

Bab 5 : Desain Antarmuka (Interface)

5.1 Pendahuluan

Antarmuka pengguna (user interface) adalah tampilan sistem yang digunakan untuk berinteraksi dengan pengguna. Desain antarmuka bertujuan memberikan kemudahan, kejelasan, dan kenyamanan dalam penggunaan sistem. Keindahan tampilan mendukung kenyamanan dan frekuensi penggunaan sistem. Komponen penting antarmuka menurut Schlatter (2013) meliputi konsistensi, hierarki, kepribadian, tata letak, tipografi, warna, gambar/icon, dan kontrol interaktif.

5.2 Tujuan dan Manfaat Desain Antarmuka

Tujuan utama desain antarmuka adalah:

- Memenuhi spesifikasi fungsional dan batasan media
- Memudahkan proses pengembangan sistem
- Memberikan gambaran jelas bagi pengembang
- Meningkatkan efisiensi, produktivitas, dan komunikasi antara pengguna dan sistem

Manfaat: Menjadi penghubung antara pengguna dan sistem baik secara perangkat lunak (GUI/CLI) maupun perangkat keras (USB, ADB).

5.3 Syarat dan Prinsip Desain Antarmuka

Prinsip desain antarmuka yang baik meliputi:

- **Yourdon (1979):** struktur, kesederhanaan, visibilitas, umpan balik, toleransi, penggunaan ulang, perhatian pengguna, bahasa sederhana.
- **Deborah J. Mayhew (1999):** kompatibilitas pengguna, produk, tugas, alur kerja, konsistensi, keakraban, manipulasi langsung, kontrol, fleksibilitas, tanggapan cepat, perlindungan, kemudahan belajar & penggunaan.
- Aktivitas berpusat pada pengguna: memahami konteks, menentukan kebutuhan, membuat solusi, dan mengevaluasi desain.

5.4 Langkah-langkah Desain Antarmuka

1. **Buat tampilan sederhana dan konsisten**, gunakan elemen standar UI.
2. **Tentukan tujuan tata letak** dengan jarak dan struktur yang rapi.
3. **Gunakan warna dan tekstur strategis** untuk menarik atau mengalihkan perhatian.
4. **Tipografi** sebagai penunjuk hierarki dan kejelasan.
5. **Komunikasikan status sistem** secara jelas (misal error, sukses, loading).
6. **Gunakan nilai default** untuk memudahkan pengisian form.
7. **Mudah diperbaiki**: izinkan pengguna mengedit data dengan mudah.

Penentuan Kebutuhan Desain Antarmuka

1. Memahami Konteks Penggunaan

- **Siapa penggunanya?** (usia, latar belakang, kemampuan teknis)
- **Di mana mereka menggunakan sistem?** (desktop, mobile, luar ruangan, dll.)
- **Untuk tujuan apa sistem digunakan?** (pekerjaan, hiburan, transaksi, dll.)

2. Menentukan Tujuan Pengguna

- Apa yang ingin dicapai pengguna melalui sistem?
- Proses atau alur kerja seperti apa yang harus difasilitasi?
- Fitur apa yang paling sering digunakan dan mana yang jarang?

3. Mengidentifikasi Kebutuhan Fungsional dan Non-Fungsional

- **Fungsional**: menu, form input, notifikasi, navigasi, dan fitur interaksi lainnya.
- **Non-Fungsional**: kemudahan penggunaan (usability), estetika, performa, responsivitas, dan keamanan.

4. Analisis Masalah dari Sistem Sebelumnya (Jika Ada)

- Apakah sistem sebelumnya membingungkan?
- Apa keluhan dari pengguna sebelumnya?
- Apakah desain lama terlalu kompleks, tidak konsisten, atau tidak ramah pengguna?

5. Melibatkan Pengguna dalam Proses Perancangan (User-Centered Design)

- Melakukan survei, wawancara, atau observasi terhadap pengguna potensial.
- Melibatkan pengguna dalam pengujian prototipe awal untuk mendapatkan masukan langsung.

6. Menentukan Platform dan Teknologi

- Apakah sistem berbasis web, desktop, atau mobile?
- Apa batasan teknis perangkat yang digunakan (resolusi layar, kecepatan internet, kompatibilitas)?

7. Menyusun Daftar Kebutuhan Desain Antarmuka

Contoh daftar kebutuhan:

- Tampilan yang **sederhana dan tidak membingungkan**
- Navigasi yang **jelas dan konsisten**
- Gunakan **ikon dan label yang mudah dipahami**
- Berikan **umpan balik langsung** saat pengguna melakukan aksi
- Warna dan font yang **serasi** serta mendukung kenyamanan membaca

Kesimpulan

Menentukan kebutuhan desain antarmuka berarti menggali sedalam mungkin bagaimana pengguna akan berinteraksi dengan sistem. Ini mencakup pemahaman konteks, tujuan

pengguna, kebutuhan teknis, dan prinsip desain antarmuka yang baik seperti **kesederhanaan, konsistensi, responsivitas, dan ramah pengguna**.

Proyek

Aplikasi To-Do List (Web)

1. Profil Pengguna

- **Target pengguna:** Mahasiswa dan pekerja produktif (usia 18–35 tahun)
- **Kemampuan teknis:** Umum, pengguna familiar dengan aplikasi web standar
- **Perangkat:** Laptop/PC, smartphone

2. Tujuan Pengguna

- Menambahkan tugas harian dengan cepat
- Menandai tugas yang sudah selesai
- Menghapus atau mengedit tugas
- Melihat semua tugas berdasarkan kategori atau tanggal

3. Kebutuhan Fungsional

- Form untuk menambahkan tugas baru (judul + deskripsi + tenggat waktu)
- Tampilan daftar tugas yang rapi dan mudah dibaca
- Fitur centang untuk menandai tugas selesai
- Fitur filter tugas (semua, selesai, belum selesai)
- Fitur edit dan hapus tugas
- Sinkronisasi otomatis (opsional)

4. Kebutuhan Non-Fungsional

- Antarmuka harus ringan dan cepat dimuat
- Desain harus simpel dan bersih (minimalis)
- Responsif di semua perangkat (mobile & desktop)
- Mode gelap (opsional)

5. Kebutuhan Visual / UI

- Navigasi horizontal sederhana (Logo, Filter, Pengaturan)
- Tombol Add Task yang terlihat jelas
- Warna tenang (misalnya: biru muda, putih, abu)
- Font sans-serif yang mudah dibaca (misal: Roboto)
- Tugas yang selesai dicoret atau transparan

6. Wireframe Sederhana

(kalau kamu mau, aku bisa bantu buat sketsa visual/wireframe-nya juga)

