

BAB 2 LANDASAN KEPUSTAKAAN

Menurut Wibowo, et al., (2013), dalam jurnalnya mengenai teknologi *Natural User Interface* menggunakan *Kinect* yang digunakan sebagai pemicu kerja perangkat keras berbasis *fuzzy interface system* menjelaskan bahwa HCI (*Human Computer Interaction*) yaitu ilmu yang mempelajari bagaimana membuat interaksi antara manusia dan komputer dapat terjadi seramah dan seefisien mungkin. Salah satu penerapan prinsip dari HCI adalah teknologi *Natural User Interface* (NUI). NUI merupakan payung dari beberapa teknologi seperti *speech recognition*, *multitouch* dan *kinectic interface* seperti *Kinect*. NUI dimaksudkan untuk menghilangkan hambatan mental dan fisik pengguna. *Kinect* adalah alat yang menerapkan NUI, dengan menangkap data citra berwarna, citra kedalaman, *gesture*, jarak, posisi dan ketinggian tubuh pengguna. Data yang berasal dari *Kinect* diubah menjadi suatu perintah yang dapat dimengerti oleh perangkat keras. Data tersebut dikombinasikan dengan *Fuzzy Interface System* metode TSK untuk mendapatkan hasil yang semaksimal mungkin dalam proses memicu kerja perangkat keras sebagai simulasi sistem *smart house*, sehingga prinsip dari HCI yakni membuat interaksi antara manusia dan komputer terjadi seramah dan seefisien mungkin dapat tercapai.

Runiawati, et al., (2012) dalam skripsinya mengenai perancangan dan implementasi *game* bergenre *fighting* berbasis *augmented reality* menggunakan *motion kinect* mengatakan bahwa untuk pembuatan *game* 3D, salah satunya yaitu menggunakan Unity3D. Salah satu keunggulan dari Unity3D adalah mempermudah *game developer* dalam pembuatan *Graphical User Interface* (GUI), terutama untuk pembuatan *game* 3D. GUI berperan sebagai *graphic interface* untuk dapat membantu *user* lebih mengerti maksud dari suatu aplikasi dan tampilan dari sebuah aplikasi terlihat lebih menarik. Pada pertengahan tahun 2009, *Microsoft* mengumumkan dimulai Proyek Natal berupa *Natural User Interface* (NUI) yang sekarang dikenal sebagai *kinect*. *Kinect* merupakan alat yang digunakan dalam pengembangan aplikasi yang mengizinkan pemain untuk bergerak bebas berinteraksi dalam *game*.

2.1 Lalu Lintas dan Penyeberangan Jalan

Undang-undang dan peraturan-peraturan lalu lintas telah menjelaskan tentang etika serta norma-norma sopan santun antara sesama pemakai jalan. Sesuai Undang-Undang No: 38 Tahun 2004 tentang Jalan tidak diatur secara jelas perihal fasilitas untuk pejalan kaki. Dalam Peraturan Pemerintah No: 34 Tahun 2006 tentang Jalan Pasal 22, diuraikan secara umum tentang perlunya jalan dilengkapi dengan perlengkapan jalan dan pada Paragraf 1 tentang Ruang Manfaat Jalan (Rumaja) pasal 34 butir 3 (tiga) dan 4 (empat) disinggung tentang trotoar (BPJN 3 Palembang, 2011).

Dalam Undang-Undang No:22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan (LLAJ) (Republik Indonesia, 2009) dinyatakan dengan tegas pada paragraf 2

(dua) tentang Penggunaan dan Perlengkapan Jalan pada pasal 25 dan 26 yang tertulis sebagai berikut:

Pasal 25 yakni setiap jalan yang digunakan untuk lalu lintas umum WAJIB dilengkapi dengan perlengkapan jalan berupa:

1. Rambu lalu lintas;
2. Marka jalan;
3. Alat pemberi isyarat lalu lintas;
4. Alat penerangan jalan;
5. Alat pengendali dan pengaman pengguna jalan;
6. Alat pengawasan dan pengamanan jalan;
7. Fasilitas untuk sepeda, pejalan kaki, dan penyandang cacat; dan
8. Fasilitas pendukung kegiatan lalu lintas dan angkutan jalan yang berada di jalan dan diluar badan jalan.

Pasal 26 yaitu penyediaan perlengkapan jalan diselenggarakan oleh:

1. Pemerintah untuk Jalan Nasional;
2. Pemerintah Provinsi untuk jalan Provinsi;
3. Pemerintah Kabupaten/Kota untuk jalan Kabupaten/Kota dan jalan Desa; atau
4. Badan Usaha Jalan Tol untuk jalan Tol.

Hal ini menjelaskan bahwa adanya fasilitas penyeberangan jalan yang telah disediakan oleh pemerintah Indonesia ditujukan agar masyarakat akan lebih berhati-hati dalam menggunakan jalan dan memperhatikan etika dalam menyeberang jalan yang benar sehingga keselamatan pengguna jalan tetap terjamin. Adapun cara menyeberang jalan yang baik dan benar (Widiyono S.H., 2017) adalah:

1. Jika termasuk orang berbadan kecil atau anak usia dini, serta tidak memiliki keberanian dalam menyeberang jalan, ada kalanya untuk tidak menyeberang sendirian dan meminta untuk ditemani atau meminta bantuan orang lain dalam menyeberang jalan.
2. Memperhatikan keadaan sekitar dan memastikan apakah terdapat *zebra cross* atau marka jalan, lampu penyeberang jalan, dan lampu isyarat lalu lintas, serta jembatan penyebrangan. Jika ada, diharapkan untuk menggunakannya dan tidak mengambil jalan pintas dengan menyeberangi jalan tidak pada tempatnya.
3. Melihat/menengok ke arah kanan atau arah datangnya kendaraan lalu ke kiri dan ke kanan lagi untuk memastikan tidak ada kendaraan sebelum bergerak untuk menyeberangi jalan.
4. memperhitungkan waktu dengan mengira-ngira berapa lama kecepatan mobil yang datang serta insting untuk perhitungkan kecepatan jalan atau lari saat akan menyeberang jalan.
5. Memberi sinyal bagi pengendara dengan menggunakan tangan agar pengendara segera memperlambat laju kendaraannya.

2.2 Game

Menurut Schell (2008) di mana ia mengutip dari Elliot dan Brian, bahwa permainan merupakan kegiatan sukarela yang secara garis besar terjadi untuk membangun sebuah sistem kontrol, di mana terdapat batasan melalui aturan-aturan tertentu untuk menghasilkan hasil yang tidak akan menang atau kalah. Selain itu permainan adalah struktur interaktif yang memiliki nilai internal yang membutuhkan pemain bertahan hingga tercapai sebuah tujuan, ada sepuluh karakteristik yang harus dimiliki dalam sebuah *game*, antara lain:

1. Permainan ini dimainkan dengan kehendak sendiri
2. Permainan memiliki tujuan
3. Permainan memiliki konflik / masalah
4. Permainan memiliki aturan
5. Permainan dapat menghasilkan kondisi / persyaratan menang atau kalah
6. *Game* interaktif
7. Permainan memiliki tantangan
8. Permainan dapat menciptakan nilai internal di dalam permainan itu sendiri
9. Permainan yang melibatkan pemain
10. *Game* adalah sistem peraturan yang tertutup

2.2.1 Pengertian *Game* Edukasi



Gambar 2.1 Metode Pembelajaran Göbel et al. (2009)

Arti *game* edukasi menurut Bjørner & Hansen (2010) adalah sebuah permainan yang dibuat dan dirancang untuk merangsang dan memaksimalkan daya ingat dan pikir, serta meningkatkan konsentrasi untuk memecahkan masalah yang ada. *Game* edukasi adalah alat belajar dengan 3 jenis gaya belajar yakni gaya belajar visual, auditori dan kinestetik. Gaya belajar visual yakni *game* dengan melibatkan gambar atau animasi. Gaya belajar auditori yakni *game* berisi suara. Sedangkan gaya belajar kinestetik yakni *game* menggunakan *mouse*, *keyboard* atau juga *touch* (Eridani & Santosa, 2014).

Metode untuk mengharmonisasikan antara menyampaikan cerita/mendongeng dalam pembelajaran dan *game* (Göbel et al., 2009) adalah:

- Mendongeng yaitu dengan menggunakan deskripsi ringan dari cerita dalam menjelaskan pengetahuan atau informasi yang diperlukan dalam permainan.
- Belajar yaitu dengan menekankan pengetahuan dan informasi yang ingin disampaikan, sehingga menjadi pembelajaran yang efektif.
- *Gaming* yaitu dengan membuat interaksi yang menarik dalam permainan, sehingga permainan bisa mencapai tujuannya.

2.2.2 Unsur Permainan

Ada empat elemen dasar dalam permainan (Adrianto, et al., 2015), yaitu:

1. Mekanika

Mekanika yakni prosedur dan aturan dalam permainan serta menjelaskan tujuan dari permainan. Bagaimana pengguna dapat mencoba untuk mencapai sesuatu atau menyebarkannya, serta apa yang terjadi ketika seorang pemain mencoba untuk mencapai sesuatu tersebut.

2. Cerita

Cerita yakni menjelaskan apa yang terjadi di dalam permainan. Dan adanya estetika dapat membantu narator dalam menceritakan cerita.

3. Estetika

Estetika yakni menjelaskan bagaimana menampilkan gambar dan suara. Estetika sangat penting dalam permainan, karena merupakan unsur yang secara langsung berinteraksi dengan para pemain.

4. Teknologi

Teknologi di mana dalam permainan tidak selalu harus maju, yang paling penting dari teknologi adalah untuk memenuhi kebutuhan *game* yang dibuat.

2.2.3 Elemen Formal *Game*

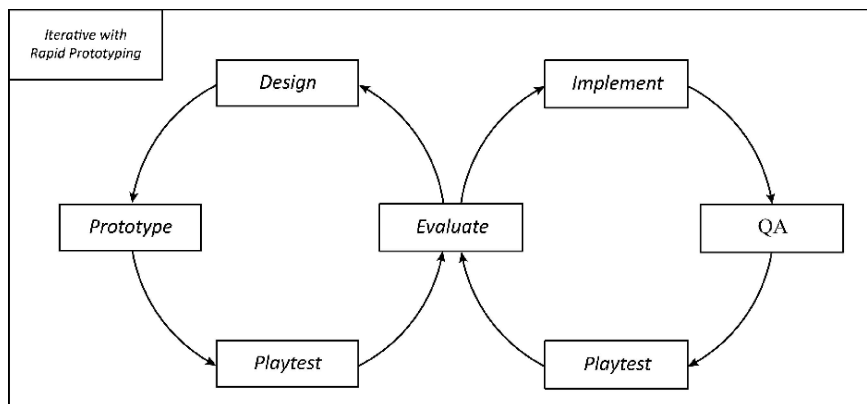
Elemen formal *game* terdiri dari 10 (sepuluh) elemen (Schreiber & Sohn, 2009), diantaranya:

1. *Player* yaitu berapa banyak pemain yang ada di dalam *game* tersebut.
2. *Goals* yaitu tujuan dibuatnya *game* dan apa yang harus dilakukan oleh pemain di dalam *game*.
3. *Rules* yaitu aturan-aturan yang ada dan ditetapkan pada suatu permainan. *Rules* dibagi menjadi tiga bagian, antara lain *setup*, *progression of play*, dan *resolution*.
 - a. *Setup* yaitu hal yang pertama kali dilakukan saat awal permainan.
 - b. *progression of play* yaitu apa yang terjadi selama permainan berlangsung, dan
 - c. *resolution* yaitu kondisi menang dan kalah dalam permainan.
4. *Resource and Resources Management* yaitu hal-hal yang berada di bawah kontrol pemain, seperti sumber daya apa yang dikendalikan pemain, serta bagaimana sumber daya tersebut dimanipulasi selama permainan berlangsung.
5. *Game state* yaitu kumpulan dari *state* yang ada dalam *game*. suatu hal yang mirip sumber daya yang tidak dimiliki oleh satu pemain tertentu, tetapi masih termasuk bagian dari *game* tersebut.
6. *Information* yaitu informasi apa saja yang tersedia dalam *game*.
7. *Sequencing* yaitu urutan-urutan yang ada pada *game* serta alur dari permainan.

8. *Player interaction* yaitu bagaimana pemain saling berinteraksi, interaksi dalam hal ini yaitu interaksi sesama pemain atau interaksi pemain dengan sistem.
9. *Theme (narrativ, backstory, setting)* yaitu pengaturan ini memberikan koneksi emosional dalam permainan, dan tema yang dipilih dengan baik bisa membuat permainan akan lebih mudah dipelajari, serta lebih mudah dimainkan.
10. *Game as system* yaitu setiap elemen formal *game* berkontribusi dalam perjalanan yang mendalam pada pengalaman pemain. Unsur-unsur ini saling terkait, dan perubahannya bisa saling mempengaruhi.

2.3 Iterative with Rapid Prototyping

Secara umum, semakin sering melakukan iterasi semakin baik pula permainan yang dibuat. Oleh karena itu, setiap proses perancangan permainan harus melibatkan iterasi (yaitu melalui keseluruhan siklus perancangan, implementasi dan evaluasi) sebanyak mungkin, serta dengan adanya hal tersebut memungkinkan iterasi yang lebih cepat sehingga menghasilkan permainan yang lebih baik pada akhirnya. Karena itu, perancang video *game* sering kali membuat *prototype* menggunakan media kertas (*paper prototyping*) terlebih dahulu, kemudian mulai melakukan pemrograman saat *prototype* tersebut siap dan perancang yakin bahwa peraturan dasarnya sangat menyenangkan. Hal itu disebut juga sebagai iterasi dengan *rapid prototyping* (Schreiber & Sohn, 2009).



Gambar 2.2 Iterasi dengan *Rapid Prototyping* (Schreiber & Sohn, 2009)

2.3.1 Paper Prototyping

Dalam interaksi manusia-komputer, *paper prototyping* adalah metode yang sering digunakan dalam proses desain berpusat pada pengguna, sebuah proses yang membantu pengembang untuk menciptakan perangkat lunak untuk memenuhi kebutuhan dan harapan pengguna. Dalam kasus ini, terutama untuk merancang dan menguji antarmuka pengguna. Disebut juga prototipe “sekali pakai” yang terdiri dari rancangan kasar, melalui sketsa tangan atau gambar dengan media kertas, *cardboard*, atau apapun dengan tujuan membuat sebuah antarmuka dan digunakan sebagai model, atau desain. *Paper prototyping* merupakan metode sederhana yang dapat menjadi metode pengujian *usability*

agar dapat memberikan banyak masukan sehingga berguna dalam menghasilkan desain produk yang lebih baik (Rizki, 2014).

2.3.2 Digital Prototyping

Dalam *digital prototyping*, *prototype* dimodelkan secara *digital*, bukan secara fisik layaknya *paper prototype*. *Prototype* ini biasanya dibuat dalam sebuah program komputer yang mampu menjalankan simulasi menggunakan parameter dunia nyata dalam pengujian produk, mengevaluasi mekanisme fisik, dan melakukan tes lain yang dilakukan pada *paper prototype*.

Digital prototyping umumnya dilakukan dengan perangkat lunak yang mencakup mesin fisika yang kuat dan komprehensif. Hal ini memungkinkan prototipe untuk menguji daya tahan, kegunaan, dan parameter umum lainnya seperti yang diuji pada *paper prototype*. Sifat *digital prototype* tersebut dapat membuat mereka lebih mudah untuk mengubah atau memodifikasi. Hal ini memungkinkan bahwa dengan *digital prototyping* waktu penyelesaian lebih cepat ketika membuat perubahan seperti warna atau perubahan materi (bisnispemasaran.info, 2017).

2.4 Natural User Interface



Gambar 2.3 Pernyataan Mengenai NUI (Wigdor & Wixdon, 2011)

Dalam bukunya *Brave NUI World*, Wigdor dan Wixdon (2011) mengatakan bahwa sebuah NUI bukan hanya sebuah "*natural user interface*", melainkan sebuah *interface*/antarmuka yang membuat pengguna berbuat sesuatu/bergerak dan merasakan suasana yang alami. Cara termudah untuk mengingatnya adalah dengan mengubah cara membaca "*natural user interface*" – bukan ***natural user interface***, melainkan ***natural user interface***.

NUI (*Natural User Interface*) merupakan istilah umum dari penggabungan beberapa teknologi seperti *multitouch*, *speech recognition* serta *kinectic interface* seperti halnya *kinect*. Di mana teknologi ini menunjukkan karakteristik lain NUI seperti interaksi antar *user* dan komputer seakan terjadi tanpa media interaksi/perantara yang akan terlihat) (Webb & Ashley, 2012). NUI merupakan antarmuka pengguna yang dirancang untuk menggunakan kembali keterampilan yang ada untuk berinteraksi langsung dengan konten (Blažica, 2015).

2.5 Kinect



Gambar 2.4 Device Kinect

Kinect merupakan sebuah perangkat input yang digunakan untuk mendeteksi gerakan di mana perangkat tersebut merupakan hasil pengembangan oleh *Rare*, yang merupakan anak perusahaan *Microsoft* yang ditujukan untuk *XBOX 360* dan *PC/personal computer* dengan sistem operasi *Windows*. Dan dalam teknologi kamera dikembangkan oleh pengembang bernama *PrimeSense* dari Israel. *PrimeSense* mengembangkan sistem, di mana kamera tersebut dapat menginterpretasikan gestur atau gerakan secara spesifik. Dengan menggunakan proyektor infra-merah dan kamera, serta *microchip* khusus untuk mendeteksi tiap pergerakan *object* dan individu pada bidang tiga dimensi, maka kontrol secara *hands-free* dapat dilakukan pada perangkat elektronik. Sistem 3D scanner tersebut dinamakan *Light Coding* yang menggunakan variasi dari rekonstruksi gambar 3D (Wisesa, 2013).

2.6 Pengenalan Unity3D Game Engine

Unity3D Game Engine merupakan mesin permainan 3D yang dibuat dan dikembangkan oleh *Unity Technologies*, di mana menawarkan platform untuk menciptakan permainan dengan desain yang indah dan menarik meliputi 2D, 3D, VR, dan *game AR* dan aplikasi, serta menjadi alat data yang tidak hanya sebatas digunakan untuk membuat atau merancang sebuah *game* tapi juga isinya/konten, *real-time* animasi 3D, dll. *Game Engine Unity3D* mendukung tiga bahasa *script* antara lain *Java Script*, *C#*, dan *Boo* yang memungkinkan interaksi cepat, dapat memanfaatkan *NET library*, dukungan *database*, ekspresi reguler, XML, jaringan, dll. Hanya saja, bahasa pemrograman yang sering digunakan dalam *unity* adalah *C#*. tidak hanya itu *Unity3D* juga mendukung lingkungan multi-platform yang dapat dijalankan pada *operating system/sistem operasi Windows*, *Mac*, *iOS* dan *Android*, juga *play station* seperti *Xbox 360*, *PS3*, *Wii*, serta pada *web browser* (restoprogram.com, 2015).

2.7 Pengujian Perangkat Lunak

2.7.1 Pengujian *Black Box*

Pengujian *black box* berperan untuk melakukan pengujian dari aspek spesifikasi fungsional suatu perangkat lunak. Tujuan dilakukannya pengujian *black box* yakni untuk mengetahui apakah fungsi *input* atau masukan dan fungsi *output* atau keluaran pada perangkat lunak telah sesuai dengan spesifikasi yang dibutuhkan. Di mana Pengujian *black box* dilakukan dengan membuat kasus uji yang bersifat coba-coba, yakni dengan mencoba semua fungsi pada perangkat lunak apakah telah sesuai dengan spesifikasi yang dibutuhkan (Salahuddin & A.S, 2013).

2.7.2 Pengujian *White Box*

Pengujian *white box* berperan untuk melakukan pengujian unit dengan teknik *basis path testing* (Salahuddin & A.S, 2013). Mengenai tahap-tahap yang dilakukan dalam pengujian unit dengan teknik *basis path testing*, yakni:

1. Mendefinisikan *flow graph* berdasarkan *pseudocode*

Pseudocode yang terdapat pada Subbab 5.3.2, disusun ke dalam *flow graph* yang memiliki sekumpulan *node (N)* dan *edge (E)*. *node (N)* merupakan pemetaan setiap proses yang terjadi pada suatu fungsi, sedangkan *edge (E)* merupakan penghubung antar *node* di mana setiap *edge* harus berakhir pada sebuah *node* walaupun tidak mempresentasikan proses apapun.

2. Menentukan ukuran kompleksitas (*cyclomatic complexity*)

Tahap selanjutnya yakni menentukan ukuran kompleksitas (*cyclomatic complexity*). Dalam hal ini *region* dan nilai kompleksitas sebuah *flow graph* dapat dihitung. Di mana *region* merupakan jumlah ruang/daerah yang dibatasi oleh *edge* dan *node*, sedangkan kompleksitas dihitung menggunakan rumus sebagai berikut:

$$\text{Complexity (C}_1\text{)} = (\sum E - \sum N) + 2$$

Keterangan:

C_1 = nilai kompleksitas

$\sum E$ = jumlah *edge* pada suatu *flow graph*

$\sum N$ = jumlah *node* pada suatu *flow graph*

Jika pada *flow graph* terdapat proses seleksi/perulangan (*prediction node*) maka dapat dihitung menggunakan rumus berikut:

$$\text{Complexity (C}_2\text{)} = \sum P(N) + 1$$

Keterangan:

C_2 = nilai kompleksitas

$\sum P(N)$ = jumlah seleksi/perulangan (*prediction node*) pada suatu *flow graph*

3. Menentukan *independent path*

Selanjutnya, menentukan *independent path* berdasarkan jalur yang dimiliki sebuah *node* pada suatu *flow graph*.

2.8 Playtesting

Playtesting atau tes permainan pada pengguna merupakan proses untuk mempelajari reaksi kesenangan dan pemahaman para pemain atau pengguna tentang *game* yang dibuat dengan tujuan memperbaiki *game* tersebut. Macam-macam *playtesting* (Schreiber & Sohn, 2009) meliputi:

1. *Bug testing (Quality Assurance)* bertujuan untuk menemukan kesalahan *game* yang berhubungan dengan kode program.
2. *Focus testing* bertujuan untuk mengetahui seberapa berhasil kah *game* tersebut memenuhi kebutuhan.
3. *Fun testing* untuk mencari tahu apakah *game* yang dibuat sudah memenuhi faktor “*fun*” yang dapat dilihat dari ekspresi pemain ketika memainkan *game* tersebut.
4. *Usability testing* untuk mengetahui apakah pemain dapat dengan mudah melakukan kontrol pada *game*.
5. *Balance testing* dilakukan untuk mengetahui keseimbangan *game*, apakah *game* tersebut sudah dapat dikatakan seimbang.