JURNAL NUANSA INFORMATIKA
Volume 17 Nomor 1, Januari 2023

p-ISSN :1858-3911* , *e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

Implementasi Algoritma *Boids* Untuk Pergerakan *NPC* Pada Game Awas Corona

Rio Andriyat Krisdiawan[1], Muhamad Abdullah Fadli[2], Sherly Gina Supratman[3]
Teknik Informatika Fakultas Ilmu Komputer Universitas Kuningan
Jl. Cut Nyak Dhien No.36 A Kuningan, Telp./Fax. (0232)874824, 873696 e-mail : [1]rioandriyat@uniku.ac.id@uniku.ac.id, [2]mfadli09@gmail.com, [3]sherly.gina.supratman@uniku.ac.id

Abstract

Developments in game technology are increasingly developing, as well as easy access to it, making games playable on various devices and all levels of society. Game technology can be used for various purposes including education, simulation and learning. During the Covid-19 pandemic, people are encouraged to implement health protocols to reduce the spread of the Corona virus. However, based on observational data, there are still many people who have not complied with the health protocol for preventing Covid-19. Researchers conducted this research by utilizing game technology as an educational medium and a simulation of implementing health protocols during a pandemic. Making educational games cannot be separated from the main components in the game, namely NPC (Non Player Characters) which make games interesting and challenging. The Boids algorithm is used to control the movement of groups of NPC. The boids algorithm has three rules namely Alignment, Separation, and Cohesion. The application of the boids algorithm to NPC games is useful for determining NPC movements based on determining the distance of each NPC, speed, and adjusting the direction of motion of NPCs so they don't collide with each other. The game development method used to make corona alert games is GDLC (Game Development Life Cycle). Based on the test results, applying the boids algorithm in group movements looks dynamic and does not crash into each other. The results of testing the user using the User Acceptance Test (UAT), this corona alert game received a fairly good response and was acceptable with an overall percentage value of 85% for elements of education, display, challenge and fun.

Keywords* : *Games Awas Corona, Non Player Characters, Boids Algorithm

Abstrak

Perkembangan teknologi game yang semakin berkembang, serta kemudahan aksesnya membuat game dapat dimainkan di berbagai perangkat dan semua kalangan masyarakat. Teknology game dapat digunakan untuk berbagai keperluan diantaranya sebagai edukasi, simulasi dan pembelajaran. Dimasa pandemi covid-19, masyarakat dihimbau untuk menerapkan protokol kesehatan untuk mengurangi penyebaran virus corona. Akan tetapi berdasarkan data observasi, masih banyak masyarakat yang belum mematuhi protocol kesehatan untuk pencegahan covid-19. Peneliti melakukan penelitian ini dengan memanfaatkan teknologi game sebagai media edukasi penerapan protokol kesehatan dimasa pandemi. Pembuatan game edukasi dan simulasi tidak lepas dari komponen utama dalam game yaitu adalah *NPC (Non Player Characters)* yang membuat game menjadi menarik dan menantang. Algoritma Boids digunakan untuk mengatur pergerakan berkelompok *NPC*. Algoritma boids memiliki tiga aturan yaitu Alignment, Separation, dan Cohesion. Penerapan algoritma boids pada *NPC* game berguna untuk menentukan pergerakan *NPC* berdasarkan penentuan jarak setiap *NPC*, kecepatan, dan menyesuaikan arah gerak *NPC* agar tidak saling bertabrakan. Metode pengembangan game yang digunakan untuk membuat game awas corona adalah *GDLC (Game Development Life Cycle)*. Berdasarkan hasil pengujian, dengan menerapkan algoritma boids dalam pergerakan berkelompok terlihat dinamis dan tidak saling tabrak satu sama lain. Hasil pengujian kepada user menggunakan User Acceptance Test

Terakreditasi* *SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911* , *e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

(UAT), game awas corona ini mendapat respon yang cukup baik dan dapat diterima dengan nilai persentase keseluruhan 85% untuk unsur edukasi, tampilan, tantangan dan kesenangan.
Kata Kunci :Game Awas Corona, Non Player Characters, Algoritma Boids

1. PENDAHULUAN

Perkembangan teknologi game yang semakin berkembang, serta kemudahan aksesnya membuat game dapat dimainkan di berbagai perangkat dan semua kalangan masyarakat. Teknology game dapat digunakan untuk berbagai keperluan diantaranya sebagai edukasi, simulasi dan pembelajaran. Game terdiri dari beberapa genre diantaranya adalah First Person Shooter (FPS), Third Person Shooter (TPS), dll. Pada game FPS terdapat musuh yang harus dilawan oleh player yang bisa disebut sebagai NPC atau Non Player Characters. Karakter non pemain (Non Players Character) adalah setiap karakter dalam game yang tidak dikendalikan oleh pemain, seperti monster, penduduk desa, dan hewan di hutan. [1]

Dimasa pandemi ini masyarakat dihimbau untuk melakukan protokol kesehatan 3M (memakai masker, mencuci tangan dengan sabun, dan menjaga jarak serta menjauh dari kerumunan) yang merupakan cara terbaik yang dapat diterapkan untuk menghentikan rantai penyebaran COVID-19. Upaya ini menuntut kedisiplinan tinggi dan harus diterapkan setiap saat secara konsisten. Hal ini sesuai dengan pernyataan World Health Organization (WHO), bahwa salah satu upaya memelihara diri agar terhindar dari penyakit ini adalah dengan rajin membersihkan tangan menggunakan sabun maupun cairan berbasis alkohol, hindari sering menyentuh bagian wajah (mata, mulut dan hidung) serta melakukan pembatasan kontak fisik dengan orang lain (social distancing).

Bersadarkan hal-hal diatas pada penelitian ini penulis membuat sebuah game bergenre FPS yang tidak hanya menghibur tapi juga untuk sebagai sarana edukasi para player nya itu sendiri, dengan berupa item-item dan perilaku dalam game yang menggambarkan pentingnya protokol kesehatan dengan menggunakan algoritma untuk mengatasi pergerakan NPC.

Penerapan algoritma boids pada NPC game berguna untuk menentukan pergerakan NPC berdasarkan penentuan jarak setiap NPC, kecepatan, dan menyesuaikan arah gerak NPC agar tidak saling bertabrakan. Algoritma ini menggabungkan tiga aturan perilaku sederhana yang menggambarkan bagaimana sebuah manuver boids individu berdasarkan pada posisi dan kecepatan kawanan terdekatnya yaitu cohesion, alignment, dan separation

2. METODE PENELITIAN

2.1 Metode Pengumpulan Data

1. Studi Pustaka

   Mengumpulkan informasi dari sumbersumber seperti buku, jurnal, ebook, dan internet. Studi pustaka dilakukan untuk mendapatkan informasi yang relevan berkaitan dengan penelitian yang lakukan yaitu mengenai algoritma boids dan juga covid-19.

2. Observasi

   Penulis melakukan observasi ke masyarakat sekitar dan melakukan penyebaran questioner tentang pemahaman protocol covid. Serta dengan menganalisis game edukasi dan simulasi mengenai fitur apa saja yang diperlukan dalam pembangunan sebuah *game* edukasi dan mencari tahu kelebihan serta kekurangan yang terdapat antara *game-game* sejenis dengan *game* yang dibangun.

2.1 Metode Penyelesaian Masalah

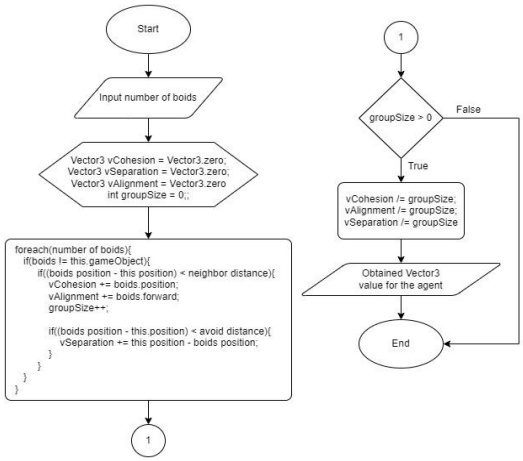
2.1.1 Penerapan Algoritma

Algoritma *Boids* adalah sebuah algoritma yang merepresentasikan gerak dari sebuah kawanan. Perilaku yang dihasilkan sangat

***Terakreditasi* *SINTA 5* - DOI : https://doi.org/10.25134/nuansa**

p-ISSN :1858-3911 , e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

mirip dengan kumpulan ikan atau kawanan burung. Gerak *boids* dihasilkan dari tiga aturan sederhana yaitu *cohesion, alignment, separation*. [2]

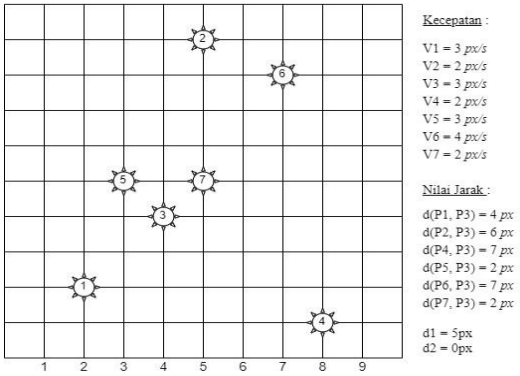


Gambar 1 Flowchart Algoritma Boids

Langkah-langkah algoritma boids :

1. Pertama masukan jumlah *boid*
2. Berikan nilai awal terhadap variabelvariabel boids
3. Dalam proses foreach untuk setiap boid yang berada dalam jangkauan jarak tetangga akan mengakumulasi nilai posisi, tujuan, jumlah boid, dan jarak kedalam variabel boids.
4. Selanjutnya apakah jumlah boid yang berada dalam jangkauan jarak tetangga lebih dari nol, jika ya maka nilai akumulasi posisi, tujuan, dan jarak akan dibagi dengan jumlah boid.
5. Nilai vector untuk pergerakan selanjutnya setiap boid diperoleh.

Contoh perhitungan algoritma boids



Gambar 2 Contoh Posisi NPC virus saat pertama kali muncul

Dalam contoh kasus ini agen 3 akan menyelaraskan dengan tetangganya

Alignment Rule

$$d(Px, Pb) \leq d1 \cap d(Px, Pb) \geq d2 \rightarrow Var = \frac{1}{n}\sum_{x}^{n} Vx$$

Agen yang memenuhi syarat untuk dilakukan perhitungan alignment adalah agen 1, agen 5 dan agen 7, yaitu :

Agen 1 : 4px≤5px ∩4px≥0px
Agen 5 : 2px≤5px ∩2px≥0px
Agen 7 : 2px≤5px ∩2px≥0px

$$Var = \frac{1}{3} V1 + V5 + V7$$

$$Var = \frac{1}{3} 3 + 3 + 2$$

$$Var = \frac{1}{3} 8 = 2.67\ px/s$$

Maka kecepatan agen 3 yang baru setelah dipengaruhi alignment rule adalah 2.67 px/s yang asalnya 3 px/s.

Separation Rule

$$d(Px, Pb) \leq d1 \cap d(Px, Pb) \geq d2 \rightarrow Vsr = \sum_{x}^{n} \frac{Vx + Vb}{d(Px, Pb)}$$

Agen yang memenuhi syarat untuk dilakukan perhitungan separation adalah agen 1, agen 5 dan agen 7, yaitu :

Agen 1 : 4px≤5px ∩4px≥0px
Agen 5 : 2px≤5px ∩2px≥0px
Agen 7 : 2px≤5px ∩2px≥0px

$$Vsr = \sum_{x}^{n} \frac{Vx + Vb}{d(Px, Pb)}$$

$$Vsr = \frac{V5+V3}{d(P5,P3)} + \frac{V1+V3}{d(P1,P3)} + \frac{V7+V3}{d(P7,P3)}$$

$$Vsr = \frac{3+3}{2} + \frac{3+3}{4} + \frac{2+3}{2}$$

$$Vsr = \frac{6}{2} + \frac{6}{4} + \frac{5}{2}$$

$$Vsr = \frac{12}{4} + \frac{6}{4} + \frac{10}{4}$$

$$Vsr = \frac{28}{4} = 7\ px/s$$

Maka kecepatan agen 3 yang baru setelah dipengaruhi separation rule adalah 7 px/s yang asalnya 3 px/s.

Terakreditasi SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911 , e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

Cohesion Rule

$$d(Px, Pb) \leq d1 \cap d(Px, Pb) \geq d2 \rightarrow Pavg = \sum_{x}^{n} \frac{Px}{n}$$

Agen yang memenuhi syarat untuk dilakukan perhitungan cohesion adalah agen 1, agen 5 dan agen 7, yaitu :

Agen 1 : 4px≤5px ∩4px≥0px

Agen 5 : 2px≤5px ∩2px≥0px

Agen 7 : 2px≤5px ∩2px≥0px

$$Pavg = \sum_{x}^{n} \frac{Px}{n}$$

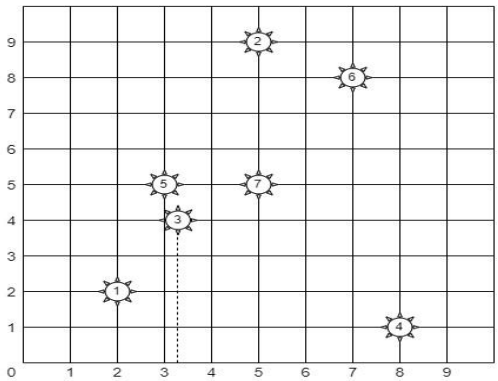
$$Pavg = \frac{P5}{n} + \frac{P1}{n} + \frac{P7}{n}$$

$$Pavg = \frac{(3,5)}{3} + \frac{(2,2)}{3} + \frac{(5,5)}{3}$$

$$Pavg = \frac{(10,12)}{3}$$

$$Pavg = (3.33, 4)$$

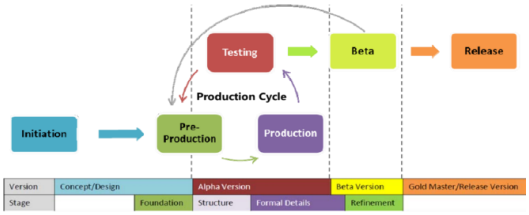
Maka agen 3 akan bergerak menuju posisi ratarata tetangga terdekatnya yaitu ke titik (3.33, 4) yang asalnya berada pada titik (4,4).



Gambar 3 Agen bergerak menuju posisi rata-rata tetangga terdekat

2.1.2 Pengembangan Game

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah GDLC (*Game Development Life Cycle*). [3]



Gambar 4 GDLC(Game Development Life Cycle)

1. Initiation

Tahapan pembuatan konsep kasar dari *game* yang akan dibuat. Pada tahap ini peneliti akan membuat sebuah konsep *game* yang akan dibangun, menganalisa protocol kesehatan yang harus tersampaikan pada game.

2. Pre-production

Pre- production adalah salah satu fase yang penting dalam siklus produksi yang melibatkan penciptaan dan revisi desain *game*. Desain game berfokus pada mendefinisikan *genre* edukasi protocol kesehatan, *gameplay* yang berhubungan dengan covid dan pencegahannya, *game* mekanik/konfensional, alur cerita penanganan covid dan protocol kesehatan, karakter, tantangan, faktor kesenangan, dan aspek teknis.

3. Production

Production adalah proses inti yang berputar di sekitar penciptaan aset, pembuatan kode sumber, dan integrasi kedua elemen. Peneliti menggunakan Bahasa pemrograman c# dengan egine unity, serta pembuatan asset dengan menggunakan blender.

4. Testing

Testing internal yang atau *alpha testing* yang dilakukan adalah dengan menguji coba game dan menguji serta menilai fungsionalitas melalui fitur *playtesting* menggunakan *BlackBox Testing dan Whitebox Testing*.

5. Beta

Pada fase ini dilakukan pengujian pihak ketiga atau *beta testing*. Dengan melakukan pengujian kepada masyarakat sekitar yang menjadi target dalam pembuatan game ini dengan metode pengujian *UAT*.

6. Release

Untuk rilis aplikasi yang dilakukan yaitu dengan sharing aplikasi melalui g.drive dan media social untuk kemudahan akses oleh masyarakat.

3. HASIL DAN PEMBAHASAN

3.1 Storyboard

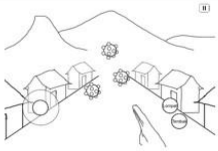
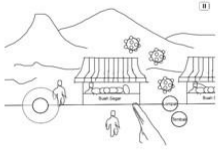
Game yang dibangun memiliki latar belakang cerita mengenai dunia yang

Terakreditasi SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911 , e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

sedang dilanda wabah, wabah tersebut adalah virus bernama Coronavirus disease yang sangat berbahaya dan mudah sekali menular baik itu melalui udara, cairan, dan sentuhan. Untuk mencegah penyebarannya harus dilakukan protokol kesehatan yaitu memakai masker, mencuci tangan, dan juga menjaga jarak.

Disuatu perumahan ada seorang pemuda pemberani yang berusaha menghentikan penyebaran wabah tersebut dengan mengalahkan virus-virus yang masih berkeliaran dengan senjata disinfektannya

No	Gambar	Keterangan
1		Pada *scene* 1 latar tempat yang digunakan adalah di sebuah perumahan, player akan disuguhi tampilan antarmuka kontrol gerak dengan *virtual* analog pada sisi kiri layar, dan tembakan pada sisi kanan layar, terdapat objek rumah dan NPC virus sebagai *enemy* yang harus *player* kalahkan
2		Pada *scene* 2 latar tempat yang digunakan adalah di sebuah pasar, terdapat objek ruko pedagang, NPC virus dan NPC manusia. Jika *player* terlalu dekat dengan NPC manusia maka akan muncul peringatan untuk menjaga jarak.

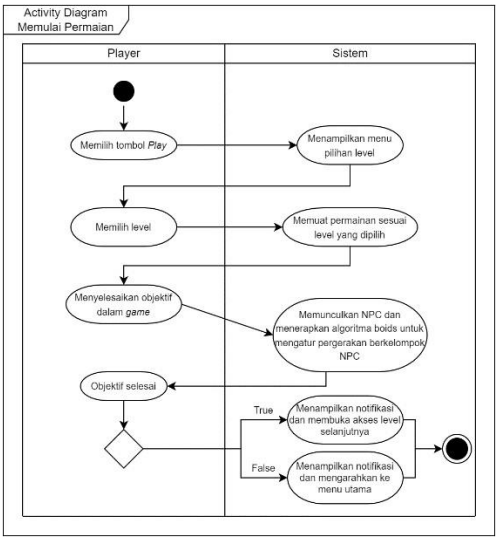
Gambar 5 Stroryboard

3.2 Use Case Diagram

Gambar 6 Use Case Diagram

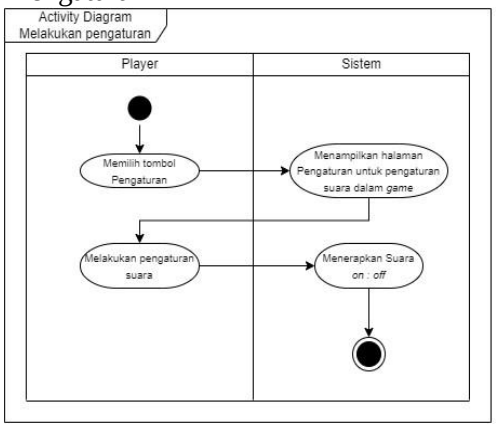
3.3 Activity Diagram

1. Activity diagram Memulai Permainan



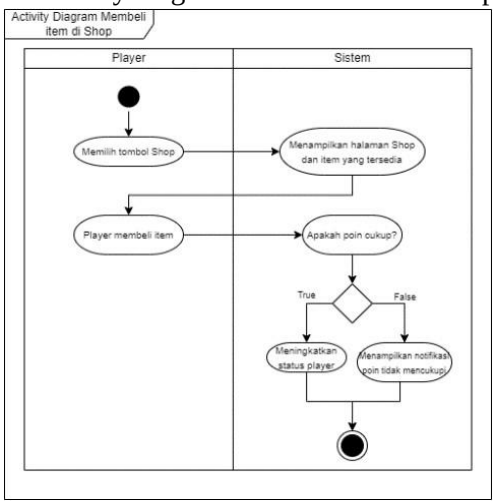
Gambar 7 Activity diagram memulai permaian

2. Activity diagram Melakukan Pengaturan



Gambar 8 Activity diagram melakukan pengaturan

3. Activity diagram Membeli item di shop



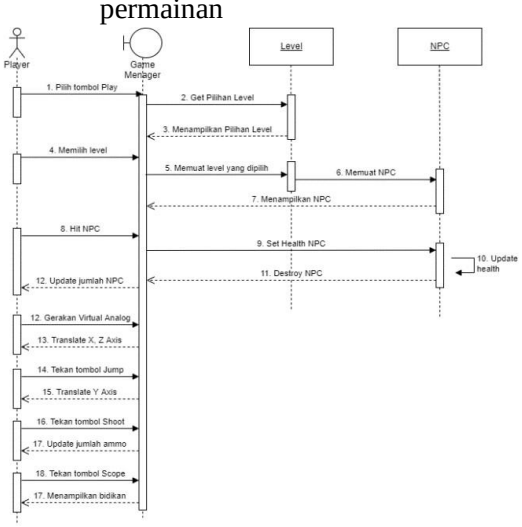
Gambar 9 Activity diagram membeli dishop

3.4 Sequence Diagram

Terakreditasi SINTA 5 - DOI : https://doi.org/10.25134/nuansa

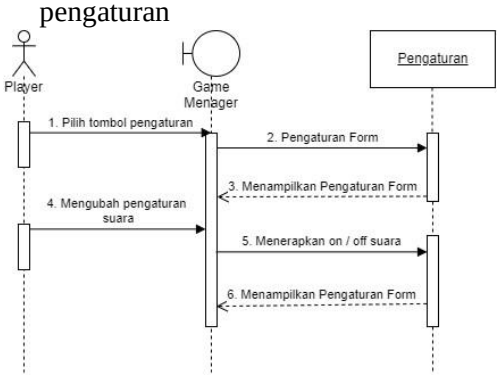
p-ISSN :1858-3911 , e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

1. Sequence diagram Memulai permainan



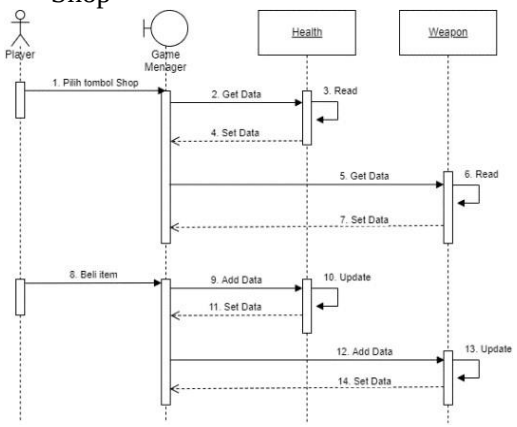
Gambar 10 Sequence diagram memulai permaian

2. Sequence diagram melakukan pengaturan



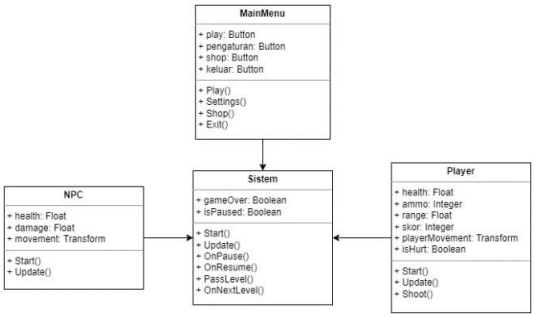
Gambar 11 Sequence diagram melakukan pengaturan

3. Sequence diagram membeli item di Shop



Gambar 12 Sequence beli dishop

3.5 Class Diagram



Gambar 13 Class Diagram

3.6 Interface Game

Antarmuka Menu Utama

Gambar 14 Antarmuka Menu Utama

Antarmuka Menu Utama pada game menampilkan tombol yaitu tombol Play, tombol Pengaturan, tombol Shop, dan tombol Keluar.

Antarmuka Menu *Play*

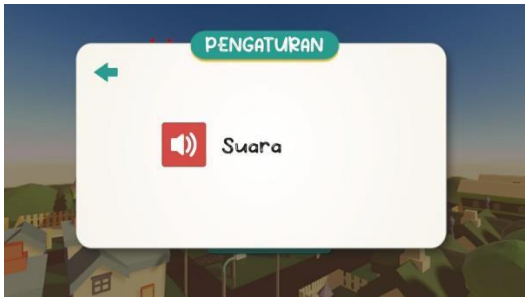
Gambar 15 Antarmuka Menu Play

Antarmuka Menu Play akan muncul dan menampilkan halaman untuk memilih stage dan level yang bisa dimainkan

Antarmuka Menu Pengaturan

Terakreditasi SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911* , *e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom



Gambar 16 Antarmuka Menu Pengaturan

Saat *player* menekan tombol Pengaturan, maka antarmuka Menu Pengaturan akan muncul dan menampilkan halaman untuk mengaktifkan atau menonaktifkan suara.

Antarmuka Menu Shop

Gambar 17 Antarmuka Menu Shop

Menu *Shop* akan menampilkan halaman dimana *player* bisa menggunakan koin yang telah dikumpulkan.

Antarmuka Menu Keluar

Gambar 18 Antarmuka Menu Keluar

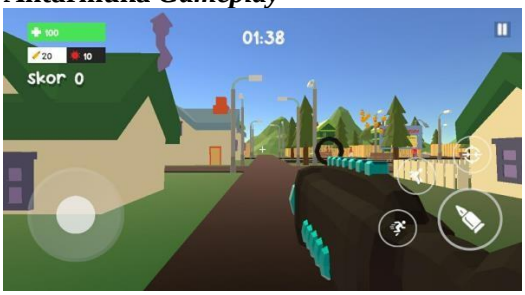
Menampilkan sebuah *dialog box* konfirmasi untuk keluar dari *game* dengan dua buah tombol yaitu tombol ya dan tidak.

Antarmuka Petunjuk Permainan

Gambar 19 Antarmuka Petunjuk Permainan

Sebelum main akan muncul terlebih dahulu petunjuk permainan atau misi yang harus diselesaikan.

Antarmuka *Gameplay*



Gambar 20 Antarmuka Gameplay

Antarmuka *gameplay* terdapat beberapa *virtual controller* yaitu *analog virtual*, tombol *Shoot*, tombol *Jump*, tombol *Scope* atau bidik, dan tombol *Pause.*

Antarmuka Menu *Pause*

Gambar 21 Antarmuka Menu Pause

Antarmuka Menu *Pause* akan muncul dan menampilkan sebuah *dialog box* yang terdiri dari beberapa buah tombol.

3.7 Pengujian Game

Pengujian game dilakukan untuk mengetahui seberapa baik dan sesuai sistem yang

Terakreditasi* *SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911* , *e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

dibangun dengan spesifikasi atau rancangan yang telah dibuat sebelumnya. Pengujian yang dilakukan terdiri dari pengujian alpha testing (*black box, white box)* dan pengujian *beta testing (*UAT(*User Acceptance Test*)).

1. Pengujian Alpha (Blackbox)

Pengujian *black box* memungkinkan pengembang untuk mengetahui apakah fungsi, *input*, dan *output* dari sistem yang dibangun sesuai dengan spesifikasi yang diperlukan.

Tabel 1 Pengujian Blackbox

No	Fungsi yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Hasil yang Didapatkan	Keterangan Hasil
1	Menu *Play*	*Player* menekan tombol *Play* pada halaman menu utama	Menampilkan tampilan pilihan level	Menu *Play* Menampilkan tampilan pilihan level	Valid
2	Menu Pengaturan	*Player* menekan tombol Pengaturan pada halaman menu utama	Menampilkan menu dengan pilihan untuk mengaktifkan dan menonaktifkan suara dan musik	Menu Pengaturan menampilkan *button* untuk mengatur suara dan musik	Valid
3	Menu *Shop*	*Player* menekan tombol *Shop* pada halaman menu utama	Menampilkan menu dengan tombol beli	Menu *Shop* menampilkan menu dengan tombol beli	Valid
4	Menu Keluar	*Player* menekan tombol Keluar pada halaman menu utama	Menampilkan *dialog box* untuk keluar dari *game*	Menu Keluar menampilkan *dialog box* dengan pilihan Ya dan Tidak	Valid
5	Menu *Pause*	*Player* menekan tombol Pause pada *interface game*	Menampilkan panel *Pause* dan menghentikan proses permainan	Menu *Pause* menampilkan panel *pause* dan menghentikan proses permainan	Valid
6	Berjalan Maju	*Player* menggerakan *virtual analog* ke atas	Membuat posisi *player* bergerak ke depan	*Player* bergerak ke depan	Valid
7	Berjalan Mundur	*Player* menggerakan *virtual analog* ke bawah	Membuat posisi *player* bergerak ke belakang	*Player* bergerak ke belakang	Valid
8	Berjalan ke kiri	*Player* menggerakan *virtual analog* ke kiri	Membuat posisi *player* bergerak ke kiri	*Player* bergerak ke arah kiri	Valid
9	Berjalan ke kanan	*Player* menggerakan *virtual analog* ke kanan	Membuat posisi *player* bergerak ke kanan	*Player* bergerak ke arah kanan	Valid
10	*Jump*	*Player* menekan tombol *jump*	Membuat *player* bergerak melompat	*Player* bergerak melompat	Valid
11	*Shoot*	*Player* menekan tombol *shoot*	Membuat *Player* melakukan aksi menembak dan mengurangi health jika terkena NPCVirus	*Player* melakukan *shoot* dan menguarangi health NPCVirus jika terkena	Valid

2. Pengujian Alpha (WhiteBox)

Pengujian *White Box* dilakukan dengan memeriksa logika dari kode program dengan cara melihat dan menganalisa cara kerja internal dan struktur kode sumber dari sebuah program. Berikut ini merupakan hasil pengujian *white box* yang berfokus pada algoritma boids yang diuraikan pada Tabel2.

Tabel 2. Pengujian White Box

1.	using System.Collections; using System.Collections.Generic; using UnityEngine;
2.	public class BoidRules : MonoBehaviour { private Boid boid;

3.	private void Awake()
	{
	boid = GetComponent<Boid>();
	}

4.	public void ApplyRules() { GameObject[] gos;

Terakreditasi* *SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911 , e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

	gos = boid.myManager.allEnemys;
	Vector3 vcentre = Vector3.zero;
	Vector3 vavoid = Vector3.zero;
	Vector3 vheading = Vector3.zero;
	float gSpeed = 0.01f;
	float nDistance;
	int groupSize = 0;

5.	foreach (GameObject go in gos) {
6.	(go != this.gameObject && go != null) {
7.	nDistance = Vector3.Distance(go.transform.position, this.transform.position);
8.	if (nDistance <= boid.myManager.neighbourDistance) {

9.	vcentre += go.transform.position;
	vheading += go.transformm.forward;
	groupSize++;

10.	if (nDistance < boid.myManager.avoidDistance) {
11.	vavoid += this.transform.position - go.transform.position; }

12.	Boid anotherBoid = go.GetComponent<Boid>();
	gSpeed = gSpeed + anotherBoid.speed;
	}
	}
	}

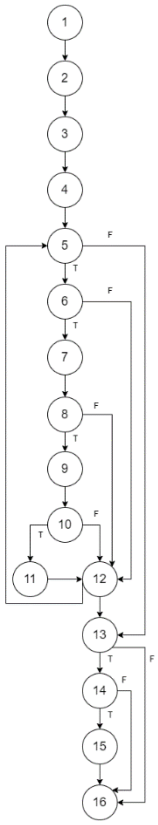
13.	if (groupSize > 0) {

14.	vcentre = (vcentre / groupSize) + (boid.myManager.goalPos - this.transform.position)).normalize;
	vheading = (vheading / groupSize).normalize;
	vavoid = (vavoid / groupSize).normalize;
	boid.speed = gSpeed / groupSize;
	Vector3 direction = (vcentre + vheading + vavoid) + (boid.ObstacleAvoidanceVector() * 100) - transform.position;

15.	if (direction != Vector3.zero) {

16.	transform.rotation = Quaternion.Slerp(transform.rotation, Quaternion.LookRotation(direction), boid.myManager.rotationSpeed * Time.deltaTime);
	}
	}
	}
	}

Berdasarkan table di atas, maka dapat dirumuskan diagram alirnya seperti yang terdapat pada Gambar 22 di bawah ini :



Gambar 22. Flowgraph kode program

Berdasarkan dari gambar di atas, maka dapat dihitung *cyclomatic complexity*, yaitu dengan diketahui :

V(G) = (E - N) + 2

V(G) = (22 - 16) + 2 = 8

Hasil dari perhitungan *cyclomatic complexity* adalah 8 yang menunjukan jumlah *independent path* atau jalur langkah pengujian. Hasil *independent path* tersebut adalah sebagai berikut :

Path 1 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11- 12 - 13 - 14 - 15 - 16

Path 2 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 12 - 13 - 14 - 15 - 16

Path 3 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 12 - 13 - 14 - 15 – 16

Path 4 : 1 - 2 - 3 - 4 - 5 - 6 - 12 - 13 - 14 - 15 - 16

Path 5 : 1 - 2 - 3 - 4 - 5 - 13 - 14 - 15 - 16

Path 6 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11- 12 - 13 - 16

Path 7 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11- 12 - 13 - 14 – 16

Path 8 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11- 12 - 5 - 13 - 14 - 15 – 16

Terakreditasi SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911* , *e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

3. Pengujian Beta (User Acceptance Testing (UAT))

User Acceptance Testing (UAT) dilakukan untuk mengukur bagaimana sistem yang sudah buat telah sesuai dan cukup mampu memenuhi kebutuhan pengguna.

Tabel 3. Bobot Nilai

Kode	Keterangan	Bobot
A	Sangat Setuju	5
B	Setuju	4
C	Netral	3
D	Tidak Setuju	2
E	Sangat Tidak Setuju	1

Tabel 4. Data Jawaban Kuisioner

No	Pertanyaan	A	B	C	D	E
1	Apakah tampilan game awas corona ini menarik?	7	6	2	0	0
2	Apakah menu-menu pada game ini mudah dipahami ?	8	4	3	0	0
3	Apakah tombol-tombol pada game ini mudah digunakan?	11	3	1	0	0
4	Apakah fitur-fitur pada game ini menarik?	6	5	4	0	0
5	Apakah level pada game awas corona ini sudah cukup baik dan seimbang?	3	7	5	0	0
6	Apakah penerapan edukasi protokol kesehatan pada game ini sudah cukup baik?	5	7	3	0	0
7	Apakah informasi pada game ini mudah dipahami?	6	8	1	0	0
8	Apakah game awas corona ini menarik secara keseluruhan?	5	8	2	0	0

Dari data yang didapat pada table diatas kemudian diolah dengan cara mengalikan setiap poin jawaban dengan bobot nilai yang telah ditentukan sebelumnya. Hasil perhitungan bisa dilihat pada Tabel 5 dibawah ini :

Tabel 5.Hasil Perhitungan Kuisioner

No	Poin Jawaban X Bobot Nilai					Jml
	A X 5	B X 4	C X 3	D X 2	E X 1	
1	35	24	6	0	0	65
2	40	16	9	0	0	65
3	55	12	3	0	0	70
4	30	20	12	0	0	62
5	15	28	15	0	0	58
6	25	28	9	0	0	62
7	30	32	3	0	0	65
8	25	32	6	0	0	63
Nilai Total						510

Berdasarkan tabel diatas maka nilai total didapat sebesar 510 dari 15 responden. Selanjutnya adalah mencari nilai maksimal yang dapat diperoleh dari perkalian antara jumlah responden, jumlah pertanyaan, dan bobot nilai maksimal (15 x 8 x 5 = 600).

Berdasarkan hasil pengujian terhadap 15 responden dapat disimpulkan bahwa *game* Awas Corona dapat diterima oleh responden dengan nilai persentase keseluruhan sebesar 85%.

4. KESIMPULAN

1. Algoritma boids dapat diimplementasikan pada game awas corona dalam pergerakan berkelompok Non Player Characters virus.
2. Game dibangun dengan menambakan sisi edukasi pentingnya protokol kesehatan dalam bentuk informasi dan tingkah laku dari Non Player Characters.
3. Berdasarkan hasil pengujian Blackbox game yang dibangun dapat menjalakan spesifikasi yang diperlukan.
4. Berdasarkan pengujian User Acceptance Testing game yang dibangun dapat diterima oleh responden dengan nilai persentase keseluruhan sebesar 85%

5. SARAN

1. Perlu adanya pengembangan game untuk sistem operasi lain seperti iOS.
2. Dapat menambahkan level, stage, dan enemy baru agar game menjadi lebih menarik dan menantang.
3. Menambahkan mode multiplayer agar dapat dimainkan dengan player lainnya.
4. Menambah fitur-fitur baru yang menarik

Terakreditasi SINTA 5 - DOI : https://doi.org/10.25134/nuansa

p-ISSN :1858-3911* , *e-ISSN : 2614-5405
https://journal.uniku.ac.id/index.php/ilkom

References

[1] A. P. E. D. F. N. A. K. Rio Andriyat Krisdiawan, "Implementation Dijkstra's Algorithm for Non-Players Characters in the Game Dark Lumber," in *Journal of Physics: Conference Series*, Kuala Lumpur Malaysia, 2021.

[2] D. R. &. I. M. B. Harefa, "Penerapan Algoritma Boids Pada NPC (NON PLAYER CHARACTER) Dalam Game Menembak Burung.," Program Studi Teknik Informatika dan Sistem Informasi, Fakultas Sains dan Teknologi, Palembang, 2019.

[3] R. A. Krisdiawan, "Implementasi Model Pengembangan Sistem Gdlc Dan Algoritma Linear Congruential Generator Pada Game Puzzle.," *Nuansa Informatika*,, vol. 12, no. 2, p. 12, 2018.

[4]. Hidayah, N. (2015). Pergerakan dan Perilaku Non Player Characters pada Permainan First Person Shooter dengan Algoritma Boids dan Logika Fuzzy. Banjarbaru.

[5]. Jaya, T. S. (2018). Pengujian aplikasi dengan metode blackbox testing boundary value analysis (studi kasus: kantor digital Politeknik Negeri Lampung). Jurnal Informatika: Jurnal Pengembangan IT, 3(1), 45-48.

[6]. Krisdiawan, R. A. (2018). Implementasi Model Pengembangan Sistem GDLC dan Algoritma Linear Congruential Generator pada Game Puzzle. Nuansa Informatika, 12(2).

[7]. Patimah, S., & Sundari, S. (2020). PEMBUATAN GAME REPTIL MENGGUNAKAN ALGORITMA BOIDS BERBASIS ANDROID. In SEMINAR NASIONAL TEKNOLOGI INFORMASI & KOMUNIKASI KE-7 (Vol. 1, No. 1, pp. 304-309)

[8]. Reynolds, C. W. (1999, March). Steering behaviors for autonomous characters. In Game developers conference (Vol. 1999, pp. 763-782).

[9]. Rosa, A. S., & Shalahuddin, M. (2015). Rekayasa Perangkat Lunak Menggunakan UML dan JAVA. Bandung: Informatika Bandung.

[10]. World Health Organization. (2021). "Pertanyaan dan jawaban terkait Coronavirus." https://www.who.int/indonesia/news/novel-coronavirus/qa/qa-for-public. 16 April 2021

Terakreditasi* *SINTA 5 - DOI : https://doi.org/10.25134/nuansa