

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

*Game Shooter Multiplayer* merupakan game tembak – tembak dengan banyak pemain didalamnya yang berkompetisi dalam suatu hal seperti meraih skor tertinggi dengan teknis tertentu, seperti game Counter-Strike dan Valorant [1][2]. Namun, dalam game multiplayer sering kali memiliki masalah masalah seperti lag [3][4][5][6]. Lag adalah suatu kondisi dimana gerakan yang terjadi pada game tidak sesuai dengan eksekusi yang dilakukan akibat kondisi perangkat keras yang kurang mumpuni atau kondisi jaringan yang memiliki latensi yang tinggi [3][4][5][6]. Akibatnya sering kali menimbulkan kerugian hingga kekalahan pada pemain [6][7][8], utamanya untuk game bertema *shooter* yang memerlukan respon cepat. Dalam kasus lag akibat koneksi yang buruk atau latensi yang tinggi, ada banyak cara untuk mengatasinya salah satunya dengan memperbaiki *Quality Of Service* (QoS) [8][10] dari segi arsitektur jaringan, namun hal tersebut memerlukan usaha tambahan dalam mengganti *hardware* dan mengganti protokol. Namun, ada cara yang lebih mudah dan murah yaitu dengan menggunakan bantuan metode kompensasi lag. Kompensasi lag adalah sebuah metode untuk memberikan sebuah kompensasi terhadap pemain game multiplayer yang memiliki lag akibat latensi yang tinggi agar merasakan permainan yang setara [1][2][5][6][7][8][12][19]. Metode kompensasi lag yang sering digunakan dalam *game shooter multiplayer* adalah *client-side prediction* dan *server-side rewind* [1][2][5].

Dalam sebuah konferensi Game Developer tahun 2001, Yahn Benrier [1] menjelaskan bagaimana metode kompensasi lag yang digunakan dalam game buatan perusahaan Valve seperti Counter-Strike. Dijelaskan bahwa metode kompensasi lag dalam game tersebut menggunakan *client-side prediction* untuk sisi klien dan *server-side rewind* untuk sisi server. Selanjutnya, ada sebuah Tesis yang ditulis oleh Jonathan Lundgren [2], ia melakukan penelitian untuk bagaimana cara

implementasi dan evaluasi kinerja registrasi hit atau pengecekan sebuah tembakan apakah benar-benar mengenai lawannya atau tidak untuk game *First Person Shooter* menggunakan metode kompensasi lag dari penelitian sebelumnya dan menggunakan Unreal Engine 4 untuk Game Engine nya. Dari hasil pengujiannya dalam berbagai latensi hasilnya cukup fluktuatif, dan terkadang bisa mendapat keberhasilan registrasi hit di bawah 50%. Alhasil peneliti tersebut mencoba mengatasi hal tersebut dengan menambahkan *fudge factor*. *Fudge factor* dalam hal ini adalah angka konstan yang ditambahkan pada proses registrasi hit untuk latensi berapa pun. Hasilnya terjadi peningkatan keberhasilan registrasi hit pada berbagai latensi. Namun menurutnya peningkatan akurasi tersebut tergantung *frame rate* yang dimilikinya. *Frame rate* adalah jumlah *frame* atau gambar yang dimuat dalam satuan tertentu biasanya detik dan biasa disebut dengan *Frame Per Second* (FPS). Menurutnya dalam memproses sebuah program dalam suatu game biasanya memiliki delay berdasarkan *frame rate*. Hal ini akan membuat keberhasilan registrasi hit menjadi lebih rendah jika player satu dengan yang lainnya memiliki perbedaan *frame rate* yang sangat jauh [10]. Hal ini dikarenakan waktu rewind pada proses registrasi hit menjadi kurang akurat, sehingga sudut atau titik yang dibandingkan dari pose mundur yang diminta dan pose yang diamati klien berbeda akibat perbedaan delay antar *frame* memiliki perbedaan yang cukup tinggi antara dua pemain sehingga membuat kegagalan pada proses registrasi hit.

Oleh karena itu, dalam penelitian ini akan dilakukan penerapan delay antar *frame* saat proses registrasi hit. Hal ini diharapkan dapat membuat pemain dan lawannya seakan-akan memiliki *frame rate* yang sama, dan akhirnya membuat waktu *rewind* lebih akurat saat melakukan registrasi hit karena waktu hit saat didaftarkan dan pose yang akan dibandingkan memiliki waktu yang lebih akurat. Hal ini didukung dengan sebuah artikel yang ditulis oleh Drew Coleman [3] tentang bagaimana penggunaan delay *frame rate* atau dalam Unreal Engine disebut *delta time* terhadap pergerakan karakter. Ia menyebutkan bahwa tanpa memperhitungkan delay *frame rate* akan membuat pergerakan pemain dua kali lebih cepat jika *frame rate* nya dua kali lebih tinggi begitu pun sebaliknya. Oleh karena itu diperlukan perhitungan tambahan menggunakan faktor *frame rate* untuk mengatasinya. Selain

itu, penelitian lain oleh Robert J. Teather [4] menyebutkan bahwa *hit rates* pemain dengan 60 FPS 14% lebih tinggi dari 30 FPS dan latensi dibawah 100ms tidak berpengaruh secara signifikan dibandingkan *frame rate*. Hal ini memperkuat hipotesis bahwa *frame rate* berpengaruh terhadap keberhasilan registrasi hit. Sehingga, dari pemaparan diatas dapat disimpulkan *frame rate* dapat mempengaruhi hasil registrasi hit saat pemain memiliki lag.

Oleh karena itu, Penulis akan mencoba meningkatkan keberhasilan registrasi hit pada pemain yang lag pada teknik *client-Side Prediction* dan *server-side rewind* untuk *game shooter multiplayer* dengan menambahkan *frame rate factor* atau delay antar *frame* ke dalam proses perbandingan hit dan pose yang ada, agar pada saat proses registrasi hit semua pemain seakan memiliki *frame rate* yang sama. Sehingga akhirnya dapat mendapatkan hasil yang lebih maksimal dalam memberikan kompensasi terhadap player yang memiliki lag.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka rumusan masalah dalam penelitian ini adalah Bagaimana pengaruh penerapan *frame rate factor* ke dalam teknik *client-side prediction* dan *server-side rewind* pada *Game shooter multiplayer* terhadap keberhasilan registrasi hit.

## 1.3 Maksud dan Tujuan

Berdasarkan rumusan masalah yang ada, maksud dari penelitian ini adalah menerapkan *frame rate factor* pada proses registrasi hit untuk optimalisasi kompensasi lag dalam *game shooter multiplayer*. Adapun tujuan yang ingin dicapai adalah untuk mengetahui pengaruh penerapan *frame rate* ke dalam teknik *client-side prediction* dan *server-side rewind* pada *game shooter multiplayer* terhadap keberhasilan registrasi hit.

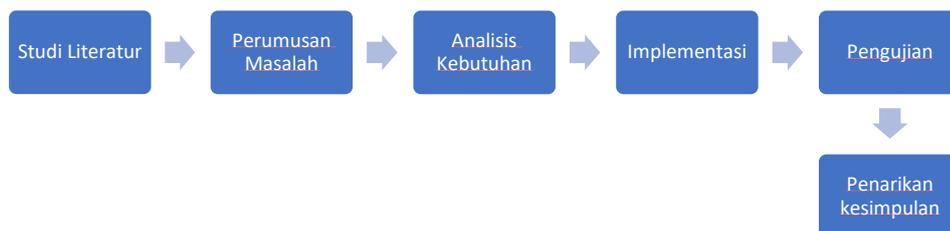
## 1.4 Batasan Masalah

Adapun Batasan-batasan masalah yang ada di dalam penelitian ini meliputi:

1. Sistem dibuat menggunakan bahasa pemrograman C++.
2. Sistem dibuat pada Game Engine Unreal Engine 5.
3. Sistem dibuat hanya untuk game bertema *shooter*.
4. Penelitian ini akan memfokuskan penggunaan teknik *Client-Side Prediction* sebagai metode kompensasi lag dari sisi klien.
5. Penelitian ini akan memfokuskan penggunaan teknik *Server-Side Rewind* sebagai metode kompensasi lag dari sisi server.
6. Algoritma interpolasi yang digunakan dalam penelitian ini hanya interpolasi linier.
7. Jenis peluru yang digunakan dalam game ini hanya berjenis *Hitscan*.
8. Rentang latensi yang diuji hanya dari 0 ms sampai 400 ms.
9. Rentang FPS yang diuji hanya dari 15 FPS sampai 50 FPS.

### 1.5 Metodologi Penelitian

Dalam penelitian ini menggunakan metodologi deskriptif. Penelitian deskriptif merupakan penelitian yang menggambarkan atau mendeskripsikan cara mengumpulkan data, analisis data, membuat pemecahan masalah dengan melakukan implementasi dan pengujian, dan menyusunnya untuk didapat kesimpulan mengenai masalah pada penelitian yang telah dilakukan. Berikut adalah tahap – tahap yang dilakukan pada penelitian ini:



Gambar 1. 1 Metodologi Penelitian

Berikut penjelasan mengenai setiap langkah pada metode penelitian yang telah dipaparkan sesuai dengan gambar 1.1 yaitu:

1. Studi Literatur

Studi literatur adalah teknik pengumpulan data dengan melakukan pengumpulan beberapa literatur, jurnal, buku, *paper*, konferensi, untuk mendapatkan pengetahuan terkait kompensasi lag dengan *client-side prediction* dan *server-side rewind* pada *game shooter multiplayer*.

## 2. Perumusan Masalah

Pada tahap ini dilakukan untuk merumuskan masalah terkait kompensasi lag dengan *Client-Side Prediction* dan *Server-Side Rewind* pada *game shooter multiplayer*. Berdasarkan hasil studi literatur yang telah dilakukan berikut rumusan masalah yang telah didapat dari latar belakang diantaranya: Bagaimana pengaruh penerapan *frame rate factor* ke dalam teknik *client-side prediction* dan *server-side rewind* pada *Game shooter multiplayer* terhadap keberhasilan registrasi hit.

## 3. Analisis Kebutuhan

Pada tahap ini dilakukan analisis kebutuhan untuk mengetahui apa saja yang diperlukan dan dilakukan terkait rumusan masalah yang dijelaskan sebelumnya mulai dari bagaimana pengaruh *frame rate* client terhadap registrasi hit pada keberhasilan teknik yang digunakan sebelumnya hingga bagaimana implementasi masukan, cara kerja (proses), dan keluaran yang dihasilkan dari teknik *client-side prediction* dan *server-side rewind* serta bagaimana implementasi dari *frame rate factor* terhadap implementasi program. Selain itu disusun juga bagaimana bentuk pengujian untuk menentukan hasil akhirnya.

## 4. Implementasi

Pada tahapan ini, hasil dari analisis kebutuhan kemudian diimplementasikan ke dalam program game.

## 5. Pengujian

Pada tahap ini, dilakukan pengujian sistem yang sudah dibangun apakah dari rumusan masalah tadi terselesaikan atau tidak berdasarkan analisis kebutuhan dan implementasi sebelumnya. Pengujian dilakukan dengan uji performansi. Dalam penelitian ini uji performansi dilakukan berdasarkan keakuratan registrasi hit berdasarkan uji *raycasting*.

## 6. Penarikan Kesimpulan.

Dan pada tahapan terakhir akan ditarik kesimpulan dari tujuan penelitian yang telah ditentukan sebelumnya.

### 1.6 Sistematika Penulisan

Sistematika penulisan bertujuan untuk memberikan gambaran secara umum mengenai isi laporan skripsi ini, maka sistematika penulisan dari laporan skripsi ini disusun sebagai berikut:

#### **BAB 1 PENDAHULUAN**

Pada bab ini menjelaskan mengenai latar belakang masalah pada *game shooter multiplayer*, melakukan perumusan masalah berdasar masalah–masalah pada latar, kemudian menentukan maksud dan tujuan yang dituju, membatasi masalah agar pembahasan terarah, menguraikan metodologi penelitian yang digunakan dan menguraikan susunan sistematika penulisan.

#### **BAB 2 TINJAUAN PUSTAKA**

Pada bab ini membahas mengenai teori – teori pendukung yang menjadi landasan penelitian ini. Adapun teori terkait game, *game shooter multiplayer*, server, klien, latensi, lag, C++, Unreal Engine, *server-side rewind*, *client-side prediction* dan teori-teori pendukung lainnya yang berkaitan dengan topik pembangunan perangkat lunak.

#### **BAB 3 PEMBAHASAN**

Pada bab ini berisi analisis sistem yang dibangun, dengan melakukan analisis masalah, analisis proses, menganalisis kebutuhan sistem pada aplikasi yang dibangun.

#### **BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab ini membahas dan menjelaskan implementasi dalam bahasa pemrograman yaitu implementasi kebutuhan perangkat keras dan perangkat lunak, dan tahapan dalam melakukan pengujian perangkat lunak.

## **BAB 5 KESIMPULAN DAN SARAN**

Bab ini berisi tentang kesimpulan yang sudah diperoleh dari hasil dari pengaruh penerapan *frame rate* pada terhadap peningkatan keberhasilan registrasi hit untuk optimalisasi kompensasi lag dalam *game shooter multiplayer* untuk masa yang akan datang.