BAB II
TEORI PENUNJANG

2.1 Wayfinding [1]

Secara etimologi, istilah *wayfinding* terdiri dari dua kata, yaitu *way* dan *find*. Menurut www.dictionary.com, salah satu pengertian kata *way* adalah "a *direction or vicinity; a path or course leading from one place to another*" suatu arah, suatu jalan yang menuntun dari satu tempat ke tempat lain. Sedangkan kata *find* merupakan kata kerja dari kegiatan menemukan atau menempatkan, "to *locate, attain, or obtain by search or effort*". Secara sekilas *wayfinding* dapat diartikan sebagai penemuan arah dari satu tempat ke tempat lain. *Wayfinding* berfungsi untuk menginformasikan seseorang mengenai ruang lingkungan sekeliling yang belum mereka ketahui. Sangatlah penting untuk memberikan informasi mengenai suatu ruang lingkungan untuk menunjukan arah tujuan yang benar.

Karena ketika seseorang berada dalam suatu ruang lingkungan yang baru atau berbeda maka hal ini diterjemahkan dan disimpan secara berbeda di dalam memori otak manusia. Hal-hal seperti jarak, lokasi dan waktu akan diingat secara berbeda dari realitas yang ada.

Sistem *wayfinding* efektif harus berdasar pada tingkah laku manusia dan memiliki karakteristik berikut ini:

- jangan membuat orang berpikir terlalu lama.
- buatlah suatu sistem komunikasi visual yang komprehensif, jelas dan konsisten dengan pesan yang singkat.
- hanya tunjukan yang perlu ditunjukan.
- hanya tunjukan informasi yang relevan untuk jalur ruang, lokasi dan navigasi.
- hilangkan informasi berlebihan.
- bagaiman acara kerja *wayfinding.*



Gambar 2.1 Cara Kerja Wayfinding

Berikut ini mempengaruhi cara kita mengintepretasikan suatu ruang lingkungan:

1. Petanda

Untuk membuat suatu lingkungan yang terbaca maka area atau lokasi spesifik harus ditandai. Hal ini memperkuat pengenalan dari tempat-tempat dan berperan penting bagi area yang lebih besar secara keseluruhan. Dengan penggunaan penanda dan ciri-ciri khusus maka area tersebut akan menjadi lebih terlihat dan dapat lebih dipahami di memori manusia. Penanda dapat berupa objek seni, bangunan, seni jalanan, atau elemen dengan ciri-ciri khusus dalam sebuah lanskap. Elemen-elemen yang telah dikombinasikan ini akan membentuk suatu identitas dari area yang belum diketahui seperti yang terlihat dari lingkungan sekitar.

2. Orientasi

Dalam mencari arah, perlu mengetahui lokasi di sebuah ruang lingkungan dan dimana destinasi lain berada. Agar lebih gampang memahami jarak suatu tempat dengan lainnya pada suatu peta maka harus dapat mengorientasikan posisi di ruang lingkungan. Dalam suatu *wayfinding*, peta memang bisa digunakan untuk mengindikasikan sebuah lokasi.

3. Navigasi

Dengan menggunakan tanda-tanda penunjuk arah maka orang akan lebih mudah mencari jalur yang tepat menuju tujuan.

2.2 Wayfinding Sejenis

Beberapa *mall* di kota Bandung sudah menggunakan *wayfinding* sebagai penunjuk arak untuk mencari toko yang ada di dalam *mall*. Seperti yang di paparkan di bawah.

2.2.1 Wayfinding Mall Bandung Indah Plaza(BIP)

Mall Bandung Indah Plaza menggunakan *wayfinding* jenis peta bangunan dengan media kertas dan di sajikan seperti papan pengumuman, seperti yang di tunjukkkan pada Gambar 2.2.



Gambar 2.2 Peta Lantai 1 Bandung Indah Plaza

Terlihat dari Gambar 2.2 bahwa peta gedung diatas tidak dinamis, peta tidak munjukan seluruh peta lantai gedung. Jika tidak ditemukan ruang yang dicari maka untuk mendapatkan peta lain harus menuju lantai berikutnya. Jadi peta yang di sajikan dengan kertas seperti ini kurang efektif.

2.2.2 Digital Wayfinding Trans Studio Mall

Berbeda dengan Trans Studio Mall yang sudah menggunakan *digital Wayfinding* dengan layar sentuh, seperti yang di tunjukkan pada Gambar 2.3.



Gambar 2.3 Digital Wayfinding Trans Studio Mall

Fitur dari sistem penunjuk arah di Trans Studio Mall adalah sebagai berikut:

1. Mempu menunjukan arah dari titik alat berada ke tempat yang ingin di cari.
2. Mempu melakukan pencarian toko atau tempat yang di tampilkan berdasarkan katergori.
3. Terdapat tombol cepat yang mempersingkat pencarian, seperti: toilet, *escalator*, *wifi zone* dan sebagainya.
4. Terdapat lima menu utama yaitu home, penunjuk, iklan video, berita dan event.

Kekurangan dari sistem penunjuk arah digital yang berada di TSM adalah:

1. Untuk mencari lokasi tujuan belum menggunkan input *text*, sistem menggunkan tombol dan took-toko di kelompokan berdasarkan kategori barang-barang yang dijual.
2. Sistem hanya di *install* di satu lantai, sehingga pengunjung tidak menggunakan sistem untuk mencari ruang ketika berada di lantai yang berbeda.

2.2.3 Map Festival City Link Mall Bandung

Sedangkan Festivalcitylink Mall Bandung menggunakan Website untuk menampilkan *Map* yang dimiliki. Tampilan *Map* pada Website Festival citylink Mall terlihat pada Gambar 2.4.



Gambar 2.4 Tampilan Peta Festival City Link

Kendala ketika menggunakan website untuk mentampilkan sistem *wayfinding* adalah membutuhkan paket data dan kecepatan Internet yang cukup untuk menemukan tempat yang dicari.

2.2.4 Perbandingan Wayfinding

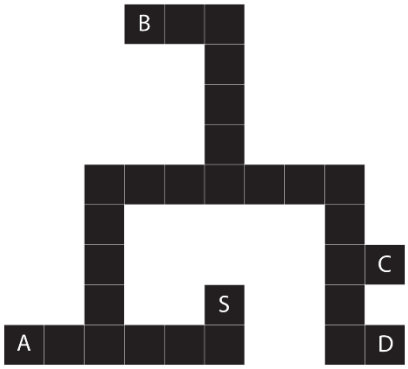
Berdasarkan Observasi lapangan yang dilakukan, dibuatlah tabel perbandingan untuk menentukan keunggulan sistem yang sedang dibangun, berikut tabel perbandingan di paparkan pada Tabel 2.1.

Tabel 2.1 Perbandingan Wayfinding

Spesifikasi	Interactive Digital *Wayfinding*	BIP	TSM	Festival City Link
Visual Menarik	✓	×	✓	×
Interaktif	✓	×	✓	✓
Mudah digunakan	✓	×	×	×
Modern	✓	×	✓	✓
Hemat bagi pengguna	✓	✓	✓	×
Mempermudah Pencarian	✓	×	×	×

2.3 Metode Pencarian Jalur

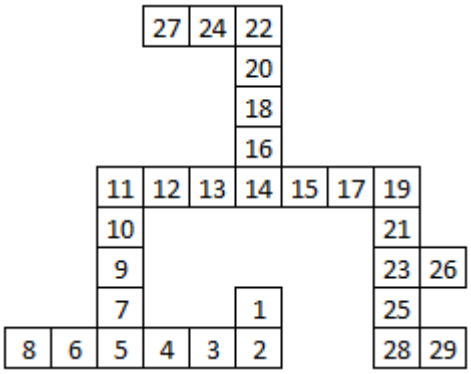
Dalam pembuatan *Interactive digital wayfinding* Berbasis *desktop programming* ini algoritma yang di gunakan "algoritma cepat untuk mempersingkat dan menyederhanakan pencarian jalur" yang memanfaatkan koordinat *pixel* dalam membentuk jalur [2]. Untuk demontrasi dasar pada algoritma yang di ajukan, akan di illustrasikan dalam kasus pada Gambar 2.5 di bawah. Pada gambar terdapat satu titik mulai (S) dan empat titik tujuan (A-D).



Gambar 2.5 Gambar Track Sederhana

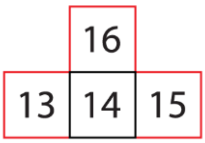
Urutan langkah-langkah yang diterapkan dalam algoritma adalah sebagai berikut:

1. Memberikan indeks angka pada setiap titik *pixel* yang berwarna hitam mulai dari titik mulai hingga ke setiap titik tujuan, seperti yang terlihat pada Gambar 2.6.



Gambar 2.6 Membuat Indeks *Pixel*

2. Membandingkan nilai indeks dari titik tujuan dengan *pixel* disekitarnya, dan memilih nilai indeks yang terkecil. Sebagai contoh kecil di ambil pada titik bernilasi 14 dan tiga titik indeks bernilai 13, 15 dan 16, maka Pencari akan memilih nilai 13 Seperti illustrasi pada Gambar 2.7:



Gambar 2.7 Membandingkan Nilai Indeks Terkecil Disekitas Pencari

3. Sebagai contoh, untuk menemukan jalur dari titik S ke titik B, maka pencarian jalur dimulai dari titik B, berdasarkan nilai indeks yang telah ditetapkan. Penceri akan mencari *pixel* yang mempunyai nilai indeks saja, dan membandingkan nilai indeks *pixel* jika terdapat lebih dari satu titik *pixel* dari titik pencari, lalu memilih nilai indeks kerkecil. Karena dari titik B hanya ada satu titik yang berindeks maka Pencari menuju titik benilai 24, selanjutnya ada dua *pixel* di sekitar pencari yang memiliki nilai yaitu 27 dan 22, maka pencari akan memilih nilai terkecil. Pencarian terus dilakukan pencari dengan melanjutkan proses mambandingkan nilai *pixel* terkecil di sekitar pencari hingga mencapai titik mulai. Dalam kasus pencarian jalur dari titik S ke titik B, pencari memilih dari titik B ke titik S *pixel* yang bernilai (27 – 24 – 22 – 20 – 18 – 16 – 14 – 13 – 12 – 11 – 10 – 9 – 7 – 5 – 4 – 3 – 2 – 1). Untuk menemukan jalur dari S ke B, bisa mengambil kebalikan dari jalur B ke S.

Karena algoritma ini di implementasikan pada "*You Are Here Map*", maka titik mulai akan selalu sama. Sehingga "indeks matriks antar titik" cukup di buat satu kali. Untuk menemukan jalur dari titik S ke titik lainnya cukup mengulangi langkah ke tiga.

2.4 Java

Java adalah bahasa pemrograman yang *multi platform* dan *multi device*. Sekali menuliskan sebuah program dengan menggunakan *Java*, dapat dijalankan hampir di semua komputer dan perangkat lain yang *support Java*, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodenya. Aplikasi dengan berbasis *Java* ini dikompulasikan ke dalam p-code dan bisa dijalankan dengan *Java Virtual Machine*. Fungsionalitas dari *Java* ini dapat berjalan dengan *platform* sistem operasi yang berbeda karena sifatnya yang umum dan non-spesifik [3].

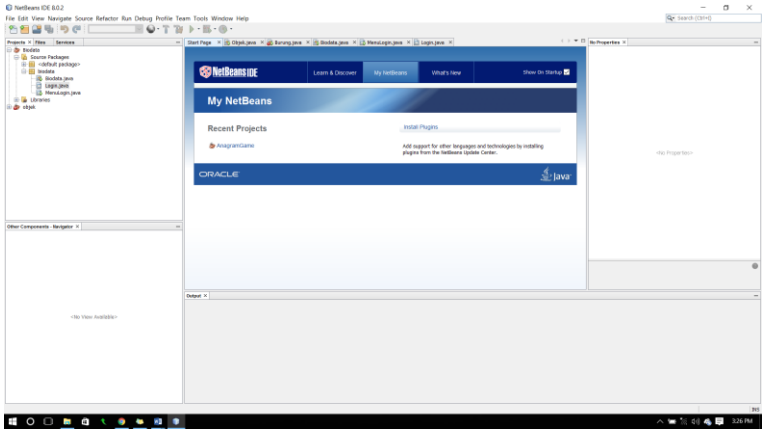


Gambar 2.8 Logo *Software Java*

Sekarang ini *Java* menjadi sebuah bahasa pemrograman yang populer dan dimanfaatkan secara luas untuk pengembangan perangkat lunak. Kebanyakan perangkat lunak yang menggunakan *Java* adalah ponsel *feature* dan ponsel pintar atau *smartphone*.

2.5 NetBean IDE

NetBeans mengacu pada kedua *platform* kerangka untuk aplikasi *desktop Java*, dan sebuah lingkungan pengembangan terpadu (IDE) untuk pengembangan dengan *Java*, JavaScript, PHP, Python, Ruby, Groovy, C, C + +, Scala, Clojure, dan lain-lain [4].



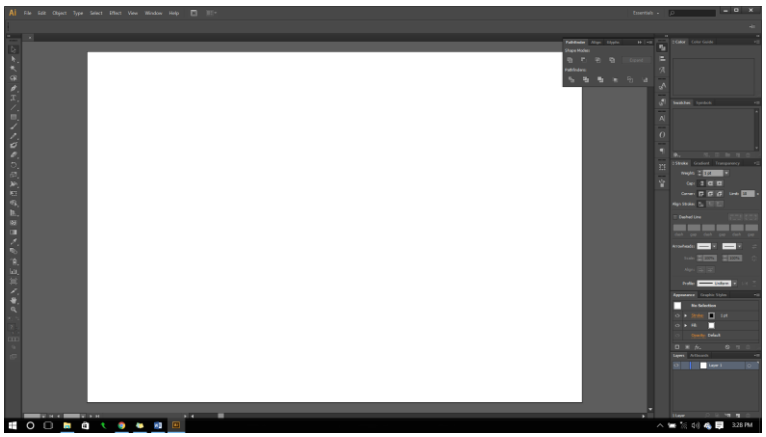
Gambar 2.9 *Interface Software* NetBean

NetBeans IDE ditulis dalam *Java* dan berjalan dimana-mana JVM diinstal, termasuk Windows, Mac OS, Linux, dan Solaris.Sebuah JDK diperlukan untuk pengembangan fungsionalitas *Java*, tetapi tidak diperlukan untuk pembangunan di bahasa pemrograman lain.

Platform NetBeans memungkinkan aplikasi untuk dikembangkan dari satu set modular komponen *software* yang disebut modul. Aplikasi berbasis *platform* NetBeans (termasuk IDE NetBeans) dapat diperpanjang oleh pengembang pihak ketiga.

2.6 Adobe Ilustrator

Adobe Illustrator adalah program *editor* grafis *vektor* terkemuka, dikembangkan dan dipasarkan oleh *Adobe Systems*. Illustrator CS6 merupakan versi terkini program ini, generasi keenam belas untuk produk Illustrator.

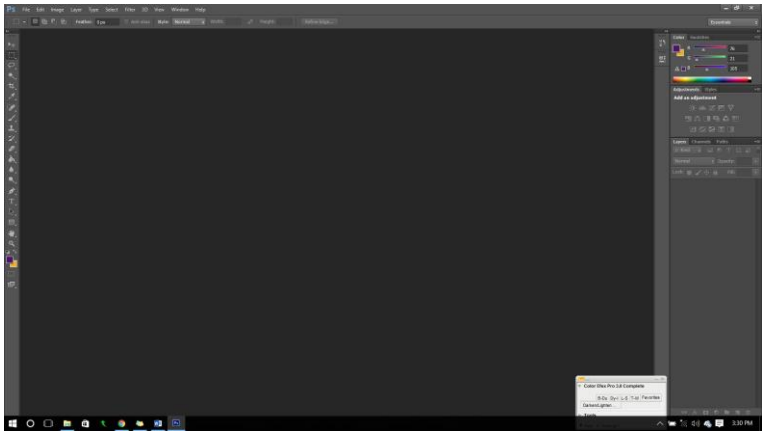


Gambar 2.10 *Interface Software* Adobe Illustrator

Dalam perancangan dan pembuatan *System interactive digital signage and wayfinding*, *Software* Adobe Illustrator di gunakan untuk membuat perancangan *Graphic User Interface (GUI)*, agar penampilan antarmuka pengguna menjadi lebih menarik.

2.7 Adobe Photoshop

Adobe Photoshop, atau biasa disebut Photoshop, adalah perangkat lunak *editor* citra buatan Adobe *Systems* yang dikhususkan untuk pengeditan foto/gambar dan pembuatan efek. Perangkat lunak ini banyak digunakan oleh fotografer digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (*market leader*) untuk perangkat lunak pengolah gambar/foto, dan, bersama Adobe Acrobat, dianggap sebagai produk terbaik yang pernah diproduksi oleh Adobe Systems. Versi kedelapan aplikasi ini disebut dengan nama Photoshop CS (*Creative Suite*), versi sembilan disebut Adobe Photoshop CS2, versi sepuluh disebut Adobe Photoshop CS3, versi kesebelas adalah Adobe Photoshop CS4, versi keduabelas adalah Adobe Photoshop CS5, dan versi terbaru adalah Adobe Photoshop CS6.

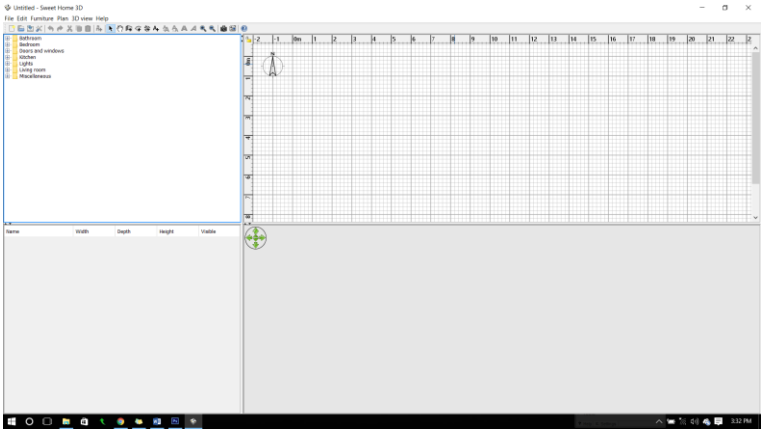


Gambar 2.11 *Software* Adobe Photoshop

Adobe Photoshop bergunakan untuk memanipulasi gambar yang digunakan dalam pembuatan *system interactive digital signage and wayfinding.* Seperti background, foto ruangan, dan sebagainya.

2.8 Sweet Home 3D

Sweet Home 3D adalah perangkat lunak yang digunakan untuk merancang bangunan dalam benduk design tiga dimensi.



Gambar 2.12 *Interface* Sweet Home 3D

2.9 Unified Modeling Language (UML)

Unified Modeling Languange (UML) adalah "bahasa standar untuk menulis *blueprint* perangkat lunak". UML dapat digunakan untuk menvisualisasikan, menentukan, membangun dan mendokumentasikan artefak pada sebuah sistem perangkat lunak berorientasi objek [5]. Karena UML ini merupakan bahasa visual untuk pemodelan bahasa berorientasi objek, maka semua elemen dan diagram berbasiskan kepada paradigma object oriented.

UML memberikan standar penulisan sebuah sistem *blueprint*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem *software*. UML digunakan untuk pemodelan aplikasi pada tahap analisis kebutuhan aplikasi. Terdapat banyak cara dalam pemodelan menggunakan UML diantaranya yang paling banyak digunakan yaitu:

1. *Use Case* Diagram

   Use Case Diagram merupakan diagram yang membantu menentukan fungsi serta kegunaan perangkat lunak dari sisi pengguna. *Use Case* Diagram menggambarkan interaksi pengguna terhadap sistem ketika system digunakan.

2. *Sequence* Diagram

   Sequence Diagram atau diagram urutan merupakan suatu diagram yang memperlihatkan interaksi dinamis diantara objek ketika sedang melakukan suatu proses. Interaksi tersebut melibatkan pengguna aplikasi, antarmuka, pesan, dan sistem sebagai penyimpan data.

3. *Collaboration* Diagram

   Collaboration Diagram merupakan diagram yang dapat memperlihatkan bagaimana objek dalam suatu sistem bekerja satu sama lain. Diagram ini menitik beratkan pemahaman atau operasi yang dilakukan objek selama aplikasi dijalankan.

4. *Class* Diagram

   Class Diagram merupakan diagram yang didalamnya terdapat atribut, operasional, dan hubungan dari pada sistem dan asosiasi dari antar kelas.