

Pembangunan game “Recon Duty” dengan Godot Engine

Arkas Dyanta Ananda Kautsar^{*1}, Tommy Bustomi², Muhammad Farman Andrijasa³

^{1,2,3} Program Studi Teknik Informatika Multimedia, Politeknik Negeri Samarinda, Samarinda, Indonesia

INFORMASI ARTIKEL

Diterima Redaksi: 21 May 2024
Revisi Akhir: 22 June 2024
Diterbitkan Online: 01 July 2024

KATA KUNCI

Game, pathfinding, Godot, Top-Down, GDLC

Keywords:

Game, pathfinding, Godot, Top-Down, GDLC

KORESPONDENSI

E-mail: arkasdyanta1@gmail.com

A B S T R A K

Game “Recon Duty” merupakan *game shooting* dengan sudut pandang kamera dari atas melihat ke bawah dan juga tampilan visual *Pixel Art* pada setiap objek, makhluk, dan lingkungan. Game ini dirancang dengan menggunakan *Godot engine* dengan bahasa *GDScript* sebagai bahasa pemrograman umum di aplikasi *game engine* tersebut. Beberapa tahapan dari *Game Development Life Cycle* (GDLC) terdiri dari inisiasi, *Pre-Produksi*, *Produksi*, *Evaluasi*, dan *Publikasi*. Penerapan *Artificial Intelligence* (AI) pada *Non-Playable Character* (NPC) menggunakan metode *pathfinding* untuk mencari jalan menuju *target*. Algoritma *A-Star* pada *Non-Playable Character* (NPC) bertujuan menentukan jalur pendek dari sekitar lingkungan, sehingga dapat mencapai tujuan. Penelitian ini menghasilkan sebuah *game Recon Duty* yang dapat digunakan sebagai sarana hiburan bagi masyarakat.

A B S T R A C T

The game “Recon Duty” is a shooting game with a camera angle from above looking down and also a *Pixel Art* visual display on every object, creature, and environment. This game is designed using the *Godot engine* with the *GDScript* language as a general programming language in the game engine application. Several stages of the *Game Development Life Cycle* (GDLC) consist of initiation, *Pre-Production*, *Production*, *Evaluation*, and *Publication*. The application of *Artificial Intelligence* (AI) to *Non-Playable Characters* (NPC) uses the *pathfinding* method to find a way to the target. The *A-Star* algorithm on *NonPlayable Character* (NPC) aims to determine the shortest path from the surrounding environment, so that it can reach the destination. This research produces a *Recon Duty* game that can be used as a means of entertainment for the community.

PENDAHULUAN

Perkembangan teknologi di era globalisasi ini semakin maju pada inovasi baru dalam berbagai segi kehidupan yang ditandai dengan adanya perkembangan teknologi di berbagai bidang. Teknologi tidak hanya untuk meringankan pekerjaan manusia, namun juga sebagai penghibur manusia salah satunya yaitu *video game*. *Video game* ialah suatu permainan yang bersifat elektronik yang berkaitan dengan hubungan antar muka dan pengguna untuk mendapatkan umpan balik yang bersifat visual. Kata *video game* bisa disebut sebagai layar *raster*. Akan tetapi lebih sering disebut sebagai *video game*. Pada saat ini *video game* dipakai pada perangkat layar dalam berbagai bentuk permainan [1].

Video game juga memiliki macam-macam model permainan seperti *Role-Playing Game* (RPG), *Action*, *Puzzle*, *Strategi*, *Fighting*, *Adventure*, dan sebagainya. Model Permainan atau *bergenre* adalah sebuah kategori dari *game* berdasarkan tantangan dalam permainan, terlepas dari latar sebuah *video game* [2].

Genre Top-Down Shooter merupakan sebuah *genre game* tembak menembak dengan ciri khas pemain melihat karakter dan serta lingkungan dari sudut pandang atas. *Genre Top-Down shooter* ambil beberapa tema ialah *modern* dan *Science fiction*.

Genre ini telah menghadirkan pengalaman bermain *game* dengan pilihan sebagai tema contoh aksi, horor, simulasi, *historical*, strategi, dan sebagainya. Ada yang lebih menarik untuk dipilih yaitu penggabungan beberapa sub-*genre* ke dalam *genre* yaitu *shooting* dan *action*. *Shooting* adalah *genre* berfokus dimana *player* menembaki musuh dengan kecepatan reaksi pemain. *Action* adalah *genre* berfokus tantangan fisik yang memerlukan koordinasi tangan-mata dan refleks yang cepat.

Game engine adalah sebuah perangkat lunak yang digunakan untuk mengembangkan *game*. *Game engine* memiliki fitur seperti suara, grafik, animasi, dan *source code* atau disebut juga sumber program. *Source code* ialah bahasa pemrograman berada di komputer yang dapat dibaca oleh manusia. Pada pembahasan pengembangan *game* ini, ada sebuah perangkat lunak *game engine* yang gratis dan mudah mengembangkan *game* dengan dukungan bahasa pemrograman *GDScript*, C#, C++ yaitu *Godot engine*. *Godot engine* dipilih oleh pengguna karena penyediaan fitur yang lengkap dan pemrograman *GDScript* dapat mempermudah pembuatan sistem *game* dalam pembangunan *game*.

Recon Duty adalah *Game Pixel* yang ber-*genre action-shooting* yang dirancang menggunakan *Godot Engine*. *Game* tersebut bercerita tentang seorang *agent* melakukan operasi membasmi para penjahat didalangi oleh organisasi terkuat. *Recon Duty* adalah *game* yang dapat di *platform* PC (*Personal Computer*) yang berbentuk *genre Top-Down shooter* dalam permainan tersebut.

Berdasarkan uraian diatas, peneliti ingin merancang dan membangun sebuah *game* yang akan terdapat tokoh enemy yang akan menjadi *Non-Player Character* (NPC), yang mana di dalam *game* tersebut *Non-Player Character* (NPC) akan dirancang untuk mencari target utama atau pemain dengan rute terpendek dan dapat menghindari penghalang yang ada di area *game* untuk memperoleh kemenangan dan akan melanjutkan level atau tahap *game* berikutnya dengan tantangan yang lebih meningkat. Untuk itu peneliti akan menggunakan algoritma A^* yang akan diterapkan pada kecerdasan karakter pada *game* yang berjudul "PEMBANGUNAN GAME "RECON DUTY" DENGAN GODOT ENGINE". Dalam penelitian ini menerapkan metode *Game Development Life Cycle* (GDLC) dapat lebih efisien dan efektifitas pada pembangunan *game*, dengan ini dapat menghasilkan kualitas pada proyek *game*.

TINJAUAN PUSTAKA

1. Video game

Game dapat diartikan sebagai kegiatan yang bertujuan untuk memperoleh kesenangan. Meskipun tujuan utama dari *game* adalah untuk bersenang-senang, namun beberapa *game* mungkin memiliki tujuan tertentu, seperti mendidik atau memperluas wawasan. *Game* dijabarkan ke dalam berbagai bentuk dan salah satunya adalah *video game*, yaitu permainan elektronik yang dimainkan mengendalikan gambar di layar video [3].

2. Dimensi Visual Game

Dimensi adalah angka yang terkait dengan sifat metrik atau topologi dari suatu benda. Berdasarkan dimensi dari objek-objeknya, *game* dibedakan menjadi beberapa jenis, antara lain [4]:

a. Game 2D

Game dua dimensi atau 2D merupakan suatu konsep di mana semua objek berada pada satu permukaan datar. Pergerakan dalam *game 2D* dibatasi secara *horizontal* dan *vertikal* atau secara koordinat gerakan pemain yang hanya dapat bergerak sepanjang sumbu X dan Y [4].

b. Game 3D

Game 3D adalah *game* yang secara matematis hanya menggunakan 3 elemen koordinat (x, y, dan z). Oleh karena itu, konsep kamera dalam visual *game* mirip dengan dunia nyata. Artinya kamera juga berputar pada sumbu tertentu [4].

3. Perancangan Game

Tentunya sebelum Anda membuat sebuah *game*, Anda memerlukan tahap desain untuk membuat *game* yang Anda inginkan. Tahapan pembuatan *game* dibagi sebagai berikut [3]:

a. High Concept

High concept memungkinkan anda mempertahankan fokus pengembangan *game* yang ada dan menghindari dari konsep yang bertentangan dengan desain *game*.

b. Gameplay

Gameplay merupakan pengalaman interaksi pemain dengan *game* saat bermain.

c. Audience

Audince dalam perancangan *game* mengacu pada kelompok atau individu sebagai konsumen sasaran utama.

d. Hardware and software

Perangkat keras (*hardware*) merupakan sekumpulan perangkat bagian fisik suatu komputer. Di sisi lain, perangkat lunak (*software*) merupakan perangkat kumpulan data yang disimpan dan dikendalikan oleh perangkat tanpa bentuk fisik yang terdapat di dalam komputer sebagai yang menjalankan pengoperasian sistem.

e. Proses pembuatan game

Seluruh konsep yang telah dirancang pada implementasi ke dalam *game* dengan membuat atau mengumpulkan aset dan skrip.

f. Testing/Prototyping

testing yang dilakukan untuk menentukan apakah hasil akhir *game* sudah sesuai dan untuk mencari konsep *error* atau *bug* pada *game* tersebut.

4. Top-Down

Top-down adalah perspektif permainan dimana protagonis dapat bergerak ke empat arah, namun *gameplay*-nya sendiri bergerak dari bawah ke atas. *Game* yang menggunakan perspektif *game* jenis ini biasanya adalah *game shooter* [5].

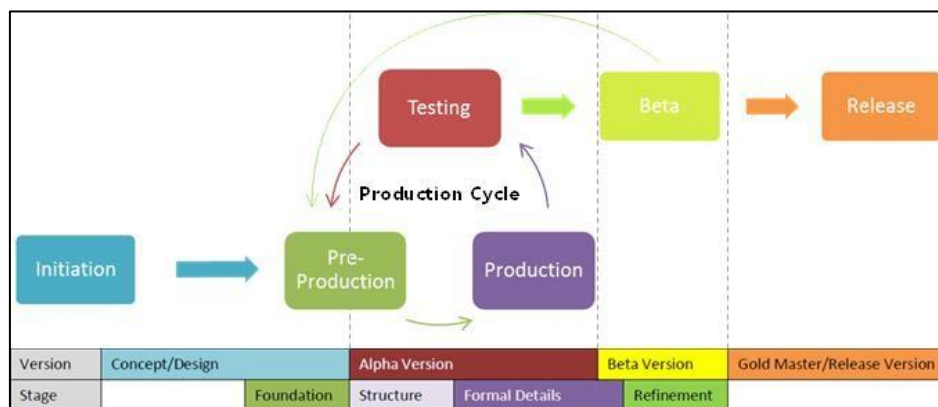
5. Godot Engine

Yang dimaksud *Godot Engine* adalah aplikasi yang menyediakan alat untuk mendukung pengembangan video *game* yang dirilis secara gratis sebagai *open source* (*Godot Engine* - Gratis dan *Open source* 2D dan 3D *Game Engine*, belum dikonfirmasi) [6]. Bahasa pemrograman khusus dari mesin *Godot* adalah *GDScript*.

Aplikasi ini didukung oleh sistem *node* dan *scene*. *Video game* membuat objek dan dijelaskan oleh *node*. *Node* ditempatkan dan ditempatkan di adegan lain [6].

6. Game Development Life Cycle

Game Development Life Cycle (GDLC) adalah pendekatan pengembangan *game* yang berasal dari *Software Development Life Cycle* (SDLC). *Game Development Lifecycle* (GDLC) merupakan proses pengembangan *game* yang menggunakan pendekatan berulang yang terdiri dari enam fase: inisiasi, praproduksi, produksi, pengujian, beta, dan rilis [7].



Gambar 1. Fase Metode GDLC

Sumber referensi: Arief Assaf, Muhammad Miftah dan Amin Fakhri (2023)

7. Flowchart

Diagram alur adalah diagram yang menunjukkan urutan dan hubungan antara suatu proses dan instruksinya. Fungsinya untuk memberikan gambaran mengenai program atau aplikasi yang dibuat [8]. *Flowchart* membantu para analis dan pemrogram untuk membuat suatu gambaran aplikasi yang dibuat dan memecah menjadi segmen-segmen yang lebih kecil sehingga lebih mudah dievaluasi, dengan ini *flowchart* dapat memudahkan penyelesaian suatu masalah [9].

8. Pathfinding

Pathfinding merupakan salah satu konsep inti dari algoritma yang digunakan untuk menggerakkan karakter dalam *game*. *Pathfinding* adalah pencarian rute tercepat dari suatu titik awal sampai ke suatu tujuan, dengan menghindari rintangan-rintangan di sepanjang jalan. *Pathfinding* merupakan salah satu masalah paling mendasar dari kecerdasan buatan dalam *game* [10].

9. Artificial Intelligence

Kecerdasan buatan (AI) adalah bidang ilmu dan teknik komputer yang berkaitan dengan pemahaman komputasi tentang apa yang biasa disebut sebagai perilaku cerdas dan penciptaan artefak yang menunjukkan perilaku tersebut [11].

Defender of Metal City adalah *game* menara pertahanan yang dirancang menggunakan AI untuk meningkatkan kegembiraan pemain melalui lawan AI dengan tingkat kesulitan yang sedang [12].

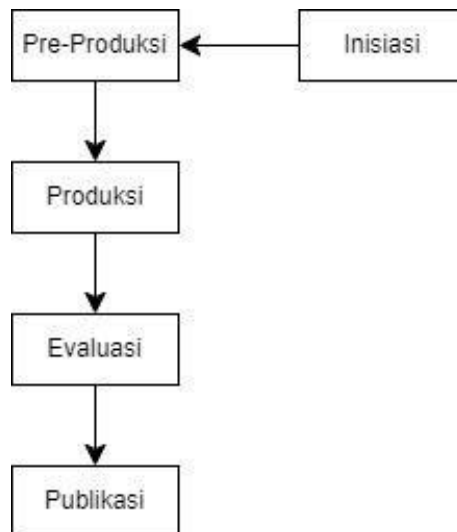
10. Algoritma A-Star

A-Star adalah algoritma pencarian generic yang dapat digunakan untuk menemukan solusi untuk banyak masalah, mencari jalan hanya menjadi satu dari fungsi algoritma tersebut [13].

METODOLOGI

1. Tahap Penelitian *Game Development Life Cycle*

Penelitian berdasarkan metode merujuk pada GDLC (*Game Development Life Cycle*). Maka itu, dilakukan analisis sesuai gambar 2.



Gambar 2. Alur diagram GDLC

a. Tahap Inisiasi

Pada tahapan ini adalah membuat konsep dasar *game* seperti apa yang akan diproduksi dan deskripsi *game* yang sangat sederhana.

b. Tahap Pre-Produksi

Pre-produksi melibatkan tahapan merevisi desain *game* dan pembuatan prototipe *game*. Fokus pada desain *game* menentukan karakter, lingkungan (*map tileset*), mekanik, *flowchart*, *Object tile*, dan *concept board* ke dalam dokumen desain *game*.

c. Tahap Produksi

Produksi adalah tahap proses pembuatan aset, pembuatan scene, menerapkan bahasa program, dan integrasi elemen. Terhubung prototipe dalam tahap ini adalah detail formal dan perbaikan. Detail formal adalah struktur yang disempurnakan dengan lebih banyak mekanik dan aset lengkap.

d. Tahap Evaluasi

Pada tahap evaluasi dilakukan 2 kali yakni pemeriksaan internal dan eksternal. Tes internal dilakukan untuk mengonfirmasi fungsionalitas operasional dan kinerja *game*. Sedangkan pengujian oleh pihak ketiga atau pihak luar kini dikenal dengan *beta testing*.

e. Tahap Publikasi

Tahapan akhir dari satu siklus pengembangan adalah merilis aplikasi *game* dirilis ke publik. Publikasi melibatkan peluncuran produk proyek, kumpulan dokumen proyek untuk pemeliharaan dan perluasan *game*.

2. Alat dan Bahan

Alat yang digunakan dalam penelitian ini terdiri dari perangkat keras dan perangkat lunak:

- a. Perangkat keras:
 1. Intel(R) Coren (TM) i7-10750H CPU @ 2.60GHz 2.59 GHz
 2. NVIDIA Geforce GTX 1650
 3. Besar memori RAM 16 GB
 4. Kapasitas hardisk 500 GB + 1TB SSD
- b. Perangkat lunak:
 1. Sistem Operasi Windows 11
 2. Godot engine
 3. Adobe Illustrator
 4. Adobe Photoshop
 5. Aseprite

Bahan-bahan penelitian terdiri dari:

- a. Model
- b. Animasi
- c. Objek
- d. Effect
- e. Sound Effect

3. Perhitungan Algoritma A-Star

Perhitungan *A-Star* adalah bagaimana cara menghitung algoritma *A-Star* menentukan jalan yang paling terpentuk untuk mencapai tujuan, berikut rumus yang digunakan dari algoritma *A-Star*:

$$f(n) = g(n) + h(n) \quad (1)$$

dimana:

$f(n)$ = biaya estimasi terendah,

$g(n)$ = biaya dari node awal ke *node n*,

$h(n)$ = estimasi biaya dari *node n* untuk mencapai *node* akhir.

HASIL DAN PEMBAHASAN

1. Tahap inisiasi

Berikut ini deskripsi *game* yang dibuat oleh penelitian:

Nama Game	: Recon Duty
Tema	: Operasi pasukan khusus
Karakter Game	: Seorang <i>Agent</i> dengan pakaian perlengkap <i>tactic special force</i> dan senjata yang digunakan. Musuh-musuh yang dihadapi oleh <i>Agent</i> diantara kelompok <i>militant</i> , tentara bayaran, dan tentara korup. <i>militant</i> adalah kelompok kecil yang dengan penampilan pakaian sederhana dan mengenakan rompi anti peluru. <i>Militant</i> hanya muncul di <i>level 1</i> . Sedangkan tentara bayaran berpenampilan pakaian lengkap seperti pasukan khusus dengan warna hitam. Tentara bayaran berada <i>level 2</i> . Dan terakhir, tentara korup penampilan pakaian tentara pada umumnya. Tentara korup dan <i>smile mask</i> muncul di <i>level 3</i> .
Skenario Cerita Game	: Seorang <i>Agent</i> melakukan misi dan menghadapi berbagai kelompok militan, tentara bayaran (<i>mercenaries</i>), serta tentara korup yang memiliki

afiliasi dengan *Scarath Circle*, organisasi misterius yang terlibat dalam serangan tragedi di dunia melalui kelompok afiliasinya. Organisasi *Scarath Circle* mempunyai pasukan sendiri yang bernama *Smile Mask*, *Smile Mask* bertugas membantu kelompok yang telah berafiliasi dengan organisasi tersebut. *Agent* berusaha menghentikan kelompok dan pasukan yang afiliasi dengan *Scarath Circle* yang mencoba menciptakan tragedi dan menguasai dunia.

<i>Genre Game</i>	: <i>Shooting dan Action</i>
Prespektif Kamera	: <i>Top-Down</i>
Grafis Visual	: <i>Pixel Art</i>
<i>Game Engine</i>	: <i>Godot Engine</i>
Bahasa Pemrograman	: <i>GDScript</i>
<i>Platform</i>	: Dipublikasikan di situs <i>web itch.io</i>

2. Tahap Pre-Produksi

a. Karakter

Penampilan karakter yang akan didefinisikan pada setiap karakter masing-masing. *Agent* dengan karakteristik berpenampilan topeng ninja tapi hanya bagian mulut, hidung dan pipi, sedangkan seragam berpakaian lengkap seperti pasukan khusus dengan warna hitam dan warna sedikit hijau di beberapa bagian seragam.



Gambar 3. Tampilan karakter *agent*

Untuk penampilan militan, karakteristik dengan baju hijau dan rompi anti peluru, celana berwarna hitam dan sepatu warna coklat, dan topeng bandage warna ungu dan putih pada. Pasukan *militant* sebagai musuh dan akan muncul di *level 1, 1.2, 1.3*. Penampilan *militant* yang telah dijelaskan sesuai dengan karakteristik pada gambar 4.



Gambar 4. Tampilan karakter *Militant*

Pada penampilan tentara bayaran (*mercenary*), berseragam lengkap seperti pasukan khusus dengan tambahan topeng ninja, dan kacamata hitam dengan warna hitam, rompi anti peluru berwarna pasir orange, sedangkan helm berwarna hitam dan bagian titik sisi kiri dan kanan berwarna merah, dan *bandage* leher berwarna merah. Pasukan tentara bayaran (*mercenary*) sebagai musuh dan akan muncul di *level 2*.



Gambar 5. Tampilan karakter Tentara bayaran

Penampilan tentara korup, sama seperti penampilan tentara bayaran tapi ubah warna merah loreng pada helm, baju dan celana. Dan terakhir, *smile mask* digambarkan penampilan baju kantor di dalam jas hitam, dasi merah, dan topeng dengan emoji tertawa. Pasukan tentara korup dan *smile mask* sebagai musuh dan akan muncul di *level 3*.

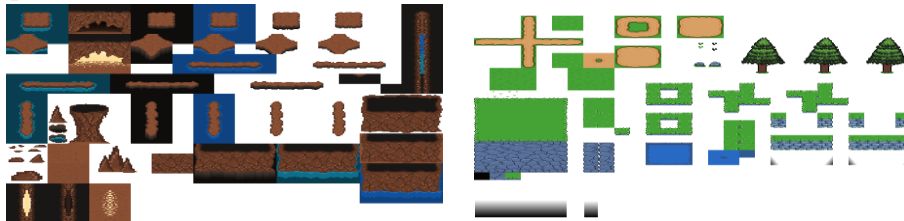


Gambar 6. Tampilan karakter Tentara korup



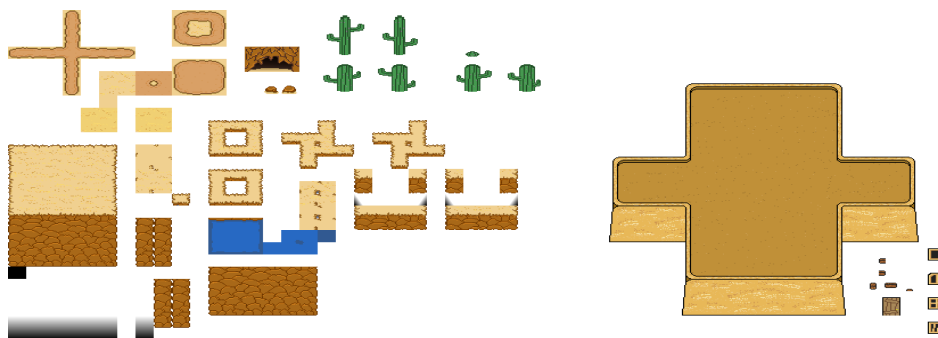
Gambar 7. Tampilan karakter *Smile mask*

b. Map tileset



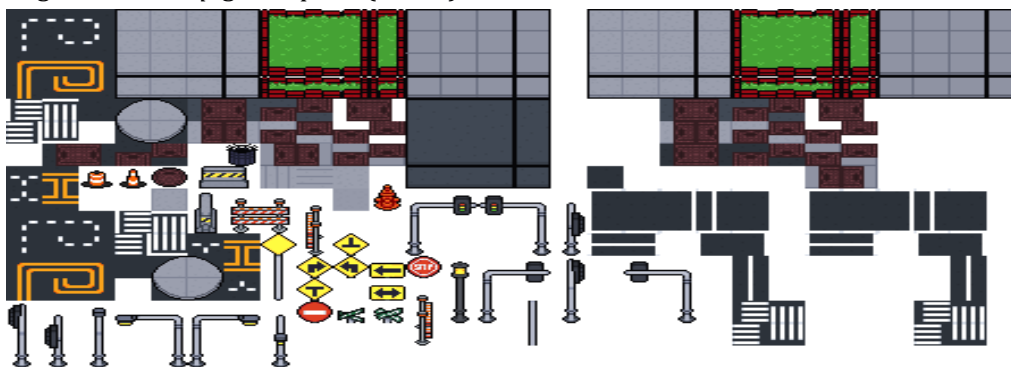
Gambar 8. Map Tileset hutan (*forest*) dan goa untuk *level 1*

Tilemap dengan daerah hutan dan di dalam goa. Tilemap hutan menunjukkan desain pada kumpulan *tile* seperti rumput hijau, bukit, tempat air (danau, laut, dan sungai), dan tempat jalan kaki. Sedangkan *tilemap* goa menunjukkan desain pada kumpulan *tile* seperti lubang goa, dinding, air berbagai warna, tanah goa, berbatuan di dalam goa, dan jurang dalam. Dua *tilemap* ini terdapat pada 3 *scene* di *level 1*, diantara dari daerah hutan pada *scene* satu dan dua, sedangkan goa pada *scene 3*. Khusus *tilemap* hutan dapat dijadikan sebagai *scene level* lain seperti *level 3*.



Gambar 9. Map Tileset gurun pasir (*desert*) untuk level 2

Tilemap dengan daerah gurun pasir (*desert*). *Tilemap* gurun pasir (*desert*) menunjukkan desain pada kumpulan *tile* seperti pandang pasir, bukit gurun pasir, tempat air (danau, laut, dan sungai), dan tempat jalan kaki gurun. *Tilemap* gurun pasir (*desert*) dimasukkan ke *scene* level 2.



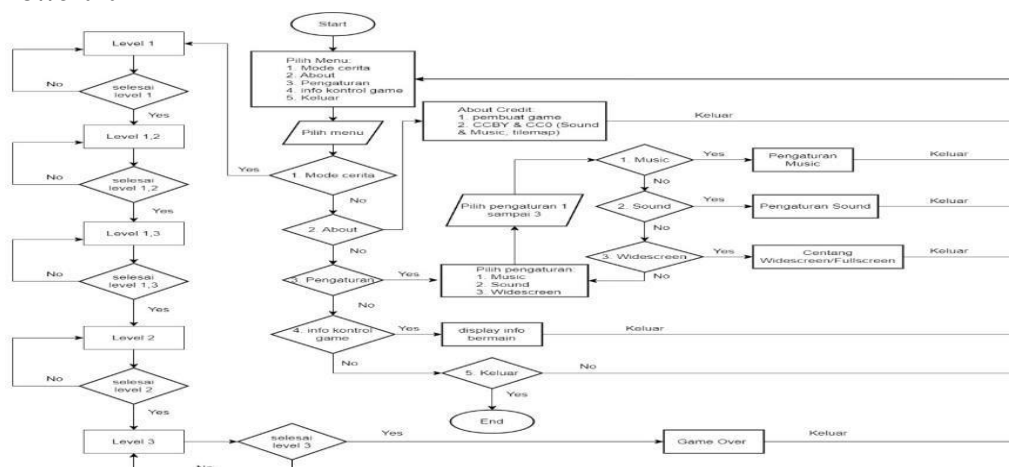
Gambar 10. Map Tileset kota (*city*) untuk level 3

Tilemap jalanan perkotaan. *Tilemap* jalanan perkotaan menampilkan desain peta diantaranya jalan trotoar, jalan khusus kendaraan transportasi, dan lantai gedung. *Tilemap* jalanan perkotaan dimasukkan ke *scene* level 3.

c. Mekanik

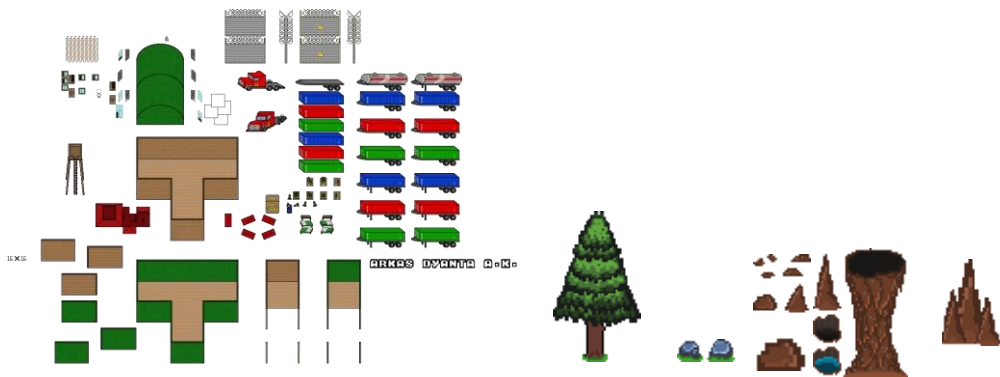
Mekanisme *game* pada *player* utama menggunakan *key* “WASD” untuk pergerakan jalan *player* mencapai tujuan dan *mouse* sebagai membidik dan menembaki pasukan musuh. *Game* terdiri dari 3 *level* dengan tingkat kesulitan berdasarkan *level* yang akan dimainkan. Setiap *level* yang sudah ditempuh, maka *level* semakin naik tingkat kesulitan *level*. Musuh diletakkan sisi awal sampai titik akhir dan bila *player* sudah sampai tujuan, maka misi telah selesai.

d. Flowchart



Gambar 11. Flowchart Game “Recon Duty”

e. Object tile



Gambar 12. Object tile hutan, markas, dan goa untuk level 1

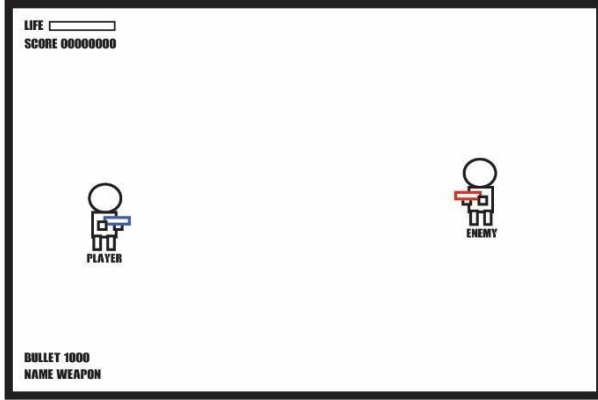
Beberapa bagian *tile* di dalam *tilemap* yang akan diletakkan pada *layer* berikut di *scene level 1*. *Tiles* ini diambil dari *tilemap* markas, hutan, dan dalam goa. *Tilemap* markas pada beberapa *tiles* terdiri dari bangunan-bangunan, kamp peristirahatan, kotak kayu, truk serta kontainer, pagar, dan sebagainya. Pada penyusunan bangunan dan kamp peristirahatan terdapat ada struktur pada bangunan-bangunan seperti pintu kayu, jendela, dan atap rumah. Dan di dalam bangunan mempunyai objek struktur di antaranya tempat tidur, karpet, dan meja. Pada *tilemap* hutan, terdapat ada dua jenis *tiles* diantaranya pohon dan batu. Sedangkan *tilemap* goa, *tiles* ini terdapat ada beberapa jenis bebatuan dan lubang kecil di dalam goa. Khusus *object tile* markas dan hutan, dapat dilakukan menambahkan *scene level* lain.

f. Concept board

Concept board bertujuan mengambil konsep gambar yang sudah dibuat akan dijadikan contoh gambar, lalu implementasikan menjadi desain gambar.

Tabel 1. Tampilan papan konsep *game*

Judul Concept board	Gambar Concept board
Menu Utama	The main menu screen features a central title 'RECON DUTY' in a blue-bordered box. To the left, there is a vertical stack of buttons: 'HOW TO PLAY', 'STORY MODE', 'ABOUT', 'SETTINGS', and 'EXIT'. On the right side, the text 'GAMBAR ATAU ANIMASI GERAK BACKGROUND' is displayed in orange.
How to play	The 'HOW TO PLAY' screen includes a 'BACK' button at the top left and an 'EXIT' button at the bottom. The main area contains a control diagram with labels for 'MAIN MENU', 'RESTART', 'MOVE LEFT', 'MOVE RIGHT', 'MOVE FRONT', 'MOVE BACK', 'MOVE AIM', and 'SHOOT'. The movement keys are represented by a set of four buttons labeled 'W', 'A', 'S', and 'D'. A small diagram of a character's movement and aiming is shown on the right.

Judul Concept board	Gambar Concept board
Gameplay	

3. Production

a. Pengumpulan Asset

Tabel 2. Material fonts

Nama	Sumber
<i>Joystix</i>	<i>dafont.com</i>
<i>Metal-slug</i>	<i>fontstruct.com</i>

Huruf *font regular* digantikan dengan *font Joystix* dan *metal slug. font* tersebut dapat dari sumber yang di cantumkan pada tabel ini. Dari angka sampai huruf mengubah sesuai dengan *font* yang baru digantikan. Setiap kumpulan *asset design* dalam *folder* terdapat beberapa *asset image*.

b. Pembuatan Scene



Gambar 13. Tampilan scene level 1

Hasil pembuatan peta pada *scene level 1*, sedangkan gambar 4.18. menampilkan hasil pembuatan peta pada *scene level 1.2*. Titik awal menuju titik tujuan dengan mengikuti pada *tile* jalan dan aset tanda panah untuk membantu arah tujuan. Di tengah jalan ada dua jalur yaitu markas kecil dan seberang jurang laut, dan bagian akhir pada di tengah laut ada 3 pulau dengan jurang jembatan sebagai jembatan menuju ke titik tujuan. Sedangkan di gambar 4.18. titik awal berada di luar markas, untuk mencari jalan dengan mengikuti *tile* jalan dan aset tanda panah. Setelah masuk wilayah markas musuh, melewati parkir truk dan *container*. Setelah melewati parkir, ada terdapat labirin pada *scene level 1.2*, letak labirin berada di dalam markas musuh. Setelah melewati labirin, menemukan lubang goa sebagai titik tujuan. Kedua gambar *scene* peta ini mempunyai *asset tilemap* yang sama. *Scene* ini hasil dari penyusunan *tilemap* hutan pada *layer* pertama, dan *asset* markas serta hutan pada *layer* berikutnya. Pada perpohonan yang banyak dan tebing bukit sebagai tembok *physic* pada *player* dan *enemy* untuk tidak keluar dari peta. Sedangkan pagar, bangunan, aset militer, truk dan *container* merupakan *objek tile physic* pada *scene* ini. Dan pada *tile* berair terdapat ada *physic* pada *tile* tersebut.



Gambar 14. Tampilan scene main menu

Scene menu utama berada di scene pertama saat memulai game. Di dalam scene menu utama terdapat ada beberapa tombol diantara lain tombol *start game*, *about*, *settings*, *exit*, dan *how to play* yang mempunyai transformasi scene masing-masing setiap tombol. Pada *background menu* utama mendapat hasil dari *screenshot* selama uji coba main game. Judul game berada ditengah atas dengan bernama *Recon Duty*.

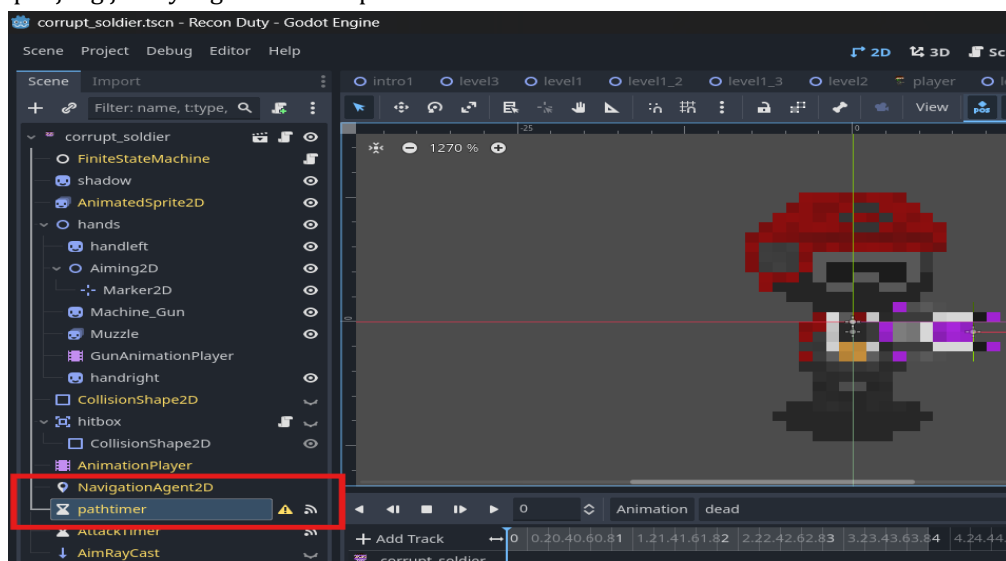


Gambar 15. Tampilan scene the end

Setelah level terakhir diselesai, scene selanjutnya akan menampilkan *scene ending* sesuai dengan gambar 4.26. scene ini berisi kalimat penutup dan pencipta. Tombol *home* merupakan tombol transformasi scene ke scene main menu.

c. Penerapan AI menggunakan Algoritma Pathfinding

Di aplikasi *Godot engine*, di dalam scene enemy ada node yang mempunyai hubungan dengan *pathfinding*. Node *pathfinding* terdapat di *godot engine* adalah *NavigationAgent2D* dan *timer*. *NavigationAgent2D* berfungsi sistem navigasi pada objek game 2D untuk bergerak otomatis dalam lingkungan telah ditentukan. *Timer* mengubah nama menjadi *pathtimer*, berguna untuk mengontrol kecepatan pergerakan objek sepanjang jalur yang telah ditetapkan.



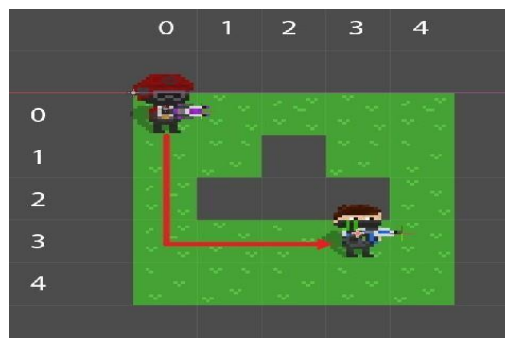
Gambar 16. Node NavigationAgent2D dan pathtimer (timer)

4. Perhitungan Algoritma A-Star

Tabel 3. Tabel perhitungan algoritma A-Star

No.	posisi	Langkah	g(n)	h(n)	f(n)
1.	(0,0)	NPC berdiam ditempat	0	6	6
2.	(1,0)	NPC tidak berjalan ke kanan	1	5	6
3.	(0,1)	NPC berjalan ke bawah	1	5	6
4.	(1,1)	NPC tidak berjalan ke kanan	2	4	6
5.	(0,2)	NPC berjalan ke bawah	2	4	6
6.	(0,3)	NPC berjalan ke bawah	3	3	6
7.	(0,4)	NPC tidak berjalan ke bawah	4	4	8
8.	(1,3)	NPC berjalan ke kanan	4	2	6
9.	(1,4)	NPC tidak berjalan ke bawah	5	3	8
10.	(2,3)	NPC berjalan ke kanan	5	1	6

Penghitungan algoritma A-Star pada lingkungan sekitar *Non-Playable Character* (NPC) dapat dilihat pada $f(n) = 6$, *Non-Playable Character* (NPC) akan berjalan ke posisi sesuai lokasi bila perhitungan sama pada $f(n) = 6$. Terdapat persamaan pada $f(n)$ dalam perbandingan seperti (1,1), (0,2). Dibagian (1,1) tidak dapat sebagai jalur yang dilewati karena di bawah dan kanan terdapat dinding, sehingga (0,2) sebagai jalur pendek untuk NPC (*Non-Playable Character*). Dibandingkan pada $f(n) = 8$, *Non-Playable Character* (NPC) tidak menuju ke posisi tersebut karena dalam perhitungan perbandingan terlalu banyak daripada inisialisasi dan tidak sama dengan inisialisasi. Hasil perhitungan dari tabel yang dapat melewati jalur pendek dimulai dari (0,0), (0,1), (0,2), (0,3), (1,3), sampai (2,3) jalur yang akan dilewati menuju player sesuai gambar 17.



Gambar 17. Hasil pencarian algoritma A-Star

5. Publikasi

Pada tahap terakhir ini (*Recon Duty*), *game* akan di publikasi ke *website* bernama *itch.io*. *itch.io* merupakan *website* koleksi *game* dan *asset* lain, setiap *game* dan *asset* mempunyai dengan versi berbayar dan versi gratis. *Game* yang akan dilakukan *upload* ke *itch.io* bernama *Recon Duty*.

KESIMPULAN DAN SARAN

Proses pembangunan *game* pada penggabungan berbagai *asset* seperti karakter, map tile set, tile objek dan *sound* yang akan dimasukkan ke *scene* dapat menciptakan pengalaman pada *scene*. Pengaturan perspektif kamera dalam pembangunan *game* menggunakan perspektif *Top-Down* pada kamera dengan pandangan pada karakter dan lingkungan sekitarnya. Penerapan sistem nyawa dan mati pada karakter, sistem *healing* pada *health box* dapat menambahkan nyawa *player* bila nyawa *player* hampir habis. Bahasa program GDScript dapat mudah memasuki kode program ke dalam *game* untuk menciptakan perintah-perintah dari program. Bentuk program dapat menciptakan sistem kontrol pada karakter utama, penerapan *Artificial Intelligence* (AI) pada *Non-Player Character* (NPC) dan dapat mengatur perintah sistem dalam *game*. Untuk meminimalisir perubahan besar dalam pembangunan *game* "*Recon Duty*" ini sangat diperlukan perencanaan yang detail serta pemahaman tentang alat yang digunakan untuk meningkatkan efisiensi pembangunan *game*. Peneliti perlu

melakukan pengujian menyeluruh setelah memperbarui versi *engine* untuk menghindari kompatibilitas atau *bug* baru agar dapat menyelesaikan pembangunan *game* dengan hasil yang memuaskan.

DAFTAR PUSTAKA

- [1] Ferdi Dan Arnomo Sasa Ani. (2022). Perancangan *Game Platformer* Pemburu Koin Menggunakan *Godot Engine*. Diss. Prodi Sistem Informasi. Jurnal Comasie Vol.06 No. 04, 109-117.
- [2] Mustofa, Putra Jordy Lasmana Dan Kesuma Chandra. (2021). Penerapan *Game Development Life Cycle* Untuk Video *Game* Dengan *Model Role Playing Game*. Computer Science (Co-Science) 1.1, 27-34.
- [3] Hendaryie, Pragantha Jeanny, Dan Haris Darius Andana. (2020). Pembuatan *Game Android Tower Defense "Myth Of Java"*. Jurnal Ilmu Komputer Dan Sistem Informasi 8.2, 236-240.
- [4] Wijaya Arif Rangga, Kanedi Indra Dan Zulfiandry Ricky. (2023). Pembuatan *Game The Legend Of Bengkulu* Menggunakan *Rpg Maker Vx Ace*. Jurnal Media Infotama 19.1, 69-78.
- [5] Husin Nanang. (2021). Rancang Bangun *Game* Berbasis *Android* Bertemakan Cerita Rakyat Betawi Si Pitung. Jurnal Esensi Infokom: Jurnal Esensi Sistem Informasi Dan Sistem Komputer 5.2, 31-36.
- [6] Ferdi Dan Arnomo Sasa Ani. (2022). Perancangan *Game Platformer* Pemburu Koin Menggunakan *Godot Engine*. Diss. Prodi Sistem Informasi. Jurnal Comasie Vol.06 No. 04, 109-117.
- [7] Arief Assaf, Muhammad Miftah Dan Amin Fakhri. (2023). Pengembangan *Game* Edukasi Dengan Metode Gdlc: Studi Kasus Mata Kuliah Algoritma Dan Struktur Data. Ijis-Indonesian Journal On Information System 8.2, 120-125.
- [8] Marzian Fadel Dan Qamal Mukti. (2017). *Game Rpg "The Royal Sword"* Berbasis *Desktop* Dengan Menggunakan Metode *Finite State Machine* (Fsm). Sisfo: Jurnal Ilmiah Sistem Informasi 1.2, 61-96.
- [9] Londjo Muammar Farhan. (2021). Implementasi *White Box Testing* Dengan Teknik Basis *Path* Pada Pengujian *Form Login*. Jurnal Siliwangi Seri Sains Dan Teknologi 7.2, 35-40.
- [10] Zuhdi Ahmad, Ahmad Imam Dan Putra Ade Dwi. (2023). Implemetation Of A* Algorithm In A Great Elephant Game With Unity 2d. Jurnal Sintech Vol.6.2, 118-123.
- [11] Zumali Cut Dan Wahjono. (2022). *Artificial Intelligence* Dalam Rekrutmen. Jurnal Infokam 18.2, 119-124.
- [12] Billy, Kuswardayan Imam Dan Khotimah Wijayanti Nurul. (2017). Implementasi *Artificial Intelligence* Pada *Game Defender Of Metal City* Dengan Menggunakan *Finite State Machine*. Jurnal Teknik Pomits 6.2, A701-A706.
- [13] Sugianto Bambang Dan Utama Gunawan Pria. (2021). Implementasi Algoritma *Pathfinding* Dan *Decision Tree* Dalam Pembuatan *Video Game* Bergenre *Third Person Shooter*. Skanika 4.2, 83-90.