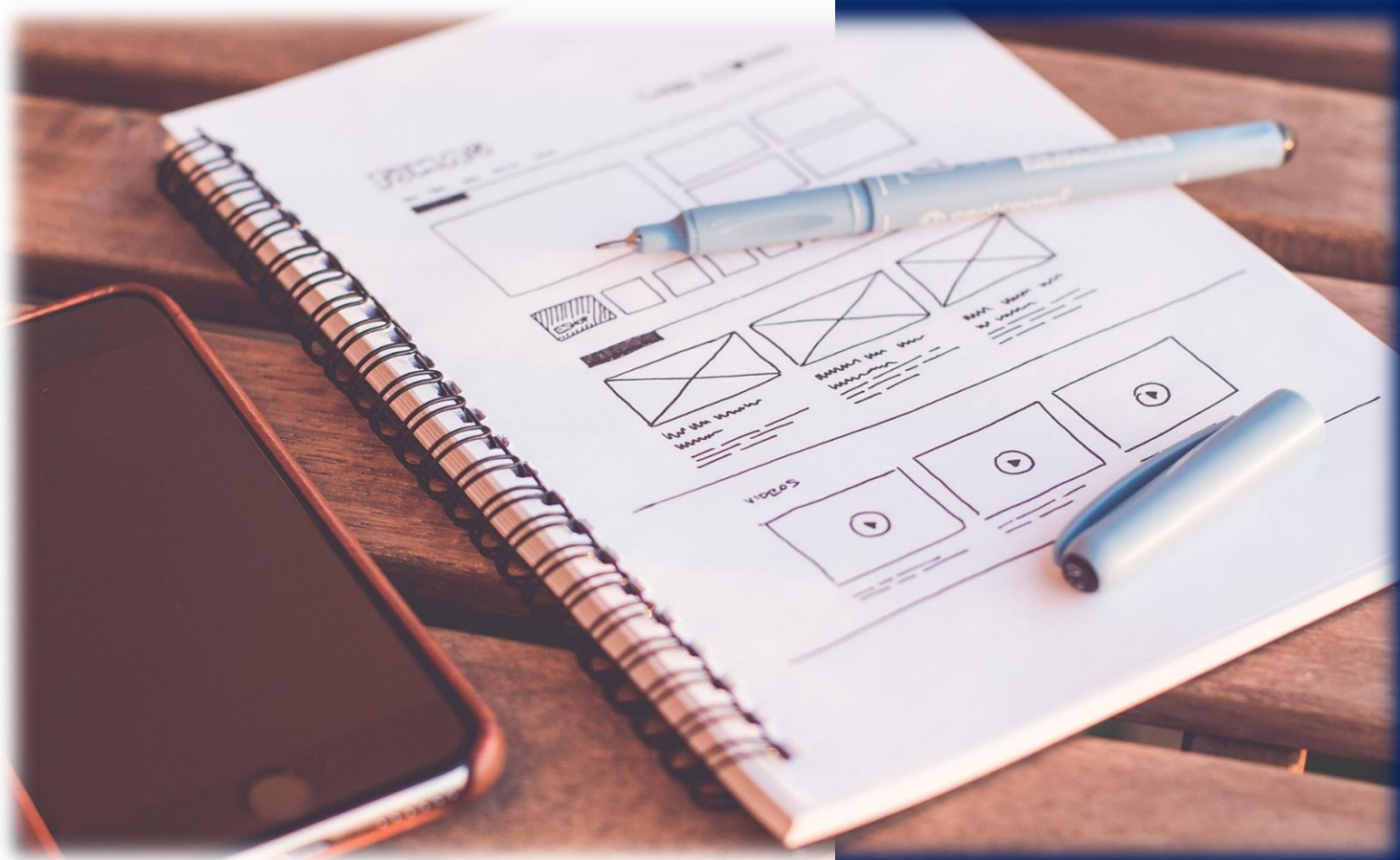


2023/24

U.D.1. PLANIFICACIÓN Y DISEÑO DE INTERFACES WEB.

1.2 Interfaces web y aplicaciones.





ÍNDICE

1. Interfaces Web	3
1.1 Estrategia y objetivos	5
1.2 Componentes o zonas de una interfaz web	6
1.3 Maquetación web	9
1.3.1 Elementos de ordenación	9
1.3.2 Dimensiones absolutas y relativas	22
1.4 Mapa de navegación	24
1.4.1 Mapas para el DISEÑO	24
1.4.2 Mapas para USUARIOS	25
1.4.3 Mapas para MOTORES DE BÚSQUEDA	26
1.4.4 Estructuras	27
1.5 Patrones de diseño web	30
1.5.1 Diseño web con PATRÓN EN Z	31
1.5.2 Diseño web con PATRÓN F	32
1.5.3 Diseño web con IMAGEN A TODA PANTALLA	34
1.5.4 Diseño web con PANTALLA DIVIDIDA	35
1.5.5 Diseño web ASIMÉTRICO	36
1.5.6 Diseño web de COLUMNA ÚNICA	36
1.5.7 Diseño web en BLOQUES	37
1.5.8 Diseño web en MOSAICO	38
1.5.9 Diseño web ESTILO REVISTA	39
1.5.10 Diseño web en FRANJAS HORIZONTALES	40
1.5.11 Plantillas de diseño	41
2. Aplicaciones	42
2.1 Prototipos	42
2.1.1 SKETCH o BOCETO	42
2.1.2 Wireframe (o Esquema de página)	43
2.1.3 Mockup o maqueta	45
2.1.4 Prototipo	46
2.2 Gestores de contenido o CMS	48



2.2.1	CMS más utilizados en la actualidad.....	49
2.3	Aplicaciones de desarrollo.....	55
2.3.1	Netbeans.....	55
2.3.2	Eclipse.....	56
2.3.3	Webstorm.....	57
2.3.4	Entornos ligeros	57
2.3.5	Entornos en la nube	59
2.3.6	Entornos de prueba.....	59



1. Interfaces Web

El diseño gráfico consiste en la programación, proyección y realización de comunicaciones visuales de aplicaciones, páginas o sitios web que, generalmente, serán transmitidos por medios industriales.

En la actualidad, esta área de desarrollo se lleva a cabo por unos profesionales que son denominados **diseñadores gráficos**. Entre sus tareas, destacan tres grandes funciones:

- 1) Función **estética**.
- 2) Función **publicitaria**.
- 3) Función **comunicativa**.

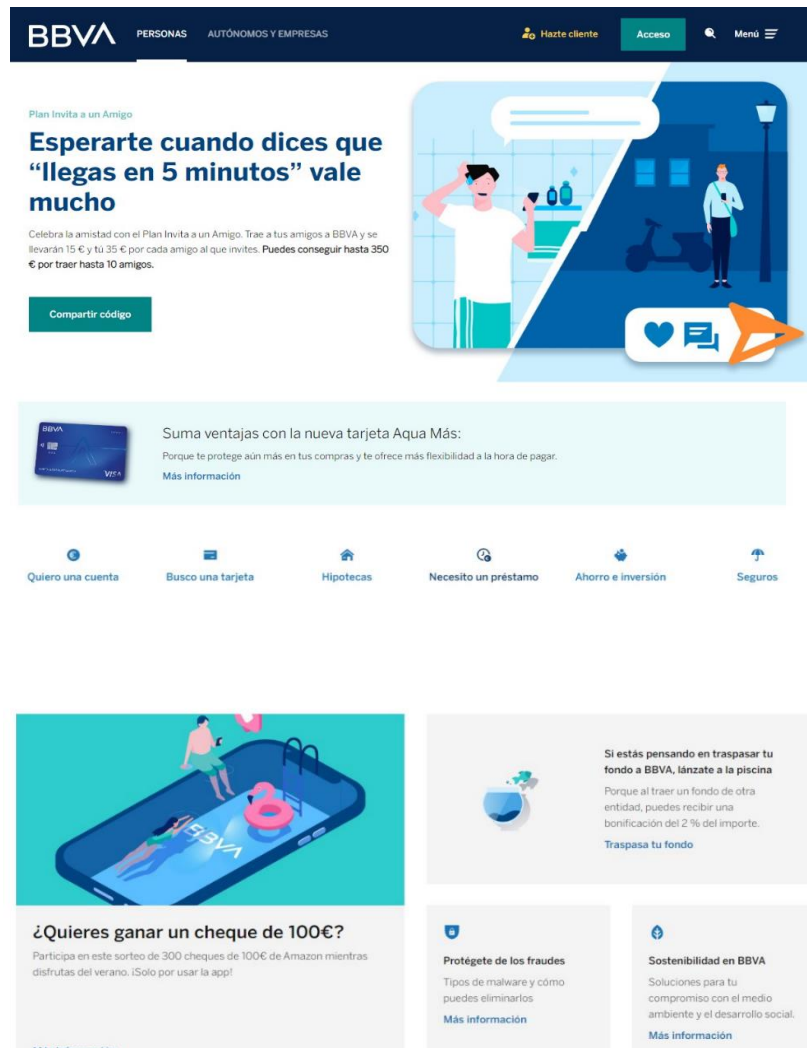
Además, en el diseño de interfaces se distinguen los diferentes elementos que ya hemos visto en apartados anteriores.

Por todo lo anterior, puede deducirse que una **interfaz web es el conjunto de elementos gráficos y el diseño de su distribución**, los cuales permiten una mejor presentación y una navegación más eficiente en el sitio web. Si ambos factores no existen y el resultado final del sitio es óptimo, será fruto de la casualidad.

En las siguientes imágenes pueden apreciarse dos diseños, uno bueno y otro no tanto.



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



Como se puede ver, aparecen dos páginas web diferentes; en la **imagen superior** puede observarse un **mal diseño web**; se trata de un sitio web en el que:

- ❖ Los elementos no aparecen claramente diferenciados.
- ❖ Utiliza una multitud de colores dispares sin seguir un claro diseño.
- ❖ No se puede leer bien el texto; no tiene buena legibilidad.
- ❖ No se adapta al tamaño de la ventana del navegador web (responsive web design), etc.

Mientras que, en el **segundo sitio web**, la navegación del usuario resulta mucho más **intuitiva**, lo que proporciona un **mayor grado de satisfacción**. Esto se debe, entre otros factores, a que:

- ❖ Los elementos aparecen claramente dispuestos y no hay sobrecarga de estos.
- ❖ Se utiliza una paleta de colores discreto y adecuada al contexto.
- ❖ Todo el texto es legible, etc.



1.1 Estrategia y objetivos

Cuando comenzamos a diseñar una interfaz web tenemos que tener en cuenta múltiples cuestiones como *¿a qué tipo de público irá dirigida la página?* *¿Queremos obtener un rendimiento económico con cada visita?* *¿Cuáles son los objetivos que se persiguen?* *¿Cuáles son los recursos temporales, personales y económicos de los que disponemos?* *¿Cuál será su contenido?*... los cuales nos orientarán sobre el enfoque que debemos darle al diseño de nuestra interfaz.

Si se comienza un proyecto web sin pensar en los usuarios a los que va a ir dirigido, es probable que no se tenga la acogida que se esperaba en un principio.

Como ya se ha visto en el apartado anterior, **una interfaz web es un sistema gráfico que permite acceder a los usuarios a los contenidos de la web mediante el uso de elementos gráficos**, los cuales son conocidos por la mayor parte de los usuarios que acceden a nuestra página.

El **objetivo principal** en el diseño de una interfaz web es que sus potenciales usuarios puedan **acceder a todos sus contenidos de la forma más rápida y sencilla posible**.

Para que un diseño web sea **efectivo**, debemos diseñar una interfaz que cubra todos nuestros objetivos, y al mismo tiempo, debe lograr que los usuarios de nuestro sitio puedan acceder con facilidad a sus contenidos, puedan interactuar con eficacia con todos sus componentes y, se sientan cómodos haciéndolo.

En cuanto a los **objetivos** de nuestra página web, debemos definir los siguientes:

- A. **Un objetivo principal** (o como mucho dos). Algunos **ejemplos** serían: aumentar el número de usuarios, aumentar el número de ventas, aumentar el número de visitas, reunir un volumen específico de emails en la base de datos para posteriormente enviar publicidad y conseguir clientes en potencia, etc.

El objetivo principal **debe ser ineludible**, esto es, el usuario tiene que encontrárselo, aunque no quiera, siempre evitando que sea de forma abusiva o agresiva.

- B. **Dos o tres objetivos secundarios** que deberán estar a pocos clics de ratón. Estos deberán ayudarnos a conseguir el objetivo principal.

En cuanto a la **experiencia de usuario**, en nuestra estrategia debemos tener en cuenta:

- ✓ La **paciencia de las personas no es ilimitada**. Cuando una persona busca una información en una página web, no permanecerá mucho tiempo en ella si no encuentra rápidamente lo que busca.
- ✓ El **gusto varía mucho de unas personas a otras**, pero no debemos olvidar que un diseño cuidadoso, una interfaz agradable y, un empleo coherente de los elementos gráficos, nunca nos hará perder visitantes.
- ✓ **Los enlaces que no funcionan** o que, sencillamente, no conducen a la información que prometían, provocan en el usuario una **sensación de rechazo**, con la consiguiente **pérdida de confianza** en nuestra página, pudiendo llegar, incluso, a la determinación de no visitarla de nuevo.

1.2 Componentes o zonas de una interfaz web

Son muchos los elementos de los que puede estar compuesta una interfaz web. El número de elementos empleados **dependerá del objetivo del sitio**. Así, las **zonas** más destacadas son las siguientes:

- ✓ **Zona de Identificación**. Está formada por los elementos de identificación, los cuales **identifican plenamente al sitio web**. El usuario, a la vista de estos elementos, debe saber a quién pertenece el sitio web. Algunos elementos que forman parte de esta zona pueden ser: el nombre de la web, el logotipo, la imagen de la cabecera, etc. A continuación, se presenta un **ejemplo**.



- 1) Título: Portada – Ministerio de Educación.
- 2) La imagen parcial de la bandera de la Unión Europea.
- 3) La imagen parcial de la bandera de España.
- 4) El escudo de España.
- 5) Nombre del organismo titular del sitio web.



✓ **Zona de Navegación.** Es aquella a través de la cual se **puede acceder a todos los contenidos** del sitio web. Algunos ejemplos de elementos de navegación son los siguientes:

- Menú principal.
- Widgets.
- Aside de navegación.
- Menú de navegación del pie de página o footer.

Esta navegación debe aportar al usuario la **sensación de saber dónde está, cómo puede moverse** por el sitio, y poder siempre **retornar de nuevo a la portada**. Para lograr este objetivo, el sitio debe disponer de, al menos:

- Elemento de **regreso a la página de inicio** o home.
- Ha de poder **consultarse el índice de contenidos** globales o menú principal.
- Debe indicarse en algún lugar **donde se está en cada momento**. Normalmente se lleva a cabo mediante una **línea de texto en la parte superior, bajo el menú** de navegación principal, que indica al usuario dónde se encuentra. Para este cometido también se pueden usar las “**migas de pan**” o **breadcrumbs**.

Las **migas de pan** o **breadcrumbs** es un elemento de navegación situado generalmente en la parte superior del contenido de una página que ayuda al usuario a ubicarla en la arquitectura de la información de la aplicación.



- ✓ **Zona de Contenidos.** Son las zonas en las que se muestra la **información relevante** de cada una de las páginas web que componen el sitio. Algunos ejemplos de elementos de contenidos pueden ser las áreas de texto de un artículo de un blog, el formulario de una página de contacto, etc.

Dentro de la zona de contenidos se debe distinguir la zona de **Título del Contenido** y la zona del **Contenido** propiamente dicho.



Es importante que los contenidos de una web sean interesantes para el lector y estén escritos de forma clara y concisa.

- ✓ **Zona de Interacción.** Son las zonas del sitio web en las que se **ofrece la realización de acciones** a los usuarios del sitio web como, por ejemplo, las cajas destinadas a que el usuario introduzca texto, el cambio de idioma del sitio web, diferentes botones, casillas de verificación, consultar el carrito de compra, suscripción en boletín informativo o newsletter, etc.





1.3 Maquetación web

La maquetación es un proceso que consiste en la **organización de los elementos** tales como texto e imágenes en un medio como una página, un cartel, etc.

Los tres grandes **puntos o preguntas que debe hacerse un diseñador de interfaces web**, para comenzar la maquetación web, y por tanto del sitio, son:

- 1) ¿Qué **elementos** va a tener el sitio web?
- 2) ¿Cuántas **páginas** va a contener el sitio?
- 3) ¿**Cómo se van a distribuir los elementos** en cada una de las páginas?

Por tanto, para diseñar un sitio web, debemos comenzar por hacer una distribución de los grandes bloques de elementos de información. De aquí sale el término de **Layout**, el cual es utilizado para referirnos a la disposición de los elementos en una página web.

1.3.1 Elementos de ordenación

Cuando diseñamos un sitio web, debemos ser **consistentes en la distribución** de los grandes bloques en todas las páginas internas del sitio, y **debemos tener en cuenta el espacio disponible** en la ventana del navegador.

Existen **elementos** comunes de toda página web, lo que no implica que sean obligatorios.

Estos son:

- La **cabecera**.
- El **menú o sistema de navegación**.
- El **cuerpo**.
- El **pie de página o footer**.

Por ejemplo, si no se quiere poner un pie de página, puede no hacerse, pero es conveniente.



Conocer el nombre de cada una de las partes de un sitio web es esencial ya que forma parte del vocabulario utilizado por los desarrolladores y diseñadores. La siguiente figura muestra la estructura general de un sitio web:



A. Cabecera, encabezado o header.

Está situado siempre en la **parte superior** de cualquier página. El objetivo de este elemento es la **identificación** del sitio web, por ello, suele contener elementos identificativos como el logotipo, el nombre de la empresa, elementos de acción que permiten cambiar el idioma de lectura, realizar búsquedas, e incluso, si el sitio es muy grande, puede contener elementos de navegación que permanecen a la vista en todas las páginas del sitio.

Este bloque **se repite en todas las páginas** de un sitio web y **debe permanecer visible** en todas ellas siempre que sea posible y la complejidad del sitio nos lo permita.

B. Sistema, barra o menú de navegación.

Este elemento muestra **el nombre de todas las páginas que componen el sitio web**, y por tanto es un elemento crucial para **ayudar a los usuarios a encontrar y acceder al contenido** que buscan. Un menú de navegación bien diseñado puede



mejorar la experiencia de usuario y aumentar la probabilidad de que los usuarios encuentren lo que buscan en la página.

Normalmente se sitúa justo **debajo de la cabecera**, pero es común encontrarlo también en la **parte izquierda** de las páginas. Al pinchar sobre cada uno de los nombres, se accede a la página del sitio deseada.

El diseño del menú de navegación debe ser **claro y fácil** de usar, con **etiquetas descriptivas** para cada enlace y una disposición lógica de las opciones. También es importante asegurarse de que el menú sea **fácilmente accesible** desde cualquier parte de la página y que se **adapte bien a diferentes tamaños de pantalla**.

C. Cuerpo o contenido.

Es la **parte central** de la página web, donde aparece la **información o contenido que se quiere ofrecer al usuario**, y la razón por la que éste visita nuestro sitio web, por lo que debemos prestar mucha atención al diseño de este bloque.

Es recomendable que todos los elementos que se sitúen en el cuerpo, mantengan un **diseño uniforme y similar** entre las diferentes páginas del sitio web.

Debemos reservar una **zona lo suficientemente grande** para que el usuario pueda leer los contenidos cómodamente, sin necesidad de realizar grandes desplazamientos. Es importantísimo **evitar** que el usuario tenga que realizar **desplazamientos horizontales** para leer el final de cada línea.

D. Pie de página o footer.

Está situado al **final de la página** y, al igual que el encabezado, **se repite en todas las páginas** del sitio web. Normalmente se emplea el pie como zona de navegación complementaria a la zona superior situada en el encabezado. En ella **se repiten algunos enlaces que se suelen colocar en el encabezado** como el enlace al mapa del sitio o el enlace a la información de contacto y, además, se colocan algunos enlaces nuevos como los enlaces a la información relativa a los **derechos de autor**,



redes sociales, privacidad e información legal, banners publicitarios, enlaces a servicios, formulario de contacto ...

El diseño del pie no suele estar tan elaborado como el del encabezado, ya **que su importancia es menor. El usuario tiene que ser consciente de que lo que está viendo es el pie.** Esto es de gran importancia en aquellos casos en los que, por ser el tamaño vertical de la página mayor que la ventana del navegador, el usuario se ve obligado a desplazarse verticalmente, pudiendo perder de vista el encabezado. Con un diseño más sencillo del pie respecto al resto de los bloques, conseguimos esa percepción por parte del usuario.

E. Barra lateral o sidebar

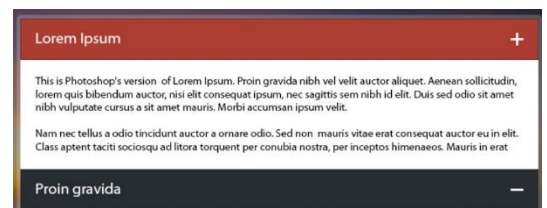
La **barra lateral o sidebar** de una página web es un área ubicada **a un lado del contenido principal**, generalmente a la derecha o a la izquierda, que puede contener **información adicional**, enlaces a otras páginas o secciones, o widgets como un calendario o una lista de etiquetas.

Puede ser utilizada para mostrar **contenido relacionado con el contenido principal** de la página, como enlaces a artículos relacionados en un blog, o para mostrar información adicional que pueda ser útil para el usuario, como una lista de categorías o etiquetas en una tienda en línea.

F. Otros componentes

Además de los componentes principales que hemos estudiado, hay una serie de elementos que suelen ser muy habituales en las páginas web. Aunque no vamos entrar en la explicación de cada uno de ellos, conviene conocerlos ya que forman parte del vocabulario técnico que todo diseñador de interfaces web debe conocer:

- **Acordeón (Accordion).** Un acordeón es un elemento de interfaz que permite mostrar y ocultar contenido en secciones expandibles y colapsables.

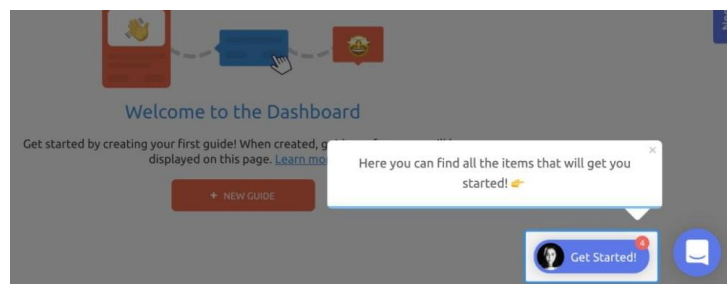




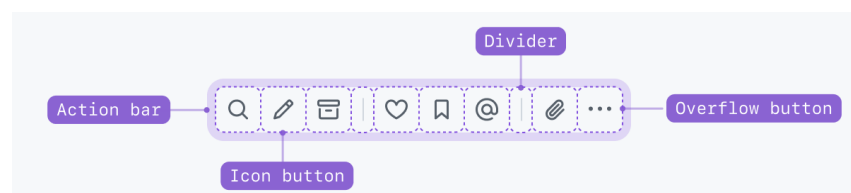
- **Anuncio (Banner, Ad or Advertisement).** Un anuncio, *banner* o *ad* es un elemento de publicidad que se muestra en una página web para promocionar un producto o servicio.



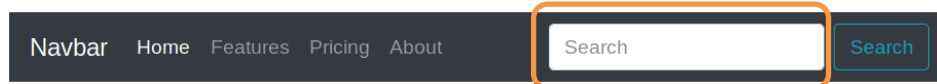
- **Ayudas Flotantes (Tooltip).** Las ayudas flotantes o *tooltips* son pequeñas cajas de texto que aparecen al pasar el cursor sobre un elemento para proporcionar información adicional.



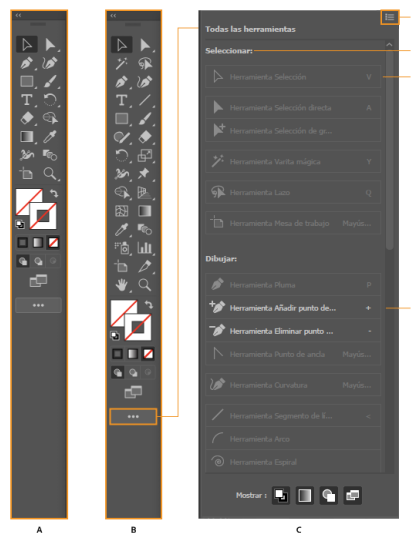
- **Barra de Acciones (Action Bar).** Una barra de acciones o *action bar* es una barra de herramientas que proporciona acceso rápido a acciones comunes en una aplicación o página web.



- **Barra de Búsqueda (Search Bar).** Una barra de búsqueda o *search bar* es un campo de entrada que permite a los usuarios buscar contenido en una página web.



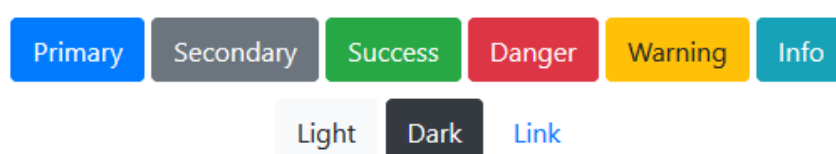
- **Barras de herramientas, cuadrículas y listas (Toolbars, grids, lists).** Son elementos utilizados para organizar y presentar información en una página web o aplicación.



- **Barra de Progreso (Progress Bar).** Una barra de progreso o *progress bar* es un elemento visual que muestra el progreso de una tarea en curso, como la carga de una página o la descarga de un archivo.

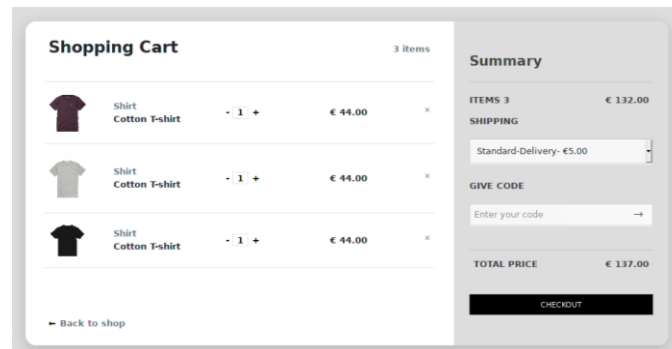


- **Botones (Buttons).** Los botones son elementos interactivos que permiten a los usuarios realizar acciones, como enviar un formulario o navegar a otra página.

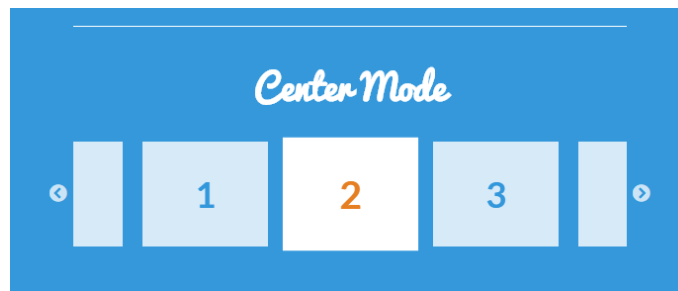




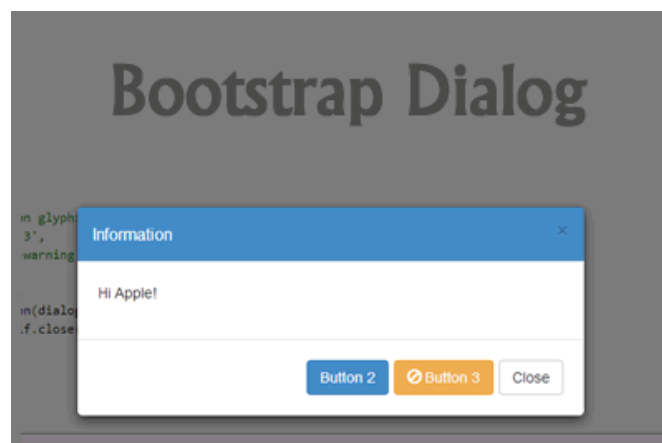
- **Carrito de compra o cesta (Shopping cart or Basket).** Un carrito de compra o cesta es un elemento en una tienda en línea que permite a los usuarios guardar productos para comprarlos más tarde.



- **Carrusel (Carousel).** Un carrusel o *carousel* es un elemento que permite mostrar múltiples elementos en un espacio limitado, desplazándolos horizontalmente.

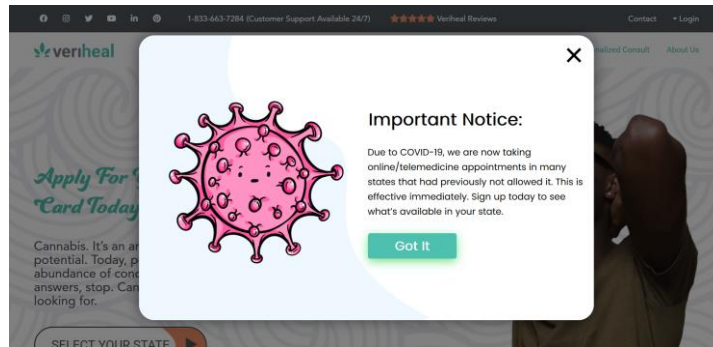


- **Cuadro de diálogo o modal (Modal or Dialog box).** Es una ventana emergente que permite al usuario interactuar con la aplicación y seleccionar opciones, **bloqueando todas las funciones para concentrar el foco en una acción particular. Aparece sobre el contenido principal.**

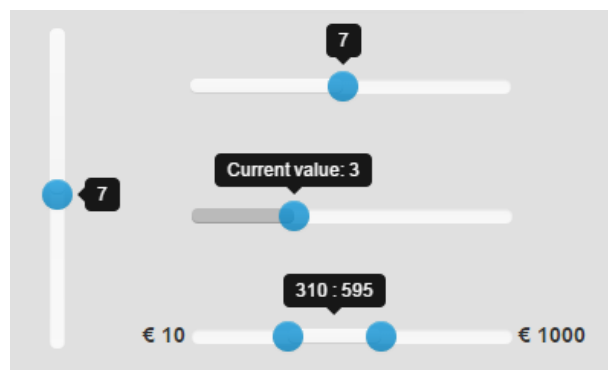




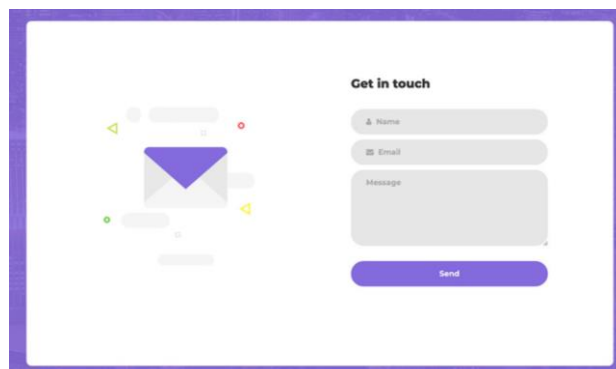
- **Ventana emergente (Popup) o popover (Popover).** Es una ventana que aparece en la pantalla sin haber sido solicitada por el usuario, de forma automática, normalmente con el fin de mostrar publicidad o información adicional.



- **Deslizador (Slider).** Un deslizador o *slider* es un elemento que permite a los usuarios ajustar un valor desplazando un control deslizante a lo largo de una barra.

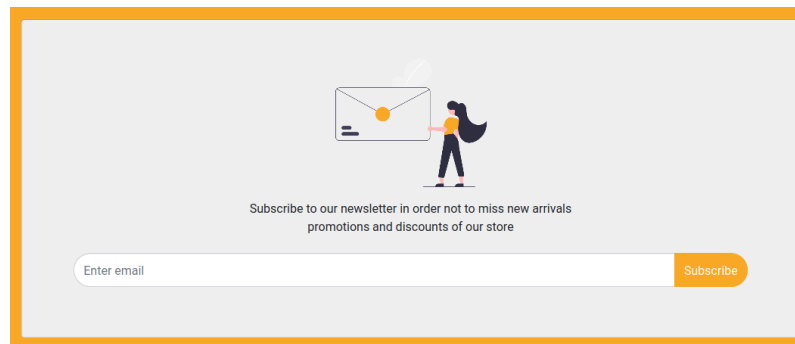


- **Fichas, campos y secciones (Tabs, fields, sections).** Son elementos utilizados para organizar el contenido en una página web o aplicación.
- **Formulario de contacto (Contact Form).** Un formulario de contacto o contact form es un formulario que permite a los usuarios enviar un mensaje al propietario de una página web.





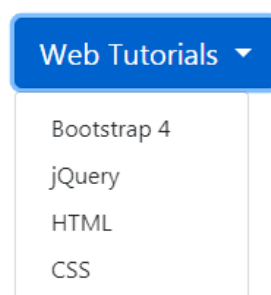
- **Inscripción al boletín de noticias (Newsletter Subscription).** La inscripción al boletín de noticias o *newsletter* es un formulario que permite a los usuarios suscribirse para recibir noticias y actualizaciones por correo electrónico.



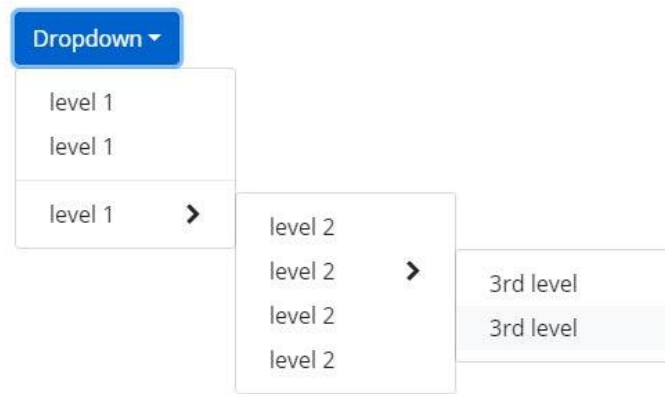
- **Llamada a la acción (Call to action).** Es un botón o enlace situado en un sitio web que busca atraer clientes potenciales y convertirlos en clientes finales, normalmente a través de un formulario en una página de destino o landing page.



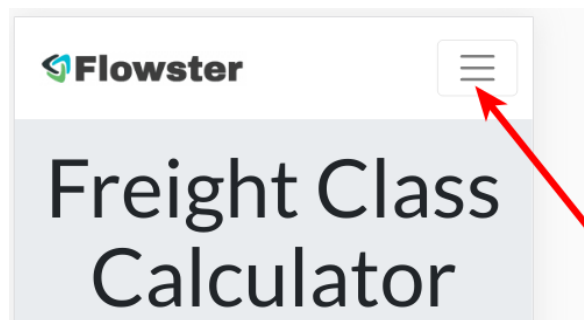
- **Menú desplegable (Drop-down menu).** Es un elemento de la interfaz gráfica de usuario que permite al usuario seleccionar una opción de una lista de opciones.



- **Menú desplegable multinivel (Multilevel drop-down menu).** Un menú desplegable multilevel o multinivel es un tipo de menú desplegable que contiene varios niveles de opciones. Cuando el usuario interactúa con una opción del menú, se despliega una lista de subopciones relacionadas con esa opción.



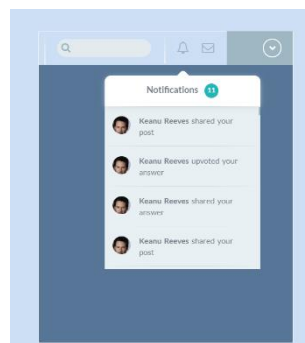
- **Menú hamburguesa** (*Hamburger menu*) o **toggle** (*Toggle*). Es un icono que se utiliza para representar un menú en dispositivos móviles. Cuando se hace clic en él, se despliega un menú con opciones.



- **Migas de pan** (*Breadcrumbs*). Son una serie de enlaces que muestran la ruta que ha seguido el usuario para llegar a una página en particular dentro de un sitio web.

[Electronics Home](#) / [Smart Phones](#) / [Brand Name Product Page](#)

- **Notificaciones** (*Notifications*). Son mensajes que informan al usuario sobre eventos importantes dentro de una aplicación.



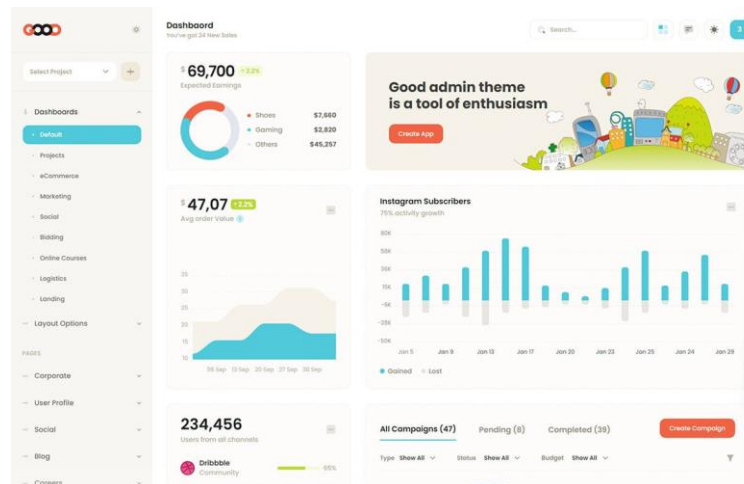


- **Paginado (Pagination).** Es el proceso de dividir el contenido en varias páginas para facilitar la navegación y mejorar la experiencia del usuario.

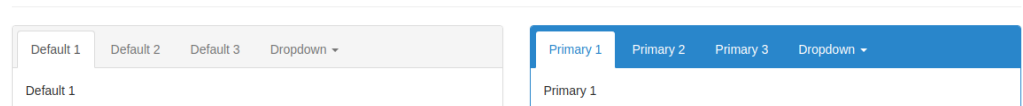
WebDesignerDepot Pagination

PREV 1 2 3 4 5 6 NEXT

- **Panel de acciones (Action panel).** Es un panel que contiene botones u opciones para realizar acciones rápidas dentro de una aplicación.



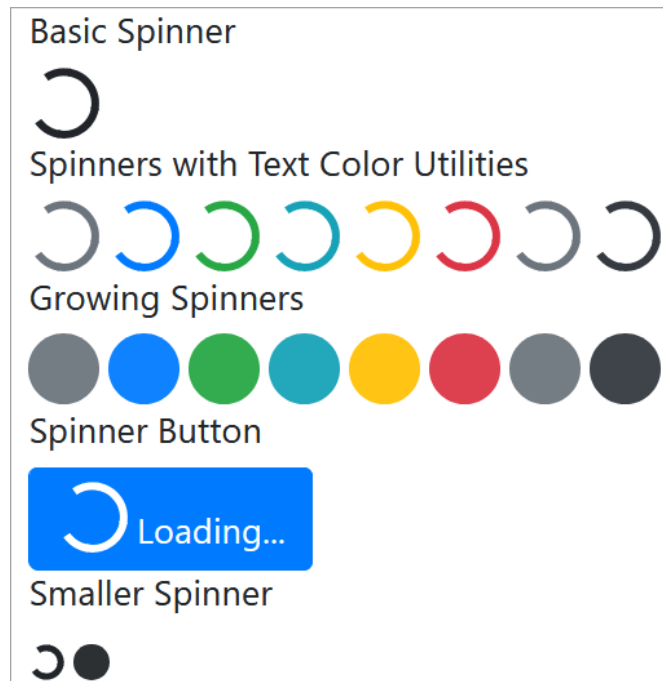
- **Pestañas (Tabs).** Son elementos de la interfaz gráfica de usuario que permiten al usuario cambiar entre diferentes vistas o secciones dentro de una misma página.



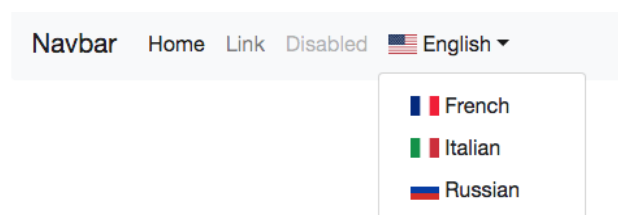
- **Políticas de privacidad y cookies (Privacy and cookie policies).** Son documentos legales que informan al usuario sobre cómo se recopilan, utilizan y protegen sus datos personales y sobre el uso de cookies en un sitio web.



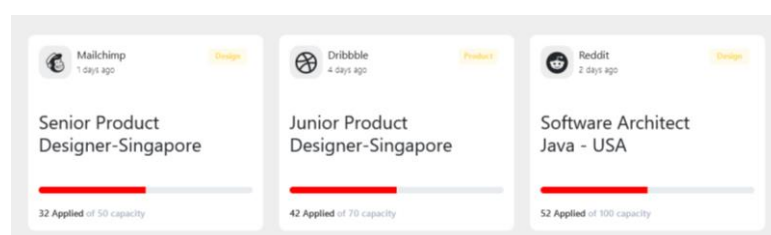
- **Redes sociales** (*Social media*). Son plataformas en línea que permiten a los usuarios interactuar entre sí y compartir contenido.
- **Rosco de carga** (*Spinner*). Es un elemento gráfico que indica al usuario que una acción está en proceso y que debe esperar.



- **Selector de idiomas** (*Language selector*). Es un elemento de la interfaz gráfica de usuario que permite al usuario seleccionar el idioma en el que desea ver el contenido de un sitio web.



- **Tarjetas** (*Cards*). Son elementos gráficos utilizados para presentar información de manera visual y organizada.



A continuación, se muestra un ejemplo más complejo de una página web y sus elementos:





1.3.2 Dimensiones absolutas y relativas

A la hora de indicar el tamaño de los elementos web existen varios tipos de dimensiones, las cuales también se pueden dividir en dos grupos: absolutos y relativos. Si observamos las unidades que se aplican en las páginas web, nos daremos cuenta de que no existe un criterio único.

A. UNIDADES ABSOLUTAS. Este tipo de unidades se aplican siempre igual al margen de cómo sea la pantalla o la resolución. Así, si establecemos que un elemento tiene 10cm de ancho, mantendrá ese aspecto en todos los dispositivos.

Dimensión	Significado	Descripción
pt	puntos	Medida utilizada para fuentes. Por norma general un punto equivale a 1/72 pulgadas.
pc	picas	Otra medida tradicional equivalente a 12 puntos o 1/6 pulgadas.
mm	milímetros	Medida métrica.
cm	centímetros	Centímetros.
in	pulgadas	Pulgadas.

B. UNIDADES RELATIVAS. Los elementos con estos tipos de unidades se ajustan a cada tipo de dispositivo. Por ejemplo, si indicamos que un elemento tiene tamaño 10em este sería distinto en cada tipo de dispositivo.

Dimensión	Significado	Descripción
px	Píxeles	Un píxel es cada uno de los puntos de la pantalla que forma la imagen. La cantidad de píxeles que pueden ser representados varía según la resolución de pantalla.
ex	Altura de x	Representa la altura del carácter x en la pantalla. Es relativo al tamaño base del navegador (16 píxeles).
em	Anchura de M	Representa el ancho del carácter M en la pantalla y suele ser el doble que la dimensión ex.
%	Porcentaje	Representa el porcentaje respecto al tamaño total de la pantalla.

En principio las unidades y dimensiones de los elementos es una configuración más propia de la presentación y debería manejarse en las hojas de estilos. No existe un tipo de dimensión mejor que otra, pero sí una serie de reglas que hay que tener en cuenta según el tipo de contenidos y medios que tengamos entre manos. En la mayoría de las situaciones y, a fin de ajustarse a la mayor parte de escenarios, **en general son mejores las unidades relativas.**



Si lo que vamos a desarrollar son principalmente páginas web o aplicaciones para móviles, las dimensiones absolutas no cobrarían mucho sentido. Si por ejemplo establecemos que el ancho de una página sea de 15 centímetros, aunque consigamos que se vea igual en todas partes, el efecto será contraproducente. Para un móvil será excesivo y para una pantalla ancha será ridículo. Por lo tanto, en general se deben evitar las unidades métricas del mundo real.

El dimensionamiento en píxeles y ems nos permite un mejor control y adaptabilidad a cualquier dispositivo.

Los ems siempre se han considerado como la mejor opción para el dimensionamiento en páginas web. El tamaño de los ems se establece según el tamaño base que tenga el navegador y eso es lo que precisamente le otorga adaptabilidad a cualquier pantalla.

Los **inconvenientes** de los píxeles y ems son:

- ⇒ El problema que puede tener el **píxel** es que no todas las pantallas son iguales y que un píxel no es igual en todas las pantallas. Esto se soluciona siempre que se pueda redimensionar el texto en las páginas.
- ⇒ El problema que se puede presentar en el caso de los **ems**, es que haya usuarios que hayan cambiado ese tamaño base en su navegador. También el hecho de usar decimales o valores entre 0 y 1 en ems ha ocasionado problemas en algunos navegadores.

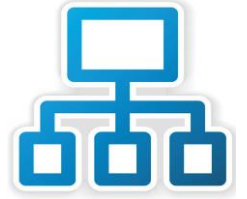
No hay ninguna regla que nos obligue a utilizar una única unidad de medida para los elementos de una página web. Por ejemplo, las anchuras de los elementos pueden ponerse en porcentajes y las dimensiones de las fuentes en em.

En cualquier caso, el diseño de interfaces y el correcto dimensionamiento de los componentes para que estos se adapten de forma óptima a cualquier dispositivo, no es una tarea fácil. Por suerte, existen plantillas y frameworks que nos facilitan enormemente ese trabajo.



1.4 Mapa de navegación

Un mapa de navegación o *sitemap* nos da **información sobre las páginas, videos y otros ficheros que contiene nuestro sitio web, así como las relaciones** que hay entre todos ellos. Por tanto, un mapa de navegación muestra todo el contenido de una web.



Debemos distinguir **tres tipos de mapas de navegación**, los que se crean para el diseño de la página web, para los usuarios finales, y para los motores de búsqueda.

1.4.1 Mapas para el DISEÑO.

Durante el proceso de maquetación del sitio web, es aconsejable crear un esquema en el que se muestren todas las secciones o páginas y la relación que existe entre cada una de ellas, es decir, **crear un mapa de navegación que facilitará el diseño óptimo** del mismo.

Los mapas de navegación proporcionan una **representación esquemática de la estructura** del sitio web, **mostrando sus secciones y páginas, y las relaciones** entre ellas. Se crea una especie de índice de contenidos a través del cual los usuarios pueden localizar más rápidamente lo que buscan.

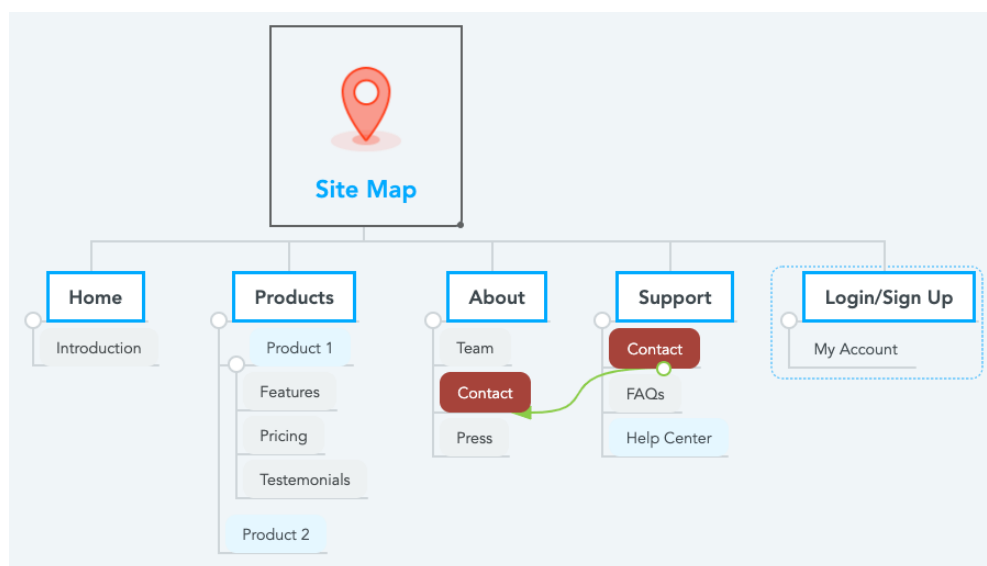
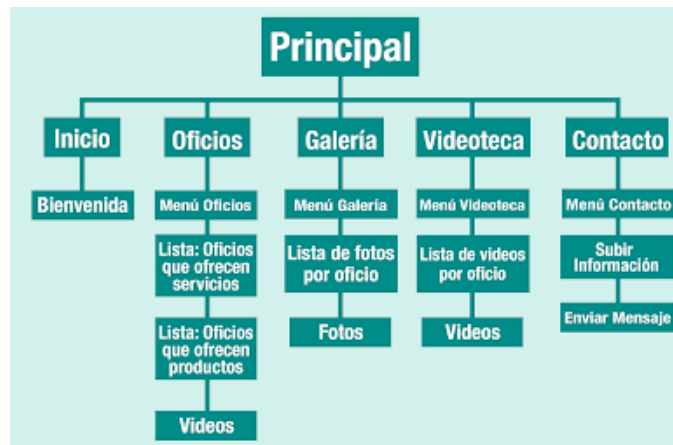
Al representar un mapa de navegación, este adquiere un aspecto similar al de un mapa conceptual. Ambos representan elementos comunes, como el nodo raíz, esto es, la página inicial en los mapas de navegación. A partir de este nodo se segrega el resto de los elementos del sitio. Además, es posible que los elementos “hijos” estén relacionados entre sí, y que sea posible pasar de una página web a otra del mismo sitio web sin necesidad de regresar al inicio.

Los **errores más comunes** a la hora de hacer un mapa de navegación son los siguientes:

- 1) Hacer una estructura de navegación **demasiado amplia y poco profunda**. Por ejemplo, cuando se ofrecen ocho opciones principales, de las cuales dos de ellas dan la posibilidad de seguir navegando.
- 2) Hacer una estructura de navegación **demasiado profunda y poco amplia**. Por ejemplo, cuando el usuario se enfrenta sólo a tres opciones de inicio, y en cada una de ellas debe hacer demasiados clics para acceder a contenidos de su interés.



Por tanto, en la creación de un mapa de navegación, **se debe buscar el equilibrio**, de manera que, para intentar facilitar la navegación al usuario, se ofrecerán pocas opciones en el menú principal, y luego, dentro de cada una de ellas, basta con uno o dos clics para acceder a contenidos específicos.



Para la realización de estos mapas, se pueden utilizar **herramientas** online tales como **draw.io**, **xmind.net**, **smartdraw.com**, **createlly.com**, **gitmind.com**, **mindmeister.com**, **lucidchart.com**, etc.

1.4.2 Mapas para USUARIOS.

En algunas páginas web y, en especial, en aquellas que tienen una información extensa y organizada de forma jerárquica, puede ser útil disponer de un mapa que permita a un usuario **saber dónde se encuentra o para facilitar la búsqueda de determinada información**.



Por **ejemplo**, en una tienda virtual se pueden organizar las páginas por productos, categorías, subcategorías, etc.

Este tipo de mapas ya **no es tan frecuente** encontrárselo en las webs actuales. En teoría, si una web está correctamente diseñada y su información sigue una estructura coherente, este tipo de recurso no es necesario. De hecho, hay quien considera que están desfasados.

A continuación, se muestran algunos ejemplos:

Comunidad Autónoma de Galicia.	https://www.xunta.gal/mapa-do-portal
Comunidad Autónoma de Euskadi.	https://www.euskadi.eus/gobierno-vasco/contenidos/informacion/mapa_web_euskadi/es_6313/sitio_web_euskadi.html
Comunidad Autónoma de Cataluña.	https://web.gencat.cat/es/ajuda/mapaweb/

1.4.3 Mapas para MOTORES DE BÚSQUEDA.

Un mapa de navegación de este tipo consiste en un fichero que contiene todas las **direcciones y recursos** que forman parte de una web **de forma exhaustiva**. Además de los enlaces a cada página, también contienen meta-información como la fecha de creación o actualización, la importancia de ese recurso en la web, etc.

Los motores de búsqueda como Google **leen este archivo para rastrear el sitio web de manera más eficiente**. Así, un mapa de navegación le dice al motor de búsqueda qué páginas y archivos cree que son importantes en su sitio, y también proporciona información valiosa sobre estos archivos. Por **ejemplo**, cuándo se actualizó la página por última vez y cualquier versión de idioma alternativo de la página.

Estos mapas se generan en **formato html o xml** y son útiles como herramienta **SEO** (*Search Engine Optimization*), porque son utilizados por los motores de búsqueda como Google para llegar a todos los rincones de una web. **El propio Google recomienda** estos mapas para mejorar el acceso a un sitio web, sobre todo en los siguientes **casos**:

- La **web** es **grande** y contiene infinidad de recursos que pueden ser pasados por alto.



- Cuando tenemos una **web de reciente creación** que todavía no está referenciada por otros.
- Cuando tenemos **recursos que no están referenciados** o no tienen ningún link para llegar a ellos.
- Cuando hay **recursos multimedia** que pueden ser trasladados a Google News.

A continuación, se muestra un **ejemplo** de un sitemap en xml, que es el formato soportado por Google. Lo podrás encontrar entero en <https://www.udemy.com/sitemap.xml>

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8" ?>
<sitemapindex xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <sitemap>
    <loc>https://www.udemy.com/sitemap/navigation.xml</loc>
  </sitemap>
  <sitemap>
    <loc>https://www.udemy.com/sitemap/courses.xml</loc>
  </sitemap>
  <sitemap>
    <loc>https://www.udemy.com/sitemap/courses.xml?p=2</loc>
  </sitemap>
  <sitemap>
    <loc>https://www.udemy.com/sitemap/courses.xml?p=3</loc>
  </sitemap>
  <sitemap>
    <loc>https://www.udemy.com/sitemap/courses.xml?p=4</loc>
  </sitemap>
  <sitemap>
    <loc>https://www.udemy.com/sitemap/courses.xml?p=5</loc>
  </sitemap>
  <sitemap>
    <loc>https://www.udemy.com/sitemap/courses.xml?p=6</loc>
  </sitemap>
</sitemapindex>
```

También se pueden encontrar más detalles sobre los sitemaps en la siguiente página web:
https://developers.google.com/search/docs/advanced/sitemaps/overview?hl=en&visit_id=637963224067752990-4059226983&rd=1

1.4.4 Estructuras.

Ahora que ya conocemos algunos mapas de navegación, vamos a ver cuáles son las **estructuras más habituales**.

El mapa de un sitio web va a tener una estructura que dependerá de la relación que tengan las páginas del sitio entre sí. Esta relación puede ser de diferentes tipos: lineal, lineal en estrella, jerárquica, lineal jerárquica, compuesta, o reticular o múltiple.

- ✓ **Lineal**. Es adecuada en aquellos sitios donde se quiere llevar un **proceso paso a paso** o donde la lectura de la página es **secuencial**. Su estructura es similar a la de



un libro donde avanzas de página en página, pero puedes volver a la página anterior y desde ésta a la anterior para releer algún párrafo.

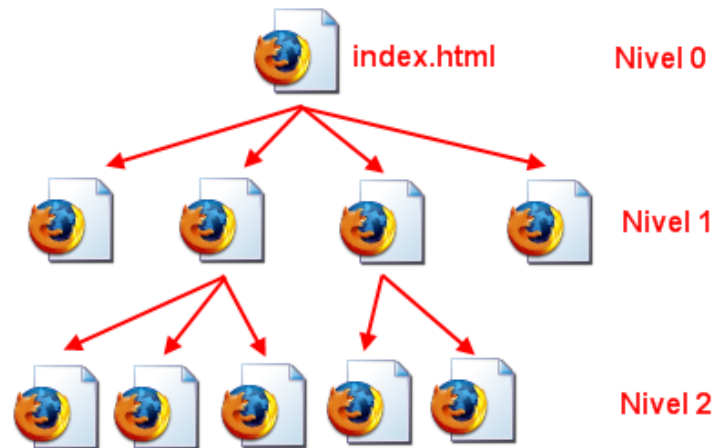


- ✓ **Lineal en estrella.** Es igual que la anterior, solo que la navegación **va y vuelve hacia el inicio**.



- ✓ **Jerárquica.** Es una de las **más comunes**. Se emplea en aquellos sitios donde existen varias secciones bien diferenciadas, pero de poca complejidad, de modo que el usuario no tiene por qué navegar de una sección a otra.





- ✓ **Lineal jerárquica.** Es también de las **más empleadas** cuando cada una de las secciones tiene un **volumen de información más elevado** y conlleva una lectura secuencial del contenido de la sección. También se emplea este método en sitios en los que sus secciones representan grados de dificultad de la información presentada y permite la **navegación entre secciones**.

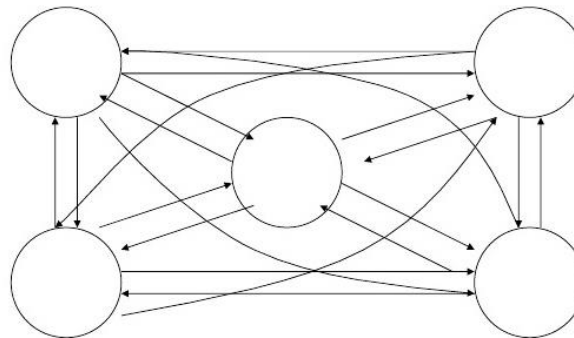


- ✓ **Compuesta.** Combina distintos tipos de navegación.





- ✓ **Reticular o múltiple.** Se emplea en aquellos sitios web en los que todas sus páginas están relacionadas entre sí. No resulta adecuado cuando el sitio está compuesto por muchas páginas porque **el usuario puede llegar a perderse**.



1.5 Patrones de diseño web

En general, un **patrón de diseño** es una solución a un problema concreto que se puede usar repetidamente en problemas similares haciendo pequeñas variaciones.

Existen infinidad de patrones. De entre ellos, podemos distinguir dos tipos:

- Patrones de diseño de software, orientados a la funcionalidad.
- Patrones de diseño de interacción, orientados a la usabilidad, y en los que se centran los patrones de diseño web.

Un **patrón de diseño web** es una forma determinada de **distribuir los elementos visuales** de una página web y optimizar la relación entre ellos. Aunque los elementos se pueden distribuir como uno quiera, seguir un patrón de diseño web **ayuda a transmitir mejor el mensaje** de la página web y a **mejorar su usabilidad**.

Elegir un patrón de diseño web es una de las primeras decisiones que debes tomar al crear una página web, y acertar con el adecuado es crucial.

El patrón del diseño es esencial en el diseño web, ya que establece un orden jerárquico visual de los elementos, define aquellos que deben captar mayor la atención del usuario, y contribuye al equilibrio general del diseño de la página web.

A la hora de elegir el patrón de diseño de una página web, existen dos **factores** que hay que tener en cuenta:



- **Ajustar el contenido de la página web.** El diseño web de una página debe apoyar el mensaje que deseas transmitir. Algunos patrones son mejores para mostrar productos, mientras que otros son preferibles para transmitir información rápida y eficazmente (un blog o un portal de noticias).
- **Usar patrones estándar.** Aunque admiten cierta creatividad, estos patrones de diseño web son generalmente la mejor opción porque han demostrado una y mil veces que funcionan. Como son muy comunes, cumplen las expectativas del usuario que ya está acostumbrado a ellos: **son más intuitivos y fáciles de usar.**

A continuación, se exponen diez patrones de diseño web, los cuales son más efectivos y extendidos hoy en día.

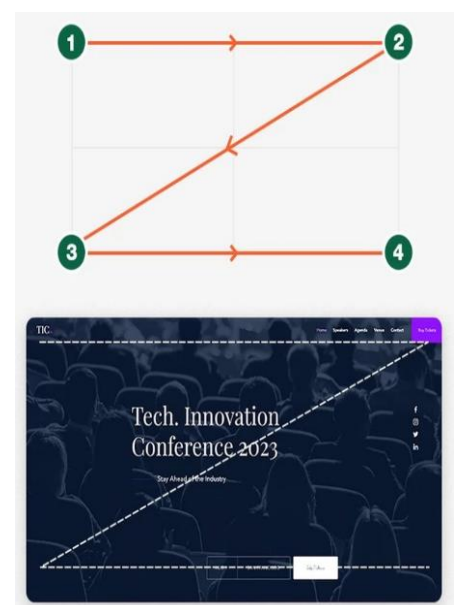
1.5.1 Diseño web con PATRÓN EN Z.

Cuando nos topamos con una página web por primera vez, enseguida le echamos un vistazo rápido para tener una idea de qué trata. Esta forma de "escanear" con los ojos la página se conoce como **lectura rápida** y la **realizamos siguiendo un patrón parecido a la letra Z.**

El **logotipo** se suele colocar generalmente en la esquina superior izquierda para que sea lo primero que perciba el usuario. **En frente**, a la derecha, se suele colocar **el menú de navegación** junto a un llamado a la acción que destaque.

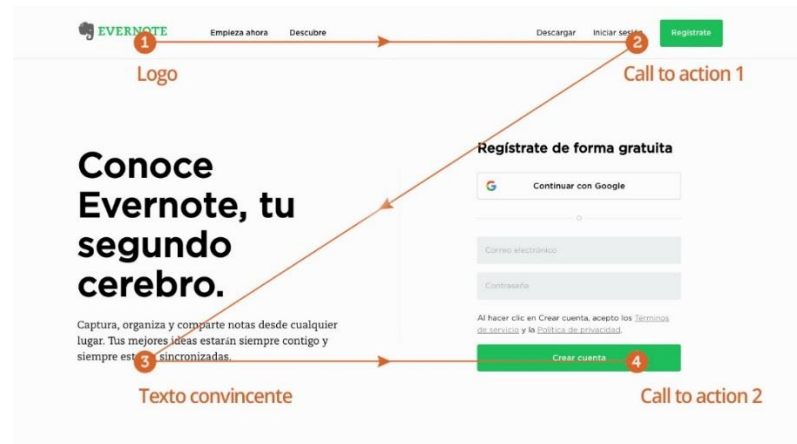
La diagonal que se extiende a lo ancho de la página, de arriba a abajo, es donde se coloca la **información** que más nos interesa que capte la atención. Esto se consigue usando imágenes seductoras acompañadas de un breve texto que explique de qué trata la página web.

La recta final del patrón Z debe ser la culminación de todo lo que la ha precedido hasta ahora. Tanto si has intentado seducir o convencer al usuario para que compre un producto o reserve un servicio, este es el momento y lugar para que lo hagan. Coloca el **llamado a la acción (CTA,**

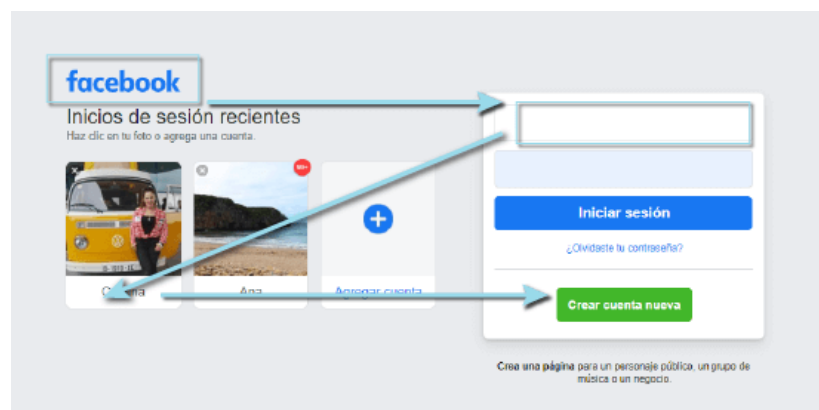




Call to Action) más importante al final del patrón Z con un botón que anime al usuario a la acción.

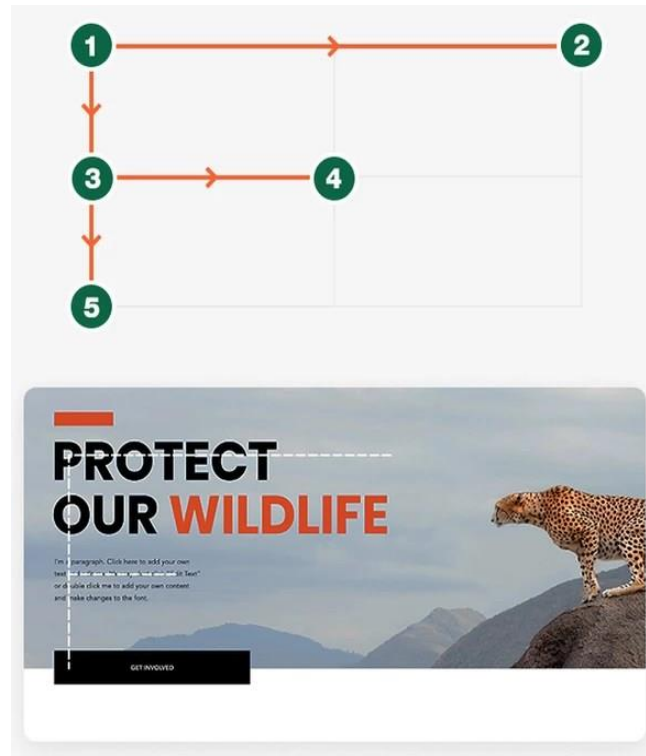


Este patrón de diseño web se usa en páginas de alto contenido visual y muy poco escrito.



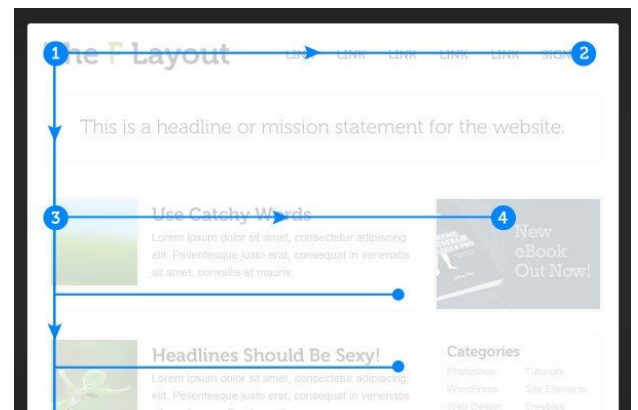
1.5.2 Diseño web con PATRÓN F.

Al igual que el patrón Z, este patrón de diseño web se basa en otro tipo de hábito de lectura. En las páginas web con mucho contenido escrito, tendemos a "escanear" el texto siguiendo un patrón de **lectura en forma de F**. Es decir, la sección superior horizontal de la página recibe gran parte de nuestra atención y, a continuación, dejamos caer nuestra mirada verticalmente hacia la izquierda que se convierte en nuestro punto de referencia.



Si decides usar este patrón de diseño, **invierte lo mejor de tus recursos en la sección superior** de tu página **donde tus visitantes probablemente permanezcan más tiempo**. Escribe en esa franja horizontal un contenido que sintetice de forma convincente de qué trata tu página. En la parte superior de la F se suele incluir un encabezado, un subtítulo y una imagen.

La línea vertical de la F del lado izquierdo tiene como propósito apoyar y hacer más atractiva la sección horizontal. Puedes conseguirlo usando todo tipo de imágenes e iconos o con otros elementos como una lista de puntos o numeración.



Este patrón de diseño web **se usa en páginas web de gran contenido escrito**. Al crear un blog, por ejemplo, este patrón de diseño web es aplicable tanto a la página de inicio como a las distintas entradas del blog.

1.5.3 Diseño web con IMAGEN A TODA PANTALLA.

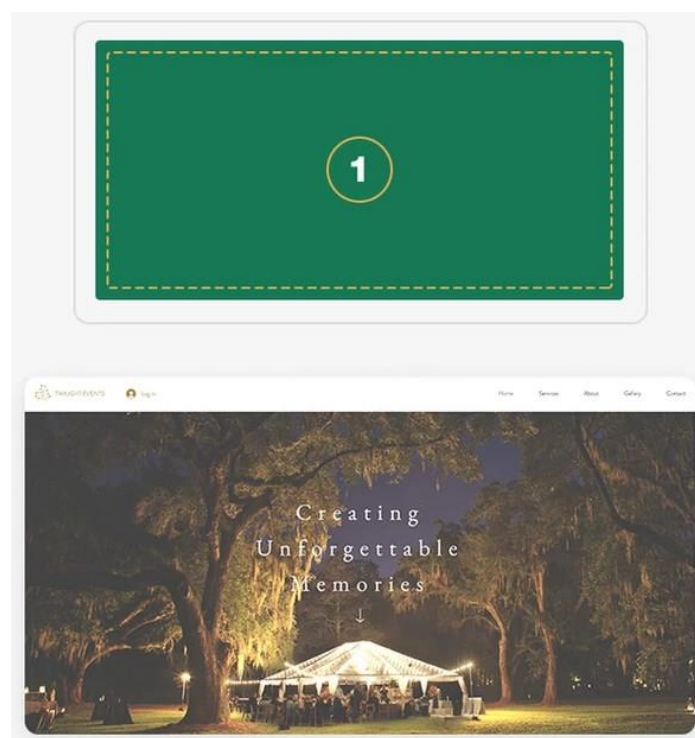
Este patrón de diseño web incluye poco texto sobre una imagen que suele ser grande u ocupar toda la pantalla. Colocar **un elemento visual como protagonista en la pantalla** consigue captar la atención de los usuarios y **generar una poderosa experiencia visual** en la página de inicio. Además, también ayuda a comunicar tu mensaje de forma inmediata y eficaz.

Los elementos visuales (fotografías, ilustraciones o videos) de gran tamaño son capaces de expresar eficazmente y al instante lo que eres y lo que haces.

En este tipo de diseño web, el elemento visual siempre viene **acompañado de una breve frase** para presentar y explicar lo que se ofrece. Por tanto, es conveniente incluir aquí un eslogan “con gancho” que despierte el interés y la curiosidad de tus usuarios.

Además, este patrón de diseño web **funciona estupendamente en su versión para móviles**.

Este patrón de diseño web **se usa en negocios que quieran destacar en un nicho de mercado** específico o un producto en partículas. Por ejemplo, la página web de un restaurante puede presentar su plato estrella, mientras que una página web de boda, a la pareja de novios felices y sonrientes.

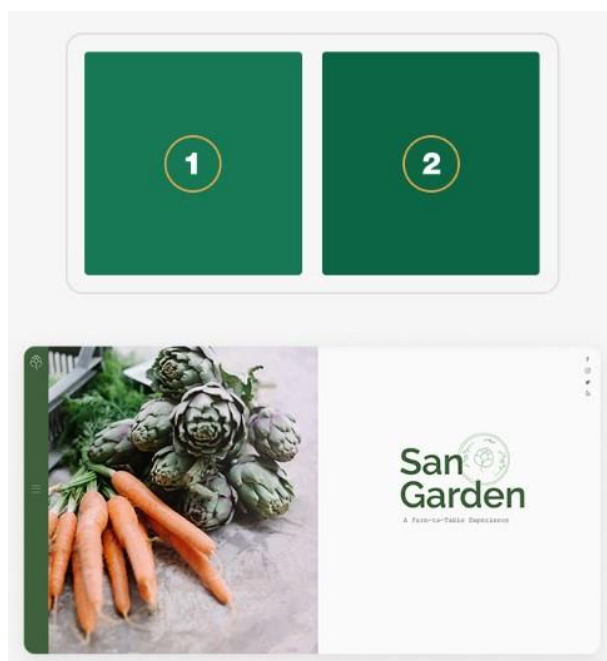




1.5.4 Diseño web con PANTALLA DIVIDIDA.

El diseño con **pantalla dividida** es una de las principales **tendencias** en diseño web. Dividiendo verticalmente la pantalla por la mitad se logra un equilibrio simétrico perfecto. La división en dos partes permite que cada sección **exprese dos ideas de forma completamente diferente o una misma idea, pero desde un ángulo distinto**.

El diseño web que se puede ver en el siguiente ejemplo, presenta una experiencia culinaria mediante una imagen a la izquierda para captar la atención, y mediante texto y arte vectorial a la derecha. El diseño **permite que ambas mitades se complementen** en lugar de competir entre ellas. Este diseño también funciona bien cuando se le pide al usuario elegir entre dos opciones opuestas como las categorías de 'Hombre' y 'Mujer' de una tienda online de moda.



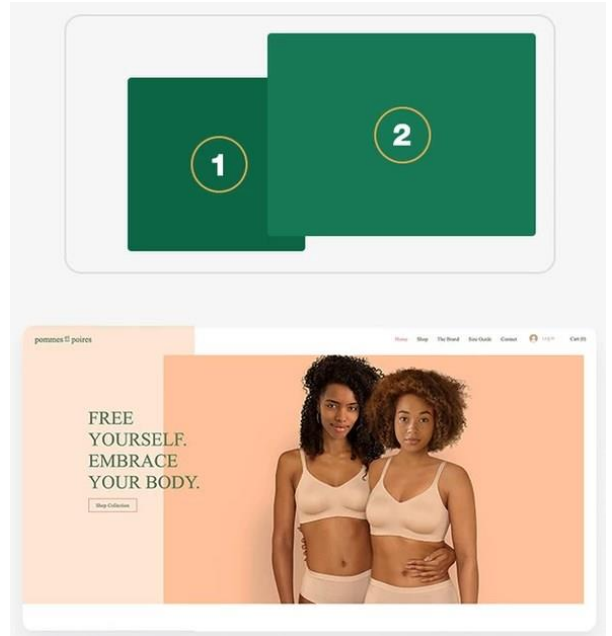
Este patrón de diseño web **se usa en páginas web que ofrecen contenido distinto** de forma significativa, **o páginas web que combinan contenido visual y escrito a partes iguales**. Las **tiendas web** que segmentan a sus usuarios por sexo o edad también encajan con este patrón de diseño web.

1.5.5 Diseño web ASIMÉTRICO.

Al igual que el diseño de pantalla dividida, este diseño web también **divide la pantalla en dos partes, solo que difieren en tamaño y densidad de contenido**. La asimetría de ambas partes genera tensión visual en la pantalla añadiendo dinamismo a la página.

Gracias a la desigual distribución de la escala, color y anchura de la página, se puede **atraer la atención del usuario en algunos elementos antes que en otros**. Para lograrlo, basta con darle más protagonismo a ciertos elementos visuales para convertirlos en puntos de atracción.

Este patrón de diseño web **se usa en páginas web que buscan un look innovador y contemporáneo** y están interesadas en generar **engagement** (generar y retener) en el usuario. Una página web para negocios o el portafolio de una agencia de diseño son buenos ejemplos.

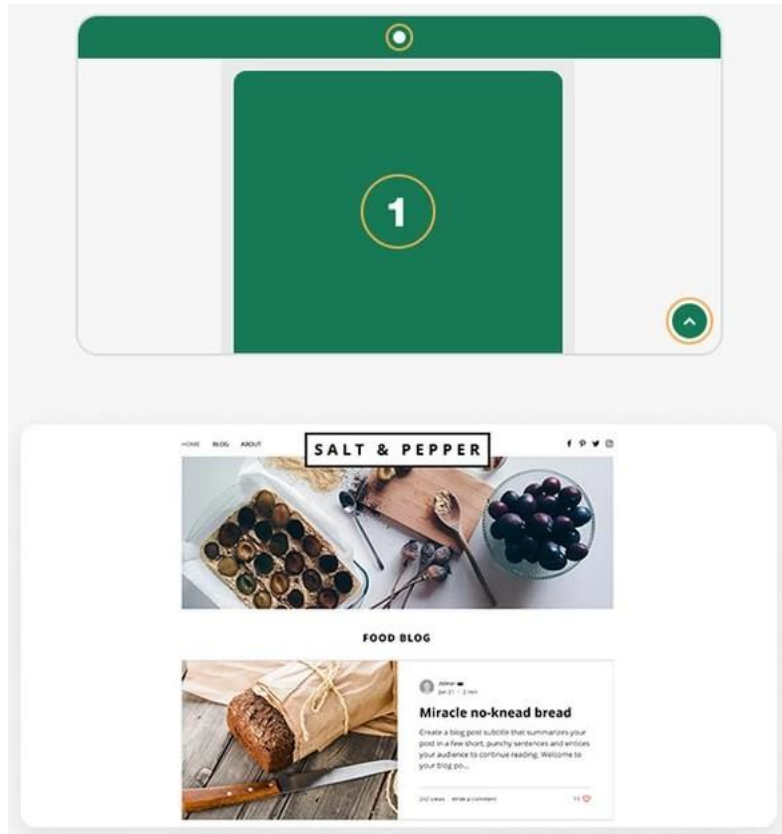


1.5.6 Diseño web de COLUMNA ÚNICA.

Este patrón de diseño web presenta todo su **contenido en una columna vertical**. Es un diseño sencillo y directo.

La navegación aquí es fácil, basta con hacer *scroll* hacia abajo para obtener más información.

Cuando utilices el diseño web de columna única para una página con bastante texto, procura fragmentarlo intercalando imágenes, títulos o subtítulos. Además, es conveniente agregar un botón para volver al principio de la página, o un menú flotante fijo para ayudar al usuario a explorar mejor la página.

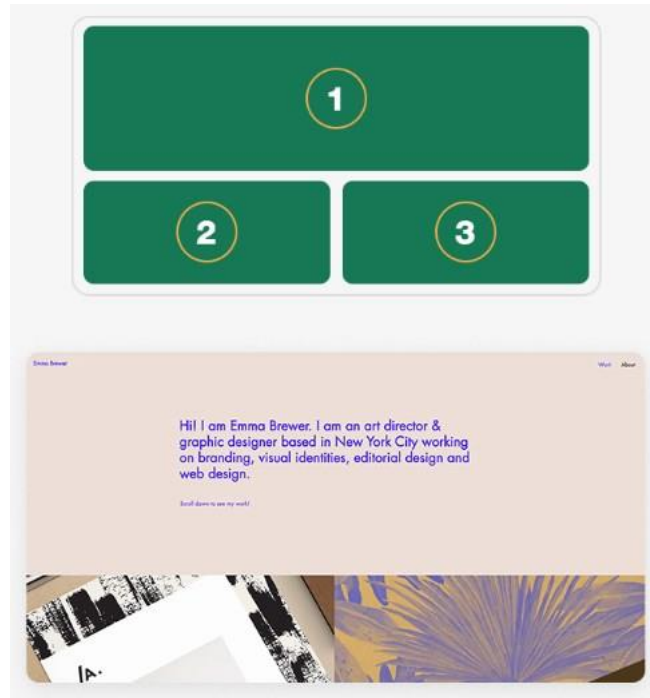


Este patrón de diseño web es ideal para **páginas web con contenido de gran formato o páginas que presentan su contenido de forma cronológica**. Idóneo tanto para **blogs** como para los **feeds de redes sociales**.

1.5.7 Diseño web en BLOQUES.

El diseño web en bloques combina muchas piezas de contenido en un solo diseño geométrico. Gracias a la hábil distribución de la información en cada bloque, los elementos no se solapan entre ellos, sino que contribuyen a dar uniformidad al look general del diseño. En este patrón, **cada bloque conduce a una página interior distinta** donde el usuario puede acceder a más información sobre el tema que más le interese.

Una práctica muy recomendable es añadir un bloque de gran tamaño que sirva de encabezado y nexo de unión entre todos los bloques. Este bloque puede incluir el título de la página y una explicación breve de su contenido.

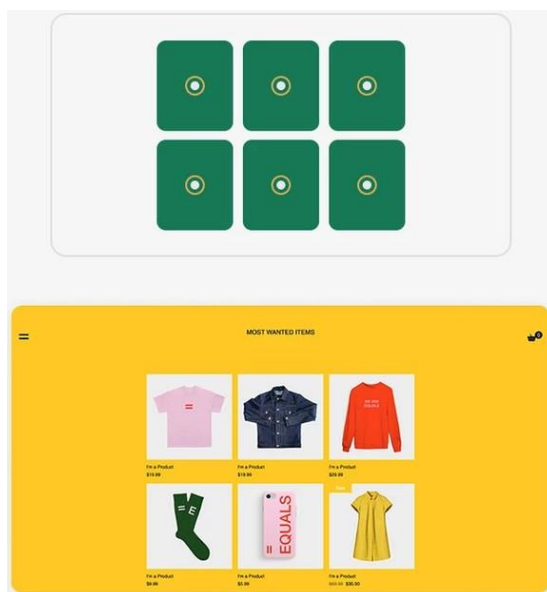


Este patrón de diseño web es ideal para **páginas web ricas en contenido**. Este patrón de diseño web es una buena solución para un **portafolio** de diseño gráfico en el que cada bloque conduce a un proyecto diferente.

1.5.8 Diseño web en MOSAICO.

Al igual que el anterior, el diseño web en mosaico también se sirve de bloques y receptáculos rectangulares para mostrar contenido variado. Este diseño web no es jerárquico, es decir, ningún bloque recibe más protagonismo que otro. Toda la información se trata por igual.

Como cada bloque es idéntico (fuente tipográfica, tamaño, etc.), importar contenido en ellos es fácil. Como resultado, el diseño adquiere un aspecto modular que se adapta perfectamente al tamaño de todo tipo de pantallas. A pesar de la elevada cantidad de contenido que puede albergar, facilita una navegación intuitiva mejorando la experiencia del usuario.



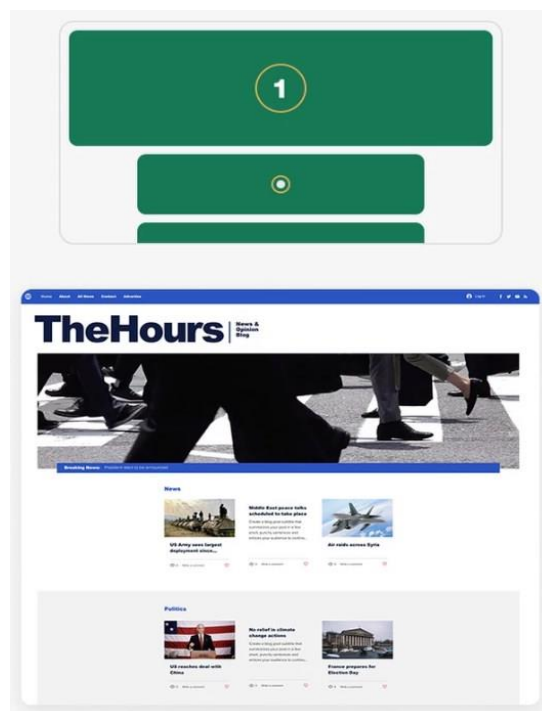
Este patrón de diseño web se usa en **páginas web con mucho contenido**. Este patrón de diseño web es idóneo para un **blog** o una **tienda online**.

1.5.9 Diseño web ESTILO REVISTA.

Inspirado en la prensa escrita, este patrón de diseño web consiste en **distribuir el contenido en varias columnas de estilo periodístico** creando una **compleja jerarquía visual**. Insertando columnas que se pueden modificar individualmente, el diseño web de estilo revista **permite priorizar los principales titulares o encabezados** sobre los artículos más pequeños.

Esto **se puede conseguir a través del tamaño** (las imágenes y los títulos de mayor tamaño son los primeros en captar la atención), **la distribución de los elementos** (el artículo superior será el primero en leerse), **o con el nivel de elaboración del diseño** (un artículo acompañado de imágenes atrae más la atención que uno con únicamente texto).

Fíjate que el diseño web estilo revista utiliza el **patrón de lectura en F** que es capaz de combatir la atención decreciente de tus lectores. Cuando





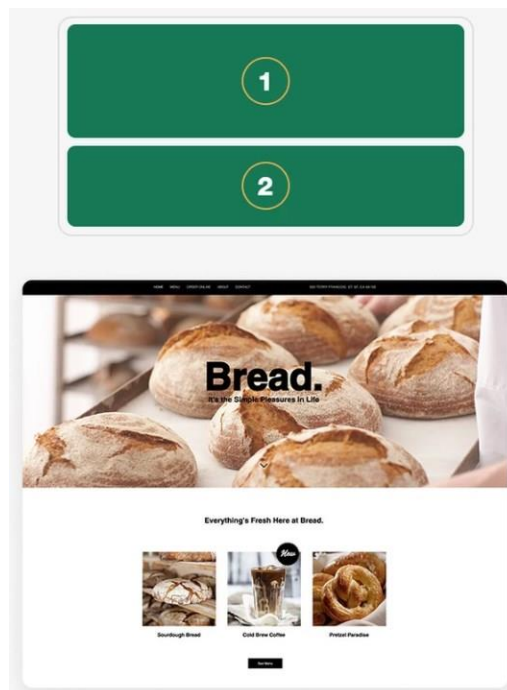
combinas el patrón F con una estructura más compleja **consigues desmenuzar grandes cantidades de información para facilitar su lectura**, y al mismo tiempo también mantienes el diseño de la página despejado y ordenado.

Este patrón de diseño web se usa en páginas web con mucho contenido como **portales de noticias y blogs**.

1.5.10 Diseño web en FRANJAS HORIZONTALES.

Este patrón de diseño web **divide el largo scroll de una página en franjas horizontales**. Cada franja actúa a modo de pantalla completa (o casi) y **convierte su contenido en una sorpresa para el usuario cada vez que hace scroll**.

Para diferenciar cada franja de la anterior, se procura variar el estilo entre una y otra.



Este patrón de diseño web se usa en **páginas web de largo scroll**. También es perfecto para **sitios web de una sola página**.



1.5.11 Plantillas de diseño

Todos estos patrones de diseño web se pueden encontrar en **plantillas de diseño**. Existen infinidad de ellas, y solucionan la tarea de la creación de una interfaz de usuario. Muchas de ellas son descargables de forma gratuita y otras, que pueden ofrecer un acabado más profesional, están disponibles pagando una cantidad determinada.

Estas plantillas, en ocasiones nos pueden solucionar la necesidad de ofrecer un diseño atractivo, pero no nos garantizan algunos requisitos importantes:

- a) Que sean **adaptables** a cualquier dispositivo (*responsive design*).
- b) Que **funcionen** correctamente **en cualquier navegador**.

En los siguientes enlaces puedes ver algunas plantillas: <http://www.webself.net/>,
<http://www.webnode.es/>



2. Aplicaciones

Existen infinidad de aplicaciones para el diseño y desarrollo web, desde editores de código sencillos hasta complejos IDEs (*Integrated Development Environment*). Además, la propia web ofrece muchas herramientas en formato SaaS (Software as a Service), es decir, aplicaciones que sin necesidad de ser instaladas se usan directamente desde el navegador.

En este apartado, nos centraremos en las aplicaciones para la generación de **prototipos**, los diferentes **gestores de contenido** que podemos usar, y por último, veremos las propias aplicaciones para **desarrollar** código.

2.1 Prototipos

Un **prototipo** es una maqueta o modelo de un diseño o dispositivo que ayuda a hacerse una idea del producto final que se obtendrá. El prototipo **permite ver el resultado** de distintos diseños finales, **comprobar alguna funcionalidad** o **realizar pruebas de usabilidad**, entre otras.

De esta forma, **se ahorra tiempo, esfuerzo y dinero**, puesto que es más sencillo realizar cambios sobre un diseño previo que sobre el producto final.

Algunas de las **ventajas** que justifican el uso de prototipos son:

- a) Mejoran la velocidad de desarrollo. Es más eficiente realizar cambios sobre un prototipo antes de comenzar su desarrollo, que sobre un diseño más definitivo donde aparezcan colores y tipografías.
- b) Involucran al cliente. El cliente es el responsable de aprobar el diseño último del sitio, por lo tanto, será más sencillo y útil proponer y hacer cambios sobre un esquema con poco detalle que sobre un diseño ya acabado.

Se pueden distinguir los siguientes **tipos de prototipos**:

2.1.1 SKETCH o BOCETO

Un sketch es el **dibujo o boceto inicial** de todo el sitio web, de los procesos y de las relaciones entre pantallas. Este **dibujo poco detallado** puede estar realizado sobre **papel, pizarra o cualquier formato que permita realizar cambios de manera rápida**.



Este primer tipo es el que suele emplearse **en la fase inicial**, y se trata de un **diseño de muy bajo detalle** en donde se visualiza por primera vez el conjunto de elementos fundamentales de una web. Se basa, esencialmente, en establecer la jerarquía de contenidos, pero sin detalles de diseño.

Puedes ver **más información** en: <https://youtu.be/4xsATP-X3vM>

2.1.2 Wireframe (o Esquema de página)

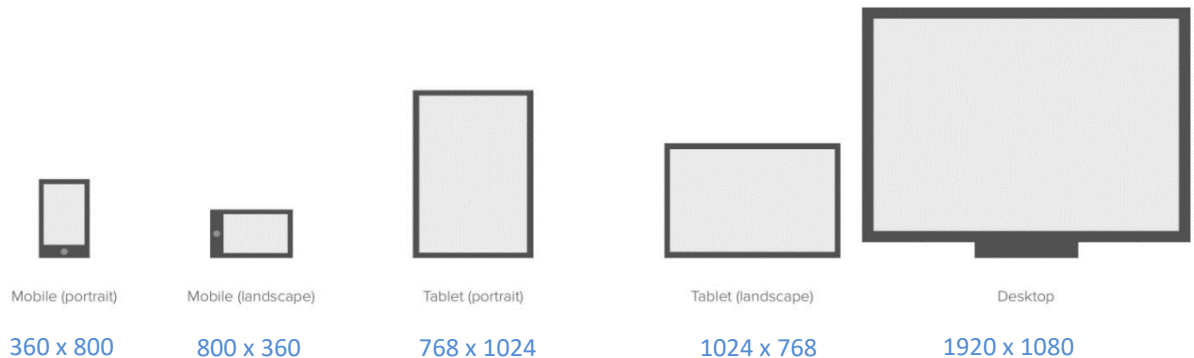
Es una representación estática, en **blanco y negro** o escala de grises, que se centra en la función que hará la página web, y así poder **esbozar eficazmente su estructura y diseño**. **Es esencialmente como un plano de un edificio**, esto es, una representación básica y visual del diseño. Por tanto, un wireframe **tiene que incluir**:

- Un **esquema** de las **principales partes** del contenido (visualizadas como cajas/cuadros/marcos).
- Su **diseño y estructura**.
- La **interfaz de usuario básica**.



Un wireframe **carece de iconografía, imágenes y marca**, lo que permite al desarrollador, al cliente o a las partes interesadas llegar a un consenso únicamente sobre la funcionalidad básica de un concepto, sin distracciones visuales.

A la hora de llevar a cabo el wireframe, veremos que existen diversos **tamaños de pantalla**, por lo que tomaremos como referencia aquellos que más se usan en la actualidad (<https://gs.statcounter.com/screen-resolution-stats/>):

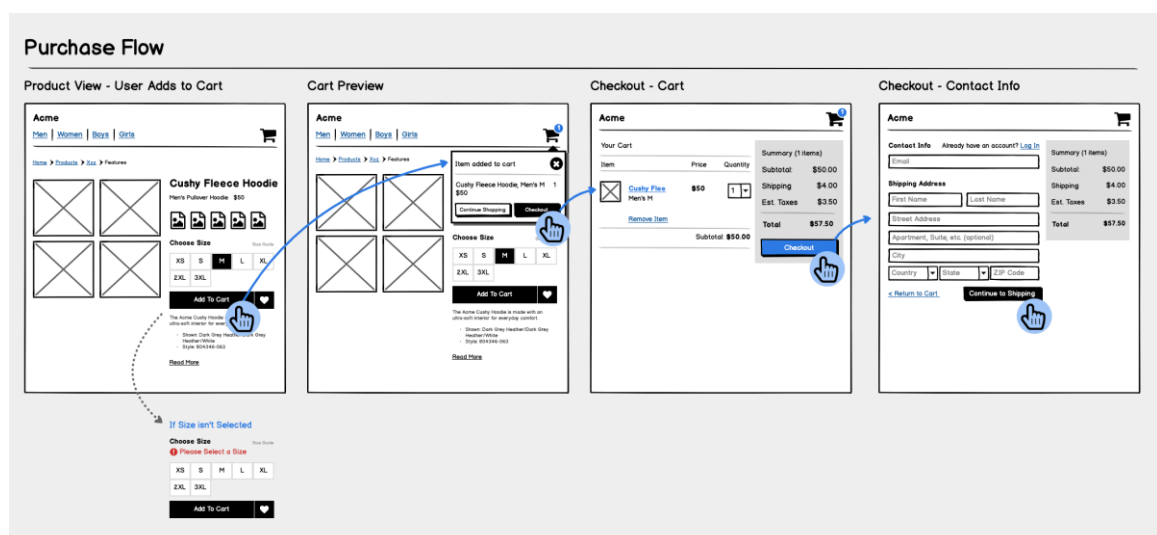


Si existen dudas respecto del tamaño a utilizar, lo **aconsejable es empezar con una resolución baja y luego aumentar la escala** para que el diseño siga pareciendo adecuado en el caso de que la ventana sea más grande.

Puedes ver un **ejemplo** en: <https://youtu.be/v8AjVL4ZdS0>

También existen los **wireflows o árboles de navegación**. Se trata de una combinación de los Wireframes con diagramas de flujo para representar interacciones. Por tanto, nos indica el comportamiento y la navegación entre las distintas pantallas de la web.

Son útiles cuando se necesita mostrar diseños en el contexto de las tareas comunes de los usuarios. Por **ejemplo**, cuando un producto recarga el contenido sin cambiar de página, es decir, el contenido de la página es **dinámico** o por ejemplo puede tener **muchos estados** (un carrito de compra de un Ecommerce o tienda online, los estados de un formulario de login o registro).



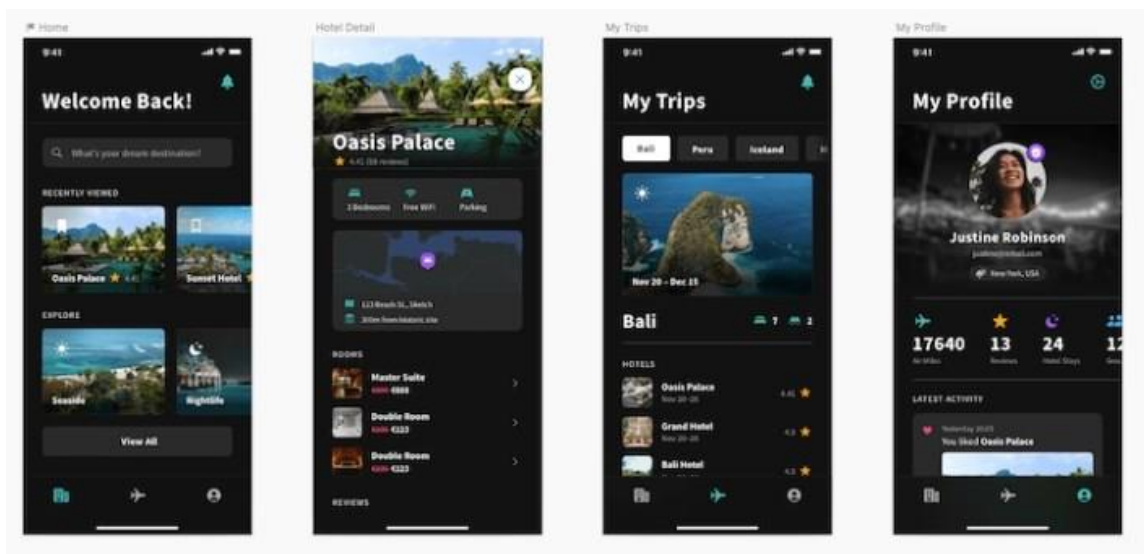
2.1.3 Mockup o maqueta.

Es un diagrama de diseño **estático de alta fidelidad**, que muestran y añaden opciones de diseño como el color, fuentes, iconos, logotipos y elementos de navegación (aunque no tienen por qué ser definitivos). Básicamente, se trata de **un paso más allá de un wireframe**.

A diferencia de un wireframe, un mockup **se parece más a un producto acabado**, pero **no es interactivo** ni se puede hacer clic en él; sigue siendo sólo un gráfico estático. Es en esta fase cuando el cliente o el usuario final revisan la visualidad del producto, cambiando los colores, las imágenes, los tipos de letra, etc.

El diseño de un mockup es una parte importante en el ciclo de desarrollo de una página web, y es uno de los últimos pasos en la etapa de diseño.

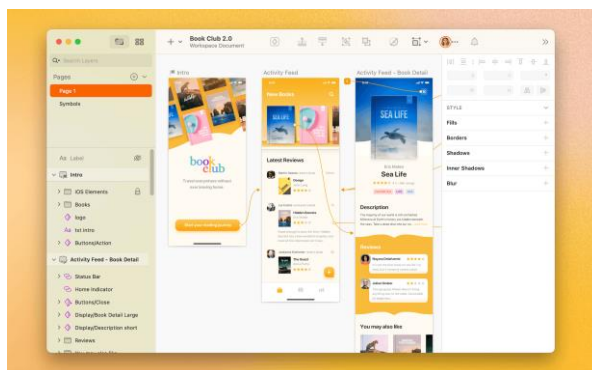
Normalmente, se proporciona más de un Mockup con diferentes opciones al cliente/usuario final, para que pueda decidir qué dirección “artística” le gustaría seguir.



Un mockup **no se puede esbozar o dibujar fácilmente con un software básico**, sino que un diseñador de UX suele utilizar un software digital más avanzado para crear y presentar las opciones, de modo que se pueda realizar una crítica visual más detallada. La comparación de los mockups **ayuda al equipo a ponerse de acuerdo sobre la dirección del diseño**, la **detección temprana de puntos débiles** y la realización de **cambios sin que sea demasiado costosa**.

2.1.4 Prototipo.

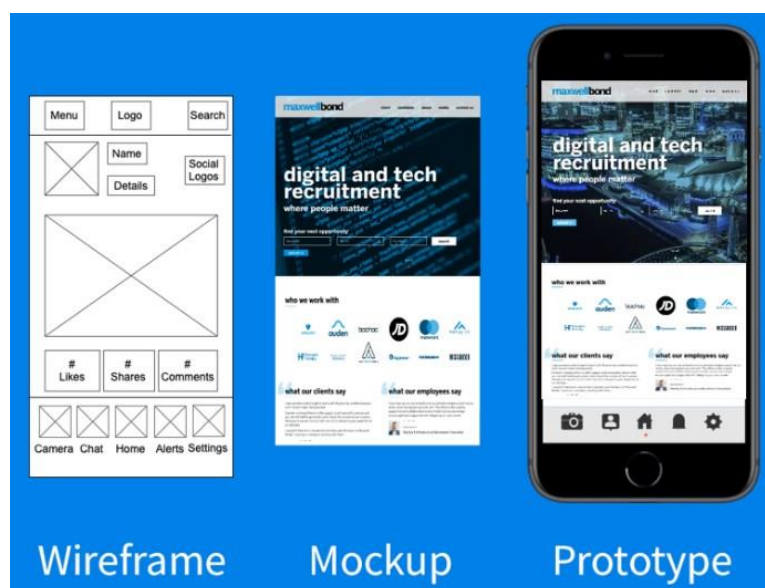
Un prototipo es una **representación de alta fidelidad del producto final** y estimula la **interacción del usuario**. A diferencia de los wireframes y los mockups, un prototipo es clicable y **permite al usuario experimentar el contenido y las interacciones en la interfaz tal y como lo haría en el producto final**. Además, se pueden simular procesos para probar la interacción y el recorrido del usuario.



La diferencia entre un prototipo y un producto final es que la interfaz y el backend no suelen estar unidos. Más bien, los elementos de interactividad se añaden utilizando herramientas de UX como InVision y Sketch, y no con código real. Esto es para ahorrar tiempo y costes de desarrollo en caso de que haya muchas solicitudes de cambio.

Un prototipo es una herramienta excelente para obtener la **opinión de los usuarios y probar el producto**, y permite conocer cómo interactúa un usuario real con el producto y cualquier problema potencial. Ofrece una gran ilustración del recorrido del usuario y de cómo interactúan y se ven los diferentes elementos.

Puedes ver un **ejemplo** en: <https://youtu.be/GVnxuWzab7M>





El **diseño del prototipo**, en cualquiera de sus fases, **debe basarse** en los siguientes aspectos, principalmente:

- 1) **Identificación de los elementos** que forman parte de cada una de las páginas web del sitio web.
- 2) **Distribución del número de elementos** de la interfaz gráfica, para que no exista una saturación de elementos, pero sí haya suficiente información en la misma y la interacción sea correcta.
- 3) **Organización de la jerarquía de elementos, orden, y disposición** de estos.
- 4) **Extensión** adecuada del diseño, para aprovechar eficientemente el espacio en función del dispositivo.
- 5) **Elección de patrones de diseño web** para estandarizar el diseño de interfaces.
- 6) Valoración de los aspectos técnicos de **usabilidad y accesibilidad**.

En la siguiente tabla se recogen algunas **opciones** de aplicaciones disponibles para la realización de wireframes, mockups, y prototipos:

	¿Qué se puede hacer?	Tipo	Enlace
Penpot	Wireframes, mockups y prototipos.	Open-source.	https://penpot.app/
Wireframe.cc	Wireframes.	Gratuita.	https://wireframe.cc
NinjaMock	Wireframes.	Freeware.	https://ninjamock.com
Mockflow	Wireframes, mockups, guía de estilos, sitemaps, testers, etc.	Freeware.	https://www.mockflow.com
iPlotz	“Wireframes navegables”.	Freeware.	https://iplotz.com
Pencil	Wireframes.	Open Source.	https://pencil.evolus.vn/
Balsamiq Mockups	Wireframes y mockups.	De pago.	https://balsamiq.com/
Figma	Mockups y prototipos.	Freeware.	https://www.figma.com
Marvel	Prototipos.	Freeware.	https://marvelapp.com/
Sketch	Prototipos.	De pago.	https://www.sketch.com/
Proto.io	Prototipos.	De pago.	https://proto.io/
Adobe XD	Prototipos.	De pago.	https://helpx.adobe.com/es/xd/get-started.html



2.2 Gestores de contenido o CMS

Además de las herramientas para la creación de prototipos, existen herramientas encargadas de la gestión de contenidos, más conocidas como **CMS** (*Content Management System*). En concreto, se trata de programas que permiten **crear, editar y publicar contenido web** a través de una interfaz gráfica **sin necesidad de tener conocimientos técnicos** avanzados. **Esto permite a los usuarios crear y mantener sitios web** profesionales sin tener que escribir código o aprender lenguajes de programación.

Entonces, **¿por qué son tan importantes estas aplicaciones?**

- a) Porque ofrecen un diseño atractivo del sitio web, lo que implica un valor añadido que puede conseguir que el usuario del sitio web lo prefiera a otro.
- b) Nos dan la **facilidad de crear, editar y publicar contenido** a través de una interfaz gráfica **sin disponer conocimientos de programación**, ya que todo funciona mediante menús y campos a rellenar tipo **WYSIWYG** (*What You See Is What You Get* o Lo que Ves Es Lo Que Obtienes).

Existen dos grandes bloques en los que clasificar los gestores de contenido; por un lado, están los de tipo software propietario, para los cuales es necesario disponer de una licencia de uso, y por otro lado, los de software libre, los cuales permiten a los desarrolladores llegar hasta el código fuente y adaptarlo más fácilmente a las necesidades requeridas. Los gestores más conocidos están basados en tecnología web con lenguaje PHP/HTML y gestores de base de datos como, por ejemplo, MySQL.

La construcción de un sitio web a través de estos gestores de contenido se fundamenta en el uso de elementos de diseño predefinidos que reciben el nombre de **plantillas**. Estas plantillas pueden ser configuradas desde cero por el usuario o, por el contrario, utilizar estructuras sencillas ya predefinidas que se denominan sitios prediseñados, puesto que ya disponen de una estructura definida sobre la que solo hay que incorporar contenidos particulares del sitio.

Una **plantilla de diseño** es una web ya prediseñada, en la que el usuario solo tiene que completarla con el contenido deseado. Se trata de la manera más sencilla de comenzar a crear un sitio web, y menos costosa.

Ten en cuenta que... Las plantillas web solucionan la tarea de creación de interfaces de usuario. Sin embargo, algunas de ellas no son completamente personalizables, pueden ofrecer problemas de visualización en los navegadores, tener una velocidad de carga elevada o que no sean adaptables a múltiples dispositivos.

2.2.1 CMS más utilizados en la actualidad

Los gestores de contenido, o CMS (Content Management System), son plataformas que permiten crear y administrar sitios web sin necesidad de conocimientos técnicos avanzados.

Según **W3Techs**, actualmente (septiembre 2023) el CMS más utilizado y popular es **WordPress** y tiene una cuota de mercado del 43.1%. Otros CMS populares incluyen **Shopify**, **Wix**, **Squarespace**, **Joomla!**, **Drupal**, **Adobe Experience**, y **Prestashop**.

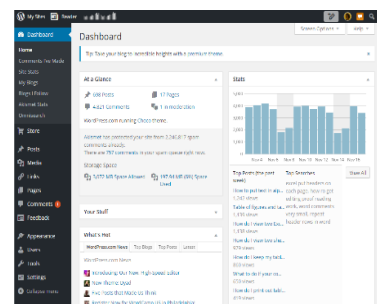
A continuación, vamos a ver las características principales de cada uno de ellos y su cuota de mercado.

1) WordPress 43.1%

Este gestor de contenidos está presente en más de 18 millones de sitios web en todo el mundo. Se trata del **software libre gestor más utilizado**.



Bonitos diseños, potentes características, y la libertad de crear lo que tú quieras. WordPress es al mismo tiempo gratis y de precio incalculable.



El uso y la instalación de WordPress son sencillos, lo que lo hace especialmente indicado para **sitios web pequeños**. A continuación, se especifican algunas de las ventajas de esta famosa plataforma:

- Gran **comunidad** de soporte, por tratarse de un gestor muy utilizado.
- **Instalación y configuración sencilla**.
- **Interfaz intuitiva**.
- Frecuentes **actualizaciones de seguridad**, lo cual favorece tanto a la seguridad como a la funcionalidad del sitio.



El catálogo de plantillas o themes de Wordpress es grandísimo, con todo tipo de layouts a nuestra disposición.

Uno de los mayores inconvenientes con los que cuenta WordPress es el **rendimiento limitado**, si se utiliza para desarrollar sitios con mucho tráfico.

DIFERENCIAS entre WordPress.org y WordPress.com

WordPress.org y **WordPress.com** son dos plataformas populares para la creación y gestión de sitios web. Aunque comparten el mismo nombre, **existen diferencias importantes** entre ambas. A continuación, se presentan algunas de estas diferencias entre ambas:

- 1) **Hospedaje:** En WordPress.org, debes descargar e instalar el software de WordPress en tu propio servidor web o en un proveedor de alojamiento de tu elección. En cambio, WordPress.com proporciona alojamiento web gratuito o de pago para tu sitio.
- 2) **Control y personalización:** Con WordPress.org, tienes un mayor control y flexibilidad sobre tu sitio web, ya que puedes acceder a los archivos del núcleo de WordPress y personalizarlos según tus necesidades. En WordPress.com, las opciones de personalización están más limitadas y no puedes acceder a los archivos del núcleo.
- 3) **Plugins y temas:** En WordPress.org, puedes instalar y utilizar cualquier plugin o tema de WordPress de tu elección para ampliar las funcionalidades y personalizar el diseño de tu sitio. En WordPress.com, la selección de plugins y temas está más restringida, y solo puedes utilizar los que están disponibles en su repositorio.
- 4) **Monetización:** WordPress.org te permite monetizar tu sitio web de diversas formas, como mostrar anuncios, vender productos o servicios, y obtener ingresos de afiliados. En WordPress.com, hay restricciones en la monetización, especialmente en los planes gratuitos.

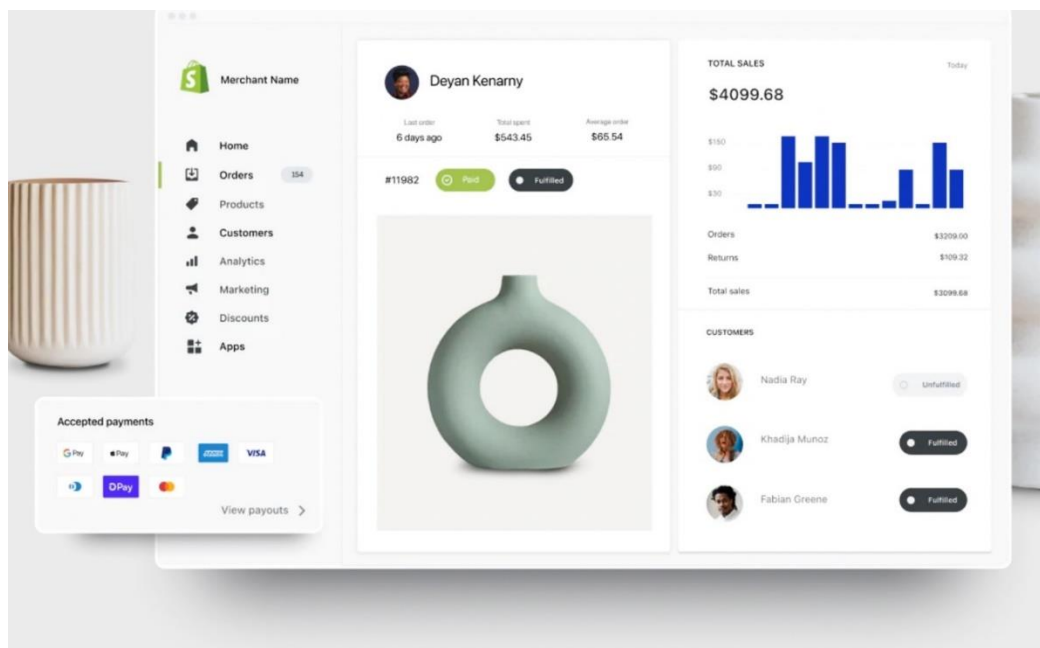


- 5) **Soporte:** WordPress.org ofrece una amplia comunidad de usuarios, foros de soporte y documentación para ayudarte con cualquier problema o duda que puedas tener. En WordPress.com, también hay soporte disponible, pero puede ser más limitado dependiendo del plan que elijas.

En **resumen**, **WordPress.org es una opción más flexible** y versátil para aquellos que desean tener un mayor control sobre su sitio web y personalizarlo según sus necesidades. WordPress.com es una opción más conveniente para aquellos que prefieren una solución de alojamiento todo en uno con menos opciones de personalización pero con facilidad de uso.

2) Shopify 4%

Shopify es una plataforma de comercio electrónico que permite a los usuarios crear y **administrar tiendas en línea**.



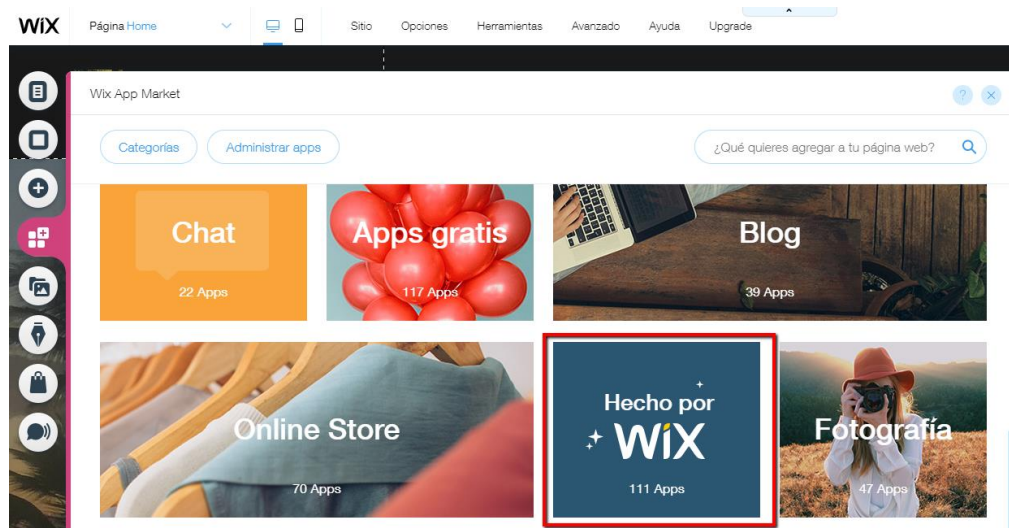
Con Shopify, puedes crear un sitio web de comercio electrónico con herramientas potentes que te ayudarán a encontrar clientes, aumentar las ventas y gestionar tu día a día. Shopify ofrece una amplia variedad de **plantillas personalizables** y **aplicaciones** para agregar funcionalidades adicionales a tu tienda en línea.



Además, Shopify **ofrece herramientas de marketing integradas** para ayudarte a promocionar tu negocio y aumentar tus ventas.

3) Wix 2.5%

Wix es una plataforma de creación de sitios web que permite a los usuarios crear y administrar sitios web profesionales sin necesidad de conocimientos técnicos avanzados.



Con Wix, puedes crear un sitio web utilizando su creador de páginas web con la **función de arrastrar y soltar**, eligiendo entre **más de 800 plantillas** personalizables. Wix también ofrece herramientas avanzadas para agregar funcionalidades adicionales a tu sitio web, como una **tienda en línea**, un **sistema de reservas** y **herramientas de SEO** para aumentar el tráfico orgánico.

4) Squarespace 2.1%

Squarespace es una plataforma de **creación de sitios web** y un **sistema de administración de contenidos** todo en uno.



SQUARESPACE

Con Squarespace, puedes crear un sitio web profesional utilizando sus plantillas personalizables y herramientas de diseño. Squarespace también ofrece herramientas para vender productos en línea, registrar dominios personalizados y realizar un seguimiento de las estadísticas de tu sitio. Aunque **no ofrece tantas herramientas**



de ventas y marketing especializadas como Shopify, Squarespace ofrece más funciones de comercio electrónico que Wix o Weebly sin tener que instalar aplicaciones de terceros.

5) Joomla 1.8%

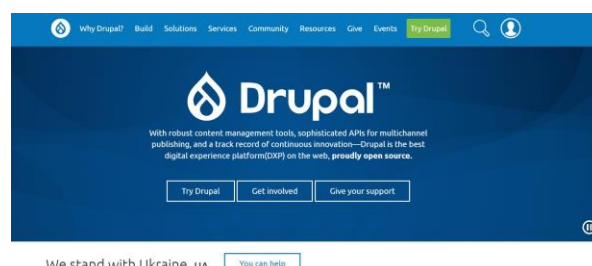
Joomla! es un sistema de gestión de contenidos de código abierto y premiado. Está indicado tanto para **expertos en programación web como aquellos que no se encuentran muy vinculados con el tema.**



Una de sus características principales es el **diseño de software orientado a objetos**, lo que permite a los usuarios más expertos desarrollar sus propias extensiones. También permite utilizar plantillas diseñadas o crear nuevas a partir de estas.

6) Drupal 1.1%

Drupal es un sistema de gestión de contenidos de **código abierto, modular y multipropósito** que permite a los usuarios crear y administrar sitios web y aplicaciones en línea. Al estar basado en una arquitectura modular permite un **alto grado de personalización**, y también ofrece una **amplia variedad de herramientas** para administrar y publicar contenido.

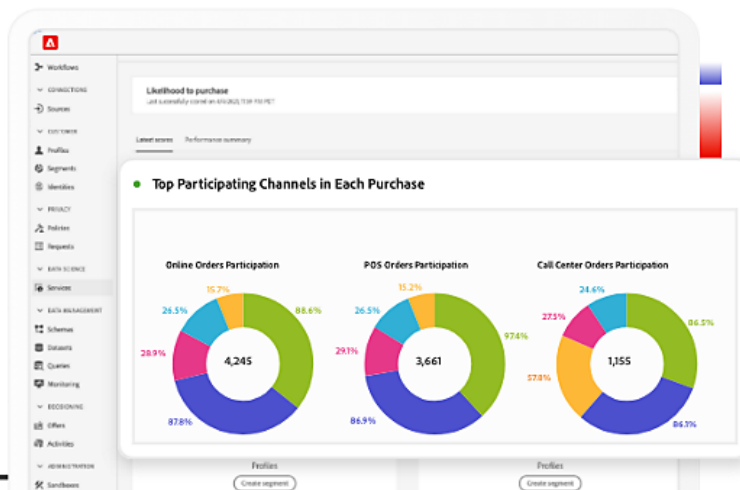


Dispone de un panel de administración, **infinidad de plugins** y, por supuesto, plantillas, aunque sin duda, uno de los recursos más importantes de Drupal es su **API**.



7) Adobe Experience 1.1%

Adobe Experience Manager es un sistema de gestión de contenido **descentralizado** que permite a los equipos desarrollar y distribuir rápidamente experiencias de cliente a través de todos los canales y dispositivos.



Con Adobe Experience Manager, puedes crear contenido una vez y reutilizarlo en cualquier canal, lo que te permite llegar a clientes, socios empresariales y empleados de manera eficiente. Adobe Experience Manager también ofrece herramientas para personalizar experiencias y realizar **pruebas A/B para optimizar el rendimiento**.

Las **pruebas A/B** en el ámbito del Márketing Digital y la Analítica web para analizar cuál de las dos variantes o versiones (A o B) atiende mejor a nuestro público o funciona mejor. Por ejemplo, un botón de CTA azul y uno amarillo.

8) PrestaShop 0.8%

PrestaShop es un sistema de gestión de contenidos libre y de **código abierto** diseñado específicamente para **crear y administrar tiendas en línea** de comercio electrónico.



PrestaShop

Con PrestaShop, puedes personalizar tu tienda en línea utilizando módulos y plantillas, y agregar funcionalidades adicionales como soluciones de pago y entrega.



2.3 Aplicaciones de desarrollo

Existen infinidad de aplicaciones para el desarrollo web, desde editores de código sencillos hasta complejos IDEs que incluyen herramientas como intellisense, scaffolding y la posibilidad de incluir múltiples plugins para tareas específicas.

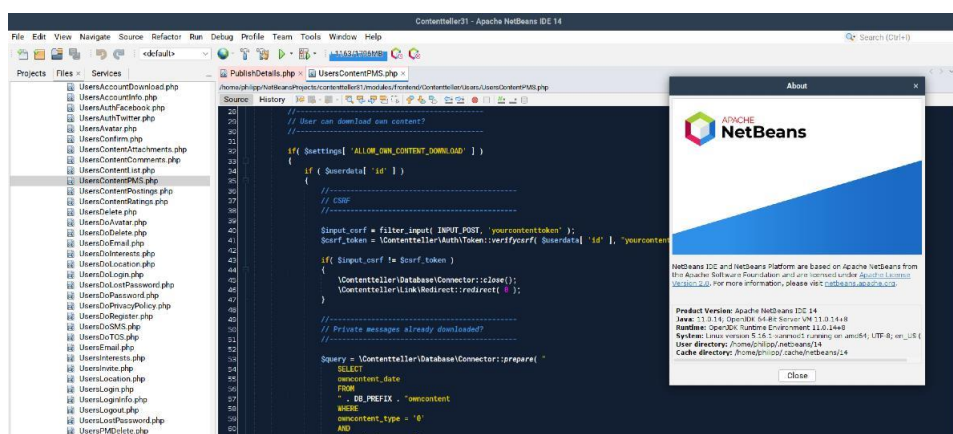
Incluso la propia web ofrece muchas herramientas en formato SaaS (Software as a Service), es decir, aplicaciones que sin necesidad de ser instaladas se usan directamente desde el navegador. En nuestro caso, nos centraremos en aplicaciones de escritorio de cierta importancia.

2.3.1 Netbeans

NetBeans es un entorno de desarrollo integrado libre, **hecho principalmente para el lenguaje de programación Java** y que además cuenta con un número importante de módulos para extenderlo. Aunque está pensado para Java, cuenta con **soporte para muchos otros tipos de proyectos**, con especial énfasis en los proyectos web (**Javascript, PHP, HTML, CSS...**).



NetBeans es un proyecto de **código abierto** de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento.



Ventajas:

- Su descarga está disponible de forma **gratuita** en la web de apache, además del soporte para bastantes lenguajes.



- Puede abrir ficheros con **varios lenguajes integrados**, y es capaz de resaltar la sintaxis correctamente y ofrecer un buen **intellisense para web** (CSS, JavaScript, etc).
- Existen **plugins** para distintas tareas que pueden resultar útiles.

Desventaja: su utilización puede hacerse **un poco pesada** en ocasiones según el número de proyectos cargados.

2.3.2 Eclipse

Es uno de los IDE más populares que existen. Su gran aportación al mundo del desarrollo es que ha dado lugar a forks o proyectos paralelos que partiendo de Eclipse, han derivado en IDEs especializados como el de Spring, Nodeclipse, Aptana, etc.



Eclipse proviene de un antiguo software de desarrollo Java llamado Visual Age que IBM encargó de liberar para que fuera mantenido y desarrollado de manera abierta. Este software ha evolucionado convirtiéndose en un entorno ampliamente soportado con infinidad de usos y plugins, aunque **su principal ámbito son las aplicaciones Java**.

A la hora de desarrollar para la web, la edición EE de Eclipse nos ofrece varias opciones por defecto:

- Proyecto web dinámico, asociado a Tomcat y JSP.
- Proyecto web estático, para **proyectos con HTML, CSS y JavaScript**.
- Proyecto JavaScript.

En cualquier caso, **si vamos a editar ficheros HTML nos ofrece la opción de ver el código o alternar código y vista**.

En general, en lugar de ofrecer un editor de tipo **WYSIWYG** (What You See Is What You Get) donde se arrastran los componentes, cada vez que seleccionamos uno de estos elementos, se abre un **asistente** que nos permite **ajustar los atributos** del componente y luego este se incluye en el código fuente.



Ventajas:

- Su descarga está disponible de forma **gratuita** en la web de eclipse.org.
- Ofrece **intellisense**, refactorización, soporte para git de serie, etc.
- También existen **plugins** para distintas tareas que pueden resultar útiles, algunas de pago, pero muchas gratuitas.

Desventaja: al igual que en otros entornos, puede hacerse **un poco pesado** según el número de proyectos cargados; no obstante, es algo que se puede resolver si cambiamos el workspace.

2.3.3 Webstorm

Webstorm (**Jet Brains**) es un entorno de desarrollo creado por JetBrains **del estilo de Eclipse y Netbeans** con la diferencia de ser **de pago**. Pese a que existen esas otras alternativas, WebStorm se distingue por ser un entorno muy profesional y actualizado para proyectos muy específicos.



En cuanto al entorno, ofrece todo lo que un desarrollador puede requerir: depuración, ejecución, refactorización, generación de código, control de versiones, etc.

2.3.4 Entornos ligeros

Existe también una amplia variedad de entornos de desarrollo ligeros y **mucho menos pesados que los IDE** tradicionales. Todos ellos tienen en común su pequeño tamaño, pero a la vez su versatilidad y su facilidad para **ampliar funcionalidades a base de plugins**.

Estos son algunos. Aunque existen muchos más como **Emacs, Notepad++, TextMate**, etc.

A. SUBLIME TEXT.

Sublime Text es un editor de código que a día de hoy sigue siendo el favorito de muchos desarrolladores por su sencillo y elegante diseño, y multitud de funcionalidades útiles para el desarrollo. A diferencia de muchos otros, **no es de código abierto, pero puede usarse sin pagar por él** aunque teniendo que soportar de vez en cuando un recordatorio sobre su compra.

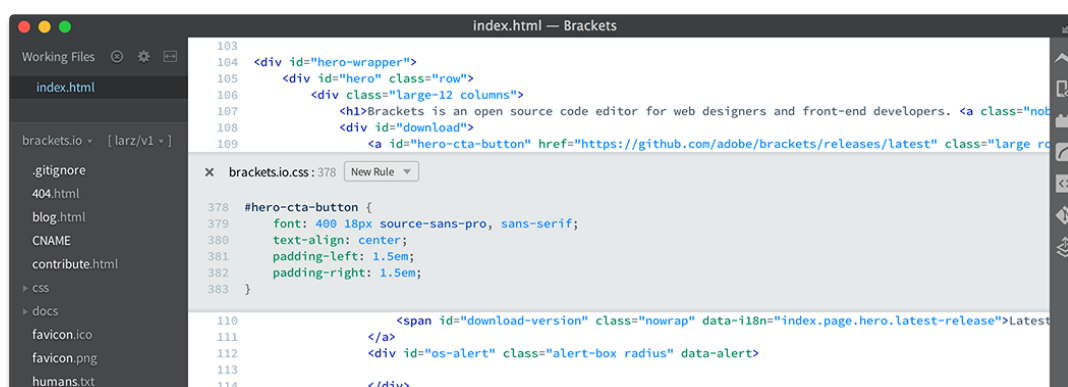




Mediante la secuencia **Ctrl+alt+p** podremos instalar diversos paquetes o plugins. Para web, podemos destacar el de **CSS3, HTML, Color Picker, Emmet...** Este último nos ayudará a completar una estructura de HTML y CSS mediante abreviaturas, con el objetivo de evitar fallos de sintaxis, así como de ahorrarnos tiempo en el desarrollo del código.

B. BRACKETS.

Brackets es un mini editor **desarrollado por Adobe**, también **basado en HTML5 y Node.js** y con opción para ser ampliado a través de **plugins**. Prácticamente ofrece los **mismos elementos básicos de Atom** y además de serie permite **ejecutar en el navegador un fichero recién editado** de forma directa. El **intellisense** para HTML es algo mejor que en Atom.



C. VISUALSTUDIO CODE.

Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows, Linux, macOS y Web.



Incluye soporte para la depuración, **control integrado de Git**, **resaltado de sintaxis**, **finalización inteligente de código**, fragmentos y refactorización de código. También es **personalizable**, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es **gratuito y de código abierto**, aunque la descarga oficial está bajo software privativo e incluye características personalizadas por Microsoft.



2.3.5 Entornos en la nube

Estamos en plena era de los servicios en la nube, donde ya no es preciso tener todos los programas instalados en local, sino que podemos apoyarnos en plataformas online.

Además de servicios de hosting de aplicaciones y de bases de datos, también existen completos entornos que poco tienen que envidiar a los entornos de escritorio, y todo ello desde un simple navegador.

Algunos entornos en la nube que podemos destacar son los siguientes:

A. AWS Cloud9

AWS Cloud9 es un producto de **Amazon** y es de **pago**. Es un IDE basado en la nube que nos permite **escribir, ejecutar y depurar nuestro código sólo con un navegador**, es decir, podemos desarrollar aplicaciones “sin servidor”. Incluye un editor de código, un depurador y un terminal.

Además, incluye **herramientas para JavaScript, Node.js, Ruby, Python, PHP**, etc.

Al ser un servicio en la nube, podemos **compartir nuestro entorno** de desarrollo.

B. Codeanywhere

Este es otro entorno **online de pago** que nos ofrece la posibilidad de desarrollar **aplicaciones fullstack**, soportando los principales lenguajes de programación: JavaScript/TypeScript, PHP, Python, Ruby, Go, Java, C/C++, C#, etc, y códigos como HTML y CSS.

Gracias a la gran variedad de aplicaciones disponibles para iOS y Android, esta herramienta permite a cualquier persona codificar literalmente en cualquier lugar.

Además de estos dos entornos en cloud, existen más como **Eclipse Che, Koding, ShiftEdit, Orion, Python Fiddle**, etc.

2.3.6 Entornos de prueba

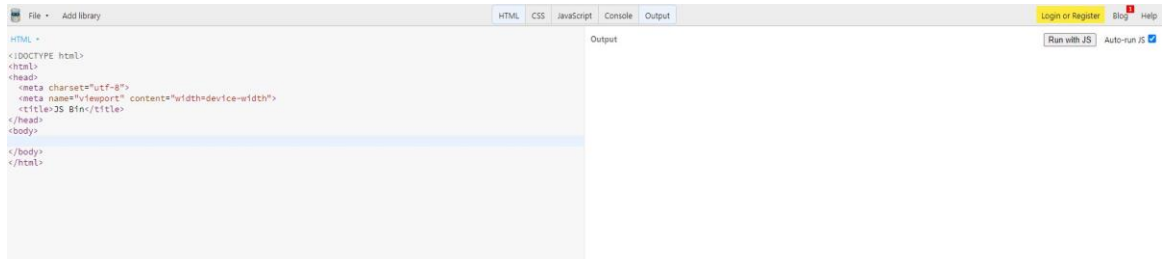
En los últimos años han proliferado herramientas para desarrolladores web que nos permiten **editar código HTML5, CSS y JavaScript de forma directa**. El propio navegador es un entorno de desarrollo ágil y en este tipo de sitios no tenemos más que **entrar y editar**, y



resultan especialmente útiles para llevar a cabo **pruebas rápidas, probar el código de otros, o crear pequeñas demos.**

A. JSBIN.

Jsbin (<https://jsbin.com>) está orientado a **JavaScript** pero puede usarse perfectamente para editar **HTML y CSS**. Conforme se escribe, vamos viendo el resultado. En caso de usar JavaScript tendremos que ejecutar pulsando un botón.



Podemos editar de forma directa nada más entrar, aunque también podemos darnos de alta y guardar lo que queramos.

B. JSFiddle.

JSFiddle (<https://jsfiddle.net>) también está pensado para desarrollar código **JavaScript**. Nos **facilita el uso de frameworks** existentes como **jQuery, Angular, Backbone**, etc, ya que podemos añadirlos en el proyecto que estemos haciendo sin necesidad de descargarlos o referenciarlo.