

# PROTOTIPE

- Bagaimana cara mengekspresikan ide-ide perancangan?
  - Tidak ada software pengkodean pada tahap ini.
- Catatan:
  - Membuat dengan cepat.
  - Perkenankan banyak fleksibilitas untuk desain yang berbeda.
  - Membuat dengan murah.
  - Meningkatkan nilai umpan balik.

## **Permasalahan yang timbul:**

1. Anda tidak dapat mengevaluasi rancangan tersebut sampai rancangan tersebut dibangun.
2. Namun, setelah pembuatan, jika ingin melakukan perubahan maka akan sangat sulit.

## **Solusinya:**

Simulasikan perancangan dengan biaya yang kecil, salah satunya dengan membuat model (prototype).

Untuk itu kita dapat menerapkan UCD (*User Centered Design*) dalam pembuatan prototype tersebut.

UCD adalah mengenai perancangan teknologi yang interaktif untuk memenuhi kebutuhan user.

## **Tahapan dalam UCD antara lain:**

1. Memahami kebutuhan user.
2. Mendeskripsikan kebutuhan user.
3. Merancang prototype sebagai alternatif.
4. Mengevaluasi perancangan.

## **Karakteristik dalam proses UCD:**

1. Memahami user dan kebutuhannya.
2. Fokus pada user pada tahap awal desain dan mengevaluasi hasil desain.
3. Mengidentifikasi, membuat dokumentasi dan menyetujui kegunaan dan tujuan pengalaman user.
4. Perulangan hampir dapat dipastikan. Para perancang tidak pernah berhasil hanya dalam satu kali proses.

Dalam bidang yang lain perancangan sebuah prototype biasanya berupa model dalam skala kecil.

Contoh: Maket Bangunan

## **Mengapa menggunakan Prototype?**

- Evaluasi dan feedback pada rancangan interaktif.
- Stakeholder (dalam hal ini user) dapat melihat, menyentuh, berinteraksi dengan prototype.
- Anggota tim dapat berkomunikasi secara efektif.
- Para perancang dapat mengeluarkan ide-idenya.
- Memunculkan ide-ide secara visual dan mengembangkannya.
- Dapat menjawab pertanyaan → membantu pemilihan di antara alternatif-alternatif.

## **Dimensi Prototype**

### 1. Penyajian

- Bagaimana desain dilukiskan atau diwakili?
- Dapat berupa uraian tekstual atau dapat visual dan diagram.

### 2. Lingkup

Apakah hanya interface atau apakah mencakup komponen komputasi?

### 3. Executability (Dapat dijalankan)

- Dapatkah prototype tersebut dijalankan?
- Jika dikodekan, akan ada periode saat prototype tidak dapat dijalankan.

### 4. Maturation (Pematangan)

Apakah tahapan-tahapan produk ini mengikuti?

- Revolusioner: mengganti yang lama.
- Evolusioner : terus melakukan perubahan pada perancangan yang sebelumnya.

## **Metode Pembuatan Prototyping Dengan Cepat**

### **Non-Computer**

(biasanya dikerjakan lebih awal dalam proses pembuatan)

**vs**

### **Computer-Based**

(biasanya dikerjakan kemudian)

## **Metode Non-Computer (Manual)**

### **Tujuan**

Ingin menyatakan gagasan desain dan mendapatkan dengan mudah dan cepat pendapat atas sistem.

### **Deskripsi Desain**

Dapat berupa deskripsi tekstual dari suatu desain sistem.

- Kelemahan yang nyata adalah seberapa jauh dari sistem yang sebenarnya.
- Tidak dapat melakukan suatu pekerjaan yang mewakili aspek dari interface.

### **Sketsa, Mock-ups**

- Paper-Based “menggambarkan” interface.
- Baik untuk mengungkapkan pendapat.
- Difokuskan pada orang dengan desain tingkat tinggi.
- Tidak terlalu baik untuk menggambarkan alur dan rinciannya.
- Murah dan cepat → umpan balik sangat menolong.

### **Storyboarding**

Pensil dan simulasi catatan atau walkthrough dari kemampuan dan tampilan sistem.

- Menggunakan urutan diagram/gambar.
- Menunjukkan kunci snap shots.
- Cepat dan mudah.

## **Skenario**

Hipotesis atau imajinasi penggunaan.

- Biasanya menyertakan beberapa orang, peristiwa, lingkungan dan situasi.
- Menyediakan konteks operasi.
- Terkadang dalam format naratif, tetapi juga dapat berupa sketsa atau bahkan video.

## **Utilitas Skenario**

- Melibatkan dan menarik.
- Mengizinkan perancang untuk melihat masalah dari pandangan orang lain.
- Memudahkan umpan balik dan pendapat.
- Dapat sangat kreatif dan futuristik.

## **Teknik Lain**

Tutorial dan Manual

- Mungkin menuliskannya lebih berguna daripada disimpan dalam kepala.
- Memaksa perancang untuk membuat keputusan dengan tegas.
- Menulis/meletakkannya di atas kertas jauh lebih berharga.

## **Metode Komputer**

Menirukan lebih banyak kemampuan sistem.

- Pada umumnya hanya baru beberapa aspek atau fitur
- Dapat berpusat pada lebih banyak detail
- Bahaya: Para pemakai jadi lebih segan untuk menyarankan perubahan sekali ketika mereka melihat prototype yang lebih realistis.

## **Terminologi**

### 1. Prototype Horizontal

Sangat luas, mengerjakan atau menunjukkan sebagian besar interface, tetapi ini dilakukan dengan cara yang licik.

### 2. Prototype Vertikal

Lebih sedikit aspek atau fitur dari interface yang disimulasikan, tetapi dilaksanakan dengan rincian yang sangat baik.

### 3. Early Prototyping

#### 4. Late Prototyping

#### 5. Low-fidelity Prototyping (prototype dengan tingkat ketepatan yang rendah)

##### **Contoh (1) storyboard:**

- Digunakan di awal desain.
- Biasanya digunakan dengan skenario, lebih terinci, dan dapat diputar ulang.
- Kumpulan dari sketsa/frame individual.
  - menyajikan urutan inti cerita.
  - menunjukkan bagaimana kemungkinan user dapat mengalami peningkatan melalui setiap aktifitas.

##### **Contoh (2) sketsa:**

- Sketsa sangat penting untuk low-fidelity prototyping.
- Jangan takut dengan kemampuan menggambar.
- Menyajikan “tampilan” yang kotor dan cepat dari interface, konsep desain, dll.

##### **Contoh (3) “wizard-of-oz”:**

- User berpikir mereka berinteraksi dengan komputer, tapi developer lebih menanggapi hasilnya daripada sistemnya.
- Biasanya dilakukan di awal desain untuk memahami apa yang diharapkan oleh user.

#### 6. Mid-fidelity prototyping (prototype dengan tingkat ketepatan sedang)

- Form skematik.
- Navigasi dan fungsi yang disimulasikan → biasanya berbasis pada apa yang tampil pada layar atau simulasi layar.
- Contoh tools yang digunakan: powerpoint, illustrator, dll.

#### 7. High-fidelity prototyping (prototype dengan tingkat ketepatan yang tinggi)

- Hi-fi prototype seperti sistem akhir.
- Menggunakan bahan baku yang sama seperti produk akhir.

Tools umum yang digunakan: Macromedia Director, Visual Basic, Flash, Illustrator.

## Prototyping Tools

1. Draw/Paint Program, contoh: Photoshop, Coreldraw
    - Menggambar setiap layar, baik untuk dilihat.
    - Prototype horisontal, tipis.
    - Adobe Photoshop.
  
  2. Scripted Simulations/Slide Show, contoh: Powerpoint, Hypercard, Macromedia Director, HTML.
    - Letakkan tampilan seperti storyboard dengan (animasi) perubahan diantaranya.
    - Dapat memberikan user catatan yang sangat spesifik.
    - Disebut *chauffeured prototyping*.
    - Macromedia Director.
  
  3. Interface Builders, contoh: Visual Basic, Delphi, UIMX.
    - Tools untuk menampilkan jendela, kendali, dan lain-lain dari interface.
- Fitur yang baik
    - Mudah dikembangkan dan memodifikasi layar.
    - Mendukung jenis interface yang dikembangkan.
    - Mendukung berbagai macam divisi Input/Output.
    - Mudah untuk memodifikasi dan menghubungkan layar.
    - Mengijinkan memanggil prosedur eksternal dan program.
    - Mengijinkan mengimpor teks, grafik, media lain.
    - Mudah untuk dipelajari dan digunakan.
    - Dukungan yang baik dari vendor.