BAB 2
TINJAUAN PUSTAKA

2.1. Profil Tempat Penelitian

2.1.1. Sejarah Instansi

Diresmikan oleh Gubernur Jenderal J.B. Van Heutsz pada tanggal 3 Januari 1909. Pada awalnya bernama koningen Wilhelmina Gasthuis voor Ooglijders dengan direktur pertamanya dr. CHA Westhoff.

Sempat menjadi Rumah Sakit Umum pada jaman pendudukan Jepang tahun 1942-1945, menggantikan posisi Rumah Sakit Rancabadak yang menjadi Rumah Sakit Militer. Kemudian berganti nama sesuai dengan nama jalan dimana rumah sakit ini berapa menjadi Rumah Sakit Mata Tjitjendo pada tahun 1980.

Rumah Sakit Mata Cicendo merupakan satu-satunya rumah sakit mata milik pemerintah republik Indonesia yang berada dibawah Direktorat Jenderal Bina Pelayanan Medik Departemen Kesehatan Republik Indonesia [2].

2.1.2. Logo Instansi



Gambar 2.1 Logo Rumah Sakit Mata Cicendo

2.1.3. Badan Hukum Instansi

Rumah Sakit Mata Cicendo merupakan sakit mata milik pemerintah republik Indonesia yang berada dibawah Direktorat Jenderal Bina Pelayanan Medik Departemen Kesehatan Republik Indonesia [2].

1. SK Menkes No. 1040 Tahun 1992, RS MATA RUJUKAN NASIONAL Tipe B Non Pendidikan
2. SK Menkes No.045/M/PER/I/2007, 15 Januari 2007

3. SK Menteri Kesehatan RI No. 059/MENKES/SK/I/2009, 16 Januari 2009 PUSAT MATA NASIONAL

2.1.4. Visi dan Misi

A. Visi

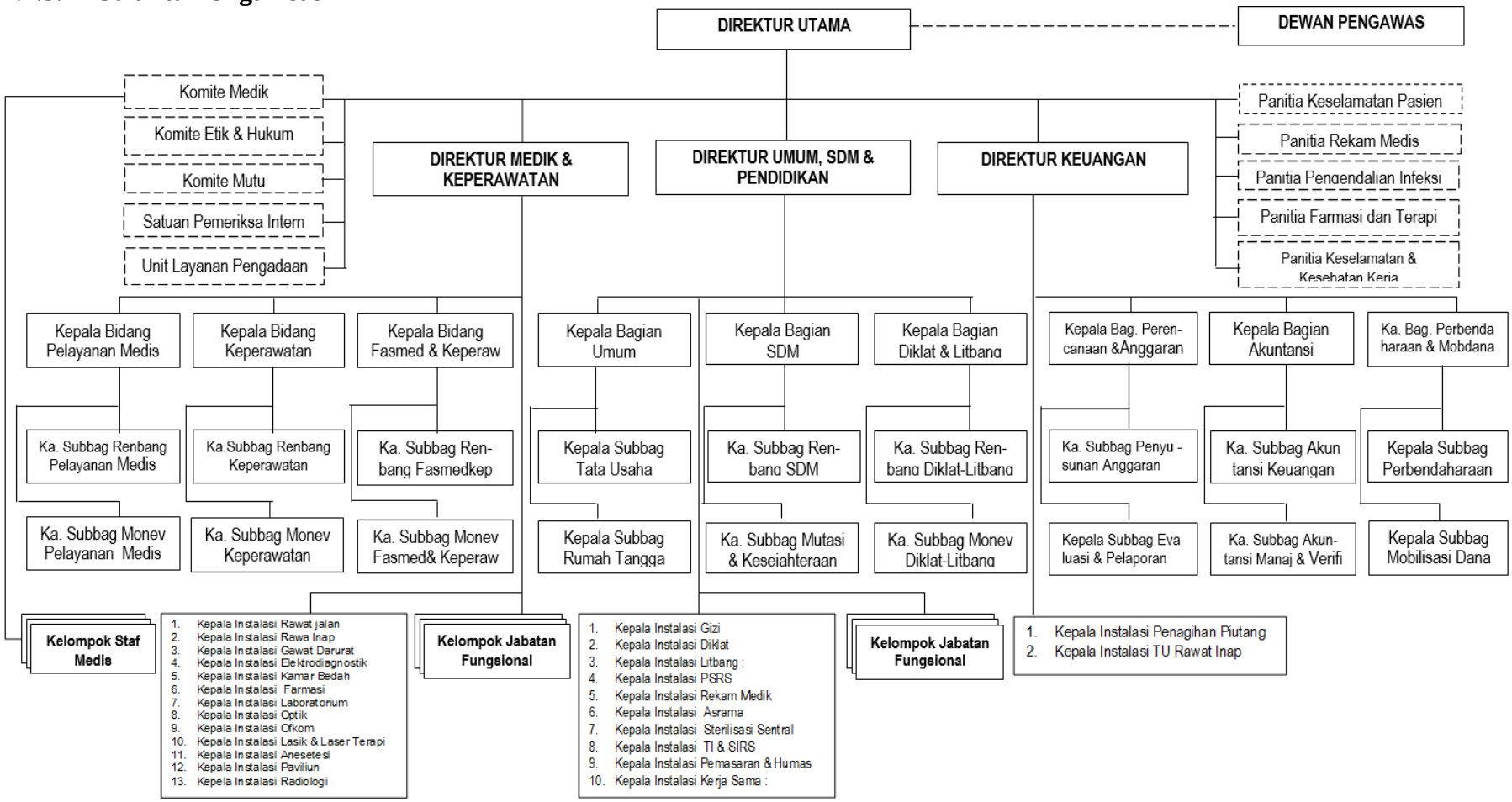
Menjadi Rumah Sakit Mata Rujukan Nasional yang berstandar internasional di bidang pelayanan, pendidikan dan penelitian kesehatan mata pada tahun 2019.

B. Misi

Rumah Sakit Mata Cicendo Bandung memiliki misi sebagai berikut:

1. Memberikan pelayanan kesehatan mata yang paripurna sesuai dengan standard dunia yang berorientasi pada kepuasan bagi seluruh lapisan masyarakat, terjangkau, merata, dan berkeadilan.
2. Memberikan peluang dan lingkungan yang kondusif untuk penyelenggaraan pendidikan kesehatan mata yang inovatif.
3. Menyelenggarakan penelitian dan pengembangan serta penapisan ilmu dan teknologi di bidang kesehatan mata.
4. Melaksanakan pengabdian dan pemberdayaan masyarakat dalam upaya pemeliharaan dan peningkatan kesehatan mata.
5. Meningkatkan upaya kemitraan secara global.
6. Meningkatkan profesionalisme pegawai.

2.1.5. Struktur Organisasi



Gambar 2.2 Struktur Organisasi Rumah Sakit Mata Cicendo Bandung

2.2. Landasan Teori

2.2.1. *Multimedia*

Multimedia adalah penggunaan komputer untuk menyajikan dan menggabungkan teks, suara, gambar, animasi dan video dengan alat bantu (*tool*) dan koneksi (*link*) sehingga pengguna dapat bernavigasi, berinteraksi, berkarya dan berkomunikasi [3]. komponen multimedia terbagi atas lima jenis yaitu:

1. **Teks**

Teks merupakan elemen multimedia yang menjadi dasar untuk menyampaikan informasi, karena teks adalah jenis data yang paling sederhana dan membutuhkan tempat penyimpanan yang paling kecil. Teks merupakan cara yang paling efektif dalam mengemukakan ide-ide kepada pengguna, sehingga penyampaian informasi akan lebih mudah dimengerti oleh masyarakat. Jenis-jenis teks seperti *Printed Text*, yaitu teks yang dihasilkan oleh *word processor* atau *word editor* dengan cara diketik yang nantinya dapat dicetak. *Scanned Text* yaitu teks yang dihasilkan melalui proses scanning tanpa pengetikan. Dan *Hypertext* yaitu jenis teks yang memberikan link ke suatu tempat / meloncat ke topik tertentu.

2. **Grafik (*image*)**

Sangat bermanfaat untuk mengilustrasi informasi yang akan disampaikan terutama informasi yang tidak dapat dijelaskan dengan kata-kata. Jenis-jenis grafik seperti *bitmap* yaitu gambar yang disimpan dalam bentuk kumpulan pixel, yang berkaitan dengan titik-titik pada layar *monitor*. *Digitized picture* adalah gambar hasil rekaman video atau kamera yang dipindahkan ke komputer dan diubah ke dalam bentuk bitmaps. *Hyperpictures*, sama seperti *hypertext* hanya saja dalam bentuk gambar.

3. ***Audio***

Multimedia tidak akan lengkap jika tanpa *audio* (suara). *Audio* bisa berupa percakapan, musik atau efek suara.

Format dasar *audio* terdiri dari beberapa jenis :

a) WAVE

Merupakan format file digital audio yang disimpan dalam bentuk digital dengan eksistensi WAV.

b) MIDI (*Musical Instrument Digital Interface*)

MIDI memberikan cara yang lebih efisien dalam merekam music dibandingkan wave, kapasitas data yang dihasilkan juga jauh lebih kecil. MIDI disimpan dalam bentuk MID.

4. ***Video***

Video menyediakan sumber yang kaya dan hidup untuk aplikasi multimedia. Dengan *video* dapat menerangkan hal-hal yang sulit digambarkan lewat kata-kata atau gambar diam dan dapat menggambarkan emosi dan psikologi manusia secara lebih jelas.

5. Animasi

Animasi adalah simulasi gerakan yang dihasilkan dengan menayangkan rentetan frame ke *layer*. *Frame* adalah satu gambar tunggal pada rentetan gambar yang membentuk animasi. Menurut Foley, Van Dam, Feiner dan Hughes [4] animate adalah untuk membuat sesuatu hidup, sebagian orang mengira bahwa animasi itu sama dengan motion (gerakan), tetapi animasi mencakup semua yang mengandung efek visual sehingga animasi mencakup perubahan posisi terhadap waktu, bentuk, warna, struktur, tekstur dari sebuah objek, posisi kamera, pencahayaan, orientasi dan *focus* dan perubahan dalam teknik *rendering*.

2.2.2. Grafika Komputer

Seperangkat alat yang terdiri dari hardware dan software untuk membuat gambar, grafik atau citra realistik untuk seni, game komputer, foto dan animasi komputer [5].

Merupakan sekumpulan alat terdiri dari *hardware* dan *software* yang digunakan untuk membuat gambar (*to create pictures*) dan berinteraksi dengan gambar dengan cara-cara seperti yang biasa digunakan [6].

Dari pengertian di atas dapat disimpulkan bahwa grafika komputer adalah teknik untuk membuat gambar obyek sesuai dengan obyek tersebut di alam nyata

dan merupakan suatu proses pembuatan, penyimpanan dan manipulasi model dan citra.

Grafika Komputer secara umum dapat dibagi menjadi dua bagian, yaitu grafika 2 dimensi dan grafika 3 dimensi. Dalam teknik penyajian, grafika 2 dimensi memiliki kekurangan, yaitu ketidakmampuan untuk merepresentasikan objek 3 dimensi. Grafika 3 dimensi dapat menggambarkan sebuah benda dalam dunia nyata menjadi sebuah objek dari berbagai sudut pandang. Tidak seperti grafika 2 dimensi, dalam grafika 3 dimensi banyak sekali hal yang harus dipertimbangkan.

2.2.3. Elemen – Elemen Grafika Komputer

2.2.3.1. Objek 3 Dimensi

Secara konseptual dalam grafika komputer objek 3 dimensi dibuat dari rangkaian poligon. Poligon adalah sejumlah dari segmen garis yang dihubungkan antara ujung garis satu dengan yang lainnya membentuk suatu bidang tertutup. Hal yang membedakan 2 dimensi dan 3 dimensi adalah kedalaman. Kedalaman didefinisikan sebagai jarak antara *viewer* terhadap benda yang dia lihat. Ini berarti berbeda dengan 2 dimensi yang hanya menggunakan 2 ukuran, yaitu panjang dan lebar, maka 3 dimensi menggunakan 3 ukuran, yaitu panjang, lebar dan kedalaman. Secara geometri ketiga ukuran tersebut disimbolkan dengan sumbu x, y, dan z. 3D tidak hanya digunakan dalam matematika dan fisika saja melainkan dibidang grafis, seni, animasi, komputer dan lain-lain. Konsep tiga dimensi atau 3D menunjukkan sebuah objek atau ruang memiliki tiga dimensi geometris yang terdiri dari kedalaman, lebar dan tinggi. Contoh tiga dimensi suatu objek atau benda adalah bola, piramida atau benda spasial seperti kotak sepatu.

2.2.4. *Virtual Reality*

Virtual Reality 3D merupakan bagian dari objek *multimedia*. *Virtual reality* adalah gambar 3D di komputer atau di dunia lain (maya) yang dapat membuat orang merasa di dunia nyata dan dapat melakukan operasi dan kontrol sistem kehidupan nyata.

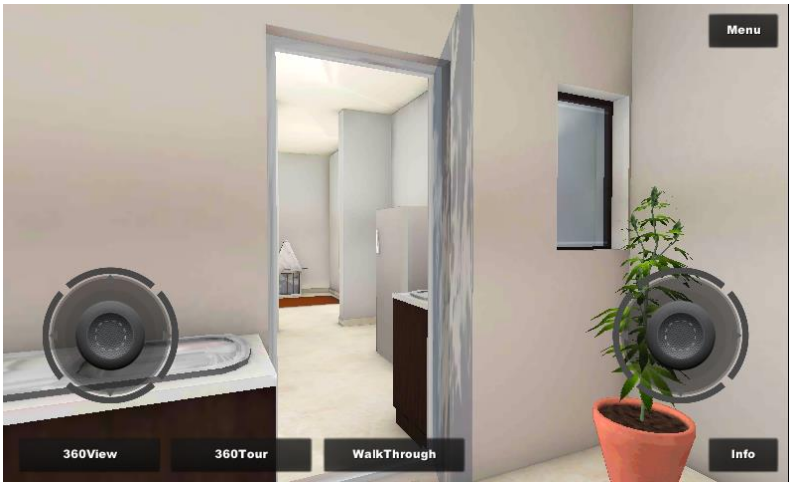
Pada umumnya *virtual reality* hanya mengutamakan pengalaman visual yang ditampilkan lewat layar komputer atau melalui *display* stereoskopik khusus, tetapi beberapa simulasi juga memiliki beberapa informasi sensorik tambahan, contohnya seperti suara melalui *speaker* atau *headphone*. *Virtual reality* juga dapat meliputi komunikasi jarak jauh yang dapat menampilkan kehadiran user secara *virtual* melalui penggunaan dari perangkat input standar seperti keyboard dan mouse, atau melalui perangkat lainnya seperti polhemus (*wired glove*). Lingkungan yang disimulasi oleh virtual reality dapat dibuat menyerupai dunia nyata, contohnya seperti simulasi untuk pilot atau pelatihan untuk pertempuran, atau dapat berbeda dari kenyataaan secara signifikan, contohnya seperti di dalam game yang berbasis virtual reality. Virtual reality sering digunakan dalam pembuatan berbagai macam aplikasi, contohnya seperti aplikasi yang berfungsi sebagai terapi kecemasan, aplikasi untuk pelatihan seperti *flight simulator* dan *ship simulator*, *multiplayer online gaming* seperti *World of Warcraft* dan *Second Life*, Nintendo Wii, dan *virtual tour* [7].

2.2.5. *Virtual Tour*

Virtual tour adalah simulasi dari sebuah lokasi yang sesungguhnya, umumnya terdiri oleh sequence video atau kumpulan foto. *Virtual tour* juga dapat menggunakan beberapa elemen multimedia lain, contohnya seperti sound effect, musik, narasi, dan teks, dan sering digunakan untuk menggambarkan varietas dari media yang berbasis video dan fotografi. *Virtual tour* merupakan bagian dari *virtual reality*. *Panoramic tour* dan *virtual tour* sering dianggap memiliki arti yang sama, yaitu sebuah virtual tour yang dibuat dengan menggunakan bantuan kamera, terdiri dari sejumlah foto yang diambil dari satu posisi yang strategis dengan cara kamera dan lensa diputar mengelilingi satu lokasi yang sering disebut dengan istilah no-*parallax point* (titik tepat di belakang lensa dimana cahaya bertemu).

Virtual Tour juga dapat dikatakan sebagai sebuah program yang menggabungkan teknologi fotografi dengan teknologi informasi (IT) yang bertujuan memberikan informasi ruang (*space*) secara menyeluruh (3 dimensi) dan

interaktif. Informasi ruang (*space*) yang bisa diolah menjadi aplikasi ini meliputi ruang *indoor* maupun *outdoor* [7].



Gambar 2.3 Contoh Virtual Tour 3D Pada Real Estate



Gambar 2.4 Contoh *Virtual Tour* Foto 360º

2.2.6. *3D Modeling*

3D *modeling* adalah proses mengembangkan matematika representasi dari setiap tiga-dimensi benda (baik benda mati atau hidup) melalui perangkat lunak khusus. Produk ini disebut sebagai model 3D. Hal ini dapat ditampilkan sebagai gambar dua dimensi melalui proses yang disebut 3D *rendering* atau digunakan

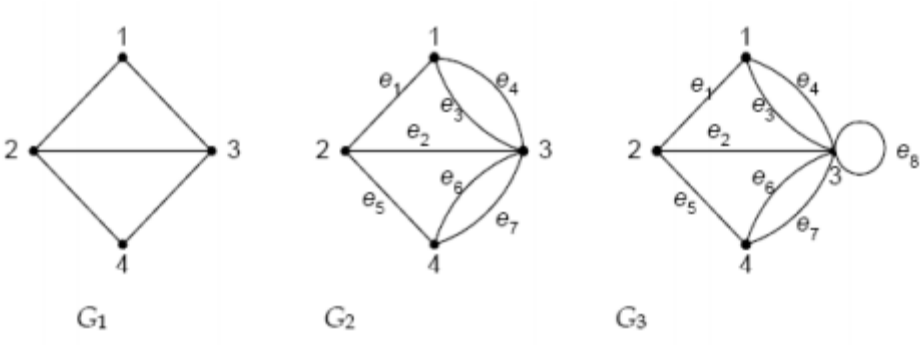
dalam komputer simulasi fenomena fisik. Model juga dapat secara fisik dibuat menggunakan perangkat *Printing* 3D. Model dapat dibuat secara otomatis atau manual. Manual proses pemodelan geometris mempersiapkan data untuk komputer grafis 3D mirip dengan seni plastik seperti mematung [8].

2.2.7. *Low Poly*

Low Poly adalah *polygon mesh* dalam 3D *computer graphics* yang memiliki jumlah polygon yang relatif kecil. Low Poly Mesh digunakan pada saat aplikasi *real-time* seperti game. Jumlah poligon dalam mesh merupakan faktor penting untuk mengoptimalkan kinerja tetapi dapat memberikan penampilan yang tidak diinginkan dengan grafis yang dihasilkan.

2.2.8. Teori Dasar Graf

Graf didefinisikan sebagai pasangan himpunan (V, E), ditulis dengan notasi G = (V, E), yang dalam hal ini V adalah himpunan tidak kosong dari simpul-simpul (vertices atau node) dan E adalah himpunan sisi (edges atau arcs) yang menghubungkan sepasang simpul [6].Simpul pada graf dapat dinomori dengan huruf, seperti a, b, c...dst, dengan bilangan asli 1, 2, 3...dst, atau gabungan keduanya. Sedangkan sisi yang menghubungkan simpul u dengan simpul v dinyatakan dengan pasangan (u, v) atau dinyatakan dengan lambang e1, e2....en dengan kata lain, jika e adalah sisi yang menghubungkan simpul u dengan simpul v, maka e dapat ditulis sebagai e = (u, v). Secara geometri graf digambarkan sebagai sekumpulan noktah (simpul) di dalam bidang dwimatra yang dihubungkan dengan sekumpulan garis (sisi).



Gambar 2.5 (G1) graf sederhana, (G2) multigraf, dan (G3) multigraf

Gambar di atas memperlihatkan tiga buah graf, G1, G2 dan G3. G1 adalah graf dengan himpunan simpul V dan himpunan sisi E adalah:

V = {1, 2, 3, 4}

E = {(1, 2), (1, 3), (2, 3), (2, 4), (3, 4)}

G2 adalah graf dengan himpunan simpul V dan himpunan sisi E adalah:

V = {1, 2, 3, 4}

E = {(1, 2), (2, 3), (1, 3), (1, 3), (2, 4), (3, 4), (3, 4)}

= {e1, e2, e3, e4, e5, e6, e7}

G3adalah graf dengan himpunan simpul V dan himpunan sisi E adalah:

V = {1, 2, 3, 4}

E = {(1, 2), (2, 3), (1, 3), (1, 3), (2, 4), (3, 4), (3, 4), (3, 3)}

= {e1, e2, e3, e4, e5, e6, e7, e8}

Pada G2, sisi e3 = (1, 3) dan sisi e4= (1, 3) dinamakan sisi-ganda (*multiple edges atau parallel edges*) karena kedua sisi ini menghubungi dua buah simpul yang sama, yaitu simpul 1 dan simpul 3. Pada G3 , sisi e8 = (3, 3) dinamakan gelang atau kalang (*loop*) karena ia berawal dan berakhir pada simpul yang sama.

2.2.9. Graf Berbobot (Weighted Graph)

Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga (bobot) [9]. Bobot pada tiap sisi dapat berbeda-beda bergantung pada masalah yang dimodelkan dengan graf. Bobot dapat menyatakan jarak antara dua buah kota, biaya perjalanan antara dua buah kota, waktu tempuh pesan (*message*) dari sebuah simpul

komunikasi ke simpul komunikasi lain (dalam jaringan komputer), ongkos produksi, dan sebagainya.

2.2.10. Lintasan Terpendek

Persoalan mencari lintasan terpendek di dalam graf merupakan salah satu persoalan optimasi. Graf yang digunakan dalam pencarian lintasan terpendek adalah graf berbobot (*weighted graph*), yaitu graf yang setiap sisinya diberikan suatu nilai atau bobot. Bobot pada sisi graf dapat menyatakan jarak antar kota, waktu pengiriman pesan, ongkos pembangunan, dan sebagainya. Asumsi yang kita gunakan di sini adalah bahwa semua bobot bernilai positif. Kata terpendek berbeda-beda maknanya bergantung pada tipikal persoalan yang akan diselesaikan. Namun, secara umum terpendek berarti meminimisasi bobot pada suatu lintasan dalam graf [9].

2.2.11. Algoritma Dijkstra

Algoritma yang ditemukan oleh Dijkstra untuk mencari path atau rute terpendek. Misalkan G adalah graf berarah berlabel dengan titik-titik V(G) = {v1, v2,..., vn} dan path terpendek yang dicari adalah dari v1 ke vn. Algoritma Dijkstra dimulai dari titik v1. dalam iterasinya, algoritma akan mencari satu titik yang jumlah bobotnya dari titik 1 terkecil. Titik-titik yang terpiih dipisahkan dan titik-titik tersebut tidak diperhatikan lagi dalam iterasi berikutnya.

Misalkan:

V (G) = {v1, v2,.., vn}

L = Himpunan titik-titik ε V (G) yang sudah terpilih dalam jalur path terpendek.

D(j) = Jumlah bobot path terkecil dari v1 ke vj

w(i,j) = Bobot garis dari titik vi ke vj

w*(1,j) = Jumlah bobot path terkecil dari v1 ke vj

Secara formal, algoritma Dijkstra untuk mencari path terpendek adalah sebagai berikut:

1) L = { };
   V = {v2, v3,..,vn}.

2) Untuk i = 2, ..., n, lakukan D(i) – w(1, i)
3) Selama vn ∉ L lakukan:
    a. Pilih titik vk ∈ V - L dengan D(k) terkecil.
       L = L ∪ {vk}
    b. Untuk setiap vj ∈ V - L lakukan:
4) Jika D(j) > D(k) + W(k,j) maka ganti D(j) dengan D(k) + W(k,j)
5) Untuk setiap vj ∈ V, w*(1, j) = D(j)

Menurut algoritma di atas, path terpendek dari titik v1 ke vn adalah melalui titik- titik dalam L secara berurutan, dan jumlah bobot path terkecilnya adalah D(n). Algoritma Dijkstra dinyatakan dalam *pseudo-code* berikut ini(Rinaldi Munir 2005 , hal. 414):

```
procedure Dijkstra (input m:matriks, a:simpul awal)
(
  Mencari lintasan terpendek dari simpul awal a ke semua simpul lainnya
  Masukan : matriks ketetanggaan (m) dari graf berbobot G dan simpul awal a
  Keluaran : lintasan terpendek dari a ke semua simpul lainnya
        Deklarasi
        s1, s2, ..., sn :integer (tabel integer)
        d1, d2, ..., dn :integer (tabel integer)
        i, j, k: integer
        Algoritma
        ( langkah 0 (Inisialisasi:)
        for i ← 1 to n do
        si← 0
        di ← mai
        endfor
        (langkah 1 :)
        Sa←1 (karena simpul a adalah simpul asal lintasan terpendek, jadi
        simpul a sudah pasti terpilih dalam lintasan terpendek )
        Da← oo (tidak ada lintasan terpendek dari simpul a ke a)
```

```
        (langkah 2, 3, ..., n-1:)
        For k ← 2 to n-1 do
          J← simpul dengan sj = 0 dan dj minimal
          Sj ← 1 {simpul j sudah terpilih ke dalam lintasan terpendek}
          {perbaharui tabel d}
          For semua simpul I dengan si= 0 do
            If dj+mji <di then
            Di←dj+mji
            Endif
          Endfor
        Endfor
)
```

2.2.12. *QR Code*

QR Code adalah image berupa matriks dua dimensi yang memiliki kemampuan untuk menyimpan data di dalamnya. *QR Code* merupakan evolusi dari kode batang (barcode). *Barcode* merupakan sebuah simbol penandaan objek nyata yang terbuat dari pola batang-batang berwarna hitam dan putih agar mudah untuk dikenali oleh komputer [10]. Contoh *QR Code* bisa dilihat pada gambar berikut.



Gambar 2.6 *QR Code*

QR Code merupakan singkatan dari *Quick Response Code*, atau dapat diterjemahkan menjadi kode respon cepat. *QR Code* dikembangkan oleh Denso Corporation, sebuah perusahaan Jepang yang banyak bergerak di bidang otomotif. *QR Code* ini dipublikasikan pada tahun 1994 dengan tujuan untuk pelacakan

kendaraan di bagian manufaktur dengan cepat dan mendapatkan respon dengan cepat pula.

Seiring berkembangnya *QR Code*, semakin banyak penelitian yang dilakukan mengenai kode simbol ini. Berbagai penelitian terus dilakukan, baik untuk menambah jumlah data yang dapat disimpan dalam *QR Code*, menambah resistensi terhadap kerusakan, dan lain-lain. Beberapa penelitian yang telah dilakukan diantaranya adalah :

1. Pembuatan aplikasi pembacaan *QR Code* menggunakan perangkat *mobile* berbasis J2ME
2. *QR Code* untuk tandatangan *digital*
3. *QR Code* untuk autentikasi novel *user*
4. *QR Code* untuk edukasi

2.2.13. *UML (Unified Modeling Language)*

Banyak sekali model dalam pengembangan software, ada yang terstruktur dan ada juga yang berorientasi objek. UML (*Unified Modeling Language*) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML merupakan pengganti dari metode objek orientasi analisis dan objek orientasi *design* (OOA&D). yang hadir sekitar akhir tahun 1980-an dan awal 1990. UML Menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi perangkat lunak, dimana aplikasi tersebutdapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dengan bahasa pemrograman apapun, tetapi karena UML menggunakan class dan *operation* dalam konsep dasar, maka akan lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek.

UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai aliran diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Berikut adalah turunan dari tiga notasi yang telah ada sebelumnya.

1. Grady Booch OOD (*Object-Oriented Design*)
2. Jim Rumbauhgh OMT (*Object Modeling Technique*)
3. Ivar Jacobson OOSE (*Object-Oriented Software enginering*)

UML merupakan bahsa untuk menspesifikasikan, memvisualisasikan, membangun dan mendokumentasikan bagian dari informasi yang digunakan atau dihasilkan dari proses pembuatan perangkat lunak (*Artifacts*), Didalamnya dapat berupa model, deskripsi atau deskripsi dari perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. UML (*Unified Modeling Languag*e) dibuat oleh Grady Booch, James

Rumbaugh, dan Ivan Jacobson dibawah naungan Rational Software Corp. UML sendiri terdiri atas pengelompokkan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Diagram adalah yang menggambarkan permasalahan maupun solusi dari permasalahan suatu model. UML mempunyai 9 diagram, yaitu; *use-case, class, object, state, sequence, collaboration, activity, 28 component*, dan *deployment diagram*. Diagram pertama adalah use case menggambarkan sekelompok use cases dan aktor yang disertai dengan hubungan diantaranya. Diagram use cases ini menjelaskan dan menerangkan kebutuhan/*requirement* yang diinginkan/dikehendaki pengguna, serta sangat berguna dalam menentukan struktur organisasi dan model dari pada sebuah sistem.

2.2.13.1.*Use Case Diagram*

Use case merupakan gambaran umum dari fungsi atau proses utama yang menggambarkan tentang salah satu perilaku sistem. Perilaku sistem ini terdefinisi dari proses bisnis sistem yang akan dimodelkan. Tidak semua proses bisnis digambarkan secara fungsional pada use case, tetapi yang digambarkan hanya fungsionalitas utama yang berkaitan dengan sistem. Use case menitik beratkan bagaimana suatu sistem dapat berinteraksi baik antar sistem maupun diluar sistem. Contoh use case diagram dapat dilihat pada gambar berikut.

2.2.13.2.*Activity Diagram*

Activity diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. Dalam beberapa hal, *activity diagram* memainkan

peran mirip diagram alir, tetapi perbedaan prinsip antara notasi diagram alir adalah activity diagram mendukung *behavior* paralel. *Node* pada sebuah *activity* diagram disebut sebagai action, sehingga diagram tersebut menampilkan sebuah *activity* yang tersusun dari *action*.

2.2.13.3.*Sequence Diagram*

Sequence diagram adalah grafik dua dimensi dimana obyek ditunjukkan dalam dimensi horizontal, sedangkan *lifeline* ditunjukkan dalam dimensi vertikal.

2.2.13.4.*Class Diagram*

Class diagram merupakan himpunan dari objek-objek yang sejenis. Sebuah objek memiliki keadaan sesaat (state) dan perilaku (*behavior*). State sebuah objek adalah kondisi objek tersebut yang dinyatakan dalam *attribute/properties*. Sedangkan perilaku suatu objek mendefinisikan bagaimana sebuah objek bertindak/beraksi dan memberikan reaksi.

2.2.13.5.*Collaboration Diagram*

Collaboration diagram adalah perluasan dari objek diagram. Objek diagram menunjukkan objek-objek yang hubungannya satu dengan yang lain. *Collaboration diagram* menunnjukkan *mesage-message* objek yang dikirim satu sama lain.

2.2.13.6.*Component Diagram*

Component diagram mempresentasikan dunia riil item yaitu *component software*. *Component software* adalah bagian fisik dari sebuah sistem karena menetap di komputer. Component diagram mengandung component, iterface dan relationship.

2.2.13.7.*Deployment Diagram*

Deployment diagram, menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian software yang berjalan pada bagian-bagian *hardware*.

2.2.14. Pengujian *Black Box*

Metode pengujian *blackbox* fokus pada keperluan penelusuran kesalahan fungsional dari software. Ujicoba *blackbox* berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

Tujuan metode ini mencari kesalaman pada:

1. Fungsi-fungsi yang salah atau hilang
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan performa
5. kesalahan inisialisasi dan terminasi

Teknik pengujian black box terdiri dari 10 jenis diantaranya *Equivalence Partitioning, Boundary Value Analysis/Limit Testing, Comparison Testing, Sample Testing, Robustness Testing, Behavior Testing, Requirement Testing, Performance Testing, Endurance Testing, Cause-Effect Relationship Testing*. Salah satunya yang akan dibahas adalah *Equivalence partitioning*.

Equivalence partioning merupakan metode ujicoba *blackbox* yang membagi domain input dari program menjadi beberapa kelas data dari kasus ujicoba yang dihasilkan. Kasus uji penanganan single yang ideal menemukan sejumlah kesalahan (misalnya : kesalahan pemrosesan dari seluruh data karakter) yang merupakan syarat lain dari suatu kasus yang dieksekusi sebelum kesalahan umum diamati.

2.2.15. Pengujian Kuesioner

Kuesioner adalah sebuah daftar pernyataan yang harus diisi oleh orang yang akan dievaluasi (responden). Metode yang digunakan dalam kuesioner pada penelitian ini adalah skala Likert. Dalam skala likert, responden diminta untuk membaca dengan seksama setiap pernyataan yang disajikan, kemudian ia diminta untuk menilai pernyataan-pernyataan tersebut.

Skala likert sering digunakan untuk mengukur sikap, pendapat, dan presepsi seseorang atau sekelompok orang tentang suatu gejala atau fenomena. Skala likert meminta kepada responden untuk menjawab suatu pernyataan dengan

jawaban sangat setuju (SS), setuju (S), belum memutuskan atau netral (N), tidak setuju (T), dan sangat tidak setuju (ST). Untuk masingmasing jawaban mempunyai bobot nilai yang berbeda, adapun pemberian bobot nilai itu sendiri tergantung jenis pernyataan yang dapat dilihat pada tabel berikut [11].

Jenis Pernyataan	Bobot Pendapat				
	SS	S	N	T	ST
Positif	5	4	3	2	1
Negatif	1	2	3	4	5

Untuk mencari persentase dari masing-masing jawaban kuesioner digunakan rumus skala likert sebagai berikut :

$$\boldsymbol{P = \frac{S}{Skor\ Ideal} \times 100\%}$$

Keterangan rumus mencari nilai presentase dari jawaban kuesioner dapat dilihat pada tabel berikut.

Tabel 2.1 Keterangan Rumus Mencari Nilai Persentase Kuesioner

Nama	Keterangan
P	Nilai persentase yang dicari
S	Jumlah frekuensi dikalikan dengan skor yang di tetapkan jawaban
Skor Ideal	Nilai tertinggi dikalikan dengan jumlah sampel

2.2.16. Autodesk 3D Studio Max

3D Studio Max merupakan software graphics yang dibuat oleh Autodesk. Program ini memadukan graphics vektor dengan raster image sehingga objek yang dihasilkan mendekati realistis. 3D Studio Max mampu menghasilkan objek dalam bentuk gambar ataupun dalam bentuk file interaktif seperti gambar animasi yang disimpan dalam bentuk file *.avi atau *.mov [8].

2.2.17. Unity

Game Engines seperti Unity adalah alat penggerak dibalik permainan-permainan (*games*) yang kita ketahui dan gemari. Unity adalah salah satu alat pengembangan yang banyak digunakan dan digemari untuk pemngembangan game dan digunakan oleh semua orang mulai dari penghobi hingga studio-studio besar untuk mengembangkan permainan - permainan (*games*) serta memberikan pengalaman interaktif dalam web, dekstop, ponsel, dan konsol [8]. Unity *Game Engine* adalah *software* yang digunakan untuk membuat *video game* berbasis dua

atau tiga dimensi dan dapat digunakan secara gratis, selain untuk membuat *Game*, Unity 3D juga dapat digunakan untuk membuat konten yang interaktif lainnya seperti, visual arsitektur dan *real-time* 3D animasi, selain sebagai *game engine* Unity 3D juga dapat digunakan sebagai sebuah *editor* bagi *game* yang sudah ada [13]. Unity 3D dibuat dengan menggunakan bahasa perogram C++, tapi pengguna tidak perlu menggunakan bahasa C++ yang sulit, karena Unity 3D mendukung bahasa program lain seperti JavaScript, C#, dan Boo, Unity memiliki kemiripan dengan game engine lainnya seperti, Blender game engine, Virtools, gamestudio, adapaun kelebihan dari Unity 3D, Unity dapat dioperasikan pada platform Windows dan Mac Os dan dapat menghasilkan game untuk Windows, Mac, Linux, Wii, iPad, iPhone, google Android dan juga browser. Untuk browser, kita memerlukan sebuah plugin, yaitu Unity Web Player, sama halnya dengan Flash Player pada Browser. Game Unity 3D juga mendukung dalam pembuatan game untuk console game Xbox 360 dan PlayStation 3 [13].

2.2.18. Android

Android adalah sistem operasi berbasis Linux bagi telepon seluler seperti telepon pintar dan komputer tablet. Android juga menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk berbagai macam piranti gerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat piranti lunak untuk ponsel. kemudian dalam pengembangan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [14].

2.2.19. Versi Android

1. Android versi 1.1



Gambar 2.7 Logo Android 1.1

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, *voice search* (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email.

2. Android versi 1.5 (*Cupcake*)



Gambar 2.8 Logo Android *Cupcake*

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaharuan untuk penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke headset Bluetooth, animasi layar, dan keyboard pada layar yang dapat disesuaikan dengan sistem.

3. Android versi 1.6 (*Donut*)



Gambar 2.9 Logo Android *Donut*

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan indikator baterai dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang dintegrasikan; CDMA / EVDO, 802.1x, VPN, *Gestures*, dan *Text-to-speech engine*; kemampuan dial kontak; teknologi *text to change speech* (tidak tersedia pada semua ponsel; pengadaan resolusi VWGA.

4. Android versi 2.0/2.1 (*Eclair*)



Gambar 2.10 Logo Android *Eclair*

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan flash untuk kamera 3,2 MP, digital Zoom, dan Bluetooth 2.1.

Agar dapat bergerak cepat dalam persaingan perangkat generasi yang berikutnya, Google melakukan investasi dengan mengadakan kompetisi aplikasi mobile terbaik (killer apps – aplikasi unggulan). Kompetisi ini berhadiah $25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik.

Dengan semakin berkembangnya dan bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka dalam sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug. Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

5. Android versi 2.2 (Froyo: *Frozen Yoghurt*)



Gambar 2.11 Logo Android Froyo

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umum terhadap versi – versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, intergrasi V8 JavaScript engine yang dipakai Google Chrome yang mempercepat kemampuan *rendering* pada *browser*, pemasangan aplikasi dalam SD Card, kemampuan WiFi *Hotspot portabel*, dan kemampuan auto update dalam aplikasi Android Market.

6. Android versi 2.3 (*Gingerbread*)



Gambar 2.12 Logo Android *Gingerbread*

Pada 6 Desember 2010, Android versi 2.3 (*Gingerbread*) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (*gaming*), peningkatan fungsi copy paste, layar antar muka (*User Interface*) yang sudah didesain ulang, dukungan format video VP8 dan WebM, efek audio baru *(reverb, equalization, headphone virtualization,* dan *bass boost*), dukungan kemampuan *Near Field Communication* (NFC), dan dukungan jumlah kamera yang lebih dari satu.

7. Android versi 3.0/3.1 (*Honeycomb*)



Gambar 2.13 Logo Android *Honeycomb*

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada *Honeycomb* juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi processor dan juga akselerasi perangkat keras (*hardware*) untuk grafis. Tablet pertama kali yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom. Perangkat tablet tersebut bernama Eee Pad Transformer yang merupakan produk dari Asus yang masuk ke pasar Indonesia pada Mei 2011.

8. Android versi 4.0 (ICS: *Ice Cream Sandwich*)



Gambar 2.14 Logo Android *Ice Cream Sandwich*

Pada tanggal 19 Oktober 2011, diperkenalkannya Android versi 4.0 yang membawa fitur *Honeycomb* untuk *smartphone*, menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, kontak jaringan sosial terpadu, perangkat tambahan fotografi, pencarian email secara *offline*, dan berbagi informasi dengan menggunakan NFC. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus.

9. Android versi 4.1 (*Jelly Bean*)



Gambar 2.15 Logo Android *Jelly Bean*

Android Jelly Bean yaang diluncurkan pada acara Google I/O membawa sejumlah keunggulan dan fitur baru. Adapun penambahan fitur baru iantaranya yaitu meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencarian melalui *Voice Search* yang lebih cepat. Google Now yang juga menjadi bagian yang diperbarui pun tak ketinggalan. Google Now memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi Android Jelly Bean 4.1 muncul pertama kali dalam produk tablet Asus, yakni Google Nexus 7.