

Pertemuan 13

Teknik dan Perangkat bantu
pengembangan sistem interaksi

Pokok Bahasan

1. Spesifikasi metode
2. Interface Building tools
3. Evaluasi dan kritik terhadap tools

1. Spesifikasi Methode

Spesifikasi Metode

- Multiparty Grammar
- Unified Modeling Language (UML)
- User Action Notation (UAN)

- Aset pertama dalam membuat perancangan adalah notasi yang baik untuk merekam dan mendiskusikan kemungkinan-kemungkinan.
 - **Spesifikasi dalam bahasa alami:** cenderung bertele-tele, samar-samar, dan membingungkan.
 - **Bahasa formal dan semiformal:** efektif untuk bahasa perintah.
 - **Menu-tree structures:** menunjukkan tata letak menu. Tidak menunjukkan seluruh aksi yang mungkin.

- **UML class diagram:** hubungan antar-elemen sistem. Lebih baik daripada menu-tree.
- **UML Statechart diagram:** sesuai untuk sistem interaktif.
- **User action notation (UAN):** notasi pendekatan untuk pengembangan sistem manipulasi langsung.

1. Multiparty Grammar

- Penggambaran interaksi dengan notasi seperti BNF (Backus-Naur Form).

- Contoh BNF:

```
<Telephone book entry> ::= <Name> <Telephone  
    number>  
<Name> ::= <Last name>, <First name>  
<Last name> ::= <string>  
<First name> ::= <string>  
<string> ::= <character>|<character> <string>  
<character> ::= A|B|C|...|W|X|Y|Z  
<Telephone number> ::= (<area code><exchange> -  
    <local number>)  
<area code> ::= <digit><digit><digit>  
<exchange> ::= <digit><digit><digit>  
<local number> ::= <digit><digit><digit><digit>  
<digit> ::= 0|1|2|3|4|5|6|7|8|9
```

- Pada *multiparty grammar* ada nonterminal yang diberi label untuk menyatakan pihak yang menghasilkan string (U: user; C: computer).
- Contoh: Proses log-in

```
<Session> ::= <U: Opening> <C: Responding>  
<U: Opening> ::= LOGIN <U: Name>  
<U: Name> ::= <U: string>  
<C: Responding> ::= HELLO [<U: Name>]
```

- *Multiparty grammar* efektif untuk rentetan perintah berorientasi teks yang berulang-ulang dipertukarkan, seperti pada terminal bank.

2. UML

Menurut Jaob Nielsen (2000) dalam Sistem Interaksi ada delapan aturan yang dapat digunakan dalam perancangan antarmuka pemakai yaitu :

1. Berusaha untuk konsisten
2. Meningkatkan frequent user menggunakan shortcut
3. Memberikan feedback yang informatif
4. Merancang dialog penutup
5. Memberikan prementif dalam kesalahan dan penanganannya
6. Memberikan pembalikan aksi yang mudah
7. Internal focus of control
8. Mengurangi beban ingatan jangka pendek.

UML class diagram menggambarkan hubungan antar elemen – elemen sistem lebih baik dari pada menu –tree atau HIPO

UML Staechart/activity diagram di rancang sesuai dengan sistem interaktif

User action notation (UAN) adalah penggunaan notasi pendekatan untuk pengembangan sistem manipulasi langsung, digunakan untuk mengatasi keanekaragaman dunia manipulasi langsung

Simbol yang digunakan sebagai berikut :

~[icon]	: bergerak menuju icon	icon-!	: icon kembali normal
~[x,y]	: menuju koordinat (x,y)	icon!	: icon berkedip
Mv	: tombol mouse ditekan	icon>~	: icon bergerak mengikuti ursor
M^	: tombol mouse dilepas	*	: dapat berulang (≥ 0 kali)
Icon!	: icon di highlight		

2. Interface-Building Tools

- Interface-building tools disebut juga:
 - Rapid Prototyper
 - User Interface Builder
 - User Interface Management System
 - User Interface Development Environment
 - Rapid Application Developer

- Fitur interface-building tools:
 - Kebebasan antarmuka pemakai
 - Metodologi dan notasi
 - Rapid prototyping
 - Dukungan perangkat lunak

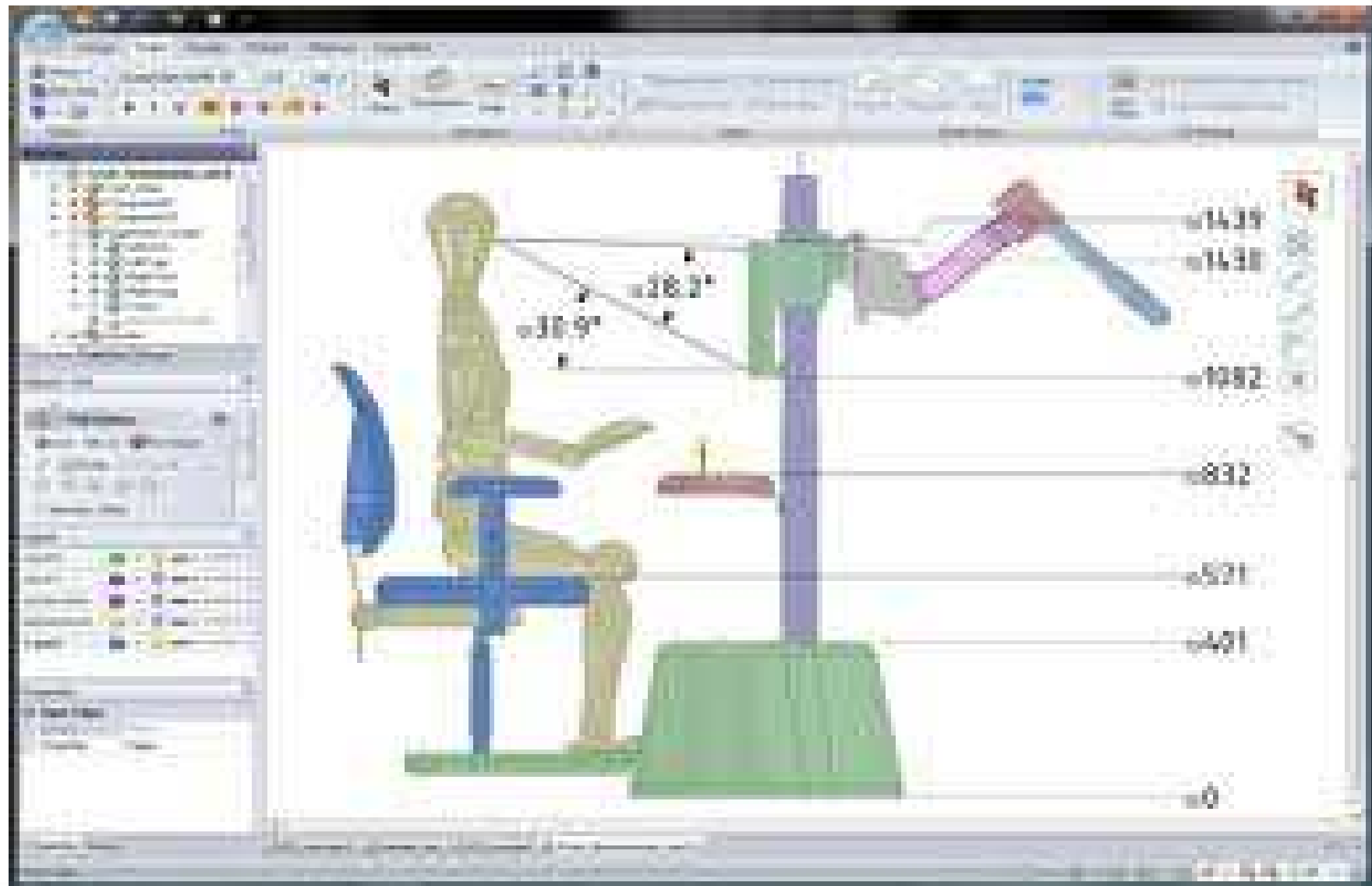
- **Kebebasan antarmuka pemakai**
 - Memisahkan perancangan antarmuka dari program internal.
 - Memungkinkan strategi *multiple user-interface*.
 - Memungkinkan dukungan *multi-platform*.
 - Memberi peranan arsitek antarmuka pemakai.
 - Menegakkan standar.

- **Metodologi dan notasi**
 - Mengembangkan prosedur perancangan.
 - Menemukan cara berbicara tentang perancangan.
 - Melakukan manajemen proyek.
- **Rapid prototyping**
 - Mencoba gagasan dengan sangat dini.
 - Uji, revisi, uji, revisi,
 - Mengikutsertakan end users, manajer, pelanggan.

- **Dukungan perangkat lunak**
 - Meningkatkan produktivitas.
 - Memberikan pemeriksaan kendala dan konsistensi.
 - Memfasilitasi pendekatan tim.
 - Memudahkan pemeliharaan.

- Design tools
- Software engineering tools
- Membuat gambaran awal dengan cepat penting di tahapan awal perancangan untuk:
 - Menjajaki berbagai alternatif;
 - Memungkinkan komunikasi dalam tim perancang;
 - Menyampaikan kepada klien seperti apa bentuk produk nantinya.

- Beberapa contoh design tools:
 - **CAI software:** Macromedia Authorware, IconAuthor, Quest.
 - **Multimedia construction tools:** HyperCard, Macromedia Director, Macromedia Flash.
 - **Slide presentation software:** Microsoft PowerPoint.
 - **Visual programming tools:** Microsoft Visual Basic (sekarang dalam Visual Studio .NET), Borland Delphi.
 - **Web design tools:** Macromedia Dreamweaver, Macromedia Fireworks.



Macromedia Dreamweaver MX - [Welcome to Global Car Rental (FinalSite/index.htm)]

File Edit View Insert Modify Text Commands Site Window Help

Title: Welcome to Global Car Rental

Insert Common Layout Text Tables Frames Forms Templates Characters Media Head Script Application

```

1 <html>
2
3 <head>
4 <title>Welcome to Global Car Rental</title>
5 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
6 <script language="JavaScript" type="text/JavaScript">
7 </script>

```

Global The International Car Rental Specialists

Home Customer Service Locations

Vintage

index.htm helloname

<head> <meta> 571 x 175 36K / 10 sec

Properties

Meta	Attribute: HTTP-equiv	Value: Content-Type
	Content: text/html; charset=iso-8859-1	

Design CSS Styles HTML Styles Behaviors

Apply Styles Edit Styles

- No CSS Style
- sectionHead
- tableDetail
- tableHeading

Code Application Files Answers

Learning Macromedia Dreamweaver MX

- What's New
- Readme
- Tutorials

Update Panel

Click Update to connect to macromedia.com and get the latest

Update

Software Tools

- Java:

```
class Test
{
    public static void main(String[] args)
    {
        for(int i = 0; i < args.length; i++)
            System.out.print(i == 0 ? args[i] :
                " " + args[i]);
        System.out.println();
    }
}
```

- JavaScript

```
<script language="JavaScript"
  type="text/javascript">
<!--
function square(i)
{
  return i * i;
}

document.write('The function returned:` +
  square(5) + `.`');
//-->
</script>
```