

# Konsep Dasar Interaksi Manusia - Komputer



Ratna Wardani  
Pertemuan #1

Pengantar



# Definisi IMK

- IMK adalah disiplin ilmu yang mempelajari tentang desain, evaluasi dan implementasi sistem komputer interaktif yang digunakan manusia dengan memperhatikan fenomena di lingkungan sekitarnya

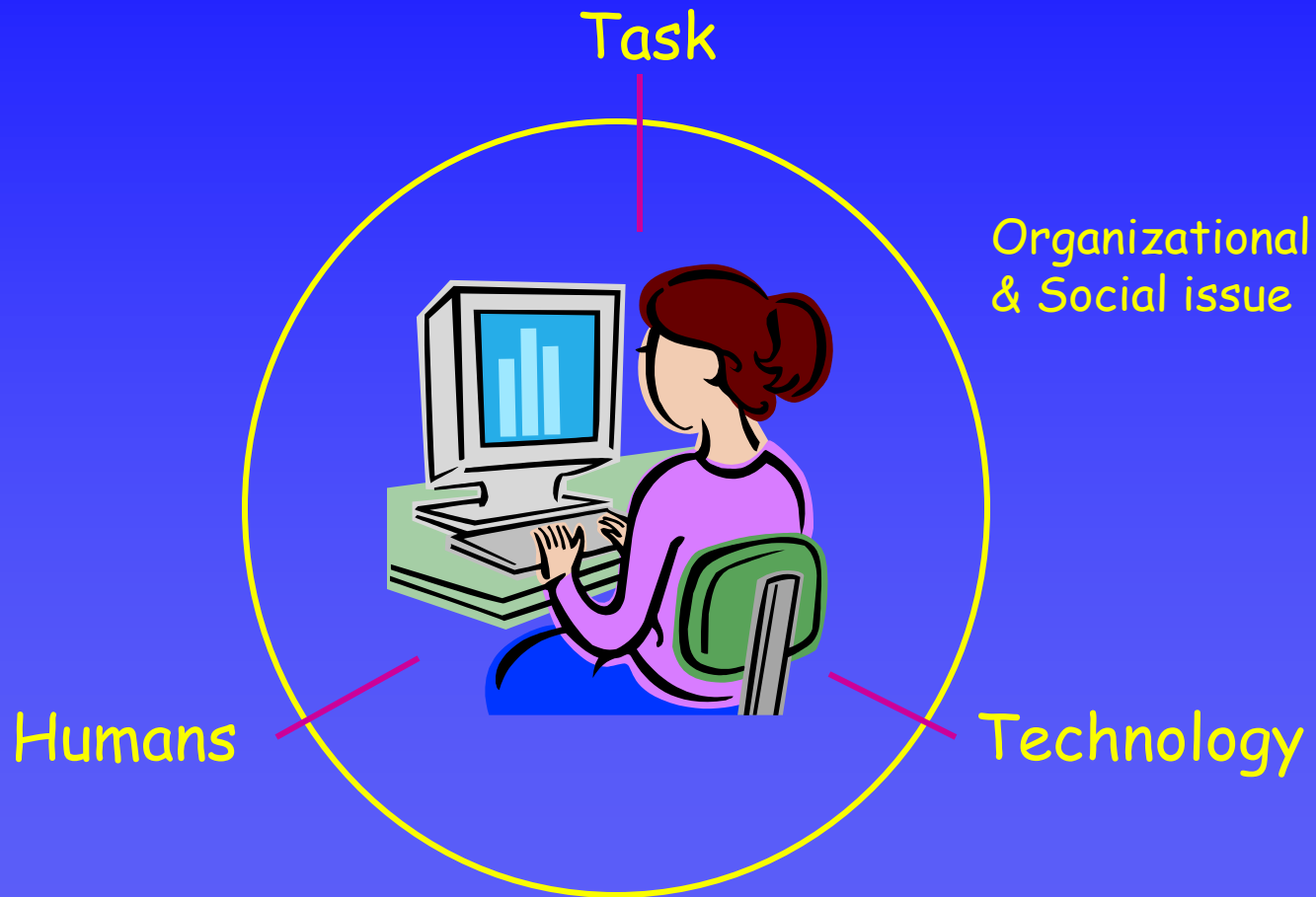
(SIGCHI / ACM @ <http://sigchi.org/cdg/cdg2.html> )

# Cakupan IMK

- Studi tentang penggunaan interface oleh manusia
- Pengembangan aplikasi baru untuk pengguna
- Pengembangan perangkat dan tool bagi pengguna

# Interaksi Manusia - Komputer ...?

---



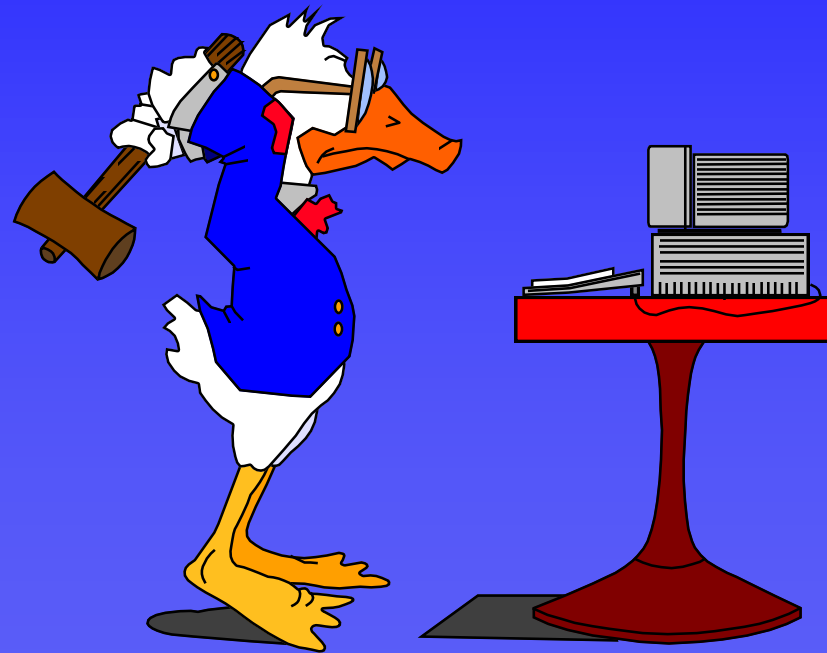
The marriage of Arts & Sciences

# Interaksi Manusia - Komputer ....?

---

- **Komponen Model IMK :**
  - Manusia → satu atau lebih
  - Pekerjaan → aktifitas bekerja, bermain, belajar, berkomunikasi dan lain-lain
  - Teknologi → bukan hanya desktop computer.....
  - Lingkungan → organisasi, aspek sosial, lingkungan fisik, dll
- **Model digambarkan dalam 3 level**
  - ❖ Level 1 → interaksi user dengan teknologi
  - ❖ Level 2 → interaksi untuk mencapai tugas khusus dalam lingkungan tertentu
  - ❖ Level 3 → aktifitas sejumlah orang membentuk lingkungan sosial dan organisasi

# User Interface



# UIs vs. HCI

---

- User Interface
  - Bagian dari aplikasi yang memungkinkan pengguna berinteraksi dengan komputer untuk menyelesaikan suatu pekerjaan
- Human Computer Interface
  - Desain, prototype, evaluasi dan implementasi dari user interface



# UIs yang Baik

---

- Mengurangi biaya coding
- Tingginya biaya bila terjadi problem interface
- Mengurangi kesalahan yang berakibat fatal
- interfaces yang baik meningkatkan penjualan produk
- Meningkatkan penggunaan komputer dalam suatu lingkungan

# Coding UIs

---

- Grafik pengembangan program user interface
  - 40 -90 % proses pengembangan digunakan untuk pengkodean user interface
  - Sebagian besar estimasinya 70 %
  - Jika tjd kesalahan, ulangi pengkodean
  - Jika tidak tepat, biaya dibebankan pengguna

# UIs Secara Ekonomi

---

- Dampak User interface yg baik :
  - Peningkatan produktifitas
  - Mengurangi biaya training
  - Mencegah kesalahan pengguna
  - Mengurangi mutasi karyawan
  - Kepuasan pengguna
  - Meningkatkan kualitas produk

# #1 Peningkatan Produktifitas

20 user

X 230 hari

X 100 layar per hari

X 10 detik per layar (penghematan)

---

= 1278 jam

atau 32 minggu

## #2 Pengurangan biaya Training

20 karyawan

X 2 sistem/aplikasi per tahun

X 2 1/2 hari per aplikasi

---

= 100 hari

atau 20 minggu

Biaya training lebih mahal daripada biaya untuk hardware dan software

# #3 Mencegah Kesalahan User

500 user

X 20 error per tahun

X 15 menit per error

---

= 2500 jam hilang  
atau 63 minggu

# #4 Peningkatan Kualitas Produk

- User tidak menghabiskan waktu pada penggunaan interface tapi lebih fokus pada problem solving, ex. Satu perintah untuk kompilasi dan eksekusi program
- Interface sesuai dengan cara fikir pengguna dalam menyelesaikan masalah. Ex. spreadsheet terlihat spt lembar accounting
- Interface meningkatkan solusi problem Ex., Tampilan data dengan berbagai cara

# Bahaya Kesalahan UIs #1

- Analisis transkrip telpon 911 untuk deteksi bom di Centennial Park ketika Olimpiade Atlanta menunjukkan bahwa diperlukan waktu 20 menit untuk pengiriman data:
  - Sistem pengiriman data harus memiliki alamat Centennial Park
  - Operator pengiriman tidak dapat menemukan seseorang yang mengetahui alamat tersebut
  - Bom diset meledak dalam 30 menit



# Bahaya Kesalahan UIs #2

- Kecelakaan pesawat tahun 1996 di gunung sekitar Colombia dengan korban seluruh penumpang
  - Pilot mengetikkan huruf "R" daripada menuliskan nama bandara
  - Sistem akan mencari nama bandara yang dimulai dengan huruf "R" yang ternyata adalah nama bandara yang salah
  - Kesalahan ini menyebabkan pesawat menabrak gunung

# Mengapa IMK

---

- Pada awalnya hanya pengguna tertentu yang bisa menggunakan komputer.
- Saat ini teknologi komputer sudah menjadi bagian dari kehidupan manusia (pervasive), sehingga interaksi dengan komputer harus intuitif dan jelas
- Keinginan manusia untuk mendapatkan kemudahan dalam menggunakan komputer untuk membantu pekerjaan manusia dalam menyelesaikan suatu persoalan.
- Perubahan teknologi → sistem harus didesain sesuai kebutuhan dan kemampuan orang yang menggunakannya dan bersifat transparan
- Desain interface tidak hanya mempertimbangkan aspek fisik saja, tapi juga harus memperhatikan kemampuan dan keterbatasan manusia

# IMK bagi IS

---

- **Personil Information Systems (IS)**
  - Memperkenalkan perkembangan aplikasi-aplikasi baru
    - Pengguna harus tahu bagaimana sistem beroperasi
  - Memperkenalkan sistem baru dalam organisasi
    - Pengguna harus dapat mengevaluasi ketergunaan sistem

# Tujuan IMK

---

- Menghasilkan sistem yang berguna dan aman.
- Mengembangkan dan meningkatkan keamanan, utilitas, efektivitas, efisiensi dan ketergunaan sistem.
  - ✓ Sistem tidak hanya H/W dan S/W, tapi juga lingkungan yang menggunakan dan dipengaruhi oleh teknologi komputer
  - ✓ Utilitas mengacu pada fungsionalitas sistem
  - ✓ Keamanan merupakan aspek/tujuan terpenting dari desain.
  - ✓ Ketergunaan merupakan konsep kunci dalam IMK, yang mengacu pada kemudahan dalam mempelajari dan menggunakan sistem.
- Memberikan feed-back yang diperlukan pengguna
- Membantu desainer dalam merancang sistem komputer yang user-friendly

# IMK dan Evolusinya

---

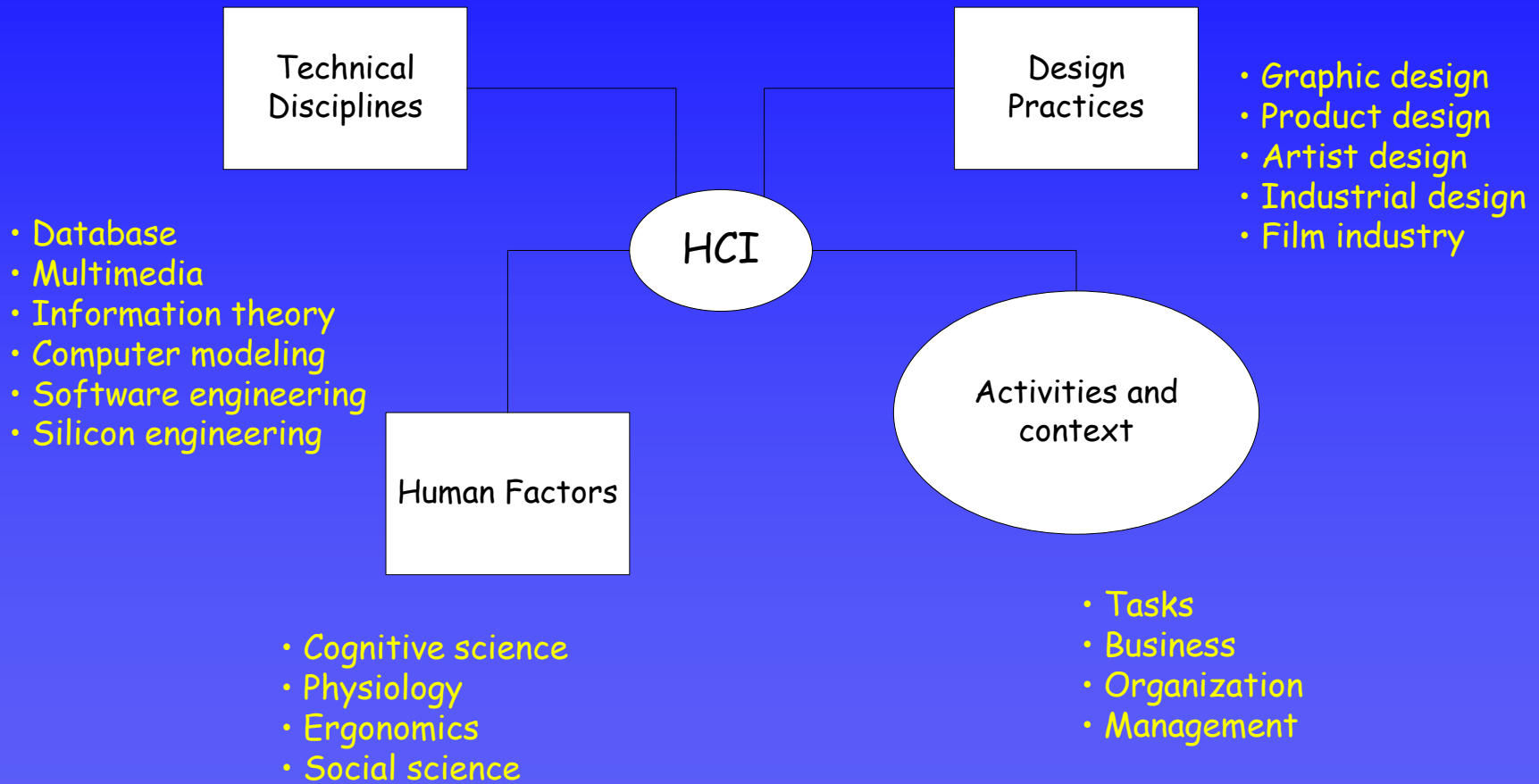
- IMK digunakan dalam konteks sosial dan organisasi
- Aplikasi yang berbeda digunakan untuk keperluan yang berbeda → diperlukan pembagian tugas antara manusia dengan mesin
- Aspek manusia :
  - ✓ Kemampuan mengolah informasi
  - ✓ Bahasa, komunikasi dan interaksi
  - ✓ ergonomis
- Aspek teknologi :
  - ✓ Teknik dialog, komputer grafis, gaya dialog, arsitektur dialog
- Tool dan teknik
- Evaluasi



IMK sebagai Multidisipliner

# Disiplin ilmu terkait

---



# Disiplin ilmu terkait

---

- ✓ Teknik elektronika dan ilmu komputer
  - ❖ Memberikan kerangka kerja untuk mendesain sistem IMK (desain aplikasi, desain GUI)
- ✓ Psikologi
  - ❖ Memberikan dasar teknik evaluasi formal untuk mengukur performansi dan opini terhadap sistem manusia-komputer (perilaku pengguna)
- ✓ Perancangan grafis dan tipografi
  - ❖ Memberi dasar dalam mendesaian interface
- ✓ Ergonomik
  - ❖ Berkaitan dengan aspek fisik untuk mendapatkan lingkungan kerja yang nyaman ketika pengguna bekerja dengan komputer



# Disiplin ilmu terkait

---

- ✓ **Antropologi**
  - ❖ Ilmu tentang manusia yang memegang peranan penting dalam sistem interaksi manusia-komputer
  - ❖ Memberi pandangan tentang cara kerja berkelompok sehingga masing-masing dapat memberi kontribusi sesuai keahlian
- ✓ **Linguistik**
  - ❖ Mempelajari tentang bahasa, aspek komputasi linguistik dan teori bahasa formal
- ✓ **Sosiologi**
  - ❖ Mempelajari pengaruh sistem manusia-komputer dalam struktur sosial (interaksi antar manusia, organisasi)

# Disiplin ilmu terkait

---

- ✓ *Artificial Intelligence*
  - ❖ Memberi pengetahuan dan metode dalam pengembangan sistem pakar yang dilengkapi interface cerdas
- ✓ *Engineering*
  - ❖ Memberi dasar software engineering
  - ❖ Penggunaan berbagai tool desain

# Faktor-faktor dalam IMK

---

- ✓ Faktor organisasi
  - ❖ Training, job design, politics, roles, work organizational
- ✓ Faktor lingkungan
  - ❖ Noise, heating, lighting, ventilation
- ✓ Faktor kesehatan dan keamanan
  - ❖ Stress, headaches, musculo-skeletal disorder
- ✓ Faktor pengguna
  - ❖ Motivation, enjoyment, satisfaction, personality, experience level
- ✓ Faktor kenyamanan
  - ❖ Seating, equipment layout

# Faktor-faktor dalam IMK

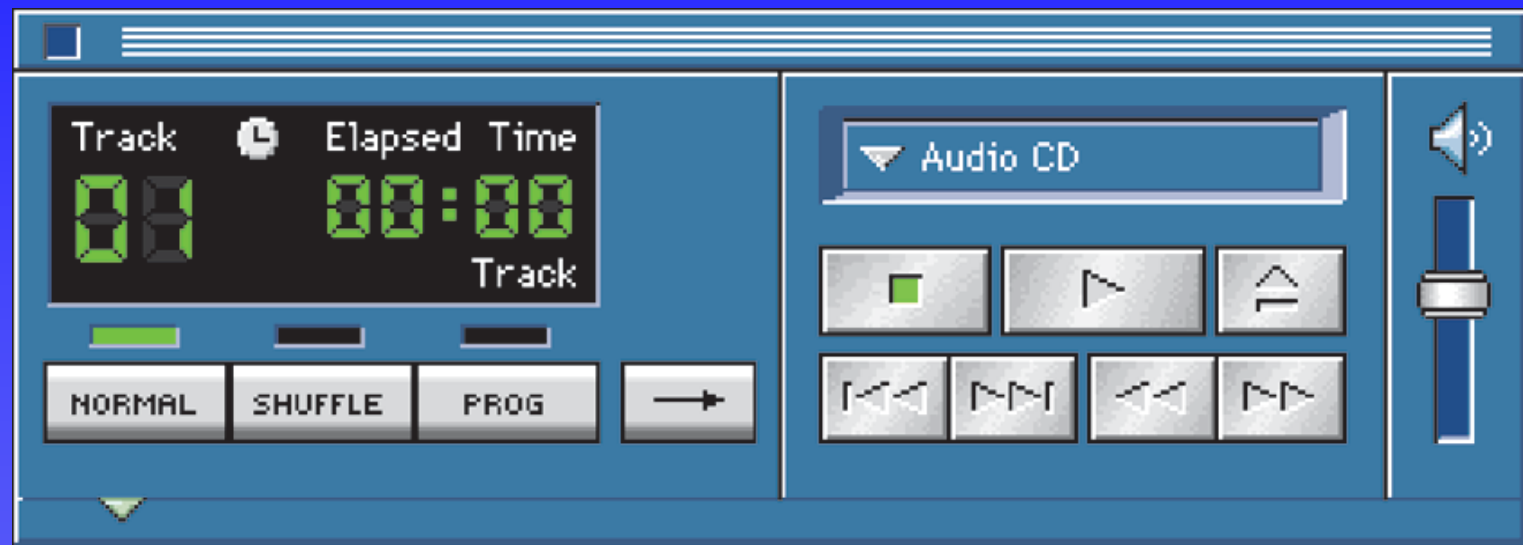
---

- ✓ **Faktor User-Interface**
  - ❖ Input device, output displays, dialogue structures, use of colour, icons, commands, graphics, natural language, 3-D, user support materials, multimedia
- ✓ **Faktor Pekerjaan**
  - ❖ Easy, complex, novel, task allocation, repetitive, monitoring, skills, components
- ✓ **Faktor constraint / batasan**
  - ❖ Costs, timescales, budgets, staff, equipment, building structure
- ✓ **Faktor fungsionalitas sistem**
  - ❖ H/W, S/W, application
- ✓ **Faktor produktivitas**
  - ❖ Increase output & quality, decrease cost, errors, labour requirements & production time, increase innovative & creative ideas leading to new product

# Contoh....

---

- ✓ GUI untuk audio CD Player (Apple Macintosh)

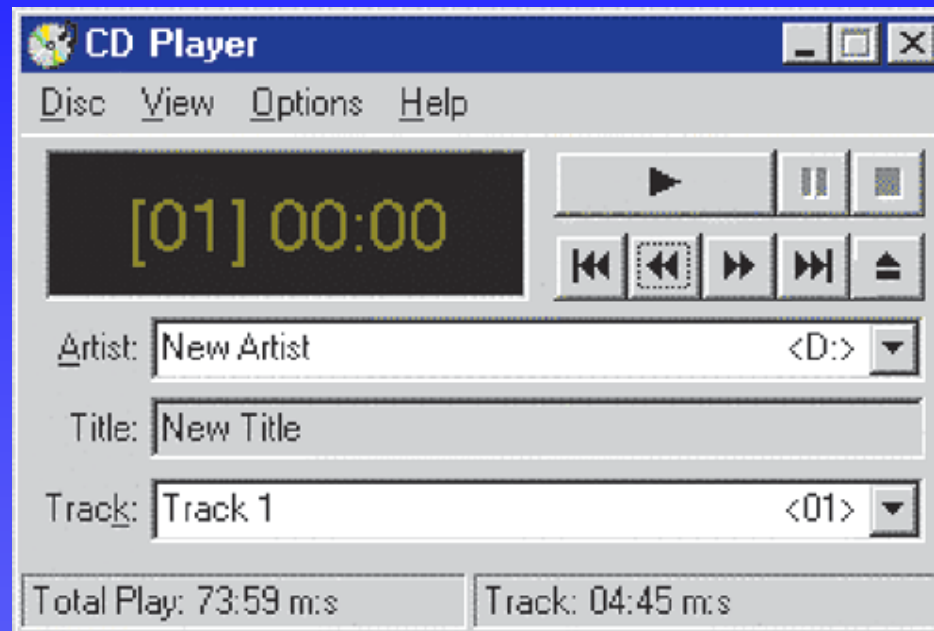


- ✓ Fitur mana yang familiar...?

# Contoh....

---

- ✓ GUI untuk audio CD Player (Windows)



- ✓ Fitur mana yang familiar...?

# Contoh....

---

- ✓ **Menu File dan Edit standar**

<b>File</b>		<b>Edit</b>	
<b>New</b>	<b>Ctrl+N</b>	<b>Undo</b>	<b>Ctrl+Z</b>
<b>Open . . .</b>	<b>Ctrl+O</b>	<b>Repeat</b>	<b>Ctrl+Y</b>
<b>Close</b>		<b>Copy</b>	<b>Ctrl+C</b>
<b>Save</b>	<b>Ctrl+S</b>	<b>Cut</b>	<b>Ctrl+X</b>
<b>Save As . . .</b>		<b>Paste</b>	<b>Ctrl+V</b>
<b>Page Setup . . .</b>		<b>Clear</b>	
<b>Print</b>	<b>Ctrl+P</b>	<b>Select All</b>	
<b>Print Preview</b>			
<b>Exit</b>			

- ✓ **Apakah SW komersial menawarkan desain GUI yang baik?**

Desain IMK





# Tujuan

- IMK didesain untuk meningkatkan kualitas interaksi manusia - komputer (fokus bukan pada interface maupun sistem)
- Diperlukan :
  - Teori tentang aktivitas pengguna
  - Teori transformasi
  - Pengetahuan tentang tujuan
  - Pengetahuan tentang kemampuan dan keterbatasan manusia
  - Pengetahuan tentang kemampuan dan keterbatasan komputer

# Tujuan (cont'd)

- Transformasi atau bukan...?
- Internet sebagai IMK
  - Perencanaan traveling
  - Cara berbelanja
- Email
  - Cara berkomunikasi
  - Cara penyelesaian pertentangan
- Database
  - Cara membuat katalog buku
  - Cara memilih resep

# Syarat Desain

- Syarat
  - ❖ User-centered → melibatkan sebanyak mungkin user
  - ❖ Integrasi pengetahuan dan keahlian dari berbagai disiplin ilmu
  - ❖ Proses desain yang iteratif → memenuhi persyaratan / kebutuhan user

# Metodologi

- Desain IMK menggunakan "star life-cycle" (berbeda dgn S/W engineering yang menggunakan pendekatan waterfall).
  - ❖ Pendekatan yang bersifat iteratif
  - ❖ Fokus desain ada pada evaluasi
  - ❖ Mendukung pengembangan bottom-up dan top-down
- Aktifitas desain :
  - ❖ Analisis user, tugas, lingkungan → analisis kebutuhan
  - ❖ Analisis teknis
  - ❖ Spesifikasi persyaratan
  - ❖ Desain, representasi desain dan coding
  - ❖ Prototyping dan penggunaan tool desain
  - ❖ Coding dan implementasi

# Problem Desain



# Problem Desain (cont'd)

- Untuk menghindari masalah desain, perlu diperhitungkan :
  - Siapa penggunanya
  - Aktivitas apa yang akan diselesaikan/dikerjakan
  - Dimana interaksi dilakukan
- Sesuaikan produk yang dihasilkan dengan aktivitas dan kebutuhan pengguna

# Problem Desain (cont'd)

- Tidak cukup training untuk pengembang interface
- Pengetahuan yg berbeda diperlukan untuk membuat desain interfaces yang baik
  - Sulit menemukan personil yang mampu
  - Tingginya permintaan thd personil yang memiliki kemampuan desain interface

# Problem Desain (cont'd)

- Cepatnya perkembangan teknologi
- Manajemen yg buruk - programmer tidak berintraksi dg user , desainer atau tim dan lainnya
- Kurangnya keterlibatan spesialis user-interface
- Programmer membiarkan arsitektur user-interface secara "default"
- Pengembang software kurang peduli terhadap aspek "usability" dan tidak tahu bagaimana mengukurnya



# Penggunaan Tools

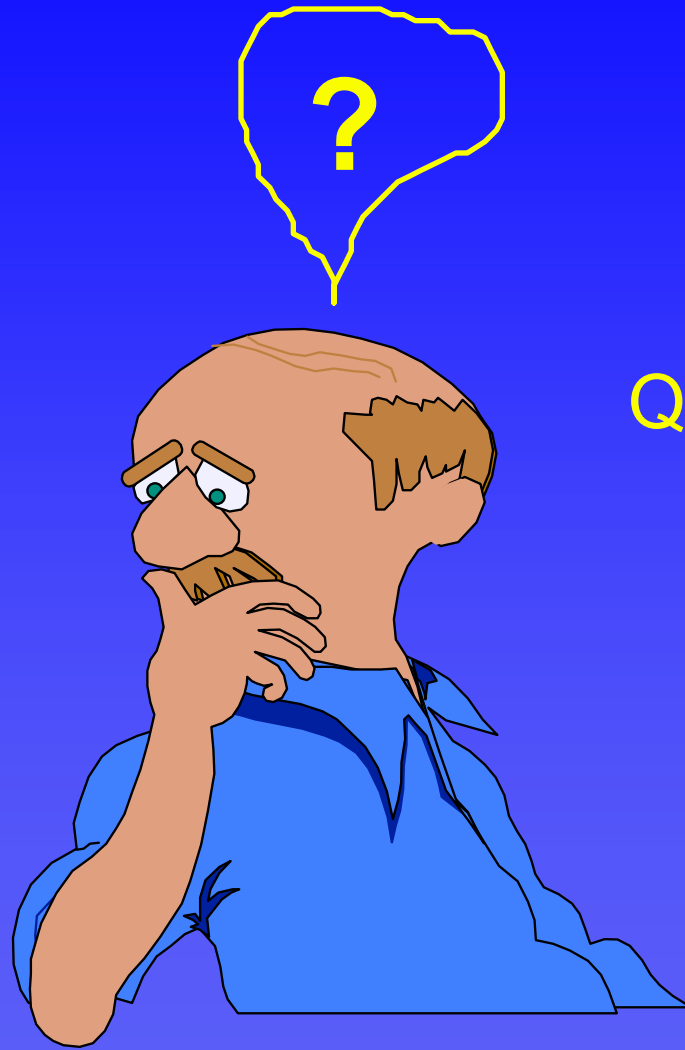
- S/W "user-friendly" → memiliki interface yang bagus, mudah dioperasikan, mudah dipelajari dan nyaman digunakan.
- Tantangan pengembangan interface:
  - ❖ Interface menangani sejumlah piranti kontrol
  - ❖ Interface harus memiliki kinerja yang ketat sehingga tidak terjadi keterlambatan respon
- Penggunaan tools (visual dBase, Visual BASIC, Borland Delphi, dll) untuk perancangan dan pengembangan interface dapat mengurangi waktu pengembangan sistem.

# Keuntungan Penggunaan Tools

- Interface yang dihasilkan menjadi lebih baik.
  - ❖ Prototipe dapat dibuat sebelum aplikasi ditulis
  - ❖ Mudah dimodifikasi
  - ❖ Aplikasi dapat memiliki lebih dari satu interface
  - ❖ Aplikasi yang berbeda memiliki interface yang konsisten krn dibangun dengan tools yang sama
  - ❖ Memberi ciri khusus pada aplikasi
  - ❖ Sinergi sejumlah ahli untuk memberi kontribusinya masing-masing (ahli grafis, psikolog, spesialis human factor)
- Kemudahan dalam penulisan program interface dan dalam pemeliharaan
  - ❖ Program interface terstruktur dan modular, karena dipisahkan dari aplikasinya.

# Keuntungan Penggunaan Tools

- Cont'd
  - ❖ Bersifat reusable karena piranti bantu menggabungkan bagian-bagian yang sama
  - ❖ Keandalan interface lebih tinggi karena di-generate secara otomatis dari aras spesifikasi yang lebih tinggi
  - ❖ Spesifikasi interface menjadi lebih mudah dinyatakan, divalidasi dan dievaluasi serta dimodifikasi
  - ❖ Dependensi terhadap piranti diisolasi dalam tools sehingga interface lebih mudah di-port ke berbagai aplikasi pada lingkungan yang berbeda



Question ?