

NOTASI DESAIN DIALOG

Interaksi Manusia dan Komputer

Ratna Wardani

A stylized silhouette of a mountain range in a darker shade of teal, located in the bottom right corner of the slide.

Pendahuluan

- ◆ **Dialog** : level sintaksis dari interaksi manusia dan komputer
- ◆ **Jenis Notasi Dialog** :
 - Diagramatik → mudah dibaca
 - Tekstual → mudah dilakukan analisis formal
- ◆ **Dialog berkaitan dengan** :
 - Semantik sistem → apa yang dilakukan sistem
 - Presentasi → bagaimana sistem ditampilkan

Pengertian Dialog

- ◆ **Dalam perancangan user interface** : dialog memiliki arti struktur percakapan antara user dengan sistem komputer
- ◆ **Bahasa komputer** terbagi menjadi 3 tingkatan:
 - **Leksikal** → bentuk icon pada layar, tombol yang ditekan. Pada bahasa manusia ekuivalen dengan bunyi dan ejaan suatu kata
 - **Sintaksis** → urutan dan struktur dari input dan output. Pada bahasa manusia ekuivalen dengan grammar atau tata bahasa suatu kalimat
 - **Semantik** → arti dari percakapan yang berkaitan dengan pengaruhnya pada struktur data internal komputer atau dunia eksternal. Pada bahasa manusia ekuivalen dengan arti dari partisipan dalam percakapan

Karakteristik Dialog User - Sistem Komputer

- ◆ Dalam user interface : dialog user - sistem komputer berada pada level sintaksis
- ◆ Karakteristik :
 - Partisipan harus menyebutkan dialognya dalam urutan yang jelas
 - Beberapa dialog telah ditetapkan sebelumnya
 - Beberapa bagian tertentu dari dialog dilakukan secara bersamaan (concurrenty)
 - Umumnya dialog berikutnya bergantung pada respon partisipan
 - Dialog tidak selalu mengakomodasi semua kejadian yang mungkin
 - Deskripsi dialog biasanya tidak menuju ke semantik / arti kata tapi pada level sintaksis

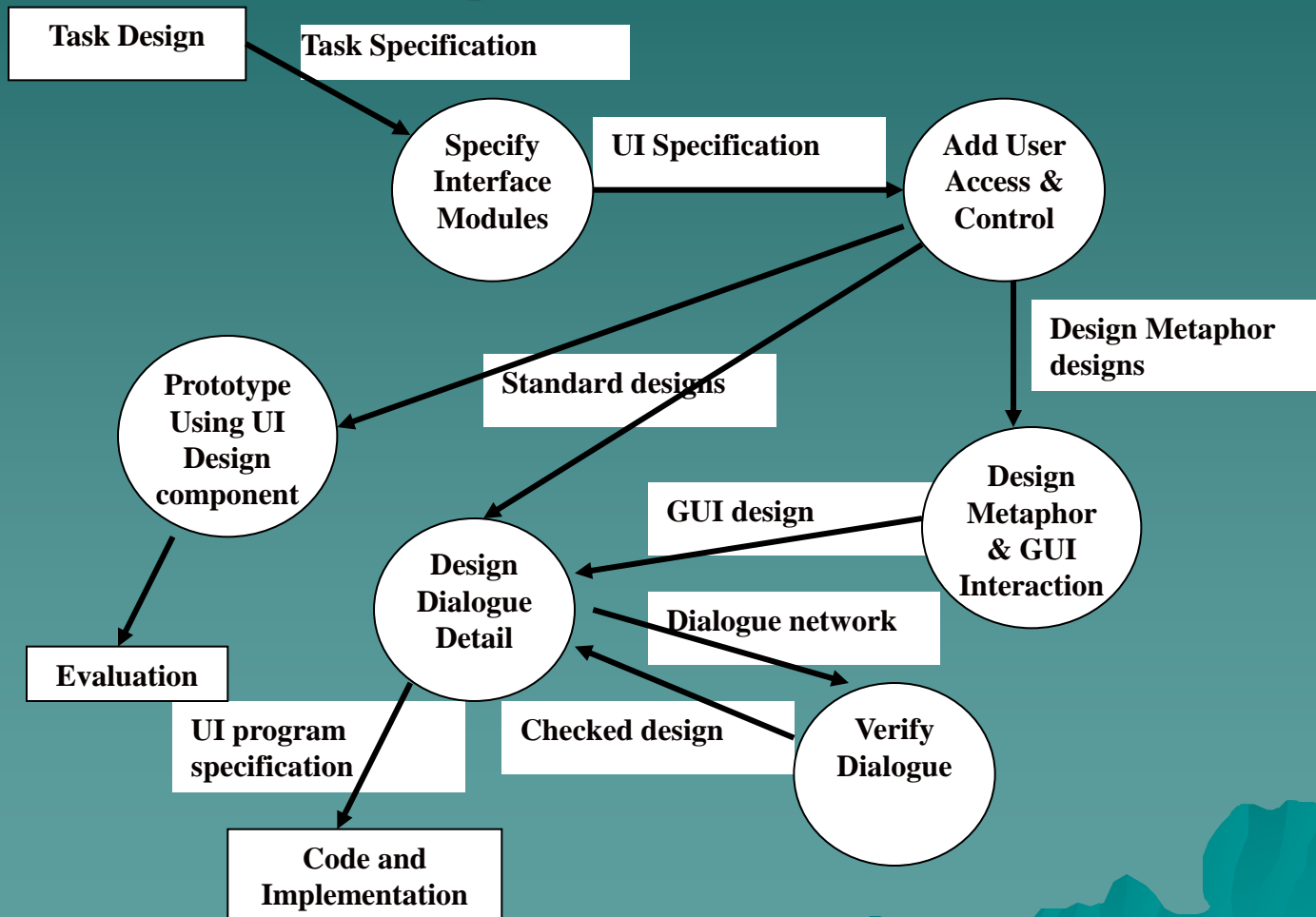
Desain Dialog

◆ Perlu diperhatikan :

- Rangkaian dialog merepresentasikan struktur tugas
- Beberapa rangkaian dialog tambahan digunakan untuk user support, misalnya Help system, tutorial sub-system
- Rangkaian dialog diurutkan sesuai struktur tugas

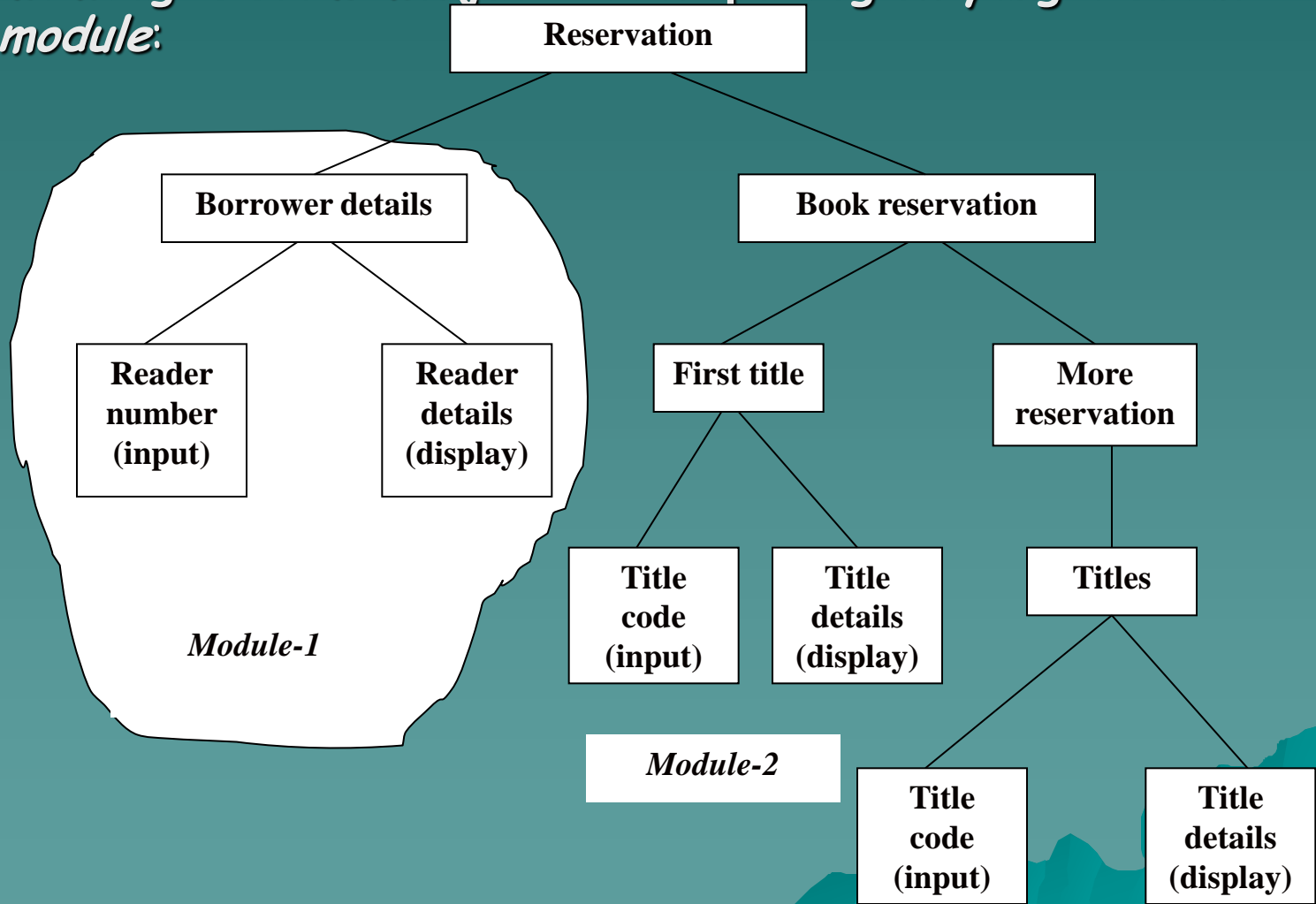
Desain Dialog

◆ DFD Desain Dialog :



Desain Dialog

- ◆ Prinsip yang digunakan dalam desain dialog adalah membagi sistem menjadi beberapa bagian yang disebut *module*:



Desain Dialog

- ◆ Pemisahan deskripsi dialog dari program secara keseluruhan, dikarenakan :
 - Agar mudah dianalisa
 - Pemisahan elemen-elemen interface dari logika program (semantik)
 - Apabila notasi dialog ditulis sebelum program dibuat, maka notasi tersebut bisa membantu desainer untuk menganalisis struktur dialog yang dibuat, dan menggunakan prototyping tool untuk menguji dialog
 - Notasi dialog merupakan salah satu cara bagi tim desainer untuk mendiskusikan desain dialog untuk diberikan kepada programmer aplikasi

Notasi Diagramatik

Notasi Diagramatik

◆ Notasi diagramatik :

- Paling sering digunakan dalam desain dialog
- Kelebihan : memungkinkan desainer melihat secara sekilas struktur dialog
- Kekurangan : sulit untuk menjelaskan struktur dialog yang lebih luas dan kompleks..

◆ Ragam notasi diagramatik :

- State Transition Networks (STN)
- Hierarchical State Transition Networks
- Harel's State Charts
- Flow Charts
- JSD Diagram

State Transition Networks

- ◆ Fokus pada state/kondisi

- ◆ Komponen :

 - Lingkaran*, menggambarkan "state" dari sistem

 - Tanda panah*, yang terdapat antara state; disebut juga transisi.

 - Tanda panah ini diberi label yang menjelaskan tentang tindakan user yang menyebabkan transisi dan response dari sistem.

- ◆ Contoh :

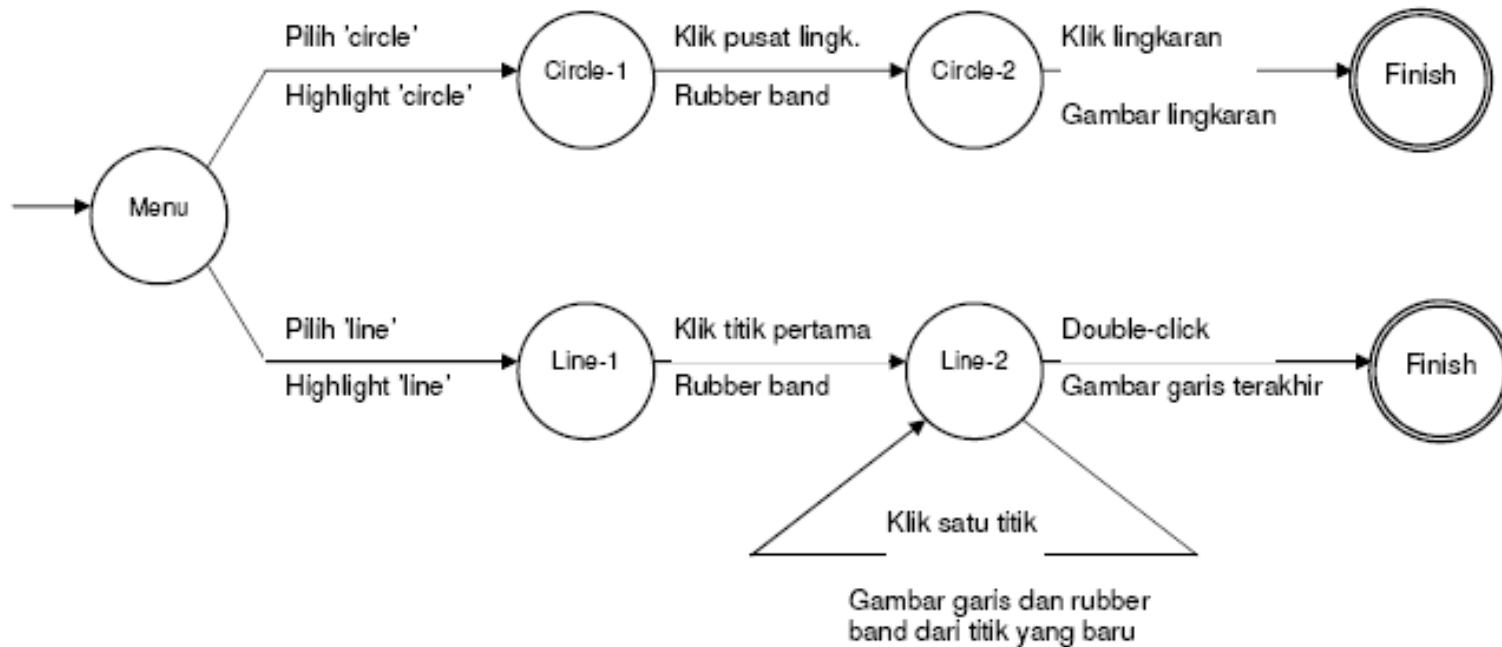
 - Menu drawing tool sederhana berbasis mouse
 - Disediakan 2 pilihan : "Circle" untuk menggambar lingkaran dan "Line" untuk menggambar polyline

State Transition Networks

- ◆ Terkait dengan dialog, STN dapat merepresentasikan :
 - Sequence (urutan) dari aksi yang dilakukan oleh user dan respon yang diberikan oleh sistem
 - Choice (Pilihan) bagi user
Contoh : Dari state "Menu" user bisa memilih "Circle" atau "Line"
 - Iteration (pengulangan)
Contoh : Pada state "Line-2" transisi dapat kembali ke "Line-2" jika user menambahkan titik baru pada polyline dan akan berpindah ke state "Finish" hingga user menekan double-click

State Transition Networks

◆ Contoh



Hierarchical State Transition Networks

◆ Karakteristik :

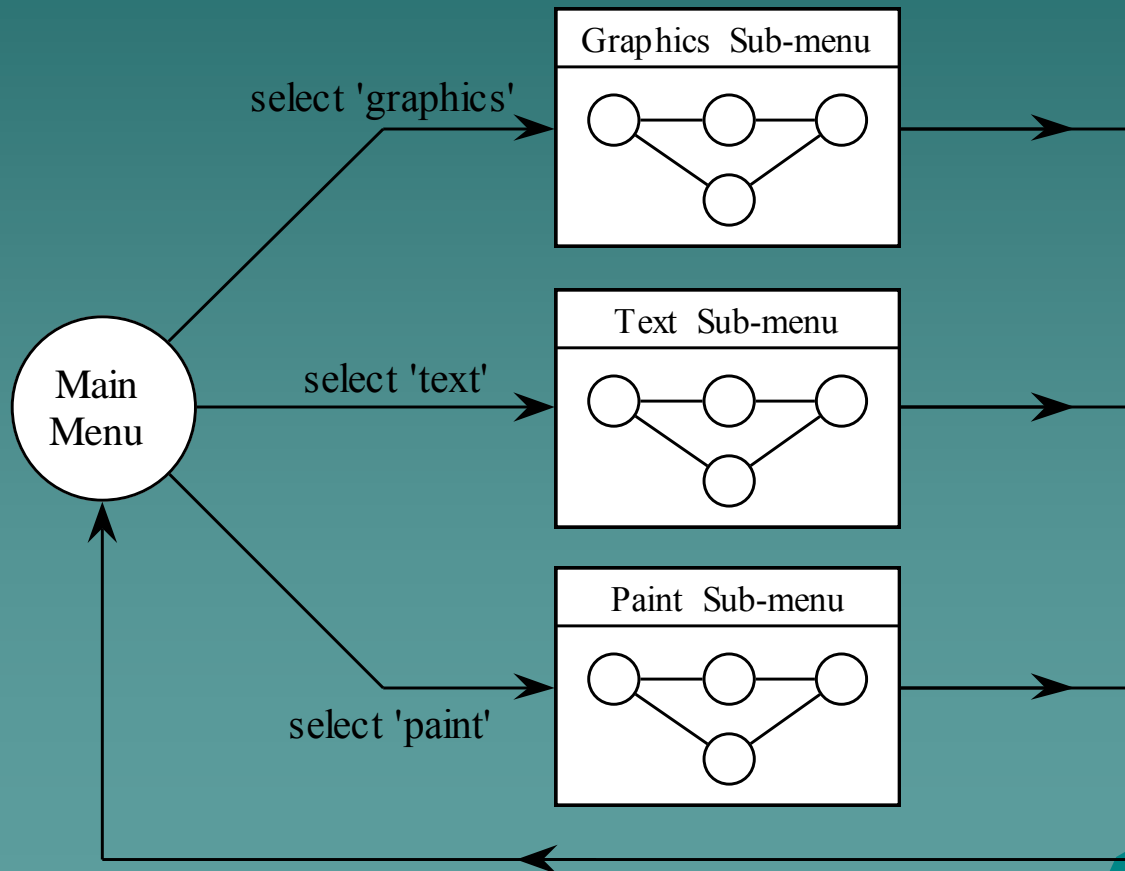
- Digunakan untuk deskripsi sistem yang lebih lengkap
- Mirip dengan STN, dengan fitur tambahan berupa gabungan state (Composite State)
- Digunakan untuk sistem-sistem yang besar

◆ Contoh :

- Pada drawing tool terdapat 3 sub-menu :
graphic, text dan paint

Hierarchical State Transition Networks

◆ Contoh :



Harel's State Charts

◆ Karakteristik :

- Digunakan untuk spesifikasi sistem reaktif yang kompleks secara visual
- Mampu mengakomodasi masalah seperti concurrency dan escape
- Struktur hirarki dalam satu diagram tunggal yang membagi elemen mana yang merepresentasikan state alternatif dan yang merepresentasikan aktifitas konkuren

◆ Contoh :

- Panel kendali televisi dengan 5 tombol kendali → On, Off, Mute, Sel dan Reset

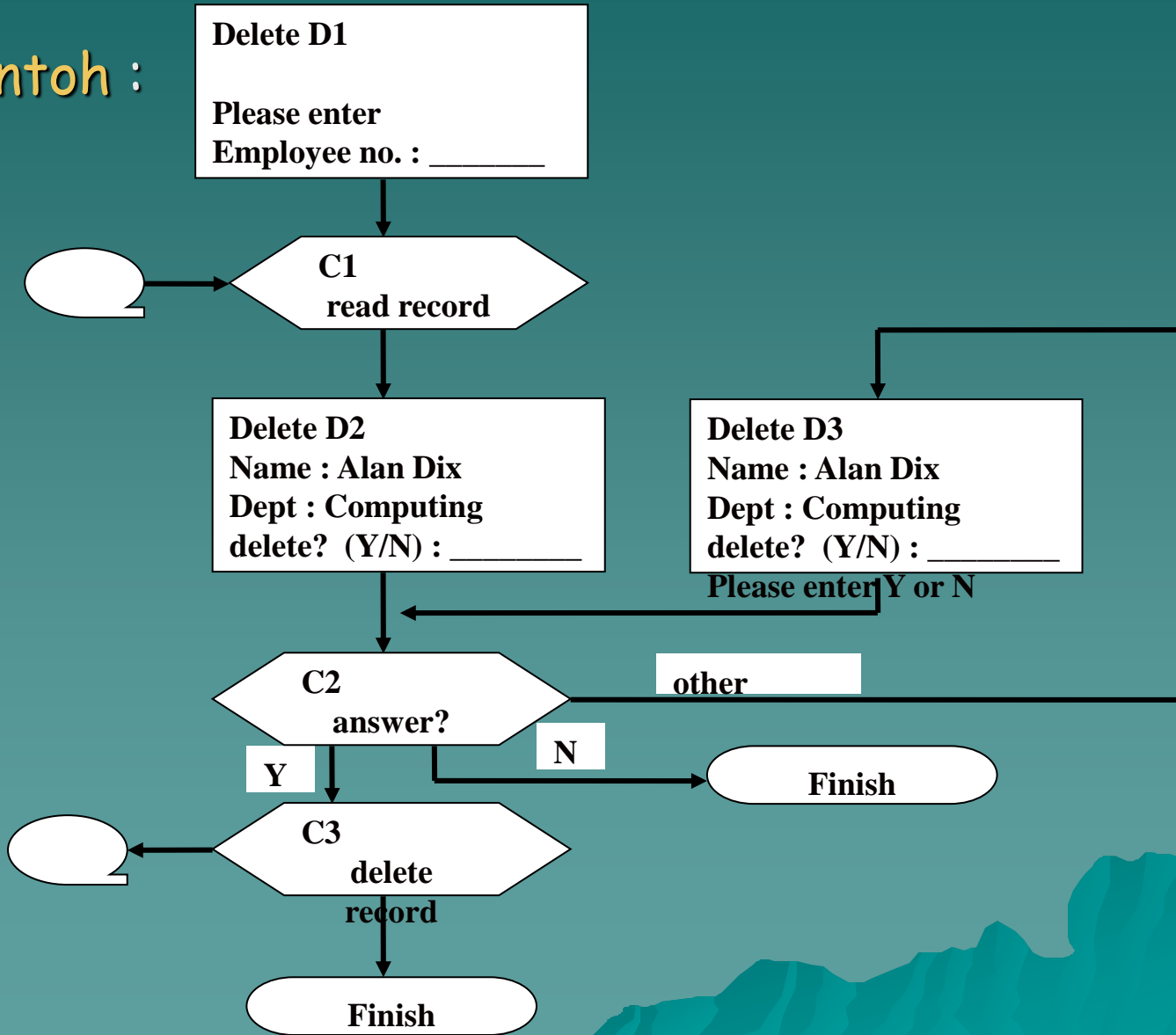
Flow Charts

◆ Karakteristik :

- Mendeskripsikan dialog yang sederhana
- Sederhana dan mudah dimengerti
- Merefleksikan sudut pandang programmer daripada user
- Simbol kotak mewakili proses atau keputusan

Flow Charts

◆ Contoh :



JSD Charts

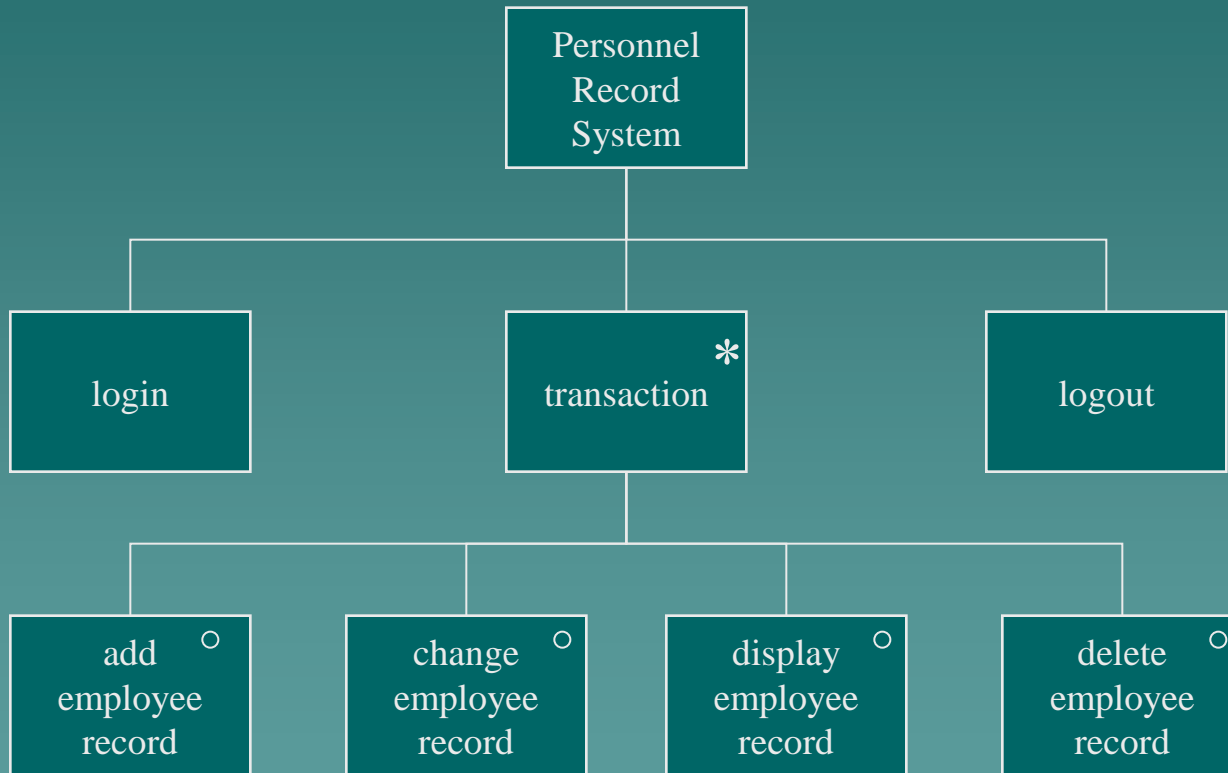
◆ Karakteristik :

- Jackson Structured Design digunakan untuk berbagai aspek dari analisis tugas dan desain dialog
- Digunakan untuk sistem yang sederhana, terbatas dan berbasis menu
- Kurang ekspresif
- Jelas



JSD Charts

◆ Contoh :



Concurrent dialogues - I

simple dialogue box

Text Style

bold

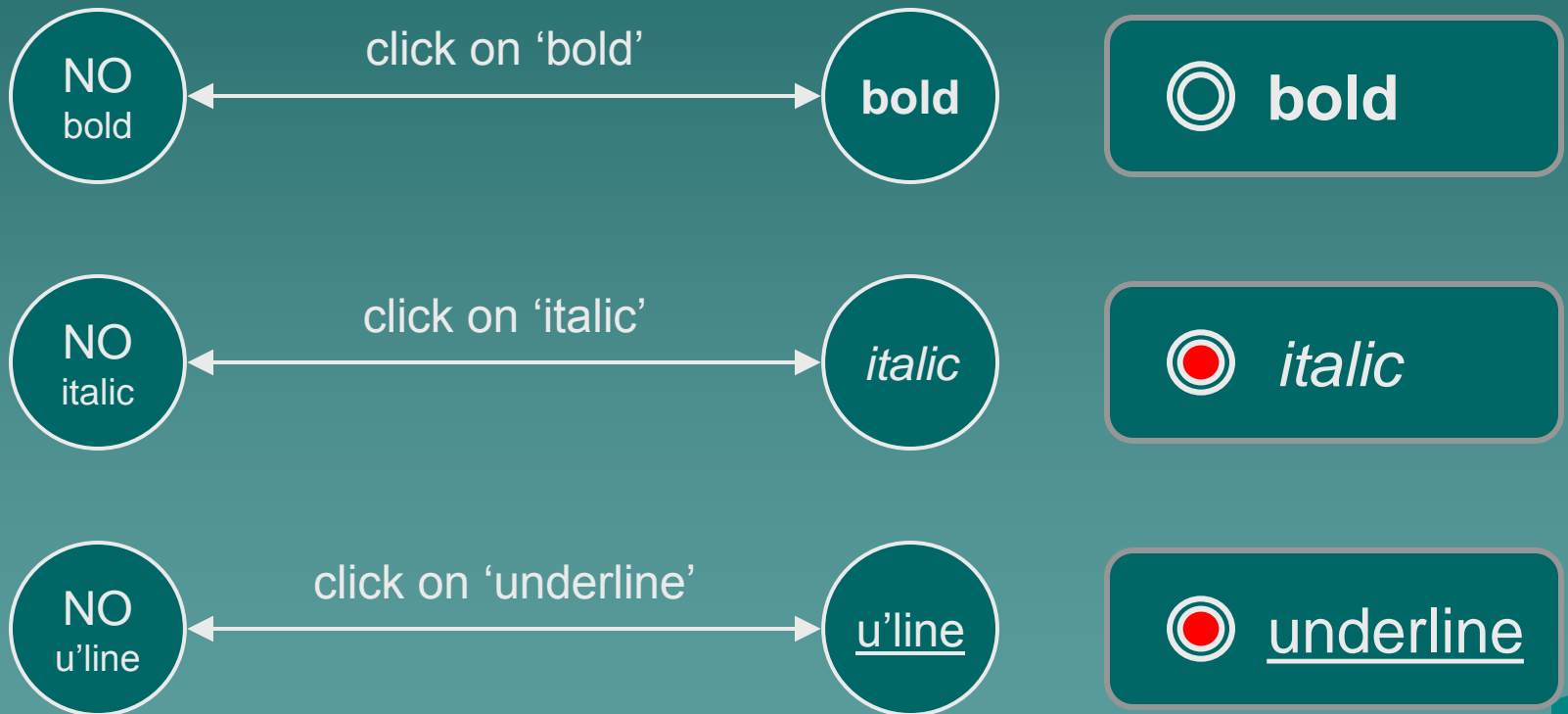
example

italic

underline

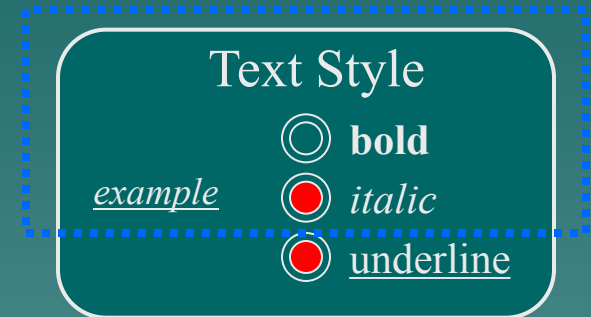
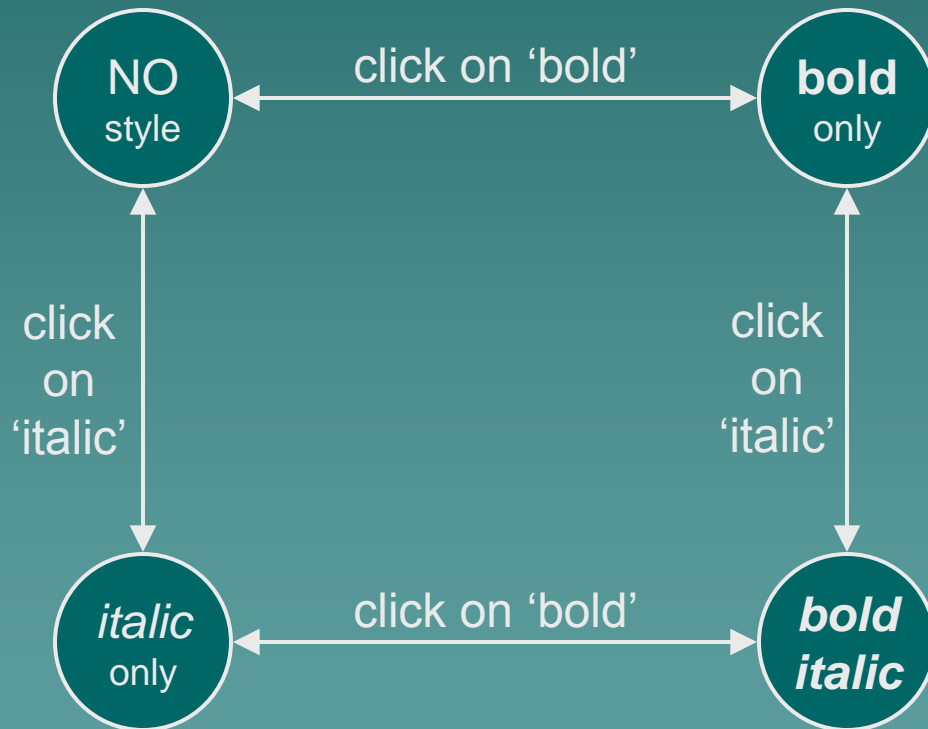
Concurrent dialogues - II

three toggles - individual STNs



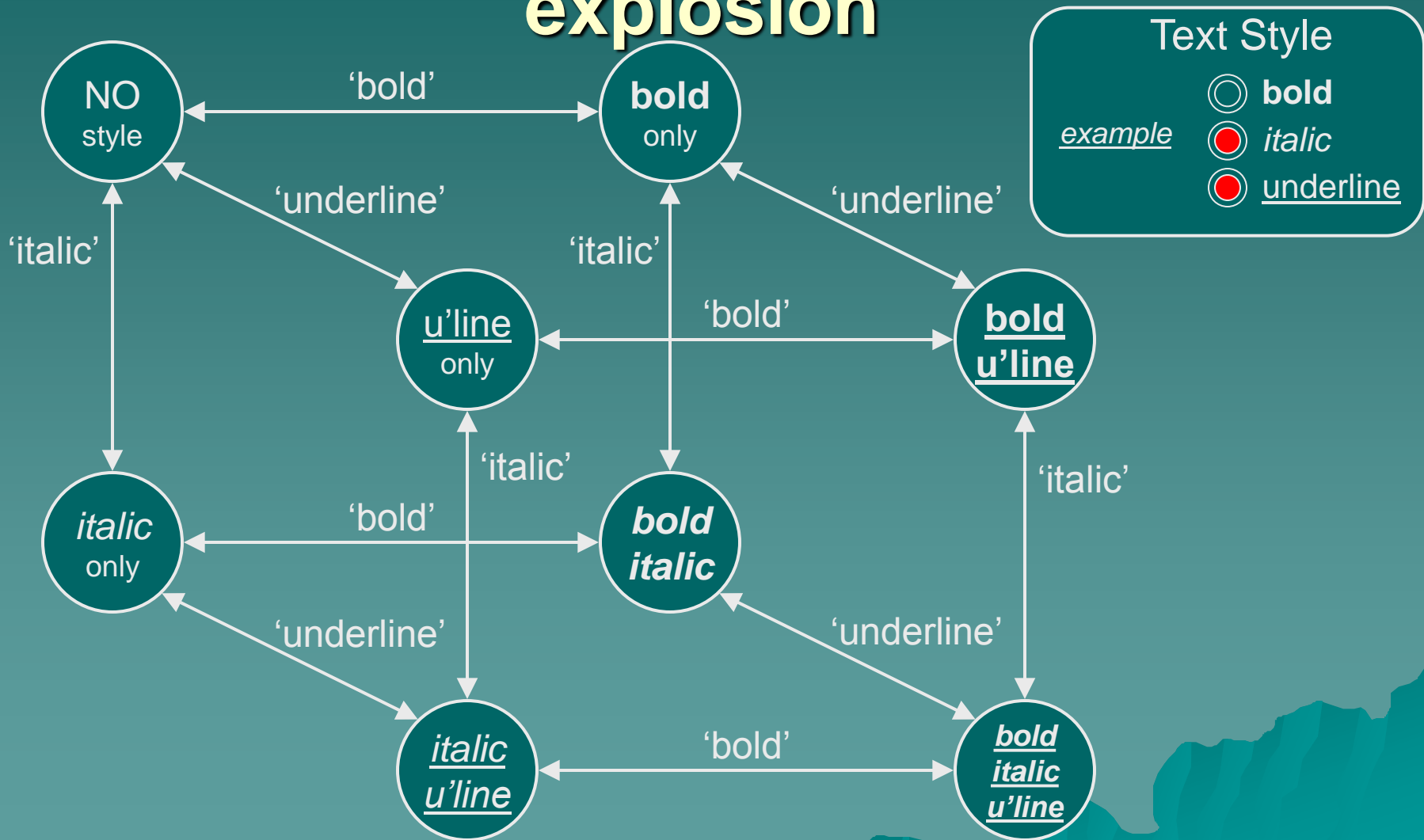
Concurrent dialogues - III

bold and italic combined



Concurrent dialogues - IV

all together - combinatorial explosion



Notasi Tekstual

Tata Bahasa / Grammar

- ◆ **Formal grammar :**
 - Notasi dialog tekstual
 - Fokus pada aksi yang dilakukan oleh user
- ◆ **Jenis :**
 - BNF (Backus-Naur Form)
 - Regular Expression
- ◆ **Karakteristik :**
 - Regular Expression banyak digunakan untuk mendeskripsikan kriteria pencarian tekstual yang kompleks
 - Regular Expression digunakan untuk analisis leksikal bahasa pemrograman

Tata Bahasa / Grammar

- ◆ Karakteristik :
 - Regular Expression dan BNF tidak digunakan untuk mere-presentasikan concurency dialog
 - Kelebihan : mudah diimplementasikan karena sudah ada tools yang memadai

Tata Bahasa / Grammar

◆ Contoh :

- RE untuk menggambar garis
select-line click click* double-click
tanda * berarti perulangan
- Pada BNF (Backus-Naur Form)
some-thing ::= thing + some-thing
tanda + berarti sequencing

Production Rules

- ◆ **Bentuk umum :**
 - *If condition then action*
- ◆ **Representasi lain**
 - Condition → action
 - Condition : action
 - Event: condition → action
- ◆ **Karakteristik**
 - Berorientasi pada event, state atau gabungan keduanya
 - Baik untuk task paralel, bukan task sequence

Production Rules

- ◆ Production Rules berorientasi event :
 - Aktivitas menggambar polyline
 - Production Rules:
 - Sel-line → start-line <highlight-line>
 - C-point start-line → rest-line <rubber band on>
 - C-point rest-line → rest-line <draw line>
 - D-point rest-line → <draw line> <rubber band off>
 - 3 jenis Event:
 - ◆ User events
 - ◆ Internal events
 - ◆ System response event

Production Rules

◆ Production Rules berorientasi event :

- User events

- ◆ Diawali huruf kapital → Sel-line (user memilih Line pada menu), C-point untuk single click dan D-point untuk double click

- Internal events

- ◆ Diawali huruf kecil, untuk mencatat history state dialog → rest-line adalah state stl titik pertama garis dipilih oleh user

- System response event

- ◆ Ditandai tanda < > → <draw line> adalah efek sistem yg dapat didengar atau dilihat

Prepositional Production System

- ◆ Berbasis State
- ◆ Attributes:
 - ◆ Mouse: { mouse-off, select-line, click-point, double-click }
 - ◆ Line-state: { menu, first, rest }
- ◆ Rules (feedback tidak ditunjukkan):
 - ◆ select-line → mouse-off first
 - ◆ click-point first → mouse-off rest
 - ◆ click-point rest → mouse-off
 - ◆ double-click rest → mouse-off menu
- ◆ Tidak baik untuk menggambarkan events

Semantics - raw code

- ◆ event loop untuk word processor
- ◆ dialogue description - sangat terdistribusi
- ◆ syntactic/semantic trade-off - tidak baik!

```
switch ( ev.type ) {
  case button_down:
    if ( in_text ( ev.pos ) ) {
      mode = selecting;
      mark_selection_start(ev.pos);
    }
    ...
  case button_up:
    if ( in_text ( ev.pos )
          && mode == selecting ) {
      mode = normal;
      mark_selection_end(ev.pos);
    }
    ...
  case mouse_move:
    if (mode == selecting ) {
      extend_selection(ev.pos);
    }
    ...
} /* end of switch */
```

Latihan

Suatu perusahaan koran akan mengeluarkan mesin penjual koran yang nantinya akan diletakkan di beberapa tempat umum, seperti bandara, stasiun KA. Untuk itu dibentuk satu tim yang terdiri dari ahli elektronik untuk menangani masalah hardware dan ahli interface designer untuk menangani masalah interface.

Dalam hal ini para interface designer benar-benar dituntut untuk mendesain suatu interface yang baik dan mudah dimengerti karena hasil penjualan sangat tergantung dari hal tersebut. Konsumen tidak akan membeli koran lewat mesin tersebut apabila mereka mengalami kesulitan atau kebingungan saat akan melakukan transaksi.

Sekarang anda diminta bertindak sebagai interface designer yang mendesain dialog antara konsumen dan mesin penjual koran.