

**IMPLEMENTASI NPC (*NON PLAYABLE  
CHARACTER*) PADA PERMAINAN 2D “*NINJA  
ADVENTURE*” DENGAN METODE *RAYCAST***

**SKRIPSI**



**LA ODE AHMAD KHAIR ASYIDIQI**

**H071181011**

**PROGRAM STUDI SISTEM INFORMASI  
DEPARTEMEN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS HASANUDDIN**

**MAKASSAR**

**2022**

**IMPLEMENTASI NPC (*NON PLAYABLE  
CHARACTER*) PADA PERMAINAN 2D “*NINJA  
ADVENTURE*” DENGAN METODE *RAYCAST***

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Sains  
pada Program Studi Sistem Informasi Departmen Matematika Fakultas  
Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin

**LA ODE AHMAD KHAIR ASYIDIQI**

**H071181011**

**PROGRAM STUDI SISTEM INFORMASI  
DEPARTEMEN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS HASANUDDIN**

**MAKASSAR**

**2022**

## PERNYATAAN KEASLIAN

Yang bertplayer tangan di bawah ini:

Nama : La Ode Ahmad Khair Asyidiqi  
NIM : H071181011  
Program Studi : Sistem Informasi  
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

**IMPLEMENTASI NPC (NON PLAYABLE CHARACTER) PADA  
PERMAINAN 2D "NINJA ADVENTURE" DENGAN METODE  
RAYCAST**

adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain dan belum pernah dipublikasikan dalam bentuk apapun.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa Sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 30 September 2022  
Yang menyatakan,



**La Ode Ahmad Khair Asyidiqi**  
NIM: H071181011

# IMPLEMENTASI NPC (NON PLAYABLE CHARACTER) PADA PERMAINAN 2D “NINJA ADVENTURE” DENGAN METODE RAYCAST

Disusun dan diajukan oleh

**LA ODE AHMAD KHAIR ASYIDIQI**

**H071181011**

**UNIVERSITAS HASANUDDIN**

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin pada tanggal 30 September 2022 dan dinyatakan telah memenuhi syarat kelulusan.

**Menyetujui,**

**Pembimbing Utama**

**Pembimbing Pertama**

**Dr. Hendra, S.Si., M.Kom.**

**A. Muh. Amil Siddik, S.Si., M.Si.**

NIP: 19760102 2002121001

NIP: 199110032019031015

**Ketua Program Studi**

**Dr. Hendra, S.Si., M.Kom.**

NIP: 19760102 2002121001



Pada tanggal 30 September 2022

## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh:

Nama : La Ode Ahmad Khair Asyidiqi  
NIM : H071181011  
Program Studi : Sistem Informasi  
Judul Skripsi : Implementasi NPC (*Non Playable Character*) Pada  
Permainan 2D "*Ninja Adventure*" Dengan Metode  
*Raycast*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Sains pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

DEWAN PENGUJI

Tanda Tangan

Ketua	: Dr. Hendra, S.Si., M.Kom.	(.....)
Sekretaris	: A. Muh. Amil Siddik, S.Si., M.Si.	(.....)
Anggota	: Rozalina Amran, S.T., M.Eng.	(.....)
Anggota	: Edy Saputra Rusdi, S.Si., M.Si	(.....)

Ditetapkan di: Makassar

Tanggal: 30 September 2022



## KATA PENGANTAR

Segala puji bagi Allah *Subhanahu Wa ta'ala*, Tuhan atas langit dan bumi beserta segala isinya. Karena, berkat nikmat dan karunianya sehingga penulisan skripsi ini dapat terselesaikan. Shalawat serta salam semoga senantiasa tercurahkan kepada Baginda *Rasulullah* Muhammad *Shallallahu Alaihi Wasallam* dan kepada para keluarga serta sahabat beliau, yang senantiasa menjadi teladan yang baik.

Alhamdulillah, skripsi dengan judul “Implementasi NPC (*Non Playable Character*) Pada Permainan 2D Ninja Adventure Dengan Metode Raycast” yang disusun sebagai salah satu syarat akademik untuk meraih gelar Sarjana Komputer pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin ini dapat dirampungkan. Tentunya, dalam penulisan skripsi ini, penulis mampu melewati berbagai hambatan dan masalah berkat bantuan moril dan materil, serta dorongan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih yang tak terhingga kepada orang tua penulis, Ibunda **Dra. Hj. Nursiah** dan Ayahanda **H. La Ode Ruslan B, SE.MM**, sebagai tempat kembali setelah pergi, terima kasih atas kasih sayang, doa, dan nasihat yang tulus sebagai bekal kehidupan. Rasa terima kasih juga penulis tujukan kepada saudara(i) tercinta yang telah menjadi motivator, dan rival dalam membanggakan kedua orang tua, terima kasih atas dukungan yang penulis dapatkan selama ini.

Penghargaan dan ucapan terima kasih dengan penuh ketulusan juga penulis ucapkan kepada:

1. Rektor Universitas Hasanuddin Makassar **Prof. Dr. Ir. Jamaluddin Jompa, M.Sc** dan seluruh Wakil Rektor dalam Lingkungan Universitas Hasanuddin.
2. Bapak Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam **Dr. Eng Amiruddin, M.Si** dan para Wakil Dekan serta seluruh staf yang telah memberikan bantuan selama penulis mengikuti pendidikan di FMIPA Universitas Hasanuddin.

3. Bapak **Prof. Dr. Nurdin, S.Si. M.Si**, sebagai Ketua Departemen Matematika FMIPA Unhas. Penulis juga berterima kasih atas dedikasi dosen-dosen pengajar, serta staf Departemen atas ilmu dan bantuan yang bermanfaat.
4. Bapak **Dr. Muhammad Hasbi, M.Sc** sebagai Ketua Program Studi Sistem Informasi Universitas Hasanuddin.
5. Bapak **Dr. Hendra, S.Si., M.Kom.**, dan Bapak **A. Muh Amil Siddik S. Si., M.Si** sebagai dosen pembimbing utama dan dosen pembimbing pertama atas ilmu yang beliau berikan selama proses perkuliahan, dan kesediaan beliau dalam membimbing, serta memotivasi penulis dalam penyusunan skripsi ini.
6. Ibu **Rozalina Amran, S.T., M.Eng** dan bapak **Edy Saputra Rusdi, S.Si., M.Si** dan sebagai dosen penguji pertama dan penguji kedua atas ilmu yang beliau berikan selama proses perkuliahan, dan saran serta masukan yang telah beliau berikan dalam penyusunan skripsi ini.
7. Teman-teman seperjuangan **Sistem Informasi Unhas 2018** tercinta yang selalu peduli, menemani, menguatkan dan menyemangati selama masa perkuliahan, serta terima kasih terkhusus saudara **Moch. Djihad Akbar Syah** yang telah senantiasa membantu dan memudahkan penulis dalam penyusunan skripsi ini.
8. Kepada **Rigel Rivaldo Subiyakto S.Kom** yang telah meluangkan waktu dalam membagi ilmu. Terimakasih atas segala ilmu yang telah diberikan.
9. Kepada **Dinda Aulia Miftahul Jannah** terimakasih telah saling menguatkan dan meningkatkan motivasi untuk segera menyelesaikan skripsi ini. Semoga kesuksesan selalu kita dapatkan dalam setiap langkah-langkah kita.
10. Semua pihak yang telah memberikan bantuan kepada penulis baik berupa materi dan non materi yang tidak dapat penulis sebutkan satu per satu, terima kasih untuk bantuan dan dukungannya.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan karena keterbatasan penulis. Oleh karena itu, saran dan kritik demi penyempurnaan skripsi ini sangat penulis harapkan. Akhir kata, semoga skripsi ini membawa manfaat dan semoga Allah Subhanahu Wata'ala membalas semua kebaikan semua pihak yang telah membantu.

Makassar, 30 September 2022



La Ode Ahmad Khair Asyidiqi

**PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR  
UNTUK KEPENTINGAN AKADEMIS**

---

Sebagai civitas akademik Universitas Hasanuddin, saya yang bertanda tangan di bawah ini:

Nama : La Ode Ahmad Khair Asyidiqi  
NIM : H071181011  
Program Studi : Sistem Informasi  
Departemen : Matematika  
Fakultas : Matematika dan Ilmu Pengetahuan Alam  
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Hasanuddin **Hak Bebas Royalti Noneklusif** (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul:

**Implementasi NPC (*Non Playable Character*) Pada Permainan 2D  
"Ninja Adventure" Dengan Metode Raycast**

Beserta perangkat yang ada (jika diperlukan). Terkait dengan hal di atas, maka pihak universitas berhak menyimpan, mengalih-media/format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Makassar pada tanggal 30 September 2022

Yang menyatakan



(La Ode Ahmad Khair Asyidiqi)

## ABSTRAK

Permainan merupakan salah satu media yang dapat digunakan untuk menyampaikan tujuan. Tujuan yang terlibat dalam permainan adalah berbagai jenis, seperti pendidikan, hiburan, dan simulasi. Permainan memiliki berbagai jenis *genre* yang bisa dimainkan seperti *action*, *fighting*, *platformer*, dan masih banyak lagi, namun pada penelitian ini akan berfokus pada *genre platformer* serta bagaimana cara mengimplementasikan *raycast* pada musuh dan bagaimana cara merancang dan membangun sebuah permainan pada permainan 2D *ninja adventure*. *Ninja adventure* merupakan permainan bergenre *platformer* dengan konsep mirip seperti permainan zaman dahulu yang sangat banyak orang mainkan yaitu super mario. Pada permainan ini akan suguhkan banyak rintangan, kunci yang harus dicari dan beberapa musuh yang berbeda serta memiliki tingkat kesulitan yang berbeda yang harus ditaklukan. Musuh dalam permainan merupakan NPC atau *Non Playable Character* yang dikendalikan sepenuhnya oleh komputer. Untuk menggerakkan NPC akan dibutuhkan sebuah algoritma atau metode, salah satunya adalah metode *Raycast*. Sebelum permainan diluncurkan harus melalui tahap pengujian *black box*. Sehingga akan dihasilkan sebuah *game 2d platformer*.

Kata Kunci: Permainan, *Platformer 2D*, *Ninja Adventure*, *Black Box*, *Raycast*.

## **ABSTRACT**

The game is one of the media that can be used to convey goals. The objectives involved in games are of various types, such as education, entertainment and simulation. The game has various types of genres that can be played such as action, fighting, platformer, and many more, but this research will focus on the platformer genre and how to implement raycast on enemies and how to design and build a game in a 2D ninja adventure game. Ninja adventure is a platformer genre game with a concept similar to the ancient game that many people played, namely Super Mario. In this game you will be presented with many obstacles, keys to look for and several different enemies and have different levels of difficulty that must be conquered. Enemies in the game are NPCs or Non Playable Characters that are fully controlled by the computer. To move the NPC will need an algorithm or method, one of which is the Raycast method. Before the game is launched, it must go through the black box testing stage. So it will produce a 2d platformer game.

**Keywords:** Game, 2D Platformer, Ninja Adventure, User Interface, Black Box, Raycast.

## DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PERNYATAAN KEASLIAN .....	ii
HALAMAN PERSETUJUAN PEMBIMBING .....	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR .....	v
PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR .....	viii
ABSTRAK .....	ix
ABSTRACT .....	x
DAFTAR ISI .....	xi
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL .....	xv
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Tujuan.....	3
1.4. Batasan Masalah.....	3
1.5. Manfaat Penelitian.....	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Sejarah <i>Game</i> .....	4
2.1.1 Pengertian <i>Game</i> .....	4
2.1.2 <i>Genre Game</i> .....	5
2.2 <i>Ninja Adventure</i> .....	6
2.3 <i>Platform</i> .....	7
2.4 <i>Non-Player Character (NPC)</i> .....	7
2.5 <i>Raycast</i> .....	7
2.6 <i>Unity Engine</i> .....	8
2.7 Metode <i>Waterfall</i> .....	9
2.8 <i>Black Box</i> .....	10
BAB III METODOLOGI PENELITIAN.....	12
3.1. Tahapan penelitian.....	12
3.2. Waktu dan Lokasi Penelitian.....	14

BAB IV HASIL DAN PEMBAHASAN .....	15
4.1. Analisis Kebutuhan Sistem.....	15
4.2. Perancangan Sistem.....	17
4.2.1 Perancangan GDD ( <i>Game Design Document</i> ).....	17
4.2.2 Perancangan Sistem.....	27
4.3. Implementasi .....	30
4.3.1 Implementasi <i>Game</i> .....	30
4.3.2 Implementasi NPC ( <i>Non Playable Character</i> ) .....	42
4.4. Uji Coba.....	45
4.4.1 Uji Coba <i>Gameplay</i> .....	45
4.4.2 Uji Coba <i>Non Playable Character</i> (NPC).....	47
BAB V KESIMPULAN DAN SARAN.....	48
5.1. Kesimpulan.....	48
5.2. Saran.....	48
DAFTAR PUSTAKA .....	50
LAMPIRAN.....	52

## DAFTAR GAMBAR

Gambar 2.1 Metode <i>Waterfall</i> .....	9
Gambar 3.1 Karakter Utama (Ninja).....	12
Gambar 3.2 Gems.....	13
Gambar 3.3 Flowchart alur <i>game</i> .....	13
Gambar 4.1 Alur Permainan.....	15
Gambar 4.2 Alur <i>raycast</i> pada <i>Non Playable Character</i> .....	16
Gambar 4.3 <i>Ninja Adventure</i> .....	18
Gambar 4.4 <i>Movement/Action</i> .....	19
Gambar 4.5 Informasi <i>GUI</i> .....	20
Gambar 4.6 Alur <i>Gameplay</i> .....	20
Gambar 4.7. Alur <i>Raycast</i> .....	21
Gambar 4.8. <i>Yoit/ Ninja</i> .....	22
Gambar 4.9. <i>Darik/iblis</i> .....	23
Gambar 4.10. <i>Darik Terkutuk</i> .....	23
Gambar 4.11. <i>Vampire</i> .....	24
Gambar 4.12. <i>Dirty Bubbles</i> .....	24
Gambar 4.13. <i>Tower Defense</i> .....	25
Gambar 4.14. <i>Duri</i> .....	25
Gambar 4.15. <i>Level I</i> .....	26
Gambar 4.16. <i>Level II</i> .....	26
Gambar 4.17. <i>Main Menu</i> .....	27
Gambar 4.18. <i>End Screen</i> .....	27
Gambar 4.19. <i>Main Menu</i> .....	28
Gambar 4.20. <i>Credit Scene</i> .....	28
Gambar 4.21. <i>Gameplay</i> .....	29
Gambar 4.22. <i>Player Lose</i> .....	29
Gambar 4.23. <i>Main Menu</i> .....	30
Gambar 4.24. Animasi <i>Start</i> .....	31
Gambar 4.25. Animasi <i>About</i> .....	31
Gambar 4.26. Animasi <i>Monsters</i> .....	32
Gambar 4.27. Animasi <i>Button</i> .....	32
Gambar 4.28. <i>About Berupa Credits Scene</i> .....	33
Gambar 4.29. <i>Start Game</i> .....	34
Gambar 4.30. <i>Level 1</i> .....	34
Gambar 4.31. <i>Level 2</i> .....	35
Gambar 4.32. <i>Scene Level</i> .....	35
Gambar 4.33. <i>Tile Palatte</i> .....	36
Gambar 4.34. Rintangan <i>Duri</i> .....	36
Gambar 4.35. <i>Box Collider 2D</i> .....	37
Gambar 4.36. Karakter Utama <i>Game Ninja Adventure</i> .....	38
Gambar 4.37. <i>Player Idle</i> (diam).....	38
Gambar 4.38. <i>Player Run</i> (lari).....	38
Gambar 4.39. <i>Player Jump</i> (lompat).....	39
Gambar 4.40. <i>Player Shoot</i> (menembak).....	39
Gambar 4.41. Sinkronisasi Animasi.....	40
Gambar 4.42. <i>Coin</i> .....	40
Gambar 4.43. Animasi <i>Koin</i> .....	41

Gambar 4. 44. <i>Gems</i> .....	41
Gambar 4.45. Animasi <i>Gems</i> .....	41
Gambar 4.46. <i>Floating button</i> .....	42
Gambar 4.47. Darik/iblis.....	42
Gambar 4.48. <i>Raycast</i> .....	43
Gambar 4.49. <i>Raycast Hit</i> .....	43
Gambar 4.50. <i>Inspector Darik/Iblis</i> .....	44
Gambar 4.51. <i>Raycast</i> tidak terlihat dalam <i>game</i> .....	44

## DAFTAR TABEL

Tabel 4.1. Hasil Pengujian Pada <i>Menu</i> Utama.....	45
Tabel 4.2. Hasil Pengujian Pada <i>Floating button</i> .....	45
Tabel 4.3. Hasil Pengujian Pada <i>controller</i> karakter .....	46
Tabel 4.4. Hasil Pengujian <i>UI</i> pada <i>Game</i> .....	46
Tabel 4.5. Hasil Pengujian Pada <i>Enemy</i> Tabel 4.5 .....	47

# BAB 1

## PENDAHULUAN

### 1.1. Latar Belakang

Perkembangan teknologi di dunia semakin cepat salah satunya adalah permainan video, di Indonesia sendiri baik sudah menjadi hiburan tersendiri bagi seluruh kalangan masyarakat baik yang muda, maupun dewasa. Permainan video bukan lagi menjadi hal yang dipandang negatif oleh banyak orang, kini permainan video atau lebih populer dengan sebutan video *game* menjadi mata pencarian seperti menjadi *game development* atau bahkan atlet profesional yang memainkan *game* itu sendiri pada salah satu bidang permainan.

Menurut (Rahmawati dkk., 2020.), permainan adalah Sebuah permainan interaktif yang membutuhkan alat atau komputer untuk bermain. Tujuan dibuatnya game adalah untuk hiburan, dan game biasanya banyak diminati oleh kalangan anak-anak hingga orang dewasa. Selain itu, ada definisi yang dikemukakan oleh para ahli lain bahwa permainan merupakan salah satu media yang dapat digunakan untuk menyampaikan tujuan. Tujuan yang terlibat dalam permainan adalah berbagai jenis, seperti pendidikan, hiburan, dan simulasi. (Teguh Martono, 2015).

*Ninja Adventure* merupakan permainan bergenre *platformer* dengan konsep mirip dengan permainan zaman dahulu yang sangat banyak orang mainkan yaitu super mario. Pada permainan ini akan suguhkan banyak rintangan, kunci yang harus dicari dan beberapa musuh yang berbeda serta memiliki tingkat kesulitan yang berbeda yang harus ditaklukan. Rintangan yang akan ditampilkan seperti duri dalam jurang, musuh yang akan mengejar serta kunci untuk menyelesaikan permainan. Cara bermainnya dengan *player* dengan *player* mengendalikan karakter atau avatar dengan harus melompati *platform* gantung, menghindari rintangan dan musuh yang harus disingkirkan serta mencari kunci untuk menyelesaikan permainan.

Permainan *platformer* secara umum merupakan sebuah genre permainan video dan subgenre dari permainan aksi. Dalam permainan *platformer 2d* *player* mengendalikan karakter atau avatar untuk melompati *platform* gantungan dan menghindari rintangan. Permainan terkenal yang bergenre *2D platformer* seperti super mario, dan permainan yang nanti akan dibuat yaitu permainan *Ninja Adventure*. *Game platformer 2D* khususnya merupakan genre permainan klasik

yang dinamakan dari *game platformer 2D* khususnya merupakan genre permainan klasik yang dinamakan dari *gameplay*-nya di mana karakter *player* di atas platform untuk melewati rintangan yang ada. Menurut (Ángel Carsí dkk., 2008) *game platform 2D* adalah genre permainan ini ditandai dengan karakter utama yang bergerak dan melompat *platform*, mengumpulkan hadiah dengan berbagai cara dan menghancurkan musuh.

Agar membuat permainan menjadi lebih menarik untuk dimainkan dan membuat permainan lebih menantang dengan musuh yang bisa mendekat serta menyerang *player* di perlukan sebuah algoritma yang berfungsi untuk mendeteksi keberadaan objek di depannya. Salah satunya adalah menggunakan metode *Raycast* yang akan digunakan. *Raycast* adalah *cast* atau melemparkan sinar (*ray*) dari titik asal, ke arah yang ditentukan, dengan panjang tertentu, sampai menabrak sesuatu yang objek yang ada didepannya yang berguna untuk melakukan pengecekan terhadap suatu objek yang ditentukan. Dengan metode ini sinar akan dipancarkan ke semua *colliders* di *scene* dan mengembalikan informasi mendetail tentang apa yang terkena (Unity, 2022<sup>d</sup>). beberapa aplikasi sebelum *launching* akan melalui tahap pengujian, salah satu metode yang biasa digunakan itu adalah *black box*.

Penerapan metode *Raycast* dalam permainan *Ninja Adventure* dibuat berdasarkan prosedur *Raycast* untuk mendeteksi objek apa saja di depannya. Setiap objek yang bergerak dan mengenai sebuah laser tidak terlihat, maka dengan asumsi bahwa musuh akan mengejar objek di depannya. Keuntungannya penggunaan *raycast* adalah mampu mendeteksi objek di depannya dengan cepat dengan jarak yang sudah ditentukan.

## 1.2. Rumusan Masalah

Rumusan masalah pada pembuatan permainan platformer *Ninja Adventure* ini adalah sebagai berikut:

1. Bagaimana cara merancang dan membangun permainan *Ninja Adventure*?
2. Bagaimana mengimplementasikan *Raycast* pada *non playable character* dalam permainan *Ninja Adventure*?
3. Bagaimana hasil pengujian permainan *Ninja Adventure* jika menggunakan *Black Box*?

### **1.3. Tujuan**

Adapun tujuan dari pembuatan permainan *platformer* “*Ninja Adventure*” ini adalah sebagai berikut :

1. Dapat merancang dan membangun permainan 2d *platformer Ninja Adventure*
2. Dapat mengimplementasikan *Raycast* pada *NPC* di game *platformer Ninja Adventure*.
3. Dapat mengetahui hasil pengujian permainan *Ninja Adventure* menggunakan metode *Black Box*.

### **1.4. Batasan Masalah**

Dalam penyusunan penelitian agar sistematis dan mudah dimengerti, maka akan diterapkan beberapa batasan masalah:

1. Pada penelitian ini hanya mengembangkan algoritma pemrograman dengan metode *Raycast* yang diimplementasikan pada karakter musuh, untuk menyerang karakter *player* jika memasuki jarak dari musuh.
2. Pada permainan *Ninja Adventure*
3. Aplikasi yang digunakan untuk membuat *game* adalah *Unity Engine*
4. Permainan ini hanya dimainkan oleh satu *player* (*single player*)
5. Permainan *ninja adventure* hanya berjalan pada desktop.

### **1.5. Manfaat Penelitian**

Adapun manfaat dari penelitian yang dilakukan adalah menambah wawasan dan pengetahuan dalam pengimplementasian metode *raycast* dan bagaimana cara merancang dan membangun permainan *2d platformer* serta dapat berjalan sesuai dengan pengujian *black box*.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sejarah Game**

Dalam buku yang berjudul “Optimalkan Potensi Anak Dengan *Game*” yang ditulis oleh (Tridhonanto, 2011) dijelaskan bahwa *game* sebenarnya sudah ditemukan sejak tiga dasawarsa lalu, kemunculannya pertama kali atas Prakarsa Steven Russel dalam proyek yang bernama *Compert Games* pada tahun 1962 dengan produk andalannya bernama *Star Wars*. Delapan tahun kemudian, sekitar tahun 1970-an muncul *game* yang cukup terkenal dikalangan para pecinta *game* saat itu, nama *game* saat itu adalah *Pong* dengan sistem disket atau *cartridge*. Setelah itu pada tahun 1980-an muncul *game* yang cukup populer dengan basis teknologi IBM OC yakni *game Pacman*. Hingga saat ini *game* tersebut masih populer di kalangan *gamers* dimana *pacman* ini memakan semua *item*, dimana diantara *item* tersebut terdapat *item* penambah tenaga yang dapat digunakan untuk memakan hantu yang selalu menghalanginya.

Pada tahun 1993 tercipta yang juga tidak kalah populer dengan *pacman* yaitu *game* *Mortal Kombat*. *Game* ini mengusung *genre* aksi dimana dua karakter akan saling beradu jurus dan ilmu untuk saling menjatuhkan. Lima tahun kemudian muncul *game* *Doom* dengan desain yang lebih canggih karena teknologinya telah menggunakan basis 3D tetapi sederhana dan sistem suara yang masih terbatas. Dalam hal lain belum lagi ditambah tampilan Gambar *game* dengan tekstur dan model yang bervariasi, lebih berwarna, nyata dan menarik. Jika *game* yang dimiliki dilengkapi dengan fasilitas 3D maka hasilnya akan sangat mirip dengan kondisi nyata. Hal ini karena didukung dengan teknologi CG (*Computer Graphics*). Dilihat dari segi komoditas *game* dari tahun 1990-an sampai tahun 2000-an sangatlah berbeda. Pada tahun 1990-an *game* sudah melampaui berbagai bidang seperti hiburan bagi semua kalangan, bisnis, simulasi, edukasi dan juga pembelajaran virtual. Perputaran uang dalam industri *game* juga sangat besar dan pertumbuhannya sangat pesat.

##### **2.1.1 Pengertian Game**

Menurut (Suryadi, 2018) menjelaskan bahwa kata *game* berasal dari bahasa Inggris. Dalam kamus bahasa Indonesia istilah “*Game*” adalah permainan. *Game*

dalam konteks ini mengacu pada *konsep Intellectual Playability Game*, yang juga dapat diartikan sebagai arena pengambilan keputusan dan aksi *game*, biasanya dalam konteks yang tidak serius atau untuk tujuan penyegaran. *Game* juga dapat disebut sebagai aktivitas permainan yang terjadi dalam situasi berpura-pura tetapi tampak realitas di mana tujuan *player* adalah untuk menang, dan sesuai dengan aturan permainan yang ditetapkan. Menurut (Prasetyo, 2015) menjelaskan bahwa Permainan berasal dari kata bahasa Inggris yang memiliki arti dasar permainan. Permainan dapat diartikan sebagai suatu bentuk hiburan dengan seperangkat aturan yang menentukan apa yang boleh dan tidak boleh dilakukan oleh seorang *player*. Permainan juga dapat diartikan sebagai tempat pengambilan keputusan dan tindakan *player*.

### 2.1.2 Genre Game

Ada banyak *genre game* yang saat ini berkembang sebut saja yang paling populer seperti *action, fighting, adventure* dan masih banyak lagi jenis *game* yang berkembang di era teknologi ini. Dalam buku yang berjudul “Langkah Mudah Membuat *Game 3D*” yang dituliskan oleh (Sibero, 2009) dijelaskan bahwa *genre game* adalah klasifikasi *game* berdasarkan interaksi *playernya*. Menurut (Pratama, 2014) dijelaskan bahwa menurut *genre-nya, game* dibagi menjadi beberapa jenis, seperti :

a. *Action* :

Sebuah permainan yang menuntut *player* untuk memiliki kecepatan refleks, akurasi dan kecepatan waktu untuk menghadapi rintangan.

b. *Fighting* :

*Game fighting* biasanya mempunyai ciri pertarungan satu lawan satu antara dua karakter, yang dimana salah satu dari karakter di kendalikan oleh *computer*.

c. *Shooter* :

*Sub-genre* dari permainan aksi, meskipun banyak *player* yang menganggap bahwa ragam ini merupakan ragam yang berdiri sendiri.

d. *Racing* :

Permainan video yang menuntut keterampilan *player* untuk mengemudi dalam sebuah kompetisi balap-membalap. *Game* ini populer dengan jenis *game* yang berkonsep menggunakan mobil atau motor.

e. *Sport* :

Permainan video yang menuntut keterampilan *player* untuk melakukan pertandingan olahraga secara virtual, seperti pertandingan sepak bola, basket, dan sebagainya.

f. *Adventure* :

*Game adventure* menggabungkan unsur-unsur jenis komponen antara *game action* dan *game adventure*, biasanya menampilkan rintangan yang berjangka panjang yang harus diatasi menggunakan alat atau item sebagai alat bantu dalam mengatasi rintangan, serta rintangan yang lebih kecil yang hampir terus-menerus ada.

g. *Strategi* :

Jenis permainan *game* seperti simulasi dengan tujuan jelas, sehingga membutuhkan strategi *player* dan melibatkan masalah strategi, taktik, dan logika.

h. *RPG (Role Playing Game)* :

Sebuah permainan yang para *playernya* memainkan peran tokoh-tokoh khayalan dan berkolaborasi untuk merajut sebuah cerita bersama.

i. *Platform* :

*Video game* yang mempunyai *gameplay* membimbing karakter untuk melompat antara *platform*, rintangan, atau keduanya untuk melanjutkan permainan.

## 2.2 *Ninja Adventure*

*Ninja Adventure* merupakan permainan bergenre *2d platformer* dengan konsep mirip dengan Super Mario. Pada permainan ini akan disuguhkan banyak rintangan, musuh serta memiliki tingkat kesulitan musuh yang berbeda-beda. *Player* diharuskan untuk menyelesaikan permainan dengan melewati setiap rintangan dan membunuh musuh yang menghalangi serta mengharuskan untuk mengumpulkan *gems*(kunci dalam *game*) untuk dapat membuka pintu untuk memenangkan permainan. Rintangan seperti duri dalam jurang, musuh dengan kemampuan yang beragam akan membuatnya menjadi lebih menarik. Cara bermainnya dengan *player* mengendalikan karakter atau avatar dengan harus melompati *platform* gantung, menghindari rintangan dan musuh yang harus disingkirkan serta

mendapatkan item berupa *gems* yang akan menjadi kunci dalam *game* untuk menyelesaikan *game* tersebut.

### **2.3 Platform**

Permainan *platformer* secara generik adalah sebuah aliran permainan video dan *subgenre* berdasarkan permainan aksi. Dalam permainan *platformer 2d* *player* mengendalikan karakter atau avatar melompati platform gantung dan menghindari rintangan untuk menyelesaikan *game*. Menurut (Reyno, dkk, 2008) *game platform 2D* adalah genre *game* dicirikan oleh protagonis yang bergerak dan melompat ke dalam platform, mengumpulkan hadiah dan menghancurkan musuh dengan berbagai cara. Menurut (Sagala dkk., 2017) dijelaskan bahwa sebuah *game platform* (atau *platformer*) adalah *video game* yang mempunyai *gameplay* membimbing karakter untuk melompat antara *platform*, rintangan, atau keduanya untuk melanjutkan permainan.

### **2.4 Non-Player Character (NPC)**

*Npc* atau *Non playable character* merupakan objek karakter dalam *game* yang dapat berupa orang, hewan, robot, dll yang tidak dapat dikendalikan oleh *player*, tetapi dapat bertindak dan melakukan aktivitas yang tampak dikendalikan oleh *player*, tetapi sebenarnya adalah *npc* yang dikendalikan oleh program komputer. Menurut Warpefelt dalam (Azzam dkk., 2021) *Non-Playable-Character* atau *NPC* adalah objek dalam *game* yang bertindak sebagai karakter yang hanya dapat dikontrol secara terprogram dalam kondisi tertentu. *Npc* biasanya memiliki pola pergerakan seperti maju mundur, mengejar serta menyerang *player* yang ada didepannya.

### **2.5 Raycast**

*Raycast* adalah *cast* atau melemparkan sinar (*ray*) dari titik asal, ke arah yang ditentukan, dengan panjang tertentu, sampai menabrak suatu objek yang ada di depannya yang berguna untuk melakukan pengecekan terhadap suatu objek yang ditentukan. Dengan metode ini sinar akan dipancarkan ke semua *colliders* di *scene* dan mengembalikan informasi mendetail tentang apa yang terkena (Unity, 2022<sup>d</sup>). *Raycast* pada unity adalah fungsi fisika yang memproyeksikan *ray* ke dalam adegan, mengembalikan nilai *boolean* jika target berhasil dipukul. Ketika ini terjadi, informasi tentang pukulan, seperti jarak posisi atau referensi ke *transform*

objek, dapat disimpan dalam variabel *Hit Raycast* untuk lebih lanjut (gamedevbeginner.com).

## 2.6 Unity Engine

Unity adalah sebuah *tools* yang mempermudah pengembangan *video game* 2D atau 3D dengan fitur yang mempermudah dalam pembuatan *game*. Contohnya *Hierarchy* yang mempermudah pengguna dalam mengatur *Game Object* dan *Scene* yang sedang dibuat, serta sistem *Prefab* dan *Scriptable Objects* yang mempermudah dalam mengatur kembali data-data yang sudah ada. Berikut penjelasan dari beberapa komponen pada *Unity* :

### a. Scene

Menurut *Website* resmi *Unity*, *scene* adalah tempat kerja dan tempat mengatur objek-objek yang ada. *Scene* bekerja seperti jendela pada *browser internet* dimana *game developer* dapat berinovasi dan meletakkan berbagai macam objek seperti rintangan, dekorasi di dalamnya (Unity, 2022<sup>a</sup>).

### b. C#

C# adalah bahasa pemrograman yang dibuat oleh Microsoft dan salah satu bahasa pemrograman yang dapat digunakan dalam dunia pengembangan game. Bahasa C# juga telah di *stplayerisasi* secara internasional oleh ECMA. Seperti halnya bahasa pemrograman yang lain, C# bisa digunakan untuk membangun berbagai macam jenis aplikasi, seperti aplikasi berbasis windows (desktop) dan aplikasi berbasis web serta aplikasi berbasis *web services*. C# memiliki fitur-fitur seperti desain berorientasi objek, seperti *garbage collection*, *reflection*, akar kelas (*root class*), dan penyederhanaan terhadap pewarisan jamak (*multiple inheritance*). Fitur-fitur tersebut di dalam C# kini telah diaplikasikan terhadap iterasi, property, kejadian (*event*), metadata, dan konversi antara tipe-tipe sederhana dan objek.

### c. Game Objects

*Game Objects* adalah konsep paling utama dalam pembuatan *game* dengan unity, dimana setiap objek yang ada pada unity adalah *Game Object*. Namun *Game Object* tidak bekerja sendirian. *Game developer* harus memasukkan berbagai macam komponen agar sebuah *Game Object* dapat bekerja dan memiliki eksistensi dalam *scene* yang ada. Contoh beberapa komponen yang 11

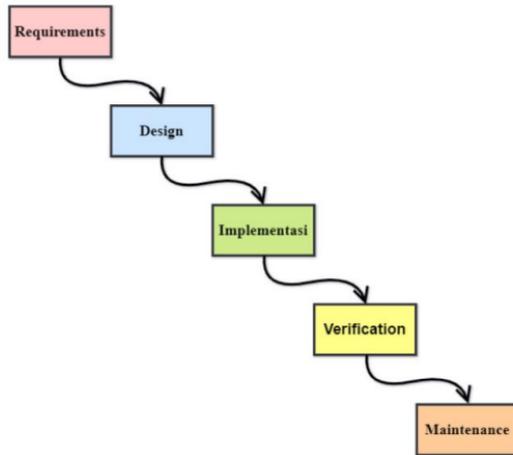
dapat bekerja secara bersamaan di dalam sebuah *Game Object* adalah : *image*, *animator*, *script* (Unity, 2022<sup>b</sup>).

d. *Components*

*Component* adalah inti dari objek dan juga pecahan-pecahan fungsional dari sebuah *Game Object* (Unity, 2022<sup>c</sup>).

## 2.7 Metode *Waterfall*

Metode *waterfall* merupakan suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan diproyeksikan sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Dalam pengembangannya metode *waterfall* memiliki beberapa tahapan yang runtut: *Requirement* (analisis kebutuhan), desain sistem (*system design*), *coding* dan *testing*, penerapan program, pemeliharaan. Tahapan-tahapan tersebut dapat dilihat pada Gambar 2.1.



Gambar 2.1 Metode *Waterfall*.

Berdasarkan tahapan-tahapan yang ada pada metode *waterfall*, berikut adalah penjelasan dari tahapan-tahapan dari metode *waterfall* menurut (Trisianto, 2022) :

- a. *Requirement* (analisis kebutuhan).  
Dalam langkah ini merupakan analisis terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa dilakukan sebuah penelitian, wawancara atau studi literatur.
- b. *Design System* (desain sistem)  
Proses desain akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural.
- c. *Coding dan Testing* (penulisan sinkode program/implementasi)  
Coding merupakan penerjemahan design dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan menerjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan nyata dalam mengerjakan suatu sistem. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang dibuat. Tujuannya adalah untuk menemukan kesalahan kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.
- d. Penerapan program  
Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, *design* dan pengkodean maka sistem yang sudah dijadikan digunakan oleh *user*.
- e. Pemeliharaan (*Operation & Maintenance*)  
Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan(*peripheral* atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.

## **2.8 Black Box**

Menurut (Ningrum, F. C dkk., 2019). *Black Box* adalah metode pengujian perangkat lunak tanpa mengkhawatirkan detail perangkat lunak. Pengujian ini

hanya memeriksa nilai keluaran berdasarkan setiap nilai masukan. Tidak ada upaya yang dilakukan untuk mendeteksi kode program yang digunakan oleh *output*. Uji coba *black box* untuk menemukannya kesalahan fungsionalitas yang hilang atau rusak, seperti kesalahan desain antarmuka atau tampilan dan kesalahan performa.