

Pertemuan 11

PERANCANGAN APLIKASI WEB

1. PENDAHULUAN

Perancangan aplikasi web memerlukan aktivitas teknis berupa:

1. Menetapkan tampilan pada web
2. Pembuatan rancangan estetika antarmuka pengguna
3. Pendefinisian struktur arsitektur aplikasi web secara keseluruhan
4. Pengembangan isi dan fungsional
5. Perencanaan navigasi

Pendahuluan (lanjutan)

Perancangan web sangat penting bagi *designer* karena:

1. Membuat model yang dapat dinilai kualitasnya dan dapat diperbaiki sebelum isi dan kode dibentuk
2. Membuat model sebelum pengujian dilakukan
3. Membuat model sebelum *end-user* yang berjumlah besar menggunakan aplikasi

Langkah-langkah *web design* dengan membuat:

1. Perancangan Isi

Dikembangkan selama tahapan analisis, dilakukan sebagai basis untuk penetapan objek-objek

2. Perancangan Estetika (Perancangan Grafis)

Membuat tampilan yang akan dilihat oleh *user*

3. Perancangan Arsitektural

Fokus pada struktur *hypermedia* untuk semua objek isi dan untuk semua fungsi pada aplikasi web

Langkah-langkah *web design* dengan membuat:

4. Perancangan Antarmuka

Menentukan tampilan dan mekanisme interaksi yang mendefinisikan *user interface*

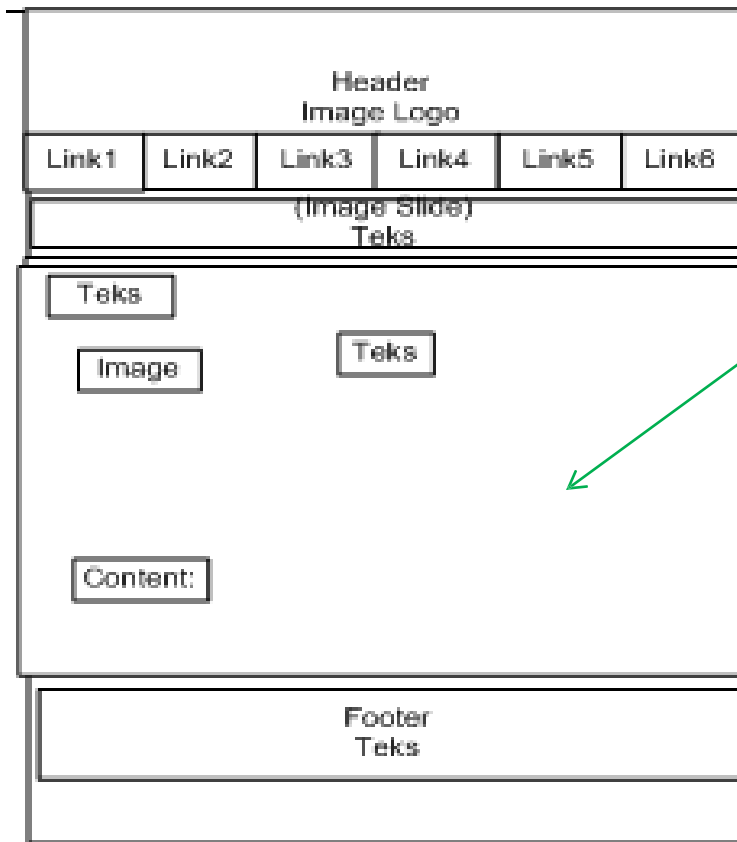
5. Perancangan Struktur Navigasi

Mendefinisikan bagaimana *end user* melakukan penelusuran untuk melintasi *hypermedia*

6. Perancangan Komponen

Merepresentasikan rincian struktur elemen-elemen fungsional aplikasi web

Contoh disain yang sulit dipahami, karena mengandung pernyataan yang umum berupa “Link”



Bagian ini tidak menjelaskan kebutuhan isi tampilannya

2. SIFAT-SIFAT APLIKASI WEB

- a. Kepadatan jaringan
- b. Keserempakan
- c. Jumlah pengguna yang tidak dapat diprediksi
- d. Kinerja
- e. Ketersediaan
- f. Digerakkan oleh data
- g. Peka terhadap isi
- h. Evolusi yang berkesinambungan
- i. Kesegeraan
- j. Keamanan
- k. Estetika

3. KUALITAS PERANCANGAN APLIKASI WEB

Atribut Kualitas:

1. Keamanan

Kemampuan aplikasi web dan lingkungan server untuk mencegah akses yang tidak sah, dan mencegah serangan-serangan yang berasal dari luar.

2. Ketersediaan

Pengukuran atas persentase waktu yang tersedia bagi aplikasi web untuk dapat digunakan oleh *user*.

3. Skalabilitas

Aplikasi web mampu mengakomodasi kebutuhan terhadap jumlah *end user* yang semakin bertambah

4. Waktu untuk masuk ke pasar

Ditinjau dari sudut pandang bisnis

Contoh disain Login dari sudut pandang keamanan

Masukan Login Anda

03:46:43 pm

17:12:16

Penggunaan
2 waktu yang
berbeda jauh

User

Pilih

Password

Login

Sebaiknya User tidak
memilih dari
beberapa data yang
disediakan

User tidak dapat
melakukan navigasi
lainnya kecuali
LOGIN

Kualitas Aplikasi Web

1. Kemudahan Penggunaan

- a. Kemudahan pemahaman situs global
- b. Umpan balik dari *user* dan fitur-fitur bantuan
- c. *User interface* dan fitur-fitur estetika
- d. Fitur-fitur khusus

2. Fungsionalitas

- a. Kemampuan pencarian dan penerimaan
- b. Fitur-fitur navigasi dan perambahan (*browsing*)
- c. Fitur-fitur aplikasi yang berhubungan dengan lingkungan

3. Keandalan

- a. Pembetulan pemrosesan tautan (*link*)
- b. Pemulihan dari kesalahan
- c. Validasi dan pemulihan pada *user*

Sistem memberikan umpan balik kepada user jika melakukan kesalahan



Kualitas Aplikasi Web (Lanjutan)

4. Efisiensi

- a. Kinerja waktu tanggap aplikasi web
- b. Kecepatan pembentukan halaman-halaman
- c. Kecepatan penggambaran grafik-grafik

5. Kemudahan Pemeliharaan

- a. Kemudahan untuk dilakukan koreksi
- b. Keamanan aplikasi web untuk beradaptasi
- c. Kemudahan aplikasi web untuk dikembangkan

Sasaran perancangan web yang baik:

1. Kesederhanaan (*Simplicity*)

Fungsi-fungsi mudah digunakan dan mudah dipahami

2. Konsisten (*Consistency*)

Konstruksi perancangan isi dibuat secara konsisten. Misalnya: jenis font yang sama pada semua dokumen teks, skema warna dan gaya yang konsisten

3. Identitas (*Identity*)

Estetika, *user interface*, dan perancangan navigasi harus konsisten dengan lingkungan aplikasi untuk apa aplikasi web itu dibuat atau dikembangkan.

4. Ketangguhan (*Robustness*)

User pada umumnya mengharapkan isi dan fungsi yang relevan terhadap kebutuhan *user*.

Sasaran perancangan web yang baik:

5. **Kemudahan melakukan navigasi dalam aplikasi**
Aplikasi web seharusnya dirancang sedemikian rupa sehingga tampilannya intuitif dan hasilnya dapat dengan mudah diramalkan.
6. **Daya tarik visual (*Visual Appeal*)**
Tampilan isi, rancangan *user interface*, pengaturan warna, keseimbangan yang harus terjadi di antara teks, grafik dan media lainnya, mekanisme navigasi sangat memiliki kontribusi pada daya tarik visual
7. **Kompatibilitas (*Compatibility*)**
Aplikasi web akan digunakan pada berbagai jenis lingkungan eksekusi aplikasi yang berbeda (*hardware, OS, browser, dan koneksi internet*)

Contoh desain web

HEADER																						
Home	Product	Sign In	Registration	Confirmation	How To Order	About Us																
<div><div>[CONFIRMATION PAYMENT]</div><table><tr><td>No. booking</td><td><input type="text"/></td></tr><tr><td>Sender's name</td><td><input type="text"/></td></tr><tr><td>Total transfer</td><td><input type="text"/></td></tr><tr><td>Information</td><td><input type="text"/></td></tr><tr><td>Bank name</td><td><input type="text"/></td></tr><tr><td>Date of transfer</td><td><input type="text"/></td></tr><tr><td>Picture</td><td><input type="text"/></td></tr><tr><td colspan="2"><input type="button" value="Send"/></td></tr></table><p>Information Teks</p></div>							No. booking	<input type="text"/>	Sender's name	<input type="text"/>	Total transfer	<input type="text"/>	Information	<input type="text"/>	Bank name	<input type="text"/>	Date of transfer	<input type="text"/>	Picture	<input type="text"/>	<input type="button" value="Send"/>	
No. booking	<input type="text"/>																					
Sender's name	<input type="text"/>																					
Total transfer	<input type="text"/>																					
Information	<input type="text"/>																					
Bank name	<input type="text"/>																					
Date of transfer	<input type="text"/>																					
Picture	<input type="text"/>																					
<input type="button" value="Send"/>																						
INFORMASI				About Us																		
				How To Order																		
				Contcat us																		
FOOTER																						

4. PERANCANGAN ANTARMUKA

- Salah satu tantangan membuat *user interface* adalah bagaimana caranya *user* masuk ke aplikasi.
- Sasaran-sasaran *user interface* adalah untuk:
 1. Menetapkan suatu jendela yang konsisten untuk meletakkan isi-isi dan fungsionalitas yang disediakan oleh *user interface*
 2. Memandu *user* melalui serangkaian interaksi dengan aplikasi web yang dikembangkan
 3. Mengorganisasikan pilihan-pilihan navigasi dan isi-isi yang dapat dilihat *user* yang dapat berupa menu navigasi, *icon* grafis, dan gambar-gambar grafis

Contoh desain web berbasis android



5. PERANCANGAN ESTETIKA

Sering juga disebut Perancangan Grafis, yang merupakan tambahan artistik yang sering digunakan untuk melengkapi aspek-aspek teknis dari perancangan aplikasi web.

Tata letak yang baik pada perancangan *interface*:

1. Jangan mengisi bagian dari halaman web dengan informasi yang akhirnya sulit untuk mengidentifikasi informasi tersebut
2. Lakukan penekanan pada isi yang merupakan alasan utama bagi *user* untuk masuk ke aplikasi web
3. Lakukan pengelompokkan fitur navigasi, isi, dan fungsi
4. Jangan perluas bagian aplikasi dengan penggunaan *scrollbar*, sebaiknya kurangi isi yang jumlahnya banyak
5. Sesuaikan resolusi layar dan ukuran jendela *browser*

Contoh

Sebaiknya tidak menggabungkan semua kebutuhan penggunaan aplikasi, pisahkan antara pemesanan dengan pengembalian mobil, serta penggunaan warna teks yang seharusnya mudah dibaca

No. Transaksi Sewa

No. Transaksi Kembali

01:39:02

Kode Pegawai

Nama Pegawai

Data Penyewa

Kode Penyewa

Nama Penyewa

No. KTP

Data Mobil

No. Polisi

Merk Mobil

Model Mobil

Warna

Tarif Sewa

Rp

/hari

Rp

/minggu

Rp

/bulan

Denda Sewa

Rp

/hari

Status

Ada

Tidak Ada

Paket Sewa

Harian

Mingguan

Bulanan

Cari No. Transaksi

CARI

PROSES PENGEMBALIAN

KEMBALIKAN MOBIL

TAMBAH

SIMPAN

BATAL

No. Polisi	Kode Supir	Merk Mobil	Tarif Sewa	Lama Sewa	Jumlah Tarif	Ov

Data Supir

Kode Supir

No. SIM

Nama Supir

Status

Ada

Tidak Ada

Lama Sewa

Tgl Sewa

Jam Sewa

Tgl Kembali

Jam Kembali

Overtime

Jumlah Tarif

Uang Muka

Sisa Bayar

Total Bayar

Uang Bayar

Kembali

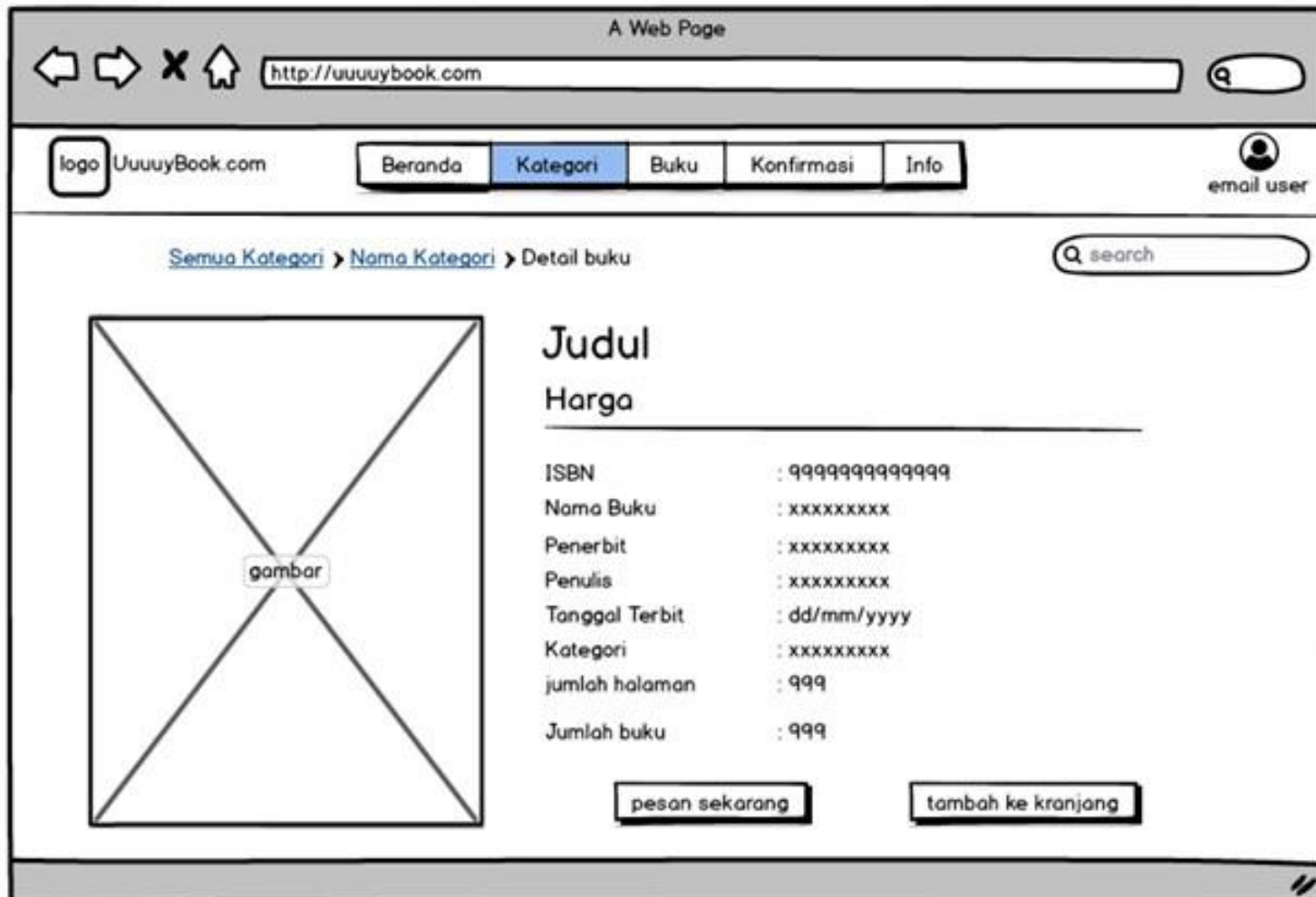
Sewa Lagi

6. PERANCANGAN ISI

- Hubungan objek isi dengan objek isi lainnya adalah sebagai bagian dari suatu model kebutuhan untuk aplikasi web.
- Permasalahan yang terjadi pada perancangan isi jika jumlah objek isi yang digabungkan untuk membentuk halaman web tunggal merupakan fungsi dari kebutuhan *user*, yang dibatasi oleh kecepatan pengunduhan koneksi ke internet, juga dibatasi oleh besarnya ukuran jendela monitor yang digunakan *user*.

Contoh

Tampilan rancangan detail isi buku (objek)



7. PERANCANGAN ARSITEKTURAL

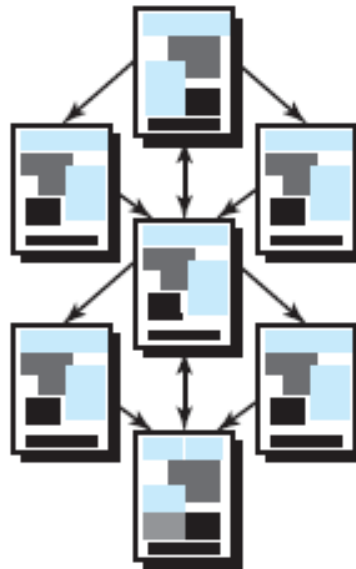
- Perancangan arsitektur web terkait dengan sasaran untuk aplikasi web, terkait dengan isi yang akan ditampilkan, *user*, dan navigasi.
- Arsitektur isi pada umumnya fokus pada bagaimana objek-objek isi distrukturkan agar layak untuk dipresentasikan kepada *user* dan menarik untuk ditelusuri.
- Aplikasi web distrukturkan untuk dapat mengelola interaksi *user* dengan aplikasi web, bagaimana menangani pekerjaan proses internal, bagaimana melakukan pengaturan navigasi, serta bagaimana menampilkan isinya.

A. Arsitektur Isi

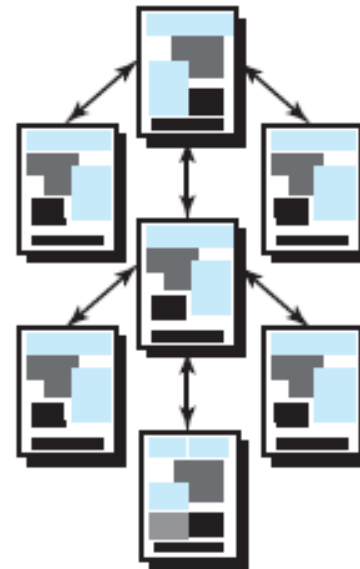
1. **Struktur Linier:** dilakukan saat interaksi *user* dengan aplikasi web secara umum memperlihatkan urutan yang dapat diramalkan. Contoh: urutan pemasukan pemesanan produk dimana informasi tertentu harus dalam urutan yang bersifat spesifik.



Linear



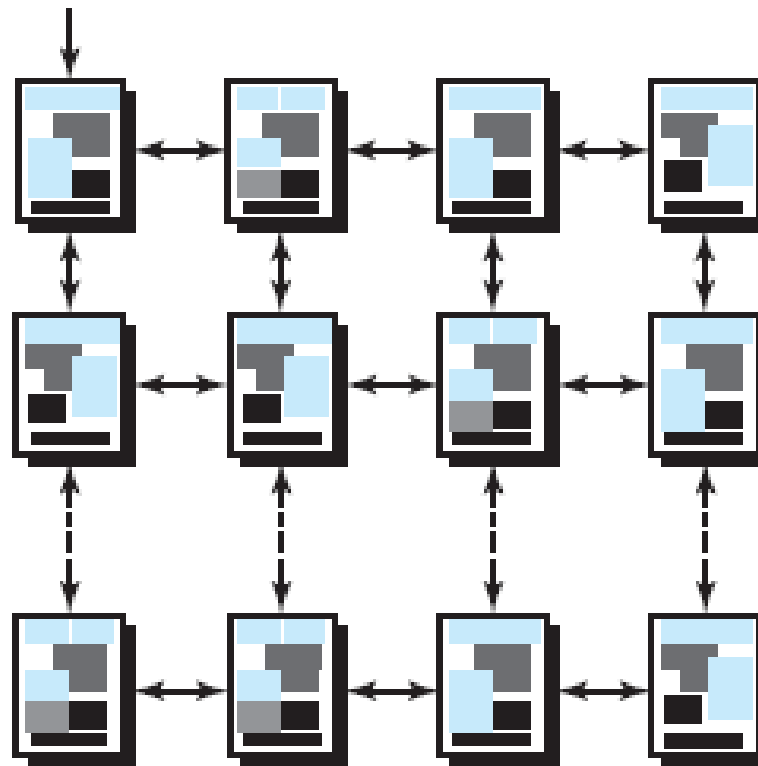
Linear
with
optional flow



Linear
with
diversions

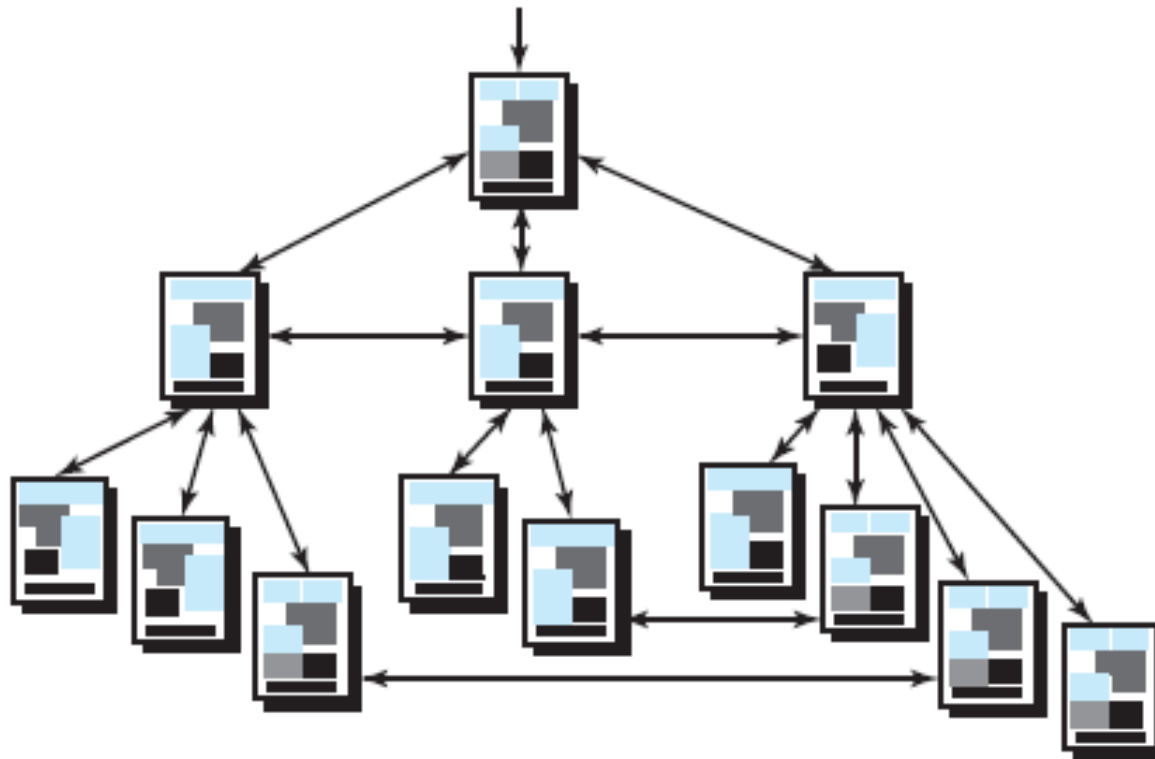
Arsitektur Isi (lanjutan)

2. **Struktur *Grid*:** arsitektur yang diterapkan saat isi aplikasi web dapat diorganisasikan menjadi 2 dimensi atau lebih. Dimensi horizontal merepresentasikan jenis-jenis produk, dan dimensi vertikal merepresentasikan penawaran yang disediakan oleh penjualnya.



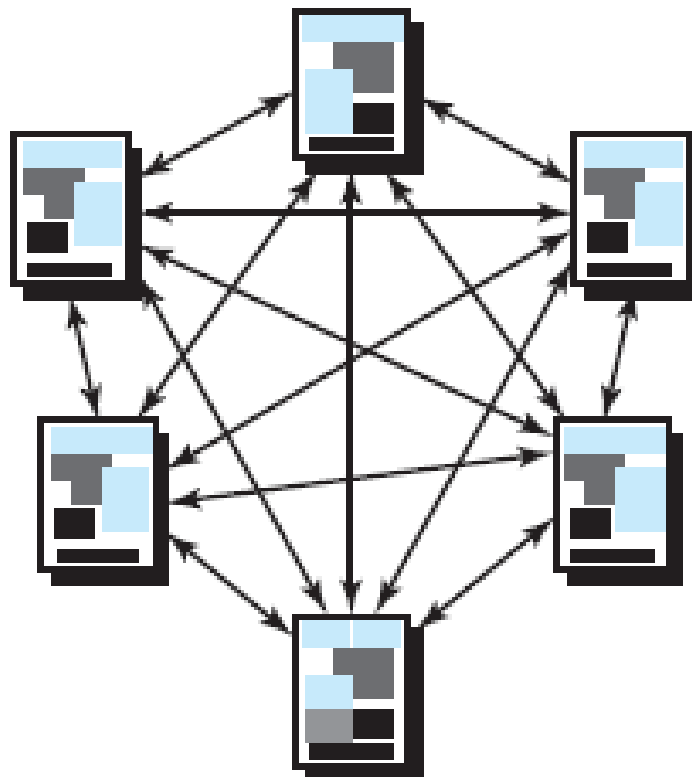
Arsitektur Isi (lanjutan)

3. **Struktur Hirarki:** rancangan dimungkinkan adanya pencabangan *hypertext* (aliran kendali) secara horizontal bergerak melintasi cabang-cabang vertikal pada struktur aplikasi web.



Arsitektur Isi (lanjutan)

4. **Struktur Jaringan (Web Murni):** struktur ini dapat digabungkan untuk membentuk struktur-struktur yang bersifat komposit.



B. Arsitektur Aplikasi Web

- Mendeskripsikan infrastruktur yang memungkinkan sistem atau aplikasi berbasis web untuk mencapai sasaran-sasaran bisnisnya.
- Arsitektur aplikasi web dibuat dalam 3 lapisan yang bertujuan untuk memisahkan antarmuka dari mekanisme navigasi dan dari perilaku yang dimiliki oleh aplikasi. Hal ini akan menyederhanakan implementasi aplikasi web dan akan meningkatkan penggunaan ulang komponen-komponennya.

Arsitektur Aplikasi Web (Lanjutan)

Arsitektur MVC (*Model View-Controller*) secara fungsional dijelaskan sebagai berikut:

- a. **Pengendalian:** mengelola akses ke model, ke view dan melakukan koordinasi aliran data di antara model dan view.
- b. **Model:** memuat semua isi yang bersifat spesifik terhadap aplikasi dan memuat logika pemrosesan, termasuk di dalamnya semua objek isi.
- c. **View:** memuat semua fungsi yang bersifat spesifik terhadap *user interface* yang di dalamnya termuat presentasi isi dan logika pemrosesan, objek isi, akses ke data dan ke sumber informasi lainnya, dan semua fungsionalitas pemrosesan yang diperlukan *end user*.

8. PERANCANGAN NAVIGASI

- Merancang lintasan-lintasan navigasi yang memungkinkan *user* mengakses isi dan fungsi-fungsi aplikasi web.

A. Semantik-Semantik Navigasi

- Unit semantik navigasi adalah sejumlah informasi dan struktur navigasi yang saling berkolaborasi untuk memenuhi kebutuhan *user* yang bersifat khusus.
- Contoh seorang pelanggan baru mungkin memiliki 3 *use case* yang berbeda yang semuanya menghasilkan akses ke informasi-informasi dan fungsi-fungsi aplikasi web yang berbeda. Semantik navigasi harus dibuat untuk masing-masing sasaran tersebut.

B. Sintak Navigasi

- **Link navigasi yang bersifat individual:** *link* berbasis teks, *icon*, tombol, pilihan, dan grafis harus sesuai dan konsisten dengan isi yang ditampilkan
- **Bar navigasi horizontal:** daftar kategori isi dan kategori fungsional utama yang berisi *link* yang sesuai
- **Kolom-kolom navigasi vertikal:**
 - Berisi daftar kategori utama dan kategori fungsional
 - Berisi semua objek isi utama
- **Tab-tab:** merepresentasikan kategori isi dan kategori fungsional sebagai *tab sheet* saat suatu *link* diperlukan
- **Peta situs:** menyediakan suatu tabel isi yang dapat digunakan untuk melakukan navigasi ke semua objek dan semua fungsionalitas

Contoh icon navigasi yang kurang jelas

Absensi

Kerangka

Absensi

Absensi

Nama

NIM

0

Semester

0

Tanggal

Hadir

◀

✓

↶

▶

CONTOH

- Dosen diminta untuk memberikan beberapa contoh aplikasi web dari internet, dan menjelaskan sesuai dengan materi yang dibahas
- Contoh tampilan aplikasi web dapat berupa tampilan yang baik dan/atau yang kurang baik

TUGAS

- Mahasiswa diminta untuk membuat disain web yang sesuai dengan materi yang telah dibahas
- Dikumpulkan pada pertemuan berikutnya
- Catatan: tugas hanya **desain web** dalam bentuk tampilan form saja, **bukan membuat program** berbasis web