

BAB 2

LANDASAN TEORI

2.1 Cloud Computing

Cloud computing adalah model untuk memungkinkan akses jaringan yang nyaman dan sesuai permintaan ke kumpulan sumber daya komputasi yang dapat dikonfigurasi (jaringan, server, penyimpanan, aplikasi, dan layanan) yang dapat dengan cepat disediakan dan dirilis dengan upaya manajemen serta minimal interaksi dengan penyedia layanan [24]. *Cloud computing* memiliki beberapa karakteristik yang penting, karakteristik tersebut dijabarkan sebagai berikut [4].

1. *On-demand self-service*

layanan *cloud computing* seperti penyimpanan, akses jaringan, server, aplikasi web, dll dapat dialokasikan secara otomatis sesuai dengan kebutuhan dan tanpa campur tangan manusia.

2. *Broad network access*

Kapabilitas yang tersedia melalui jaringan dan diakses melalui mekanisme standar yang mempromosikan penggunaan oleh platform klien yang heterogen

3. *Measured Service*

Sistem *cloud* secara otomatis mengontrol dan mengoptimalkan sumber daya penggunaan dengan memanfaatkan kemampuan pengukuran pada beberapa tingkat abstraksi yang sesuai dengan jenis layanan. Penggunaan sumber daya dapat dipantau, dikontrol, dan dilaporkan transparansi bagi penyedia dan konsumen dari layanan yang digunakan.

4. *Resource pooling*

Sumber daya komputasi fisik dan virtual digabungkan ke dalam suatu *cloud* sehingga sumber daya tersebut tidak bergantung kepada suatu lokasi tertentu.

5. *Rapid elasticity*

Sumber daya komputasi dapat dengan cepat disediakan dan dilepaskan secara fleksibel berdasarkan permintaan pelanggan seolah-olah sumber daya tidak

terbatas dan dapat dibeli kapan pun dan dalam jumlah berapa pun sesuai keinginan pelanggan.

2.2 Game Elements

Dalam mendesain sebuah game terdapat elemen-elemen yang menjadi dasar pembuatan game, yaitu *Formal Elements* dan *Dramatic Elements*. *Formal elements* merupakan elemen yang mendesain struktur sebuah game secara keseluruhan. *Formal elements* terbagi menjadi 8 (delapan) bagian [20].

1. *Players*, merupakan elemen untuk pemain yang memainkan *game* berinteraksi.
2. *Objectives*, merupakan elemen yang menentukan tujuan utama dari *game*.
3. *Procedures*, merupakan elemen yang berisikan instruksi untuk memainkan sebuah *game*.
4. *Rules*, merupakan elemen yang menetapkan peraturan yang sudah ditentukan selama bermain *game*.
5. *Resources*, merupakan elemen yang menentukan sumber daya, seperti item dalam *game* untuk membantu pemain selama bermain *game*.
6. *Conflict*, merupakan elemen yang menentukan permasalahan utama dari narasi yang diceritakan dalam *game*.
7. *Boundaries*, merupakan elemen yang menentukan batas-batas serta limitasi dari sebuah *game*.
8. *Outcome*, merupakan elemen yang menentukan hasil yang akan diperoleh pemain saat menyelesaikan *game*.

Dramatic element merupakan elemen yang bertujuan untuk menarik perhatian dari pemain untuk menciptakan sebuah pengalaman bermain *game* yang dramatis. *Dramatic element* terbagi menjadi 6 (enam) bagian [20].

1. *Challenge*, elemen yang memberikan tantangan kepada pemain selama bermain *game* dengan tujuan agar alur bermain *game* tidak terlalu polos.

2. Play, elemen yang menentukan tipe dari sebuah *game*. Play terdiri dari 2 jenis yaitu *free-form play* dan *rule-based play*. *Free-form play* adalah jenis improvisasi bermain, sedangkan *rule-based play* menggunakan aturan dari pemain.
3. Premise, merupakan elemen dramatisasi berupa cerita latar belakang dari sebuah narasi pada *game*.
4. Character, elemen yang menentukan bahwa pemain berperan sebagai tokoh utama dalam sebuah *game*.
5. Story, berupa elemen dalam bentuk narasi cerita pada sebuah *game*.
6. World Building, merupakan elemen yang berupa ruang permainan dari sebuah *game*.

2.3 Game Design Document (GDD)

Game design document adalah sebuah dokumen yang berisikan informasi detail mengenai permainan yang akan dikembangkan. GDD ditujukan untuk mempermudah kolaborasi antar pengembang *game* untuk menciptakan suatu permainan [25]. Seiring dengan perkembangan permainan, GDD akan mengalami perubahan yang berkelanjutan. Perubahan ini disebabkan oleh berbagai komponen utama yang terdapat dalam GDD, seperti ringkasan permainan, desain level, desain permainan, desain antarmuka pengguna, desain konten, evaluasi permainan, dan desain sistem [26].

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

2.4 Binary Space Partitioning

Algorithm 1 Binary Space Partitioning Algorithm

function BINARYSPACEPARTITION(L, S, th, bs)

if $S.w \times S.h > th$ **then**

if $S.w > S.h$ **then**

$r \leftarrow \text{RandomInt}(S.x, S.x + S.w)$

$S1 \leftarrow \text{Space}(S.x, S.y, r - S.x, h)$

$S2 \leftarrow \text{Space}(r, S.y, w - (r - S.x), h)$

 BINARYSPACEPARTITION($L, S1, th, bs$)

 BINARYSPACEPARTITION($L, S2, th, bs$)

else

$r \leftarrow \text{RandomInt}(S.y, S.y + S.h)$

$S1 \leftarrow \text{Space}(S.x, S.y, w, r - S.y)$

$S2 \leftarrow \text{Space}(S.x, r, w, h - (r - S.y))$

 BINARYSPACEPARTITION($L, S1, th, bs$)

 BINARYSPACEPARTITION($L, S2, th, bs$)

end if

else

$q \leftarrow \text{Random}(0, 1)$

if $q > bs$ **then**

 APPENDTOLIST(L, S)

end if

end if

return L

end function

Binary Space Partition (BSP) adalah algoritma partisi yang secara rekursif membagi ruang yang diberikan menjadi dua bagian yang lebih kecil sampai kondisi akhir terpenuhi [27]. Algoritma *Binary Space Partitioning* (BSP) Dungeon adalah

salah satu algoritma yang paling umum digunakan dan algoritma ini didasarkan pada pembagian ruang dan penambahan ruang berikutnya di ruang yang dihasilkan, berdasarkan pohon BSP.

Pseudocode 1 merupakan pseudocode *binary space partitioning*. Parameternya adalah L, yang merupakan *array list* untuk menyimpan ruang yang dihasilkan, S adalah ruang yang akan dikerjakan oleh algoritma, th adalah batas untuk ukuran ruang, dan terakhir bs adalah probabilitas untuk memblokir sebuah ruang. Algoritma pertama menemukan sisi yang lebih panjang, baik lebar, atau tinggi untuk menentukan arahnya untuk membagi ruang. Bergantung pada arahnya, algoritma ini secara acak memilih posisi di tepi horizontal atau vertikal dan terus beroperasi pada ruang yang baru dibuat secara rekursif. Setelah ruang menjadi lebih kecil dari ambang batas, algoritma akan membandingkan nilai yang dihasilkan secara acak dengan parameter bs, dan jika nilainya lebih besar, menambahkan S ke daftar ruang yang dilambangkan dengan L. Dalam *list*, fungsi-fungsi *Random*, *RandomInt*, *AppendToList* direferensikan sebagai fungsi utilitas yang akan tersedia pada pustaka bahasa pemrograman apa pun. Variabel S adalah sebuah objek, dan *Space* direferensikan sebagai konstruktornya [18].

2.5 Random Walk

Random Walk adalah algoritma ritme paling dasar yang digunakan untuk menghasilkan level seperti sebuah gua. Algoritma *Random walk* bekerja dengan cara memilih sebuah titik acak dan bergerak secara acak [28]. Jalur ini diulang-ulang sampai mencapai level yang diinginkan.

$$S_n = \sum_{i=1}^n X_1 + \dots + X_n \quad (2.1)$$

Rumus 2.1 merupakan rumus *random walk*. S_n yang terdapat pada rumus merupakan jumlah langkah yang bergerak secara acak dari langkah X_n , lalu S_n akan membentuk sebuah jalan dari jumlah langkah X_n [29].

$$S_{t+1} = S_t + W_t \quad (2.2)$$

Rumus 2.2 juga rumus dari *random walk*. S_t merupakan keberadaan dari lokasi saat ini yang didasarkan pada nilai dari t, dan W_t merupakan langkah atau variabel acak dengan nilai distribusi [30].

2.6 Game User Experience Satisfaction Scale

Game User Experience Satisfaction Scale (GUESS) adalah alat pengukuran kepuasan pemain terhadap suatu game yang telah dikembangkan. Dengan 55 pertanyaan, GUESS terstruktur dalam sembilan subskala yang dapat diselesaikan dalam rentang waktu lima hingga sepuluh menit. Subsкала-subskala tersebut mencakup *Usability/Playability*, Narasi, Keterlibatan Bermain, Kesenangan, Kebebasan Kreatif, Estetika Audio, Kepuasan Pribadi, Koneksi Sosial, dan Estetika Visual [31]. Namun dalam mengisi kuesioner, terutama yang memiliki total pertanyaan sebanyak 55 butir, seringkali membutuhkan waktu yang cukup lama untuk menyelesaikannya. Oleh karena itu, digunakan versi singkat GUESS yang disebut GUESS-18, yang terdiri dari hanya 18 pertanyaan. Meskipun lebih ringkas, kuesioner ini tetap mencakup kesembilan konstruk GUESS yang sama. Menggunakan GUESS-18 akan memerlukan waktu yang lebih singkat, sekitar 3-5 menit, untuk menyelesaikan kuesioner, sementara masih dapat memberikan hasil yang akurat. Kuesioner ini menggunakan skala Likert dengan tujuh poin untuk menilai respons, dan setiap poin peringkat tersebut dijelaskan dengan jelas [32].

Peringkat penggunaan skala likert dalam 7 poin penilaian yang digunakan untuk mengukur jawaban responden dapat diperhatikan seperti sebagai berikut.

- Sangat Setuju (SS) dengan nilai 7
- Setuju (S) dengan nilai 6
- Cukup Setuju (CS) dengan nilai 5
- Netral (N) dengan nilai 4
- Cukup Tidak Setuju (CTS) dengan nilai 3
- Tidak Setuju (TS) dengan nilai 2
- Sangat Tidak Setuju (STS) dengan nilai 1

Setiap pertanyaan yang dijawab oleh responden akan memiliki jumlah masing-masing yang berdasarkan dari nilai yang dipilih, selanjutnya setiap nilai tersebut akan dihitung kembali untuk memperoleh nilai rata-rata menggunakan rumus berikut.

$$\begin{aligned}
 \text{Rata - rata}(\%) = & ((\text{JumlahSTS} \times 1) + (\text{JumlahTS} \times 2) + (\text{JumlahCTS} \times 3) + \\
 & (\text{JumlahN} \times 4) + (\text{JumlahCS} \times 5) + (\text{JumlahS} \times 6) + \\
 & (\text{JumlahSS} \times 7)) / (\text{JumlahResponden} \times \text{SkalaTertinggi}) \times 100\%
 \end{aligned}
 \tag{2.3}$$

Tabel 2.1. Interval Skala GUESS

Interval (%)	Deskripsi
0 - 14	Sangat Buruk
15 - 28	Buruk
29 - 42	Cukup Buruk
43 - 56	Netral
57 - 70	Cukup Baik
71 - 84	Baik
> 84	Sangat Baik

2.7 Skala Likert

Skala Likert adalah alat pengukuran yang sering digunakan dalam penelitian survei. Keberhasilan skala ini dalam penelitian disebabkan oleh kemampuannya untuk mengukur persepsi, sikap, dan pendapat seseorang terhadap suatu topik atau peristiwa yang menjadi fokus penelitian. Skala Likert biasanya terdiri dari serangkaian pertanyaan atau pernyataan, di mana responden diminta untuk menilai sejauh mana mereka setuju atau tidak setuju dengan pernyataan tersebut, menggunakan tujuh opsi yang tersedia [33].

Tabel 2.2. Skala Likert

Pernyataan	Skor Positif	Skor Negatif
Sangat Setuju	7	1
Setuju	6	2
Agak Setuju	5	3
Netral	4	4
Agak Tidak Setuju	3	5
Tidak Setuju	2	6
Sangat Tidak Setuju	1	7

Selanjutnya dari hasil pengisian skala likert akan digolongkan sesuai dengan Tabel 2.3.

Tabel 2.3. Interval Skala GUESS

Jumlah Skor	Kriteria
81% - 100%	Sangat Baik
61% - 80%	Baik
41% - 60%	Cukup Baik
21% - 40%	Kurang Baik
0% - 20%	Tidak Baik

Perhitungan persentase skor sebagai berikut.

$$((f1 \times 1) + (f2 \times 2) + (f3 \times 3) + (f4 \times 4) + (f5 \times 5) + (f6 \times 6) + (f7 \times 7)) / (7 \times fr) \times 100 \quad (2.4)$$

Berikut keterangan dari rumus perhitungan persentase skor.

1. f1 = jumlah responden yang menjawab sangat tidak setuju untuk pernyataan positif atau sangat setuju untuk pernyataan negatif.
2. f2 = jumlah responden yang menjawab tidak setuju pada pernyataan positif atau setuju pada pernyataan negatif.
3. f3 = jumlah responden yang menjawab agak tidak setuju pada pernyataan positif atau agak setuju pada pernyataan negatif.
4. f4 = jumlah responden yang menjawab netral pada pernyataan positif maupun negatif.
5. f5 = jumlah responden yang menjawab agak setuju pada pernyataan positif atau agak tidak setuju untuk pertanyaan negatif.
6. f6 = jumlah responden yang menjawab setuju pada pernyataan positif atau tidak setuju pada pernyataan negatif.
7. f7 = jumlah responden yang menjawab sangat setuju untuk pernyataan positif atau sangat tidak setuju untuk pernyataan negatif.
8. fr = jumlah responden