

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Sebuah *game* atau permainan adalah suatu sistem dimana pemain berinteraksi dalam sebuah konflik buatan yang didefinisikan oleh aturan-aturan yang memiliki hasil yang terkuantifikasi [1]. Kamus Cambridge mendefinisikan *Computer Game* sebagai “Sebuah permainan yang dimainkan menggunakan komputer, dimana gambar-gambar yang muncul di layar dikontrol dengan menekan tombol-tombol atau menggerakkan *joystick* [2].”

Game development adalah *software development* yang membuat suatu *software* yang mempunyai seni, audio, cerita atau simulasi, dan *gameplay* yang interaktif [3]. Dalam membangun sebuah *game*, *game developer* akan membuat sebuah terrain atau level atau yang biasa disebut area permainan untuk menampilkan sebuah cerita, situasi, kondisi, dan suasana yang ada dalam permainan tersebut agar dapat menarik pemain untuk merasakan situasi yang ada dalam *game* yang telah dibuat.

Tujuan utamanya adalah untuk membuat level atau area permainan yang realistis secara visual, dimana memiliki kriteria yang realistis seperti: tinggi, lekukan pada lereng dan retakan pada area permainan [4]. Melakukan hal – hal ini secara manual sangatlah lama, mahal, dan membuat file memiliki ukuran yang besar.

Semua masalah ini dapat diatasi dengan menggunakan *Procedural Content Generation*. Pendekatan ini memungkinkan pembuatan konten secara otomatis dengan mengikuti aturan dan algoritma tertentu, sehingga mengurangi waktu, biaya, dan ukuran *file* dalam pembuatan konten. Salah satu jenis konten yang dapat dibangkitkan dengan menggunakan *Procedural Content Generation* adalah area permainan atau level, dimana proses ini dinamakan *Procedural Level Generation*. Dalam penggunaannya, *Procedural Level Generation* memerlukan sumber keacakan agar hasil pembangkitan terasa lebih natural.

Noise dapat digunakan sebagai salah satu sumber keacakan dalam penggunaan *Procedural Level Generation*. *Noise* merupakan fungsi primitif tak teratur yang dapat digunakan untuk menghasilkan tekstur prosedural acak, dimana fungsi ini berguna untuk menghilangkan kesamaan monoton dari pola-pola yang mungkin terlalu reguler. Keacakan yang dimaksud bukanlah *true randomness*, tetapi merupakan keacakan dalam bentuk *pseudorandom* [5].

Salah satu contoh dari *noise* adalah algoritma *Simplex Noise*, yang dikembangkan oleh Ken Perlin pada tahun 2001 sebagai pengembangan dari algoritma *Perlin Noise*. Algoritma *Simplex Noise* dirancang untuk mengatasi kekurangan dari *Perlin Noise*, terutama pada dimensi tinggi. Dibandingkan dengan pendahulunya, *Simplex Noise* memiliki kompleksitas komputasi yang lebih rendah, dan dapat diskalakan ke dimensi tinggi dengan beban komputasi yang lebih sedikit. *Simplex Noise* juga tidak memiliki artefak direksional, sehingga cocok digunakan sebagai pembangkit *Procedural Level Generation* [6].

1.2 RUMUSAN MASALAH

Bagaimana menggunakan *Procedural Level Generation* dengan algoritma *Simplex Noise* pada game eksplorasi pada platform *Personal Computer* (PC).

1.3 BATASAN MASALAH

Adapun batasan masalah dalam penelitian ini yaitu:

1. Menggunakan algoritma *Simplex Noise* dalam *Procedural Level Generation*
2. Diimplementasikan pada pembangkit level game eksplorasi
3. Sistem dikembangkan dengan menggunakan *Game Engine* Unity dan bahasa pemrograman C#

1.4 TUJUAN DAN MANFAAT PENELITIAN

Penelitian ini bertujuan untuk mengimplementasikan *Procedural Level Generation* dengan algoritma *Simplex Noise* pada pembangkit level game eksplorasi. Penelitian ini bermanfaat sebagai *showcase* dari *Procedural Level Generation* sebagai alternatif dalam menyusun sebuah level pada suatu *game*.

1.5 SISTEMATIKA PENULISAN

Laporan tugas akhir ini terdiri dari 6 bab, antara lain:

1. BAB I PENDAHULUAN

Bab ini berisi penjelasan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, serta sistematika penulisan.

2. BAB II LANDASAN TEORI

Bab ini berisi penjelasan mengenai definisi *Simplex Noise*, *Procedural Level Generation*, dan *Game*, serta penjelasan tentang teori pendukung topik tersebut.

3. BAB III METODOLOGI PENELITIAN

Bab ini berisi penjelasan tentang tahapan proses, metode, serta alat bantu yang akan digunakan selama penelitian dan perancangan perangkat lunak.

4. BAB IV ANALISIS DAN PERANCANGAN

Bab ini berisi analisis tentang *game* yang akan dikembangkan, termasuk di dalamnya konsep *game*, analisis masukan dan keluaran, analisis kebutuhan data, rancangan tampilan, rancangan *input*, rancangan struktur data, rancangan struktur program, dan rancangan algoritma program.

5. BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisi hasil implementasi pengembangan *software*, uji coba sistem, serta analisis hasil perangkat lunak.

6. BAB VI PENUTUP

Bab ini berisi penjelasan mengenai kesimpulan penelitian, serta saran-saran yang berhubungan dengan hasil penelitian.