INTERAKSI MANUSIA DAN MESIN

Penanganan Kesalahan & Dokumentasi Help

Telkom University

Budhi Irawan, S.Si, MT

TIPE KESALAHAN DAN SLIP

- Hampir tidak ada sistem yang berjalan sempurna sebelum melewati berbagai rentetan ***kesalahan (error)***. Semakin besar sistem yang dibuat, semakin banyak kesalahan yang dapat timbul.
- Compaq pernah mempertimbangkan untuk mengubah "***Press Any Key***" menjadi "***Press Enter Key***" dikarenakan banyak orang yang menanyakan letak tombol "***Any***" di keyboard.
- Cara <u>mengurangi kesalahan user</u>, misal: ***seleksi user yang tepat***, ***pelatihan yang memadai***, ***desain peralatan***, ***prosedur*** dan ***lingkungan***.

PENANGANAN KESALAHAN

- **Tipe-tipe Kesalahan (Errors):**
  - Kesalahan Persepsi
  - Kesalahan Kognitif
  - Kesalahan Motorik (Gerak)

- **Tipe-tipe Slip:**
  - Kesalahan *Capture*
  - Kesalahan Deskripsi
  - Kesalahan *Data Driven*
  - Pengaktifan Asosiatif
  - Hilangnya Pengaktifan
  - Kesalahan Mode

TIPE KESALAHAN

- Perceptual error
- Cognitive error
- Motorik error

PERCEPTUAL ERROR

- Gagal menangkap informasi
- Gagal menangkap perhatian pengguna
- Contoh : 8/B, Z/2, I/1

COGTINITIVE ERROR

- Ketiadaan bantuan
- Inkonsistensi
- Ketiadaan status informasi
- Kemampuan yang kurang
- Contoh : argumen yang tidak konsisten pada pada sintak command language

MOTORIC ERROR

- Lemahnya koordinasi tangan dan mata
- Tekanan kecepatan
- Memerlukan kemampuan yang tinggi pada koordinasi tangan dan mata
- Memerlukan jenis keahlian lainnya
- **Contoh : Click mouse : single atau double, mengetik dengan cepat membuat banyak kesalahan**

TIPE SLIP

- **Kesalahan Capture ;** Kesalahan karena terlalu sering atau kebiasaan.
- **Kesalahan Deskripsi ;** Kesalahan dalam melakukan aksi pada objek yang salah.
- **Kesalahan Data Driven ;** Kesalahan karena pengaruh data dari area edit
- **Pengaktifan Asosiatif ;** Kesalahan karena pengaruh data yang ada dipikirkan pada saat ini.
- **Hilangnya Pengaktifan ;** Kesalahan karena lupa apa yang harus dilakukan.
- **Kesalahan Mode ;** Kesalahan akibat lupa pada mode yang sedang dilakukan

PANDUAN PENCEGAHAN KESALAHAN

- Menghapus mode-mode atau menyediakan ***petunjuk*** yang terlihat untuk mode-mode tersebut.
- Gunakan ***teknik koding yang baik*** (warna, gaya).
- Memaksimalkan pengenalan dan mengurangi hafalan.
- Merancang urutan gerak atau perintah yang tidak sama
- Mengurangi kebutuhan untuk mengetik.
- Uji dan memantau kesalahan-kesalahan dan memperbaikinya.
- Memungkinkan pertimbangan ulang aksi-aksi yang dilakukan oleh user, misalnya memindahkan file dari *recycle bin*

PANDUAN RECOVERY KESALAHAN

- Menyediakan tipe-tipe tanggapan yang sesuai.
- Query : bertanya pada user apa yang sudah dilakukan, kemudian melegalkan tindakan yang salah.
- Menyediakan fungsi “undo” dan pembatalan dari proses yang sedang berjalan.
- Meminta konfirmasi untuk perintah yang drastis dan bersifat merusak.
- Menyediakan pengecekan yang beralasan pada masukan data.
- Mengembalikan kursor ke area kesalahan, memungkinkan untuk melakukan perbaikan.
- Menyediakan beberapa kecerdasan buatan.
- Menyediakan akses cepat kepada bantuan untuk konteks-sensitif.

JENIS-JENIS DARI DOKUMENTASI/HELP

- **Quick Reference ;** Digunakan sebagai pengingat untuk user dari suatu yang detail yang secara dasar sangat familiar dan biasa digunakan.
- **Task-Spesifik Help ;** Digunakan untuk membantu user menghadapi masalah atau tidak pasti mengambil tindakan dalam memecahkan masalah yang khusus.
- **Full Explanation ;** Suatu alat bantu atau perintah yang dapat membantu memahami secara lengkap.
- **Tutorial ;** Khusus untuk user baru yang menyediakan perintah secara *step by step.*

KEBUTUHAN DARI USER SUPPORT

- **Availability.** User dapat menggunakan bantuan pada setiap waktu selama berinteraksi dengan sistem. User tidak perlu keluar dari aplikasi selama bekerja untuk membuka aplikasi bantuan.
- **Accuracy dan completeness**. Bantuan ini seharusnya tersedia secara lengkap dan akurat.
- **Consistency.** Konsisten pada sistem yang ada, antara jenis online dan jenis tercetak, juga konsisten dari sisi content, terminologi dan bentuk presentasi.
- **Robustness**. Biasanya digunakan oleh orang yang sedang dalam kesulitan karena sistem mempunyai perilaku yang tidak diharapkan atau mempunyai kesalahan.
- **Flexibility**. Flexibility membantu setiap user berinteraksi sesuai dengan keinginannya.
- **Unobtrusiveness**. Seharusnya tidak mencegah user dalam melanjutkan pekerjaannya atau terpengaruh dengan aplikasi user. Untuk menghindari ini digunakan presentasi pada layar yang terpisah

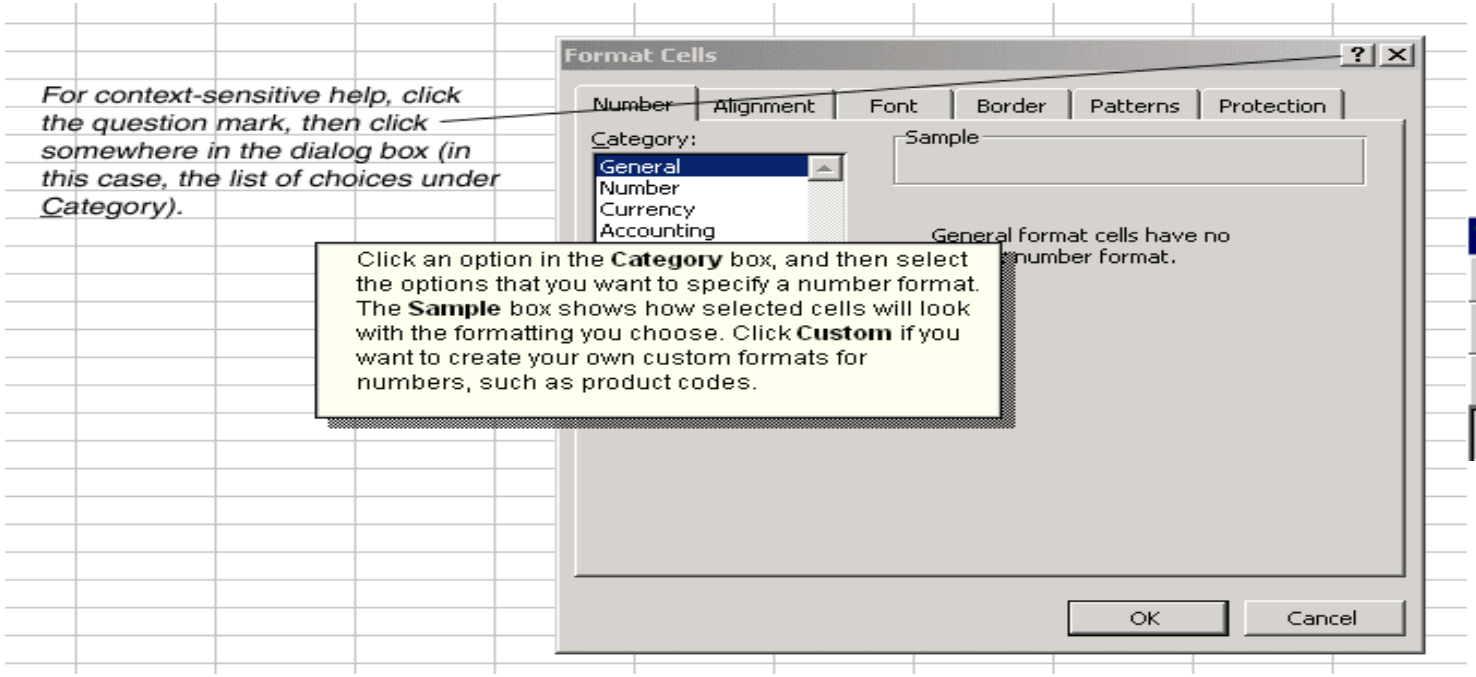
BENTUK USER SUPPORT

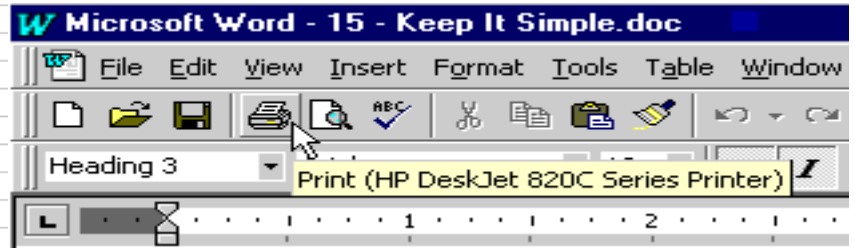
- **Command assistance**. Ditemui pada sistem DOS dan UNIX. Pengguna dapat menggunakan bantuan dengan mengetikkan parameter tertentu.
- **Command prompts**. Jenis bantuan yang biasanya tampil bila pengguna melakukan kesalahan sederhana misalnya kesalahan sintaks
- **Context sensitive help**. Jenis ini berbentuk menu based system yang menyediakan bantuan pada menu option.
- **Online tutorial.** Mengijinkan user bekerja melalui aplikasi dasar dengan lingkungan percobaan. User dapat melihat kemajuan sesuai dengan kecepatan dan dapat mengulangi bagian dari tutorial yang dia inginkan. Kebanyakan tutorial online tidak mempunyai bagian intelligent, karena tidak mempunyai pengetahuan tentang user dan pengalaman user sebelumnya.
- **Online documentation.** Jenis ini tersedia secara online di internet/ intranet, khususnya untuk aplikasi yang diproduksi secara massal.
- **Intelligent Help System**. Dioperasikan untuk memonitoring aktifitas user dan mengkonstruksikan model sesuai dengan user. Model ini termasuk pengalaman, preferences, kesalahan user atau kombinasi dari semuanya.

COMMAND BASE METHODS

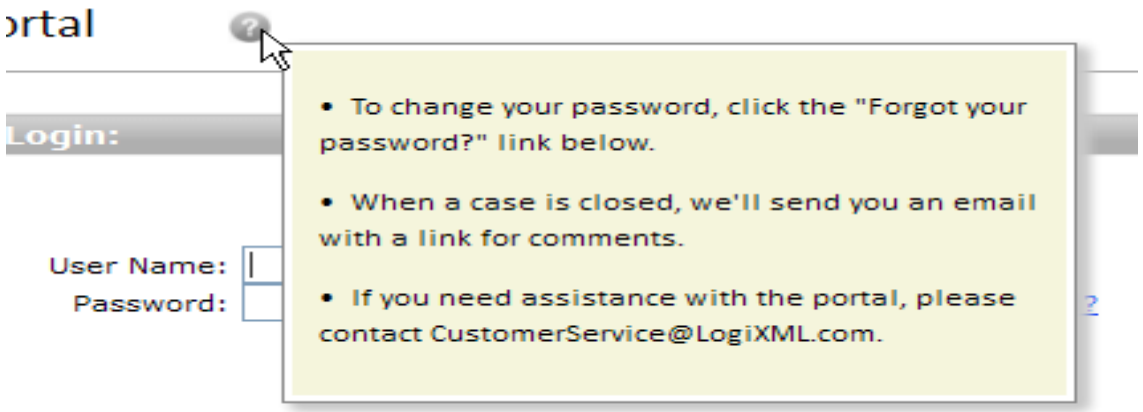
```
C:\WINDOWS\system32>dir /?
Displays a list of files and subdirectories in a directory.

DIR [drive:][path][filename] [/A[[:]attributes]] [/B] [/C] [/D] [/L] [/N]
  [/O[[:]sortorder]] [/P] [/Q] [/S] [/T[[:]timefield]] [/W] [/X] [/4]

  [drive:][path][filename]
              Specifies drive, directory, and/or files to list.

  /A          Displays files with specified attributes.
  attributes   D  Directories                R  Read-only files
               H  Hidden files               A  Files ready for archiving
               S  System files               -  Prefix meaning not
  /B          Uses bare format (no heading information or summary).
  /C          Display the thousand separator in file sizes.  This is the
              default.  Use /-C to disable display of separator.
  /D          Same as wide but files are list sorted by column.
  /L          Uses lowercase.
  /N          New long list format where filenames are on the far right.
  /O          List by files in sorted order.
  sortorder    N  By name (alphabetic)       S  By size (smallest first)
               E  By extension (alphabetic)  D  By date/time (oldest first)
               G  Group directories first    -  Prefix to reverse order
  /P          Pauses after each screenful of information.
  /Q          Display the owner of the file.
  /S          Displays files in specified directory and all subdirectories.
  /T          Controls which time field displayed or used for sorting
  timefield   C  Creation
              A  Last Access
              W  Last Written
  /W          Uses wide list format.
  /X          This displays the short names generated for non-8dot3 file
              names.  The format is that of /N with the short name inserted
              before the long name. If no short name is present, blanks are
              displayed in its place.
  /4          Displays four-digit years
```

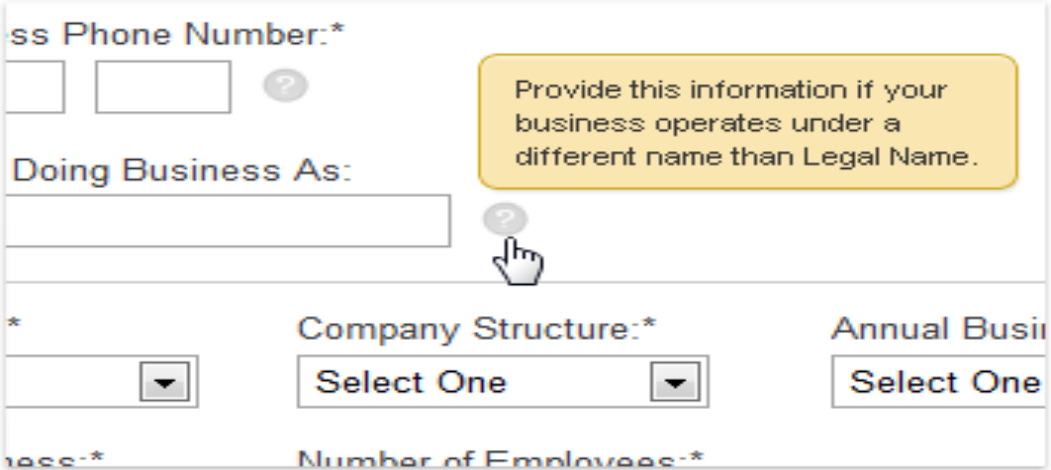
CONTEXT SENSITIVE HELP





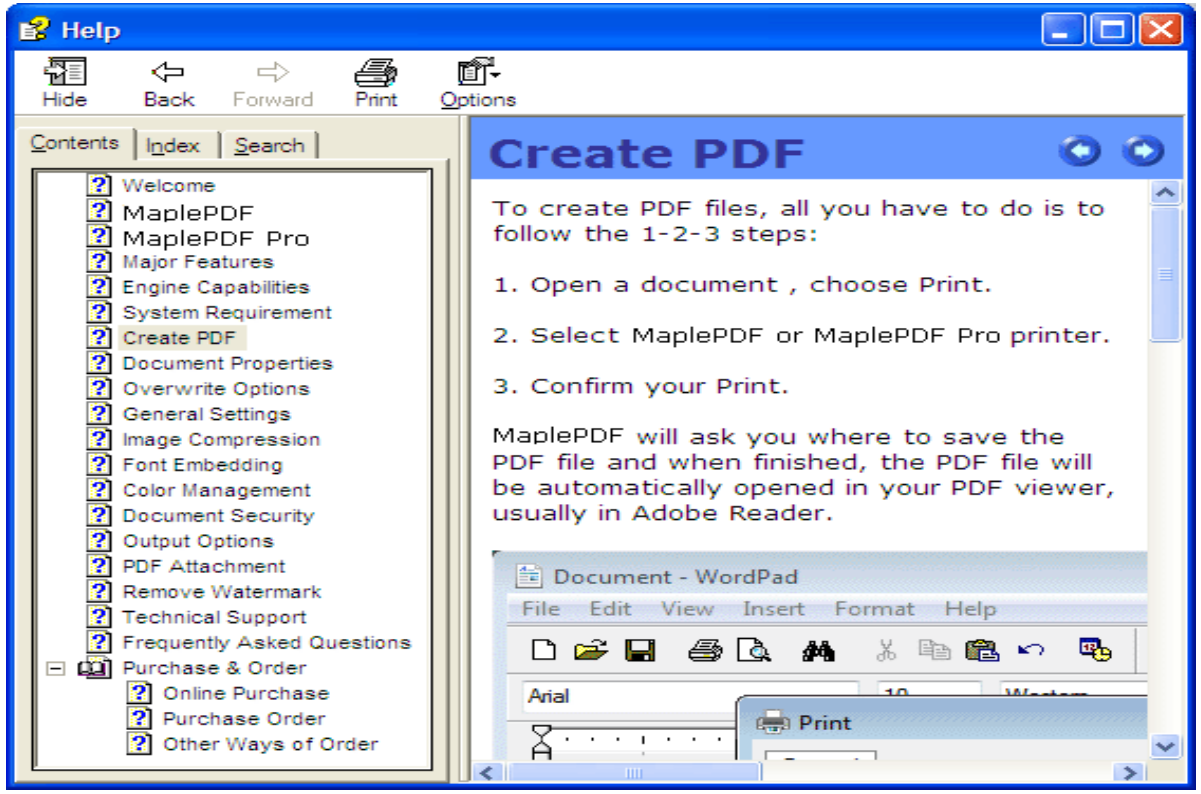
CONTEXT SENSITIVE HELP

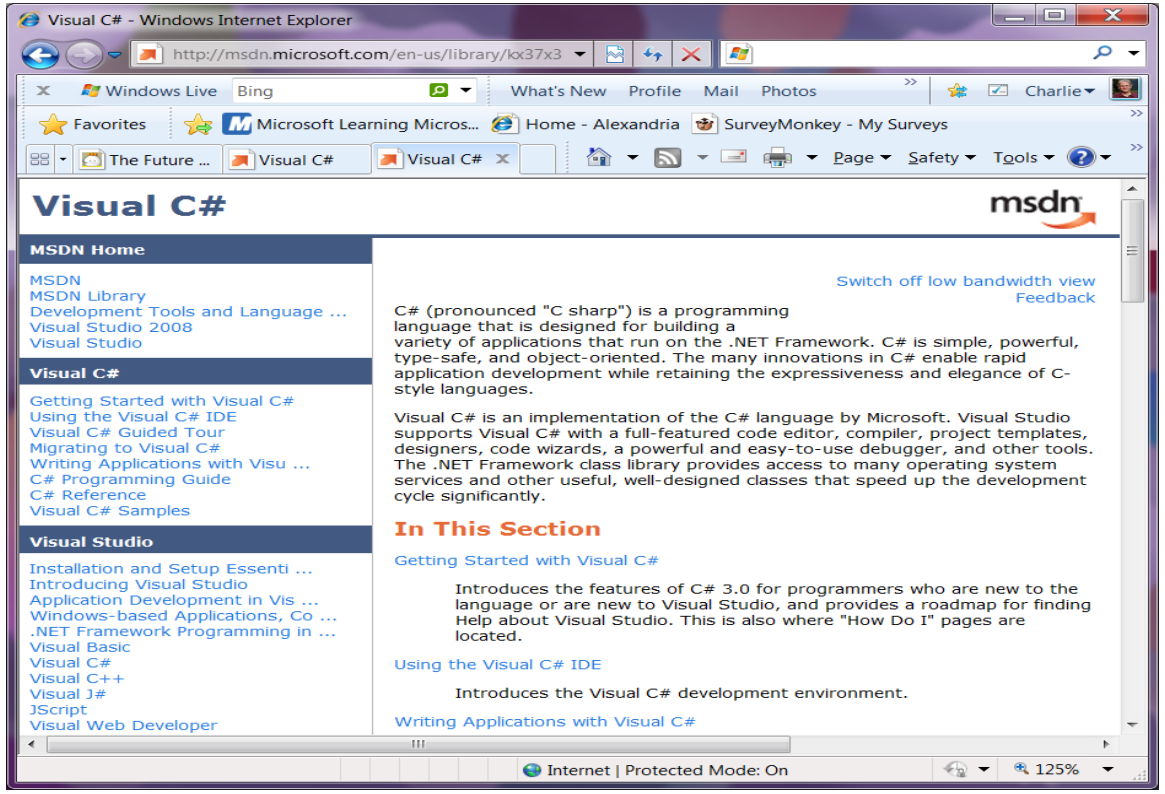




TUTORIAL HELP

ONLINE DOCUMENTATION





INTELEGENT HELP

KNOLWLEDGE REPRESENTATION : USER MODELLING

- **Quantification**. Model yang sederhana dari user modelling yang menggunakan jumlah tingkatan dari keahlian yang akan merespon kearah yang berbeda.
- **Stereotypes**. Berbasiskan pada karakteristik user dan kemungkinan sederhana, seperti membuat perbedaan antara user baru dan user yang ahli atau yang lebih kompleks, seperti membuat stereotype yang berbasiskan pada lebih dari satu informasi.
- **Overlay Models**. Merupakan model yang ideal yang membandingkan perilaku user. Hasilnya ditampilkan dalam dua model atau perbedaan. Keuntungan dari model ini dapat melihat secara pasti bagian dari aktifitas suatu sistem. Pendekatan yang sama digunakan pada error bases model dimana sistem menyimpan rekaman kesalahan dan perilaku sebenarnya dari user serta membandingkannya.

KNOWLEDGE REPRESENTATION : DOMAIN DAN TASK MODELLING

- Pendekatan yang umum dari masalah ini adalah untuk mewakili tugas user dari urutan perintah yang tersedia untuk mengeksekusinya.
- Sebagaimana pada tugas user, command digunakan untuk membandingkan urutan tugas yang telah disimpan dan mencocokkan dengan urutan tepat. Jika urutan command user tidak cocok, maka dibutuhkan bantuan. Pendekatan ini digunakan pada sistem PRIAM.

KNOWLEDGE REPRESENTATION : MODELLING ADVISORY STRATEGY

- Sistem ini kadang disebut dengan intelligent help yang membuat modelling advisory atau strategi tutorial.
- Pada sistem ini tidak hanya membolehkan memilih nasehat yang cocok untuk user, tetapi juga menggunakan metode yang cocok

TEKNIK UNTUK KNOWLEDGE REPRESENTATION

1. **Rule Based Techniques**. Pengetahuan digunakan untuk mengetahui sekumpulan aturan dan kenyataan. Teknik ini digunakan untuk domain yang relatif besar dan dapat mewakili kegiatan yang menampilkan pengetahuan.
2. **Frame Based Techniques**. Digunakan untuk mewakili situasi yang umum terjadi. Frame merupakan suatu struktur yang berisi slot yang diberi label yang memiliki ciri yang berhubungan.
3. **Network Based Techniques**. Mewakili pengetahuan tentang user dan sistem yang merupakan hubungan antara kenyataan, contoh yang paling umum adalah semantic network. Network merupakan suatu hirarki dan child dapat berhubungan dengan parent-nya.
4. **Examples Based Techniques**. Mewakili pengetahuan yang secara implisit dengan struktur keputusan dari suatu klasifikasi sistem.

MASALAH DENGAN KNOWLEDGE REPRESENTATION DAN MODELLING

- Pengetahuan mewakili suatu issue pusat dalam intelligent help system, tetapi tidak tanpa masalah itu sendiri, pengetahuan kadang sulit didapatkan, terutama jika ada domain expert yang tidak tersedia.
- menginterpretasikan informasi yang cocok.
- Inisiatif, haruskan user mempertahankan pengawasan yang lengkap terhadap sistem, haruskah sistem langsung berinteraksi atau haruskah mendukung penggabungan dialog?
- Effect, para perancang seharusnya memperhatikan efek dari modelling dan adaptasi.
- Scope, para perancang perlu memperhatikan scope dari bantuan dimana digunakan pada level aplikasi atau sistem yang luas.

MERANCANG DOKUMENTASI / HELP

Terdapat banyak cara untuk merancangnya dan semua itu diserahkan pada perancang untuk memilih cara yang terbaik akan tetapi hal yang perlu diperhatikan adalah :

- Perancangan seharusnya tidak seperti “add-on” pada sistem. Secara ideal seharusnya merupakan bagian integral dalam sistem.
- Perancangan harus memperhatikan isi dari bantuan dan konteks sebelum teknologi tersedia.

MASALAH PRESENTASI

- **How is help request?**

  Pilihan pertama bagi perancang untuk membuat bagaimana bantuandapat diakses oleh user. Terdapat beberapa pilihan. Bantuan ini dapat berupa command, button fungsi yang dapat memilih on atau off atau aplikasi yang terpisah.

- **How is help displayed?**

  Bagaimana bantuan dapat dilihat oleh user. Dalam system window mungkin ditampilkan dalam window yang baru. Dalam sistem lain mungkin dalam layar yang penuh atau bagian dari layar. Alternatif lain dapat berbentuk pop-up box atau tingkat command line.

- **Effective presentation of help**

  Tidak menjadi masalah teknologi apa yang digunakan untuk membuat akan tetapi yang perlu diperhatikan akan menjadi suatu prinsip, yaitu keefektifan.

MASALAH IMPLEMENTASI

- Para perancang harus membuat keputusan untuk implementasi berupa secara fisik maupun pilihan yang tersedia untuk user. Keputusan ini sudah termasuk dalam pernyataan command operating system, apakah berbentuk meta-command atau aplikasi. Hambatan fisik berupa screen space, kapasitas memori dan kecepatan.

- Masalah lain adalah bagaimana struktur data bantuan: apakah berbentuk single file, hierarchy file atau database.

Masalah Implementasi

Para perancang harus membuat keputusan untuk implementasi berupa secara fisik maupun pilihan yang tersedia untuk user. Keputusan ini sudah termasuk dalam pernyataan command operating system, apakah berbentuk meta-command atau aplikasi. Hambatan fisik berupa screen space, kapasitas memori dan kecepatan.

Masalah lain adalah bagaimana struktur data bantuan: apakah berbentuk single file, hierarchy file atau database.