak. Algoritma *Brute Force* sudah jarang digunakan pada *game turn-based* yang sudah kompleks karena sudah menggunakan algoritma yang lebih efisien untuk mengurangi proses yang dilakukan walaupun hasil akhir yang didapatkan tidak terlalu tepat. Tapi untuk *game* yang berskala kecil, algoritma *Brute Force* dapat digunakan untuk mendapatkan solusi terbaik dengan waktu yang cepat sehingga dapat membuat sebuah AI pada *game* berskala kecil yang dapat memilih gerakan yang terbaik. Oleh karena itu, pada *game* RPG *turn-based* yang akan dibuat ini, akan diterapkan algoritma *Brute Force* sebagai perbandingan gerakan terbaik oleh AI musuh.

BAB III

ANALISIS

Bab ini akan berisi tahapan *concept* pada metode MDLC (*Multimedia Development Life Cycle*) berupa identifikasi persyaratan dasar aplikasi untuk penentuan tujuan dan manfaat aplikasi, target pengguna aplikasi, konsep, *rule* dan algoritma dari aplikasi.

3.1 Identifikasi Persyaratan Dasar Aplikasi

Pada bagian ini akan dilakukan analisis persyaratan-persyaratan dasar aplikasi, seperti konsep dasar *game*, mekanik dan *rule game* dan spesifikasi teknologi.

3.1.1 Analisis Game Serupa

Untuk menentukan persyaratan-persyaratan dasar aplikasi, akan dilakukan analisis terhadap beberapa *game* serupa, dalam hal ini memiliki *game* yang memiliki *genre Turn-Based*. Hal-hal yang akan dianalisis adalah konsep dari *game*, fitur-fitur yang dimiliki, dan perilaku AI *game* pada saat melawan pemain. Pada bagian ini, akan diambil 3 buah *game* yang akan dianalisis, yaitu *Pokémon Sword and Shield*, *Civilization VI*, dan *Final Fantasy XIII*. Berikut ini adalah penjelasan mengenai ketiga *game* tersebut :

1. Pokémon Sword and Shield

Game ini memiliki konsep di mana pemain menggunakan karakter yang hidup pada dunia penuh dengan monster-monster yang unik, namun monster-monster tersebut dapat ditangkap dan dibuat menjadi *partner* dari pemain [16].

Tujuan dari *game* ini adalah menangkap semua jenis pokemon yang ada dan menjadi *trainer* pokemon terbaik. *Game* ini memiliki dua jenis *gameplay*, yaitu mode *semi open-world* di mana pemain dapat berjalan-jalan di kota dan desa, mengambil misi dan melakukan interaksi yang bermacam-macam, salah satunya adalah menantang untuk melakukan pertarungan pokemon, serta mode kedua yaitu *turn-based battle* di mana pemain melakukan pertarungan pokemon dengan pokemon lainnya [16].

Pada pertarungan pokemon, pemain dan juga lawannya memiliki pokemon dengan elemen yang berbeda-beda yang memiliki keuntungan dan kekurangan masing-masing. Setiap pokemon memiliki gerakan yang berbeda-beda, dan pemain maupun AI dapat memilih gerakan yang diinginkan untuk memukul musuh pada tiap gilirannya [16].



Gambar 3.1 Tampilan *Gameplay Semi Open-World Pokemon* [16]



Gambar 3.2 Tampilan *Gameplay Pokemon Battle* [16]

2. *Civilization VI*

Game ini memiliki konsep di mana pemain memiliki peran sebagai pemimpin suatu peradaban. Pemain akan memulai permainan dengan peradaban