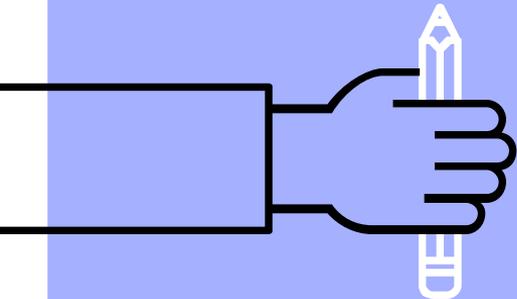
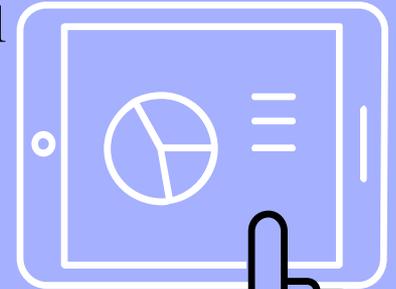
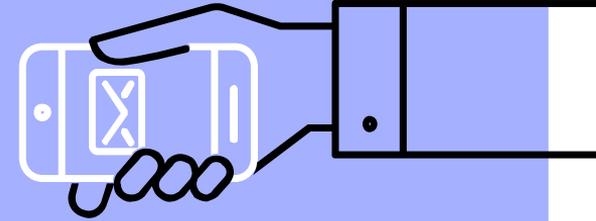


PERANCANGAN TAMPILAN (1)

Jenis dokumentasi rancangan
Kategori program aplikasi dan pendekatan
yang dilakukan
Prinsip dan petunjuk Perancangan 1



PENDAHULUAN

- ✓ Salah satu kriteria penting dari sebuah antarmuka adalah tampilan yang menarik.
- ✓ Yang harus dimiliki oleh perancang tampilan:
 - Jiwa seni yang memadai
 - Mengetahui selera user secara umum
- ✓ Seorang perancang tampilan HARUS mendokumentasikan semua pekerjaan yang telah dia kerjakan selama ini, sebagai bahan evaluasi pembuatan tampilan yang baru atau memperbaiki tampilan sebelumnya
- ✓ PROGRAM YANG DIBUAT BUKAN UNTUK ANDA, NAMUN UNTUK ORANG LAIN

SYARAT MEMBUAT TAMPILAN

- ✓ Harus mempunyai jiwa seni yang memadai.
- ✓ Harus mengerti selera user secara umum
- ✓ Harus bisa meyakinkan pemrogramannya, agar yang dibayangkan dapat diwujudkan (diimplementasikan) dengan piranti bantu (tool) yang tersedia
- ✓ Harus mendokumentasikan semua pekerjaan yang dilakukan

Dokumentasi Perancangan

Dokumentasi rancangan dapat dikerjakan dalam beberapa cara :

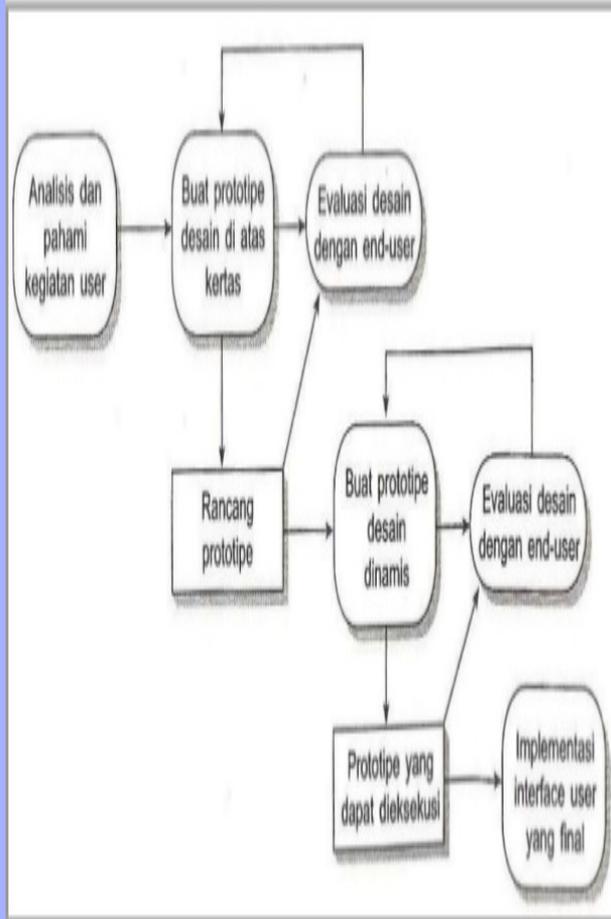
1. Membuat Sketsa pada kertas
2. Menggunakan piranti prototipe GUI
3. Menuliskan tekstual yang menjelaskan tentang kaitan antara satu jendela dengan jendela lainnya.
4. Menggunakan piranti bantu yang disebut CASE (Computer Aided Software Engineering)

PERANCANGAN TAMPILAN

Salah satu kriteria penting dari sebuah antarmuka adalah tampilan yang menarik. Hal ini yang perlu disadari oleh seorang perancang tampilan adalah bahwa ia harus bisa meyakinkan programmer, bahwa apa yang ia bayangkan dapat diwujudkan (diimplementasikan) dengan piranti bantu (tools) yang tersedia.

Bagi perancang antarmuka, hal yang sangat perlu diperhatikan adalah bahwa disetiap pekerjaan perlu dilakukan dokumentasi, sehingga apabila ia ingin mengubah rancangannya atau menyesuaikan rancangannya dengan keinginan user, perancang hanya memerlukan penyesuaian saja. Dokumentasi rancangan dapat dikerjakan atau dilakukan dalam beberapa cara, yaitu:

1. Membuat sketsa pada kertas
2. Menggunakan peranti prototype GUI
3. Menuliskan tekstual yang menjelaskan tentang kaitan antara satu jendela dengan jendela yang lain.
4. Menggunakan peranti bantu yang disebut CASE (Computer-Aided Software Engineering).



KATEGORI PROGRAM APLIKASI DAN PENDEKATAN YANG DILAKUKAN

Program aplikasi pada dasarnya dapat dikelompokkan kedalam dua kategori besar ditinjau dari calon penggunaannya, yaitu:

1. Program aplikasi untuk keperluan khusus dengan pengguna yang khusus pula (special purpose software). Biasanya aplikasi ini diperuntukan untuk kebutuhan tertentu, seperti inventori gudang, pengolahan data akademik, reservasi hotel dan lain sebagainya. Untuk aplikasi ini ada dua pendekatan yang dilakukan, yaitu:
 - a. Pendekatan secara user-centered design adalah perancangan antarmuka yang melibatkan pengguna
 - b. Pendekatan user design approach Pengguna sendirilah yang merancang wajah antarmuka yang diinginkan
2. Program aplikasi yang akan digunakan oleh banyak pengguna (general purpose software). Salah satu kunci penting dalam pembuatan modul antarmuka untuk program-program aplikasi ini adalah dengan melakukan customization, sehingga pengguna dapat menggunakan program aplikasi dengan wajah antarmuka yang sesuai dengan selera masing-masing pengguna. Misal: Word, Excel, Powerpoint

PRINSIP- PRINSIP PERANCANGAN

Prinsip-prinsip perancangan antarmuka pengguna, secara alamiah terbagi menjadi empat komponen, yaitu:

1. Model pengguna; merupakan model konseptual yang diinginkan pengguna dalam memanipulasi informasi dan proses yang diaplikasikan pada informasi tersebut
2. Bahasa perintah; merupakan piranti pemanipulasian model
3. Umpan balik; sebagai kemampuan sebuah program dalam membantu pengguna untuk mengoperasikan program tersebut. Umpan balik dapat berbentuk: pesan-pesan penjelasan, pesan penerimaan perintah, indikasi adanya obyek terpilih, penampilan karakter yang diketikkan lewat keyboard
4. Tampilan informasi; digunakan untuk menunjukkan status informasi atau program ketika pengguna melakukan suatu tindakan

PRINSIP- PRINSIP PERANCANGAN

Prinsip 1 : Kenali Perbedaan → Dalam hal ini perbedaan berdasarkan jenis-jenis pemakai

a. Novice (first time user): biasanya pemakai belum memiliki konsep tugas dan antar muka dangkal sehingga perancang harus membatasi jumlah pilihan, membuat umpan balik yang inovatif dan membuat manual dan tutorial online yang efektif.

b. Knowledgeable Intermittent users : pemakai sudah memiliki konsep tugas yang stabil dan konsep antarmuka luas namun sulit mengingat sintaktik, sehingga perancang dapat membuat struktur menu yang rapi, konsisiten, kejelasan antarmuka yang jelas, serta perlindungan dari bahaya karena eksplorasi fitur.

c. Expert frequent users: pemakai sudah terbiasa dengan konsep tugas dan antarmuka, selain itu juga cenderung menginginkan pekerjaan cepat selesai, perancang dapat membuat rancangan yang makro, shortcuts, singkatan, dsb.

PRINSIP-PRINSIP PERANCANGAN

Prinsip 2 : Gunakan Delapan Aturan Emas Perancangan Antarmuka Pemakai

- a. Berusaha untuk konsisten
- b. Menyediakan usability universal
- c. Memberikan umpan balik yang inovatif
- d. Merancang dialog yang memberikan penutupan (keadaan akhir).
- e. Memberikan pencegahan kesalahan dan penanganan kesalahan yang sederhana.
- f. Memungkinkan pembalikan aksi yang mudah.
- g. Mendukung pusat kendali internal (internal locus of control).
- h. Mengurangi beban ingatan jangka pendek.

PRINSIP- PRINSIP PERANCANGAN

Prinsip 3 : Mencegah Kesalahan

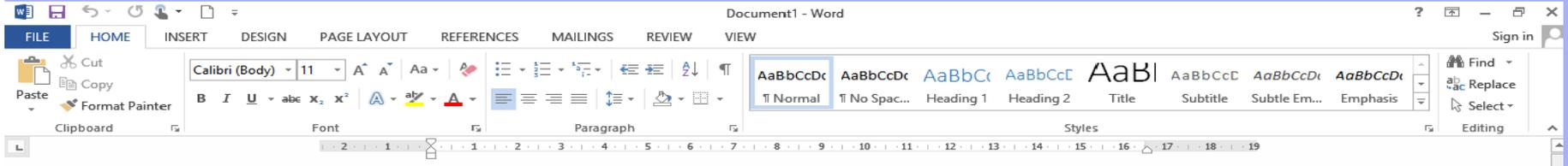
- a. Membetulkan perintah : Mengenali kekurangan perintah dan melengkapinya, memberikan pilihan sebagai ganti menetik serta membatasi pilihan.
- b. Melengkapi urutan aksi: Memungkinkan menggabungkan aksi-aksi menjadi suatu aksi baru dengan makro atau sejenisnya serta melakukan aksi-aksi yang tergantung aksi lain secara otomatis.
- c. Membetulkan pasangan yang bersesuaian : menempatkan tanda pembuka dan penutup dalam suatu aksi dan mengingatkan pemakai bahwa tanda penutup belum dipasang.

PEDOMAN PERANCANGAN

1. Pedoman data display → Tujuan tingkat tinggi data display (Smith and Mosier)
 - a. Konsistensi tampilan data
 - b. Asimilasi informasi yang efisien oleh pemakai.
 - c. Beban ingatan pemakai yang minimal
 - d. Kompatibilitas tampilan data dengan pemasukan data.
 - e. Fleksibilitas kendali pemakai



PEDOMAN PERANCANGAN



2. Pedoman untuk mengarahkan perhatian pengguna (Wickens and Holland, 2000)
 - a. Intensitas : Gunakan dua tingkatan , dengan penggunaan intensitas tinggi hanya untuk menarik perhatian.
 - b. Penandaan : Gunakan garis bawah dan lain-lain.
 - c. Ukuran : Gunakan maksimal 4 ukuran, ukuran yang lebih besar lebih menarik perhatian
 - d. Pilihan Font : gunakan maksimal 3 jenis fonts.
 - e. Blingking : Gunakan secara terbatas.
 - f. Warna : Gunakan maksimal 4 warna standar.
 - g. Audio : Gunakan nada lembut untuk feedback biasa dan nada keras untuk kondisi darurat.

PEDOMAN PERANCANGAN

3. Pedoman data entri : Tujuan tingkat tinggi Data Entry (Smith and Mosier, 1986):

- a. Konsistensi transaksi pemasukan data.
- b. Aksi pemasukan yang minimal oleh pemakai.
- c. Beban ingatan pemakai yang minimal
- d. Kompatibilitas pemasukan data dengan tampilan data.
- e. Fleksibilitas kendali pemakai.

PROGRAM APLIKASI NILAI MAHASISWA

Nama Mahasiswa

Input Nilai :

Nilai UTS

Nilai UAS

Nilai Harian

Output :

Nama Mahasiswa

Nilai Rata-rata

Grade

Form PKL-01

FORM PENGAJUAN PRAKTEK KERJA LAPANGAN

NIM / NAMA :
JENJANG : D-III
PROGRAM STUDI : MANAJEMEN INFORMATIKA
SEMESTER : TA :

Alternatif 1
TOPIK/JUDUL :
:
:
NAMA PERUSAHAAN / INSTANSI / LEMBAGA :
BIDANG USAHA :
ALAMAT :
:

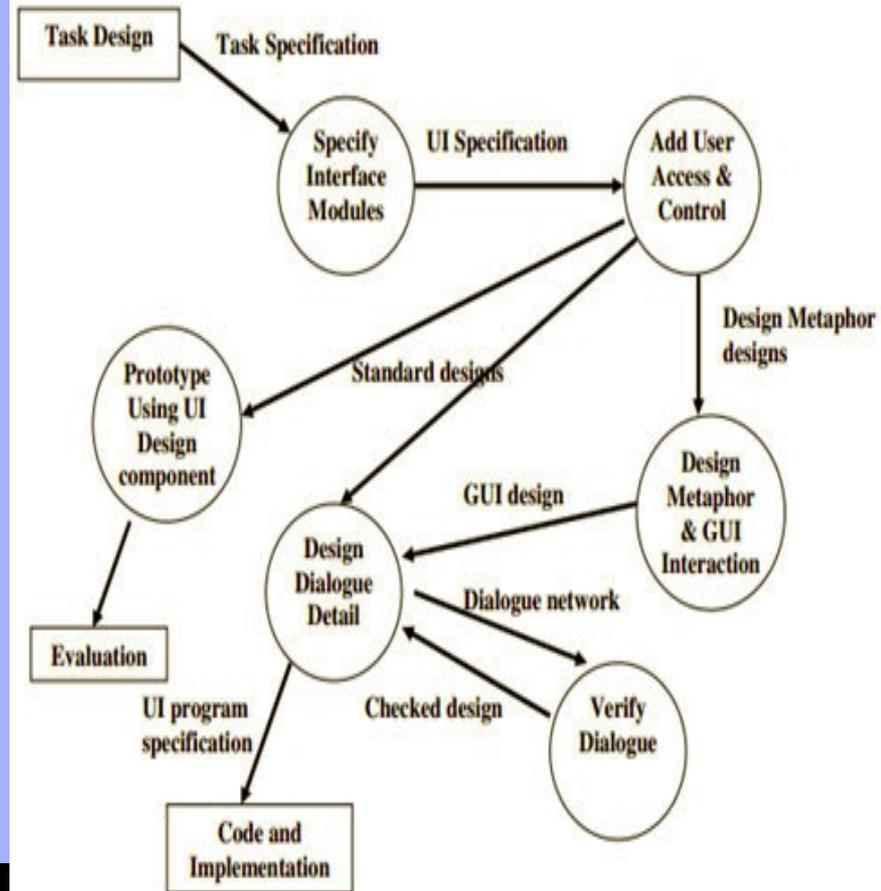
Alternatif 2
TOPIK/JUDUL :
:
:
NAMA PERUSAHAAN / INSTANSI / LEMBAGA :
BIDANG USAHA :
ALAMAT :
:

LAMPIRAN : KHS
Matanglumpangdua, 2013
Mahasiswa yang Mengajukan

URUTAN PERANCANGAN

Setelah mengetahui beberapa prinsip dan petunjuk perancangan, maka dalam merancang antarmuka juga dapat dikerjakan berdasarkan urutan berikut:

1. Pemilihan ragam dialog → pilihlah ragam dialog yang menurut pikiran cocok untuk tugas tersebut.
2. Perancangan struktur dialog → tahap ini adalah melakukan analisis tugas dan menentukan model pengguna dari tugas tersebut untuk membentuk struktur dialog yang sesuai.
3. Perancangan format pesan → pada tahap ini tata letak tampilan dan keterangan tekstual secara terinci harus mendapat perhatian lebih.



URUTAN PERANCANGAN

4. Perancangan penanganan kesalahan → suatu kesalahan dalam menjalankan suatu aplikasi tidak dapat dihindarkan, hal ini bisa timbul baik kesalahan user maupun kesalahan komputer. Bentuk-bentuk penanganan kesalahan dapat dilakukan, diantaranya adalah:
 - a. Validasi pemasukan data; mekanisme untuk mengulang pemasukan data bila data yang dimasukkan keliru. Seharusnya datanya numerik tapi memasukkan character.
 - b. Proteksi pengguna; program memberikan peringatan ketika pengguna melakukan suatu tindakan tidak disengaja, misalnya penghapusan berkas
 - c. Pemulihan dari kesalahan; tersedianya mekanisme untuk membatalkan tindakan yang baru saja dilakukan - penampilan pesan salah pada saat yang tepat sesuai dengan kesalahan yang terjadi pada saat itu.
 - d. Penampilan pesan salah yang tepat dan sesuai dengan kesalahan yang terjadi pada waktu itu.
5. Perancangan struktur data → dalam hal ini menentukan struktur data yang dapat digunakan untuk menyajikan dan mendukung fungsionalitas komponen-komponen antarmuka yang diperlukan.

PERANCANGAN TAMPILAN BERBASIS TEKS

Antarmuka pada berbagai program aplikasi hampir dipastikan berbasis tekstual, yakni yang mendasarkan karakter ASCII. Pada perancangan tampilan untuk antarmuka berbasis teks agar memperoleh tata letak tampilan yang berkualitas tinggi maka ada enam faktor yang harus dipertimbangkan, yaitu:

1. Urutan penyajian;
2. Kelonggaran (spaciousness); Teks tidak berhimpitan satu dengan lainnya
3. Pengelompokan; Data yang saling berkaitan dikelompokkan agar mudah dalam penstrukturan tampilan secara keseluruhan
4. Relevansi; Tampilkan hanya pesan-pesan yang relevan dengan topik yang sedang ditampilkan dilayar monitor
5. Konsistensi; Penggunaan ruang tampilan yang seefisien mungkin
6. Kesederhanaan; Informasi yang disajikan mudah dipahami oleh pengguna

Pengisian Data Pribadi

Nama: Okky Wisnu Murti Santosa
Tempat/tgl lahir: Klaten, 8 Januari 1961
Alamat: Jalan Pelan-pelan No. 7A Yogyakarta
Kelamin: Laki-laki
Pekerjaan: Wiraswasta
Pendidikan:
SD: SDN Ungaran I, Yogyakarta
SMP: SMPN 87, Medan
SMU: SMUN 24, Jakarta
Sarjana: Universitas Merdeka, Jakarta

a)

urutan penyajian

Pengisian Data Pribadi

Nama: Okky Wisnu Murti Santosa
Tempat/tgl lahir: Klaten, 8 Januari 1961
Kelamin: Laki-laki
Alamat: Jalan Pelan-pelan No. 7A Yogyakarta

Pendidikan:

SD:	SDN Ungaran I	Yogyakarta
SMP:	SMPN 87	Medan
SMU:	SMUN 24	Jakarta
Sarjana:	Unbar	Jakarta

Pekerjaan: Wiraswasta

b)

relevansi

kesederhanaan

pengelompokan

konsistensi

kelonggaran

1. Urutan penyajian;
2. Kelonggaran (spaciousness); Teks tidak berhimpitan satu dengan lainnya
3. Pengelompokan; Data yang saling berkaitan dikelompokkan agar mudah dalam penstrukturan tampilan secara keseluruhan
4. Relevansi; Tampilkan hanya pesan-pesan yang relevan dengan topik yang sedang ditampilkan di layar monitor
5. Konsistensi; Penggunaan ruang tampilan yang seefisien mungkin
6. Kesederhanaan; Informasi yang disajikan mudah dipahami oleh pengguna

PERANCANGAN TAMPILAN BERBASIS GRAFIS

Dengan antarmuka berbasis grafis berbagai kemudahan dalam hal pengontrolan format tampilan dapat dikerjakan dengan lebih mudah dan fleksibilitas tampilan dapat semakin dirasakan oleh perancang tampilan maupun penggunaannya..

Terdapat lima faktor yang perlu diperhatikan pada saat kita merancang antarmuka berbasis grafis, yaitu:

1. Ilusi pada obyek-obyek yang dapat dimanipulasi.

a). Gunakan kumpulan obyek yang disesuaikan dengan aplikasi yang akan dibuat

b). Penampilan obyek-obyek grafis harus dilakukan dengan keyakinan bahwa pengguna akan mengerti

c). Gunakan mekanisme yang konsisten untuk memanipulasi obyek yang akan muncul di layar

2. Urutan visual dan fokus pengguna; Gunakan simbol berkedip untuk menunjukkan posisi kursor, gunakan warna dengan kontras yang nyata untuk menunjukkan adanya suatu pilihan

3. Struktur internal; Adanya reveal code (suatu tanda khusus yang digunakan untuk menunjukkan adanya perbedaan font style).

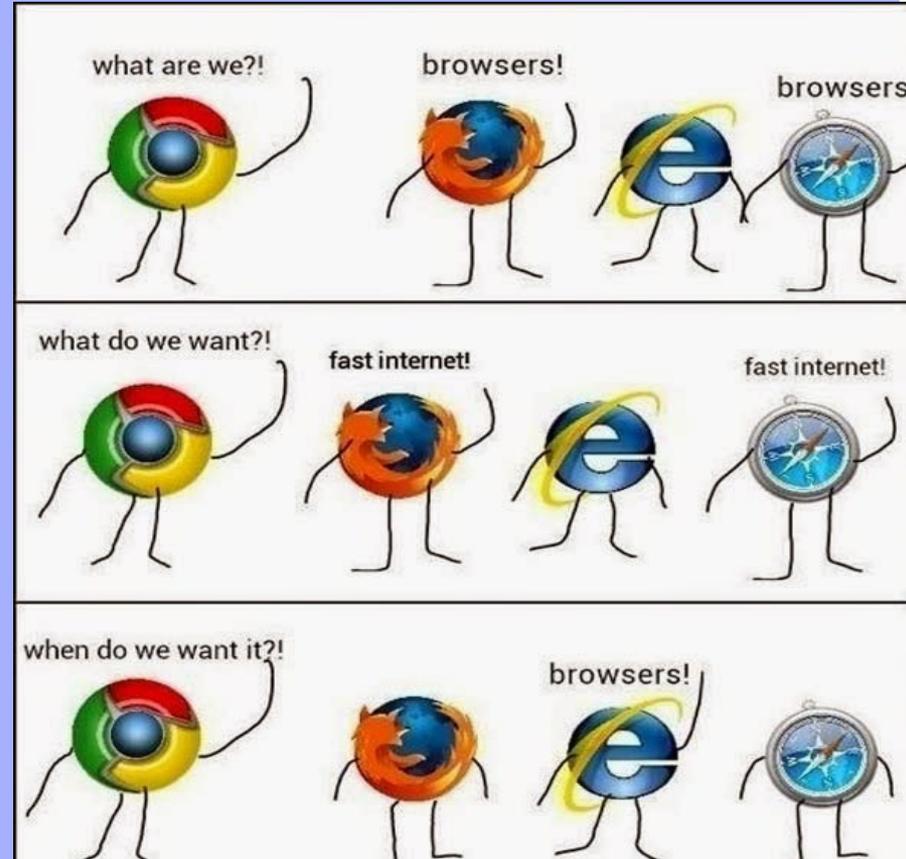
4. Kosakata grafis yang konsisten dan sesuai; Penggunaan simbol untuk menjelaskan sesuatu seperti gambar disket untuk menjelaskan penyimpanan data, gambar printer untuk menyatakan proses mencetak.

5. Kesesuaian dengan media; Tampilan akan berubah sesuai dengan resolusi tampilan yang digunakan.

WAKTU TANGGAP

Waktu tanggap yang baik memang tidak dapat ditentukan, karena ada beberapa aspek yang mempengaruhinya, antara lain yaitu ragam interaksi yang diinginkan dan kefasihan pengguna dalam menjalankan program aplikasi tersebut, serta waktu tanggap yang berbeda-beda dapat mempengaruhi konsentrasi pengguna yang pada gilirannya akan mempengaruhi kinerja pengguna..

Waktu tanggap yang ideal adalah kurang dari 2 detik, sehingga konsentrasi pengguna tidak akan terpecah keaktifitas yang lain, yang pada akhirnya akan menghambat kinerjanya.



PENANGANAN KESALAHAN

Kesalahan dapat dibagi menjadi dua bagian, yang pertama yaitu kesalahan pada saat implementasi program, biasanya kesalahan tersebut merupakan kesalahan sintaksis yang secara langsung akan terdeteksi oleh compiler, sehingga sering disebut dengan istilah compile time error, dan yang kedua adalah kesalahan logika ketika program sedang berjalan, kesalahan ini terjadi pada saat program sedang dijalankan, biasanya efek yang ditimbulkan adalah berhentinya eksekusi program secara abnormal.

Penyebab kesalahan logika dapat dikelompokkan menjadi dua penyebab, yaitu datang dari pengguna, contohnya pengguna mengisikan data yang tidak sah atau sesuai. Dan penyebab selanjutnya adalah pada saat proses eksekusi, contohnya ketika program perlu membaca suatu data ternyata data tersebut tidak tersedia.

Penanganan kesalahan harus dilakukan dengan tepat, biasanya dilakukan dengan memberikan pesan umpan balik kepada pengguna akan keadaan eksekusi program saat ini. Dan hal ini dapat dilakukan jika hanya didalam program tersebut memiliki modul perangkap kesalahan.

PERANTI BANTU SEDERHANA

Dalam hal ini peranti bantu yang dapat digunakan adalah sebuah peranti bantu berbentuk lembaran kertas yang tidak perlu disiapkan secara khusus, tetapi dapat menggunakan sebuah kertas kosong. Untuk mempermudah penamaan maka lembaran kertas yang dimaksud diberi nama dengan lembar kerja tampilan (LKT). LKT pada dasarnya terdiri dari empat bagian:

- 1.Nomor lembar kerja,
- 2.Bagian tampilan : berisi sketsa tampilan yang akan muncul dilayar
- 3.Bagian Navigator: bagian ini antara lain menjelaskan kapan tampilan itu akan muncul dan berubah menjadi tampilan lain.
- 4.Bagian keterangan : berisi penjelasan singkat tentang atribut tampilan yang dipakai.

The diagram shows a rectangular form with a dashed border, representing a simple assistive device (LKT). The form is divided into four sections:

- No**: A small box in the top-left corner for the document number.
- Tampilan**: A large rectangular box in the top-left quadrant for a sketch of the display.
- Navigasi**: A rectangular box in the top-right quadrant for navigation details.
- Keterangan**: A wide rectangular box at the bottom for a brief explanation of the display attributes.

No. sheet : 5

PENGISIAN DATA PRIBADI

Nama:

Alamat:

Kota:

TGL. Lahir: / /

Kelamin : Laki-laki Perempuan

Pendidikan : SD Sarjana S1
 SLTP Master
 SLTA Doktor

NAVIGASI :

- Klik 'Selesai', ganti tampilan dengan No.1.
- Klik 'Simpan', aktifkan penyimpanan berkas dan kosongkan borang.
- Klik 'Batal', kosongkan borang.

Keterangan :

- * Judul jendela pakai Arial, 12 point.
- * Teks pakai Arial, 10 point, hitam.
- * Medan data, Arial, 10 point, miring.
- * Teks 'push button' Arial, 12 point.

LKT pada dasarnya terdiri dari empat bagian:

1. Nomor lembar kerja,
2. Bagian tampilan : berisi sketsa tampilan yang akan muncul dilayar
3. Bagian Navigator: bagian ini antara lain menjelaskan kapan tampilan itu akan muncul dan berubah menjadi tampilan lain.
4. Bagian keterangan : berisi penjelasan singkat tentang atribut tampilan yang dipakai.

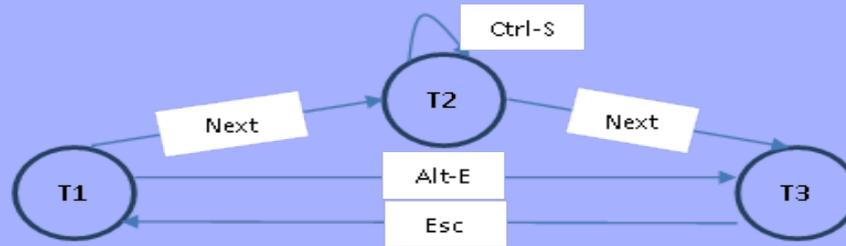
JARINGAN SEMANTIK TAMPILAN

Menggambarkan arah navigasi pada setiap lembar kerja.

Dalam perancangan tampilan harus memungkinkan pengguna dapat berdialog dengan komputer. Maka sebelum merancang antarmuka pemrogram memerlukan navigasi pada setiap lembar kerjanya, maka untuk mempermudah programmer untuk menentukan navigasinya maka dibuatlah sebuah bagan yang disebut jaringan semantik tampilan.

Jaringan semantik tampilan terdiri dari dua komponen yaitu **nomor tampilan**, dalam LKT disebut nomor lembar kerja, dan **transisi** yang menyebabkan perpindahan ketampilan yang lain. Dalam jaringan semantik lembar kerja diberi symbol lingkaran dan transisi diberi symbol anak panah.

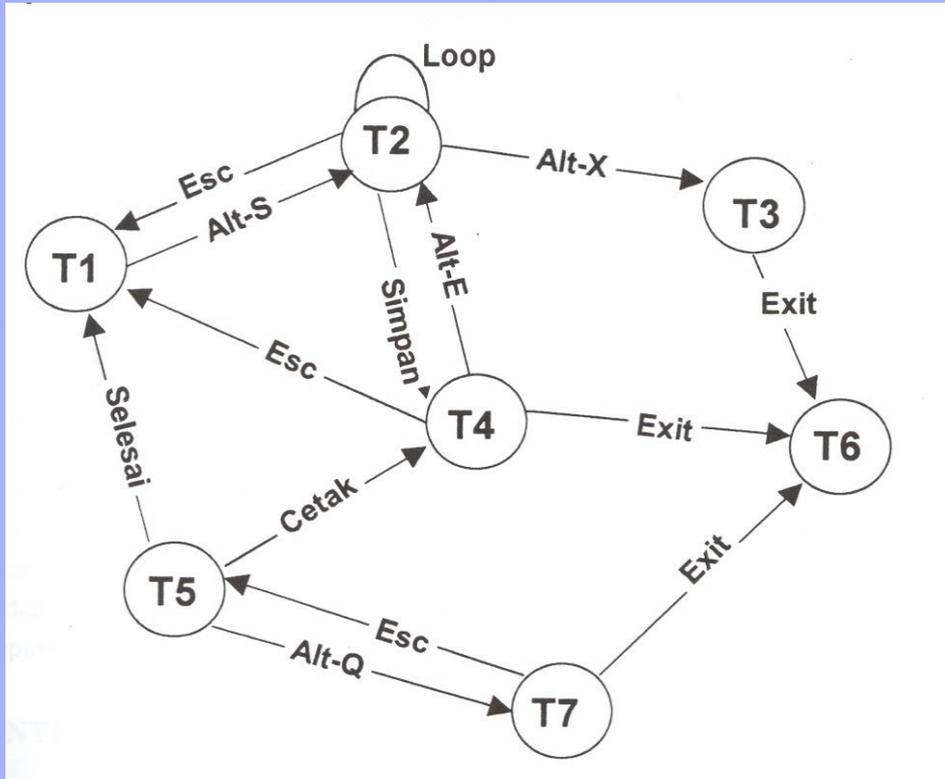
JARINGAN SEMANTIK TAMPILAN



Dari gambar, diketahui sebuah aplikasi dengan tampilan T1, T2, dan T3.

1. Pada tampilan T1, jika user menekan Next, maka tampilan berubah menjadi tampilan T2.
2. Pada tampilan T2, , jika user menekan Next, maka tampilan berubah menjadi tampilan T3.
3. Jika user menekan CTRL+S T2 maka akan terjadi looping pada tampilan T2
4. Pada tampilan T3, jika user menekan Esc maka tampilan berubah menjadi tampilan T1
5. Dan jika user menekan ALT+E maka tampilan T1 berubah menjadi tampilan T3

Jaringan Semantik Tampilan



- T1, T2,...,T7 adalah nomor tampilan (Lembar Kerja)
- Tulisan Alt-S, Exit, Esc, dll adalah event
- Anak panah menunjukkan transisi

Contoh :

Jika pengguna sedang berada di tampilan T5, dan kemudian ia menekan tombol Alt-Q (ada event penekanan papan ketik), maka tampilan akan berubah menjadi T7. Jika ia menekan tombol tekan Selesai, maka akan kembali ke tampilan T1.

THANK YOU

