

INTRODUCCIÓN AL DISEÑO DE INTERFACES WEB

Contenido

1.	INTRODUCCIÓN	3
2.	ELEMENTOS DE DISEÑO: PERCEPCIÓN VISUAL.....	3
3.	FUNDAMENTOS DE LA COMPOSICIÓN.....	3
4.	COLOR, TIPOGRAFÍA, ICONOS.....	5
5.	COMPONENTES DE UNA INTERFAZ WEB.....	8
6.	LENGUAJES DE MARCAS. MAQUETACIÓN.....	10
7.	MAPA DE NAVEGACIÓN. PROTOTIPOS	12
8.	GUÍAS DE ESTILO	12
9.	PROTOTIPADO WEB	13
10.	HERRAMIENTAS PARA DESARROLLO WEB.....	14
11.	PLANTILLA DE DISEÑO	16
12.	INTERACCIÓN PERSONA-ORDENADOR	16

1. INTRODUCCIÓN

Un sitio web es un conjunto de páginas web agrupadas bajo un dominio y que comparten una dirección en la Web.

La mayoría de los sitios web tienen una página principal (homepage) desde la que se accede a todos los contenidos ofrecidos por el sitio. Desde el punto de vista del diseño uno de los objetivos de este módulo es que todas las páginas que componen un mismo sitio web cumplan criterios de homogeneidad y consistencia. Un usuario podría acceder al homepage o a cualquiera de las otras páginas del sitio y el diseño debe ser común porque será la imagen del sitio y, por tanto, la imagen de la persona u organización a la que representa.

Siempre que se crea un sitio web es porque se quiere comunicar algo, ya sea con fines comerciales o personales y es, en consecuencia, imprescindible "venderse bien". Nuestro diseño tendrá que ser atractivo y funcional. El diseñador Web no puede controlar los contenidos que su cliente desea mostrar, pero sí decide cómo se organiza el sitio y cómo se muestra la información.

2. ELEMENTOS DE DISEÑO: PERCEPCIÓN VISUAL

El primer impacto de un sitio web en un usuario será el visual. La forma, tamaño, ubicación, color, tipografía de cada uno de los elementos influirá en él. A modo de ejemplo, imaginemos una página para el Real Madrid diseñada con los colores azul y grana y como lo percibirían los aficionados del Real Madrid que son los destinatarios de la información que allí se expone. O quizá una tienda virtual donde no sea fácil encontrar el enlace para hacer una compra.

Habrá que buscar un equilibrio entre los elementos que constituyen la interfaz para que haya una comunicación eficaz. No es conveniente excederse en la utilización de elementos por el mero hecho de ponerlos o para que se "luzca" el diseñador; un excesivo ruido o distracciones puede enmascarar lo que pretendemos comunicar.

Lo primero que se debe determinar es el área de diseño, o sea, qué tamaño se asigna a cada uno de los elementos. Podemos usar sólo texto o sólo imágenes o bien mezcla de ambos. Podemos hacer uso de espacios en blanco para equilibrar los elementos gráficos. Pero sobre todo debe ser un diseño adecuado al mensaje que se quiere transmitir.

3. FUNDAMENTOS DE LA COMPOSICIÓN

La composición se define como una distribución o disposición de todos los elementos que incluiremos en un diseño o composición, de una forma perfecta y equilibrada.

Podemos resumir diciendo que el objetivo es ordenar todos los elementos de nuestra web intentando impactar visualmente al público receptor. No hay reglas mágicas que garanticen esto, pero sí hay pautas sobre la percepción que el diseñador puede seguir.

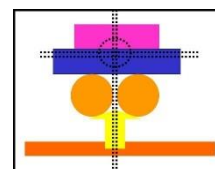
- **Componentes psicosomáticos del sistema nervioso:** Las distintas formas, ubicaciones, colores, etc. suelen interpretarlas nuestro cerebro con un significado definido.
- **Componentes de tipo cultural:** Influyen en la interpretación por causa de nuestra cultura y

educación. Por ejemplo, en Occidente el luto se relaciona con el color negro, mientras que en los países orientales se relaciona con el blanco.

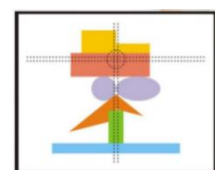
- **Experiencias compartidas con el entorno:** Hay parejas de conceptos altamente arraigados: verde/hierba, azul/cielo, blanco/frío. Son dualidades que aprendemos desde nuestra infancia y que posteriormente usamos como patrones para interpretar lo que vemos.

3.1. Equilibrio visual

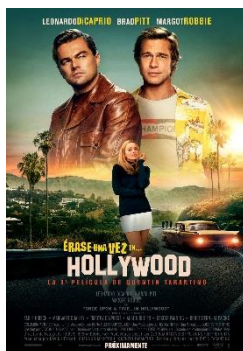
El equilibrio formal o simétrico es aquel que está basado en la simetría, es decir que posee igual peso en ambos lados del eje central (centro óptico) de nuestra web o pantalla. Es una composición poco creativa, pero aporta calma y estabilidad.



El equilibrio informal o asimétrico, sin embargo, está cargado de fuerza y dinamismo al prescindir de la simetría al contrastar los pesos visuales, tanto en formas como en colores, buscando una asimetría intencionada.



- Las **formas grandes** tienen más peso visual que las pequeñas.
- Las **formas irregulares**, más que las regulares. También aumentan el peso visual los **colores luminosos**.
- Y también la **posición** de los elementos es fundamental a la hora añadirles peso.



3.2. La tensión compositiva

La tensión compositiva es lo opuesto al equilibrio. Su objetivo es fijar la mirada del observador. Se puede conseguir con formas agudas e irregulares. Hay técnicas para provocar esa tensión:

- **Técnica sugestiva:** consiste en dirigir la atención a un punto determinado utilizando elementos de apoyo. Por ejemplo, imágenes de personas que miran hacia un punto determinado (que sería el punto de interés).
- **Técnica rítmica:** basada en la tendencia innata del ojo humano a completar secuencias de elementos.

Analiza la sensación que te provocan los diseños de las siguientes páginas:

<http://art.yale.edu>

http://www.vatican.va/holy_father/special_features/hf_jp_ii_xxv_en.htm.

4. COLOR, TIPOGRAFÍA, ICONOS

4.1. Color

En los entornos gráficos digitales, los colores se forman a partir de tres básicos: rojo, verde y azul. La intensidad de cada componente se expresa como un número entre 0 y 255 (generalmente, entre 00 y FF, en sistema hexadecimal). Así, cada color se representa como un número hexadecimal de 6 cifras donde las dos primeras corresponden al rojo, las dos siguientes al verde y las dos últimas al azul.

Ejemplos:

- #FF0000 Rojo
- #00FF00 Verde
- #0000FF Azul
- #0000CC Azul un poco más oscuro.
- #FFFFFF Blanco
- #A7A7A7 Gris (Todas las combinaciones que asignan el mismo valor a los tres componentes producirán un color gris. Según ese valor, será más o menos oscuro).
- #000000 Negro
- #FFFF00 Amarillo (mezcla de rojo y verde).

Elegir una combinación de colores apropiada para nuestro sitio es una de las tareas más difíciles. Algunos expertos piensan que es una habilidad que algunas personas poseen y otras no.

Las matemáticas podrán ayudarnos a los que no la poseemos, porque existe software para ayudarnos en esa tarea:

- Creando combinaciones de colores a partir de uno de referencia.
- Obteniendo combinaciones de colores presentes en una imagen dada.
- Obtener el código del color que estamos viendo en pantalla.
- Buscar imágenes que satisfagan un patrón concreto.

Algunas de esas herramientas son:

- **ColorPix** (para Windows) o **Picksel** (para Linux): permite conocer los códigos, coordenadas y número de píxeles de cada color presente en la pantalla. Traduce entre distintas representaciones de colores. Otro ejemplo es [ColorCop](#).
- [Color Schemer Designer](#): aplicación web gratuita para crear combinaciones de colores a partir de un color de partida.
- [Whats Its Color](#): servicio web gratuito para encontrar colores complementarios para una imagen.
- **Gimp**: Herramienta “Recoge-color”, ...

Propiedades del **color**:

- **Tono (hue)**: propiedad que hace distintos los colores (rojo, amarillo, ...)

- **Saturación (*saturation*):** intensidad o pureza del color.
- **Luminosidad (*brightness*):** cantidad de luz que refleja un color.

La elección de una paleta de colores adecuada es muy importante en el diseño de una interfaz. También hay que tener en cuenta la percepción psicológica del color, ya que los colores influyen en el ánimo del usuario.

Hay recursos en la web con multitud de paletas prediseñadas.

Ejemplos: www.colourlovers.com/palettes, <https://color.adobe.com/es/create/color-wheel>, <https://paletadecolores.com.mx/>, <https://coolers.co/>, <https://www.colorhunt.co/>, <https://www.toptal.com/designers/colourcode>, ...

Tarea: Busca otras aplicaciones para:

- Crear combinaciones de colores a partir de uno de referencia.
- Obtener los colores en una imagen dada.
- Obtener el código del color que vemos en pantalla.
- Elegir entre diferentes paletas de color.

Coméntalas en clase para verlas entre los compañeros.

Psicología del color



4.2. Tipografía

En una web será casi imprescindible usar texto. Disponemos de infinidad de fuentes o tipos de letra. Una de las cosas que tenemos que tener en cuenta es que no todos los Sistemas Operativos disponen de las mismas fuentes, aunque actualmente se van acercando entre sí. Hay que asegurarse que nuestros textos tengan una apariencia similar en los distintos navegadores y sistemas operativos. Existe la norma de no usar más de 3 tipos de fuentes distintas en una web.

Aparte de esto, también tendremos que buscar adecuación entre la fuente usada y el mensaje, procurar legibilidad y, como con los colores, combinar con buen gusto.

A grandes rasgos se puede dividir en estas familias:

- **Serif:** Con pequeños adornos en los remates de las letras, y son más adecuadas para papel impreso, por su legibilidad y menor monotonía: Garamond, Courier, Times New Roman, ...
- **Sans-serif:** Sin ese remate. Las fuentes Sans Serif son adecuadas para mostrar textos en pantalla: Arial, Verdana y Helvética.
- **Script:** imitan en cierta forma la escritura manual: *Mistral*, *Palace Script*, ...
- **Decorativa:** muy informales



También se pueden separar en **monoespaciadas o no**, dependiendo de si cada carácter ocupa el mismo espacio.

- Luminosidad: Fuente Liberation sans.
- Luminosidad: Fuente Liberation mono.

Hoy en día hay multitud de repositorios de fuentes en Internet que pueden ser añadidos en nuestras páginas web, uno de los más conocidos es [Google Fonts](https://www.google.com/fonts).

Usando CSS podremos indicar tipos distintos para mostrar nuestro sitio en pantalla y en impresora.

4.3. Iconos

Un icono es una imagen pequeña que es metáfora de acciones que podemos realizar. Es necesario que exista una **relación entre el icono y lo que representa**, como el disco para "Guardar" o la lupa para "Buscar". Los iconos pueden acompañarse también de texto.

El objetivo es mostrar las opciones disponibles más rápidamente, pero esto sólo es posible si el

icono está bien elegido, porque si un usuario no lo entiende habría sido preferible usar texto.

Hay que tener en cuenta la estandarización. Crear un icono nuevo cuando ya existe uno realmente extendido es muy arriesgado.

No se recomienda usar iconos en las operaciones críticas y, en cualquier caso, conviene hacer varios diseños y testarlos con usuarios reales.

Al igual que en la tipografía, existen repositorios de iconos en internet: Iconos8.es, flaticon.es, etc. También existen herramientas que convierten imágenes en iconos (ficheros .ico) y software para crear tus propios iconos desde cero, por ejemplo con [X-Icon Editor](#).

Desde el punto de vista de los colores, fuentes e iconos, compare estos dos sitios web (aspectos positivos y negativos):

<http://www.lingscars.com/>

<http://www.avis.es>.

5. COMPONENTES DE UNA INTERFAZ WEB

Desde las primeras páginas web hasta la actualidad, los diseños han evolucionado haciéndose cada vez más homogéneos. Algunos de los elementos se han asentado al demostrarse su utilidad y comprensión por parte de los usuarios. Los encontraremos en casi todas las páginas y su diseño y funcionalidad será similar en todas ellas.

5.1. Cabecera

Zona superior de anchura generalmente igual al total de la página y altura variable. Suele alojar el logo y nombre de la organización a la que pertenece el sitio, aunque también son frecuentes fotografías y otros elementos como login o banners publicitarios.

El objetivo de la cabecera es:

- Identificar al emisor del mensaje.
- Homogeneizar todas las páginas del sitio.
- Crear una separación entre el borde superior y el contenido central de la página para facilitar la lectura.

El motivo por el que la cabecera está en la parte superior y el logo suele estar en la izquierda obedece a que en la cultura occidental leemos de izquierda a derecha y de arriba abajo.

Suele ser rectangular, aunque los nuevos diseños ofrecen muchas formas y colores. En cualquier caso, debe permitir que se vea con claridad el logo y nombre.

La cabecera no siempre ocupa todo el ancho e incluso es posible que una página no tenga cabecera (ejemplo: <https://www.zara.com/es/>).

5.2. Sistemas de navegación

Son los elementos de una interfaz que permiten moverse por las distintas páginas y secciones

que componen el sitio. Generalmente se presentan como menús de opciones con las que el usuario puede interaccionar.

Los menús pueden tener texto, iconos o ambos y usar efectos dinámicos que cambian el aspecto de las opciones a las que apunta el usuario. También es posible hacer aparecer y desaparecer porciones del menú.

Los hay de muchos tipos:

- Horizontales de pestañas que suelen ubicarse bajo la cabecera.
- Verticales en forma de lista o árbol generalmente en un lateral.
- Del tipo "estás aquí" que suelen situarse en la parte superior del área principal. Estos últimos presentan en forma de texto la ruta que se ha seguido para llegar a la página actual.

Conviene tener en cuenta:

- Si la altura de la página es tal que hay que desplazarse verticalmente para verla completamente es conveniente situar una versión reducida del menú principal en la parte inferior.
- Cuando hay demasiadas opciones es conveniente desdoblar en varios menús o bien jerarquizarlos y, en este caso, los menús secundarios deben identificarse claramente como tales.

La ubicación de los menús es muy importante: debe permitir un acceso fácil, pero sin estorbar al resto de elementos.

Analice los siguientes sitios web desde el punto de vista de la navegación:

<http://www.zincbistroaz.com>

<https://www.uma.es/>

5.3. Cuerpo

Es la parte donde se presenta la información de la página. Es el objetivo del sitio, lo que el usuario quiere ver. Por eso debe ocupar el mayor espacio, entre el 50% y el 85% del total, siempre en la ubicación central por debajo de la cabecera (si la hay) y al lado del menú de navegación (si lo hay).

La forma de los contenidos puede ser muy variada: texto, imágenes, tablas, formularios, etc. Pero hay elementos que siempre deberían estar presentes: un título resaltado bien por tamaño de letra o bien mediante colores distintos.

Los elementos que aparecen en el cuerpo de la página deben de estar en armonía con el resto de elementos de la interfaz.

5.4. Pie de página

Situado en la parte inferior de la página, bajo el cuerpo, no hay que menospreciar su utilidad.

Un uso común es mostrar enlaces a servicios concretos del sitio, como contratación de publicidad, contacto, ofertas de empleo, condiciones de uso, etc.

Tal y como comentamos en la sección de menús, si la página es muy alta, en el pie podríamos situar un menú que evite tener que volver arriba.

5.5. Espacios en blanco

Son, a pesar de lo que pueda parecer, un elemento importante en el diseño de una página. Permite compensar el peso visual del resto de elementos, creando márgenes entre ellos.

Es una buena recomendación tenerlos en cuenta desde el principio, tomándolos como un elemento más. Establecen el lugar que ocuparán los otros elementos y evitan que el diseño esté sobrecargado.

Si procuramos que los márgenes entre elementos y los márgenes laterales y superior tengan la misma anchura estaremos proporcionando simetría a la composición final.

Identifique en estos sitios web los elementos descritos en esta sección.

<http://stonelab.osu.edu>
<http://www.educa.jccm.es>
<http://www.gio.upm.es>
<http://www.zara.com>

Todas estas secciones son las típicas de una web clásica. Actualmente están siendo más conocidas las **Single Web Pages** en las que el contenido principal ocupa prácticamente toda la pantalla, minimizando el menú y reforzando el diseño para hacerlas más atractivas. Puedes ver algunos ejemplos [aquí](#).

6. LENGUAJES DE MARCAS. MAQUETACIÓN

Los lenguajes de marcas permiten crear documentos junto con indicaciones (marcas) que aportan información sobre la estructura del mismo. Así, es posible indicar que un trozo de texto es un párrafo, o un título o representa un concepto importante en el contexto del documento.

Ejemplos de lenguajes de marcas son XML, JSON o HTML.

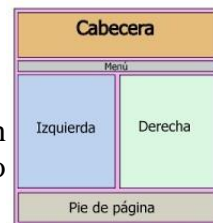
El lenguaje HTML (*Hyper Text Markup Language*, Lenguaje de marcado de hipertexto) es el que se usa como base para construir los contenidos de una página web. Tradicionalmente HTML aportaba también aspectos sobre la presentación de estos contenidos, pero la intención de la W3C (*World Wide Web Consortium*, organización que se encarga de regular el funcionamiento de Internet) ha ido en la línea de separar ambos aspectos. A partir de la versión HTML4 se produjo el dilema de si derivarlo hacia un lenguaje más estricto (XHTML, que es HTML adaptado a XML) o, como pasó al final, a uno compatible con versiones anteriores y más flexible. Así, nació la actual versión HTML5.

En la actualidad el desarrollo de una página web sería tener forzosamente que separar ambos aspectos, utilizando por un lado lenguajes de marcas para concretar los contenidos y su estructura y, por otro, estilos para indicar la presentación de esos contenidos (por ejemplo, con CSS). Esto permitirá que los cambios en un terreno no afecten al otro.

6.1. Maquetación web

La maquetación es la distribución en el espacio de los elementos de una página.

Hace unos años, cuando no estaba muy clara la separación contenidos/presentación, la maquetación se hacía mediante tablas HTML, creando problemas de restricciones de espacio y complejidad de código.



Actualmente la maquetación se hace mediante hojas de estilo (CSS), que serán tratadas ampliamente en este módulo.

En principio, conviene comenzar por hacer una distribución en grandes bloques de elementos de información y mantener cierta consistencia entre las distintas páginas de la web.

Con respecto a la posición de los mismos (**layout**), y dado los numerosos tamaños y resoluciones de pantallas actuales (ya sean ordenadores, móviles, pantallas, televisores, etc.), se suelen usar:

Elementos flotantes: normalmente, los elementos en una página web incluidos en la etiqueta `<body>` se sitúan de izquierda a derecha y de arriba abajo. Con las hojas de estilos, podemos situar un elemento en un lado que se indique y hacer que “flote”, es decir, que se eleve dentro de su contenedor. Así mismo, podemos posicionar otros elementos con respecto a éste.

Posicionamiento absoluto y relativo: Un elemento tendrá un posicionamiento absoluto cuando esté situado siempre en el mismo sitio, independientemente del resto de elementos (por ejemplo, arriba a la izquierda). En caso contrario (su posición depende de otros elementos) se dirá que tiene un posicionamiento relativo.

Dimensiones absolutas y relativas: Para indicar el tamaño de elementos de la web, se pueden usar ambos tipos de relaciones. Las absolutas no dependen de la pantalla o la resolución de la misma. Así una imagen de 5x5 cm (o puntos, o pulgadas, etc) se mostrará siempre de ese tamaño. Las relativas se indicarán en función de:

- Píxeles de pantalla (ojo: no todos los píxeles son igual de grandes): **px**
- La altura del carácter x en pantalla: **ex**
- La anchura del carácter m en la pantalla: **em**
- El porcentaje respecto al tamaño total de la pantalla: **%**

6.2. Distribución de elementos en la interfaz: capas, marcos

Al principio del HTML se solía estructurar el contenido de una página mediante tablas, creando así las zonas de una página web (encabezado, navegación, contenidos, pie, ...) Esto se conseguía anidando unas tablas dentro de otras asignándoles tamaños absolutos o relativos. Fácil, pero laborioso.

Las *capas*, también llamadas **DIV** o *layout*, son bloques con contenido HTML que pueden posicionarse de manera dinámica y anidarse. Básicamente, lo que se hace es definir cómo se

posiciona en la página web, su colocación y su tamaño.

Entre sus ventajas está que sólo pueden aprovecharse totalmente usando estilos CSS.

Los *marcos* dividen la pantalla en varias zonas y cargan una página en cada una de ellas. Es una idea similar a la de las capas (más anticuada y problemática).

Una alternativa a los marcos tradicionales son los marcos flotantes (IFRAME), que son como capas que pueden alojar una nueva página dentro de la nuestra y, al mismo tiempo, posicionarse de manera dinámica.

7. MAPA DE NAVEGACIÓN. PROTOTIPOS

Muestran todo el contenido de una web. Distinguimos dos tipos:



Mapas para usuarios: en páginas con mucha información y organizada de forma jerárquica, ayudan al usuario a encontrar información o situarse. Son como el índice de un libro. Hay quien lo considera desfasado, ya que, en teoría, en una web correctamente diseñada (buena usabilidad), no sería necesario.

Mapas para motores de búsqueda: es un fichero que contiene todas las direcciones y recursos de una web, así como meta-información (fecha creación, de actualización, etc.). Se generan generalmente en XML (*ej. sitemap.xml, robots.txt, ...*) o HTML y son útiles como herramienta SEO (de posicionamiento web), ya que son usadas por los buscadores. Existen diversos formatos e incluso herramientas que los generan, por ejemplo: <https://www.xml-sitemaps.com/>.

[Más info.](#)

Tarea: Busca otras aplicaciones para generar el sitemap de un sitio web y coméntalas en clase.

8. GUÍAS DE ESTILO

Una guía de estilo es un documento que define la pautas y normas de calidad que debe seguir una interfaz web para un determinado sitio. Garantiza la **coherencia del sitio y su homogeneidad**. Abarca aspectos de uso, accesibilidad, diseño gráfico, marketing, etc.

Una guía de estilo debería responder a preguntas como: ¿Qué colores tendrá la web y que tonos? ¿Qué fuentes se usarán? ¿Qué formato de fuente se usará para los títulos, subtítulos, encabezados y el texto principal? ¿Cuál será la estructura? ¿Habrá encabezado, pie de página o menús? ¿Habrá un menú o varios? ¿Cuántos y dónde colocarlos? ¿Qué imágenes se mostrarán? ¿Dónde se colocarán? ¿Habrá logotipo? ¿Dónde se colocará? ¿Se tratarán la accesibilidad de la página y criterios de calidad de uso? ¿Qué tamaños de imágenes se usarán en función de la pantalla en la que se visualice la web?...

Las guías de estilo no son sólo una herramienta para el diseño de un sitio web, sino que además son **imprescindibles para su mantenimiento o desarrollo**, sobre todo si pueden modificar la web varias personas. Con la guía podrán realizar estas variaciones conformes a los parámetros que marca, manteniendo la coherencia y estilo de la misma.

En su confección habrá que atender a diversos aspectos. Entre otros:

- Marca o empresa tras la página, que puede tener su guía de estilos.
- Tipografía e iconos.
- Tono del lenguaje: formal, coloquial, etc.
- Imágenes, sonidos o videos
- Controles de formulario: botones, cajas de texto, selectores, etc.
- Elementos del interfaz: listas, tablas, paneles...

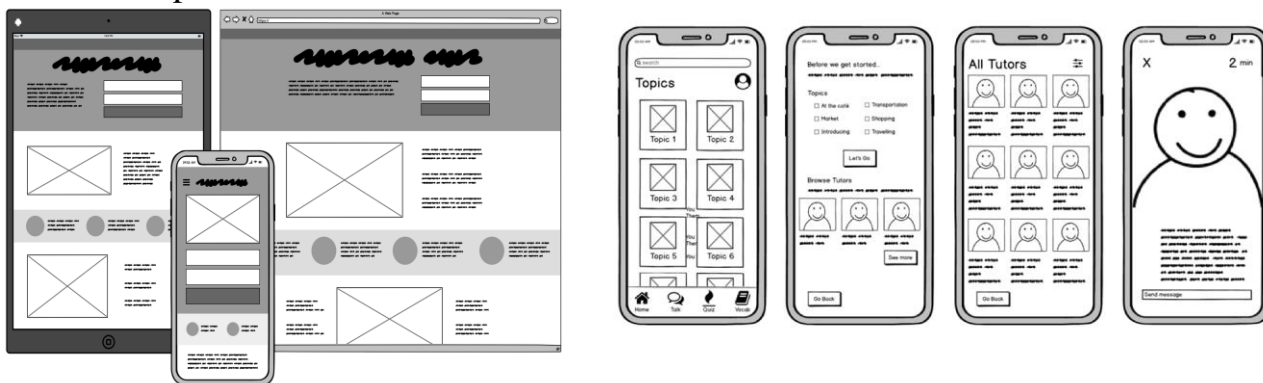
9. PROTOTIPADO WEB

Son métodos y herramientas para estructurar, organizar y etiquetar los componentes que conforman los entornos informacionales, y su objetivo es facilitar el acceso a la información contenida en ellos mejorando su utilidad por parte de sus usuarios, por lo tanto, esta disciplina sustenta, en gran parte, la potencial utilidad de los sitios web.

El resultado principal que se obtiene con el prototipado son una serie de diagramas o prototipos que se diseñan para cubrir dos funciones básicas: una representacional y otra comunicativa.

Tipos de prototipado

- **Blueprints:** también llamados sitemaps o mapas del sitio, ya los hemos visto anteriormente. Representan la estructura arquitectónica del sitio web con todas sus páginas, las relaciones existentes entre ellas y los contenidos que las componen.
- **Wireframes:** Son un esquema donde se representa el contenido y la arquitectura de la información de cada tipo de página, mostrando su aspecto desde la perspectiva de su arquitectura haciendo hincapié en cómo se agrupan y ordenan sus componentes.



Algunas aplicaciones para el desarrollo de los diagramas del prototipado:

- Microsoft Visio Desarrollo de diagramas de organización y funcionamiento.

- Con procesadores de textos (Writer, Word...) se pueden realizar diagramas de organización.

Online:

- www.ninjamock.com
- <https://www.fluidui.com/>
- <https://www.glify.com/>

10. HERRAMIENTAS PARA DESARROLLO WEB

Existen multitud de ellas, gratuitas o de pago, online o descargables, simples o más complejas. Realmente podríamos usar únicamente un editor de textos simple para crear una página web.

Podemos destacar:

- **Netbeans:** interfaz oficial de Oracle. Principalmente para desarrollar aplicaciones Java, pero se puede usar para diseños web (HTML, CSS, Javascripts).
- **Eclipse:** Uno de los más utilizados y base de otras aplicaciones derivadas. Gratuito y existen plugins muy útiles.
- **WebStorm:** Desarrollado por JetBrains, del estilo de los anteriores, pero de pago y con multitud de prestaciones. Otro IDE similar, muy utilizado y también de JetBrains, es PhpStorm.
- **Visual Studio Code:** gratuito y creado por Microsoft. Se pueden instalar multitud de extensiones genéricas o para lenguajes concretos.

También existen entornos ligeros y de menor tamaño que los anteriores, pero muy versátiles y con posibilidad de ampliar funcionalidades a base de plugins.

- **Sublime:** El favorito de muchos programadores. Es sencillo, de elegante diseño y con multitud de funcionalidades.
- **Atom:** De código abierto y con multitud de plugins.
- **Brackets:** Mini editor de Adobe ampliable por plugins.

Entornos en la nube, donde no hay que instalar software:

- **Ace** (<https://ace.c9.io/>).
- **Mónaco.** Mónaco es el editor que le da vida a VSCode y está soportado por todos los navegadores como Edge, Opera, Chrome, Safari y Firefox. Exceptuando, los navegadores móviles. Incluye validación de código, resaltado de sintaxis, intelisense, autocompletado y lo puedes probar aquí: <https://microsoft.github.io/monaco-editor/index.html>.
- **CodeMirror** (<https://codemirror.net/>). Es el editor de código en el navegador más usado. Una de las desventajas de Mónaco es que no es compatible con navegadores móviles, mientras que Codemirror no tiene ese problema. Además, Codemirror es completamente modular, así que se puede ir extendiendo y es hace que sea el editor más usado por empresas como Codepen, Mozilla, Replit, WordPress, entre otras.
- **CodeSandbox** (<https://codesandbox.io/>). Es un IDE en la nube orientado al desarrollo web. Puedes usarlo desde el navegador o desde iOS (iPad, iPhone). Está optimizado para tecnologías web como React, Vue, Angular, Svelte y más. Además, está integrado con npm para que puedas instalar paquetes públicos o gestionar paquetes privados.

- **Cloud9**: gratuito, con acceso a una máquina virtual en la que podemos gestionar el SO. Tiene proyectos basados en plantillas y permite descargarse el código fuente de repositorios.
- **Codenvy**: uno de los entornos más potentes.
- **HyperDev**: se centra en la agilidad de editar y ejecutar el código.

Por último, citar los entornos de pruebas, que son páginas web donde podemos editar y ejecutar código HTML, CSS y Javascript de forma directa en el navegador. Sirven para hacer pruebas rápidas o comprobar códigos hechos por otros. Destacan [JSFiddle](#), [Codepen](#), [JS Bin](#), [LiveWeave](#),

Por último, decir que los entornos de desarrollo en la nube (CDE) es una plataforma centralizada para que los equipos de desarrollo de software colaboren, construyan, prueben e implementen aplicaciones. Todos los procesos computacionalmente intensivos, como compilar código, ejecutar pruebas automatizadas o construir contenedores, ocurren en la nube. Los desarrolladores solo utilizan sus dispositivos locales como clientes ligeros para acceder al espacio de trabajo en la nube, reduciendo los costos generales de hardware.

	Desarrollo Local	Desarrollo Cloud
Ubicación del Entorno de Desarrollo	Los desarrolladores deben instalar manualmente componentes requeridos como editores y bases de datos en sus máquinas locales. Esto conduce a cadenas de herramientas complejas y fragmentadas en diferentes dispositivos.	Las plataformas en la nube manejan la configuración centralizada y el alojamiento del entorno de desarrollo, al cual los ingenieros acceden de forma remota a través de un navegador.
Consistencia del Entorno	Replicar entornos en las laptops de los desarrolladores es desafiante, lo que lleva a errores de “funciona en mi máquina”.	Los entornos estandarizados están versionados junto con el código en una infraestructura centralizada.
Distribución de Cómputo	Las compilaciones y pruebas se ejecutan localmente, ralentizando las máquinas.	La nube asigna potencia informática según la demanda, evitando cuellos de botella de productividad en los dispositivos de los desarrolladores.
Colaboración	El código permanece en ramas aisladas localmente hasta la revisión.	Los IDE en la nube permiten la edición en tiempo real por parte de múltiples usuarios, comentarios y visualización de código.
Integración y Acceso	Configurar nuevas máquinas locales es difícil y consume tiempo. Restringe el acceso de los desarrolladores.	Los desarrolladores pueden iniciar sesión instantáneamente en entornos predefinidos desde cualquier dispositivo.

Ejemplos de estos entornos están en Azure de Microsoft o AWS de Amazon, entre otros.

11. PLANTILLA DE DISEÑO

No siempre hay que iniciar una web desde cero, si hay un diseño que ya está hecho y funciona, podemos usarlo para ahorrar tiempo, así reutilizamos el código. Las plantillas de diseño son una buena opción para hacer un desarrollo profesional sin invertir demasiado tiempo.

Son sitios webs prediseñados, que disponen de una estructura definida y sólo necesitan incorporar los contenidos.

Los gestores de contenido (CMS, *Content Management Systems*) hacen uso de plantillas de diseño. De hecho, actualmente, existen empresas interesadas en diseñar plantillas (*templates*) para ser incluidas en estos entornos.

Sirvan como ejemplo (entre muchos otros):

<http://www.webself.net/>

<http://www.webnode.es/>

Tarea: Busca otras webs para obtener plantillas (tanto de pago como gratuitas) y coméntalas en clase.

12. INTERACCIÓN PERSONA-ORDENADOR

La Interacción Persona-Ordenador (IPO) es la disciplina que estudia el intercambio de información entre las personas y los ordenadores (el término en inglés es *HCI*, *Human Computer Interaction*). Su objetivo es que este intercambio sea más eficiente, minimizando errores e incrementando la satisfacción. La IPO ofrece un marco empírico con todo lo referente a calidad de uso e interacción de interfaces de usuario.

La IPO se basa en la psicología cognitiva, que estudia la percepción, la memoria, la adquisición de habilidades y el aprendizaje, la resolución de problemas, el movimiento, las tareas de juicio, de búsqueda o procesamiento de información y de la comunicación, es decir, procesos todos cuyo conocimiento se requiere para el adecuado diseño de mecanismos de interacción del usuario.

En el mundo de la IPO se relacionan diversas especialidades y profesionales para lograr la máxima eficacia. Por nombrar algunas: informática, psicología, diseño, ergonomía, ingeniería, sociología, semiótica (estudio de los signos) ...

Los principios fundamentales de la IPO se pueden resumir en:

- Conocer al usuario.
- Ofrecer información rápidamente (minimizando la memorización), facilitando la entrada de datos y manteniendo coherencia en las estructuras.

- Basándonos en la observación del uso, reorganizar la estructura de la información y optimizar operaciones.
- Evitar errores, y si ocurren, proporcionar información adecuada.

El concepto de IPO ha ido evolucionando con el tiempo hacia la **usabilidad**, que es hacer que la IPO sea lo más fácil de usar que sea posible. Para ello el diseño de la interfaz es fundamental.