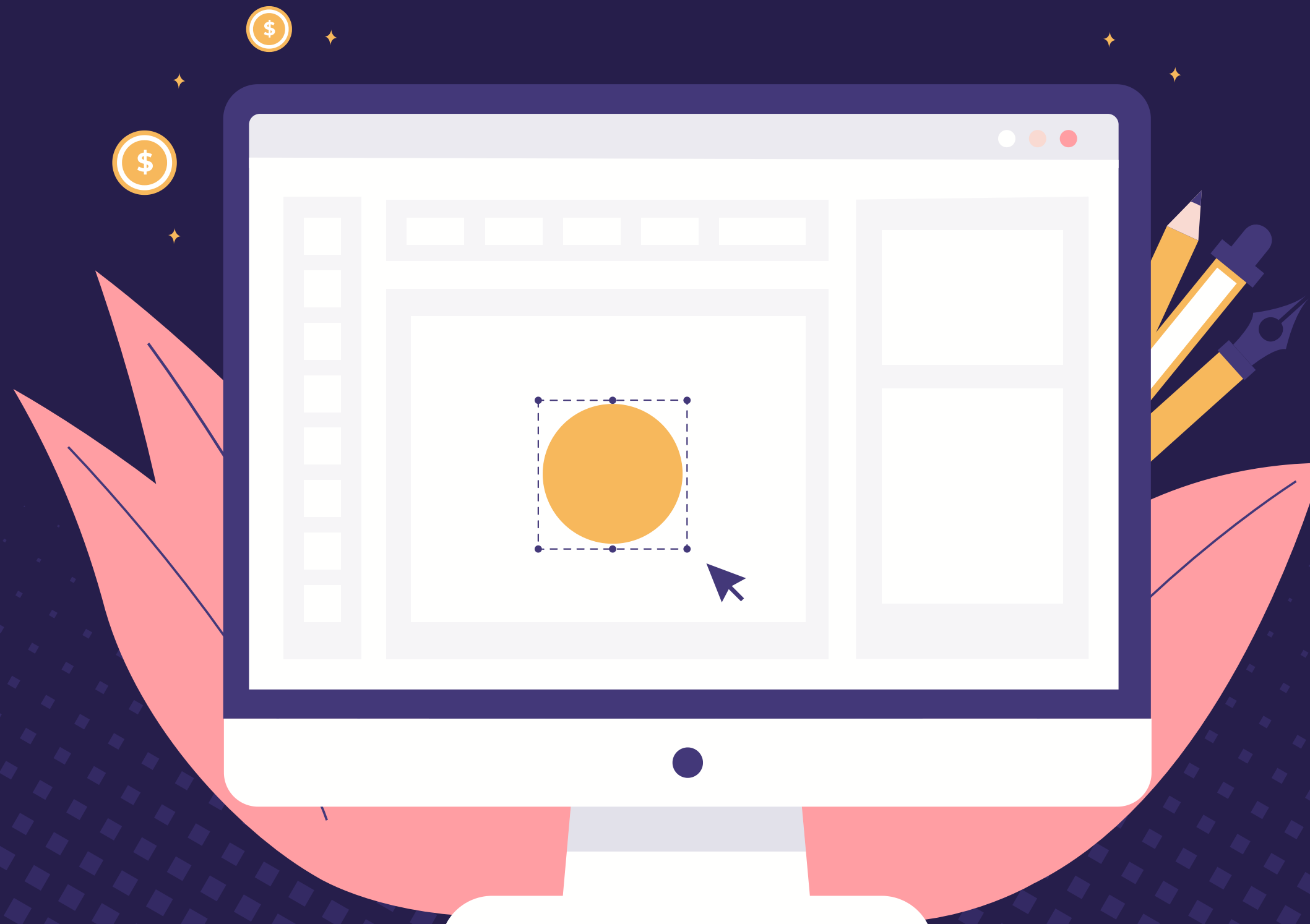


# UI/UX TESTING

KELOMPOK 2



# ANGGOTA KELOMPOK

## KELOMPOK 2



**HERY MANGALIK**  
**H071231013**



**FARIZ IDHAM RAMDHANI**  
**H071231066**



**KEVIN RYADI RANTE**  
**PAMUMBU**  
**H071231054**



**MUH. FAJRIN SUHAR**  
**H071231029**



**ANDI RISWANDA**  
**H071231008**



**NUR FADILLAH**  
**H071231080**



**DESTIN KENDENAN**  
**H071231058**

**01**

# PERBEDAAN MENDASAR ANTARA UI & UX TESTING

- ✓ UI Testing : Fokus pada tampilan antarmuka aplikasi : warna, ikon, ukuran tombol, layout dapat ditampilkan dengan benar, konsisten, dan responsif  
→ **Contoh pengujian:** memastikan tombol Login muncul di posisi yang benar, ukuran font terbaca dengan baik, atau tampilan tetap konsisten di desktop maupun mobile.
- ✓ UX Testing : Fokus pada pengalaman pengguna secara keseluruhan saat menggunakan aplikasi.  
→ **Contoh pengujian:** Menguji apakah pengguna bisa menemukan produk dan menyelesaikan pembelian dengan mudah tanpa merasa bingung.



# ***Fokus UI Testing***

## **Konsistensi Visual**

Semua halaman harus memiliki gaya seragam: warna, ikon, font, ukuran tombol.

Contoh testing:

- Checklist manual
- Automated visual regression testing → Percy, AppliTools, atau Selenium + visual plugin.

Contoh, Cek tombol “Login” di halaman A dan halaman B → warna, ukuran, dan posisi harus sama.

# ***Fokus UI Testing***

## **Responsivitas**

Desain tetap nyaman dipakai di berbagai ukuran layar (desktop, tablet, smartphone).

Contoh testing:

- Manual testing
- Automated testing → BrowserStack, LambdaTest, atau Responsively App.

Contoh, Buka website di HP 5", tablet 10", laptop 14" → pastikan teks & gambar tetap terbaca.

# ***Fokus UI Testing***

## **Kompatibilitas**

UI harus bekerja dengan baik di berbagai browser (Chrome, Firefox, Safari, Edge) dan sistem operasi (Windows, macOS, Android, iOS).


Contoh testing:


- Cross-browser testing → BrowserStack, Sauce Labs, LambdaTest.

Contoh, Periksa apakah animasi atau ikon tetap tampil di iOS dan Android.

# Fokus UX Testing


## ALUR KERJA

Alur kerja UX testing adalah proses iteratif yang terintegrasi dalam siklus pengembangan perangkat lunak, khususnya dalam pendekatan agile atau waterfall. Alur kerja (workflow) UX testing biasanya dimulai dari perencanaan (menentukan tujuan dan peserta), rekrutmen pengguna, pelaksanaan sesi uji (observasi dan wawancara), analisis data, hingga iterasi desain. 


Contoh Testing: Shopify (Pengujian Profil Pakar): Alur kerja dimulai dengan wawancara pengguna generatif untuk perencanaan, dilanjutkan dengan card sorting dan tree testing untuk rekrutmen dan eksekusi. Dan juga temuan seperti preferensi pengguna terhadap profil yang lebih manusiawi digunakan untuk iterasi desain, mengoptimalkan arsitektur informasi halaman profil. 

# Fokus UX Testing

## KEGUNAAN

Kegunaan atau usability mengacu pada seberapa mudah dan efektif pengguna dapat berinteraksi dengan perangkat lunak. Usability testing adalah metode untuk mengevaluasi ini dengan melibatkan pengguna nyata dalam tugas-tugas spesifik, sambil mengamati kesulitan mereka. Manfaatnya termasuk mengidentifikasi cacat desain awal, mengurangi biaya perbaikan, dan meningkatkan kepuasan pengguna. 

Contoh testing:

Movista (Pengiriman Pesan): Alur kerja menggunakan usability testing remote dengan prototipe high-fidelity via Maze untuk eksekusi unmoderated, tugas dan pertanyaan follow-up untuk analisis. Feedback menggeser langkah pemilihan penerima lebih awal, diikuti iterasi sebelum peluncuran dengan sampel besar untuk validasi. 



# Fokus UX Testing

## AKSEBILITAS (ACCESSIBILITY TESTING) ✓

Aksesibilitas adalah aspek UX yang memastikan perangkat lunak dapat digunakan oleh semua orang, termasuk penyandang disabilitas seperti gangguan penglihatan, pendengaran, atau motorik. Pengujian aksesibilitas melibatkan evaluasi terhadap standar seperti WCAG (Web Content Accessibility Guidelines), termasuk penggunaan screen reader, navigasi keyboard, dan kontras warna. ✓

Contoh testing:

Pengujian Kontras Warna: Memeriksa kontras teks dengan latar belakang untuk memenuhi standar WCAG, seperti rasio minimum untuk teks normal dan besar. Ini memastikan pengguna dengan penglihatan rendah dapat membaca konten dengan jelas. ✓

# Metode & Tools UI/UX Testing

## Heatmaps

- Dilakukan secara langsung oleh QA tester atau user.
- Fokus: mengecek tampilan (UI) dan alur interaksi (UX) secara step-by-step.
- Contoh tools: Checklist, Prototype, Figma, Zeplin.



## A/B Testing

- Visualisasi area yang paling sering di-klik atau diperhatikan user.
- Fokus: mengetahui perilaku pengguna terhadap UI (misalnya tombol sering dilewati atau tidak terlihat).
- Contoh tools: Hotjar, Crazy Egg, Microsoft Clarity.



# ***Heuristic Evaluation***

## **VISIBILITAS STATUS SISTEM**

Sistem harus selalu memberi tahu pengguna apa yang sedang terjadi melalui umpan balik yang jelas dalam waktu yang wajar

## **KECOCOKAN ANTARA SISTEM DAN DUNIA NYATA**

Sistem harus menggunakan bahasa, ikon, dan konsep yang familiar bagi pengguna, bukan yang hanya dipahami oleh sistem

## **KONTROL DAN KEBEBASAN PENGGUNA**

Pengguna harus bisa dengan mudah membatalkan aksi yang tidak sengaja mereka lakukan. Sediakan "pintu darurat" yang jelas

## **KONSISTENSI DAN STANDAR**

Elemen desain, terminologi, dan alur harus konsisten di seluruh platform. Pengguna tidak perlu bingung apakah kata atau aksi yang berbeda memiliki arti yang sama

## **PENCEGAHAN KESALAHAN**

Desain yang baik harus bisa mencegah pengguna melakukan kesalahan sejak awal, bukan hanya menampilkan pesan error yang baik

# ***Heuristic Evaluation***

## **MENGENALI DARIPADA MENGINGAT**

Minimalkan beban ingatan pengguna dengan membuat objek, aksi, dan pilihan terlihat jelas.

## **FLEKSIBILITAS DAN EFISIENSI PENGUNAAN**

Sistem harus bisa melayani pengguna pemula maupun ahli. Sediakan jalan pintas (shortcuts) bagi pengguna ahli untuk mempercepat interaksi.

## **DESAIN ESTETIS DAN MINIMALIS**

Antarmuka tidak boleh berisi informasi yang tidak relevan atau jarang dibutuhkan

## **BANTU PENGGUNA MENGENALI, MENDIAGNOSIS, DAN MEMULIHKAN DIRI DARI KESALAHAN**

Pesan error harus ditulis dalam bahasa yang mudah dimengerti (bukan kode error), menunjukkan masalahnya dengan tepat, dan menyarankan solusi

## **BANTUAN DAN DOKUMENTASI**

Bantuan harus mudah dicari dan fokus pada tugas pengguna

**TERIMA KASIH**