

BAB 2 LANDASAN TEORI

2.1 Game Elements

Sebuah *game* terbentuk dari 2 jenis elemen yang menjadi dasar dalam pembuatan suatu konsep *game*, elemen tersebut adalah *formal elements* dan *dramatic elements*. *Formal elements* merupakan elemen-elemen yang membantu menciptakan struktur suatu permainan. Tanpa adanya *formal elements* ini, maka sebuah *game* tidak bisa disebut sebagai *game* yang lengkap [14]. Berikut ini merupakan 8 *formal elements* yang dapat dijelaskan sebagai berikut.

1. *Players* : *Players* merupakan pemain yang memainkan dan berinteraksi dengan suatu *game*. Elemen ini berkaitan dengan jumlah pemain yang terlibat, jenis *game* yang dimainkan, peran yang didapatkan oleh para pemain, dan tipe interaksi pemain.
2. *Objectives* : *Objectives* merupakan tujuan atau suatu hal yang ingin dicapai oleh pemain dengan menyelesaikan suatu permasalahan yang ada dengan aturan-aturan yang sudah ditetapkan pada suatu permainan sehingga dapat memotivasi para pemain untuk terlibat langsung dalam permainan. Contoh dari *objectives* adalah mendapatkan XP paling banyak di akhir permainan, mendapatkan semua *item* yang diinginkan, tetap hidup hingga akhir *level* dan berhasil mengalahkan semua musuh.
3. *Procedures* : *Procedures* merupakan aksi-aksi atau suatu tindakan yang diperbolehkan dan dapat dilakukan oleh pemain sesuai dengan *rules* yang telah ditetapkan pada suatu *game* tersebut. *Procedures* juga dapat merujuk ke suatu kumpulan kontrol tertentu bagi para pemain. *Procedures* dapat menentukan tindakan yang tidak mungkin dilakukan pemain di dalam *game*.
4. *Rules* : *Rules* merupakan suatu kumpulan aturan yang sudah ditetapkan untuk membatasi atau menentukan semua tindakan yang dapat dilakukan dan tidak dapat dilakukan oleh para pemain di dalam *game* tersebut.
5. *Resource* : *Resource* merupakan kumpulan *game object* atau *asset-asset* yang memiliki arti dan nilai bagi pemain yang dapat digunakan untuk mewujudkan tujuannya dalam *game*. Nilai dari *resource* ini ditentukan dari kelangkaan dan

kegunaannya. Contoh dari *resource* adalah *lives, health, currency, inventory, time* dan *actions*.

6. *Conflict* : *Conflict* merupakan hasil dari adanya *procedures* dan *rules* dalam permainan yang menghalangi pemain untuk mencapai *objective game*. *Objective* sering memandu pemain ke situasi *conflict* ini. Terdapat 3 jenis *conflict* dalam suatu *game* yaitu, hambatan, lawan dan dilema.
7. *Boundaries* : *Boundaries* merupakan batasan dunia pada permainan, bisa berhubungan atau berkaitan dengan *game space* yang merupakan lingkungan tempat permainan yang ada di dalam *game*.
8. *Outcome* : *Outcome* merupakan hasil akhir dari suatu permainan yang telah selesai. Suatu hasil yang terukur dan tidak pasti harus dimiliki oleh *outcome* agar dapat mendorong minat pemain untuk memainkan *game* tersebut.

Dramatic elements dapat didefinisikan sebagai elemen-elemen yang melibatkan para pemain untuk merasakan pengalaman bermain *game* secara emosional. *Dramatic elements* sendiri dapat diintegrasikan dengan *formal elements* untuk membuat pengalaman bermain *game* menjadi lebih bermakna bagi pemain. *Dramatic elements* memiliki elemen dasar seperti *challenge* dan *play* yang bisa ditemukan di semua *game* dan elemen yang lebih rumit seperti *premis, character* dan *story*. Berbeda dengan *formal elements*, tidak semua *dramatic elements* harus dimasukkan ke dalam *game* [15]. Berikut ini merupakan 5 *dramatic elements* yang dapat dijelaskan sebagai berikut [16].

1. *Story* : *Story* merupakan *narrative* atau suatu cerita yang ada dalam suatu permainan. *Story* pada umumnya akan memulai cerita dari awal permainan dan akan berjalan sampai proses dari *story* tersebut selesai. Pada *story* juga akan diceritakan peran yang dimiliki oleh pemain untuk meningkatkan pengalaman dalam memainkan *game*. Dengan adanya *story* dalam *game* maka hal itu akan membantu pemain dalam memahami isi cerita yang diceritakan oleh *game*.
2. *Challenge* : *Challenge* berkaitan dengan keahlian atau kemampuan yang dimiliki oleh pemain di dalam *game*. Rasa kesenangan dan kepuasan dalam bermain *game* akan muncul ketika suatu tantangan telah berhasil diselesaikan. *Challenge* sendiri dapat berupa suatu permasalahan ataupun tugas-tugas yang harus di hadapi oleh pemain di dalam *game* untuk mencapai tujuannya.

3. *Play* : *Play* atau bermain dapat dikatakan suatu aktivitas yang bisa dilakukan oleh para pemain untuk memperoleh keterampilan dan rasa kesenangan. Pemain bisa dengan bebas menentukan tindakan yang diinginkan saat bermain selama pemain masih mematuhi peraturan atau *rules* yang ada di dalam *game*.
4. *Premise*: *Premise* dapat dikatakan sebagai suatu elemen drama tradisional. *Premise* ini merupakan dasar atau landasan dari cerita di dalam *game*. *Premise* ini yang menetapkan apa yang nantinya akan terjadi dalam *game* sehingga dapat membuat pemain termotivasi untuk bermain *game*.
5. *Characters*: *Characters* merupakan agen yang menjalankan cerita dalam suatu permainan. Pemain bisa terhubung ke dalam suatu cerita dan kejadian melalui karakter itu sendiri yang dimainkan ketika berada di dalam *game*. *Character* di dalam *game* perlu menyeimbangkan konsep agensi (memungkinkan pemain memiliki kebebasan mengontrol dalam permainan dan mewakili pemain) dan empati (berkaitan dengan perasaan pemain dan keterikatan emosional dengan karakter)

2.2 Role Playing Game

Dalam permainan RPG biasanya pemain berperan sebagai agen dengan tujuan dan karakteristik mereka untuk mengontrol sebagian atau semua tindakan di dunia *game*. Agen dunia *game* yang diaktifkan dan dikendalikan oleh pemain disebut PC (*Player Characters*), misalnya seorang pemain mungkin mengendalikan salah satu ksatria untuk mengalahkan *monster* [5].

2.3 Procedural Content Generation

Procedural Content Generation (PCG) adalah proses untuk menghasilkan konten untuk *video game* secara otomatis menggunakan algoritma. Berbagai macam konten dapat dihasilkan seperti *art assets* untuk *map*, *texture* bahkan struktur permainan seperti alur cerita dan karakter. PCG tidak hanya memiliki potensi untuk membangun konten secara dinamis oleh pengembang, namun dapat meningkatkan *replayability* pada permainan sehingga pemain akan disajikan dengan pengalaman baru setiap kali mereka memulai permainan [17].

Menurut Jonathan Roberts dan Ke Chen untuk menghasilkan konten *game* itu mahal karena membutuhkan banyak keahlian yang berbeda dan waktu yang

lama dalam pengembangan konten *game* misalnya, menghasilkan animasi untuk satu karakter bisa memakan waktu beberapa bulan, oleh karena itu yang paling penting adalah menciptakan konten secara tepat dengan waktu seefisien mungkin [17].

Selain itu beberapa studio *game* mulai menggunakan PCG dibandingkan mempekerjakan *artist* dengan tujuan agar dapat memproduksi *game* lebih cepat dan lebih murah dengan tetap menjaga kualitas. Sementara itu bagi banyak studio *game indie* yang menganggap PCG adalah satu-satunya cara untuk menghasilkan konten *game* dengan jumlah yang memadai dan mendukung nilai *replayability* [18]. Bukan hanya itu saja namun dengan menggunakan PCG dapat membantu untuk menghemat biaya produksi dan ruang penyimpanan disk [19].

Secara umum PCG dalam dunia *video game* merupakan bidang yang relatif baru karena sebagian besar PCG banyak digunakan pada bidang pendidikan formal untuk ilmu komputer dan untuk bidang penelitian [20]. PCG memiliki hierarki berupa tabel berbentuk piramida yang menunjukkan dimana jika semakin posisi hierarkinya semakin kebawah maka semakin banyak algoritma yang sudah siap untuk digunakan, sementara sebaliknya semakin keatas hierarkinya maka semakin sedikit algoritma yang bisa digunakan karena sistemnya masih berupa eksperimental dan juga beberapa algoritma masih belum siap serta sulit untuk dilakukan implementasi [20]. Menurut Nicholas Barriga PCG dapat diklasifikasikan menjadi 6 tipe meliputi.

1. *Game Bits* adalah unity dasar dari suatu konten *game*. *Game bits* biasanya tidak akan melibatkan pengguna secara langsung, jika *game bits* tersebut dapat berdiri sendiri. Contoh dari *game bits* biasanya mencakup seperti *texture*, arsitektur, vegetasi, suara, dan efek grafis seperti api, air, dll.
2. *Game Space* adalah lingkungan tempat permainan berlangsung. Contoh dari *game space* yaitu *map* atau *terrain*.
3. *Game System* adalah sistem yang mensimulasikan lingkungan yang lebih kompleks. Contoh dari *game system* seperti ekosistem, jalan, dan lingkungan perkotaan.
4. *Game Scenario* mengatur level sebelumnya agar menjadi sesuai dengan rencana atau sesuai dengan urutan kejadian seperti *stories*, *puzzle* dan *level*.
5. *Game Design* adalah suatu peraturan dan mekanisme dari suatu permainan.

6. *Derived Content* dapat dibuat sebagai pendamping dunia *game*, seperti papan peringkat dan *item* yang mencakup aksi pemain di dunia *game*.

2.6 / 4.6 Derived Content	6.1 News and Broadcasts	6.2 Leaderboards		
2.5 / 4.5 Game Design	5.1 System Design	5.2 World Design		
2.4 / 4.4 Game Scenarios	4.1 Puzzles	4.2 Storyboards	4.3 Story	4.4 Levels
2.3 / 4.3 Game Systems	3.1 Ecosystems	3.2 Road Networks	3.3 Urban Environments	3.4 Entity Behavior
2.2 / 4.2 Game Space	2.1 Indoor Maps	2.2 Outdoor Maps	2.3 Bodies of Water	
2.1 / 4.1 Game Bits	1.1 Textures	1.2 Sound	1.3 Vegetation	1.4 Buildings
	1.5 Behavior	1.6 Fire, Water, Stone, & Clouds		

Gambar 2.1. Klasifikasi jenis konten game pada PCG

Sumber: [21]

Selain dari 6 tipe klasifikasi yang sudah dijelaskan diatas, menurut Nicholas Barriga PCG memiliki 3 kategori metode yaitu metode tradisional, metode *search-based* dan metode *machine learning* yang dapat dijelaskan sebagai berikut.

1. Metode tradisional bersifat *pseudo-random* dan sudah pernah diterapkan dalam *video game* komersial dan telah banyak digunakan dalam pembuatan *dungeon* dan *labyrinth generation*. Keuntungan utama dari metode tradisional ini adalah kesederhaan dan kecepatan dalam proses implementasi untuk menghasilkan suatu konten. Bukan hanya itu saja, tetapi penggunaan *fractals* dan *noise* juga termasuk pada metode tradisional ini yang dapat digunakan untuk menghasilkan konten alam pemandangan seperti pengunungan, bebatuan ataupun vegetasi [20].
2. Metode *search-based* dilakukan dengan *evolutionary algorithm* yang merupakan suatu prosedur pencarian atau pengoptimalan dalam memproduksi suatu konten hingga suatu algoritma terpisah menilai bahwa itu memenuhi beberapa kriteria yang telah ditentukan. Metode ini biasanya didefinisikan oleh 3 komponen, yaitu *the space representation*, *the evaluation function*, dan *the search algorithm* [20].
3. Metode *machine learning* digunakan untuk memprediksikan atau mengklasifikasikan masalah *generated content game* dengan dilakukannya pembelajaran menggunakan suatu kumpulan data atau dataset [20].

2.4 Perlin Noise

Perlin noise merupakan algoritma untuk menghasilkan *noise* dengan dilakukan pengambilan sampel dan menginterpolasi titik dalam *grid* vektor secara acak. Mengubah skala dan menambahkan beberapa tingkat *noise* setiap titik pada *heightmap* yang menghasilkan struktur alami contohnya seperti pembuatan pegunungan. *Perlin noise* dikembangkan oleh Ken Perlin pada tahun 1985 dan dijelaskan pada paper yang berjudul *image Synthesizer*. Pada paper tersebut menjelaskan implementasi dari fungsi *noise* yaitu sebagai berikut [22].

1. Menentukan himpunan dari semua titik dalam ruang 3D yaitu x , y , dan z dimana dari titik titik tersebut semua koordinat bernilai *integer*. Himpunan tersebut dapat disebut sebagai *integer lattice*.
2. Kemudian kaitkan dengan setiap titik dalam *integer lattice pseudo random* dan nilai gradien x , y , dan z . Setiap urutan tiga bilangan *integer* ditentukan ke dalam urutan *pseudo-random* dari 4 bilangan a , b , c , d .

$$[a, b, c, d] = H([x, y, z]) \quad (2.1)$$

$[a, b, c, d]$ mendefinisikan persamaan linear dengan gradien tiga dimensi $[a, b, c]$. x, y , dan z merupakan titik-titik koordinat yang bernilai *integer* dan nilai d merupakan nilai *noise* pada $[x, y, z]$. $H()$ adalah diimplementasikan sebagai fungsi hash.

3. Jika fungsi *Noise* $[x, y, z]$ merupakan bilangan *integer*, maka dapat didefinisikan sebagai berikut.

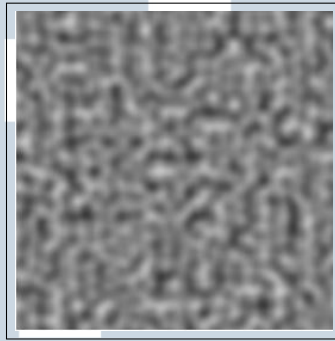
$$\text{Noise}([x, y, z]) =^d [x, y, z] \quad (2.2)$$

Jika fungsi *Noise* $[x, y, z]$ tidak berada pada bilangan bulat, maka dilakukan perhitungan secara *smooth* interpolasi persamaan *integer lattice* yang diterapkan pada sumbu x , y dan z .

Ketika *procedure* diatas selesai maka bisa lanjut pada tahap selanjutnya adalah melakukan kalkulasi interpolasi terhadap titik (x, y, z) dengan menghitung nilai gradien dengan *pseudo-random* di masing-masing delapan *vertices* terdekat pada *integer lattice*. Dapat dirumuskan sebagai berikut

$$s(t) = 3t^2 - 2t^3 \quad (2.3)$$

s adalah fungsi dari interpolasi dan t merupakan nilai gradien dari titik a, b, c terdekat. Algoritma ini memiliki nilai *output* yang dapat direpresentasikan dengan warna, dimana nilai yang mendekati 0,0 ditunjukkan dengan warna yang lebih gelap, dan nilai mendekati 1,0 untuk warna yang cerah.



Gambar 2.2. Perlin Noise

2.5 Game User Experience Satisfaction Scales

Game User Experience Satisfaction Scales adalah metode skala permainan yang tervalidasi dan komprehensif dengan tujuan untuk melakukan pengukuran dan mengevaluasi suatu permainan. GUESS terdiri dari 55 pertanyaan berdasarkan 9 *subscales* yaitu *Usability/Playability*, *Narratives*, *Play Engrossment*, *Enjoyment*, *Creative Freedom*, *Audio Aesthetics*, *Personal Gratification*, *Social Connectivity*, dan *Visual Aesthetics* [23]. Berikut adalah penjelasan dari masing-masing *subscales*.

1. *Usability/Playability* : Kemudahan dimana *game* dapat dimainkan dengan tujuan atau sasaran yang jelas serta dapat mengurangi kesulitan untuk mengakses antarmuka dan kontrol pengguna.
2. *Narratives* : Aspek cerita pada permainan yang berkaitan dengan peristiwa, karakter dan kemampuan permainan untuk menarik minat pemain dan membentuk emosi pemain.
3. *Play Engrossment* : Kemampuan seberapa jauh permainan dapat menarik perhatian dan minat pemain.
4. *Enjoyment* : Kesenangan dan kegembiraan yang dirasakan oleh pemain sebagai hasil dari memainkan permainan tersebut.

5. *Creative Freedom* : Seberapa jauh permainan mampu meningkatkan kreativitas dan keingintahuan pemain dan memungkinkan pemain untuk mengekspresikan individualitasnya secara bebas saat bermain *game*.
6. *Audio Aesthetics* : Aspek dari berbagai suara yang dimasukkan ke dalam *game* untuk memberikan kesan yang menarik ketika bermain *game*.
7. *Personal Gratification* : Aspek yang dapat memberikan motivasi pemain agar memiliki keinginan untuk terus bermain permainan dengan tujuan menyelesaikan permainan.
8. *Social Connectivity* : Seberapa jauh permainan memfasilitasi hubungan sosial antara pemain melalui alat dan fitur pada permainan.
9. *Visual Aesthetics* : Memiliki grafik permainan yang menarik pada permainan dengan tujuan agar pemain memiliki minat untuk memainkan *game* tersebut.

Dalam mengukur tingkat kepuasan pemain dalam permainan, pengujian pada permainan dilakukan berdasarkan skala likert yang berisi beberapa pertanyaan yang dapat dipilih satu opsi di setiap pertanyaan oleh pengguna. Dalam penilaian menggunakan GUESS jika ada *game* yang dievaluasi tidak memiliki salah satu aspek dari 9 *subscales*, maka memiliki opsi untuk menghilangkan *subscales* tersebut dari penilaian keseluruhan [23]. Dikatakan pada penjelasan sebelumnya jika pada pengisian terhadap GUESS memiliki 55 pertanyaan. Namun jika pertanyaannya sebanyak itu maka tidak semua orang akan mau atau merasa malas untuk mengisi 55 pertanyaan kuisisioner, sehingga untuk mengatasi hal tersebut digunakanlah GUESS-18 yang hanya memiliki 18 pertanyaan berdasarkan 9 *subscales* pada GUESS sebelumnya [24]. Pengisian kuisisioner GUESS sebelumnya membutuhkan waktu rata-rata 10-15 menit untuk menyelesaikannya, sedangkan GUESS-18 hanya membutuhkan waktu sekitar 3-5 menit [24]. Hasil dari GUESS-18 memberikan hasil model pengukuran yang kuat untuk data yang dianalisis. Seperti GUESS versi lengkap, GUESS-18 memiliki skala likert 7 poin yang dapat digunakan sebagai berikut [23].

1. Sangat Tidak Setuju bernilai = 1
2. Tidak Setuju bernilai = 2
3. Cukup Tidak Setuju bernilai = 3
4. Netral bernilai = 4

5. Cukup Setuju bernilai = 5

6. Setuju bernilai = 6

7. Sangat Setuju = 7

Penilaian yang dilakukan dengan cara mencari skor rata-rata dari setiap *sub-scales* lalu skor rata-rata dari setiap *subscales* dapat dijumlahkan untuk mendapatkan hasil dari tingkat kepuasan pada permainan [24].

$$\begin{aligned} \text{Rata - Rata}(\%) = & ((\text{JumlahSangatTidakSetuju} * 1) + (\text{JumlahTidakSetuju} * 2) \\ & + (\text{JumlahCukupTidakSetuju} * 3) + (\text{JumlahNetral} * 4) \\ & + (\text{JumlahCukupSetuju} * 5) + (\text{JumlahSetuju} * 6) \\ & + (\text{JumlahSangatSetuju} * 7)) / (\text{JumlahResponde} * \text{SkalaTertinggi}) \\ & * 100\% \end{aligned} \tag{2.4}$$

Hasil nilai presentase dari GUESS-18 yang didapatkan dari setiap pertanyaan dapat dibagi menjadi beberapa bagian yaitu.

1. Interval nilai 0% sampai 14% mendapatkan hasil = Sangat Buruk
2. Interval nilai 15% sampai 28% mendapatkan hasil = Buruk
3. Interval nilai 29% sampai 42% mendapatkan hasil = Cukup Buruk
4. Interval nilai 43% sampai 56% mendapatkan hasil = Netral
5. Interval nilai 57% sampai 70% mendapatkan hasil = Cukup Baik
6. Interval nilai 71% sampai 84% mendapatkan hasil = Baik
7. Interval nilai diatas 84% mendapatkan hasil = Sangat Baik