

Diseño aplicado a interfaces web con Flutter

Unidad 01 - Fundamentos de diseño UI/UX



Autor: Sergi García



Actualizado Noviembre 2025

Licencia



Reconocimiento - No comercial - CompartirlGual (BY-NC-SA): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se ha de hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán diferentes símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

Importante

Atención

Interesante

ÍNDICE

1. Introducción: Diseño UI/UX - Dos caras de una misma moneda	3
2. Usabilidad y experiencia de usuario	4
3. Tipos de interfaces gráficas	6
4. UI vs UX: La colaboración perfecta entre ciencia y arte	8
5. Principios de diseño de interfaces gráficas	9
6. Arquitectura emocional: diseño centrado en el usuario	11
7. Modelos mentales y diseño intuitivo: hablar el idioma del usuario	12
8. Jerarquía visual e interacción: el arte de guiar la mirada	13
9. Diseño centrado en tareas y objetivos del usuario	15
10. Diseño “Mobile first” y su aplicación en diseño web	16
11. Recursos recomendados para aprender UI/UX	17

UNIDAD 01 - FUNDAMENTOS DE DISEÑO UI/UX

1. INTRODUCCIÓN: DISEÑO UI/UX - DOS CARAS DE UNA MISMA MONEDA

En el mundo digital, el diseño de interfaces (UI) y la experiencia de usuario (UX) son como el cerebro y el corazón de un producto: uno gestiona la funcionalidad y el otro da vida a la interacción. Aunque suelen mencionarse juntos, representan disciplinas distintas pero profundamente conectadas.

¿UI o UX? Diferencias clave

- **UI (User Interface):** Se enfoca en lo tangible: colores, tipografías, botones, iconos y layouts. Es el "rostro" del producto, lo que el usuario ve y toca.
Ejemplo: Un botón rojo brillante con una animación al hacer clic es decisión de UI.
- **UX (User Experience):** Abarca lo intangible: emociones, eficiencia, accesibilidad y satisfacción durante el uso. Es el "viaje" completo del usuario.
Ejemplo: Que ese botón rojo esté ubicado donde el usuario lo espera (y no lo lleve a un error) es trabajo de UX.

¿Por qué son complementarios?

Un producto con UI hermoso, pero UX pobre es como un auto deportivo sin motor: atractivo, pero inútil. Por el contrario, una UX sólida con UI descuidado puede frustrar a los usuarios, aunque funcione. El equilibrio es clave.

El usuario en el centro

El diseño UI/UX exitoso sigue una regla de oro:

"No diseñes para ti, diseña para quienes usarán tu producto".

Esto implica:

- Investigar necesidades reales (entrevistas, testeos).
- Prototipar e iterar basado en feedback.
- Priorizar usabilidad sobre tendencias estéticas pasajeras.

Ejemplo práctico:

Imagina una app de banca móvil:

- UI define cómo se ve el saldo (tipografía legible, color que denote seguridad).
- UX asegura que el usuario pueda transferir dinero en menos de 3 pasos, sin errores.

Impacto en el negocio

Un buen diseño UI/UX no solo mejora la satisfacción del usuario, también:

- ✓ Reduce costos (menos errores = menos soporte técnico).
- ✓ Aumenta la retención (usuarios felices vuelven).
- ✓ Diferencia tu producto de la competencia.

Por si no se ha entendido correctamente, vamos a plantear otro ejemplo:

Imagina entrar a un restaurante:

- **El UI (User Interface)** sería la presentación del plato, la decoración del lugar, el menú visual y los colores de la ambientación. Es todo lo que ves y tocas.
- **La UX (User Experience)** sería la experiencia completa: el tiempo de espera, la amabilidad del camarero, el sabor de la comida e incluso cómo te sientes al pagar.

¿Por qué está comparación?

- Un restaurante con platos hermosos, pero comida mala (UI bueno, UX malo) te decepcionará, aunque las fotos para Instagram sean perfectas.

- Un local con comida exquisita, pero platos descuidados y mala iluminación (UX buena, UI malo) puede ahuyentar a clientes antes de que prueben el primer bocado.

El Diseño UI/UX en Acción

Piensa en un restaurante "Fast food" versus un restaurante gourmet:

- **Restaurante "Fast food":** UI simple (colores vibrantes, íconos claros) + UX eficiente (comida rápida, procesos estandarizados).
- **Restaurante gourmet:** UI sofisticado (vajilla elegante, iluminación tenue) + UX memorable (servicio personalizado, ritmo pausado).

Ambos son exitosos, pero cada uno optimiza su UI y UX para su público objetivo.

El usuario es el comensal

El diseño centrado en el usuario funciona igual que un buen restaurante:

1. **Investiga** (¿Qué prefieren tus comensales? ¿Rapidez o ambiente?).
2. **Prototipa** (Prueba platos nuevos antes de lanzarlos).
3. **Itera** (Ajusta el menú según feedback).

Ejemplo digital:

- UI: Un botón de "Pedir ahora" rojo y brillante (como el letrero de "Abierto" de un restaurante).
- UX: Que el proceso de pedido sea tan fácil como hacer un gesto a un camarero (menos de 3 clics, sin errores).

¿Por qué es crucial este equilibrio?

- Negocios digitales: El 88% de los usuarios no vuelve a un sitio tras mala experiencia (UX).
- Branding: La identidad visual (UI) es lo primero que notan, pero la usabilidad (UX) es lo que los hace quedarse.

 **Importante:** Antes de diseñar, pregúntate: ¿Estoy construyendo un "fast food" digital (eficiente) o un "restaurante de experiencia" (memorable)?

2. USABILIDAD Y EXPERIENCIA DE USUARIO

2.1 Definición y principios de usabilidad

La **usabilidad** se refiere a la facilidad con la que un usuario puede aprender a usar un sistema, completar sus tareas y hacerlo de forma satisfactoria y eficiente. Es un aspecto fundamental del diseño de interfaces que garantiza que el producto sea accesible y efectivo para sus usuarios.

- **Principios básicos de usabilidad:**
 - **Facilidad de aprendizaje:** Los usuarios nuevos deben poder entender rápidamente cómo usar la interfaz.
 - **Eficiencia de uso:** Los usuarios frecuentes deben poder realizar sus tareas de forma rápida y sin obstáculos.
 - **Memorabilidad:** Los usuarios que vuelven después de un tiempo sin usar la interfaz deben poder retomarla sin problemas.
 - **Tasa de errores baja:** La interfaz debe minimizar los errores del usuario y permitir su recuperación fácil.
 - **Satisfacción:** La experiencia debe ser agradable y cumplir con las expectativas.

2.2 Diferencias entre usabilidad y experiencia de usuario (UX)

Aunque están relacionadas, **usabilidad** y **experiencia de usuario (UX)** no son lo mismo:

- **Usabilidad** se centra en cómo de fácil y eficiente es usar un producto.
- **Experiencia de usuario (UX)** es más amplia e incluye todas las percepciones, emociones y reacciones del usuario antes, durante y después de usar el producto. Incluye usabilidad, pero también diseño emocional, accesibilidad, estética, rendimiento y contexto de uso.
- La usabilidad es una parte importante, pero no la única que conforma una buena experiencia de usuario.

2.3 Métricas de evaluación

Para medir la usabilidad de una interfaz se usan varias métricas. Algunas de las métricas más utilizadas para evaluar una interfaz son:

- **Eficacia**
 - Mide el **porcentaje de éxito** con que los usuarios completan tareas propuestas.
 - Ejemplo: si 8 de 10 usuarios logran realizar una acción, la eficacia es del 80%.
- **Eficiencia**
 - Mide el **tiempo y recursos** (clics, pasos) que los usuarios necesitan para completar tareas. Cuanto menos tiempo y menos pasos, mayor eficiencia.
- **Satisfacción del usuario**
 - Se evalúa con escalas y cuestionarios que miden la **percepción y opinión** del usuario sobre la interfaz.
 - Ejemplos de herramientas: cuestionarios SUS (System Usability Scale), encuestas con preguntas tipo Likert (de “muy de acuerdo” a “muy en desacuerdo”).

2.4 Heurísticas de usabilidad de Jakob Nielsen

Jakob Nielsen definió 10 reglas heurísticas para evaluar y diseñar interfaces usables:

1. **Visibilidad del estado del sistema:** el usuario debe saber en todo momento qué está pasando.
2. **Correspondencia entre el sistema y el mundo real:** usar lenguaje y conceptos familiares para el usuario.
3. **Control y libertad del usuario:** permitir deshacer y rehacer acciones fácilmente.
4. **Consistencia y estándares:** la interfaz debe ser coherente y seguir convenciones.
5. **Prevención de errores:** diseñar para evitar que el usuario cometa errores.
6. **Reconocer antes que recordar:** facilitar el reconocimiento de opciones y elementos, no depender de la memoria.
7. **Flexibilidad y eficiencia:** permitir atajos para usuarios avanzados.
8. **Estética y diseño minimalista:** evitar la sobrecarga de información.
9. **Ayudar a reconocer, diagnosticar y recuperarse de errores:** mensajes claros y comprensibles.
10. **Ayuda y documentación:** Ofrecer ayuda accesible aunque el sistema sea fácil de usar.

2.5 Accesibilidad: principios básicos

La accesibilidad es un componente esencial para que cualquier usuario, incluyendo personas con discapacidades, pueda usar la interfaz. Existen multitud de elementos a tener en cuenta para hacer una aplicación más accesible. A continuación, comentamos algunos de los principios más

importantes a tener en cuenta para un diseño accesible.

- **Principios básicos de accesibilidad:**

- **Contraste adecuado:** Colores que permitan distinguir texto y elementos visuales, importante para personas con dificultades visuales.
- **Navegación clara:** Que sea posible navegar usando teclado o lectores de pantalla.
- **Diseño inclusivo:** Evitar depender únicamente del color para transmitir información (ejemplo: rojo para error debe ir acompañado de un ícono o texto).
- **Tamaño de fuente legible:** Textos suficientemente grandes y con tipografías claras.
- **Compatibilidad con tecnologías asistivas:** Interfaces que funcionen con lectores de pantalla, ampliadores de pantalla, etc.

3. TIPOS DE INTERFACES GRÁFICAS

Las interfaces gráficas de usuario pueden clasificarse según el tipo de dispositivo o la tecnología con la que se interactúa. Cada tipo tiene características propias, retos específicos y usos comunes. Es importante conocer estas diferencias para planificar y diseñar interfaces adecuadas y efectivas.

3.1 Interfaces de escritorio (Windows, MacOS, Linux)

Estas interfaces son las más clásicas y tradicionales, diseñadas para ordenadores personales y estaciones de trabajo.

- **Características principales:**

- Uso con teclado y ratón.
- Ventanas múltiples, menús desplegables, barras de herramientas.
- Suelen permitir gran cantidad de información y funcionalidades en pantalla.
- Espacios amplios para elementos visuales.

- **Sistemas operativos comunes:**

- **Windows:** dominante en entornos empresariales y domésticos.
- **MacOS:** usado en equipos Apple, reconocido por su diseño cuidado.
- **Linux:** popular en entornos técnicos y desarrolladores, con interfaces variadas (GNOME, KDE, etc.).

3.2 Interfaces móviles (Android, iOS)

Diseñadas para dispositivos con pantalla táctil, como smartphones y tablets.

- **Características principales:**

- Interacción táctil: toques, deslizamientos, pellizcos.
- Espacios limitados, necesidad de simplificar la interfaz.
- Uso frecuente en movimiento y con una sola mano.
- Interfaces orientadas a gestos, accesos rápidos y notificaciones.

- **Sistemas operativos principales:**

- **Android:** Sistema abierto y altamente personalizable.
- **iOS:** Sistema cerrado, enfocado en experiencia uniforme y controlada.

3.3 Interfaces híbridas (app multiplataforma)

Son aplicaciones que funcionan en varios tipos de dispositivos con una sola base de código.

- **Características:**

- Se adaptan automáticamente a pantallas grandes y pequeñas.
- Permiten un desarrollo más rápido y económico.
- Usan tecnologías como React Native, Flutter, Ionic, Quasar, Kotlin Multiplatform.
- Desafío: lograr buen rendimiento y experiencia nativa en cada plataforma.

3.4 Realidad Aumentada (AR) y Realidad Virtual (VR)

Interfaces que integran o crean entornos digitales para aumentar o sustituir la realidad.

- **Realidad Aumentada (AR):**

- Superpone información digital sobre la imagen del mundo real.
- Ejemplos: juegos como Pokémon Go, aplicaciones de navegación o diseño.

- **Realidad Virtual (VR):**

- Sumergen al usuario en un entorno completamente digital.
- Usan gafas o cascos especiales.
- Aplicaciones en videojuegos, formación, simulación.

- **Características comunes:**

- Interfaces 3D, inmersivas.
- Requieren sensores de movimiento, cámaras o mandos.
- Desafíos: evitar mareos, diseñar interacciones intuitivas en 3D.

3.5 Interfaces gestuales (Kinect, sensores)

Permiten controlar dispositivos mediante movimientos del cuerpo o gestos con las manos.

- **Ejemplos:**

- **Kinect:** sensor de Microsoft para videoconsolas que detecta movimientos.
- Cámaras y sensores en dispositivos para reconocer gestos.
- Aplicaciones en juegos, presentaciones, control sin contacto.

- **Características:**

- Eliminan la necesidad de dispositivos físicos (ratón, teclado).
- Interacción natural, pero pueden ser menos precisas.
- Requieren diseño claro de gestos para evitar confusión.

3.6 Interfaces por voz (asistentes personales: Siri, Alexa...)

Interacción mediante comandos y respuestas habladas.

- **Ejemplos:**

- **Siri (Apple), Alexa (Amazon), Google Assistant**
- Se usan en smartphones, altavoces inteligentes, coches, etc.

- **Características:**

- Permiten controlar dispositivos sin usar manos.
- Ideales para multitarea o accesibilidad.

- Desafíos: reconocimiento de voz preciso, interpretación del lenguaje natural, respuesta contextual.

4. UI vs UX: LA COLABORACIÓN PERFECTA ENTRE CIENCIA Y ARTE

Aunque UI (Interfaz de Usuario) y UX (Experiencia de Usuario) suelen confundirse, son disciplinas complementarias pero distintas, como un arquitecto (UX) y un diseñador de interiores (UI). Uno asegura que la casa no se caerá; el otro, que sea hermosa y acogedora.

Diferencias Clave Entre UI y UX

Aspecto	UX (User Experience)	UI (User Interface)
Enfoque	¿Funciona bien? (Lógica, flujos, usabilidad).	¿Se ve bien? (Estética, interactividad visual).
Objetivo	Solucionar problemas del usuario.	Comunicar la marca de forma atractiva.
Proceso	Investigación, wireframes, testeos.	Diseño visual, animaciones, prototipos.
Herramientas	Miro, FigJam, Hotjar, UserTesting.	Figma, Adobe XD, Sketch, After Effects.
Métrica	Tasa de éxito en tareas, satisfacción.	Atractivo visual, consistencia de marca.

Ejemplo práctico:

- **UX en una app de banco:**
 - Asegura que transferir dinero toma 2 pasos (no 5 pasos).
 - Investiga por qué los usuarios abandonan el proceso.
- **UI en la misma app:**
 - Diseña un botón de "Transferir" visible y con feedback táctil.
 - Usa colores que inspiren confianza (azules profundos).

¿Cómo Colaboran?

Un producto excelente requiere que ambos roles trabajen en simbiosis:

1. UX Define el "Qué" y "Por Qué"

- **Actividades clave:**
 - Investigar necesidades reales (entrevistas, mapas de empatía).
 - Crear wireframes (esqueletos funcionales) y user flows.
 - Testear prototipos con usuarios.

Ejemplo:

El equipo UX de Spotify descubre que los usuarios quieren saltar intros de podcasts.

2. UI Diseña el "Cómo"

- **Actividades clave:**

- Transformar wireframes en interfaces visualmente coherentes.
- Aplicar guías de estilo (colores, tipografía, microinteracciones).
- Prototipar animaciones (ej: cómo se expande un menú).

Ejemplo:

El equipo UI de Spotify diseña un botón de "saltar 15 segundos" con un ícono intuitivo y un sonido satisfactorio.

Errores Comunes en la colaboración

- ✗ "Trabajar en silos": UI diseña sin entender los insights de UX (resultado: bonito pero inusable).
- ✗ Saltarse la investigación: UI empieza a diseñar pantallas antes de validar flujos con UX.
- ✗ Priorizar estética sobre funcionalidad: Ej: botones "minimalistas" tan sutiles que los usuarios no los ven.

Caso real:

- Snapchat rediseñó su UI en 2018 sin considerar hábitos de usuarios. Resultado: 1.2M de quejas y caída del 25% en ingresos.

Flujo Ideal de Trabajo Conjunto

1. **UX Investiga:** Identifica problemas y crea wireframes básicos.
2. **UI Propone:** Diseña opciones visuales basadas en esos wireframes.
3. **Juntos testean:** Validan con usuarios y ajustan.
4. **Iteran:** Mejoran continuamente (UI ajusta detalles, UX optimiza flujos).

Ejemplo de éxito:

- Airbnb:
 - UX investigó que los usuarios querían sentir "confianza" al reservar.
 - UI respondió con fotos grandes de anfitriones y reseñas prominentes.

5. PRINCIPIOS DE DISEÑO DE INTERFACES GRÁFICAS

Diseñar una interfaz efectiva va más allá de hacerla "bonita". Es como construir señales en una carretera: si son inconsistentes o confusas, los usuarios terminarán perdidos y frustrados. Debemos evitar que tus usuarios se sientan como en un laberinto sin salida. Estos principios son tu GPS para la usabilidad:

5.1 Consistencia (El "déjà vu" positivo)

¿Por qué? Los usuarios no deberían preguntarse: "¿Este botón hace lo mismo que el de ayer?"

- **Reglas de oro:**
 - Mismo color para acciones similares (ej: verde = guardar, rojo = eliminar).
 - Tipografía uniforme: Un solo tipo de fuente para títulos, otro para cuerpo.
 - Iconos familiares: Usar el mismo ícono de "lupa" para buscar en toda la app.
- **Ejemplo real:**
 - Google usa azul para los enlaces en todos sus productos (Gmail, Drive, etc.).
- **Error común:**
 - Cambiar el lugar del "Carrito" en un e-commerce (asesino de conversiones!).

5.2 Visibilidad del estado del sistema (¿Esto sigue vivo?)

¿Por qué? Nada genera más ansiedad que un botón que no responde y nos genera la duda de si ¿Esto sigue vivo? ¿Está funcionando? ¿Se ha "colgado"?

- **Tácticas:**
 - Feedback inmediato: animaciones al hacer clic (ej: botón que se "hunde").
 - Indicadores claros:
 - Carga: spinners o barras de progreso.
 - Éxito/Error: mensajes como "Tu pedido se guardó" o "Corrige el campo en rojo".
- **Ejemplo real:** LinkedIn muestra un círculo de progreso al subir un archivo.
- **Error común:** un formulario que no indica que se está enviando (usuarios lo llenan 5 veces).

5.3 Control y libertad del usuario (La "puerta de emergencia")

¿Por qué? Todos merecemos una segunda oportunidad (incluso en digital). Es bastante habitual equivocarse usando una aplicación, especialmente las primeras veces que se utiliza.

- **Claves:**
 - Deshacer/rehacer: como el Ctrl + Z en editores de texto.
 - Salidas obvias: botón "Cancelar" en diálogos o "Volver atrás" visible.
- **Ejemplo real:** Gmail permite recuperar emails borrados dentro de 30 segundos.
- **Error común:** un proceso de compra sin opción de editar el carrito antes de pagar.

5.4 Prevención de errores (Mejor un "no puedes" que un "ups")

¿Por qué? Corregir errores cuesta 10 veces más que prevenirlos.

- **Estrategias:**
 - Desactivar botones hasta que se completen campos obligatorios.
 - Validación en tiempo real: Mensajes como "La contraseña necesita 8 caracteres".
- **Ejemplo real:** Twitter muestra en rojo cuando un tweet supera el límite de caracteres.
- **Error común:** un formulario que solo muestra errores después de enviar.

5.5 Reconocimiento > Recuerdo (¿Dónde estaba eso? ¡Ah, aquí!)

¿Por qué? La memoria humana es frágil (y los usuarios son perezosos).

- **Cómo aplicarlo:**
 - Menús explícitos: opciones visibles (ej: pestañas en lugar de comandos ocultos).
 - Jerarquía visual: lo importante primero (como un periódico).
- **Ejemplo real:** Amazon muestra categorías en un menú desplegable gigante (nada está "escondido").
- **Error común:** app de banca que esconde la opción de transferencia en un menú hamburguesa.

6. ARQUITECTURA EMOCIONAL: DISEÑO CENTRADO EN EL USUARIO

No basta con que una interfaz funcione; debe hacer que los usuarios se sientan capaces, satisfechos y hasta felices. El diseño va más allá de la funcionalidad: diseñar experiencias que conecten emocionalmente) La arquitectura emocional es el arte de diseñar no solo para los ojos, sino para el corazón.

¿Por qué las emociones importan en el diseño?

- El 95% de las decisiones de compra son subconscientes (guiadas por emociones).
- **Ejemplo real:**
 - Apple no vende tecnología, vende experiencias de lujo y simplicidad (el "placer" de deslizar un iPhone).
 - Duolingo convierte aprender idiomas en un juego (con mascotas animadas y recompensas instantáneas).

Claves para diseñar emociones

1. Empatizar: Entender al usuario más allá de los datos

- **Herramientas:**
 - Mapas de empatía: ¿Qué siente, piensa, oye y ve tu usuario al usar tu producto?
 - User "Personas" con emociones: no solo "Juan, 30 años", sino "Juan siente ansiedad al pagar online y necesita seguridad".
- **Ejemplo práctico:**
 - Spotify usa playlists como "Canciones para días lluviosos" (empatiza con estados de ánimo).

2. Crear experiencias memorables (¡No solo usables!)

- Técnicas:
 - Microinteracciones: pequeñas animaciones que dan feedback (ej: un corazón que "late" al dar like).
 - Sorpresa y deleite: detalles inesperados, como el "gracias" animado de Slack al cerrar sesión.
- **Ejemplo real:**
 - Google celebra fechas especiales con Doodles interactivos (genera conexión emocional).

3. Eliminar fricciones (El enemigo de las emociones positivas)

- ¿Qué es una fricción?: Cualquier obstáculo que genere frustración (ej: formularios largos, pasos redundantes).
- **Cómo evitarlas:**
 - Simplifica flujos: ¿Realmente necesitas 5 pasos para crear una cuenta?
 - Lenguaje humano: en lugar de "Error 404", usa "Ups, esta página se fue de paseo".
- **Ejemplo práctico:** Amazon con *"1-Click Order"* (elimina el carrito de compras para usuarios frecuentes).

Ejemplo Profundo: El Botón que "Cobra Vida"

Imagina un botón de "Donar" en una ONG:

- **Sin emociones:** Botón gris, estático. Resultado: Usuario duda ("¿Funcionará?").
- **Con arquitectura emocional:**
 - Color cálido (naranja = urgencia + positividad).

- Animación sutil al pasar el mouse (efecto de "respirar").
- Feedback al clicar: Mensaje "¡Gracias por salvar vidas!" + sonido de campana.
- Resultado: El usuario siente impacto inmediato y satisfacción.

Técnicas Avanzadas para conectar emocionalmente

1. Psicología del color:

- Azul = Confianza (usado en bancos como Chase).
- Rojo = Urgencia (ej: Netflix para el botón "Reproducir").

2. Storytelling en UI:

- Airbnb no muestra propiedades, cuenta historias de viajes (fotos de anfitriones, textos personales).

3. Sonidos emocionales:

- El "pling" de WhatsApp al enviar un mensaje refuerza la instantaneidad.

7. MODELOS MENTALES Y DISEÑO INTUITIVO: HABLAR EL IDIOMA DEL USUARIO

Los modelos mentales son los atajos cognitivos que los usuarios han aprendido a lo largo de años de interacción con el mundo digital y físico. Un diseño verdaderamente intuitivo no se explica: un diseño se debe comprender al instante, como un gesto universal.

¿Qué es un Modelo Mental?

- Definición: es la representación interna que una persona tiene sobre cómo funciona un sistema, basada en experiencias previas.
- Ejemplo clásico:
 - Disquete = Guardar: aunque los disquetes ya no existen, el ícono persiste porque los usuarios lo asociaron por décadas con "guardar documentos".

Claves para Diseñar con Modelos Mentales

A. Metáforas visuales universales

¿Por qué funcionan?

El cerebro procesa imágenes 60,000 veces más rápido que el texto.

Ejemplos efectivos:

- Papelera → Eliminar (aunque los archivos vayan a la "papelera de reciclaje").
- Carpeta → Agrupar contenidos (como en un escritorio físico).
- Lupa → Buscar (incluso niños que nunca usaron una lupa real lo entienden).

Caso de estudio:

- Apple usa una calculadora skeumórfica (con texturas 3D que imitan objetos reales) porque los usuarios reconocen inmediatamente su función.

Error común:

Crear íconos abstractos sin referencia cultural (ej: un triángulo verde para "guardar").

B. Patrones de interacción aprendidos

Los usuarios esperan que ciertos elementos funcionen siempre igual:

- Menú Hamburguesa (≡) → aunque criticado, es universalmente reconocido como "aquí está la navegación".
- Swipe izquierda → en listas, significa "eliminar" (gracias a Mail de Apple).

- Arrastrar y soltar → cómo mover archivos entre carpetas en un escritorio real.

Ejemplo real:

- Tinder popularizó el swipe right = like, y hoy plataformas como LinkedIn lo usan para reclutamiento (sin necesidad de explicarlo).

¿Cuándo romper el patrón?

Solo si aportas un valor claro (ej: Pinterest usó "scroll infinito" cuando otros tenían paginación).

C. Diseño que no necesita instrucciones

Regla de oro: Si necesitas un tutorial, tu diseño ha fallado.

Técnicas para lograrlo:

- **Jerarquía visual obvia:** El botón principal es el más prominente (ej: Spotify hace que "Reproducir" sea verde y gigante).
- **Affordances claras:** Elementos que "invitan" a interactuar (ej: un botón 3D que parece clickeable).
- **Agrupación lógica:** Campos de formulario relacionados visualmente (como direcciones en un mismo bloque).

Ejemplo sencillo y elegante:

- Google Search: Un solo campo de texto centrado, con la lupa al lado. Cero explicaciones necesarias.

Casos de fracaso por ignorar modelos mentales

1. **Microsoft Word (Menú "Ribbon"):** En 2007, reemplazó menús tradicionales por pestañas horizontales. Los usuarios tardaron años en adaptarse.
2. **Instagram (rediseño de 2020 – botón de "Crear" reemplazado por Reels):** Instagram movió el botón central de "crear una publicación" para colocar "Reels", priorizando su nueva función sobre el comportamiento habitual de los usuarios. Los usuarios estaban acostumbrados a crear contenido con ese botón y de repente les llevaba a una sección distinta. El cambio fue confuso y rompió el flujo mental habitual.

8. JERARQUÍA VISUAL E INTERACCIÓN: EL ARTE DE GUIAR LA MIRADA

La jerarquía visual es el GPS de tu interfaz: le dice al usuario dónde mirar primero, segundo y tercero, sin que tenga que pensar. Un buen diseño jerárquico hace que la experiencia sea intuitiva, rápida y libre de estrés.

¿Por qué es crucial la jerarquía visual?

- El cerebro decide en 50 ms si una interfaz es confiable (Google Research).
- El 94% de los usuarios abandona un sitio web si la información es difícil de escanear (NNGroup).

Técnicas para Dominar la Jerarquía Visual

A. Tamaño = Importancia

Regla básica: Lo más grande se ve primero.

- **Ejemplo:**
 - Un título (h1) debe ser 2-3x más grande que el texto normal.
 - Botón principal (CTA) más grande que los secundarios.
- **Caso real:**

- Spotify hace el botón "Reproducir" enorme y verde, mientras que "Guardar" es más pequeño y discreto.

B. Color y Contraste

¿Qué funciona?

- Colores cálidos (rojo, naranja) atraen más la atención que los fríos (azul, verde).
- Contraste alto entre texto y fondo mejora la legibilidad.
- Un solo color acentuado para elementos clave (ej: botón de compra en rojo).

Ejemplo:

- Amazon usa amarillo para el botón "Añadir al carrito" (contraste alto sobre fondo blanco).

C. Ubicación Estratégica

Patrones de lectura comunes:

- F-Shaped Pattern (web): Los usuarios escanean en forma de "F" (primero horizontal, luego vertical).
- Z-Shaped Pattern (landing pages): Ideal para sitios con pocos elementos.

Zonas clave:

- Esquina superior izquierda: Logo y menú principal.
- Centro superior: Título principal.
- Final de la "F": Llamados a la acción (botones clave).

Ejemplo:

- Netflix coloca el botón "Reproducir" justo en el centro de la pantalla.

D. Espaciado y Agrupación (Ley de Proximidad)

¿En qué consiste?

- Los elementos relacionados deben estar juntos.
- Los elementos no relacionados deben tener espacio entre ellos.

Cómo aplicarlo:

- Formularios: Agrupar campos relacionados (ej: "Datos personales" vs. "Datos de pago").
- Tarjetas de producto: Usar márgenes claros para separar productos en un e-commerce.

Ejemplo:

- Airbnb agrupa precio, valoración y ubicación cerca de la imagen de la propiedad.

E. Tipografía y Peso Visual

Jerarquía textual:

1. Título principal (bold, grande).
2. Subtítulo (menos tamaño, semibold).
3. Cuerpo del texto (normal, legible).

Ejemplo:

- The New York Times usa tipografía en escalas claras:
 - Título: Georgia, 32px, bold.
 - Subtítulo: Georgia, 22px, regular.

- Cuerpo: Helvetica, 16px.

Ejemplo Práctico: Análisis de una Landing Page

Objetivo: Suscripción a un newsletter.

1. Título grande y en negrita ("Únete a nuestra comunidad").
2. Subtítulo más pequeño ("Recibe tips semanales").
3. Campo de email destacado (borde oscuro, placeholder claro).
4. Botón CTA en color contrastante ("Suscribirse" en verde sobre fondo blanco).
5. Texto legal pequeño y gris (para no distraer).

Errores Comunes que Rompen la Jerarquía

- ✗ Todos los elementos compiten por atención (todo está en negrita o colores brillantes).
- ✗ Falta de contraste (texto gris claro sobre fondo blanco).
- ✗ Espaciado inconsistente (los elementos parecen flotar sin relación).
- ✗ Demasiadas fuentes diferentes (más de 2-3 tipografías).

9. DISEÑO CENTRADO EN TAREAS Y OBJETIVOS DEL USUARIO

El diseño efectivo no se trata de lo que tú quieras mostrar, sino de lo que el usuario necesita hacer. Se debe evitar que tu interfaz sea como un aeropuerto con señales confusas. Es como ser un buen anfitrión: anticipas lo que tu invitado necesita antes de que lo pida.

¿Por qué enfocarse en tareas?

- Dato clave: El 55% de los usuarios abandona una app si no encuentra lo que busca en menos de 15 segundos (Google).
- **Ejemplo real:**
 - Uber no muestra "la historia de la empresa" al abrir la app. Muestra un botón gigante para pedir un viaje.

Los tres pilares del “Diseño centrado en tareas”

A. Conocer los contextos de uso (¿Dónde y cómo usan tu producto?)

- **Escenarios comunes:**
 - App de reparto → Usuarios con prisa, mirando el móvil en la calle.
 - Banca online → Usuarios en casa, revisando saldos con cuidado.
- **Herramientas:**
 - Entrevistas con usuarios: "¿Qué haces primero al abrir nuestra app?"
 - Mapas de calor: Ver qué zonas de tu interfaz se tocan más.

Ejemplo:

- WhatsApp pone la cámara accesible desde el chat (porque los usuarios suelen enviar fotos rápidas).

B. Priorizar lo relevante (Menos es más)

Regla del 80/20:

El 20% de las funciones generan el 80% del uso. Identificarlas y hacerlas obvias.

Técnicas:

- **Jerarquía por frecuencia de uso:**
 - En Spotify, "Buscar" y "Reproducir" están siempre visibles; "Ajustes" está escondido.

- **Eliminar el "ruido visual":**

- Ej: Google Search es solo un campo de texto (todo lo demás es secundario).

Error común:

- Saturar la pantalla de opciones "por si acaso" (como los menús de TV de pago con 500 canales).

C. Minimizar pasos (Cada clic es un riesgo de abandono)**Estadísticas impactantes:**

- Cada paso extra en un formulario reduce la conversión un 10% (Baymard Institute).

Cómo simplificar:

- Autocompletado: Direcciones, tarjetas de crédito guardadas.
- Progreso visible: "Paso 1 de 3" en checkouts.
- Acciones con un solo clic: Como el "Comprar ahora" de Amazon.

Ejemplo brillante:

- Duolingo permite empezar una lección sin registro (solo pide datos después de que el usuario ya está enganchado).

Ejemplo Práctico: App de Reparto de Comida

Objetivo principal del usuario: Saber dónde está su pedido y cuándo llegará.

Diseño centrado en tareas:**1. Pantalla principal:**

- Mapa grande con la ubicación del repartidor (80% de la pantalla).
- Tiempo estimado en negrita y grande.
- Botón de "Contactar al repartidor" flotante.

2. Escondido:

- Ajustes, historial de pedidos y promociones (accesibles con swipe o en menú).

¿Qué pasa si no se prioriza?

- Si la app muestra primero "Ofertas" o "Perfil del usuario", los usuarios se frustran.

Errores que matan la eficiencia

✗ Esconder la función principal (ej: hacer que "Pedir comida" esté en un menú hamburguesa).

✗ Sobrecargar con información irrelevante (ej: noticias de la empresa en un dashboard de banco).

✗ Flujos redundantes (ej: pedir confirmación en cada paso de un proceso simple).

10. DISEÑO “MOBILE FIRST” Y SU APLICACIÓN EN DISEÑO WEB

En la actualidad, la mayoría de los accesos a internet se realizan desde dispositivos móviles. Según diferentes estudios de uso digital, más del 60% del tráfico web mundial proviene de smartphones y tablets, mientras que el escritorio ha pasado a un segundo plano. Este cambio de hábitos obliga a los diseñadores a adoptar un enfoque donde lo móvil es el punto de partida del diseño.

¿Qué es el enfoque Mobile First?

El diseño Mobile First consiste en planificar y diseñar primero para pantallas pequeñas (smartphones) y después escalar y adaptar la interfaz a dispositivos más grandes (tablets, portátiles, monitores).

Este enfoque se basa en tres principios fundamentales:

- Priorizar lo esencial: el espacio reducido obliga a mostrar solo lo más importante.
- Optimizar la experiencia: interfaces simples, claras y rápidas para usuarios que suelen estar en movimiento.
- Escalar progresivamente: una vez definido lo móvil, añadir mejoras visuales y funcionales para pantallas más grandes.

Ventajas del Mobile First

Responde a los hábitos actuales de los usuarios: si la mayoría navega en móvil, la experiencia debe estar optimizada primero para ese entorno.

- Diseño más claro y minimalista: obliga a eliminar lo innecesario y centrarse en lo que realmente importa al usuario.
- Facilidad de adaptación: es más sencillo ampliar un diseño móvil a escritorio que intentar reducir un diseño de escritorio al tamaño de un móvil (donde se pierde claridad y funcionalidad).
- Mejor rendimiento: un diseño ligero pensado para móvil suele cargar más rápido, mejorando la experiencia y el SEO.

Aplicación práctica en diseño web

En un proyecto Mobile First, el flujo de trabajo sería:

- Definir el contenido esencial: ¿qué necesita hacer el usuario en móvil? (ejemplo: en un e-commerce, ver productos, filtrar y comprar en pocos pasos).
- Diseñar interfaces adaptadas a pantallas pequeñas: botones grandes, menús simplificados (hamburguesa), formularios cortos.
- Usar breakpoints para ampliar a escritorio: en tablet/desktop se añade más espacio para imágenes, información adicional o elementos secundarios.
- Testear en móvil primero: validar que la navegación es fluida y sin fricciones antes de pensar en la versión desktop.

👉 **Ejemplo real:** Instagram nació como una app móvil; su diseño pensado para pantallas pequeñas fue tan exitoso que la versión web se adaptó después manteniendo la misma lógica visual.

11. RECURSOS RECOMENDADOS PARA APRENDER UI/UX

Libros

1. "No me hagas pensar" - Steve Krug
2. "The Design of Everyday Things" - Don Norman
"Articulating Design Decisions" - Tom Greever

Herramientas

1. Figma (Diseño colaborativo)
<https://www.figma.com/>
Herramienta líder de diseño UI/UX

 **Blogs y Comunidades**

1. NNGroup (Nielsen Norman Group)
<https://www.nngroup.com/articles/>
Artículos basados en investigación UX
2. Smashing Magazine
<https://www.smashingmagazine.com/>
Recursos para diseñadores web
3. UX Collective (Medium)
<https://uxdesign.cc/>
Comunidad activa de diseñadores

 **Canales de YouTube**

1. AJ&Smart
<https://www.youtube.com/c/AJSmartDesign>
Design Sprints y metodologías UX
2. Flux
<https://www.youtube.com/c/FluxWithRanSegall>
Diseño UI/UX y freelancing
3. Figma
<https://www.youtube.com/c/Figmadesign>
Tutoriales oficiales de Figma

 **Podcasts Recomendados**

1. Design Matters
<https://www.designmattersmedia.com/>
Entrevistas con diseñadores líderes
2. UX Coffee Break
<https://uxcoffeebreak.com/>
Charlas cortas sobre UX

 **Plataformas de Inspiración**

1. Dribbble
<https://dribbble.com/>
Portafolios de diseño UI
2. Awwwards
<https://www.awwwards.com/>
Los mejores diseños web
3. Behance
<https://www.behance.net/>
Plataforma para mostrar trabajos de diseño