

Desarrollo de Interfaces

Unidad 01. Planificación de interfaces gráficas



Autor: Sergi García



Actualizado Septiembre 2025

Licencia



Reconocimiento - No comercial - Compartir igual (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 y contexto profesional	4
2. Tipos de interfaces gráficas	5
2.1 Interfaces de escritorio (Windows, MacOS, Linux)	5
2.2 Interfaces móviles (Android, iOS)	6
2.3 Interfaces híbridas (app multiplataforma)	6
2.4 Realidad Aumentada (AR) y Realidad Virtual (VR)	6
2.5 Interfaces gestuales (Kinect, sensores)	6
2.6 Interfaces por voz (asistentes personales: Siri, Alexa...)	7
2.7 Resumen de los distintos tipos de interfaces	7
3. Usabilidad y experiencia de usuario	7
3.1 Definición y principios de usabilidad	7
3.2 Diferencias entre usabilidad y experiencia de usuario (UX)	8
3.3 Métricas de evaluación	8
3.4 Heurísticas de usabilidad de Jakob Nielsen	8
3.5 Accesibilidad: principios básicos	9
4. Diseño visual y percepción	9
4.1 Teoría de la Gestalt aplicada al diseño	9
4.2 Colorimetría y representación del color	9
4.3 Tipografía y legibilidad	10
4.4 Composición y coherencia visual	11
4.5 Iconografía y semiótica visual	11
5. Guías de estilo	11
5.1 ¿Qué es y para qué sirve una guía de estilo?	11
5.2 Elementos comunes en una guía de estilo	12
5.3 Ventajas de usar una guía de estilo	12
5.4 Ejemplos reales de guías de estilo	12
6. Componentes y estructuras de una interfaz	13
6.1 Componentes típicos	13
6.2 Estructura de ventanas	13
6.3 Interfaces SDI vs MDI	13

6.4 Patrones de interacción	13
7. Planificación del diseño de la interfaz	14
7.1 Investigación del usuario	14
7.2 Modelado conceptual	14
7.3 Flujos de navegación y diagramas	14
7.4 Mapa de pantallas o módulos	14
7.5 Wireframes (estructuras básicas)	14
7.6 Mockups (maquetas visuales detalladas)	14
7.7 Documentación del diseño	15
8. Herramientas de diseño y prototipado (sin código)	15
8.1 Introducción a herramientas visuales	15
8.2 Comparativa básica: funcionalidad y uso educativo	15
8. Bibliografía	16

UNIDAD 01. PLANIFICACIÓN DE INTERFACES GRÁFICAS

1. INTRODUCCIÓN Y CONTEXTO PROFESIONAL

Una **interfaz gráfica de usuario (GUI - Graphical User Interface)** es el espacio visual y funcional donde el usuario interactúa con un sistema informático, una aplicación o un dispositivo. Está formada por elementos visuales como ventanas, botones, menús, iconos, formularios, textos y otros componentes gráficos que permiten que el usuario controle y reciba información del sistema de manera sencilla, intuitiva y eficiente.

- **Características clave de como deben ser las interfaces gráficas:**
 - Visual y táctil (en pantallas táctiles).
 - Basada en metáforas visuales reconocibles (carpetas, papeles, botones).
 - Interactiva: responde a las acciones del usuario (clics, toques, gestos).
 - Puede estar en dispositivos diversos: ordenadores, móviles, tablets, dispositivos inteligentes.
- **El papel del diseñador de interfaces en proyectos digitales**
 - El **diseñador de interfaces gráficas** es el profesional encargado de crear estos espacios visuales donde los usuarios interactúan con la tecnología. Su trabajo permite que la experiencia sea:
 - **Fácil de entender:** que el usuario sepa qué hacer sin explicaciones complicadas.
 - **Eficiente:** que pueda realizar sus tareas con el mínimo esfuerzo y tiempo.
 - **Atractiva:** que el diseño sea visualmente agradable y coherente.
 - **Accesible:** que pueda usarla cualquier persona, independientemente de sus capacidades.
- **Funciones principales:**
 - Investigar y comprender las necesidades y comportamientos de los usuarios.
 - Diseñar estructuras, componentes y flujos de navegación.
 - Elegir colores, tipografías, iconos y estilos que transmitan la identidad del producto y faciliten su uso.
 - Elaborar prototipos, wireframes y mockups para mostrar la interfaz antes de desarrollarla.
 - Colaborar con desarrolladores para asegurar que el diseño se implemente correctamente.
 - Realizar pruebas y mejoras continuas basadas en el feedback de usuarios.
- **Colaboración con equipos de desarrollo, UX y cliente**
 - El diseño de interfaces es un proceso **colaborativo** que involucra varios perfiles profesionales:
 - **Diseñadores UX (Experiencia de Usuario):** Se centran en la investigación del usuario, análisis de necesidades, arquitectura de la información y la usabilidad general.
 - **Diseñadores UI (Interfaz de Usuario):** Se encargan del aspecto visual y la interacción directa, es decir, el diseño gráfico y la planificación de componentes.
 - **Desarrolladores front-end:** Programan la interfaz en código, implementando los diseños con tecnologías web, móviles o de escritorio.
 - **Clientes y usuarios finales:** Proveen los requisitos, expectativas y validan el producto para que cumpla con sus objetivos.
 - El diseñador de interfaces actúa como puente entre estas partes, traduciendo las

necesidades del usuario y cliente en un diseño visual y funcional que los desarrolladores puedan construir.

- **Objetivos de una interfaz gráfica**

- Una interfaz gráfica eficaz debe cumplir tres objetivos fundamentales:

- **Facilitar la interacción**

- Permitir que el usuario realice sus tareas sin confusión ni esfuerzo excesivo.
- Ofrecer una navegación clara y componentes reconocibles.
- Proveer feedback inmediato a las acciones del usuario para que se sienta en control.

- **Transmitir información**

- Mostrar datos, estados o mensajes de forma comprensible y visualmente organizada.
- Priorizar la información importante para evitar saturar al usuario.
- Usar códigos visuales (colores, iconos, tipografía) que ayuden a interpretar el contenido.

- **Generar confianza**

- Que el usuario perciba la interfaz como profesional, fiable y segura.
- Mantener coherencia visual que refuerce la identidad del producto o marca.
- Evitar errores y ofrecer ayudas o mensajes claros cuando algo no funcione.

2. 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.

2.1 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 desplegados, 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.).

2.2 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.

2.3 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.

2.4 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.

2.5 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.

2.6 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.

2.7 Resumen de los distintos tipos de interfaces

Tipo de interfaz	Dispositivo típico	Interacción principal	Retos principales
Escritorio	PC, portátil	Ratón y teclado	Complejidad, multitarea
Móvil	Smartphone, tablet	Táctil	Espacio reducido, gestos
Híbrida	Móviles y escritorio	Táctil y ratón	Adaptación multi-dispositivo
AR / VR	Gafas, smartphones	Movimiento, visual 3D	Inmersión, ergonomía, mareos
Gestual	Sensores, cámaras	Movimientos corporales	Precisión, aprendizaje de gestos
Voz	Altavoces, smartphones	Comandos de voz	Reconocimiento, contexto, ruido

3. USABILIDAD Y EXPERIENCIA DE USUARIO

3.1 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.

3.2 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.

3.3 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”).

3.4 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.

3.5 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 icono 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.

4. DISEÑO VISUAL Y PERCEPCIÓN

4.1 Teoría de la Gestalt aplicada al diseño

La **Teoría de la Gestalt** proviene de la psicología y explica cómo percibimos patrones y estructuras visuales. En diseño, se utiliza para crear interfaces que el usuario pueda entender de forma rápida y natural, gracias a la forma en que organizamos los elementos.

Principios clave:

- **Proximidad:** elementos cercanos se perciben como relacionados.
- **Semejanza:** elementos similares en forma, color o tamaño se agrupan visualmente.
- **Continuidad:** la mente tiende a seguir líneas y patrones continuos.
- **Figura-fondo:** distinguir un objeto (figura) de su fondo para enfocar la atención.
- **Cierre:** la mente completa formas incompletas para crear figuras completas.
- **Simetría:** elementos simétricos se perciben como parte de un mismo grupo o como visualmente agradables.

Estos principios ayudan a organizar contenido, menús, botones y otros elementos para mejorar la comprensión y usabilidad.

4.2 Colorimetría y representación del color

El color es una herramienta poderosa en el diseño visual. Su correcta aplicación mejora la comunicación y la experiencia del usuario.

Modelos de color más comunes:

- **RGB (Red, Green, Blue):** modelo aditivo usado en pantallas, combina luz roja, verde y azul para crear colores.
- **HSL (Hue, Saturation, Lightness):** modelo que describe el color en términos de tono, saturación y luminosidad, útil para seleccionar colores armónicos.
- **HEX:** código hexadecimal que representa colores en formatos digitales, muy usado en diseño web y desarrollo.

Psicología del color

Cada color puede transmitir diferentes emociones y significados, por ejemplo:

- **Rojo:** energía, urgencia, alerta.
- **Azul:** confianza, calma, profesionalismo.
- **Verde:** naturaleza, crecimiento, seguridad.
- **Amarillo:** atención, optimismo, advertencia.

El diseño debe considerar estos aspectos para alinear la interfaz con el mensaje y el público objetivo.

Contraste, armonía y accesibilidad visual

- **Contraste:** Importante para que el texto y los elementos sean legibles y destacables, especialmente para personas con dificultades visuales.
- **Armonía:** Combinación de colores que resultan agradables y coherentes.
- **Accesibilidad:** Garantizar que combinaciones de color funcionen para usuarios con daltonismo u otras discapacidades visuales.

4.3 Tipografía y legibilidad

La tipografía es mucho más que elegir una fuente "bonita". Es una herramienta de comunicación esencial que:

- **Transmite emociones y tono:** una tipografía puede hacer que un mensaje se sienta serio, divertido, elegante o tecnológico antes de que la gente termine de leerlo.
- **Mejora (o empeora) la legibilidad:** la elección correcta permite leer durante mucho tiempo sin fatigarse. La incorrecta cansa la vista y hace que el lector abandone el contenido.
- **Establece jerarquía y orden:** usando diferentes pesos (negrita, light) y estilos, podemos guiar al lector, indicándole qué es importante, qué es un título y qué es un cuerpo de texto.
- **Construye identidad de marca:** tipografías específicas se asocian inmediatamente a marcas (como la futura de Disney o la Helvetica de Panasonic).

Principales tipos de fuentes:

- **Serif:** Fuentes con remates (ej. Times New Roman), asociadas a textos impresos, formales y tradicionales.
- **Sans-serif:** Fuentes sin remates (ej. Arial, Helvetica), más modernas y legibles en pantalla.
- **Monoespaciadas:** Todas las letras ocupan el mismo espacio (ej. Courier), usadas en programación y algunos diseños técnicos.
- **Caligráficas:** imitan la escritura a mano, con trazos fluidos y, a menudo, letras conectadas entre sí.
- **Decorativas:** son tipografías altamente estilizadas y temáticas. Su diseño prioriza el impacto visual sobre la legibilidad.

Además de la fuente, existen elementos que transforman un bloque de texto plano en una composición clara, atractiva y fácil de navegar: **tamaño, jerarquía, interlineado y espaciado.**

- **Tamaño:** Importante para la legibilidad y jerarquía visual.
- **Jerarquía:** Usar diferentes tamaños y estilos (negrita, cursiva) para organizar la información y destacar títulos o secciones.

- **Interlineado:** Espacio entre líneas que mejora la lectura y evita saturación.
- **Espaciado:** Margen y padding para separar elementos y mejorar la composición.

4.4 Composición y coherencia visual

Una adecuada composición con coherencia visual es lo que hace que un sitio web o una app se sienta sólido, profesional y fácil de usar, incluso si el usuario no sabe por qué, además de facilitar la navegación y comprensión de la aplicación. Algunos de los principales aspectos a tener en cuenta al realizar nuestra composición son los siguientes:

Rejillas, alineación y márgenes

- Las **rejillas** estructuran el espacio para colocar los elementos de manera ordenada.
- La **alineación** asegura que los elementos visuales están bien organizados y crean fluidez.
- Los **márgenes** generan espacio entre elementos para evitar aglomeración.

Consistencia visual entre pantallas

Mantener una coherencia en el uso de colores, tipografías, botones e iconos entre diferentes pantallas ayuda a que el usuario reconozca patrones y reduzca la curva de aprendizaje.

4.5 Iconografía y semiótica visual

Los iconos y símbolos comunican de forma rápida e intuitiva.

- **Metáforas visuales**
 - Usar imágenes o iconos que representen conceptos de manera figurada (ejemplo: un sobre para el email, una lupa para buscar).
- **Simbología y lenguaje icónico**
 - El lenguaje icónico debe ser universal y entendible para evitar confusiones.
 - Los iconos deben ser simples, claros y coherentes con el estilo general del diseño.

5. GUÍAS DE ESTILO

5.1 ¿Qué es y para qué sirve una guía de estilo?

Una **guía de estilo** es un documento o conjunto de normas que establece las reglas y recomendaciones para el diseño visual y la interacción dentro de un proyecto digital o una marca. Su objetivo principal es garantizar la **coherencia** visual y funcional a lo largo de todos los productos, aplicaciones o sitios web, independientemente de quién realice el diseño o desarrollo.

Esta herramienta sirve para:

- Mantener una imagen y experiencia unificada.
- Facilitar la colaboración entre diseñadores, desarrolladores y otros equipos.
- Acelerar el proceso de diseño y desarrollo al contar con elementos y pautas predefinidas.
- Asegurar la accesibilidad y usabilidad mediante estándares claros.
- Permitir la **escalabilidad** del proyecto, incorporando nuevos elementos sin perder la identidad visual.

5.2 Elementos comunes en una guía de estilo

Las guías de estilo suelen incluir definiciones y ejemplos concretos de varios elementos visuales y funcionales clave, tales como:

- **Paleta de colores:** Conjunto definido de colores principales, secundarios y de acento, con códigos específicos (HEX, RGB, etc.) que garantizan uniformidad en todas las interfaces.
- **Tipografías:** Fuentes tipográficas autorizadas, tamaños, estilos (negrita, cursiva), jerarquías y reglas para asegurar legibilidad y coherencia.
- **Botones:** Diseño de botones incluyendo formas, tamaños, colores, estados (normal, hover, clic, deshabilitado) y animaciones o transiciones.
- **Iconos:** Conjunto de iconos oficiales, estilo gráfico (lineal, relleno, flat), tamaños y uso recomendado para facilitar la comunicación visual.
- **Componentes:** Elementos reutilizables como menús, formularios, barras de navegación, tarjetas, modales, etc., con sus estilos y comportamientos definidos.

Además, pueden incluir directrices sobre:

- Espaciado y márgenes.
- Estilos para textos (links, listas, citas, cajas de texto).
- Imágenes y gráficos, así como interacciones y animaciones.
- Accesibilidad (contrastes, tamaños mínimos, navegación por teclado).

5.3 Ventajas de usar una guía de estilo

Implementar y seguir una guía de estilo aporta múltiples beneficios en el proceso de diseño y desarrollo:

- **Coherencia:** mantiene un aspecto visual homogéneo en todas las pantallas y productos, fortaleciendo la identidad de marca y la experiencia del usuario.
- **Escalabilidad:** facilita la incorporación de nuevos elementos o funcionalidades sin romper el diseño establecido, permitiendo el crecimiento ordenado del proyecto.
- **Reutilización:** favorece el uso repetido de componentes y estilos, ahorrando tiempo y esfuerzo en el diseño y codificación.
- **Colaboración:** simplifica la comunicación y el trabajo conjunto entre diseñadores, desarrolladores y otros roles, evitando malentendidos o duplicidad de trabajo.
- **Calidad:** ayuda a mantener estándares de usabilidad, accesibilidad y estética, mejorando la satisfacción del usuario final.

5.4 Ejemplos reales de guías de estilo

Existen varias guías de estilo ampliamente utilizadas en la industria que pueden servir como referencia o inspiración:

- **Material Design (Google):** guía de diseño visual y de interacción desarrollada por Google para aplicaciones web y móviles. Incorpora principios de diseño basado en materiales físicos y enfatiza animaciones, profundidad y claridad.
- **Fluent UI (Microsoft):** conjunto de directrices para el diseño y desarrollo de interfaces en productos Microsoft, centrado en la fluidez, accesibilidad y adaptabilidad a diferentes dispositivos.
- **Apple Human Interface Guidelines (HIG):** normas oficiales para diseñar aplicaciones en el ecosistema Apple, con énfasis en la simplicidad, claridad y consistencia en macOS, iOS,

watchOS y tvOS.

6. COMPONENTES Y ESTRUCTURAS DE UNA INTERFAZ

6.1 Componentes típicos

Las interfaces gráficas están formadas por diversos elementos visuales y funcionales que permiten la interacción entre el usuario y el sistema. Los componentes más comunes incluyen:

- **Botones:** Elementos clicables que ejecutan acciones específicas. Pueden variar en forma, tamaño, color y estado (activo, inactivo, presionado).
- **Menús:** Listados desplegables o fijos que agrupan opciones o comandos para la navegación o selección.
- **Formularios:** Conjuntos de campos donde el usuario introduce datos (texto, selección, casillas, etc.). Incluyen validaciones y ayudas para facilitar la entrada correcta.
- **Cuadros de diálogo:** Ventanas emergentes que solicitan información, presentan alertas o confirman acciones del usuario.
- **Sliders (deslizadores):** Controles para seleccionar valores dentro de un rango continuo o discreto, como volumen o brillo.

Otros componentes pueden ser pestañas, listas desplegables, barras de progreso, iconos interactivos, etc.

6.2 Estructura de ventanas

La organización de la interfaz en ventanas es clave para gestionar la información y las acciones del usuario. Dos conceptos fundamentales son:

- **Ventanas modales:** son ventanas que requieren la atención inmediata del usuario, bloqueando la interacción con otras partes de la interfaz hasta que se cierre o complete una acción. Ejemplos: diálogos de confirmación, formularios críticos.
- **Ventanas no modales:** permiten al usuario interactuar simultáneamente con otras ventanas o elementos de la interfaz, facilitando multitarea o consultas paralelas. Ejemplo: ventanas de ayuda o paneles flotantes.

6.3 Interfaces SDI vs MDI

- **SDI (Single Document Interface):** cada ventana de la aplicación maneja un único documento o tarea. Ejemplo: editores de texto donde cada documento está en su propia ventana independiente.
- **MDI (Multiple Document Interface):** la aplicación contiene múltiples documentos o ventanas dentro de una ventana principal, permitiendo organizarlos y cambiar entre ellos fácilmente. Ejemplo: programas de edición gráfica o desarrollo que gestionan varios archivos simultáneamente.

6.4 Patrones de interacción

Para lograr interfaces intuitivas y eficientes, es importante aplicar patrones y principios que mejoren la interacción:

- **Consistencia:** mantener un diseño uniforme en estilos, comportamientos y nomenclaturas para que el usuario aprenda y reconozca fácilmente cómo usar la interfaz.
- **Feedback:** proporcionar respuestas claras e inmediatas a las acciones del usuario, como

cambios visuales, mensajes o sonidos, para confirmar que una operación se está procesando o ha concluido.

- **Accesibilidad:** garantizar que la interfaz sea usable por personas con diferentes capacidades, incluyendo navegación por teclado, contraste adecuado, tamaños legibles y soporte para tecnologías de asistencia.

7. PLANIFICACIÓN DEL DISEÑO DE LA INTERFAZ

7.1 Investigación del usuario

La planificación comienza con la **investigación del usuario**, fundamental para comprender a quién va dirigida la interfaz y cómo debe adaptarse a sus características y necesidades.

- **Personas:** son perfiles o arquetipos representativos que sintetizan características, motivaciones, comportamientos y objetivos de grupos de usuarios reales. Ayudan a diseñar con foco en usuarios concretos y a anticipar sus expectativas y problemas.

7.2 Modelado conceptual

El modelado conceptual traduce la información obtenida sobre los usuarios en una estructura lógica que guía el diseño.

- **Tareas y necesidades:** identificación de las acciones que los usuarios deben realizar y los requisitos que la interfaz debe cubrir.
- **Estructura lógica:** organización clara y coherente de los contenidos y funciones, facilitando la navegación y el acceso a la información.

7.3 Flujos de navegación y diagramas

Para representar cómo los usuarios interactuarán con la interfaz se utilizan:

- **Flujos de navegación:** diagramas que muestran las rutas posibles entre pantallas o módulos según las acciones del usuario, identificando caminos comunes y alternativos.
- **Diagramas de interacción:** modelos visuales que detallan las transiciones, estados y respuestas del sistema ante eventos del usuario.

7.4 Mapa de pantallas o módulos

Un **mapa de pantallas** es un esquema que presenta todas las pantallas o módulos principales de la interfaz y sus relaciones jerárquicas o funcionales. Permite visualizar la estructura global y planificar el desarrollo progresivo.

7.5 Wireframes (estructuras básicas)

Los **wireframes** son esquemas simplificados que muestran la disposición general de los elementos en cada pantalla, sin detalles gráficos, centrados en la funcionalidad y la jerarquía visual.

7.6 Mockups (maquetas visuales detalladas)

Los **mockups** son representaciones más elaboradas y realistas de las pantallas, que incluyen colores, tipografías, iconos y estilo gráfico final. Permiten evaluar el aspecto visual y la experiencia antes de la implementación.

7.7 Documentación del diseño

Todo el proceso y las decisiones deben quedar registradas en documentación clara y accesible, que incluya:

- Descripciones de personas y casos de uso.
- Diagramas y mapas.
- Wireframes y mockups.
- Normas y estándares aplicados.
- Requisitos funcionales y de usabilidad.

Esta documentación es clave para la comunicación entre equipos y para futuras revisiones o actualizaciones del diseño.

8. HERRAMIENTAS DE DISEÑO Y PROTOTIPADO (SIN CÓDIGO)

8.1 Introducción a herramientas visuales

En el diseño de interfaces, las herramientas de prototipado permiten crear representaciones visuales y funcionales de la interfaz sin necesidad de programar, facilitando la comunicación, iteración y validación antes del desarrollo.

Algunas de las más populares son:

- **Figma:** herramienta online colaborativa que permite diseñar interfaces, crear prototipos interactivos y trabajar en equipo en tiempo real. Es muy usada por su facilidad y capacidades avanzadas.
- **Adobe XD:** software de Adobe orientado al diseño y prototipado de interfaces con herramientas intuitivas, integración con el ecosistema Adobe y opciones para compartir prototipos.
- **Sketch:** aplicación para macOS enfocada en diseño UI/UX, conocida por su simplicidad, gran cantidad de plugins y uso profesional en diseño web y móvil.
- **Pencil Project:** herramienta de código abierto para crear diagramas y wireframes, ideal para entornos educativos o quienes buscan software gratuito y sencillo.
- **Balsamiq:** software que simula el estilo de boceto a mano para wireframes rápidos y comprensibles, favoreciendo la concentración en la estructura y usabilidad antes del diseño visual.
- **Canva:** plataforma online muy popular por su simplicidad y biblioteca de plantillas. Aunque está más orientada al diseño gráfico general, también permite crear prototipos básicos, presentaciones de interfaces y materiales visuales para comunicar ideas de diseño de manera rápida y accesible.

8.2 Comparativa básica: funcionalidad y uso educativo

Herramienta	Plataformas	Nivel de dificultad	Colaboración	Precio
Figma	Web, Windows, Mac	Bajo - Medio	Sí	Gratis y suscripción
Adobe XD	Windows, Mac	Medio	Sí	Gratis y suscripción

Sketch	Mac	Medio	Limitada	Pago
Pencil Project	Windows, Mac, Linux	Bajo	No	Gratis
Balsamiq	Web, Windows, Mac	Bajo	Limitada	Pago
Canva	Web, Windows, Mac	Bajo - Medio	Sí	Gratis y suscripción

Con estas herramientas se pueden:

- **Crear wireframes:** diseño básico de la estructura, disposición y funcionalidad sin distracciones visuales.
- **Desarrollar mockups:** maquetas visuales detalladas que representan el aspecto final, incluyendo colores, tipografías, iconos y estilos.
- **Prototipar interacciones:** definir flujos y transiciones para simular la experiencia de usuario y validar la navegación.

8. BIBLIOGRAFÍA

8.1. Usabilidad y Experiencia de Usuario (UX)

- Usabilidad web (Slideshare):
<https://www.slideshare.net/laurafolgadogalache/usabilidad-web-58227098>
- Usabilidad móvil (Slideshare):
<https://www.slideshare.net/laurafolgadogalache/usabilidad-mvil-58227090>
- Sistema de escalas de usabilidad (UX Panol):
<https://uxpanol.com/teoria/sistema-de-escalas-de-usabilidad-que-es-y-para-que-sirve/>
- Métricas para cuantificar usabilidad (TutsPlus):
<https://webdesign.tutsplus.com/es/tutorials/3-metrics-for-quantifying-usability--cms-29150>
- 10 reglas heurísticas de Jakob Nielsen (Braintive):
<http://www.braintive.com/10-reglas-heuristicas-de-usabilidad-de-jakob-nielsen/>
- Diferencias entre ventanas modales y no modales:
<https://blog.ida.cl/disenio/ldiferencia-modal-pop-up-pop-over-light-box/>
- SDI vs MDI (diferencias en interfaces de ventanas):
<https://diffzi.com/mdi-vs-sdi/>
- Wireframe vs Mockup vs Prototype (UX Planet):
<https://uxplanet.org/wireframe-mockup-prototype-what-is-what-8cf2966e5a8b>

8.2. Diseño Visual y Percepción

- Introducción a la teoría de la Gestalt aplicada al diseño:
<https://www.interaction-design.org/literature/topics/gestalt-principles>
- Psicología del color y teoría del color:
<https://www.sessions.edu/color-calculator/>
- Modelos de color (RGB, HSL, HEX) explicados:

- https://www.w3schools.com/colors/colors_picker.asp
- Tipografía: fundamentos y tipos de fuentes:
<https://www.fonts.com/content/learning/fyti/typeface-anatomy>
- Semiótica visual y metáforas en diseño:
<https://www.nngroup.com/articles/visual-metaphors/>

8.3. Guías de estilo y diseño coherente

- ¿Qué es una guía de estilo? (Adobe XD Ideas):
<https://xd.adobe.com/ideas/process/ui-design/what-is-a-style-guide/>
- Material Design (Google):
<https://material.io/design>
- Fluent UI (Microsoft):
<https://developer.microsoft.com/en-us/fluentui>
- Apple Human Interface Guidelines (HIG):
<https://developer.apple.com/design/human-interface-guidelines/>

8.4. Tipos de Interfaces Gráficas

- Introducción a interfaces AR y VR (Unity Learn):
<https://learn.unity.com/tutorial/introduction-to-vr>
- Interfaces gestuales con Kinect (Microsoft Docs):
<https://learn.microsoft.com/en-us/azure/kinect-dk/>
- Asistentes de voz (Amazon Alexa Developer):
<https://developer.amazon.com/en-US/alexa>

8.5. Planificación y Prototipado

- Metodologías de planificación de interfaces (Interaction Design Foundation):
<https://www.interaction-design.org/literature/topics/interface-design>
- Herramientas de prototipado sin código:
 - Figma: <https://www.figma.com/>
 - Adobe XD: <https://www.adobe.com/products/xd.html>
 - Sketch: <https://www.sketch.com/>
 - Pencil Project: <https://pencil.evolus.vn/>
 - Balsamiq: <https://balsamiq.com>
 - Canva: <https://canva.com>