



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

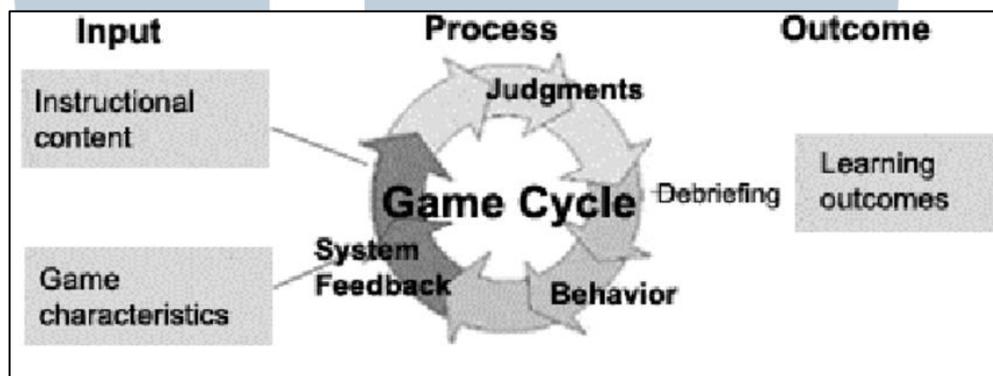
### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB II LANDASAN TEORI

### 2.1 Game-Based Learning

*Game-based learning* adalah metode pembelajaran yang menggunakan elemen permainan dan simulasi untuk memfasilitasi proses pembelajaran. Tujuan utama dalam penggunaan permainan dalam metode pembelajaran adalah keuntungannya yang melibatkan pengguna dalam permainan. (Mathis, 2008).



Gambar 2.1 *Game-Based Learning Model* (Garris & Ahlers, 2002)

Pada *Game-Based Learning*, yang dilakukan pertama kali adalah mendisain program yang instruksional yang menggabungkan fitur atau karakteristik dari *game*. Selanjutnya penggabungan kedua fitur ini akan menghasilkan sebuah siklus yang didalamnya termasuk penilaian pengguna atau reaksi seperti kesenangan atau niat, tingkah laku pengguna seperti waktu penyelesaian, dan umpan balik sistem. Ketika instruksional program dan karakteristik *game* berhasil disatukan dengan baik, siklus ini akan menghasilkan *game play* yang berulang dan memotivasi pengguna. Tahap terakhir akan menghasilkan permainan yang mengacu pada pencapaian objektif dan hasil pembelajaran (Garris & Ahlers, 2002).

## 2.2 Struktur Game

Pada tiap permainan, ada berbagai jenis elemen yang membentuk permainan tersebut. Elemen ini terdiri dari dua jenis, yaitu elemen formal dan elemen dramatis. Elemen formal adalah elemen yang membentuk struktur dari suatu permainan, sementara elemen dramatis adalah elemen yang digunakan untuk memberikan konteks pada permainan. Kedua elemen ini dikombinasikan untuk menjadi suatu hal yang dapat membuat pemainnya tertarik (Fullerton, 2008).

### 2.2.1 Formal Elements

Jenis – jenis dari elemen formal permainan adalah sebagai berikut (Fullerton, 2008).

#### 1. Players

Sebuah permainan adalah pengalaman yang didisain untuk pemain. Pemain harus mengikuti aturan dan batasan agar dapat bermain untuk memenuhi objektif permainan.

#### 2. Objectives

Suatu objective atau tujuan dalam permainan berfungsi sebagai tujuan pemain dalam memainkan sebuah permainan. Objective yang baik adalah objective yang dapat memberi pemainnya tantangan yang sulit tetapi masih dapat diselesaikan oleh pemain.

#### 3. Procedures

*Procedure* adalah metode permainan yang dapat dilakukan pemain untuk mencapai *objective*.

4. Rules

Rules menentukan tindakan yang boleh dan tidak boleh dilakukan oleh pemain dalam memainkan permainan.

5. Resources

Resources merupakan aset yang dimiliki atau diberikan kepada pemain untuk menyelesaikan suatu objective tertentu.

6. Conflict

Conflict adalah sesuatu yang membuat pemain tidak dapat menyelesaikan objective dengan mudah karena tindakan yang dibatasi oleh rules.

7. Boundaries

Boundaries adalah hal-hal yang memisahkan permainan dari hal-hal lain yang bukan permainan. Sebagai contoh yaitu keterbatasan gerakan pemain dalam suatu map dalam permainan karena adanya tembok yang membatasi pemain.

8. Outcome

Outcome merupakan hasil dari permainan yang telah selesai atau sesuatu yang didapat pemain setelah permainan berhasil diselesaikan. Outcome dapat berwujud sebagai skor dalam sebuah permainan.

### 2.2.2 Dramatic Elements

Selain formal elements, ada juga dramatic elements yang akan dijelaskan sebagai berikut (Fullerton, 2008).

1. Challenge

*Challenge* adalah tantangan yang dihadapi pemain dalam suatu permainan. Tantangan ini harus berupa suatu misi yang memberikan rasa kepuasan saat diselesaikan. Tantangan harus memiliki tingkat kesulitan yang sesuai agar tidak terlalu sulit untuk diselesaikan tetapi juga tidak terlalu mudah sehingga pemain menjadi bosan.

2. Play

*Play* dapat dianggap sebagai suatu kebebasan dalam dunia permainan. Permainan dibatasi oleh peraturan dan batasan-batasan, dan *play* adalah kebebasan yang dimiliki pemain di dalam aturan dan batasan tersebut.

3. Premise

*Premise* adalah elemen drama tradisional dalam permainan seperti *background story* atau *setting* dari cerita sebuah permainan. Jika tidak ada premise, cerita permainan akan terasa terlalu abstrak bagi pemainnya.

4. Characters

*Characters* atau karakter adalah media interaksi pemain di dalam dunia permainan. Karakter adalah yang berperan dalam suatu cerita dalam permainan dan akan melakukan segala hal sesuai pilihan pemain.

5. Story

*Story* adalah cerita dari sebuah permainan yang akan dibaca oleh pemain seiring pemain bermain. Sebuah cerita yang menarik akan membuat pemainnya tertarik sehingga pemain menjadi ingin menyelesaikan permainannya untuk mengetahui *outcome* dari cerita tersebut.

6. World Building

*World Building* adalah desain dunia fiksi yang diciptakan oleh permainan, seperti map atau peta dari dunia dalam permainan tersebut.

### 2.3 Algoritma

Konsep dari algoritma sebenarnya sudah ada sejauh jaman dulu kala, sebelum terbuatnya komputer pertama. Saat ini, definisi universal masih belum disepakati. (Moschovaskis, 2001). Secara intuitif, algoritma adalah sebuah urutan dari aturan-aturan yang harus dieksekusi sesuai dengan susunan yang tepat agar dapat menjalankan tugasnya dengan baik. Maka dari itu, algoritma adalah contoh dari logika yang menghasilkan *output* dari *input* yang diberikan (OECD, 2017).

Terdapat 10 algoritma yang digunakan dalam penelitian ini. Algoritma-algoritma ini dipilih berdasarkan apa yang telah diajarkan di bangku perkuliahan Universitas Multimedia Nusantara. Algoritma tersebut adalah sebagai berikut.

#### 1. Breadth-First Search

*Breadth-First Search* atau BFS ini masuk dalam kategori *uninformed search*, yang berarti BFS bukan merupakan fungsi heuristik dan bukan juga merupakan fungsi pencarian objektif. BFS dimulai dari simpul awal, lalu menyebar ke tetangga terdekatnya hingga sampai pada simpul yang diinginkan (Everitt & Hutter, 2018).

#### 2. Depth-First Search

Sama seperti BFS, *Depth-First Search* atau DFS juga termasuk *uninformed search*. Berbeda dengan BFS, DFS mengikuti satu jalur selama-lamanya, lalu ketika tidak dapat maju lagi, DFS akan melakukan *backtracking*. Proses

ini akan terus diulang hingga mencapai objektif yang diinginkan (Everitt & Hutter, 2018).

### 3. Binary Search

*Binary search* adalah tipe algoritma pencarian yang menggunakan data yang telah disortir terlebih dahulu. Karakteristik utama *binary search* adalah membagi dua data yang telah disortir, lalu membandingkan data pada titik tengah dengan objek yang ingin dicari. Dengan cara ini, ketika data tengah lebih kecil dari data yang dicari, setengah data sebelum titik tengah dapat diabaikan, dilanjutkan dengan membagi dua setengah data selanjutnya, dan terus berulang hingga objek tercapai (Oommen & Pal, 2014).

### 4. A\* Algorithm

Algoritma A\* adalah salah satu algoritma terbaik yang paling umum digunakan untuk *pathfinding*, terutama untuk AI dalam *game*. Algoritma A\* selalu memeriksa semua lokasi yang belum dieksplor. Ketika sudah dieksplor, ia akan mencatat semua lokasi tetangganya untuk eksplorasi selanjutnya sampai menuju titik objektifnya (Cui & Shi, 2011).

### 5. Prim's Algorithm

Algoritma Prim dimulai dari *root* lalu menyebar ke simpul terdekat dengan memilih berat yang paling kecil. Proses ini terus dilakukan sampai semua simpul tereksplor namun tanpa ada simpul yang dikunjungi dua kali (T, B, & T, 2011).

### 6. Kruskal's Algorithm

Algoritma Kruskal dibuat oleh Joseph Kruskal pada 1956. Algoritma Kruskal dimulai dari menyortir simpul ke simpul berdasarkan beratnya. Lalu

memilih berat yang paling kecil terlebih dahulu, hingga semua simpul dieksplor namun tanpa ada simpul yang dikunjungi dua kali (R, 2012).

7. Genetic Algorithm

*Genetic Algorithm (GA)* adalah sebuah algoritma pencarian, terutama dalam hal optimisasi, yang mencari *the fittest individual* dari sebuah populasi. GA dibentuk dari individual yang diset secara acak. Objektif dari GA biasanya memaksimalkan atau meminimumkan nilai dari individual ini. Populasi selanjutnya terbentuk dari hasil *crossover* dengan tingkat mutasi yang kecil pada populasi sebelumnya (Mardle, Pascoe, & Tamiz, 1998).

8. Bubble Sort

*Bubble Sort* adalah algoritma yang dimulai dengan membandingkan dua objek pertama, jika objek pertama lebih besar dibanding yang kedua, maka posisi mereka akan ditukar. Lalu pindah ke objek kedua dan ketiga sampai akhir. Diulang lagi dari awal, dan akan berhenti atau dikatakan berhasil ketika tidak ada pertukaran yang terjadi dari awal sampai akhir (Astrachan, 2003).

9. Backpropagation

Backpropagation merupakan salah satu metode *machine learning* yang menyerupai jaringan syaraf otak yang mempunyai bentuk fleksibel. Metode ini sangat baik untuk menangani masalah pengenalan pola-pola kompleks.

Setiap unit yang berada di *input layer* terhubung dengan setiap unit pada *hidden layer*. Setiap unit pada *hidden layer* terhubung pada *output layer* (Fa'rifah & Busrah, 2017).

10. Neural Network

*Neural Network* sama seperti *biological neuron* pada manusia. Algoritma ini memiliki banyak *input* dan hanya satu *output*. Di dalamnya terdapat banyak proses sederhana yang saling berhubungan satu sama lain (Sharma, Rai, & Dev, 2012).

#### **2.4 Pretest-Posttest**

Pretest-Posttest adalah pengembangan dari metode Posttes. Pada metode ini terdapat dua grup, satu yang mendapatkan perlakuan (eksperimen grup), satu lagi yang tidak mendapatkan perlakuan (kontrol grup) (Dimitrov & Rumrill, Jr, 2003).

Analisa dilakukan antara keduanya untuk menentukan cara mana yang memiliki efek yang lebih besar. Setelah dilakukan *pretest* dan *posttest*, dilakukan penghitungan nilai rata – rata *gain* dari setiap grup, lalu akan dibandingkan. Jika hasil dari eksperimen grup lebih tinggi, maka aplikasi ini dapat dikatakan berhasil.

#### **2.5 Hedonic Motivation System Adoption Model (HMSAM)**

Berdasarkan Domarco, D., & Iswari, N. M. S. (2017), HMSAM adalah sebuah model untuk mengukur sebuah sistem yang mengadaptasi motivasi hedonis. Ada lima faktor yang menjadi fokus pengukuran pada HMSAM, yaitu *perceived usefulness*, *perceived ease of use*, *curiosity*, *control*, dan *joy* yang akan mempengaruhi *behavioral intentional to use* dan *immersion* suatu aplikasi.

Aspek-aspek tersebut adalah sebagai berikut:

1. *Perceived usefulness*, persepsi kegunaan; berguna atau tidaknya aplikasi menurut pengguna.

2. *Perceived ease of use*, persepsi kemudahan penggunaan oleh pengguna. Kurangnya bobot ini akan mengakibatkan pengguna frustrasi, lebih acuh dan akan berhenti memainkan permainan
3. *Curiosity*, sejauh mana suatu sistem dapat meningkatkan rasa ingin tahu dalam aspek kognitif.
4. *Control*, persepsi pengguna bahwa dirinya yang berkuasa dalam interaksi.
5. *Joy*, aspek kesenangan yang didapatkan oleh pengguna dari interaksi dengan sistem.
6. *Behavioral intention to use*, keinginan pengguna untuk menggunakan aplikasi.
7. *Immersion*, total keterlibatan antara pengguna dan aplikasi.

Berikut merupakan rumus untuk menghitung BIU dan I (Santo & Satvika, 2017).

$$\text{Persentase Skor BIU} = \frac{PU + Cu + J}{3} \quad \dots (2.1)$$

$$\text{Persentase Skor I} = \frac{Cu + J + C}{3} \quad \dots (2.2)$$

Keterangan:

BIU = *Behavioral Intention to Use*

I = *Immersion*

PU = *Perceived Usefulness*

Cu = *Curiosity*

J = *Joy*

C = *Control*

## 2.6 Skala Likert

Banyak skala pengukuran yang dapat digunakan dalam merancang skala pengukuran pada penelitian dan skala likert bertujuan untuk menduga preferensi individu dengan menggunakan nilai responnya (Budiarji, 2013). Skala likert yang digunakan memiliki tujuh titik pilihan pada setiap butir pertanyaan dan memiliki nilainya masing-masing. Cara menghitungnya adalah dengan cara mencari dahulu total skor masing-masing pilihan responden lalu mengalikannya dengan tujuh untuk pilihan sangat setuju dan terus menurun dengan yang sangat tidak setuju dikalikan dengan 1 dan untuk mengetahui persentasenya masukan ke rumus indeks persentase berikut (Kusuma, 2016).

Tabel 2.1 Tabel Perhitungan Skala Likert

SS	7 x Responden yang menjawab	A
S	6 x Responden yang menjawab	B
MS	5 x Responden yang menjawab	C
N	4 x Responden yang menjawab	D
MTS	3 x Responden yang menjawab	E
TS	2 x Responden yang menjawab	F
STS	1 x Responden yang menjawab	G
	Total	Z

$$\text{Index Persentase} = \frac{Z}{Y} \times 100\% \quad \dots (2.3)$$

Keterangan :

SS = Sangat Setuju

TS = Tidak Setuju

S = Setuju

STS = Sangat Tidak Setuju

MS = Mungkin Setuju

Y = Total Responden x 7

N = Netral

MTS = Mungkin Tidak Setuju