

BAB II

LANDASAN TEORI

II.1. Game User Research

Game user research (GUR) merupakan suatu kegiatan untuk meneliti pengguna, yang dalam hal ini pengguna merupakan pemain *video game*. Hal ini dilakukan dengan tujuan untuk mengetahui karakteristik pemain dan mengoptimalkan pengalaman yang akan didapat pemain saat memainkan *video game* atau permainan [6].

Dalam pelaksanaan GUR, terdapat tahapan-tahapan yang harus dilakukan dalam penelitian ini, diantaranya adalah sebagai berikut [6].

II.1.1. Interview

Interview atau wawancara merupakan percakapan terstruktur yang dilakukan dengan pemain yang berpotensi. Percakapan yang dilakukan dalam wawancara berisikan pertanyaan-pertanyaan peneliti terhadap pemain, yang nantinya jawaban pemain akan membuat peneliti memahami preferensi dan sikap pengguna [6]. Dalam wawancara terdapat struktur dasar sebagai pedoman untuk melakukan wawancara. Struktur dalam melakukan wawancara adalah sebagai berikut [9]:

1. Introduction

Introduction atau perkenalan harus dilakukan terlebih dahulu oleh pewawancara dan menjelaskan maksud dan tujuan dari wawancara yang akan dilakukan ke pada pihak yang akan diwawancarai, serta meminta izin secara eksplisit untuk merekam proses wawancara.

2. Body (Content)

Body atau isi merupakan langkah yang dilakukan setelah melakukan perkenalan. Pewawancara membuka percakapan dengan mengarahkan pembicaraan terhadap

topik utama yang akan dituju. Dalam tahapan ini jawaban dari pihak yang diwawancarai merupakan opini.

3. Conclusion

Conclusion atau kesimpulan merupakan tahapan akhir dalam melakukan wawancara. Setelah pewawancara sudah mendapatkan informasi yang dibutuhkan wawancara diakhiri dengan cara yang baik.

II.1.2. Initial Experience Playtest

Initial experience playtest merupakan teknik survei yang menilai sikap dan persepsi yang dimiliki pemain mengenai permainan yang dimainkan selama 2 jam pertama [6].

II.1.3. Focus Group

Focus group merupakan teknik penelitian yang bersifat kualitatif, di mana sekelompok orang dikumpulkan pada suatu tempat dan ditanyakan mengenai pendapat mereka mengenai *game* yang sudah dimainkan. Pendekatan ini sangat baik untuk menghasilkan ide-ide baru mengenai pengembangan yang nanti dapat diperhitungkan untuk dimasukkan ke dalam *game* [6].

II.2. Level Design

Level design merupakan cakupan pada pengembangan *game* untuk membuat rintangan yang akan dihadapi oleh pemain dalam memainkan *game* [5]. Dalam tujuan pembuatan *level design* berbeda dalam pembentukan tujuan dalam *design* pada umumnya. Dalam pembentukan *level design* dibentuk tujuan bukan untuk memudahkan pengguna, tetapi dibuat tujuan yaitu untuk membuat pengguna merasa tertantang dalam memainkan permainan [10]. Dalam tahapan membuat *level design*, terdapat 8 tahapan yang harus dilakukan, yaitu [11]:

1. Brain Storming

Pada tahap ini, dilakukan proses pembuatan ide mengenai *level design* seperti apa yang akan dibuat.

2. Drafting

Pada tahap ini, dilakukan proses pembuatan *prototype* dari *level design* yang akan dibuat dengan cara “paper prototyping”, yaitu pembuatan *level design* yang digambarkan di sebuah kertas.

3. Commit Level Design

Pada tahap ini, merupakan tahap di mana ditentukan *level design* yang akan direalisasikan atau diimplementasi.

4. Initial Construction

Pada tahap ini, dilakukan realisasi atau implementasi terhadap *level design* yang sudah ditetapkan.

5. Function Testing

Pada tahap ini, dilakukan *testing* terhadap *level design* yang sudah dibuat, untuk mencari tahu apakah komponen yang sudah terealisasi sudah berjalan dengan baik atau belum.

6. Detailing

Pada tahap ini, dilakukan perincian pada *level design* yang sudah dibuat (penambahan detil pada *asset game*). Hal ini dilakukan agar menambah *experience* yang akan didapat oleh pemain.

7. Level Design Testing

Pada tahap ini, dilakukan *testing* terhadap *level design* yang sudah dibuat berdasarkan tahap-tahap sebelumnya.

8. Fix and Ship

Pada tahap ini, dilakukan perbaikan terhadap *level design* jika ditemukan masalah pada tahap “Level Design Testing”.

Dalam penelitian ini terdapat beberapa teori pendukung yang digunakan dalam pembuatan *level design game* pada penelitian ini.

II.2.1. Level Design Document

Level design document merupakan sebuah dokumen berisikan tentang acuan dari *level design* yang akan diimplementasikan. Hal ini dilakukan agar hasil implementasi tidak jauh dari hasil perancangan *level design* yang sudah dibuat [10].

II.2.2. Player Centered Level Design

Player centered level design merupakan pendekatan untuk membuat *level design* dengan mendekati pendapat pemain potensial dalam melakukan pembuatan atau pengembangan *level design* [12]. Hal ini dilakukan untuk mengetahui karakteristik pemain dalam bermain *game* agar *level design* yang dibuat tidak terlalu sulit atau mudah [12].

II.2.3. Multiplayer Level Design Consideration

Pembuatan *level design game* yang diteliti pada penelitian ini, merupakan *game* yang memiliki genre *multiplayer* atau permainan yang dapat dimainkan oleh lebih dari satu orang. Oleh karena itu, terdapat beberapa hal yang harus diperhatikan dalam pembuatan *level design* pada penelitian ini, yaitu [10]:

1. Human Element

Dalam pembuatan *level design game* bergenre *multiplayer*, factor manusia harus diperhitungkan, karena tingkah laku pemain dan jumlah pemain yang terdapat pada permainan berpengaruh terhadap tingkat kesulitan terhadap *level design* yang dibuat.

2. Clearly Defined Competitive Challenges

Dalam pembuatan *level design* pada *game* bergenre *multiplayer*, tujuan yang diberikan kepada pemain harus terdefiniskan dengan jelas. Hal ini ditujukan untuk memberikan pemain motivasi dan kesan kompetitif terhadap permainan (jika terdapat unsur kompetitif), seperti dalam pemilihan mode permainan, apakah mode tersebut merupakan mode kooperatif atau kompetitif.

II.3. Player Profiling

Player modelling adalah proses pengumpulan data pemain untuk membuat kerangka terhadap perilaku mereka (partisipan). *Player profiling* merupakan subkategori pada *player modelling* yang berfokus pada informasi yang tidak spesifik terhadap *game*. Contoh data yang biasanya dimasukkan dalam *player profiling* adalah kepribadian, latar belakang budaya, jenis kelamin, dan usia [13]. Hal ini dilakukan dengan tujuan untuk membuat tingkat kesulitan pada *level design* menjadi lebih dinamis, menambah konten yang dapat ditambahkan pada pengembangan selanjutnya, meningkatkan konten marketing yang terdapat dalam *game* [13]. Dalam pembuatan *player profiling* pada penelitian ini, peneliti menggunakan *tools* “Play-persona”. Dalam penelitian ini terdapat teori pendukung dalam melakukan pembuatan *player profiling*.

II.3.1. Persona

Persona merupakan salah satu *tools* yang digunakan untuk membuat gambaran karakteristik dari para pengguna sebuah aplikasi dan sebagainya, yang nantinya akan digunakan sebagai bahan pertimbangan dalam pada pengembangan produk [14]. Hal ini dilakukan untuk memudahkan dalam memanfaatkan data dari pengguna dalam pengembangan selanjutnya pada produk, memudahkan dalam menentukan sebuah fitur karena gambaran dari persona yang sudah dibuat, memudahkan komunikasi antara pengembang produk dan *designer*, menghindari

pembuatan *design* atas dasar pengada-ngadaan, membuat *designer* fokus terhadap masalah yang dihadapi dan efisien dalam pembuatan *design* [12].

II.3.2. Play-persona

Play-persona merupakan pengembangan dari persona yang sudah dibuat oleh Alan Cooper yang didalamnya terdapat 3 faktor dalam pembentukan play-persona [12]. Play-persona membantu *designer* dalam melakukan *player modeling* pemain saat dan sesudah memainkan permainan [12]. Play-persona mendeskripsikan perilaku pemain secara procedural [12]. Data yang diberikan merupakan data yang didapat dari percobaan terhadap *game* secara langsung [12]. Dalam pembuatan play-persona terdapat 6 proses yang harus dilakukan, yaitu [12]:

1. Gameplay Analysis and High-level Behaviors

Pada tahap ini dilakukan pemisahan terlebih dahulu terhadap mekanik *game* yang akan diuji coba untuk pengambilan data pemain dan membuat kemungkinan dari tindakan yang dilakukan oleh pemain. *Low-level mechanics* merupakan aktifitas yang dapat dilakukan oleh pemain pada *game*, *high-level behavior* merupakan asumsi dari tingkah laku yang akan dilakukan oleh pemain dalam sesi permainan.

2. Play-persona Matrix

Pada tahap ini akan dibuat kelompok berdasarkan kemungkinan-kemungkinan dari perilaku pemain terhadap *game*.

3. Selecting Cast Play-persona

Pada tahap ini akan dipilih prioritas perilaku berdasarkan hasil *test* berdasarkan “Play-persona Matrix” yang sudah terbentuk. Hal ini akan menjadi pertimbangan dalam pengembangan *game* selanjutnya.

4. Associate Affordances with Behaviours

Pada tahap ini akan dibuat sebuah asumsi yang berupa solusi dari play-persona yang sudah terbentuk pada tahap “Selecting Cast Play-persona”.

5. Weave and Orchestrate Play-personas

Pada tahap akan dibuat sebuah usulan yang dapat memotivasi pemain dalam sesi permainan.

6. Play-persona as Tools

Pada tahap ini dilakukan evaluasi terhadap play-persona yang sudah dibuat. Hal ini dilakukan untuk membuat keputusan yang berhubungan dengan pengembangan *game* selanjutnya.

II.4. Level Design Testing

Level design testing merupakan tahapan yang dilakukan untuk melakukan uji coba terhadap *level design* yang sudah dibuat. Pada penelitian ini digunakan metode *focus group playtesting* dalam melakukan uji coba terhadap *level design* yang sudah dibuat. Digunakannya metode *focus group playtesting* dikarenakan kecocokan terhadap masalah yang didapat dalam genre *game* pada penelitian ini, yaitu 2D platformer [7].

Focus group playtesting merupakan metode uji coba yang dilakukan terhadap lebih dari satu orang, berada di ruangan yang sama dan melakukan wawancara terhadap mereka (partisipan) mengenai pandangan mereka setelah mencoba permainan yang dimainkan [6].

II.5. Video Game

Video game merupakan permainan elektronik yang menggunakan media komputer, mesin konsol, ataupun mobile. *Video game* melibatkan interaksi antar tampilan pada *video game* untuk menghasilkan reaksi pada suatu perangkat kepada pengguna. Perangkat yang digunakan untuk bermain *game* biasa disebut dengan “platform”. Pada *game* dibutuhkan suatu *input device* yang biasa disebut *game controller*. Adapun elemen-elemen yang pada umumnya terdapat pada *video game* yaitu: *game title*, *title screen*, *credits*, *intro*, *control panel*, *user interface*, *help screen*, *mouse cursor* atau *pointer*, *music* dan *sound effect*, *art*, *storyline*,

playability, levels, demo mode, exit screen, documentations, copyrights, setup program. Elemen-elemen tersebut merupakan komponen dasar pada *game*.

Video *game* juga memiliki banyak jenis ataupun genre yang beragam. Jenis *game* tersebut dilihat dari bagaimana cara memainkannya sampai aturan-aturan dalam bermain *game* tersebut. Jenis-jenis *game* ada berbagai macam diantaranya sebagai berikut: *Maze, boardd, card, battle, quiz, puzzle, shoot them up, sice scroller, fighting, racing, simulasi, strategy, first person shooter, third person shooter, role paying, adventure, full motion video, educational, sports* [15].

II.6. 2D Platformer Game

Pada sebuah *game* terdapat beberapa genre yang terkait pada *game* tersebut, salah satunya adalah genre 2D platformer. Genre ini merupakan gabungan dari 2 genre yaitu: 2 dimensi (2D) dan Platformer. Ada pun teori pendukung mengenai kedua genre tersebut.

II.6.1. Dua Dimensi (2D)

Video game dengan genre 2D merupakan *video game* yang melibatkan control pergerakan pada mekanik dalam *game* hanya dapat bergerak secara vertikal dan horizontal [16].

II.6.2. Platformer

Platformer merupakan genre yang dimana pemain melakukan pergerakan dengan memanfaatkan pijakan (platform). Hal ini berlaku pada *game* 3 dimensi maupun 2 dimensi [17].

II.7. Multiplayer Game

Multiplayer pada *game* merupakan genre yang diberikan terhadap *game* yang dapat dimainkan oleh lebih dari satu pemain dalam sesi permainannya. *Game* dengan genre ini melibatkan hubungan atau interaksi antara pemain dengan pemain lainnya sehingga menimbulkan *feedback* percakapan yang berhubungan

dengan *game* yang dimainkan, baik dalam bentuk kooperatif ataupun kompetitif [10].

II.8. Player Gameplay Experience Mapping

Player gameplay experience mapping merupakan metode yang digunakan untuk menggambarkan pengalaman pemain dalam memainkan permainan dalam sesi permainan. *Player gameplay experience mapping* dibuat untuk memaparkan aktifitas pemain dalam sesi permainan beserta *pain point* yang dirasakan oleh pemain [6].

II.9. Unified Modeling Language

Unified Modeling Language (UML) merupakan standar bahasa modeling dalam dunia pengembangan *software*. Sulit untuk menggambarkan desain sistem yang memiliki cakupan cukup besar. Dalam mendesain sistem, *designer* harus menggambarkan hal-hal penting yang terdapat di dalam sistemnya seperti: mengelola hal yang kompleks, membuat model dapat membantu dalam melihat keseluruhan sistem yang dibuat, membantu untuk fokus terhadap suatu aspek, merekam apa yang harus dikerjakan, membuat dokumentasi terhadap sistem, dan memberikan informasi terhadap aspek penting dalam sistem yang akan dibuat. Untuk membuat sistem yang efektif, desainer memerlukan bahasa dan model yang dapat dijelaskan. *Modeling language* dapat menggambarkan *pseudo-code*, *actual code*, gambar, diagram, atau deskripsi yang cukup panjang. UML memiliki enam kelebihan dibandingkan bahasa modeling lainnya yaitu: menggunakan bahasa formal yang di setiap elemennya dapat secara kuat menjelaskan arti dari sebuah sistem atau model yang akan dibuat, dapat menjelaskan suatu elemen secara singkat dan sederhana, dapat menjelaskan hal yang luas dengan seefektif mungkin, memiliki skala yang jelas, dibuat berdasarkan pelajaran, dan memiliki standar yang sudah diakui oleh dunia [18].

Pada penelitian ini, digunakan 4 macam diagram untuk membuat model sistem berorientasi objek dalam dokumentasi UML. Berikut merupakan teori pendukung mengenai keempat diagram yang digunakan [18].

II.9.1. Use Case Diagram

Use case adalah situasi di mana sistem yang dibuat dapat berguna untuk memenuhi kebutuhan yang dibutuhkan oleh pengguna. *Use case* menggambarkan setiap fungsional yang terdapat dalam sistem. Selain itu, *Use case* juga merupakan langkah tepat untuk memulai pembangunan sistem, desain, testing, dan dokumentasi berorientasi objek. Maka dari itu *Use case* dibuat untuk memenuhi dan menjelaskan kebutuhan fungsional pada sistem yang akan dibuat. *Use case* merupakan bentuk dari sebuah pekerjaan tertentu misalnya masuk ke dalam *game*, menembak, berbelok, dan semacamnya. Selain itu juga terdapat entitas aktor yang didefinisikan sebagai manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu yang berhubungan dengan sistem yang dibuat. Adapun komponen-komponen yang terdapat dalam *Use case* yaitu [18]:

1. Actor

Actor pada UML dapat berupa manusia, proses, atau sesuatu dari luar sistem yang berinteraksi dengan sistem, sub sistem, ataupun *class* yang dibangun. Setiap aktor memiliki akses pada satu atau lebih dari satu *use case*. Aktor merupakan entitas yang berkaitan dengan sistem namun bukan bagian dari sistem itu sendiri. Aktor digambarkan dengan *stick person* kecil dengan nama di bawahnya [18].

2. Use Case

Use case merupakan gambaran umum dari fungsi atau proses utama yang menggambarkan tentang setiap perilaku yang dapat dilakukan oleh sistem. Penggambaran *use case* digambarkan sebagai oval yang terdapat di dalam kotak yang merepresentasikan sistem yang dibangun. *Use case* harus memiliki kriteria yang jelas tentang keberhasilan atau kegagalan suatu *use case* bila dijalankan. Pengembang, *tester*, *technical writer*, dan pengguna harus dapat mengetahui secara eksplisit ketika sistem dapat memenuhi kriteria sistem atau tidak [18].

3. Communication Flow

Setelah membuat *actor* dan *use case* maka selanjutnya akan dibuat *communication flow* atau jalur komunikasi antara *actor* dengan *use case* ataupun *use case* dengan *use case*. Tujuan dari penggambaran jalur komunikasi adalah menunjukkan bahwa *actor* memiliki keterlibatan pada suatu *use case* [18].

4. System Scope

System scope atau lingkup sistem merupakan batas pemisah secara implisit antara *actor* yang berada di luar sistem dan *use case* yang berada di dalam sistem yang menandakan lingkup sistem, UML memberikan notasi untuk *designer* agar batas pemisah antara *actor* dan *use case* terlihat lebih jelas. Menggambar kotak untuk menandakan lingkup sistem dengan *actor* berada di luar kotak dan *use case* berada di dalam kotak. Kotak tersebut akan merepresentasikan lingkup sistem yang akan dibuat dengan pemberian nama di bagian atas kotak tersebut [18].

II.9.2. Activity Diagram

Activity diagram merupakan gambaran mengenai bagaimana sistem dapat mencapai tujuan yang sudah digambarkan dan dijelaskan pada skenario *use case*. Gambar yang digambarkan dalam *activity diagram* yaitu rantai proses yang merepresentasikan proses yang terjadi dalam sistem. *Activity diagram* memiliki struktur yang mirip dengan *flowchart* pada perancangan terstruktur. *Activity diagram* dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram* [18].

II.9.3. Sequence Diagram

Sequence diagram merupakan alat bantu dalam menunjukkan informasi yang tepat akan perilaku pada sebuah skenario. Komponen utama dari *sequence diagram* adalah objek yang digambarkan dengan kotak persegi empat atau bulat. Manfaat dari penggambaran *sequence diagram* adalah memberikan gambaran secara lebih detail dari setiap *use case* diagram yang telah dibuat sebelumnya [18].

II.9.4. Class Diagram

Class diagram merupakan diagram yang dianggap populer untuk menggambarkan struktur sebuah *class*. *Class diagram* menggambarkan relasi antar *class* yang dibuat oleh pengembang dan struktur pada suatu *class* tersebut agar mencapai tujuan [18].

II.10. Level Designer

Level designer merupakan suatu bidang tersendiri pada *game development* yang bertanggung jawab dalam pembentukan *environment* yang dirasakan oleh pemain dan membuat arsitektur permainan yang menyenangkan dalam sesi permainan (*gameplay*) berdasarkan aturan-aturan yang ada [19].