BAB II
LANDASAN TEORI

2.1 *E-government*

E-government memiliki arti yang berbeda dengan definisi *e-commerce* maupun *e-business* yang cenderung universal, *e-government* sering digambarkan atau dideskripsikan secara cukup beragam oleh masing-masing individu atau komunitas.

2.1.1 Pengertian *E-government*

E-government, sebagai sebuah konsep memiliki prinsip-prinsip dasar yang universal, tetapi pengertian maupun penerapannya di sebuah negara tidak dapat dipisahkan dari sejarah, budaya, pendidikan, pandangan politik, kondisi ekonomi masing-masing negara. *E-government* didefinisikan sebagai suatu mekanisme interaksi baru antara pemerintah dengan masyarakat dan pihak-pihak lain yang berkepentingan, dimana pemanfaatan teknologi informasi dan teknologi komunikasi dengan tujuan meningkatkan kualitas pelayanan publik (Indrajit,2002).

E-government memiliki tiga karakteristik yang sangat penting, yaitu sebagai berikut:

1. Merupakan suatu mekanisme interaksi baru (modern) antara pemerintah dengan masyarakat dan kalangan lain yang berkepentingan (stakeholder).
2. Melibatkan penggunaan teknologi informasi (terutama pengguna internet).
3. Memperbaiki mutu atau kualitas pelayanan publik.

Secara umum, visi dan misi *e-government* dilandasi pada empat prinsip-prinsip dasar *e-government*, yaitu :

1. Memberikan perhatian penuh pada jenis-jenis pelayanan publik, dengan prioritas sebagai berikut:
    a. Memiliki volume transaksi yang besar dan melibatkan banyak sekali sumber daya manusia.
    b. Membutuhkan interaksi dua arah antara pemerintah dengan masyarakat.
    c. Memungkinkan terjadinya kerja-sama pemerintah dengan swasta maupun LSM dan Perguruan Tinggi.
2. Membangun lingkungan yang kompetitif, dimana sektor swasta maupun LSM dapat berperan dalam hal pelayanan publik. Bahkan sektor swasta dan LSM dapat bersaing dengan pemerintah dan dapat melayani dengan lebih baik.
3. Memberikan penghargaan pada inovasi dan memberi ruang kesempatan pada kesalahan.
4. Memusatkan pada pencapaian efisiensi, yang dapat dinilai dengan besarnya manfaat dan pemasukan anggaran dari penggunaan e-government.

2.1.2 Manfaat *E-government*

Secara jelas, dua negara besar yang terdepan dalam mengimplementasikan konsep *e-government*, yaitu Amerika dan Inggris melalui Al Gore dan Tony Blair,

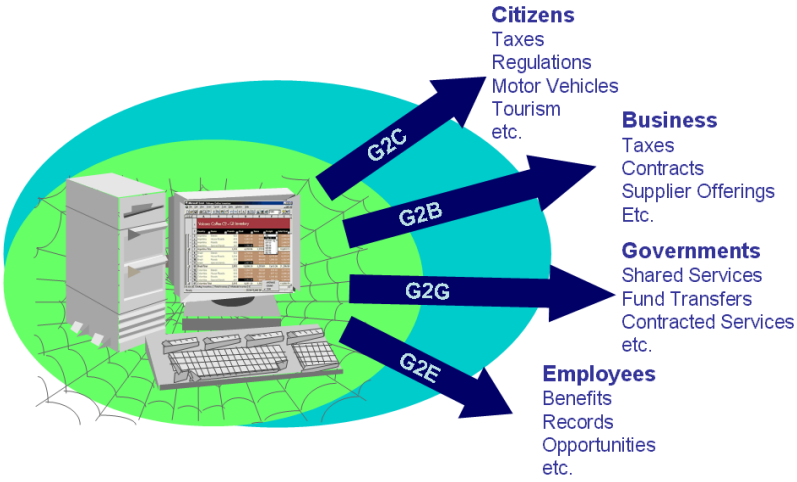
telah secara jelas dan terperinci menggambarkan manfaat yang diperoleh dengan diterapkannya konsep *e-government* bagi suatu negara, antara lain:

1. Memperbaiki kualitas pelayanan pemerintah kepada para penggunanya (masyarakat, kalangan bisnis, dan industri) terutama dalam hal kinerja efektivitas dan efisiensi di berbagai bidang kehidupan bernegara.
2. Meningkatkan transparansi, kontrol, dan akuntabilitas penyelenggaraan pemerintahan dalam rangka penerapan konsep *Good Corporate Governance.*
3. Mengurangi secara signifikan total biaya penggunaistrasi, relasi, dan interaksi yang dikeluarkan pemerintah maupun penggunanya untuk keperluan aktivitas sehari-hari.
4. Memberikan peluang bagi pemerintah untuk mendapatkan sumber-sumber pendapatan baru melalui interaksinya dengan pihak-pihak yang berkepentingan.
5. Menciptakan suatu lingkungan masyarakat baru yang dapat secara cepat dan tepat menjawab berbagai permasalahan yang dihadapi sejalan dengan berbagai perubahan global dan trend yang ada.
6. Memberdayakan masyarakat dan pihak-pihak lain sebagai mitra pemerintah dalam proses pengambilan berbagai kebijakan publik secara merata dan demokratis.

2.1.3 Empat Tipe *E-goverment*

Seperti halnya di dalam dunia aplikasi *e-commerce* yang kerap diklasifikasikan menjadi dua jenis, yaitu tipe B-to-B dan B-to-C, di dalam konsep

e-government dikenal pula empat jenis klasifikasi, yaitu: G-to-C, G-to-B, G-to-G, dan G-to-E.



Gambar 2.1 Tipe Relasi *E-government*

1. *Government to Citizens*

   Tipe G-to-C ini merupakan aplikasi *e-government* yang paling umum, yaitu dimana pemerintah membangun dan menerapkan berbagai portofolio teknologi informasi dengan tujuan utama untuk memperbaiki hubungan interaksi dengan masyarakat (rakyat). Dengan kata lain, tujuan utama dari dibangunnya aplikasi *e-government* bertipe G-to-C adalah untuk mendekatkan pemerintah dengan rakyatnya melalui kanal-kanal akses yang beragam agar masyarakat dapat dengan mudah menjangkau pemerintahnya untuk pemenuhan berbagai kebutuhan pelayanan sehari-hari.

2. *Government to Business*

Salah satu tugas utama dari sebuah pemerintahan adalah membentuk sebuah lingkungan bisnis yang kondusif agar roda perekenomian sebuah negara dapat berjalan sebagaimana mestinya. Dalam melakukan aktivitas sehari-harinya, entiti bisnis semacam perusahaan swasta membutuhkan banyak sekali data dan informasi yang dimiliki oleh pemerintah. Disamping itu, yang bersangkutan juga harus berinteraksi dengan berbagai lembaga kenegaraan karena berkaitan dengan hak dan kewajiban organisasinya sebagai sebuah entiti berorientasi profit. Diperlukannya relasi yang baik antara pemerintah dengan kalangan bisnis tidak saja bertujuan untuk memperlancar para praktisi bisnis dalam menjalankan roda perusahaannya, namun lebih jauh lagi banyak hal yang dapat menguntungkan pemerintah jika terjadi relasi interaksi yang baik dan efektif dengan industri swasta.

3. *Government to Governments*

Di era globalisasi ini terlihat jelas adanya kebutuhan bagi negara-negara untuk saling berkomunikasi secara lebih intens dari hari ke hari. Kebutuhan untuk berinteraksi antar satu pemerintah dengan pemerintah setiap harinya tidak hanya berkisar pada hal-hal yang berbau diplomasi semata, namun lebih jauh lagi untuk memperlancar kerjasama antar negara dan kerjasama antar entiti-entiti negara (masyarakat, industri, perusahaan, dan lain-lain) dalam melakukan hal-hal yang berkaitan dengan

penggunaistrasi perdagangan, proses-proses politik, mekanisme hubungan sosial dan budaya, dan lain sebagainya.

4. *Government to Employees*

Pada akhirnya, aplikasi *e-government* juga diperuntukkan untuk meningkatkan kinerja dan kesejahteraan para pegawai negeri atau karyawan pemerintahan yang bekerja di sejumlah institusi sebagai pelayan masyarakat.

2.2 Interaksi Manusia dan Komputer

Interaksi manusia dan komputer atau *human computer interaction* memiliki pengertian sebagai berikut :

Human-computer interaction is the study, planning, and design of how people and computers work together so that a person's needs are satisfied in the most effective way (Galitz, 2007).

Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them (Hewett et al, 1996).

Human Computer Interaction atau interaksi manusia dan komputer secara umum adalah ilmu yang mempelajari suatu interaksi yang terjadi antara manusia dan komputer.

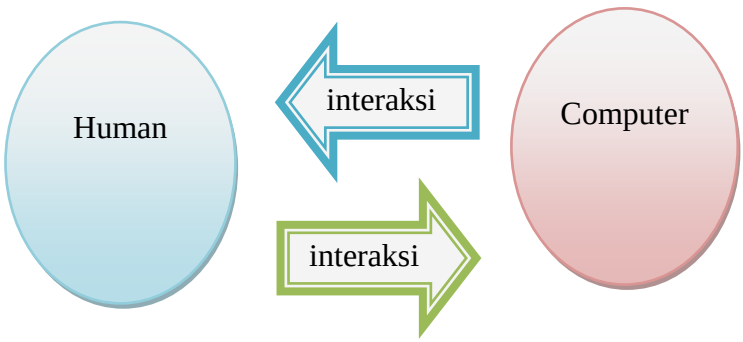
Antara manusia dengan manusia dapat terjadi suatu interaksi, yang beragam cara berinteraksinya, salah satunya yakni dengan cara berdialog, menggunakan suatu bahasa yang di kenal antar manusia atau melalui simbol-simbol yang diciptakan oleh manusia itu sendiri.

Sedangkan interaksi antara manusia dengan komputer dapat berdialog atau berinteraksi dengan mengguakan tampilan antarmuka pengguna atau *user interface.* Berikut terdapat perbedaan antara manusia dan komputer yang tampak pada Tabel 2.1.

Tabel 2.1 Pebedaan Manusia dan Komputer

Manusia	Komputer
Fleksibel & mampu beradaptasi,	Tidak fleksibel/tidak mampu beradaptasi,
Dapat belajar bagaimana bekerja di lingkungan yang baru,	*Input* harus dalam format yang jelas & *output* harus didefinisikan sebelumnya,
Estimasi, intuisi, kreativitas,	Tidak dapat belajar,
Pengenalan pola,	Dapat didesain ulang,
Pengetahuan dunia,	Kalkulasi akurat, konsisten,
Kesalahan manusiawi.	Pengolahan data,
Sering melakukan kesalahan	Bebas dari kesalahan.

Adapun hubungan antara manusia dan komputer tampak pada Gambar 2.2 yakni sebagai berikut:



Gambar 2.2 Hubungan Antara Manusia dan Komputer

Berikut ini, merupakan penjelasan mengenai Gambar 2.2, yakni sebagai berikut:

1. Pengguna (*user*)

   Pengguna secara individu, suatu grup dari pengguna yang bekerja sama, atau sekelompok pengguna dalam organisasi; dimana masing-masing berhubungan dengan bagian sama dari suatu kerjaan atau proses. Pengguna berusaha menyelesaikan pekerjaannya dengan menggunakan bantuan teknologi (komputer)

2. Komputer

   Komputer adalah suatu teknologi yang digunakan untuk mengontrol suatu proses atau sebuah system. Jenisnya dapat berupa komputer yang berskala kecil /desktop sampai keskala besar seperti komputer super. Komputer dapat juga berupa suatu teknologi yang dapat ditempelkan/dicangkokkan (*embedded system*) ke suatu benda.

3. Interaksi

   Setiap komunikasi antara pengguna (*user*) dengan komputer, secara langsung maupun tidak langsung. Interaksi tersebut melibatkan suatu *dialog* dengan suatu umpan-balik (*feedback*) dan kontrol hasil kinerja dari suatu kerja. Interaksi yang bagus antar pengguna dengan pemberi dapat memberikan suatu kemudahan dalam melakukan pekerjaan sehingga dapat menghasilkan suatu hasil yang diinginkan tepat waktu.

2.3 *User Interface* (UI)

Pengguna sering menilai sistem dari *interface*, bukan dari fungsinya. Jika desain *user interface*nya yang buruk, maka itu sering jadi alasan untuk tidak menggunakan sebuah aplikasi. Selain itu *interface* yang buruk dapat menyebabkan pengguna membuat kesalahan yang fatal. Pada subbab berikutnya akan dijelaskan mengenai definisi, macam-macam UI, komponen antarmuka dan prinsip-prinsip umum *user interface.*

2.3.1 Definisi *User Interface* (UI)

User interface (UI) atau antarmuka pemakai merupakan bagian dari komputer dan perangkat lunaknya yang dapat dilihat, didengar, disentuh, dan diajak bicara, baik secara langsung maupun dengan proses pemahaman tertentu.

User interface yang baik adalah *user interface* yang tidak disadari, dan *user interface* yang memungkinkan pengguna fokus pada informasi dan *task* tanpa perlu mengetahui mekanisme untuk menampilkan informasi dan melakukan *task* tersebut.

2.3.2 Macam-macam *User Interface* (UI)

User interface (UI) terdiri dari dua jenis, diantaranya adalah sebagai berikut :

1. *Graphical User Interface* (GUI)

   Graphical user interface biasanya menggunakan unsur-unsur multimedia seperti gambar, suara dan video agar dapat berkomunikasi dengan pengguna.

2. *Text Based*

Text Based biasanya menggunakan sintaks atau rumus yang sudah ditentukan untuk memberikan perintah.

2.3.3 Komponen Antarmuka Pengguna

Antarmuka pengguna atau *user interface* (UI) terdiri dari empat komponen, yakni sebagai berikut:

1. Model Pengguna

   Model pengguna merupakan model konseptual yang diinginkan pengguna dalam memanipulasi informasi dan proses yang akan dibuat dalam aplikasinya.

2. Bahasa perintah

   Bahasa perintah merupakan piranti yang digunakan untuk memanipulasi model.

3. Umpan balik

   Umpan balik merupakan kemampuan program untuk membantu pengguna dalam mengoperasikan program yang telah dibuat.

4. Penampilan informasi

   Penampilan informasi merupakan petunjuk status informasi atau program ketika pengguna melakukan tindakan, perlu dirancang pesan-pesan yang efektif untuk membantu *user* dalam menggunakan aplikasi.

2.3.4 Prinsip-prinsip Umum Perancangan *User Interface*

Ada beberapa prinsip umum yang melatarbelakangi perancangan *user interface*, yakni sebagai berikut:

1. *User Compability*

   User interface yang akan dibangun harus memenuhi prinsip *user compability*, karena semua *user* adalah tidak sama dan semua *user* tidak seperti *developer*. Para pembangun sistem harus paham mengenai pengetahuan psikologi dasar dari *user* tersebut. Hal ini juga berarti bahwa *user interface* harus memiliki tampilan yang sesuai dengan tipikal dari tiap *user*, karena setiap *user* memiliki kebutuhan tampilan yang berbeda-beda. Misalnya, jika membangun aplikasi yang dibuat untuk anak-anak, maka sebaiknya jangan menggunakan istilah atau tampilan untuk orang dewasa.

2. *Product Compability*

   Kompatibilitas antar produk harus diperhatikan dan dipertahankan, karena seringkali sebuah aplikasi menghasilkan hasil yang berbeda dengan sistem manual yang ada. Hal tersebut tentu saja tidak diharapkan oleh perusahaan yang akan membuat aplikasi tersebut.

3. *Task Compatibility*

   Task Compatibility berarti struktur dan aliran sistem harus sesuai dan mendukung tugas *user*, yakni fungsionalitas dari *task* atau tugas yang dibutuhkan oleh masing-masing pengguna harus sesuai dengan antarmuka (*interface*) yang akan dibuat.

4. *Work Flow Compatibility*

Antarmuka yang akan dibuat sebaiknya memenuhi prinsip *work flow compatibility*, yakni sistem yang akan dibuat harus diorganisasikan dengan baik sehingga dapat memfasilitasi setiap transisi antar tugas yang dilakukan oleh *user*.

5. *Consistency*

Dalam membuat perancangan antarmuka, pembangun sistem harus konsisten, maksudnya agar membuat para pengguna atau *user* berfikir dengan menganalogikan dan memprediksi bagaimana melakukan sesuatu yang belum pernah dilakukan sebelumnya.

6. *Familiarity*

Antarmuka sebisa mungkin didesain sesuai dengan antarmuka pada umumnya, mulai dari segi tata letak, model dan sebagainya. Hal ini dapat membantu *user* dalam berinteraksi dengan sistem melalui antarmuka yang familiar bagi *user*. Misalnya, *icon* disket seringkali digunakan sebagai *icon* untuk menyimpan data.

7. *Simplicity*

Kesederhanaan atau *simplicity* perlu diperhatikan pada saat membangun antarmuka. Tidak selamanya antarmuka yang memiliki menu yang banyak adalah antarmuka yang baik. Justru, pada umumnya sering terjadi kesalahan pada perancangan *interface* yang berusaha untuk menyediakan semua fungsionalitas. Pembangun sistem cukup menyediakan default saja dalam merancang antarmuka yang akan dibuat.

8. Direct Manipulation

Direct Manipulation dimaksudkan agar *user* dapat melihat secara langsung aksi yang telah dilakukannya pada objek yang terlihat.

9. Control

Pengguna sebaiknya diberikan hak untuk mengontrol secara penuh sistem yang akan dibuat, tentunya dengan sebaik mungkin dan jangan sampai merusak sistem. Biasanya, pengguna tidak suka jika terlalu banyak aturan, hal itu dapat membuat frustasi dan demoralisasi bagi pengguna, jika mereka merasa dikontrol oleh mesin.

10. What You See Is What You Get (WYIWYG)

WYIWYG maksudnya adalah apa yang dilihat, maka itulah yang akan didapat. Misalnya, adanya korespondensi satu ke satu antara informasi di layar dengan informasi yang dicetak atau berupa file.

11. Flexibility

Flexibility adalah mengijinkan makin banyak kontrol *user* dan mengakomodir skill *user* yang bervariasi. Misalnya masalah *tools*/alat yang bisa digunakan oleh *user*, jangan terpaku pada keyboard saja atau mouse saja, bisa saja keduanya digunakan.

12. Responsiveness

Komputer harus selalu merespon dengan segera setiap input dari *user*. Misalnya, dengan menggunakan progress bar atau mungkin menggunakan animasi *icon* waktu.

13. Invisible Technology

User sebaiknya mengetahui sesedikit mungkin detil teknis bagaimana sistem diimplementasikan.

14. Robustness

Robustness artinya handal, jadi sistem sebaiknya mentolerir kesalahan manusia yang umum dan tidak dapat dihindar. Crash system harus diminimalisir, dan menyediakan recovery yang mudah dipahami jika terjadi crash.

15. Protection

Sistem seharusnya memproteksi dari hasil-hasil yang menyebabkan 'bencana' karena kesalahan umum manusia. Misalnya dengan menyediakan fitur back atau undo.

16. Ease of Learning

Ease of Learning adalah system yang mudah dipelajari bagi pemula tanpa harus ada pelatihan, tanpa harus baca tutorial dan alat bantu lainnya.

17. Ease of Use

Ease of Use merupakan sistem yang mudah digunakan terutama bagi para expert.

2.3.5 *User Interface* untuk Aplikasi berbasis web [5]

Dalam mendesain antarmuka pengguna aplikasi suatu sistem yang berbasis web, ada beberapa atribut yang perlu diperhatikan untuk membangun website dengan halaman yang efektif, diantaranya:

1. ***Textual content***

Content yang ditampilkan dalam sebuah website memiliki nilai yang lebih penting dibandingkan desain dari website itu sendiri. Informasi yang sesuai merupakan hal yang sangat penting dalam sebuah website. Ada beberapa hal yang perlu diperhatikan untuk menampilkan sebuah informasi, yaitu :

a. Kemampuan membaca pada monitor lebih lambat daripada kertas
b. User cenderung hanya membaca header, highlights dan paragraf tertentu

2. ***Graphic design***

Tampilan grafis harus memperhatikan masalah konsistensi, susunan antara teks, gambar, kontras, warna dan fungsionalitas dari gambar yang ditampilkan. Berikut ini merupakan hal-hal yang perlu diperhatikan dalam mendesain tampilan grafis, yakni :

a. Pemilihan format gambar yang sesuai dan pertimbangan masalah ukuran file gambar yang akan mempengaruhi kecepatan akses dan waktu tunggu (*delay*)
b. Menyesuaikan tampilan grafis dengan dimensi halaman website yang akan ditampilkan di layar
c. Apabila monitor menampilkan resolusi 640 x 480 pixel maka dimensi rata-rata yang dapat digunakan untuk tampilan halaman website pada sebuah web browser yakni 595 x 295 pixel.

d. Dimensi umum yang digunakan untuk halaman website yaitu 800 x 640 pixel.
e. Usahakan setiap halaman hanya menggunakan satu layar, hindari scrolling layar jika memungkinkan.
f. Jika tidak memungkinkan, scrolling hanya sesuai untuk content, hindari scrolling untuk halaman navigasi. Gunakan scrolling layar ke atas/bawah, jangan ke samping (kiri/kanan).
g. Letakkan link yang penting pada bagian atas halaman website.
h. Meskipun monitor telah mampu menampilkan jutaan warna, namun hanya 216 warna yang biasa digunakan untuk aktivitas browsing.
i. Pilih format file gambar yang sesuai, misalnya GIF, JPEG, dan PNG.

3. *Navigation*

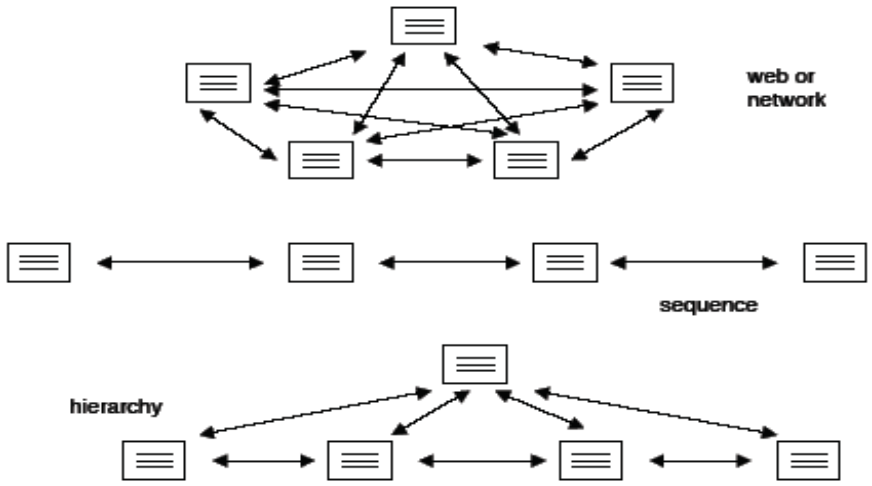
Navigasi merupakan elemen kritis dalam menentukan efektivitas antarmuka sebuah website. Navigasi harus memberikan penjelasan tentang struktur informasi pada website, dengan menyediakan:

a. Daftar isi (*site map*)
b. *Index*
c. *Navigation bar*
d. Kemampuan *Searching*

4. *Structure*

Struktur website yang harus diperhatikan dalam membangun aplikasi berbasis web, antara lain *connectivity and compactness,branching*

factor, page length dan number of links. Macam-macam struktur pada aplikasi berbasis web digambarkan pada Gambar 2.3.



Gambar 2.3 Macam- macam struktur

Homepage merupakan halaman yang terpenting dalam website. Tampilan homepage sebaiknya diberikan tampilan yang menarik dan diberi penjelasan tentang informasi apa saja yang dapat ditemukan pada website.

Pada homepage, sebaiknya ditampilkan beberapa real content maupun berita aktual. Selain itu, jika link yang dibuat pada homepage menggunakan gambar, maka berikan pula label berupa teks, serta tampilkan informasi dinamis pada homepage.

5. Link

Link pada website memiliki beberapa kriteria, yakni sebagai berikut :

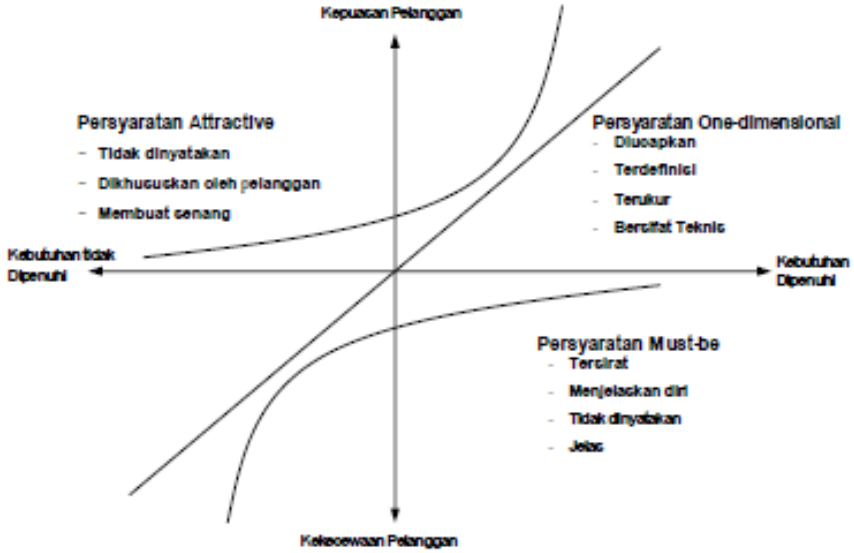
a. Link sama seperti gaya dialog buttons, harus memberikan penjelasan tentang apa yang dapat dilakukan pada halaman berikutnya.

b. Sebuah link yang baik berpedoman pada beberapa hal berikut ini, yakni :
    a) Bagaimana user dapat memprediksi tujuan sebuah link.
    b) Bagaimana user dapat membedakan antara sebuah link dengan link lainnya yang memiliki kemiripan.
    c) Sesuai tidaknya content yang ditampilkan oleh link.
c. Pastikan semua link pada website dapat bekerja sesuai dengan fungsi masing-masing.
d. Hati-hati terhadap penggunaan kata "*here*"/"disini" pada sebuah link.

2.4 Metode Kano [11]

Dalam merencanakan suatu produk atau layanan, kita dapat membuat suatu daftar kebutuhan yang dapat membuat produk atau layanan tersebut sebisanya memuaskan calon pelanggan (*customer*). Menemui secara langsung pelanggan yang sudah ada atau mereka yang berpotensi untuk menjadi pelanggan, adalah cara yang baik untuk memperoleh masukan tentang hal apa saja yang harus ada di dalam daftar keperluan dari pelanggan yang potensial tadi. Untuk mengetahuinya, harus dilakukan penyelidikan terhadap setiap daftar kebutuhan yang dibuat sedetail mungkin untuk lebih memahami persyaratan apa yang benar-benar perlu ada dalam produk atau layanan akhir.

Salah satu metode yang dapat digunakan untuk menganalisis tersebut adalah metode kano yang ditemukan oleh Propesor Noriaki kano dari Tokyo Rika University. Gambaran tentang metode terhadap kepuasan pelanggan terlihat pada Gambar 2.4, yaitu:



Gambar 2.4 Penggolongan Tipe Metode Kano

Berikut ini, merupakan penjelasan dari Gambar 2.4, yakni model Kano terhadap kepuasan pelanggan :

1. M : **Must Be requirements** yaitu fungsi atau fitur yang disediakan oleh sebuah produk harus ada. Hal ini dapat diartikan jika fitur ada maka tidak meningkatkan kepuasan pengguna, tetapi jika tidak ada maka pengguna tidak puas.
2. O : **One-Dimensional Requirenments** yaitu kondisi ini terjadi jika kepuasan konsumen terhadap suatu produk proporsional terhadap kinerja suatu produk .
3. A : **Attractive Requirenments** merupakan kondisi yang terpenuhi jika kehadiran sebuah produk akan meningkatkan kepuasan konsumen tetapi ketidakhadiran sebuah produk tidak membuat konsumen tidak puas.
4. I : **Indifferent Requirenments** merupakan kondisi yang terpenuhi jika fitur yang disediakan tidak dipedulikan kehadirannya oleh pengguna.

5. **Reversal** : kondisi dimana sebuah fitur yang diperkirakan mendapat jawaban positif tetapi ternyata mendapatkan jawaban negativ dari pengguna
6. **Questionable requirements**: kondisi terjadi jika pengguna menjawab kuesioner dengan jawaban yang sama pada pertanyaan yang berbanding terbalik.

Langkah-langkah pembuatan kuesioner dengan meggunakan metode kano yakni sebagai berikut :

1. Tetapkan item-item fitur/fungsionalitas yang akan ditanyakan.
2. Buatlah pertanyaan yang bersifat *functional* dan *disfunctional* untuk tiap item fitur, kemudian acak penempatan pertanyaannya.
3. Tes questionernya terlebih dahulu,
4. Jika membingungkan perbaiki terlebih dahulu, jika sudah ok maka disebarkan pada responden.
5. Pengolahan hasil questioner dengan mencerminkan pada Kano *evaluation table.*
6. Untuk tiap item cari requirement tertinggi, buat persentase nya dan hitung *extent of satisfaction/dissatisfaction* dengan rumus sebagai berikut:

Extent of satisfaction : $\boxed{\frac{A+O}{A+O+M+I}}$

Extent of dissatisfaction : $\boxed{\frac{O+M}{(A+O+M+I)\ x\ (-1)}}$

7. Dan langkah terakhir, yaitu membuat kesimpulan dan analisis hasil kuesioner.

Keuntungan dari pengklasifkasian kebutuhan pelanggan dengan menggunakan metode Kano ini diantaranya adalah:

1. Prioritas pada pengembangan produk.

   Sebagai contoh, tidak banyak keuntungannya jika kita menginvestasikan pada perbaikan persyaratan *must-be* yang memang sudah ada pada tingkat kepuasan, tetapi lebih baik meningkatkan persyaratan *one dimensional* atau *attractive* yang memang jelas berpengaruh pada kualitas produk dan juga mempengaruhi tingkat kepuasan pelanggan.

2. Syarat produk lebih dimengerti.

   Kriteria produk yang memiliki pengaruh terbesar pada kepuasan pelanggan dapat diidentifikasi. Penggolongan persyaratan produk ke dalam dimensi *must-be*, *one dimensional*, dan *attractive* dapat digunakan untuk lebih fokus pada sesuatu.

3. Kepuasan pelanggan menggunakan model Kano *dapat secara optimal dikombinasikan dengan penyebaran fungsi kualitas*.

   Suatu prasyarat mengidentifikasi kebutuhan, hirarki dan prioritas pelanggan (Griffin/Hauser, 1993). Model Kano digunakan untuk menetapkan pentingnya fitur produk untuk kepuasan pelanggan dan itu dapat menciptakan prasarat yang optimal pada kegiatan pengembangan produk berorientasi proses.

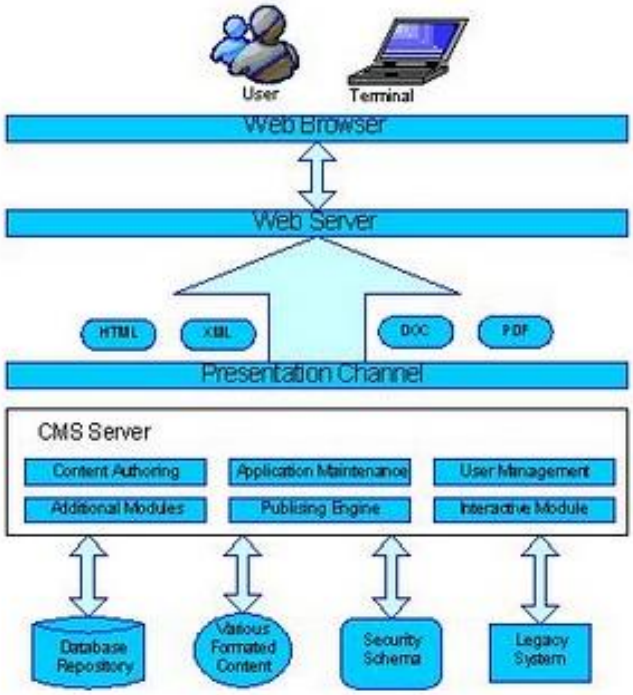
2.5 *Content Management System* (CMS)

CMS merupakan singkatan dari *content management system*, yaitu sebuah software yang dipergunakan untuk membangun sebuah website. Berikut ini akan dijelaskan engenai definisi dan fungsi dari CMS.

2.5.1 Definisi CMS

Content Management System (CMS) adalah suatu sistem yang digunakan untuk mengelola dan memfasilitasi proses pembuatan, pembaharuan, dan publikasi content secara bersama (*colaboration content management*). *Content* mengacu pada informasi dalam bentuk teks, grafik, gambar, maupun dalam format-format lain yang perlu dikelola dengan tujuan memudahkan pembuatan, pembaharuan, distribusi, pencarian, analisis, dan meningkatkan fleksibilitas untuk ditramformasikan dalam bentuk lain. Terminologi CMS sendiri cukup luas, diantararanya mencakup *software* aplikasi, database, arsip, *workflow*, dan alat bantu lain yang dapat dikelola sebagai bagian dari mekanisme jaringan informasi suatu perusahaan maupun global.

Jadi, *Content Management System* (CMS) adalah sebuah sistem yang dapat membuat, mengatur, mendistribusikan, mempublikasikan dan menjaga informasi perusahaan atau institusi, dimana pengguna tidak harus mengetahui tag HTML untuk menjalankanya. Secara umum, arsitektur CMS digambarkan seperti Gambar 2.5.



Gambar 2.5 Arsitektur CMS

2.5.2 Fungsi CMS [9]

Content Management System (CMS) mempunyai beberapa fungsi yakni sebagai berikut :

a. Content Making

Pembuatan content dengan menggunakan CMS sangat mudah, karena bisa dilakukan oleh siapapun tanpa harus mengetahui dasar HTML. Kebanyakan, saat ini CMS penulisan contentnya dikembangkan dilingkungan berbasis web, dengan implementasi yang lebih sederhana serta mengijinkan untuk diupdate tanpa harus meghapus dulu.

b. Content Management

Satu halaman telah dibuat, maka CMS akan menyimpan halaman tersebut pada tempat penyimpananya. Semuanya disimpan dengan link

yang terkait. Pusat penyimpanan juga menyediakan fasilitas untuk mengelola content dengan metode:

a) Menjaga traking semua *content* sehingga bisa mengerahui siapa dan kapan *content* tersebut diubah.
b) Memastikan bahwa setiap *user* hanya mempunyai privasi yang sesuai dengan hak aksesnya.
c) Mengitegrasikan dengan sistem yang sesuai dengan spesifikasi dari content tersebut.

c. Publication

Content Management System (CMS) sebaiknya memiliki fasilitas publish, sehingga dapat dikelola *content* apa saja yang bisa dipublikasikan.

d. Presentation

Content Management System (CMS) menyediakan penomoran untuk memindah kualitas dan efektifitas dirinya. Sebagai contoh CMS akan membangun navigasi dengan membaca struktur pengeluaran yang urut dari content yang teleh disimpannya. Sehingga akan membantu dalam mempresentasikan

2.6 Basis Data

Basis data merupakan sekumpulan data *persistence* yang saling terkait, menggambarkan suatu organisasi (*enterprise*). Sistem Manajemen Basis data (DBMS) suatu sistem yang mengelola data dan menyediakan data tersebut apabila dibutuhkan. Berikut dijelaskan mengenai konsep dasar basis data dan model intetitas basis data.

2.6.1 Konsep Dasar Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasi data-data tersebut.

Basis data terdiri dari kata basis dan data. Basis dapat diartikan gudang atau tempat bersarang dan data yang berarti representasi fakta dari dunia nyata yang mewakili objek seperti manusia, hewan, peristiwa, konsep dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya. Jadi, basis data juga dapat diartikan sebagai sekumpulan data persistence yang saling terkait yang menggambarkan suatu organisasi (enterprise).

Sistem basis data adalah suatu sistem informasi yang mengintegrasikan dari kumpulan data yang saling berhubungan satu dengan yang lainnya dan membuat tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi. Komponen utama Sistem Basis data yaitu sebagai berikut:

1. Data yang disimpan dalam basis data.
2. *Hardware* yang berupa *storage, processor, memory*.
3. *Software* yang terdiri dari DBMS, *Report-writer*, design arts dan lain-lain.
4. Penggguna yang terdiri dari:
    a. Pengguna Awam (*Naive User*)
    b. Pengguna Biasa (*Casual User*)
    c. Programmer
    d. Penguji

Tujuan utama pengelolaan data dalam sebuah basis data adalah agar dapat memperoleh menemukan kembali data (yang dicari) dengan mudah dan cepat. Di samping itu, pemanfaatan basis data untuk pengelolaan data, juga memiliki tujuan-tujuan lain.

Secara lebih lengkap, pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (*objektif*) seperti berikut ini:

1. Kecepatan dan kemudahan (speedy)
2. Efisiensi ruangan penyimpanan (space)
3. Keakuratan (accuracy)
4. Ketersediaan (availability)
5. Kelengkapan (completeness)
6. Keamanan (security)
7. Mengelola integritas (keterjaminan akurasi) data
8. Independensi data, yakni kekebalan aplikasi terhadap perubahan struktur penyimpanan dan teknik pengaksesan data.

2.6.2 Model Entitas

Peter Pin-Shan Chen, "*The Entity-Relationship Model-Toward a unifield view of data*", 1976. Model data E-R didasarkan pada persepsi bahwa dunia nyata merupakan sekumpulan dari sejumlah obyek dasar (entitas) dan relasi antar obyek-obyek data tersebut. Termasuk dalam kelompok object-based logical. Merupakan contoh kelas Extended atau semantic models. Komponen pada perancangan basis data secara konseptual terdiri dari:

1. Entitas (*Entity*)

Sebuah entitas adalah sebuah obyek yang dapat dibedakan dari obyek-obyek lainnya. Entitas dikelompokkan sebagai berikut:

a. Entitas konkrit / nyata dan entitas abstrak.

b. Entitas biasa (*Regular/Strong*) dan entitas lemah (*Weak*).

Himpunan entitas (*entity set*) adalah kumpulan sejumlah entitas yang memiliki tipe yang sama. Sebuah entitas memiliki sejumlah properti (atribut). Setiap atribut memiliki sekumpulan nilai yang diizinkan, yang disebut domain. Sebuah basis data mengandung sekumpulan himpunan entitas, yang masing-masingnya memiliki sejumlah entitas dari tipe yang sama.

2. Batasan (Kekangan)

Kekangan atau batasan digunakan untuk melindungi integritas data misalnya, untuk melindungi kesalahan pada saat pengisian data.

3. Atribut

Atribut merupakan karakteristik dari entity atau relationship, yang menyediakan penjelasan detail tentang entity atau relationship tersebut. Nilai dari suatu atribut merupakan suatu data aktual atau informasi yang disimpan pada suatu atribut di dalam suatu entity atau relationship.

a. Key Vs Non Key

Key adalah jika atribut tersebut mampu mewakili identitas keunikan dari suatu entitas. *Key* dapat dibangun dari satu atribut atau gabungan beberapa atribut. *Key* terbagi atas beberapa jenis, diantaranya :

a) Super Key

b) Primary Key

c) Candidate Key

d) Foreign Key

b. Atribut sederhana Vs Atribut Komposit

Atribut sederhana adalah atribut yang tidak perlu di-*breakdown* kembali. Atribut Komposit adalah atribut yang perlu di-*breakdown* kembali.

c. Atribut bernilai tunggal Vs Banyak

Atribut bernilai tunggal adalah atribut yang isinya hanya 1, sementara atribut bernilai banyak adalah atribut yang memiliki isi banyak.

d. Atribut Mandatory dan Non Mandatory.

Atribut Mandatory adalah atribut yang harus diisi tidak boleh kosong, sebaliknya Non mandatory boleh null (kosong).

e. Atribut Turunan

Atribut yang nilainya dapat diproleh dari atribut lain/proses lain.

4. Relasi (*Relationship*)

Sebuah relasi menggambarkan suatu asosiasi antar sejumlah entitas. Himpunan relasi (*Relationship set*) adalah kumpulan sejumlah relasi yang memiliki tipe yang sama. Jumlah entitas terlihat dalam 2 buah relasi disebut derajat. Kebanyakan relasi yang muncul adalah relasi binary, ada beberapa yang ternary, lebih dari itu sangat jarang. Fungsi sebuah

entitas di dalam relasi disebut peran (*role*). Sebuah relasi dapat memiliki atribut.

5. Kardinalitas Relasi (Derajat Hubungan)

   Beberapa kardinalittas relasi (derajat hubungan) yang dimungkinkan antar entitas, yaitu 1-1, 1 – N, N – 1, N – N/N – M.

6. Diagram E- R

   Diagram E-R menggambarkan struktur lojik keseluruhan basis data. Simbol–simbol yang digunakan, antara lain :

   a. Persegi empat, merepresentasikan himpunan entitas (untuk entitas lemah diberi garis ganda).
   b. Elips, merepresentasikan atribut.
   c. Wajik, merepresentasikan himpunan keterhubungan.
   d. Garis, menghubungkan simbol-simbol pada diagram.
   e. Label dari persegi empat, elips, dan wajik menunjukkan nama.
   f. Kardinalitas pemetaan dinyatakan dengan 2 cara :
      a) [*Korth*] garis berarah (1) dan garis tidak berarah (Banyak).
      b) [*Date*] menuliskan kardinalitasnya pada garis.
   g. Peran dapat dituliskan sebagai label dari garis.

7. Varian Entitas

   Entitas terdiri dari berbagai macam varian, diantaranya sebagai berikut:

   a. Entitas Kuat yaitu entitas yang keberadaannya tidak tergantung dari entitas lain.

b. Entitas Lemah yaitu entitas yang keberadaanya tergantung dari entitas lain.

c. *Sub Entitas* (Generalisasi dan Spesialisasi), Generalisasi dibentuk dengan metode *Bottom Up*, sementara Spesialisasi dengan metode *Top Down*.

8. Varian Relasi

Selain entitas yang memiliki beberapa varian, relasi ternyata juga terdiri dari berbagai macam varian, diantaranya sebagai berikut:

a. Relasi tunggal yaitu relasi yang dihasilkan dari 1 entity set.

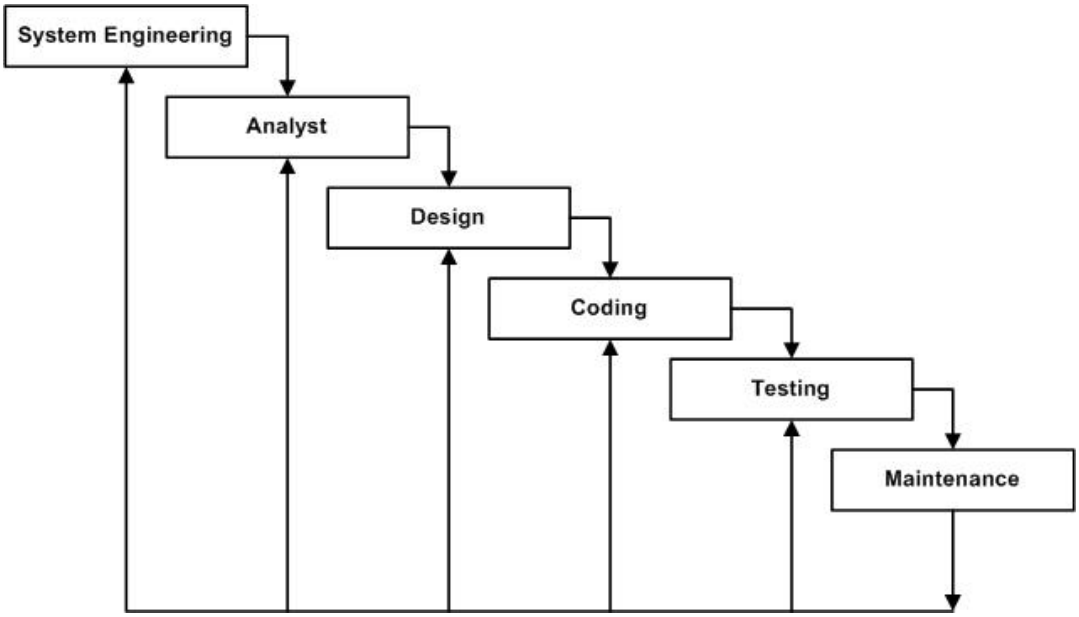
b. Relasi multi entitas yaitu relasi yang terjadi apabila memang ketiganya saling berelasi.

c. Relasi ganda merupakan relasi yang dihasilkan oleh 2 entity yang saling terhubung pada 1 proses.

d. Agregasi adalah relasi yang tergantung pada keberadaan relasi lain.

2.7 Linear Sequential Model/ Waterfall Model

Linear Sequential Model (Model Sekuensial Linier) atau sering disebut *Waterfall Model* (Model Air Terjun) merupakan paradigma rekayasa perangkat lunak yang paling tua dan paling banyak dipakai. Model ini mengusulkan sebuah pendekatan perkembangan perangkat lunak yang sistematik dan sekunsial yang dimulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain , kode, pengujian, dan pemeliharaan. Tahapan dari model sekuensial linear ditunjukkan pada Gambar 2.6.



Gambar 2.6 Metode *linier sequential/waterfall*

Berikut ini, merupakan penjelasan mengenai tahapan dari metode *linier sequential/waterfall* yang tampak pada Gambar 2.6.

a. Rekayasa Sistem (*System Engineering*)

Rekayasa sistem merupakan tahap untuk menetapkan berbagai kebutuhan dari semua elemen yang diperlukan sistem dan mengalokasikannya ke dalam pembentukan perangkat lunak.

b. Analisis Sistem (*System Analisis*)

Analisis sistem merupakan tahap untuk menganalisis hal-hal yang diperlukan dalam pelaksanaan pembuatan perangkat lunak.

c. Mendesain Sistem (*System Design*)

Desain sistem adalah tahap penerjemahan dari data yang telah dianalisis kedalam bentuk yang mudah dimengerti oleh pengguna aplikasi.

d. Koding Sistem (*System Coding*)

Mengkoding adalah tahap penerjemahan data atau pemecahan masalah yang telah dirancang kedalam bahasa pemrograman tertentu.

e. Pengujian Sistem (*System Testing*)

Pengujian sistem merupakan tahap pengujian terhadap perangkat lunak yang dibangun.

f. Pemeliharaan Sistem (*System Maintenance*)

Pemeliharaan sistem tahap akhir dimana suatu perangkat lunak yang sudah selesai dibagun dapat mengalami perubahan-perubahan atau penambahan sistem sesuai dengan permintaan pengguna aplikasi.

Selain itu, ada beberapa keunggulan dan kelemahan dari Model Sekuensial Linier yakni sebagai berikut:

1. Keunggulan

Keunggulan dari model sekuensial linear atau metode *waterfall* adalah sebagai berikut:

a. Mudah aplikasikan
b. Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

2. Kelemahan

Beberapa kelemahan dari model sekuensial linear atau metode *waterfall* yakni sebagai berikut:

a. Jarang sekali proyek pada dunia nyata mengikuti aliran sekuensial yang dianjurkan model karena model ini bisa melakukan itersi tidak

langsung. Hal ini berakibat ada perubahan yang diragukan pada saat proyek berjalan.

b. Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk megakomodasi ketidakpastian pada saat awal proyek.
c. Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyrk dilalui. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.
d. Pengembang sering malakukan penundaan yang tidak perlu karena anggota tim proyek harus menunggu tim lain untuk melengkapi tugas karena memiliki ketergantungan hal ini menyebabkan penggunaan waktu tidak efesien.

2.8 Pemodelan Rekayasa Perangkat Lunak

Pada bagian ini akan dijelaskan mengenai beberapa pemodelan rekayasa perangkat lunak yang akan digunakan untuk mendukung dalam perancangan sistem yang akan dibangun. Pemodelan rekayasa perangkat lunak meliputi Diagram alir dokumen (*flowmap*), Diagram Konteks, *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD) dan Kamus Data.

2.8.1 Diagram Alir Dokumen (*flowmap)*

Diagram alir dokumen (*flowmap*) adalah diagram yang menunjukkan alir di dalam program atau prosedur sistem secara logika. Diagram alir digunakan terutama sebagai alat bantu komunikasi dan untuk dokumentasi. Diagram alir dokumen (*flowmap*) digunakan untuk menggambarkan prosedur kerja secara fisik

dan mendefinisikan hubungan antara bagian (pelaku proses), proses (manual atau berbasis komputer) dan aliran data dalam bentuk dokumen keluaran dan masukan.

Flowmap bermanfaat untuk menganalisis pengendalian suatu sistem atau menganalisis pemisahan wewenang dan tanggung jawab. *Flowmap* dibuat secara berkolom dan masing-masing kolom mewakili suatu unit atau entitas. Hubungan antar unit yang sekaligus menunjukkan arus dokumen ditunjukkan dengan tanda panah.

2.8.2 Diagram Konteks

Konteks diagram merupakan proses yang mewakili proses dari semua sistem. Konteks diagram menggambarkan hubungan *input* dan *output* antara sistem dengan entitas luarnya. Diagram konteks adalah hubungan antara sistem dengan entitas luarnya yang berfungsi sebagai transformasi dari suatu proses yang melakukan transformasi data *input* menjadi data *output* .

Diagram konteks terdiri dari sebuah simbol proses tunggal yang menggambarkan sebuah sistem dan menunjukan data aliran utama untuk dan dari terminator. Diagram konteks dapat mendefinisikan jangkauan proses penurunan sistem informasi yaitu menentukan apa yang menjadi bagian dari sistem informasi dan apa yang tidak menjadi bagian sistem informasi.

2.8.3 Data Flow Diagram (DFD)

Menurut Jogiyanto, "*Diagram aliran data atau data flow diagram (DFD) adalah sebuah teknik grafis yang menggambarkan aliran informasi yang diaplikasikan pada saat bergerak dari input menjadi output. Data flow diagram*

merupakan gambaran secara logika dan tidak tergantung pada hardware, software, struktur data, ataupun organisasi file yang digunakan".

Data flow diagram digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau data tersebut akan disimpan. Beberapa simbol yang digunakan dalam Data Flow Diagram (DFD) antara lain:

1. Proses (Process)

   Proses adalah simbol pertama data *flow diagram*. Proses dilambangkan dengan lingkaran, dimana proses ini menunjukan bagian dari sistem yang 29 mengubah satu atau lebih *input* dan *output.* Nama proses dituliskan dengan satu kata, singkatan atau kalimat sederhana.

2. Aliran Data (*Flow*)

   Aliran Data digambarkan dengan tanda panah dari proses. Aliran data juga digunakan untuk menunjukan bagian – bagian informasi dari satu bagian ke bagian lain. Pembagian nama untuk aliran ini menunjukan sebuah arti untuk sebuah aliran. Untuk kebanyakan sistem yang dibuat, aliran data sebenarnya mengambarkan data yakni angka, huruf, pesan, dan macam - macam informasi lainnya.

3. Simpanan Data (*Storage)*

   Simpanan data digunakan sebagai penyimpanan bagi paket – paket data. Notasi penyimpanan data digambarkan dengan garis horizontal yang pararel. Simpanan data merupakan simpanan data dari data yang berupa

suatu file atau *database* di sistem komputer ataupun berupa arsip atau catatan manual. Nama dari simpanan data menunjukan nama filenya.

4. Kesatuan Luar (*External Entity*)

   Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Kesatuan luar merupakan kesatuan (*entity*) dilingkungan luar sistem yang dapat berupa orang, Organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau output dari sistem.

2.8.4 *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram atau ERD merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antara penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini relatif kompleks. ERD dapat digunakan untuk menguji model dengan mengabaikan proses yang harus dilakukan. ERD menggunakan sejumlah notasi dan symbol untuk menggambarkan struktur dan hubungan antar data. ERD terdiri atas elemen–elemen yaitu Entitas, Atribut, Relasi dan Kunci (*Key*). Berikut ini merupakan penjelasan dari elemen–elemen tersebut:

1. Entitas

   Entitas adalah suatu objek dasar atau individu yang mewakili sesuatu yang nyata dan dapat dibedakan dengan yang lainnya. Himpunan entitas adalah sekumpulan entitas dengan berbagai atribut yang sama.

2. Atribut

Atribut merupakan sifat–sifat atau karakteristik yang melekat dalam sebuah entitas.

3. Relasi

Relasi adalah hubungan di antara sejumlah entitas yang berasa dari sejumlah himpunan entitas yang berbeda. Relation atau hubungan dapat memiliki atribut. Sebuah kelas hubungan dapat melibatkan banyak kelas entitas.

4. Kunci (Key)

Kunci merupakan suatu atribut yang unik yang dapat digunakan untuk membedakan suatu entitas dengan lain dalam suatu himpunan entitas. Terdapat tiga macam kunci, yaitu :

a. Super key, adalah himpunan yang terdiri dari satu atau lebih yang dapat membedakan setiap baris data dengan unik pada sebuah table.
b. Candidate key, adalah himpunan atribut minimal yang dapat membedakan setiap baris data dengan unik dalam sebuah table.
c. Primary key, merupakan kunci yang paling unik, lebih ringkas, dan digunakan sebagai acuan.

2.8.5 Kamus Data

Kamus data menurut Jogiyanto, ”merupakan katalog data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi”. Adapun elemen-elemen yang disebut dengan item data tersebut, dapat digambarkan dengan menyebutkan nama

dari item-item datanya. Untuk menunjukan informasi tambahan di dalam kamus data digunakan notasi yang dijelaskan pada Tabel 2.2.

Tabel 2.2 Notasi Kamus Data

Notasi	Arti
=	Terbentuk dari
+	And
[ ]	Salah satu dari elemen-elemen data di dalam kurung braket ini.
$\{ \}^n$	Iterasi (pengulangan ke-n)
( )	Data optional (elemen data dalam kurung paratheis sifatnya optional, dapat ada dan tidak ada)
**	Keterangan di dalam tanda ini adalah komentar.

2.9 Perangkat Lunak Pendukung

Perangkat lunak pendukung yang digunakan untuk membangun aplikasi yang memodelkan *interface* situs pemerintahan (*e-government*) diantaranya Adobe Dreamwever CS3 sebagai *tools* pembangunnya, MySQL sebagai DBMSnya dan WampServer 2.0i sebagai server lokalnya.

2.9.1 Adobe Dreamweaver CS3

Adobe Dreamweaver merupakan program penyunting halaman web keluaran Adobe Systems yang dulu dikenal sebagai Macromedia Dreamweaver keluaran Macromedia. Program ini banyak digunakan oleh pengembang web karena fitur-fiturnya yang menarik dan kemudahan penggunaannya. Versi terakhir Macromedia Dreamweaver sebelum Macromedia dibeli oleh Adobe Systems yaitu versi 8. Versi terakhir Dreamweaver keluaran Adobe Systems adalah Adobe

Dreamweaver CS5 yang ada dalam Adobe Creative Suite 5. Tapi *tools* yang akan digunakan adalah adobe dreamweaver CS3. Logo Adobe dreamweaver tampak pada Gambar 2.3.



Tabel 2.3 Logo Adobe Dreamweaver

2.9.2 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang *multithread, multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia

Pada MySQL terdapat beberapa perintah. Perintah-perintah pada MySQL ini hampir sama dengan perintah-perintah pada *database server* lainnya. Perintah-perintah MySQL itu antara lain adalah sebagai berikut :

1. *Create database*, digunakan untuk membuat *database* pada *database server*.
2. *Use database*, digunakan untuk menunjuk *database* yang akan digunakan.
3. *Create table,* digunakan untuk membuat tabel pada *database* yang digunakan.
4. *Insert,* digunakan untuk menambahkan record pada tabel.
5. *Update*, digunakan untuk mengubah record yang sudah ada pada tabel.
6. *Drop table,* digunakan untuk menghapus tabel.

7. *Show tables,* digunakan untuk menampilkan tabel-tabel yang telah dibuat dalam *database* yang aktif.
8. *Show field*, digunakan untuk menampilkan seluruh field dalam suatu tabel.
9. *Alter table*, digunakan untuk menambah,merubah, dan menghapus field dalam suatu tabel.

Ada beberapa kelebihan yang dimiliki MySQL sehingga dapat menarik banyak pengguna. Beberapa kelebihan yang dimiliki antara lain, sebagai berikut:

1. Fleksibilitas.

   Saat ini, MySQL telah dioptimasi untuk duabelas platform seperti *HP-UX, Linux, Mac OS X, Novell Netware, OpenBSD, Solaris, Microsoft Windows* dan lain-lain. MySQL juga menyediakan *source code* yang dapat diunduh secara gratis, sehingga pengguna dapat mengkompilasi sendiri sesuai platform yang digunakan. Selain itu, MySQL juga dapat dikustomisasi sesuai keinginan penggunanya, misalnya mengganti bahasa yang digunakan pada antarmukanya.

2. Performansi.

   Sejak rilis pertama, pengembang MySQL fokus kepada performa. Hal ini masih tetap dipertahankan hingga sekarang dengan terus meningkatkan fiturnya.

3. Lisensi.

   MySQL menawarkan berbagai pilihan lisensi kepada penggunanya. Lisensi *open source* yang ditawarkan yaitu lisensi *GNU*

General Public License dan *Free/Libre and Open Source Software (FLOSS) License Exception*. Selain itu ditawarkan juga lisensi komersil berbayar yang memiliki fasilitas dukungan teknis.

2.9.3 *Wamp Server*

Wamp adalah sebuah aplikasi yang dapat menjadikan komputer sendiri menjadi sebuah server. Wamp server berguna untuk membuat jaringan lokal sendiri, jadi dapat membuat website secara *offline* untuk masa coba-coba di komputer sendiri.

Para perancang sebuah web biasanya akan merencanakan (*planing*), kemudian membangun (*buliding*) web tersebut pada komputer lokal atau bisa juga di jaringan lokal, tidak langsung mengerjakannya di jaringan internet. Oleh karena itu di komputer perlu ada server lokal. Hal itu, bisa diatasi dengan menggunakan wamp server. Jadi, jika web tersebut sudah selesai dibuat, tinggal di upload ke jaringan internet.