



### **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 Artificial Intelligence and Smart Agent

Artificial Intelligence (AI) adalah kemampuan sistem untuk secara tepat menginterpretasikan data eksternal, mempelajari dari data tersebut, dan menggunakan data tersebut untuk mencapai tujuan tertentu melalui adaptasi yang fleksibel (Kaplan, 2018). Teknik AI terbagi menjadi dua, yaitu deterministik dan non-deterministik. AI deterministik merupakan perilaku yang dapat ditebak karena tidak mempunyai faktor ketidakpastian, contohnya AI yang secara sederhana mengejar karakter pemain. AI non-deterministik mempunyai ketidakpastian pada tingkat tertentu, sehingga perilakunya lebih sulit ditebak, misalnya AI yang dapat beradaptasi terhadap serangan pemain (Bates, 2004).

*Game* memiliki karakter yang disebut sebagai Non-Playable Character (NPC). NPC merupakan objek-objek dinamis yang tidak terikat terhadap kendali apapun dari pengguna dan memutuskan terhadap diri mereka sendiri dan beroperasi dalam ruang semu (Kim, dkk., 2006). NPC bertindak sebagai agen di dalam *game*. Dalam AI, agen merupakan segala sesuatu yang menerima masukan dari lingkungan tempatnya berada melalui sensor dan melakukan aksi terhadap masukan yang diterima melalui *actuator* (Russel, 2012). NPC dapat menjadi faktor kunci untuk melibatkan pemain dalam permainan dan memberikan pengalaman. Meningkatkan perilaku NPC dalam suatu *game* agar mereka dapat menjadi lebih alami, diklaim menjadi salah satu faktor penting untuk meningkatkan pengalaman pemain (Chowanda, dkk., 2016).

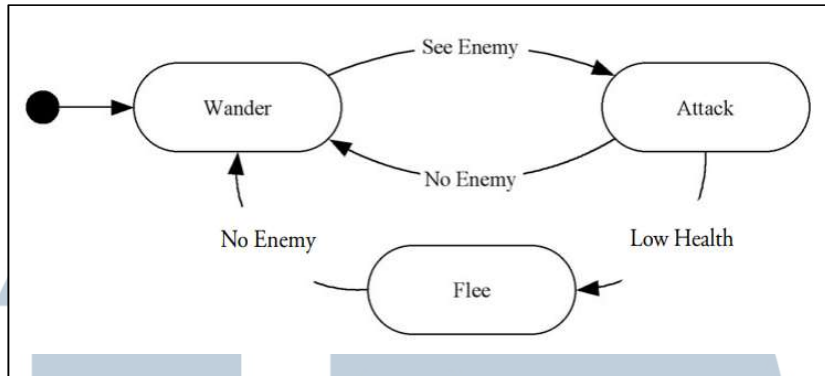
## 2.2 Rule-Based Reasoning

*Rule-based reasoning* (RBR) adalah salah satu paradigma yang populer digunakan dalam AI. Arsitektur *reasoning* dari sistem *rule-based* terdiri dari dua komponen utama, yaitu *knowledge base* dan *inference engine*. *Knowledge base* terdiri dari satuan aturan jika-maka yang merepresentasikan domain pengetahuan. *Inference engine* terdiri dari beberapa mekanisme inferensi domain-independen seperti *forward* atau *backward chaining* (Dutta, 1993).

### 2.2.1 Finite-State Machine

*Finite State Machine* (FSM) merupakan model abstrak dari komputasi yang terdiri dari beberapa *state*, *state* awal, *input vocabulary*, dan fungsi transisi yang memetakan input dan *state* aktif menuju ke *state* berikutnya (Rabin, 2010). Keunggulan dari FSM meliputi kesederhanaan konseptual, waktu eksekusi yang cepat, dan implementasi yang mudah (Watson, dkk., 2014).

FSM biasanya digunakan dalam praktek untuk mendefinisikan perilaku dari AI sederhana untuk karakter yang bukan pemain. Unit dalam permainan perang biasanya menggunakan *state* tersebut untuk menyerang, bertahan, dan patrol (Adams, dkk., 2013). FSM biasanya digunakan untuk mengorganisir dan merepresentasikan alur eksekusi, yang berguna dalam implementasi AI dalam *game*. “Otak” dari musuh, misalnya, dapat diimplementasikan menggunakan FSM. (Bevilacqua, 2013). Karena FSM secara elegan dapat menangkap kondisi mental atau perilaku dari agen, maka FSM merupakan pilihan natural untuk mendefinisikan karakter dari AI (Rabin, 2010).



Gambar 2.1 Diagram FSM perilaku agen dalam bentuk UML (Rabin, 2010).

Suatu FSM dapat direpresentasikan dalam bentuk sebuah graf, dimana *node* yang ada merupakan *state* dan *edges* merupakan transisi. Setiap *edge* memiliki label yang menginformasikan waktu terjadinya transisi (Bevilacqua, 2013).

Banyak cara yang dapat digunakan untuk mendefinisikan FSM. Cara yang paling mudah adalah pendekatan langsung, yaitu mengimplementasikan kode dari FSM ke dalam bahasa pemrograman yang digunakan *game* itu sendiri. Berikut contoh kode yang diimplementasikan dalam C++:

```

bool AgentFSM::States(StateMachineEvent event, MSG_Object * msg, state, substate )
{
    DeclareState( STATE_Wander )
        OnUpdate
            Wander();
            if( SeeEnemy() ) ChangeState( STATE_Attack );
        OnMsg( MSG_Attacked )
            ChangeState( Attack )
    DeclareState( STATE_Attack )
        OnEnter
            PrepareWeapon();
        OnUpdate
            Attack();
            if( LowOnHealth() )
                ChangeState( STATE_Flee );
    ...
}
  
```

Gambar 2.2 Contoh pseudokode FSM (Rabin, 2010)

Potongan kode pada Gambar 2.2 merupakan contoh dari *pseudocode* dari FSM pada Gambar 2.1 yang terdiri dari tiga *state* dan empat transisi. Menurut Rabin (2010), dengan menerapkan *event handler* yang akan berjalan dalam kondisi tertentu, diperoleh keuntungan sebagai berikut:

1. Struktur FSM di-enforce oleh apa yang akan diterima *script compiler*.
2. *Event* dapat ditangani (melalui konvensi *OnEvent*), begitu juga dengan *polling*.
3. Ketika *state* dimasukkan untuk pertama kalinya, konstruk *OnEnter* dapat digunakan untuk menjalankan inialisasi khusus apa pun. Sebaliknya, ada konstruksi *OnExit* untuk melakukan *cleanup code*, berlaku untuk apa pun yang memicu transisi. Konstruk *OnExit* membuat *script* lebih eksplisit dan mengurangi kode yang berlebihan.
4. FSM yang ditulis berdasarkan data dapat ditentukan oleh perancang *game* dan artis yang tidak terbiasa dengan bahasa pemrograman tradisional. Sayangnya, tidak mudah untuk membuat FSM

Konstruk `'Execute'` dan `'if'` menunjukkan bahwa fungsi tertentu harus dipanggil oleh nama yang ditentukan. Ini mengharuskan nama di dalam tanda kurung terikat pada nama fungsi sebenarnya dalam kode *game*. Dalam hal pernyataan `'if'`, fungsi akan mengembalikan Boolean sehingga skrip tahu apakah akan menjalankan pernyataan berikutnya. Perhatikan bahwa bahasa skrip khusus ini tidak memiliki kurung keriting dan titik koma, yang hanya merupakan pilihan desain bahasa (Rabin, 2010).

### 2.2.2 Forward Chain Reasoning

Algoritma ini terdiri dari tiga fase dasar, yaitu pencocokan peraturan berdasarkan fakta yang tersimpan dalam memori, eksekusi kejadian bila memenuhi kondisi, dan mencari aksi yang harus dieksekusi apabila terjadi beberapa aturan yang cocok. Pada fase pertama, sistem akan mencari aturan yang cocok tersimpan pada memori, ini merupakan bagian IF dari setiap aturan yang didefinisikan.

Kemudian sistem akan menjalankan kejadian yang dideskripsikan pada bagian THEN. Namun apabila terdapat beberapa aturan sama yang memenuhi kondisi, sistem dapat memilih aturan mana yang harus dijalankan. Hal ini mengarah pada fase yang disebut *conflict resolution phase*. (Bates, 2004).

Pada *conflict resolution phase*, semua aturan yang cocok harus di cek dan memilih kejadian mana yang harus dijalankan. Metode yang paling umum adalah menjalankan kejadian pada peraturan pertama. Dalam kasus lain, setiap aturan diberikan bobot dan aturan yang memiliki bobot paling tinggi akan dijalankan (Bates, 2004).

### 2.3 Uncertainty Factor

Satu masalah dengan *first-order logic*, dan dengan demikian pendekatan *logical-agent*, bahwa agen hampir tidak pernah memiliki akses ke seluruh kebenaran tentang lingkungan mereka. Beberapa pernyataan dapat dipastikan langsung dari persepsi agen, dan lainnya dapat disimpulkan dari persepsi saat ini dan sebelumnya bersama dengan pengetahuan tentang sifat-sifat lingkungan. Akan ada pertanyaan penting yang agennya tidak dapat menemukan jawaban kategoris. Agen karenanya harus bertindak di bawah ketidakpastian. Sebagai contoh, agen pada *game* Wumpus sering akan menemukan dirinya tidak dapat menemukan yang mana dari dua kotak berisi lubang. Jika kotak-kotak itu dalam perjalanan ke emas, maka agen mungkin harus mengambil kesempatan dan memasukkan salah satu dari dua kotak itu (Russel, 2012).

Beberapa hal berikut adalah penerapan probabilitas pada *game* menurut David Bourg (2004) sehingga dapat memberikan faktor ketidakpastian pada *game*:

1. Randomness: Pengembang umumnya menggunakan teknik ini untuk menghadirkan beberapa ketidakpastian kepada pemain, membuatnya lebih sulit untuk memprediksi di mana unit akan bergerak ketika berhadapan.
2. Hit Probability: Melibatkan karakter pada *game* dan pemain untuk melakukan *hit* terhadap lawan dalam pertempuran. Biasanya, pengembang *game* mendefinisikan beberapa probabilitas, mengingat karakteristik tertentu dari pemain dan lawannya.
3. Character Abilities: Mendefinisikan kemampuan kelas karakter atau jenis makhluk.
4. State Transitions: Kemampuan agen dapat ditingkat selangkah lebih maju dengan menggabungkan probabilitas dengan keadaan transisi pada FSM yang digunakan untuk mengelola berbagai *state* agen.
5. Adaptability: Penggunaan probabilitas yang agak lebih meyakinkan dalam *game* melibatkan pembaruan probabilitas tertentu saat *game* dimainkan dalam upaya memfasilitasi unit pembelajaran atau adaptasi yang dikendalikan komputer.

### 2.3 Game Design

*Game design* merupakan tindakan menentukan apa yang harusnya menjadi permainan dalam *game* tersebut (Schell, 2008). *Game* dapat digambarkan juga sebagai sebuah *finite state machine*. Suatu *game* bermula dari *initial state*, dan aksi yang dilakukan pemain dan mekanik yang ada dalam *game* tersebut, membawa pemain menuju *state* baru hingga *state* akhir dicapai. Dalam kasus ini, terutama pada *game single-player*, pemain akan menang, kalah atau keluar. *State* dari *game* biasanya mencerminkan lokasi pemain, pemain lain, teman, musuh, dan distribusi



untuk sumber daya penting yang ada dalam game tersebut. Dengan mengacu pada game sebagai *finite state machine*, peneliti dapat menentukan aturan yang membuat game berkembang dari satu *state* ke *state* lainnya. Beberapa metode yang sukses membuat ilmuwan komputer untuk merancang, memetakan model, dan mengimplementasi *state machine* dengan jumlah *state* yang terbatas (Bates, 2004).

Hasil akhir dari suatu *game* tidak boleh terlihat jelas dari awal permainan: hingga tingkat tertentu, *game* harusnya tidak bisa diprediksi. *Game* yang dapat diprediksi biasanya tidak seseru itu. Langkah mudah untuk menciptakan *game* yang memiliki hasil akhir yang sulit ditebak salah satunya dengan memberikan *element of chance*. Selain itu, menciptakan *game* yang memiliki hasil akhir yang sulit ditebak juga dapat dilakukan dengan memberikan pilihan pada pemain dan alur permainan kompleks yang diciptakan berdasarkan peraturan permainan (Adams, 2012).

Menurut Nacke (2014), suatu *game* dirancang berdasarkan *game design document* yang terbagi menjadi dua bagian, yaitu elemen formal dan elemen dramatis. Elemen formal membantu untuk membuat struktur sebuah *game*. Relasi antara elemen-elemen formal ini yang membentuk sebuah *game*. Elemen formal tersebut adalah *players, objectives, procedures, rules, resources, conflict, boundaries, outcome*. Memanfaatkan elemen formal saja dapat membuat sebuah *game* yang dapat dimainkan, namun hasilnya tidak akan menarik dengan sendirinya. Elemen dramatis harus dimanfaatkan semaksimal mungkin untuk menghasilkan pengalaman yang menawan.



### 2.4.1 Formal Elements

*Players* (pemain) merupakan volutir, partisipan aktif, dalam suatu lingkup aktivitas hiburan. Pemain mengambil bagian, menggunakannya dan berinvestasi di dalamnya. Desain game meminta pemain untuk berinteraksi antar pemain dan terhadap sistem dalam game tersebut (Nacke, 2014).

*Objectives* merupakan hal yang penting untuk memotivasi pemain untuk terlibat dalam suatu alur permainan. Tujuan akhir yang terbaik terlihat dapat diperoleh tetapi tetap harus dipandang sebagai hal yang menantang. Pemain harus menyelesaikan *objectives* untuk menjadi tolak ukur keterlibatan pemain dalam game tersebut. Beberapa tipe *objectives* adalah *capture*, *chase*, *race*, *alignment*, *rescue*, *forbidden act*, *construction*, *exploration*, *solution*, dan *outwit* (Nacke, 2014).

1. Capture: Pemain menghindari tertangkap atau terbunuh saat menghancurkan beberapa properti milik musuh (biasanya dalam bentuk *terrain* maupun *units*).
2. Chase: Pemain harus menghindari atau menangkap lawan.
3. Race: Pemain harus mencapai tujuan sebelum yang lain.
4. Alignment: Pemain harus mengurutkan pecahan-pecahan objek tertentu dalam bentuk spatial atau formasi konseptual.
5. Rescue / Escape: Pemain harus mendapatkan beberapa unit atau objek yang ditentukan agar aman tanpa dikompromikan.
6. Forbidden Act: Pemain harus membuat lawan untuk melanggar aturan atau meninggalkan strateginya.

7. Construction: Pemain harus membangun, menjaga atau memajemen objek-objek pada game.
8. Exploration: Pemain harus mengeksplorasi area game tertentu.
9. Solution: Pemain harus memecahkan masalah atau *puzzle* (Dalam kondisi tertentu sebelum musuh melakukannya).
10. Outwit: Pemain harus mendapatkan dan menggunakan *knowledge* untuk melampaui lawannya.

*Procedures* merupakan tindakan atau metode main yang diizinkan oleh aturan main *game*. Prosedur dapat berupa instruksi spesifik untuk aksi tertentu dalam game atau mereferensikan diri sebagai suatu set kontrol (Nacke, 2014).

*Rules* merupakan objek pasti dan konsep dari suatu game yang membangun blok-blok dari suatu sistem game. Segala aksi untuk setiap situasi yang dapat terjadi pada game dijelaskan sebagai aturan. Aturan ini merupakan hal yang dapat dan tidak dapat dilakukan oleh pemain (Nacke, 2014). Tiga tujuan utama peraturan pada *game* menurut Nacke (2014) adalah:

1. Mendefinisikan objek dan kondisi.
2. Mebatasi aksi pemain.
3. Menentukan efeknya pada pemain.

*Resources* merupakan objek pada game yang mempunyai nilai bagi pemain dalam mencapai tujuannya. Nilai dari objek ini dapat ditentukan dari penyebaran dan kegunaannya (Nacke, 2014).

*Conflict* muncul melalui prosedur dan aturan-aturan yang ada pada game, yang mencegah pemain untuk mencapai tujuannya. *Objectives* biasanya

mengarahkan pemain pada konflik ini (Nacke, 2014). Tipe konflik yang muncul pada *game* menurut Nacke (2014) adalah:

1. **Obstacles:** Merupakan hambatan yang dapat berbentuk fisik maupun mental. Hambatan fisik bisa berupa panjang pemantul pada game Pinball atau bumper yang dipantulkan bola. Hambatan mental dapat berupa barang yang hilang, yang digunakan untuk memecahkan teka-teki dalam suatu game, misalnya kunci pada game Resident Evil 2.
2. **Opponents:** Pemain lain dalam suatu game, dapat dikendalikan manusia atau komputer.
3. **Dillemas:** Pilihan problematik yang membuat pemain harus menentukan antar dua pilihan, dengan mempertimbangkan efek kedepannya.

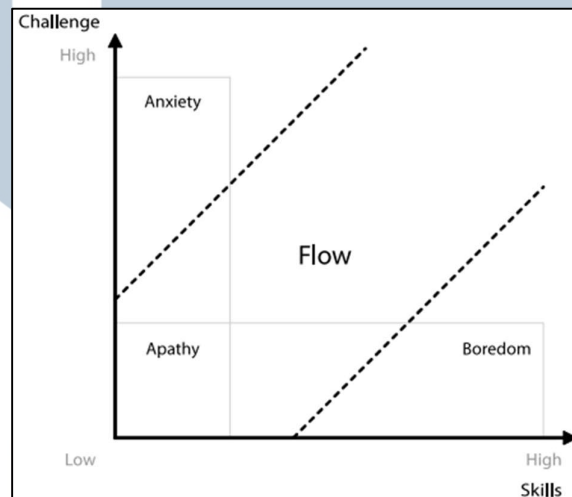
*Boundaries* berkaitan dengan tindakan yang hanya mungkin dilakukan dalam game tetapi akan memiliki konsekuensi yang jauh berbeda di luar batas game. Itu juga bisa berhubungan dengan arena bermain, *skybox*, atau geometri pembatas lainnya di dunia game (Nacke, 2014).

*Outcome* merupakan keluaran dari suatu game setelah selesai dan hasilnya harus tidak pasti untuk meningkatkan daya tarik pemain. Dalam game, biasanya *outcome* dapat dihitung seperti skor dan tidak sebanding seperti hanya satu orang yang dapat menang. Kondisi menang berbeda dari tujuan pemain. Setelah pemain menginvestasikan waktu dan emosi saat bermain game, cukup sulit untuk membuat resolusi yang memuaskan investasi pemain (Nacke, 2014).

## 2.4.2 Dramatic Elements

Cerita / narasi memberikan elemen formal arti. Secara umum narasi membantu pengguna untuk memproses informasi dan membuat banyak hal masuk akal dalam kehidupan nyata (Nacke, 2014).

*Challenge* merupakan rasa pencapaian dan kenikmatan di game dan berkorelasi dengan kemampuan pemain. Saat pemain menyatakan bahwa mereka menikmati game yang menantang, artinya mereka dapat berkompetisi atas tugas tertentu di game, dimana rasa kepuasan mereka itu datang (Nacke, 2014).



Gambar 2.3 Alur *challenge* yang ada pada game menurut Lennart Nacke.

Gambar 2.3 menunjukkan daerah *flow* merupakan kemampuan pemain saat cocok pada tantangan yang ada pada game. Ketidakcocokan karena terlalu sulit akan menyebabkan rasa gelisah dan terlalu mudah akan menyebabkan rasa bosan (Nacke, 2014).

*Play* dapat terjadi dalam beberapa cara yang berbeda. Mungkin tidak terlihat seperti aktivitas yang memiliki arti, tetapi dapat membantu pemain menambah kemampuan. Kemampuan ini yang nantinya dapat berguna diluar game. *Play* dapat

diartikan sebagai *experiment* untuk melampaui batas dalam mencoba hal baru. *Play* lebih merupakan pendekatan untuk suatu kegiatan daripada hanya satu hal yang kaku (Nacke, 2014).

*Premise* menetapkan aksi yang akan terjadi pada game atau sebuah cerita. *Premise* membuat latar maupun metafora didalam dunia game. Premis dramatis mempunyai makna mendalam untuk elemen formal pada game. *Premise* harus memiliki deskripsi mengenai waktu, tempat, karakter utama, tujuan utama, dan tindakan yang mendorong cerita untuk maju (Nacke, 2014).

Karakter merupakan objek yang digunakan untuk menjalankan cerita pada game. Karakter merupakan refleksi dari pemain, mulai dari perasaan, keinginan dan hasrat. Dalam pengembangan karakter, dapat digunakan beberapa faktor yaitu keinginan, kebutuhan, harapan, dan ketakutan (Nacke, 2014).

## 2.4 Metode Pengumpulan Data

### 2.4.1 Random Sampling

Disebut juga sebagai *probability sampling* atau *representative sampling*, merupakan metode pengumpulan data yang setiap anggota dari suatu populasi mempunyai probabilitas bukan nol untuk dimasukan sebagai sampel. Dalam pengambilan sampelnya, metode ini melakukannya secara acak. Probabilitas dapat ditetapkan untuk setiap populasi secara objektif. Karena hal itu, teknik ini membutuhkan populasi yang didefinisikan secara tepat. Teknik ini tidak boleh digunakan untuk populasi yang terlalu umum. Misalnya jika populasi target didefinisikan sebagai mahasiswa, artinya orang yang belajar di perguruan tinggi manapun di dunia adalah elemen dari populasi yang dimaksud (Alvi, 2016).

### 2.4.2 Likert Scale

Likert scale dikembangkan untuk mengukur sikap secara sains (Joshi, dkk., 2015), metode ini sudah divalidasi sejak 1932 (Edmondson, 2005). Likert scale merupakan sekumpulan pernyataan yang ditawarkan terhadap situasi nyata atau hipotesis untuk suatu penelitian. Partisipan akan ditanya untuk menunjukkan tingkat kesetujuan (biasanya mulai dari setuju hingga tidak setuju) terhadap pernyataan yang diberikan dalam skala metrik. Dalam hal ini, seluruh kombinasi pernyataan menunjukkan dimensi spesifik sikap responden terhadap isu karena antar pernyataan saling berhubungan. (Joshi, dkk., 2015).

### 2.4.3 Evaluating Game AI

Tes Turing biasanya merupakan tes di mana seorang interogator di satu ruangan menggunakan terminal komputer untuk memainkan permainan tanya jawab dengan dua subjek yang berlokasi di ruangan lain. Salah satu subjek adalah manusia sedangkan yang lainnya adalah mesin; tugas interogator adalah menentukan subjek mana yang manusia dan subjek mana yang mesin. Jika interogator tidak dapat memberi tahu, maka mesin tersebut harus dianggap cerdas. Sampai saat ini, tidak ada program yang dimasukkan untuk Penghargaan Loebner yang berhasil menipu para juri untuk berpikir bahwa itu adalah manusia - dan prospek yang terjadi dalam waktu dekat bisa masih dibilang jauh (Livingstone, 2012).

Masalah utama pada tes Turing adalah tidak bisa dikompromi: AI pada *game* akan lulus atau gagal. Seperangkat kriteria yang diuraikan perilaku yang dibutuhkan untuk bermain seperti manusia akan lebih bermanfaat. Dengan kriteria seperti itu akan mungkin untuk menguji terhadap berbagai aspek masalah, dan

memungkinkan mereka untuk ditingkatkan di mana AI lemah, serta di tempat pertama. Kriteria yang lebih terperinci juga akan memungkinkan kemungkinan melakukan lebih banyak evaluasi dengan tinjauan ahli. Ini akan mengurangi, tetapi mungkin tidak menghilangkan, kebutuhan untuk evaluasi mahal oleh *gamer* (Livingstone, 2012).

Masalah lain dengan kriteria Turing adalah bahwa itu tidak berlaku untuk AI NPC, di mana secara umum diketahui oleh pemain *a priori* bahwa karakternya dikendalikan oleh komputer. Kriteria yang lebih terperinci menyediakan sarana untuk mengevaluasi perilaku tanpa harus mempertimbangkan apakah pengontrol di balik perilaku itu manusia atau bukan (Livingstone, 2012).

Tabel 2.1 merupakan seperangkat kriteria yang mungkin bermanfaat diterapkan dalam memutuskan apakah atau AI pada *game* tertentu *believable*. Kriteria ini dibuat oleh Livingstone berdasarkan temuan oleh Laird dan Duchi pada 2000 dan berdasarkan percobaan Wetzel untuk menyediakan daftar kecacatan yang lazim ditemukan pada AI dalam *game*. Dalam penyaringan yang dilakukan Livingstone, dicari benang merah dalam kegagalan AI yang diamati untuk menghasilkan serangkaian kriteria yang dapat dikelola. Perilaku AI terbagi menjadi tiga kategori besar, yaitu, *Plan*, *Act*, dan *React*. Kriteria tetap agak terbuka; dan untuk setiap kriteria, perilaku yang tepat yang diperlukan secara alami akan bervariasi sesuai dengan sifat permainan. Kriteria tidak membedakan antara *game* dan simulasi atau perbedaan yang dihasilkan dalam perilaku AI, sehingga pengembang perlu menyadari audiens (Livingstone, 2012).



Tabel 2.1 Kriteria AI yang *believable* menurut Livingstone

	Ketentuan
Plan	(1.1) Mendemonstrasikan strategi atau perencanaan pada tingkat tertentu. (1.2) Dapat mengkoordinasikan aksinya dengan pemain atau AI lain. (1.3) Tidak secara berulang mencoba cara yang sebelumnya telah gagal.
Act	(2.1) Bertindak dengan kecepatan reaksi dan kemampuan manusia.
React	(3.1) Bereaksi terhadap kehadiran dan tindakan pemain. (3.2) Bereaksi terhadap perubahan terhadap lingkungan lokal (3.3) Bereaksi terhadap kehadiran lawan atau teman.
Pengecualian	(1) Mungkin tidak berlaku di mana <i>plot call</i> menyerukan karakter impulsif atau bodoh, atau untuk hewan (2) Mungkin tidak berlaku di mana <i>plot call</i> meminta karakter dengan keunggulan lebih besar atau kemampuan lebih rendah (3) Mungkin tidak berlaku di mana desain game / <i>plot call</i> untuk karakter dengan kesadaran terbatas

*Believability* bukan satu-satunya tujuan AI pada *game*. Tujuan utama, dibantu oleh AI yang *believable*, adalah permainan yang menghibur (Darryl, 2003). AI tidak hanya harus *believable*, tetapi AI juga harus menyediakan pengalaman yang menantang, namun tidak berlebihan (Livingstone, 2012).

#### 2.4.4 Game User Experience Satisfaction Scale

Gaming User Experience Satisfaction Scale (GUESS) merupakan *likert scale* yang digunakan untuk evaluasi *game* yang secara psikometri sudah tervalidasi dan komprehensif yang layak digunakan untuk *playtesting* dan evaluasi *game* (Phan, 2017). Subkala yang ada pada GUESS menurut Phan (2017), antara lain:

1. Usability / Playability (UP): Permainan dapat dimainkan dengan tujuan yang jelas, dengan gangguan kognitif yang minimal ataupun gangguan dari user interface dan kontrol.

2. Narratives (N): Aspek cerita dari suatu game (kejadian, karakter) dan kemampuannya untuk menarik perhatian pemain dan membentuk emosi pemain.
3. Play Engrossment (PE): Kemampuan game untuk mempertahankan ketertarikan pemain.
4. Enjoyment (E): Kenikmatan yang dirasakan pemain sebagai hasil memainkan game terkait.
5. Creative Freedom (CF): Sejauh mana game dapat menumbuhkan kreativitas peserta dan rasa ingin tahun pemain, dan memungkinkan pemain untuk secara bebas mengekspresikan individualitasnya saat memainkan game terkait.
6. Audio Aesthetics (AA): Perbedaan aspek auditori dalam game dan bagaimana aspek tersebut meningkatkan pengalaman pemain.
7. Personal Gratification (PG): Aspek motivasi yang meningkatkan rasa pencapaian pemain dan kemauan untuk melanjutkan game.
8. Social Connectivity (SC): Tingkat fasilitasi hubungan sosial antar pemain.
9. Visual Aesthetics (VA): Grafis permainan dan bagaimana pemain tertarik dengan grafis tersebut.

Pertanyaan yang ada pada kuisisioner GUESS yang diindikasikan dengan kode sub-skala yang dimaksud adalah sebagai berikut:

Tabel 2.2 Tabel Pertanyaan Kuisisioner GUESS oleh Phan (Phan, 2012)

Kode Variabel	Pertanyaan
UP01	Saya pikir mudah untuk mempelajari cara bermain <i>game</i> terkait.

Tabel 2.2 Tabel Pertanyaan Kuisisioner GUESS oleh Phan (Phan, 2012)

UP02	Saya mudah menemukan kontrol <i>game</i> .
UP03	Saya selalu tahu cara mencapai sasaran / tujuan saya dalam <i>game</i> .
UP04	Saya mudah melakukan navigasi pada antarmuka <i>game</i> .
UP05	Saya tidak perlu melalui <i>tutorial</i> yang panjang atau membaca manual untuk memainkan <i>game</i> .
UP06	Saya menemukan menu <i>game</i> ini ramah pengguna.
UP07	Saya merasa <i>game</i> ini melatih saya dengan baik di semua kontrol <i>game</i> .
UP08	Saya selalu tahu tujuan saya berikutnya ketika saya menyelesaikan suatu <i>event</i> dalam <i>game</i> .
UP09	Saya merasa <i>game</i> memberi saya informasi yang diperlukan untuk mencapai tujuan dalam <i>game</i> .
UP10	Saya pikir informasi yang disediakan dalam <i>game</i> (misalnya pesan pada layar) jelas
UP11	Saya merasa sangat percaya diri saat bermain <i>game</i> .
N01	Saya pikir karakter dalam <i>game</i> berkembang dengan baik.
N02	Saya terpicat dengan cerita permainan dari awal.
N03	Saya menikmati fantasi atau cerita yang disediakan oleh <i>game</i> .
N04	Saya dapat mengidentifikasi dengan karakter dalam <i>game</i> .
N05	Saya terharu secara emosional oleh berbagai peristiwa dalam <i>game</i> .
N06	Saya sangat tertarik melihat bagaimana acara dalam <i>game</i> akan berkembang.
N07	Saya dapat dengan jelas memahami cerita <i>game</i> .
PE01	Saya merasa terpisah dari dunia luar saat bermain <i>game</i> .
PE02	Saya tidak peduli untuk memeriksa peristiwa yang terjadi di dunia nyata selama <i>game</i> .
PE03	Saya tidak tahu bahwa saya lelah saat bermain <i>game</i> .
PE04	Terkadang saya lupa waktu saat bermain <i>game</i> .
PE05	Untuk sementara saya melupakan kekhawatiran sehari-hari saya saat bermain <i>game</i> .
PE06	Saya cenderung menghabiskan lebih banyak waktu bermain <i>game</i> daripada yang telah saya rencanakan.
PE07	Saya dapat memblokir sebagian besar gangguan lain saat bermain <i>game</i> .
PE08	Setiap kali saya berhenti bermain <i>game</i> , saya tidak sabar untuk mulai memainkannya lagi.
E01	Saya pikir permainan ini menyenangkan.
E02	Saya menikmati bermain <i>game</i> .
E03	Saya merasa bosan saat bermain <i>game</i> .
E04	Saya cenderung merekomendasikan <i>game</i> ini kepada orang lain.
E05	Jika diberi kesempatan, saya ingin memainkan <i>game</i> ini lagi.
CF01	Saya merasa permainan memungkinkan saya untuk menjadi imajinatif.
CF02	Saya merasa kreatif saat bermain <i>game</i> .

Tabel 2.2 Tabel Pertanyaan Kuisisioner GUESS oleh Phan (Phan, 2012) (Lanjutan)

CF03	Saya merasa permainan memberi saya cukup kebebasan untuk bertindak sesuai keinginan saya.
CF04	Saya merasa <i>game</i> ini memungkinkan saya untuk mengekspresikan diri.
CF05	Saya merasa saya dapat menjelajahi hal-hal dalam <i>game</i> .
CF06	Saya merasa keingintahuan saya terangsang sebagai hasil dari bermain <i>game</i> .
CF07	Saya pikir <i>game</i> ini unik atau asli.
AA01	Saya menikmati efek suara dalam <i>game</i> .
AA02	Saya menikmati musik dalam <i>game</i> .
AA03	Saya merasakan audio <i>game</i> (misalnya efek suara, musik) meningkatkan pengalaman bermain <i>game</i> saya.
AA04	Saya pikir audio <i>game</i> sesuai dengan mood atau gaya permainan.
PG01	Saya dalam ketegangan tentang apakah saya akan berhasil dalam <i>game</i> .
PG02	Saya merasa sukses ketika saya mengatasi hambatan dalam <i>game</i> .
PG03	Saya ingin melakukan hal terbaik selama <i>game</i> berlangsung.
PG04	Saya sangat fokus pada kinerja saya sendiri saat bermain <i>game</i> .
PG05	Saya merasa <i>game</i> terus memotivasi saya untuk melangkah lebih jauh ke tahap atau tingkat berikutnya.
PG06	Saya menemukan keterampilan saya secara bertahap meningkat melalui cara mengatasi tantangan dalam <i>game</i> .
SA01	Saya menemukan <i>game</i> mendukung interaksi sosial (misalnya obrolan) antara pemain.
SA02	Saya suka memainkan <i>game</i> ini dengan pemain lain.
SA03	Saya bisa memilih untuk bermain <i>game</i> dengan pemain lain.
SA04	Saya menikmati interaksi sosial dalam permainan.
VA01	Saya menikmati grafis <i>game</i> .
VA02	Saya pikir grafik <i>game</i> sesuai dengan suasana hati atau gaya <i>game</i> .
VA03	Saya pikir <i>game</i> ini menarik secara visual.