



**Abogacía
Española**
CONSEJO GENERAL

Libro de Usos

Versión 1.9

Lista de distribución:
Interna, empresas colaboradoras

Elaborado por:
Héctor Bernardo para RedAbogacía

Nº de páginas: 83

La información contenida en este documento y, en su caso, en cualquier fichero anexo al mismo tiene carácter confidencial, está exclusivamente dirigida a miembros del Consejo y colaboradores externos, siendo propiedad del Consejo General de la Abogacía Española.



Imprime este documento sólo si es necesario: el medio ambiente es cosa de todos.

Índice

1. Definición. Consideraciones.

1.1	Definición de usabilidad	4
1.2	Accesibilidad	6
1.3	¿Por qué son importantes?	6
1.4	Redacción	7
1.5	Diseño. Consideraciones generales	8
1.6	Navegación	9
1.7	Configuración	10
1.8	Ayuda	10
1.9	Publicidad	12
1.10	Heurísticas de usabilidad de Nielsen	13
1.11	Maquetación	15
1.12	Velocidad de carga	16
1.12	Resolución y diseño adaptable	17
1.13	Soporte en los principales navegadores	18
1.14	Organización de los datos, tecnologías y nomenclatura	19
1.15	Testeo	20

2. Elementos Principales

2.1	Consideraciones tipográficas	23
2.2	Colores	26
2.3	Títulos y subtítulos	27
2.4	Imágenes	29
2.5	Iconografía	31
2.6	Formularios. Consideraciones generales	33
2.7	Estructura Intranet	36
2.8	Mensajes	37
2.9	Enlaces	42
2.10	Botones	44
2.11	Labels e inputs	48
2.12	Selects	51
2.13	Paginación	52
2.14	Tablas	52

2.15 Captchas	53
2.16 Pestañas	54
2.17 Icono semáforo	55
2.18 Buscadores	55
2.19 Vídeo y contenido flash	56

3. Dispositivos Móviles

3.1 Usabilidad móvil. La importancia	58
3.2 Web móvil, aplicación nativa y aplicación híbrida	59
3.3 Características y contextos de uso	60
3.4 Diseño y maquetación	61

4. Utilidades

4.1 Evaluación Heurística	78
4.2 Ubicación de archivos en red	80
4.3 Enlaces de interés	81

1.1 Definición de usabilidad

Usabilidad es la medida de los atributos que hacen que una interfaz de usuario resulte fácil de utilizar para cualquier usuario. Independientemente del soporte utilizado para acceder a dicha interfaz o el nivel de familiaridad de los usuarios con las nuevas tecnologías. Cualquier diseño, en cualquier ámbito, debería ser en la medida de lo posible, el que debe adaptarse al usuario, y no a la inversa.



Hay diferentes características que deben contemplarse a la hora de desarrollar cualquier proyecto, siendo muchas de ellas fundamentales para asegurar su calidad:

Facilidad de Uso

Durante el primer uso de la aplicación, a un usuario no debería resultarle complicada la realización de -al menos- las tareas básicas.

Eficiencia

La aplicación debe facilitar que las tareas para la que fue concebida y ser ejecutada con rapidez, una vez ha sido aprendida por el usuario.

Fácilmente memorizable

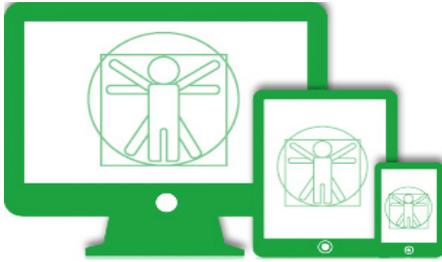


Tras su aprendizaje, aunque haya transcurrido cierto periodo de tiempo, cualquier usuario debería retomar la actividad rápidamente sin otro nuevo periodo de aprendizaje.

Errores

Una buena práctica de usabilidad en cualquier servicio es evitar que los usuarios cometan demasiados errores, lo que aumenta la rapidez en la consecución del objetivo y un mayor grado de satisfacción con el mismo. El ideal sería una aplicación tan sencilla e intuitiva que un usuario no se equivocara.

Utilidad



Es vital que los objetivos de una aplicación sean realmente los que necesitan los usuarios. La aplicación debe ser desarrollada partiendo de este principio. Así mismo debe evitarse hacer más compleja una aplicación con funcionalidades que no sirven de manera práctica a la consecución de los objetivos.

Estandarización

En los últimos años se ha propagado la utilización de las nuevas tecnologías. Ha existido un periodo un tanto caótico en el que aplicaciones informáticas y especialmente sitios web, eran muy heterogéneos.

Desde hace años se ha evolucionado y la tendencia es a seguir determinados estándares hasta el punto de que los usuarios han desarrollado el **hábito**. Esto supone que el conocimiento ya se ha asimilado con antelación, por lo que el seguir dichos estándares mejora considerablemente la experiencia del usuario.



Por ejemplo, asociar el icono de lupa con la función de búsqueda.

Existen otros atributos que se irán aplicando a lo largo de este libro.

Identidad e Información

Debe mostrarse de forma rápida, precisa y completa qué contenidos o servicios ofrece la aplicación o sitio en el que nos encontramos.

El acceso a los datos de la compañía debe estar al alcance del usuario en cualquier sección o proceso de la aplicación. Debe existir una coherencia estética y estructural a lo largo de todo el sitio que permita al usuario identificar de una manera muy clara que no ha abandonado la navegación principal de nuestro sitio.

Los datos de la empresa deben ser accesibles al usuario en cualquier momento, facilitando siempre que sea posible asistencia directa con un servicio de orientación al usuario. Es especialmente recomendable un servicio de atención telefónica.

Satisfacción

El conseguir los objetivos de manera rápida, cómoda, eficaz, y con el mínimo de errores posibles contribuyen a mejorar la experiencia de usuario, junto con el resto de puntos vistos en este libro. Uno de los propósitos de todo diseño es lograr el mayor grado de satisfacción en todas las posibles personas que van a hacer uso del mismo. Es siempre recomendable medir la satisfacción de los usuarios mediante tests para así llevar un seguimiento de la usabilidad de una aplicación.

1.2 Accesibilidad

Es la posibilidad de que un producto o servicio web pueda ser accedido y usado por el mayor número posible de personas, indiferentemente de las limitaciones propias del individuo o de las derivadas del contexto de uso.

La accesibilidad y en general todo diseño está supeditada al uso equitativo: el producto debe ser útil para cualquier grupo de usuarios. Hay que evitar la segregación y la estigmatización de los usuarios.

1.3 ¿Por qué son importantes?

Un usuario es reacio a perder tiempo investigando en el funcionamiento de una aplicación web. Un usuario puede abandonar una web si:

- 1) **Es difícil.**
- 2) **La página principal deja de indicar lo que puede hacer en el sitio.**
- 3) **Se pierde.**
- 4) **La información es difícil de leer o no responde a sus preguntas claves.**



En comercio electrónico, esto implicaría, por ejemplo, una venta perdida. Y en cualquier caso, la misma utilidad de la aplicación es inversamente proporcional al número de personas que abandonan el sitio.

No sólo puede perder utilidad, incluso puede ser **perjudicial** para la imagen de la compañía, el organismo o institución, si genera frustración en el usuario.

En cuanto a la accesibilidad, cualquier aplicación debiera tener en cuenta las diferentes discapacidades de algunos usuarios para no ser discriminativa. Por ejemplo, un mal contraste entre un texto y el fondo puede dificultar a un amplio sector de la población con dificultades de visión.

Además, hoy en día, dada la diversificación de medios para acceder a diferentes aplicaciones (equipos

de escritorio, móviles, tablets, etc) la accesibilidad ha cobrado aún mayor importancia.

Respecto a la **Intranet**, la accesibilidad y usabilidad afectan directamente a la **productividad** de sus usuarios y a su grado de satisfacción.

1.4 Redacción

Se debe emplear un lenguaje **amigable, claro, y conciso**.

A pesar de la naturaleza jurídica del Consejo, no se recomienda el uso de **tecnicismos** si no es imprescindible, especialmente en aquellas webs públicas cuyo público sea tanto jurídico como de interés general.

El trato dispensado en el sitio web debe ser el mismo en todos los canales de contacto de la Marca. Para ser coherentes con la mayoría de los contenidos, es recomendable el tuteo, con la excepción de los textos de contenido legal. En el caso de elementos de formularios, por ejemplo botones, se empleará preferentemente un lenguaje más impersonal, mediante el uso de infinitivos, para evitar en la medida de lo posible un tono imperativo.

Utiliza los servicios telemáticos que te presta tu Colegio a través de RedAbogacía

RedAbogacía es la infraestructura tecnológica de la Abogacía Española. Sus aplicaciones están diseñadas para el ejercicio de la profesión, contribuyendo a hacer más eficiente el trabajo diario y ahorrar costes y tiempo.

■ ACCESOS DIRECTOS REDABOGACÍA

- CENSO
- CORREO
- PASES A PRISIÓN

■ Servicios CON certificado ACA

A través de la Red de Servicios de la Abogacía podrá realizar de forma telemática gestiones profesionales de forma rápida y eficaz con su carné ACA

- PASES A PRISIÓN
- BUROSMS
- BUROMAIL
- PREVENCIÓN DE BLANQUEO DE CAPITAL
- LEXNET
- CIP
- OFICINA POSTAL VIRTUAL
- E-MENSAJES
- MI CUENTA
- REDABOGACÍA LIBROS
- ABOGADO EUROPEO

PUBLICIDAD

TAMBIÉN DESDE TU ORDENADOR

Las **abreviaturas** no son recomendables. De cualquier modo, en el caso de su utilización, deben tener un tooltip explicativo.

Se utilizará en los títulos el uso especificado por las reglas de la gramática, la primera palabra en mayúsculas y las restantes en minúsculas. Esta regla debe aplicarse de manera genérica en la redacción de contenidos.

Se establecen las excepciones para el siguiente listado de casos, en los cuales la primera letra irá en mayúsculas exceptuando las preposiciones y artículos (de, para, con, el, etc.):

- Nombres de instituciones.

- Nombres de Servicios, por ejemplo Justicia Gratuita, Ventanilla Única.
- Nombres de leyes, estatutos, reglamentos, por ejemplo: Ley de Tasas Judiciales.
- Eventos, por ejemplo: XV Encuentros Penitenciarios.
- Términos de uso frecuente en los que exista una justificación. Por ejemplo: Certificado Digital, debería ponerse siempre con la primera letra de cada palabra en mayúsculas para dar énfasis a un concepto importante.

1.5 Diseño. Consideraciones generales.

Combinar creatividad y usabilidad eligiendo siempre la función sobre la forma.

Estructurar el contenido es de gran importancia para la claridad y organización del sitio y de cada una de las pantallas.

De esta manera se facilita la comprensión y el aprendizaje al usuario y es una ayuda para lograr la accesibilidad del sitio.

Para lograr dicha estructuración debe haber una clara jerarquización por medio de módulos, títulos y subtítulos.

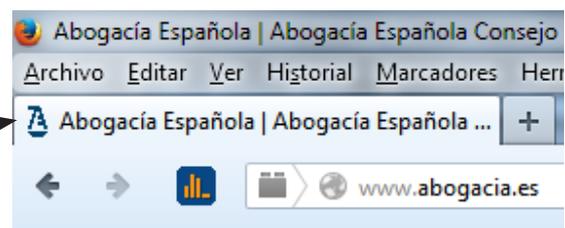
Los contenidos deben dividirse por afinidad de datos, y en el caso de recogida de datos mediante formularios, si fueran muchos, se trataría de dividir el proceso en **pasos**. Dichos pasos estarán **representados gráficamente** para que el usuario sepa en todo momento en qué paso está, y **cuánto** le queda **para finalizar el objetivo**.

El usuario escanea con los ojos la pantalla **de izquierda a derecha** y de arriba a abajo, por lo que es preferible colocar lo de **mayor importancia la izquierda y arriba**.

El funcionamiento que todo usuario espera cuando pincha sobre el **logotipo**, es que sea un **enlace que le lleve a la página principal** de la web.

Debe evitarse un diseño sobrecargado, con exceso de imágenes, elementos en movimiento, y sin espacios en blanco donde el usuario pueda descansar la vista. Una mala práctica propia de muchos portales es querer ofrecer demasiada información, saturando al usuario desde la misma página de inicio, lo cual es negativo para la experiencia del usuario, que se siente asaltado por una sobredosis de información y estímulos visuales agresivos.

Toda aplicación, en especial las públicas, debe tener su propio **Favicon** (el pequeño icono asociado a determinada página web)



1.6 Navegación

- **El usuario no debe sentirse perdido en ningún momento**, siendo esta una de las pautas imprescindibles para la usabilidad de los diferentes sitios dependientes del Consejo.
- Se utilizarán los elementos gráficos, colores, y títulos para que el usuario sepa en todo momento dónde se encuentra.
- Es recomendable el uso de **migas de pan** (“breadcrumbs”) en el caso de que el usuario se halle en aplicaciones que cuenten con una cierta profundidad en el árbol de navegación.



- Siempre debe estar presente un **enlace a la página de Inicio** de la web. Independientemente de ello, al pulsar sobre el logotipo, situado en la parte superior izquierda de la web, también se dirigirá al usuario a la página de inicio. Se utilizará la etiqueta html **title**, para texto y/o imagen para indicar esta funcionalidad.
 - El usuario debe poder **acceder a cualquier apartado de la web desde cualquier pantalla**, a través del sistema de navegación.
- Toda aplicación debe contar con una sección de **mapa web**. Esto, aparte de ser positivo para la usabilidad de un sitio, es una buena práctica para el posicionamiento natural en buscadores (SEO).
 - Es recomendable el uso de un botón de **“Volver”** en los casos en los que esté justificado por necesidades de la interfaz de usuario.
 - Cuando exista un proceso que conlleve un tiempo considerable al usuario, dividir el proceso en **pasos lógicos**, indicando gráficamente en qué paso se encuentra el usuario en todo momento, y dándole la posibilidad de avanzar o retroceder en dicho proceso mediante botones, sin que al pasar de una pantalla a otra pierda los datos ya introducidos.
 - El usuario debe estar adecuadamente informado. Ante una decisión crítica, se le debería informar de manera adecuada sobre la repercusión de su acción. Por ejemplo, con un botón de eliminar que suponga la pérdida de datos, sería recomendable que apareciera un **mensaje de aviso/confirmación**. No obstante, en el caso de que esta acción pueda ser utilizada frecuentemente, debería permitir al usuario la opción de ocultar dicho aviso/confirmación para agilizar el proceso y evitar la frustración del usuario. Por ejemplo, un check con el texto “No volver a mostrar este mensaje.”
 - Los **mensajes emergentes** son útiles para mostrar información relevante, pero no es conveniente abusar de ellos. Suponen una interrupción en el proceso que está ejecutando el usuario, lo que ralentiza el proceso y puede generar frustración. Debe sopesarse en cada caso si está justificada su utilización. En el caso de que sea una acción que pueda ejecutarse con frecuencia, permitir al usuario la opción de ocultar dicho aviso, como en el ejemplo visto en el punto anterior. No entran en esta categoría los mensajes informativos sobre el estado del sistema, como por ejemplo una barra de progreso.

1.7 Configuración

Una aplicación que se precie debe cumplir con la máxima: **“decide por mí, pero déjame tener la última palabra.”** En ese sentido, siempre es interesante plantearse qué opciones pueden resultar útiles en usuarios avanzados, para que puedan personalizar la aplicación y que se adapte lo mejor posible a las características y necesidades de cada usuario.

En este sentido, también hay que ser cuidadoso y no crear aplicaciones con un grado tan grande de configuración que finalmente la haga excesivamente compleja y poco intuitiva. Siempre hay que preguntarse: ¿es realmente útil? ¿el usuario ha manifestado su interés en la implementación de dicha utilidad? ¿Dificulta la escalabilidad y mantenimiento de una aplicación? ¿Es más costoso que útil?



A esta configuración se la suele llamar de diferentes maneras: opciones, personalización, preferencias, configuración... Para seguir una coherencia semántica en las diferentes aplicaciones del Consejo, a esta pantalla se la denominará: **Configuración...**

A la sección de configuración se le debe dar **poca importancia jerárquica**, ya que es una opción a la que se accederá con poca frecuencia. Además, su posición está más o menos estandarizada. El lugar habitual en -por ejemplo- un menú, es **inmediatamente antes de la ayuda**.

Es recomendable en aplicaciones con un alto grado de opciones configurables, una **acción que permita devolver la aplicación a su estado original por defecto**. Esta opción confiere a los usuarios una sensación de seguridad y anima a experimentar.

1.8 Ayuda

En el caso de la ayuda, es especialmente importante que sea clara, concisa, utilice el mismo lenguaje que los usuarios y evite tecnicismos. Se debe empatizar con el usuario y no dar por hecho que los usuarios “ya saben”. Una aplicación ideal desde el punto de vista de la usabilidad, no necesitaría ayuda, ya que todo estaría de por sí lo suficientemente claro, pero no debería faltar si conlleva cierto grado de complejidad.



A la hora de desarrollar una ayuda efectiva, es importante seguir las siguientes recomendaciones:

- Que disponga de un **buscador**. Y obviamente, que el buscador cumpla su funcionalidad. Poner un buscador que haga las búsquedas directamente en google, no es resulta útil. El buscador de un portal debe buscar sólo y exclusivamente en los contenidos del mismo.
- Que disponga de un apartado de **preguntas frecuentes**, cuyos contenidos deberían ser el resultado de la experiencia con las dificultades más comunes que se encuentran los usuarios.

- **Una imagen vale más que mil palabras.** Decantarse por explicaciones con fotografías, gráficos... en lugar de largas explicaciones con textos.
- Los usuarios no leen: escanean la pantalla en busca de un indicio de información que resuelva rápidamente su problema. Si no lo encuentran rápidamente es probable que se marchen. Hay que ayudarles con diseños donde **los textos respiren: existan suficientes espacios en blanco, ayudados por listas, una jerarquización clara y gráficos y/o fotos que faciliten la búsqueda.**
- Preferiblemente **que la ayuda sea contextual**, es decir, explicar sobre la misma pantalla el uso de los elementos de dicha pantalla.
- El recurso de mostrar una pantalla de **ayuda al inicio de una aplicación debe evitarse** a no ser que se juzgue realmente necesario, y **en cualquier caso debe facilitarse al usuario de una manera muy rápida y clara el salir** de dicha ayuda

Se establecen 4 tipos posibles de ayuda:

- Ayuda **general** de la aplicación.
- Ayuda **contextual** sobre la pantalla actual.
- Ayuda **sobre un elemento** concreto de formulario.
- Ayuda **sobre un grupo** lógico de formularios.

La iconografía utilizada cambiará según el tipo de ayuda:



Ayuda **general** de la aplicación



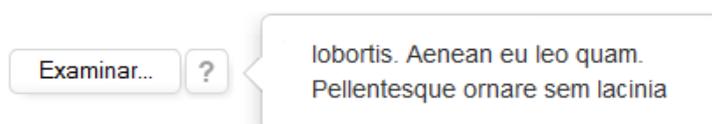
Ayuda **contextual** sobre la pantalla actual



Ayuda sobre un **grupo lógico de formularios**

[Ver ejemplo: Iconografía de Ayuda](#)

Ayuda sobre un elemento concreto:



En formularios, la ayuda sobre un elemento concreto, se utilizará **exclusivamente en los campos**

que realmente puedan ofrecer dudas sobre la información que se esté solicitando.

La ventana emergente o popup se hace visible al colocar el puntero sobre el icono 

[Ver ejemplo: Ayuda sobre elemento](#)

1.9 Publicidad

1.9.1 Consideraciones generales

Debe evitarse un **aspecto excesivamente publicitario** en las webs públicas: los usuarios tienden a evitar instintivamente este tipo de información, incluso en ocasiones si la web tiene un aspecto sobrecargado de publicidad, ante un primer vistazo, puede salir de la página y seguir buscando la información en otra.



En este sentido, por ejemplo, los **contenidos publicitarios invasivos** restan interés y dan una impresión de baja calidad a la página y sus contenidos informativos. Un ejemplo de publicidad invasiva sería una ventana que oculta el contenido de la web durante un tiempo determinado impidiendo ver su contenido.

En el caso particular del Consejo esto es especialmente importante, siendo recomendable fomentar una **imagen de entidad oficial**, en detrimento de un aspecto excesivamente publicitario.

1.9.2 Banners

Claves visuales para que un banner resulte efectivo:

- Que los contenidos de la publicidad estén relacionados con los contenidos del sitio.
- Que se identifique que es publicidad con un subtítulo.
- Un banner siempre tiene que sorprender y despertar la curiosidad del usuario.
- Evitar una animación excesivamente rápida. Tiene que permitir una lectura pausada del contenido, teniendo siempre en cuenta, que cada fotograma del banner debe ser tratado de forma independiente. De tal manera que si cada uno de esos fotogramas pudiera ser impreso, sirviera como publicidad independiente.
- El peso tiene que estar perfectamente optimizado, para que el tiempo de descarga sea mínimo, se aconseja que no exceda de 10 a 20kb. Para ello no tienen que contener demasiadas escenas.
- Diseñar un banner integrado al diseño de la página, que se identifique como publicidad

pero sin salirse de la línea de diseño.

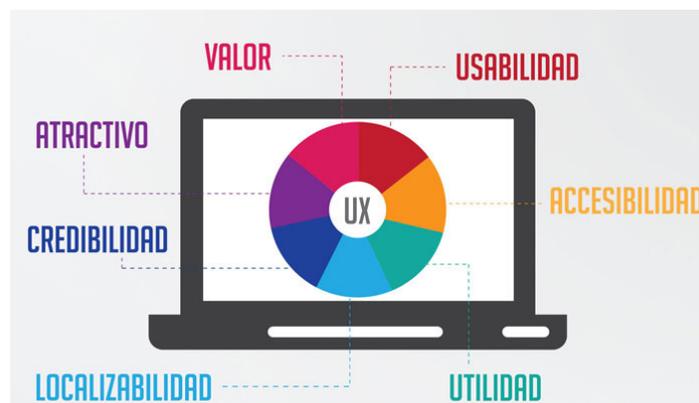


- El mensaje debe ser fijado **desde el primer fotograma del banner**, ya que el tiempo que el usuario puede dedicar a nuestro banner es apenas unos segundos.
- Es recomendable que todos los banners de empresas externas, lleven el logotipo, o la dirección web, o una identificación gráfica de nuestra página, producto o empresa, en todos los fotogramas.
- No es recomendable que un banner reproduzca sonido si no es con el consentimiento previo del usuario, mediante el control correspondiente: botón de audio, play...

1.10 Heurísticas de Usabilidad de Nielsen

Jacob Nielsen, autoridad reconocida en el campo de la usabilidad, propone -entre otros- **diez principios que una web debería cumplir**:

1. **Visibilidad del estado del sistema.** El sistema debe informar a los usuarios del estado del sistema, dando una retroalimentación apropiada en un tiempo razonable (por ejemplo, si el sistema está realizando un proceso que pueda tardar cierto tiempo, debe informar al usuario).
2. **Utilizar el lenguaje de los usuarios.** El sistema debe utilizar un lenguaje al que los usuarios estén habituados, con palabras o frases que le sean conocidas, en lugar de los términos que se utilizan en el sistema, para que al usuario no se le dificulte utilizar el sistema.



3. **Control y libertad para el usuario.** En casos en los que los usuarios elijan una opción del sistema por error, éste debe contar con las opciones de deshacer y rehacer para proveer al usuario de una salida fácil sin tener que utilizar diálogo extendido.
4. **Consistencia y estándares.** El usuario debe seguir las normas y convenciones de la plataforma sobre la que está implementando el sistema, para que no se tenga que preguntar el significado de las palabras, situaciones o acciones del sistema.
5. **Prevención de errores.** Es más importante prevenir la aparición de errores que generar buenos mensajes de error. Hay que eliminar acciones predisuestas al error o, en todo caso, localizarlas y preguntar al usuario si está seguro de realizarlas.



6. **Minimizar la carga de la memoria del usuario.** El sistema debe minimizar la información que el usuario debe recordar mostrándola a través de objetos, acciones u opciones. El usuario no tiene por qué recordar la información que recibió anteriormente. Las instrucciones para el uso del sistema deberían ser visibles o estar al alcance del usuario cuando se requieran.
7. **Flexibilidad y eficiencia de uso.** Los aceleradores permiten aumentar la velocidad de interacción para el usuario experto de tal manera que el sistema pueda atraer a usuarios principiantes y experimentados. Es importante que el sistema permita personalizar acciones frecuentes para así acelerar el uso de éste.
8. **Diálogos estéticos y diseño minimalista.** La interfaz no debe contener información que no sea relevante o se utilice raramente, pues cada unidad adicional de información en un diálogo compete con las unidades relevantes de la información y disminuye su visibilidad relativa.
9. **Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores.** Los mensajes de error deben expresarse en un lenguaje claro, indicar exactamente el problema y ser constructivos.

10. **Ayuda y documentación.** A pesar de que es mejor un sistema que no necesite documentación, por su facilidad de uso, puede ser necesario disponer de ésta. Si así es, la documentación tiene que ser fácil de encontrar, estar centrada en las tareas del usuario, tener información de las etapas a realizar y no ser muy extensa.

1.11 Maquetación

Debe evitarse el uso de **frames o iframes, siempre que sea posible.**

Evitar las tablas para otro uso que no sea el de mostrar información de módulos tabulares: no utilizarlas para maquetar sobre el sitio. Esto es especialmente importante en la parte pública.



Debe buscarse la mayor **rapidez de carga** de las diferentes webs utilizando todos los medios posibles para este fin. La rapidez de carga facilita la usabilidad, accesibilidad y posicionamiento natural en buscadores (SEO).

El **código** debe ser lo más **limpio** posible. Debe evitarse el uso de css fuera de los archivos .css, y los scripts de javascript fuera de los archivos .js, siempre que sea posible.

La limpieza en el código:

- Facilita la indexación de las páginas en los buscadores, (SEO).
- Disminuye el tiempo de carga, pues al haber funcionalidades en archivos fuera de la propia página, pueden ser utilizadas por otras páginas sin necesidad de repetirlas.
- Facilita el trabajo de cara a otras personas que tengan que trabajar sobre el código.
- Mantiene la escalabilidad en cualquier proyecto. Por ejemplo, modificando unas pocas líneas en una hoja de estilo, se puede modificar el aspecto de todo un portal en muy poco tiempo.

Es recomendable el uso de **comentarios** para facilitar la comprensión de las diferentes funcionalidades que conlleva el código.

En **bloques de texto** se recomiendan líneas de entre 60 y 80 cpl (caracteres por línea), no siendo recomendable más de 100 cpl. Líneas más largas intimidan al usuario, y dificultan la búsqueda del siguiente renglón para continuar la lectura.

Por otra parte, tampoco es efectivo el que el ancho de las líneas sea excesivamente corto, ya que también dificulta la legibilidad y cansa al usuario, al forzar la vista a realizar frecuentes saltos de línea.

Debe **evitarse el justificado del texto** en la web por las siguientes razones:

- A algunos usuarios con discapacidad les resulta más difícil la lectura de textos justificados en pantalla.
- El espaciado entre las palabras puede afectar negativamente a la estética del sitio.

Usar listas. Las listas son más atractivas, “enganchan” más al usuario, que un párrafo extenso. Si se puede convertir un párrafo largo en una lista, es recomendable hacerlo. Por otra parte siempre que sea prescindible, debe evitarse el uso de listas con listas anidadas en su interior, ya que provocan el efecto contrario.

1.12 Velocidad de carga

Especialmente importante en los dispositivos móviles, debido a la **inestabilidad y limitación** de internet desde los dispositivos en algunos contextos. Aparte de que el consumir ancho de banda puede traducirse en un mayor **desembolso económico** y los usuarios son conscientes de ello.

Para optimizar la carga de una página deben contemplarse los siguientes aspectos:

- **Optimizar imágenes** utilizando la máxima compresión con la mínima pérdida de calidad, utilizando los formatos adecuados: png, jpg (ver apartado 2.1. *Imágenes*).
- **Evitar imágenes de gran tamaño.**
- Utilización de **sprites** siempre que sea posible para la carga de imágenes de pequeño tamaño (por ejemplo iconos) para minimizar el número de peticiones al servidor, optimizando así el tiempo de carga.
- Debe **evitarse el abuso del contenido multimedia**. En ningún caso los contenidos multimedia como vídeo o sonido deben reproducirse automáticamente, dejando en cualquier caso la opción de reproducirse a criterio del usuario.



- **Comprimir los archivos** javascript y css, y evitar en la medida de lo posible utilizar varios. Es mejor unificarlos, siempre y cuando los archivos resultantes no sean excesivamente grandes y contengan excesivo código que la web no esté realmente utilizando (para más

información, ver el apartado: 1.13. Organización de los datos, tecnologías y nomenclatura).

- **Simplicidad y calidad de código.** A mayor simplicidad, mejor tiempo de carga. Una mala programación puede derivar en un exceso de consultas a bases de datos, lo que ralentiza el tiempo de carga.
- El **hosting** también influye, a la hora de contratar dicho servicio, hay que tener en consideración su **calidad** para los tiempos de carga.
- **Evitar el uso de tablas** para maquetar una página, ya que incrementan el tiempo de carga.
- Aprovechar la característica de **almacenamiento en caché.** Consiste en crear versiones estáticas de las distintas páginas que se van renderizando de forma dinámica. El objetivo de este sistema es reducir el número de llamadas que se hacen al servidor, agilizando de esta forma la carga de la página.
- Economización y simplificación de divs. Es decir, evitar crear capas innecesarias; simplificar siempre que sea posible una capa dentro de una capa, que está dentro de otra capa, que está dentro de otra...
- Utilización de **cargas asíncronas.** Con el uso de cargas en segundo plano se mejora notablemente la sensación de velocidad de carga en una aplicación web o una app. Un ejemplo práctico lo tenemos en Instagram. Si escribes un comentario en una foto inmediatamente sale como agregado haciendo todo el proceso transparente para el usuario. Sólo si el proceso falla se informa al usuario de que ha habido un problema y se quita el comentario de la lista de los mismos.

1.13 Resolución y Diseño adaptable

Se utilizarán técnicas de **diseño web adaptable (responsive web design)** Esto será aplicable siempre al portal público y en las secciones del portal privado. Se tendrán en cuenta su visualización correcta en el mayor número de resoluciones de pantalla posibles. El que un sitio web sea multidispositivo, también es beneficioso para el posicionamiento natural en buscadores (SEO).



Para resoluciones típicas en equipos de escritorio, es recomendable una medida mínima de 1024px optimizado para más del 90% de resoluciones de pantalla (según estadísticas Enero 2013). Sin detrimento de que en pantallas más pequeñas dicha resolución se ajuste, o se estime conveniente en determinados casos aumentar dicha medida en pantallas más grandes.

Así mismo se ocultarán **elementos prescindibles** para no saturar las **pantallas de móviles**, y centrar al usuario en los elementos principales. Un ejemplo de un elemento prescindible sería una imagen decorativa, un campo de formulario no obligatorio, o un anuncio invasivo. En **cada aplicación** de la intranet **se estudiará qué elementos pueden ser obviados sin restar funcionalidad.**

Debe estudiarse detenidamente lo prescindible de dichos elementos, para no caer en el error común de ofrecer una versión excesivamente simplificada que deje insatisfecho al usuario.



El uso de las medidas será preferiblemente porcentual, para facilitar el diseño fácilmente a diferentes resoluciones, sin detrimento de utilizar medidas absolutas en los casos en los que para la maquetación resulte necesario. Por ejemplo, un submenú desplegable en un menú horizontal, es habitual que tenga un posicionamiento absoluto para evitar que “desplace” al resto de contenidos de una página.

Este tipo de maquetación se realizará en combinación con las herramientas que css3 facilita al respecto, por ejemplo, **media-query**, **max-width** y **min-width**.

Para más información sobre responsive web design, ver el capítulo 3 - *Dispositivos móviles*.

1.14 Soporte en los principales navegadores

Es fundamental que las páginas se visualicen correctamente y sus funcionalidades sean compatibles con el mayor número posible de navegadores posibles.

Como mínimo esto debería aplicarse a los principales navegadores:

- **Internet Explorer (desde la versión 9)**
- **Chrome**
- **Mozilla Firefox**
- **Safari**

Es muy recomendable que en versiones más antiguas de dichos navegadores, las webs del Consejo sean, como mínimo legibles. Para ello se pueden utilizar las herramientas existentes para lograr compatibilidad entre nuevas tecnologías, como HTML5 y CSS3 en dichos navegadores. Por ejemplo usando scripts que permitan en navegadores más antiguos al Internet Explorer 9, la lectura de las nuevas etiquetas HTML5.



Sobre el uso de dichos navegadores, existen discrepancias en los porcentajes de uso, pero de lo que

no cabe duda es que los tres mencionados son los principales, y con una frecuencia de uso que se debería tener siempre en cuenta.

Enlaces con estadísticas sobre el uso de los navegadores:

<http://percentoftheinternet.com>

<http://gs.statcounter.com/>

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

1.15 Organización de los datos, tecnologías y nomenclatura

Páginas renderizadas en **html5**, y **CSS3**, aprovechando la versatilidad de **jquery**, independientemente de las tecnologías escogidas por ser las más apropiadas para los diferentes proyectos del Consejo.

Deben utilizarse **.css** y **.js** generales, para todas las páginas que compartan o vayan a compartir estilos y/o scripts.

Pueden utilizarse varios **.css** y **.js** generales, por ejemplo, "general.css" para todas las páginas con estilos comunes. Aparte de los archivos generales, cada página debería tener su propio **.css** y si fuera necesario, su propio **.js**. El nombre de los archivos de cada página específica, debe coincidir con el nombre de la página de la que depende.

Todo esto es aplicable para cualquier tipo de tecnología que se vaya a utilizar en varias páginas.

Habrá que comprobar siempre cómo ha afectado la modificación de los archivos de los que dependen varias páginas, sobre cada una de ellas.



Las hojas de estilo que se utilizan en las páginas van a dar forma tanto a archivos html como a jsp, o en su caso, php. Es recomendable que todas ellas lleven el mismo nombre que el html, jsp o php al que se asocia.

Además, en caso de que sea necesario, los html, jsp y php se vincularán, además de a la propia, a hojas de estilo de carácter general, como general.css, tratando siempre de **evitar la duplicidad de código** a cargar.

La sintaxis para asociar las hojas de estilo se basan en las clases y en los identificativos (**class** e **id**):

```
<div id="nombreID" class="nombreClase"></div>
```

La utilidad de las clases consiste en que al aplicarlas en una capa (**<div></div>**) o etiqueta (p ej., **<p></p>**) se le asocian determinados estilos gráficos. Asignando la misma clase a otra capa se repiten dichos estilos, ahorrando código y facilitando posteriores cambios gráficos a varias capas y elementos modificando solamente las hojas de estilo. Esta es una de las razones por las que es **desaconsejable aplicar estilos sobre la misma web**, en vez de sobre las hojas de estilo correspondientes, a no ser que sea imprescindible.

Un identificativo (**id**) se utiliza para aplicar estilos gráficos exclusivos para una capa o etiqueta. **No se debe repetir un mismo id en un mismo documento** html, jsp, etc.

Aparte del aspecto gráfico, tanto las clases como los ids permiten interactuar con diferentes lenguajes de programación, como por ejemplo javascript.

El **nombre de los archivos**, ya sean el nombre del html, jsp, imágenes, pdf, etc, es importante para el **SEO** (posicionamiento natural en buscadores). Es recomendable que los archivos tengan nombres representativos de los contenidos que albergan. Así pues, se puede conseguir **un mejor posicionamiento** si se nombra a un archivo pdf "ley_organica_poder_judicial_propuesta_reforma.pdf" que si a este mismo archivo se le nombra como "ley001". Las palabras en el nombre del archivo deben ir separadas por guion bajo o guion medio para ser reconocibles por los motores de búsqueda.

También es muy recomendable el **uso correcto de las etiquetas HTML5** para distribuir los contenidos de **manera semántica**, lo cual también es importante para SEO.

Estas consideraciones son aplicables, excepto en los casos en que por necesidades del proyecto se utilicen frameworks que tengan su propia manera de organizar los archivos.

1.16 Testeo

Es importante recopilar información sobre el uso de una aplicación para saber qué se puede mejorar de cara a la experiencia de usuario. Existen diferentes modos de recopilar dicha información:

- Testeo con usuarios en vivo, observando el tiempo invertido en la consecución de determinados objetivos, medición de errores, etc. Puede incluir dispositivos que crean mapas de calor en los puntos donde los usuarios fijan la vista, donde clican, etc. Cuanto mayor sea el número de usuarios sobre los que se haga la prueba, mayor fiabilidad se tendrá sobre los resultados. Este tipo de evaluación suele ser costosa.



- Mantener un contacto fluido con los usuarios para escuchar su opinión sobre la facilidad o dificultad a la hora de manejar las aplicaciones, su grado de satisfacción, etc.
- Utilización de herramientas analíticas para comprobar cómo se comportan los usuarios en las aplicaciones, qué secciones son las más visitadas, uso de determinadas acciones, estadísticas de utilización, dispositivos utilizados, etc. Algunas de las herramientas online disponibles:

<http://www.google.com/analytics/>

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

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

<http://clicky.com/>

<http://www.w3counter.com/>

- Si la aplicación cuenta con un equipo de Soporte (lo cual siempre es recomendable), solicitarles información sobre cuáles son los puntos que más errores y dudas generan en los usuarios de determinada aplicación.

2. Elementos Principales

2.1 Consideraciones tipográficas

Familia Tipográfica

En las aplicaciones sobre soportes digitales en los Portales Públicos la tipografía a utilizar será la siguiente:

Escrito de la Abogacía para que los afectados paralicen el desahucio

Requiere al juzgado, con fundamentos jurídicos, la paralización de la ejecución hipotecaria tras la sentencia del TJUE sobre cláusulas abusivas

Open Sans Condensed en sus variantes Light y bold para entradillas, encabezados, titulares, sumarios y destacados.

Ubuntu para el texto genérico, esta última especialmente recomendable para cuerpos reducidos.

ACTUALIDAD

- ABOGACÍA**
XV
Cáritas y Banco de Alimentos, Jazmín Barrios, Mónica Bernabé y Enrique Martínez Reguera, Premios DDHH
- ABOGACÍA**
12-J: El Día de la Justicia Gratuita y el Turno de Oficio, celebrado en todos los Colegios de Abogados
- ABOGACÍA**
12-J. José Ramón Felipe: Acogió en su casa a una pareja a la que defiende de oficio
- ABOGACÍA**
12-J. César Pinto, primer abogado (de Oficio) que consiguió que el TEDH paralizara un desalojo
- ABOGACÍA**
Datos reales de la Justicia: los asuntos tramitados ante los órganos jurisdiccionales son 1.833.608
- ACTUALIDAD JUSTICIA GRATUITA**
La inversión en Justicia Gratuita disminuye por tercer año consecutivo, un 9% en 2012
- ABOGACÍA**
12J: Paulino Pérez Riveiro, defensor de oficio del jefe de máquinas del Prestige
- ABOGACÍA**
Aumenta el apoyo de Administraciones Públicas a los Colegios de Partido
Los consideran imprescindibles por el arraigo en sus localidades y el gran servicio que prestan a los ciudadanos

Se recomienda para el texto genérico de las páginas que utilicen la tipografía **Ubuntu** una apariencia gráfica de 14px y un interlineado de 20px para una legibilidad óptima.

[Ver ejemplo: Tipografía para soportes digitales en Portales Públicos](#)

Para los textos pertenecientes a las aplicaciones de la **Intranet** y para **dispositivos móviles con pantallas de tamaño inferior, se utilizarán las fuentes propias del sistema:**

Arial o una similar (Sans- Serif) si no estuviera disponible. Para el texto genérico se recomienda un tamaño en apariencia de 14px y un interlineado de 20 px. Independientemente de la apariencia, se recomienda utilizar para lograr dicha apariencia medidas porcentuales en el texto.

Al mismo tiempo se aplicará en la hoja de estilos asociada el siguiente valor:

CSS:

```
body {  
  font-size=100%;  
}
```

De este modo, **al modificar tan sólo la medida 100% del body, se modifica el tamaño de todas la tipografías de un proyecto**, debido a su carácter porcentual, optimizando cualquier cambio futuro y facilitando los cambios para el diseño web adaptable (Responsive web design).

En cualquier caso, cualquier texto debe ser lo suficientemente grande para que pueda ser leído con facilidad en pantalla. No debería ser menor a 11 px (apariencia gráfica) excepto en casos realmente justificables.

CSS:

```
font-family: Arial, sans-serif;
```

[Ver ejemplo: Texto Genérico para Intranet.](#)

En general, para cualquier aplicación los textos que se vayan a visualizar en pantalla utilizar siempre una tipografía **Sans Serif**, ya que son más fácilmente legibles en pantalla.

La principal diferencia entre las letras Serif y Sans Serif es que las Serif tienen detalles adicionales en los bordes de los contornos de las fuentes:



Serif



Sans Serif

El **uso de negrita** se reservará para los **datos que sean realmente importantes** para resaltar. Se usará preferiblemente el uso de la etiqueta html **** siempre que los términos a resaltar se consideren de importancia a la hora de considerar el posicionamiento natural en buscadores (SEO)

El **uso de cursiva será limitado** por ser más ilegible en pantalla. Es recomendable utilizarlo con poca frecuencia, y hacerlo para dar énfasis o en definiciones. Para citas, es más recomendable el uso de comillas “ ”.

Evitar el uso continuado de mayúsculas en textos largos, aparte de la connotación de “gritar” al que está asociado, también disminuye la legibilidad en pantalla.

Es importante no utilizar el subrayado en los textos, excepto en los enlaces que lo requieran, para evitar confusiones pues el subrayado está asociado a los enlaces (a otras páginas webs externas, direcciones de correo electrónico).

Para la representación de números con más de tres dígitos usar el punto (1.000 y no 1000).

Texto como Imagen

No se deben utilizar imágenes para representar textos salvo casos de necesidad, como por ejemplo, imagotipos. Esto es importante por las siguientes razones:

- Aumento del tiempo de carga de la página.
- Los buscadores no indexan este texto, lo que afecta a su posicionamiento natural en buscadores (SEO).
- Los usuarios no pueden aumentar el texto si tienen problemas de legibilidad.
- El texto en la imagen no puede ser seleccionado, lo que causa una reacción del sitio no esperada por un usuario poco experto.
- Algunos navegadores especiales para personas con problemas de visión pueden leer el texto pero no las imágenes, aún más si estas no tienen un **alt** bien definido. De cualquier modo no todos los lectores de pantalla son capaces de interpretar dicho **alt**.

2.2 Colores

Principal



R: 0
G: 67
B: 117
#004375

Secundario



R: 142
G: 163
B: 193
#8da3c1

Variaciones



R: 178
G: 198
B: 213
#b1c6d4



R: 203
G: 216
B: 226
#cbd8e2



R: 230
G: 236
B: 242
#e6ecf1

Grises



R: 178
G: 198
B: 213
#b1c6d4



R: 123
G: 124
B: 126
#7b7c7e



R: 230
G: 236
B: 242
#a7b8aa



R: 203
G: 216
B: 226
#c6c7c8



R: 230
G: 236
B: 242
#d9dad9

Complementarios



R: 197
G: 70
B: 19
#c54613



R: 147
G: 53
B: 136
#933588



R: 246
G: 175
B: 129
#f6af81



R: 170
G: 180
B: 42
#aab42a

Entre el color de fondo y el texto debe haber el suficiente **contraste** para que este resulte legible por el mayor número de personas posible y en diferentes situaciones:

- Pantallas con diferentes ajustes de brillo y contraste
- Personas con problemas de visibilidad
- Situaciones en las que la luz natural provoque brillo sobre la pantalla, dificultando la visión. Esto es especialmente importante en dispositivos móviles.

En un futuro sería conveniente disponer de una utilidad que permita al usuario elegir una versión de las aplicaciones web en **alto contraste**, para tener una correcta accesibilidad.

Sería posible, adjuntando, por ejemplo, una hoja de estilos que permita visualizar al usuario una versión con colores de alto contraste. Dicha hoja de estilo se cargaría en el caso de que el usuario eligiera dicha opción.

Una herramienta interesante para comprobar el grado de contraste entre el color del fondo y el color del texto, y el grado de cumplimiento con la normativa de la WCAG es Colour Contrast Analyser.

<http://files.paciellogroup.com/resources/CCA-2.2.zip>

2.3 Títulos y subtítulos

Son elementos con formato de texto que ordenan y separan el contenido para facilitar la lectura, y estructurar la información.

Para los títulos siempre se utilizarán la etiquetas html: `<h1></h1>`, `<h2></h2>`, `<h3></h3>`, `<h4></h4>`, en función de la jerarquía.

En las diferentes secciones del documento se utilizarán las etiquetas respetando la jerarquización semántica que HTML5 ofrece. Ejemplo:

```
<header>
  <h1>El título más importante de la página</h1>
</header>
```

```
<article>
  <h1>El título del artículo</h1>
  <h2>Subtítulo del artículo</h2>
  <p>el cuerpo de texto</p>
```

```
</article>
```

```
<article>
```

```
<h1>El título del artículo</h1>  
<p>el cuerpo de texto al que hace referencia el título anterior</p>  
<h2>El subtítulo del artículo</h1>  
<p>el cuerpo de texto al que hace referencia el subtítulo anterior</p>  
</article>
```

En este ejemplo vemos como tenemos la etiqueta **header** que correspondería al encabezado de la página. En el mismo, iría el título más importante, el de la página.

Vemos también dos etiquetas **article**, las cuales son apropiadas para secciones de noticias, posts en un foro, entradas de blog. En las mismas se pueden utilizar de nuevo los encabezados h1 y h2 sin la penalización respecto al posicionamiento natural en buscadores (SEO) que pudiera suponer antiguamente el uso de varios h1, con palabras repetidas, en una misma página.

No está recomendado el uso de <hgroup>. Considerada una de las importantes innovaciones en HTML5, **ha sido recientemente (2013) declarada obsoleta**, pasando a no cumplir los estándares definidos por la [World Wide Web Consortium](#). De hecho, la [herramienta de validación online](#) del Consorcio, declara dicha etiqueta como tal.

Para las Webs públicas se utilizarán los títulos personalizados en función de las características de las mismas, teniendo siempre en consideración el punto anterior **2.1 Consideraciones tipográficas**.

Para la Intranet se contemplarán los siguientes casos de jerarquización siempre y cuando las características particulares de una aplicación no requieran un tratamiento especial:

Título de la Aplicación

Se utilizará la etiqueta dentro del <header> de la página, para priorizar el nombre de la aplicación en la que el usuario se encuentra en ese momento.

Etiqueta: **<h1></h1>**
Tamaño: **A la espera de tener la fuente**
Tipografía: **Pendiente de registrar**
Color: **#004375**

Título Jerarquía 1.

Etiqueta: **<h1> Título</h1>**
Tamaño: 26px
Tipografía: Arial
Color: Negro

Título Jerarquía 2

Etiqueta: **<h2>Título</h2>**
Tamaño: 22px
Tipografía: Arial
Color: #516283

Título Jerarquía 3

Etiqueta: `<h3>Titulo</h3>`

Tamaño: 18px

Tipografía: Arial

Color: #4A6AAA

Título Jerarquía 4

Etiqueta: `<h4>Titulo</h4>`

Tamaño: 14px

Tipografía: Arial

Color: #6380BC

Variante: Mayúsculas

[Ver ejemplo: Títulos y Subtítulos de la Intranet.](#)

2.4 Imágenes

Es **muy importante** que las imágenes estén bien **optimizadas** para que “pesen” lo menos posible.

Esto beneficia el tiempo de carga de las webs, e indirectamente beneficia al posicionamiento en buscadores, que contemplan como medida de calidad dicho tiempo.

Debe utilizarse el formato **png y jpg** en función del tipo de imagen. Deben ajustarse las características de las mismas según el tipo de imagen y su formato (número de colores, uso o no de la transparencia, calidad...). Para optimizar desde Photoshop, se puede acceder desde el menú **Archivo > Guardar para Web** (atajo de teclado **Ctrl+Alt+s**).

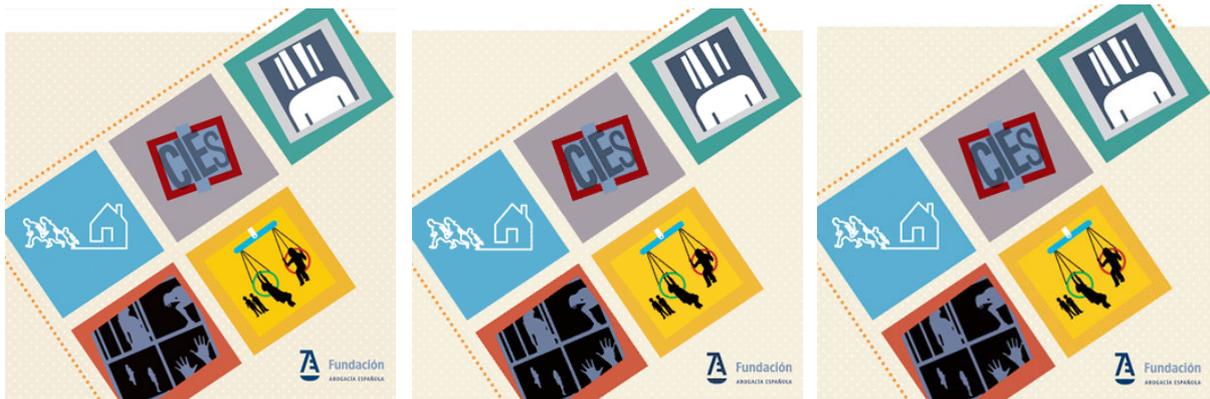


Imagen original. Tamaño 150 kb

Imagen optimizada en formato png. Tamaño 19 kb

Imagen optimizada en formato jpg. Tamaño 35 kb

En el ejemplo anterior se puede observar cómo una imagen original con un tamaño de 150 kb se puede reducir hasta 19Kb utilizando los valores óptimos para este tipo de imagen. Normalmente la decisión sobre utilizar gif, png o jpg vienen dadas por las diferentes criterios. Veamos algunos de ellos:

- **¿Gif estático o Png?** No hay grandes diferencias, si bien, el formato png, permite más opciones, y en las mismas condiciones de calidad, el png reduce el tamaño final de la imagen.
- **¿Gifs animados?** Si bien el formato gif contempla la posibilidad de estar animado, no es recomendable su uso, ya que existen tecnologías para realizar animaciones, consumiendo menos ancho de banda. Aparte de que suelen ser molestos, desvían la atención de los usuarios, y pueden ser nocivos en personas con epilepsia. Están prácticamente obsoletos excepto en contextos muy puntuales.
- **¿Png o Jpg?** Depende de las características de la imagen:
 1. **PNG.** Para imágenes con un número reducido de colores y tamaño medio o pequeño, como gráficos e iconos, el formato más adecuado es el **png-8**, reduciendo el número de colores al máximo sin que la imagen pierda calidad. En el ejemplo anterior, claramente sería la opción más recomendable. En imágenes que requieran transparencia de calidad, el formato más adecuado es **png-24**.
 2. **JPG.** Para imágenes con bastantes colores, como una fotografía, y de tamaño medio o grande.

Normalmente, la elección se toma al guardar la imagen, jugando con los valores: número de colores, calidad, etc, y comprobando cuál es la opción que menos degrada la calidad de la imagen original reduciendo al máximo su tamaño.

Es recomendable utilizar la técnica de **sprites**, si en la misma pantalla se están utilizando varias imágenes de pequeño tamaño, como por ejemplo iconos en la pantalla de un aplicativo de la intranet, para mejorar el tiempo de carga al disminuir las peticiones al servidor.

Un sprite consiste en una sólo mapa de bits en el cual se distribuyen las diferentes imágenes utilizadas en una determinada página. Utilizando las propiedades **height**, **width** y **background-position** se muestra la imagen que nos interesa.

Ejemplo de sprite:



El atributo **alt** se considera una **etiqueta obligatoria**, sirve para mostrar una descripción de la imagen en caso de que no esté disponible (por ejemplo en un navegador que tenga desactivadas las imágenes) También es necesaria por accesibilidad: los dispositivos lectores de pantalla, pueden leer la descripción del alt, por lo que este atributo es un sustituto textual de las imágenes de un sitio web.

Title es una descripción emergente (tooltip) que puede indicar una información útil al usuario. Si una imagen es un enlace debe llevar un title que indique donde lleva dicho enlace.

Es importante el uso de **medidas porcentuales** para **diseños adaptables**, mediante la utilización de **@media queries**.

CSS 3 nos ofrece la propiedad **max-width** para mantener el control en dispositivos de gran tamaño. Deben utilizarse medidas porcentuales que permitan adaptar las imágenes a resoluciones más pequeñas (**tablets**). Para **smartphones**, en el caso de imágenes grandes, deben cargarse **versiones de imágenes de tamaño inferior** para **optimizar la velocidad de carga** de la web.

Para iconos, es recomendable la utilización de imágenes vectoriales, dado que su peso es normalmente menor que el de un mapa de bits. Aparte, se pueden convertir varios iconos en una fuente personalizada, simplificando su utilización durante el desarrollo de la maquetación.

2.5 Iconografía

Se debe contemplar la coherencia gráfica en la iconografía de los diferentes aplicativos para evitar confusión en el usuario. Los iconos expuestos a continuación serán utilizados en relación al concepto situado bajo los mismos. No se debe utilizar, por ejemplo, el icono de lupa para otro contexto que no se corresponda con el concepto "buscar".

NOTA: la siguiente iconografía es genérica y monocroma. Sólo contempla forma y asociación. Respecto al tamaño, color y otras especificaciones, se atenderán a las características de cada proyecto.



Inicio



Zona segura



Cerrar sesión



Abrir



Editar



Eliminar



Adjuntar



Actualizar



Etiquetar



Guardar



[Pulsa aquí para descargar los iconos](#)

2.6 Formularios

2.6.1 Consideraciones generales

- Poner un título **descriptivo** al formulario.
- Pedir **información estrictamente necesaria**.
- Los **campos obligatorios** estarán diferenciados de los no obligatorios: los campos obligatorios tendrán su correspondiente etiqueta **en negrita**. Por la lógica del punto anterior, se evitarán en la medida de lo posible los campos no obligatorios. Utilizar la propiedad aria required para optimización de la accesibilidad.
- Siempre que sea posible, se **agruparán los campos obligatorios al principio** del formulario.
- **Agrupar los campos de forma lógica**.
- En formularios de varios pasos, permitir que el usuario navegue al paso anterior o siguiente **sin que pierda los datos ya introducidos**.
- **No pedir los mismos datos** en más de una ocasión.
- En la **validación de campos de texto, se recomienda aceptar algunos “errores” comunes** como espacios en los números de teléfono, los puntos de millares o el uso indistinto de mayúsculas o minúsculas.
- Realizar la validación de campos en el navegador, y preferiblemente mientras el usuario lo completa. Esto se realizará, al pasar de un campo obligatorio sin ser rellenado, al siguiente. **En ese caso, se resaltará el campo con un mensaje informativo al lado:**

Por favor, rellene este campo

- Si la validación la hacemos en el servidor, informar al usuario detalladamente de los **campos incorrectos, y no borrar la información que ya ha completado**.
- Evitar **elementos visuales** que puedan **molestar** o desconcentrar al completar el formulario. Por ejemplo, una animación.
- **No pedir información que podemos obtener nosotros** a partir de otros campos. Por ejemplo, podemos obtener la provincia a partir del código postal o la letra del DNI a partir del número de documento.

- Se cuidará que el **tabindex** esté ordenado adecuadamente para facilitar la navegación con la tecla tabulador (siempre desde el **campo superior al inferior**).
- Asignar la tecla “**Intro**” para que realice la **acción principal**. Para realizar dicha tarea se mostrará el tipo de **botón Normal Destacado**. En los casos en que haya campos multilínea, habrá que desactivar en consecuencia esta funcionalidad cuando el foco se encuentre en el campo, y volverse a activar cuando esté fuera.
- En los campos (**input text**) en los que sea factible por tener un número de caracteres fijo (DNI, código postal), tras completarlos, pasar al siguiente campo automáticamente.
- El texto que acompaña a los diferentes campos de formulario, deben ser la etiqueta html: **label**
- Hasta que no estén todos los elementos obligatorios del formulario rellenos, dicho botón mostrará su estado como Desactivado:



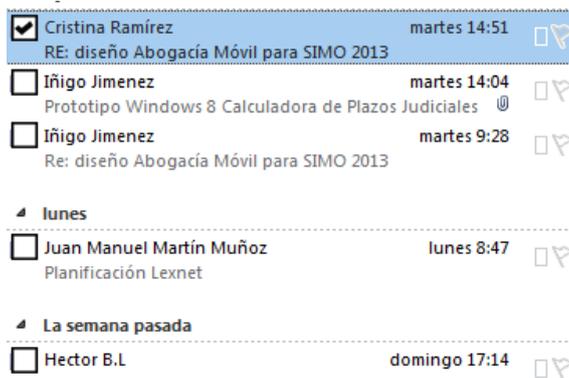
(Ver apartado 2.10 Botones)

- Campo **Contraseñas**. Es necesario que se informe al usuario de las normas de seguridad para completar el proceso de establecer la contraseña. Un texto informativo del tipo: “La contraseña debe tener al mínimo 6 caracteres y debe contener al menos dos números”.
- 
- Dicho texto debe estar visible **antes** de que el usuario complete el proceso, y no sólo como mensaje informativo de error.
 - Es recomendable disponer de un control que informe de la fortaleza de la contraseña. Así mismo es recomendable usar una utilidad de mostrar caracteres, ocultos por defecto, especialmente en las versiones móviles de las webs.
 - Se contemplará la utilidad **autocompletar** para los campos en que sea factible.
 - Si el proceso de envío del formulario es lento, mostrar un mensaje “**Enviando formulario**”.
 - En caso de errores, mostrar información detallada.
 - Se recomienda evitar la funcionalidad de “limpiar” todos los campos de un formulario

mediante -por ejemplo- un botón, para evitar que un usuario borre accidentalmente un formulario ya rellenado. En el caso de que por la naturaleza de la aplicación se considere su uso, debería informarse al usuario previamente con un mensaje de aviso (*ver apartado 2.8 Mensajes*)

- Es recomendable facilitar el **uso de teclas para facilitar las acciones** a los usuarios avanzados (por ejemplo la tecla May pulsada para seleccionar varios elementos y la tecla Cntrl para añadir/quitar elementos seleccionados. Es importante en aplicaciones que vayan a utilizarse de manera continua por determinados usuarios (en un formulario de registro, por ejemplo, no tendría sentido).

Ejemplo:



Se selecciona un mensaje. Se mantiene la tecla mayúsculas presionada.



May

Se pulsa sobre otro mensaje. Todos los mensajes entre el primero y el último quedan seleccionados

2.6.2 Ejemplos de formularios y módulos reutilizables

A continuación se muestra el enlace a un formulario tipo, con diversos elementos de maquetación:

[Ejemplo de formulario](#)

En el ejemplo se están utilizando las tecnologías comentadas anteriormente (html5, css3, jquery...) estipuladas. En el caso. independientemente de la tecnología utilizada, el aspecto gráfico y funcionalidad deben respetar lo máximo posible a los establecidos en este libro.

Aparte, hay módulos cuya frecuencia en los diferentes aplicativos del Consejo General de la Abogacía, hace que resulte útil una maquetación estándar para que puedan ser copiados tanto internamente como por parte de otros colaboradores externos. A continuación se encuentran los enlaces de muestra:

[Ejemplo Módulo de Datos Personales](#)

[Ejemplo Módulo de Solicitudes](#)

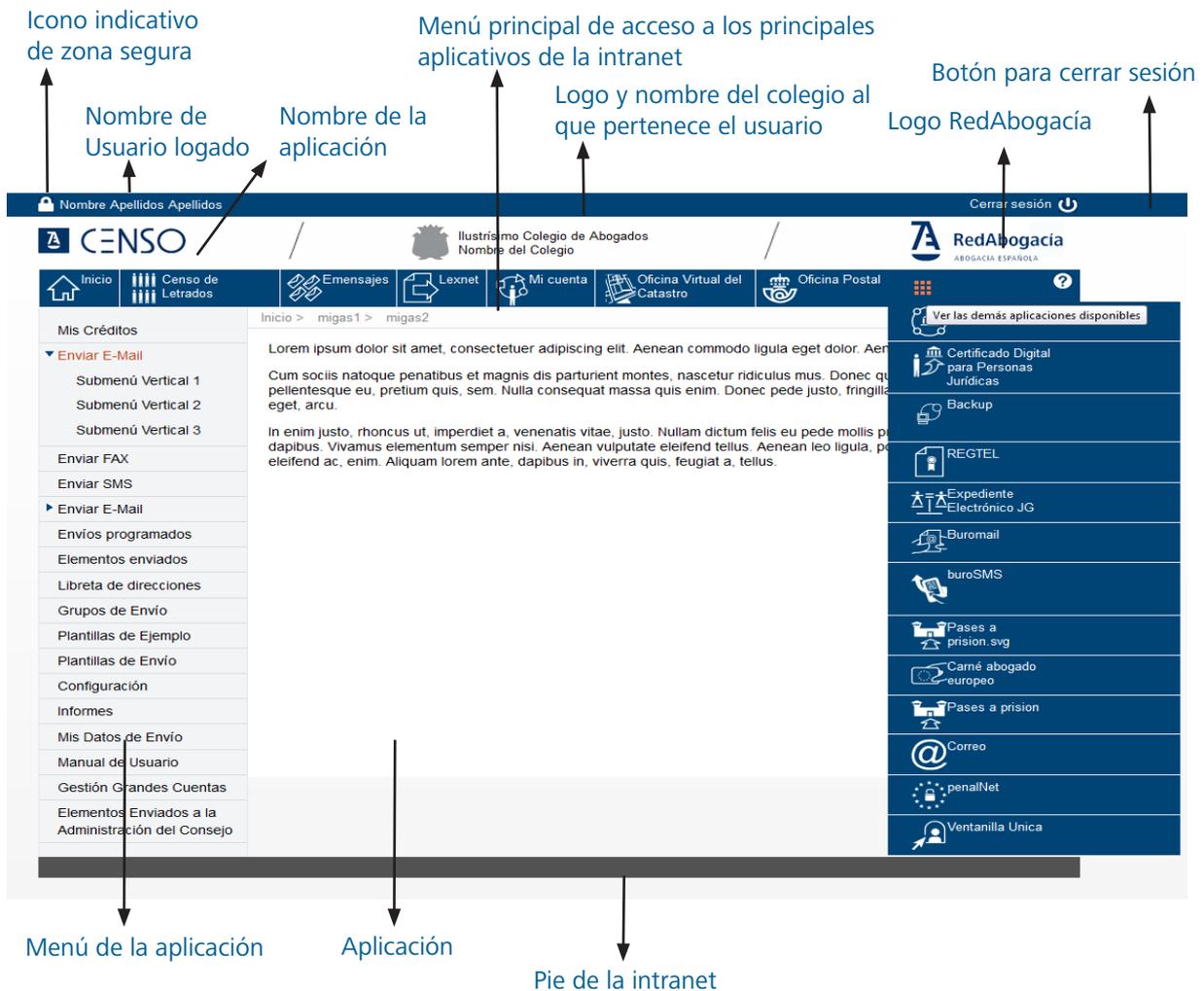
2.7 Estructura Intranet

2.7.1 Estructura gráfica

La estructura general incluye la cabecera, el menú superior, y el pie. Estos elementos estarán siempre presentes en las diferentes aplicaciones. Entre el menú superior y el pie se presentarán las diferentes aplicaciones que constituyen la Intranet.

Cada una de las aplicaciones cuenta con su propio menú, situado a la izquierda de la página, diferenciado gráficamente con una jerarquía inferior al menú general.

En la página siguiente se puede ver una imagen con indicaciones de los diferentes elementos estructurales:



[Ver ejemplo: Estructura Intranet](#)

2.8 Mensajes

Se deben establecer medidas para **disminuir** en el mayor grado posible los **mensajes de error**. Un número elevado de errores es sintomático de una mala planificación de la usabilidad del sitio y provocan frustración en el usuario. Deben evitarse, en la medida de lo posible, anticipándose a los posibles errores y tratando de solucionarlos de antemano.

Los mensajes deben ser **claros, precisos y escuetos**. Tienen que informar al usuario de la tarea que realiza. Debe utilizarse un **lenguaje amigable** en los mismos.

El mensaje debe mostrar por cada párrafo una única información y evitar los párrafos vacíos de información.

Deben ser **gramaticalmente correctos**, con sentido y cumplir con las reglas de **ortografía** del castellano.

Los términos que se considere aconsejable enfatizar, se resaltarán utilizando la negrita.

Ante **errores significativos, reportar a un teléfono.**

2.8.1 Mensajes de error de base de datos y aplicación

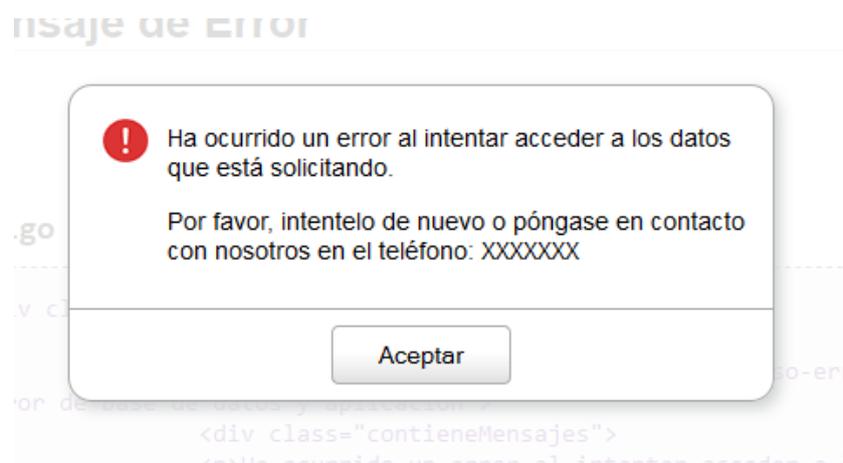
Función forzosa que consiste en impedir que el usuario prosiga con la acción hasta que deje el sistema en un estado correcto.

Este tipo de errores se devuelve sobre la misma página dejando esta inactiva hasta que el usuario no vuelva a la página.

Es recomendable:

- Contener un **texto descriptivo del error** siempre que sea posible, evitando el lenguaje técnico y abreviaturas. Utilizar un lenguaje claro y comprensible.
- Usar frases amigables, evitar que el usuario se sienta culpable. Un mensaje no recomendable, sería, por ejemplo: "ha ejecutado un comando no autorizado").
- Ser preciso: Hacer una descripción exacta del problema y su **solución**, siempre que sea posible.
- Incluir un teléfono de contacto.

El aspecto gráfico será el siguiente:



[Ver ejemplo: Mensaje de error de base de datos y aplicación.](#)

Al pulsar el botón de aceptar se redirigirá a la Home de la Intranet.

2.8.2 Mensajes de error de protocolo estándar HTTP:

En el caso de que sea un mensaje de error de **protocolo estándar HTTP**:

- 4 – error de cliente
- 5 – error de servidor
- 3 – redirección
- 04 – no encontrado

se mostrará una página corporativa del Consejo con el texto: **¡Fuera de Servicio!** Sentimos no poder atenderte en estos momentos. Para más información llámanos al **902 411 141**. El sistema se encuentra temporalmente **fuera de servicio**. Disculpa las molestias.



¡Fuera de Servicio!

Sentimos no poder atenderte en estos momentos.
El sistema se encuentra temporalmente **fuera de servicio**.
Para más información, llámanos al **902 411 141**.

Disculpa las molestias.

2.8.3 Mensajes de error de la Intranet

Se redireccionará a la web de error vista en el apartado anterior.

2.8.4 Mensajes de error de formulario

El error se mostrará gráficamente, **inmediatamente después** del campo del formulario que ha generado dicho error. El campo sobre el cual se ha producido el error se mostrará resaltado.

Este tipo de mensaje se devuelve sobre la misma página dejando esta **inactiva** hasta que el usuario no elija **aceptar** o **cancelar** la acción que provocó el mensaje.

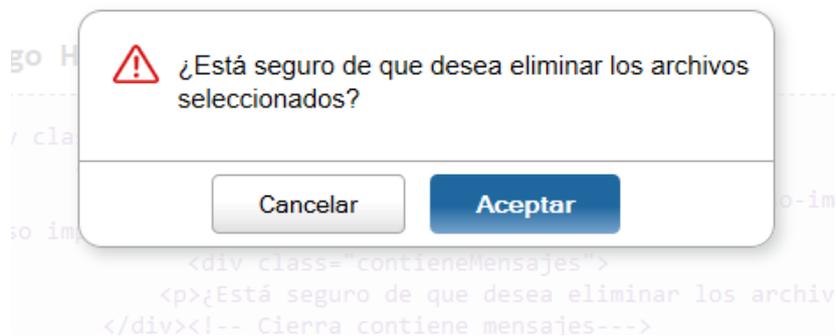
Ejemplo gráfico:

Por favor, rellene este campo

2.8.5 Mensajes de aviso

Se mostrarán cuando el usuario vaya a realizar una **acción de alta importancia**.

Su aspecto gráfico será el siguiente:

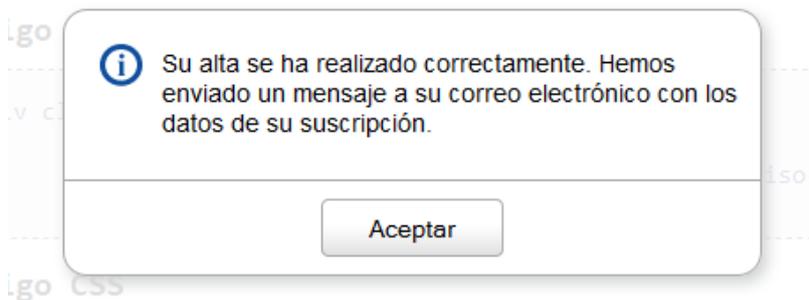


[Ver ejemplo: Mensaje de aviso](#)

2.8.6 Mensajes informativos

Se mostrarán cuando el usuario vaya a realizar una acción de cierta importancia.

Su aspecto gráfico será el siguiente:



[Ver ejemplo: Mensaje informativo](#)

2.8.7 Mensajes informativos sin botón de confirmación

En el caso de que determinadas informaciones se prevea que se van a repetir lo cual puede resultar demasiado reiterativo e innecesario. Para estos casos se contempla la utilización de mensajes que desaparecen automáticamente al cabo de pocos segundos. Normalmente este tipo de mensajes son del tipo informativo, ya que de considerarse una comunicación realmente relevante sí que debe pedirse la confirmación del usuario mediante un botón, ya que de lo contrario corren el peligro de pasar inadvertidos.

Debe estudiarse y tenerse muy claro en qué casos utilizarse, ya que utilizar indistintamente este tipo de mensajes junto con los informativos estándar, llevaría a confusión al usuario al utilizar diferentes medios para un mismo objetivo en el mismo contexto.

2.8.8 Mensajes de Espera

Se mostrarán cuando un proceso de la aplicación pueda tardar un tiempo. Mientras se complete el proceso, si el usuario no debe realizar otra acción paralela, la pantalla le aparecerá desactivada hasta que el proceso finalice.

Si el lapso de tiempo es **muy breve** se utilizará el siguiente gif animado:



[Ver ejemplo: Mensaje de Espera muy breve.](#)

Si el proceso va a ser **breve**, la pantalla se utilizará una barra animada, y un texto: Esta operación puede tardar unos segundos...

		Buscar	Limpiar
Nombre	Apellidos	Nº Colegiado	
	HERNANDEZ CAMACHO		62974
	FLOREZ E		
	PEREZ GUTIERREZ		52425
JEL	LARA MORAL		51409
	MARCOS OLIVA		61083
	HERNANDO AGUILAR		56853
A	SANTOS FERNANDEZ		42022

[Ver ejemplo: Mensaje de Espera breve.](#)

En el caso de que se esté realizando una operación que tarde de manera considerable, y pueda ser medible, se utilizará una **barra de progreso** que informe gráficamente sobre el progreso realizado:

HERNANDEZ CAMACHO	62974	MADF
FLOREZ ESCOBAR	59304	MADF
P		MADF
LARA MORAL	51409	MADF
Descripción de la operación que se está realizando...		
HERNANDO AGUILAR	56853	MADF
SANTOS FERNANDEZ	42022	MADF

[Ver ejemplo: Barra de Progreso](#)

2.9 Enlaces

2.9.1 Consideraciones

La capacidad de las páginas Web de enlazarse unas con otras constituye el verdadero motor de la red. Sin el **hiperenlace**, la comunicación y la interactividad se harían sustancialmente más complicadas, y la Web se convertiría en una mera biblioteca de documentos aislados y de difícil acceso.

La capacidad del hiperenlace ha creado una forma revolucionaria de lectura y organización de los documentos. Es el llamado **hipertexto**.

El hipertexto se diferencia del resto de modos de lectura en la linealidad. Mientras que los modos clásicos se componen de textos completamente lineales, con un orden, un principio y un fin establecidos, el hipertexto permite saltar entre diversos documentos sin la necesidad de seguir una línea uniforme. Este tipo de navegación guarda semejanza con la estructura de la mente humana.

El tamaño y el estilo de la tipografía tiene que mantenerse en los estados de un enlace. Un cambio en la fuente provoca un movimiento innecesario del texto, y para la lectura del texto es necesario que tenga una maquetación fija.

2.9.2 Representación

El uso de subrayado es exclusivo para el estilo de los enlaces. Esto no quiere decir que se represente necesariamente subrayado, aunque esta sea una buena práctica pues es una representación habitual a la que está acostumbrado cualquier usuario.

Para la parte pública el estilo de los enlaces obedecerán a las peculiaridades gráficas de las diferentes páginas dependientes del Consejo General de la Abogacía Española. En cualquier caso, siempre debe cumplirse que el **enlace se distinga claramente del resto del texto**.

Para la Intranet, el aspecto gráfico será el siguiente:

Enlace activo:

Ejemplo de enlace

Estilos:

```
border-color: #336699;  
border-style: dashed;  
border-width: 0 0 1px;
```

- Los enlaces que no tengan un contenido textual claro deben llevar un **title** que especifique textualmente dónde enlaza el link.
- El **color** del enlace visitado debe tener más brillo, luminosidad y saturación, que de aspecto de “utilizado”.
- Evitar la **redundancia**, enlaces del tipo pulse aquí o más información no se deben utilizar, es mucho más comprensible para el usuario que el texto del enlace ya le esté avisando de la información que se le va a proporcionar.
- En enlaces que se incluyen dentro de un párrafo, el texto debe estar integrado de forma natural en el párrafo, de forma que si no existiese enlace siguiese teniendo sentido.
- En enlaces colocados en serie, en lugar de ir integrados dentro de un texto, como por ejemplo, un menú, es importante que estos formen parte de **listas** de elementos en HTML. De esta forma la lista de enlaces tiene un **sentido semántico marcado**, siendo esta práctica positiva para SEO (posicionamiento natural en buscadores).
- Al pasar por encima del enlace (hover), que el **fondo del texto del enlace cambie** para resaltar el enlace, ayuda a **identificarlo**, junto con el subrayado de las palabras que componen dicho enlace.
- No son muy recomendables los enlaces para navegar de una sección a otra dentro de una misma página. El usuario tiene un modelo mental bien definido acerca del comportamiento de los enlaces, cuando pincha en un link espera ser dirigido a una página web distinta de la que está visitando. Los enlaces de ancla violan ese modelo mental al enlazar a zonas dentro de una misma página, lo que puede confundir al usuario.

2.9.3 Enlaces rotos

Evitar los enlaces rotos es una de las consideraciones básicas en la experiencia de usuario sobre cualquier web. Afecta también a **SEO: se penaliza** el posicionamiento natural en buscadores de los

sitios con este tipo de incidencias, ya que suponen una desmejora en la calidad del sitio.

Es una práctica muy positiva llevar un control sobre las webs públicas, hacer un seguimiento periódico y corregir este tipo de incidencias. En el caso de que los enlaces sean gestionados por otros usuarios (como por ejemplo en Ventanilla Única), sería importante también mantener un control y corregir dichos enlaces.

Los enlaces rotos también afectan a la reputación online de cualquier entidad.

Existen multitud de herramientas que facilitan esta labor, como las Herramientas para webmasters de Google (Errores de rastreo), la herramienta del W3C (World Wide Web Consortium): Link Checker, etc.

2.10 Botones

Se utilizará el infinitivo para definir la acción que realizará el botón (Buscar, guardar, aceptar, etc.)

2.10.1 Botón de Máxima jerarquía y dependientes del mismo

Acción importante

Se utilizará para acciones importantes, teniendo un cierto carácter promocional, por lo que está más pensado de cara al **portal público, y no para** acciones de formularios más complejos de la **intranet**.

Acción 2

Se utilizará como botón dependiente de la acción iniciada por el botón de máxima jerarquía.

Acción 3

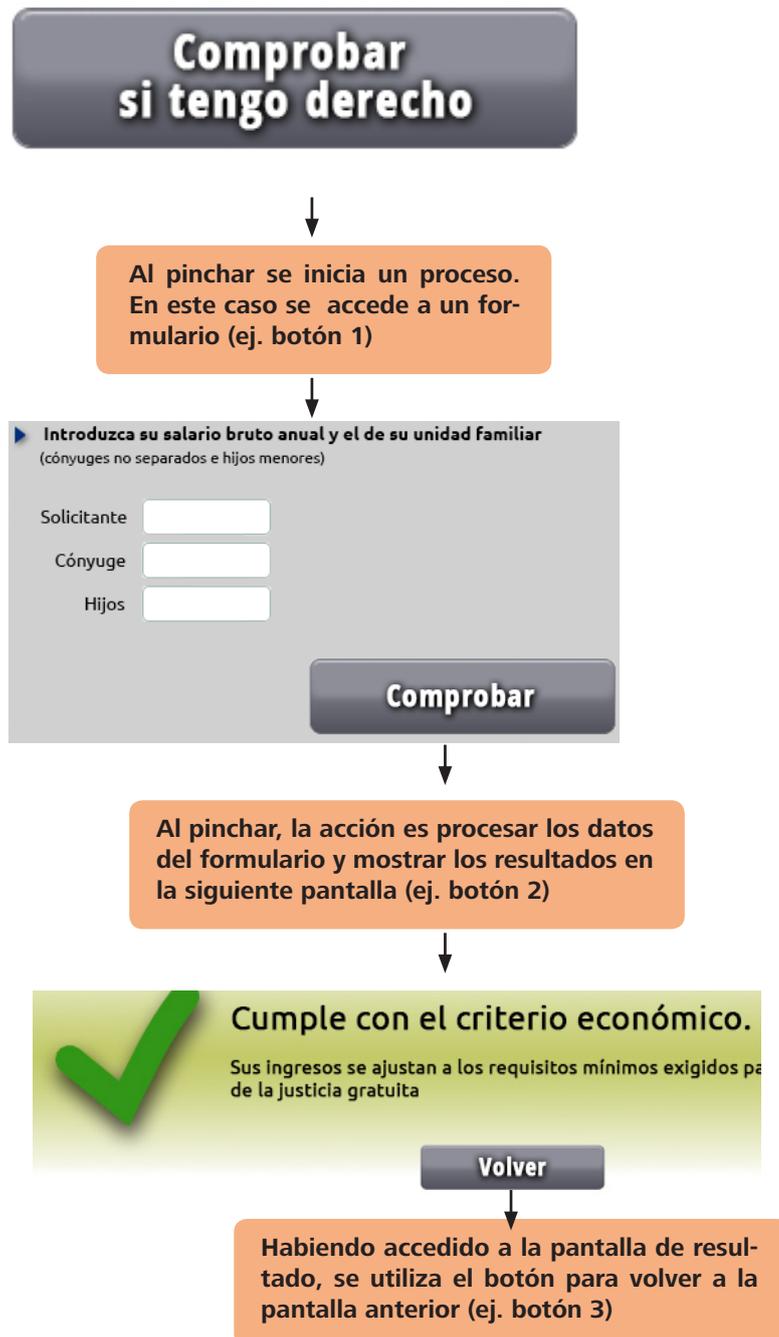
Se utilizará como botón dependiente de los dos anteriores, en los casos que sea necesario. A continuación se muestra un ejemplo de uso:

Bienvenido a JusticiaGratuita.es

Un servicio 24 horas al día, 365 días al año

La **Constitución Española**, en su artículo 119, establece que **la justicia será gratuita cuando así lo disponga la ley** y, en todo caso, respecto de quienes acrediten insuficiencia de recursos para litigar.

Es un servicio que prestan 39.000 letrados con su 83 Colegios de Abogados.



2.10.2 Botones. Consideraciones Generales.

Los botones se componen de un texto y/o una imagen que **comunican claramente la acción que se producirá cuando un usuario pulse dicho botón.**

Debe utilizarse una imagen como botón cuando su significado esté tan estandarizado y sea tan claro que no necesite ningún texto explicativo adicional. Un ejemplo sería un botón de lupa como metáfora de “buscar”.

Otro tipo de botón son los formados por un **conjunto de texto e imagen**, cuando la información de ambos se complementa y den claridad al usuario.

Por último, los botones consistentes en un **texto con un fondo**. Este tipo de botones tienen un peso considerable en la composición gráfica de las pantallas, por lo que se deben utilizar con cierta moderación.

Es aconsejable su utilización cuando:

- Se quiere llamar la atención de los usuarios para que realicen una acción que nos convengan (por ejemplo, un regístrate).
- Se requiere una confirmación del usuario ante un punto de decisión importante (¿Quiere borrar este mensaje?)
- Cuando el usuario está a punto de realizar una acción decisiva (Por ejemplo, “borrar todo” o “comprar”)

Como norma, usar el infinitivo para definir la acción que realiza el botón (buscar, guardar, aceptar...)

El texto del botón debe ser lo suficientemente explícito, y explicativo. Se establece la primera letra en mayúsculas y el resto en minúsculas.

La fuente utilizada será Arial, y tendrá un aspecto gráfico de 14px.

2.10.3 Botones de Formulario. Tipos.

Botones Normales Destacados

Realizan acciones de formulario importantes, tales como Aceptar, Guardar Datos, que impliquen una acción fundamental en un formulario.

Estado Normal



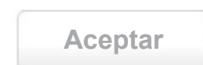
Sobre



Al pulsar



Desactivado



Botones Normales

Realizan acciones de formulario normales.



Botones Pequeños destacados

Realizan acciones de formulario importantes, tales como Continuar, Guardar Datos, que impliquen una acción fundamental en un formulario. La importancia de la acción que realice será menor a la del Botón Normal Destacado, si existen ambos en un mismo proceso.



Botones Pequeños

La importancia de la acción que realice será menor a la del Botón Normal, si existen ambos en un mismo proceso.



Botones de avance y retroceso.

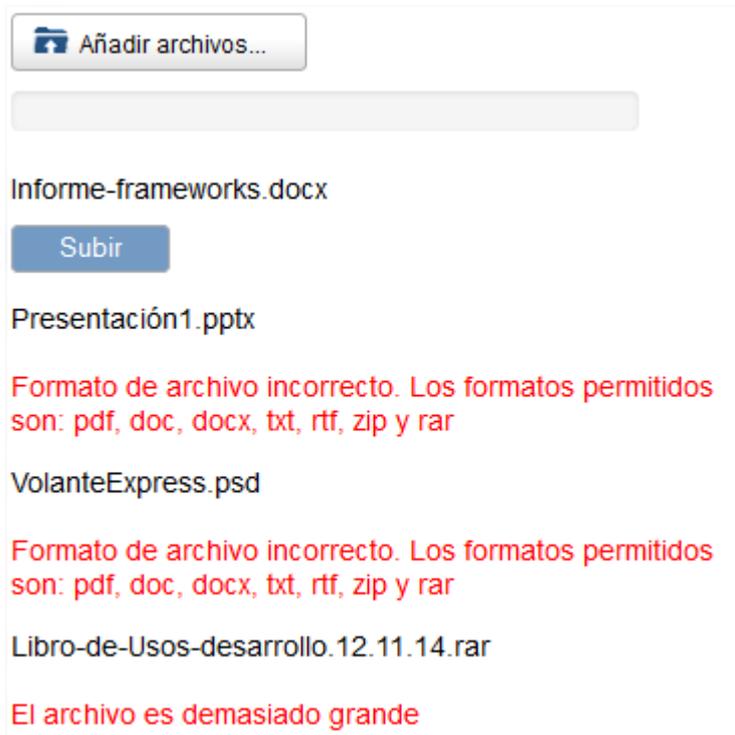
Los botones que impliquen una acción de **avance** dentro de determinado proceso, **se situarán siempre a la derecha**. Tendrán el símbolo **»** El código html para renderizarlo es **»**;

Los botones que impliquen una acción de **retroceso** dentro de determinado proceso, **se situarán siempre a la izquierda**. Tendrán el símbolo **«** El código html para renderizarlo es **«**;



[Ejemplo: Botones de Intranet.](#)

Botón añadir archivos



A screenshot of a web interface for uploading files. At the top, there is a button labeled "Añadir archivos..." with a folder icon. Below it is a horizontal bar representing a file list. The list contains the following items:

- Informe-frameworks.docx
Subir
- Presentación1.pptx
Formato de archivo incorrecto. Los formatos permitidos son: pdf, doc, docx, txt, rtf, zip y rar
- VolanteExpress.psd
Formato de archivo incorrecto. Los formatos permitidos son: pdf, doc, docx, txt, rtf, zip y rar
- Libro-de-Usos-desarrollo.12.11.14.rar
El archivo es demasiado grande

Como se puede comprobar, el componente contiene un control sobre los tipos de archivo soportados y el tamaño máximo de archivo permitido. Estos valores son personalizables modificando el código de la función que renderiza el componente:

```
acceptFileTypes: /(\.|\/) (gif|jpe?g|png|txt|doc|docx|zip|rar|pdf|rtf) $/i,  
maxFileSize: 5000000,
```

[Ver ejemplo de Botón Añadir Archivos](#)

2.11. Labels e Inputs

2.11.1 Inputs y labels, consideraciones generales.

Es fundamental que cualquier tipo de **input pueda ser seleccionado al pulsar no sólo sobre el mismo, sino sobre la etiqueta label que los acompaña**. Esto es especialmente importante en radiobuttons y checkbox, ya que **aumentamos sensiblemente el espacio de interacción**, habitualmente reducido. Esto es especialmente importante en dispositivos móviles. En la maquetación adaptable, además, se debe considerar especialmente el aumento de tamaño en los radiobuttons y los checkbox, ya que deben ser seleccionados de manera táctil.

Para que este comportamiento se haga efectivo, hay que recurrir al atributo **for**, el cual asocia a un input su label correspondiente:

```
<label for="idNombre">Label asociado</label>  
<input id="idNombre" type="radio">
```

En los formularios debe tomarse en consideración la utilidad de que aparezca el primer campo obligatorio preseleccionado al cargar la página.

Cuando un campo input text es seleccionado, debe facilitarse de manera gráfica que ese campo está seleccionado, diferenciándolo del resto. Para ello se puede utilizar el atributo "**autofocus**":

```
<input id="idNombre" type="radio" autofocus>
```

No es recomendable utilizar colores vivos en los bordes, tipografías, fondos en el estado por defecto en los inputs y labels, ya que normalmente este uso está asociado a resaltar los campos en situaciones determinadas. Por ejemplo, para indicar a un usuario que un campo obligatorio debe ser rellenado, o al ser seleccionado un campo.

En dispositivos móviles debe tenerse en consideración la importancia del tamaño de los inputs, puesto que se seleccionan mediante una interacción táctil. Deben ser más grandes y claramente identificables como tales.

En input text, para campos cuyo número máximo de caracteres sea predecible (por ejemplo, fecha), debe limitarse el número de caracteres a introducir para evitar posibles equívocos en usuarios, por ejemplo:

```
<input type="text" maxlength="10">
```

2.11.2 Input Fecha

Es uno de los campos del formulario que más errores genera.

El formato más adecuado:

dd/mm/aaaa

En el campo debe ponerse en gris dd/mm/aaaa, como ayuda e indicativo de que se puede introducir manualmente la fecha.

Al lado de los campos de fecha se establece el incluir un calendario para la introducción de fechas.

Al pulsar el icono del calendario se debe procurar que no tape el literal de la fecha y que el calendario aparezca junto a la fecha, a la derecha.

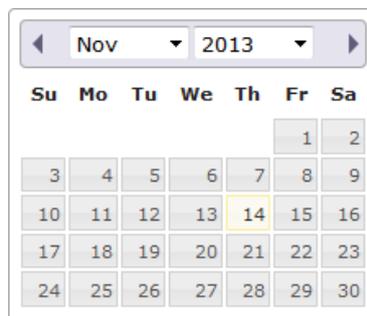
Fecha: 

Se establecen dos tipos de calendarios a desplegar, en función del requerimiento del dato. Si la fecha a seleccionar se estima pueda ser cercana, se utilizará un modelo como el siguiente:



[Ver ejemplo: Input Fecha](#)

Para los casos en los que se estime que el usuario tenga que seleccionar una fecha lejana en el tiempo, se habilitará otro modelo para facilitar la selección:



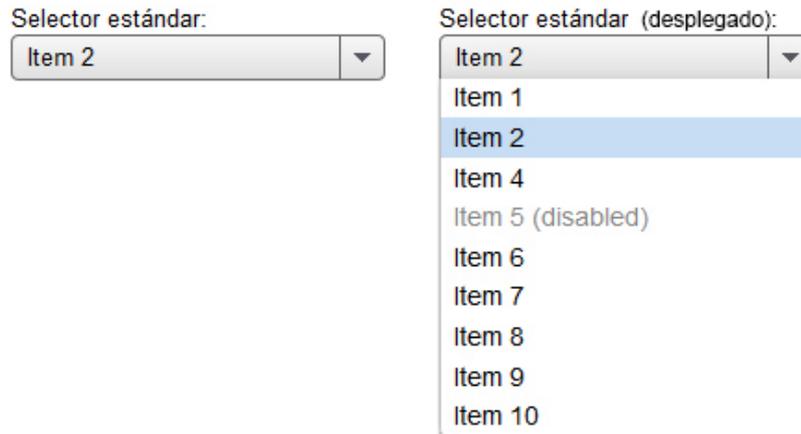
[Ver ejemplo: Input Fecha Lejana](#)

Así mismo, también se contempla el caso en el que sea necesario establecer además la hora exacta:

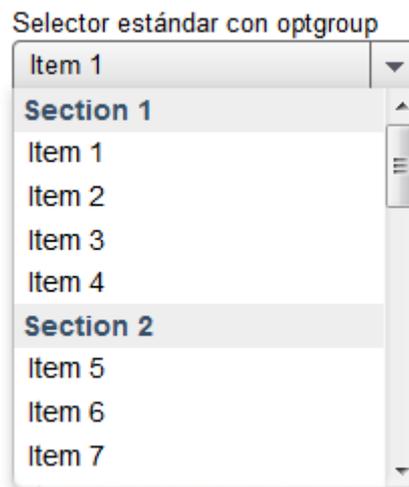
[Ver ejemplo: Input Fecha Hora](#)

2.12 Selects

El aspecto gráfico será el siguiente:



Se puede contemplar una jerarquización entre las opciones mostradas, para facilitar una rápida elección entre diferentes categorías, usando la etiqueta html **optgroup**:



No se recomienda el uso de selectores multiopción, pues hay más probabilidad de que los usuarios se equivoquen al hacer la selección.

[Ver ejemplo: Selects](#)

2.13 Paginación.

Se utilizará el siguiente formato en las aplicaciones:



Al situar el cursor sobre las páginas, el elemento cambiará su color de fondo, al igual que al seleccionar una de las páginas. Los iconos-flecha deben tener un "title" que al situarse sobre los mismos indiquen cuál es su funcionalidad (avanzar, retroceder, ir a la primera página, ir a la última página).

[Ver ejemplo: paginación](#)

2.14 Tablas.

2.14.1 Consideraciones Generales

Los datos más importantes deben colocarse en las columnas situadas más **a la izquierda**.

Evitar tablas muy complejas, con muchas columnas. Siempre que sea posible es preferible utilizar dos o más tablas, agrupando sus valores de forma lógica.

Se evitarán textos en filas con salto de línea. Cuando el contenido de una celda sea lo suficientemente largo como para sobrepasar el ancho de la celda, en lugar de generar una segunda línea, debe mostrarse la información en un tooltip al situarse sobre el texto en sí.

Como medida excepcional, para tablas con muchas columnas, se podrán crear tablas con scroll horizontal, aunque si es posible, se evitará dicha práctica si los datos de dicha tabla pueden dividirse en dos o más tablas.

Ejemplos gráficos para la Intranet:

Mostrar registros Buscar:

Browser	Platform(s)	Engine version	CSS
All others	-	-	
All others	-	-	
All others	-	-	
All others	-	-	
AOL browser (AOL desktop)	Win XP	6	
Camino 1.0	OSX.2+	1.8	
Camino 1.5	OSX.3+	1.8	
Dillo 0.8	Embedded devices	-	
Epiphany 2.20	Gnome	1.8	
Firefox 1.0	Win 98+ / OSX.2+	1.7	

Para las webs públicas, se utilizará o bien este modelo u otro que se adapte al estilo gráfico de cada web, se estudiará en cada caso de modo particular.

La secuencia de uso es la siguiente:

Se selecciona un registro de la tabla.

Se le informa al usuario gráficamente de que ese registro está efectivamente seleccionado. Para ello es conveniente utilizar 3 estados en cada fila seleccionable: En reposo, sobre, y seleccionado.

La acción de eliminar registro se mantendrá en la esquina inferior izquierda. Toda acción que implique eliminar un registro, debe llevar un mensaje de advertencia. Las acciones sobre registros tienen un icono asociado y un texto explicativo de la acción que realizan.

[Ver ejemplo: tablas](#)

2.15 Captchas.

CAPTCHA es la abreviatura de: Completely Automated Public Turing test to tell Computers and Humans Apart (Prueba de Turing Pública y Automática para diferenciar Máquinas y Humanos).



Utilidad: Evitar que robots, (o bots), puedan vulnerar la seguridad de un sitio. Un robot es un programa informático que imita el comportamiento humano y puede realizar funciones rutinarias.

Para detectar si es un robot, se obliga a una interacción humana que descarte que se trata de un robot, sirviendo como muro.

Desde el punto de vista de la usabilidad y la accesibilidad, **debe evitarse el uso de captchas siempre y cuando sea posible**, relegando las tareas de seguridad siempre que sea posible.

Controversias actuales sobre su uso:

- Obligan a realizar pasos extras a un usuario para conseguir un objetivo.
- Los bots son cada vez más sofisticados, pudiendo incluso imitar a un humano mediante técnicas de OCR (Reconocimiento óptico de caracteres). Su efectividad es inversamente proporcional a la sofisticación de los bots, lo que obliga a complicar los caracteres a reconocer por el usuario.

- Al hacerlos más complicados, existen mayores posibilidades de equivocación, mayor detrimento de la usabilidad, y la accesibilidad.
- Es uno de los elementos de formulario que más frustración provocan en el usuario.

No obstante en los casos en los que por seguridad del sistema sea imprescindible usarlos, se justifica su uso bajo unos mínimos de accesibilidad y usabilidad. Todo captcha debe tener como mínimo:

- Opción de refresco.
- Botón con funcionalidad de deletrear sonoramente los caracteres del captcha.

Como recomendación, utilizar el re-captcha: <https://www.google.com/recaptcha/intro/index.html>

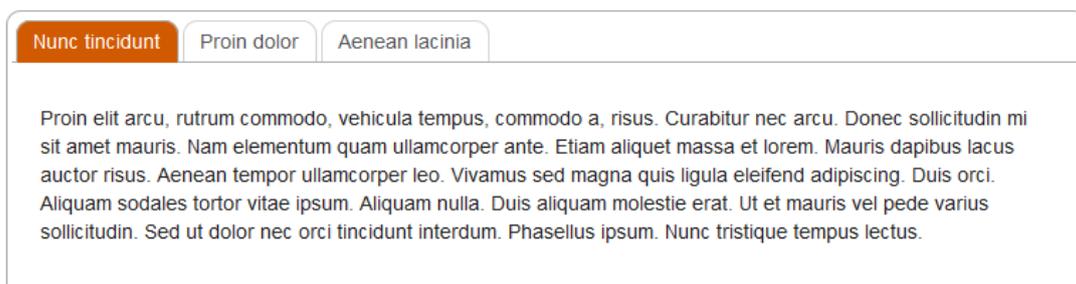
Más información: <http://es.wikipedia.org/wiki/Captcha>

2.16 Pestañas

Las pestañas ayudan a mejorar la organización del contenido, **dividen el contenido en secciones lógicas** que ocupan menos espacio en la pantalla. Es una manera de que los usuarios puedan acceder fácilmente al contenido que les interesa.

Para aplicativos de la Intranet preferiblemente se utilizará una sola fila de pestañas, aunque en casos específicos se puede utilizar una doble fila de pestañas. No es conveniente en cualquier caso abusar de varias filas de pestañas pues satura la pantalla y se pierde la facilidad de navegación.

Aspecto gráfico:



[Ver ejemplo: pestañas](#)

En las aplicaciones públicas, el aspecto gráfico de las pestañas si fuera necesario aplicarlas, dependerá de las peculiaridades de la propia aplicación, siendo definidas a criterio del diseñador.

Las secciones de mayor importancia se situarán más **a la izquierda**. Se mostrará siempre la primera

sección seleccionado (con fondo naranja y en negrita).

2.17 Icono semáforo

Son iconos que se pueden reutilizar para distintas aplicaciones.

Rojo: Valor negativo. (No se puede continuar, tenemos pendientes una tarea con retraso, opción no recomendada, prioridad baja, etc).

Naranja: Valor intermedio. Como una advertencia moderada (prioridad intermedia).

Verde: Valor positivo. (Propuesta para continuar, tarea que va sin retraso y con tiempo para desarrollarla, opción recomendada, etc).

Formas básicas semáforo:



Siempre que sea posible, es importante contemplar que la información no se base sólo en el color, por accesibilidad (por ejemplo, a una persona con daltonismo o acromatopsia le resultaría complicado comprender dicha información).

[Ver ejemplo de icono semáforo.](#)

2.18 Buscadores

En las webs dependientes del Consejo con un tamaño considerable, es muy recomendable la implementación de un buscador. El aspecto gráfico se contemplará en función de las peculiaridades de cada caso.

En las aplicaciones de la Intranet, en las que se considere de utilidad, es recomendable estudiar su implementación. Su estructura básica consiste en un input de entrada de texto y un botón de buscar.

La utilidad del navegador es fundamental en la experiencia de usuario, ya que le puede evitar tiempo buscando en secciones y textos más o menos amplios. Sin embargo, es preferible que no haya un buscador, a que haya uno que no sea realmente útil y provoque frustración en el usuario.

Son recomendables una serie de prácticas para que un buscador resulte realmente útil:

- Que no falle. Parece una obviedad, pero ocurre con cierta frecuencia encontrar buscadores

que no realizan su cometido o lo hacen de una manera tan irregular que dejan una pésima impresión en el usuario.

- Que haga la búsqueda dentro de los contenidos del propio portal, y no de todo internet. Si un usuario quisiera buscar un contenido en internet, se irá a su buscador de internet preferido, no utilizará un buscador dentro de nuestra web para hacerlo. Cualquier usuario sobreentiende que la búsqueda se centra en los contenidos del portal en el que está en ese momento.
- Que no sea restrictivo con las tildes, es decir, que si buscas “abogacia”, sin tilde, muestre los resultados donde aparezca la palabra abogacía, con tilde.
- Que sea lo suficientemente restrictivo al hacer las búsquedas, es decir, que si busco “formación”, no me incluya resultados que incluyan, palabras como “de**formación**”, con**formación**, etc.
- Que permita filtros de búsqueda avanzados por fecha, sección, etc, para que los usuarios puedan afinar las búsquedas y encontrar con mayor facilidad los contenidos que están buscando.
- Que ofrezcan los resultados de una manera ordenada.

[Ver ejemplo: buscador](#)

2.19 Vídeo y contenidos swf

No se recomienda el uso de flash como plugin para reproducir vídeo debido a la incompatibilidad en algunos dispositivos móviles con esta tecnología. Esto mismo es aplicable a las animaciones y otros contenidos en formato swf. Aparte de la compatibilidad, al no ser los textos indexados por los motores de búsqueda de los buscadores, afecta al posicionamiento orgánico o natural en buscadores (SEO).

En su lugar se recomienda utilizar html5, css3 y javascript.

En los vídeos, no es recomendable que el vídeo se reproduzca automáticamente al cargar la página. Esta práctica es una intromisión visual y sonora que no ha escogido el usuario. Cualquier usuario espera que se reproduzca un vídeo cuando pulsa en el control adecuado en el reproductor, es decir, el play.

Aparte, muchos usuarios abren varias pestañas al mismo tiempo, algunas de ellas con vídeo. Si se reproducen automáticamente varios de ellos al tiempo, aparte de consumir recursos en el ancho de banda del usuario, se produce una molesta mezcla sonora al reproducirse varios vídeos a la vez.

En general es poco recomendable que una web reproduzca sonidos sin la previa autorización del usuario mediante el control adecuado (los más usuales son el “play” o el control de volumen).

[Ver ejemplo: vídeos](#)

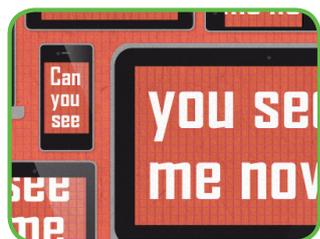
3. Dispositivos Móviles

3.1 Usabilidad Móvil. La importancia.

La progresiva utilización de los dispositivos móviles para acceder a internet y realizar diferentes tipos de tareas es ya un **hábito ampliamente extendido** en todo ámbito social. Especialmente en el mundo desarrollado y con importancia creciente en entornos rurales y países en vías de desarrollo, donde la falta de infraestructuras hacen de dispositivos móviles el principal medio para el **acceso a internet**, y el uso, cada vez más extendido de diferentes **aplicaciones móviles** (apps).



Las recomendaciones y requisitos contemplados en este libro, son aplicables también a las webs y aplicaciones adaptadas a dispositivos móviles. En este medio, son además particularmente importantes, y en algunos casos imprescindibles. Esto es debido a las particularidades de estos medios y la casuística de su uso, que difiere del uso de los equipos de escritorio convencionales.



No tener una web móvil correctamente adaptada, usable, accesible y que induzca a una buena experiencia de usuario repercute negativamente sobre la **imagen de marca** y el **número de usuarios** que demanden nuestros servicios.

La importancia de tener una presencia de calidad en este medio viene subrayada por diferentes áreas de interés, entre otras:

- **Potencial publicitario.** Indiscutible dada la viralidad de este medio en la actualidad. Además, las características de muchos dispositivos ofrecen la posibilidad de recopilar información del usuario a través de -por ejemplo- el servicio de localización geográfica.
- **Utilidad social.** Existe una amplia variedad de aplicaciones que facilitan infinidad de tareas a los usuarios, facilitando además su uso en casi cualquier circunstancia. Por otra parte en ciertas situaciones (oportunidad de comunicarse y localizarse), el uso de dispositivos móviles puede ser de gran ayuda para las personas con discapacidad.

3.2 Web móvil, aplicación nativa y aplicación híbrida.

3.2.1 Web móvil y app, diferencias.

Desde hace tiempo, la web evolucionó hasta el punto de que funcionalidades que antes eran propias y específicas de programas orientado a ordenadores de escritorio, invadieron el universo online. Los sitios web pasaron de ser escaparates estáticos que ofrecían información, a auténticas aplicaciones online, ofreciendo todo tipo de servicios con el consiguiente aumento de su utilidad e interacción entre máquinas y usuarios.

En las últimas décadas nos hemos encontrado con un nuevo paradigma: la popularización de equipos portátiles y la aparición de los teléfonos inteligentes, algunos de los cuales son auténticos ordenadores en miniatura. A la par, la globalización de internet, no sólo en hogares y empresas, sino en prácticamente cualquier ámbito. En este nuevo escenario, se han normalizado las aplicaciones orientadas a estos dispositivos.



La tecnología permite una funcionalidad muy similar en una web online, a la que podría ofrecer cualquier aplicación o app. Si bien, una web móvil, o lo que es lo mismo, una web adaptada a dispositivos móviles, puede ser informativa y funcional, mientras que una app está muy asociada a una funcionalidad o grupo de funcionalidades concretas.

A continuación vamos a ver las principales **diferencias entre una web móvil y una app**:

- Una app puede ser **independiente de una conexión** a internet, una web no.
- Una app necesita ser instalada, el acceso a la web es inmediato (teniendo conexión).
- **Compatibilidad.** Una web es compatible con cualquier sistema operativo. En una app se tienen que hacer versiones de dicha aplicación para las diferentes plataformas.
- **Actualización.** La actualización de una web es inmediata. En una app se depende de una descarga y posterior actualización del software.
- Una app puede hacer **uso de las características del dispositivo móvil**, como la cámara, gps, etc. Una web móvil no.

3.2.2 Aplicaciones nativas y aplicaciones híbridas.

Aparte de Web móviles y aplicaciones, estas dos últimas se pueden dividir en dos categorías, según la tecnología utilizada para su desarrollo.

Una **aplicación nativa** es aquella que está **programada en el lenguaje propio del sistema operativo** para el cual está destinada.

Por el contrario, una **aplicación híbrida** está realizada utilizando tecnologías más propias de la web (**html5, javascript, css**), pudiendo también aprovechar determinadas características del dispositivo (gps, cámara, etc.)



Una de las ventajas de una aplicación híbrida es su **versatilidad**, ya que se adapta a cualquier plataforma mucho más fácilmente. Esto, entre otras cosas reduce el coste de desarrollo pues no hay que hacer diferentes aplicaciones para cada uno de los sistemas operativos. El mantenimiento se reduce pues, aunque por otra parte, lo que se gana en homogeneidad se pierde en la riqueza de personalización aprovechando las diferentes características de cada dispositivo.

En el Consejo General de la Abogacía Española se utilizarán aplicaciones híbridas, salvo que los casos puntuales en los que se especifique lo contrario. Se recomienda el uso de **Phonégap** y para mantener homogeneidad entre los diferentes aplicativos, el framework **Jquery Mobile**.

3.3 Características y contextos de uso.

Características propias de este medio son:

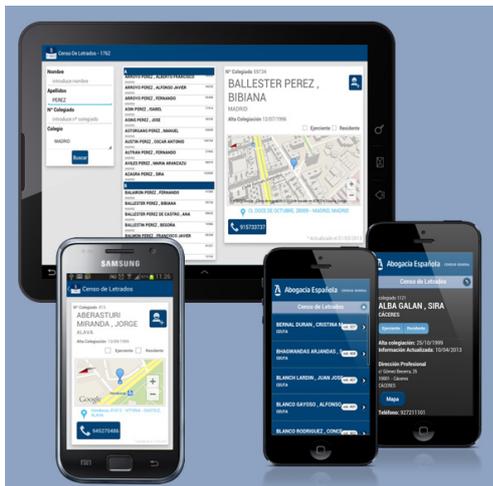
- Condiciones variables de **luminosidad**.
- Capacidad de **conexión** a internet limitada en según qué contextos.
- Tamaños de **pantalla** menores que en otras plataformas.
- **Funcionalidades** como hacer una llamada o enviar un mensaje con un simple click, detección de movimiento, geolocalización, etc.
- **Heterogeneidad** en los terminales: diferentes tamaños de pantalla, teclados, software...
- **Dificultad** para escribir texto, **inestabilidad** de las conexiones.
- Hay que tener siempre en cuenta que en movilidad, las conexiones a internet pueden

implicar **costes adicionales al usuario**, por lo cual hay que tener siempre en cuenta el tamaño de carga y considerar el contenido multimedia (vídeo por ejemplo) , que es el que más recursos de ancho de banda consume.

Hay que poner especial atención en los **múltiples contextos de uso** que se dan al usar los dispositivos móviles por un usuario. En la actualidad el uso se puede dar prácticamente en cualquier situación cotidiana.

Es usual que la capacidad de **atención del usuario** se vea **reducida** por diferentes causas. Un usuario puede tener prisa, por ejemplo, al consultar su cuenta bancaria en un lugar público antes de hacer una compra. La ejecución de tareas puede interrumpirse por una llamada entrante, pérdida de cobertura, o cualquier distracción.

3.4 Diseño y Maquetación



Hay que ser consciente a la hora de diseñar una interfaz orientada al usuario, que ante estas circunstancias, no hay que confiar en la **memoria** del mismo. Se debe partir del principio de no hacer que el usuario tenga que recordar, por lo que es especialmente importante que se muestre en todo momento información exacta de **dónde se encuentra** de un vistazo rápido.

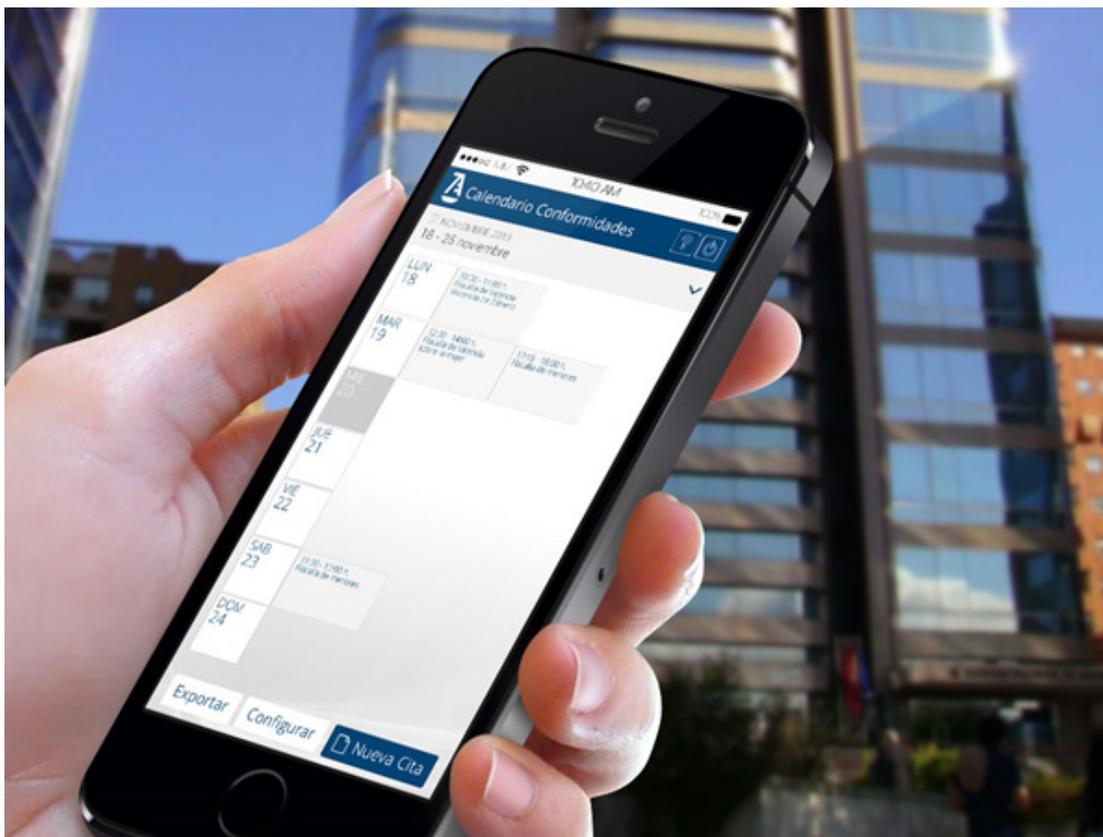
En tan amplios contextos de uso, la **claridad de información, la sencillez y la correcta organización** de los diferentes elementos cobran especial importancia.

Teniendo en cuenta el número de aplicaciones que se pueden encontrar en las app stores, con cifras que superan las 800.000 aplicaciones (datos de Enero de 2013 en Google Play), es muy importante destacar. Las claves para el éxito de una app son **que aporte una utilidad real** en el usuario, tenga una **buena usabilidad** y un **buen diseño**. (Hacer consideración sobre ASO)

3.4.1 Recomendaciones generales

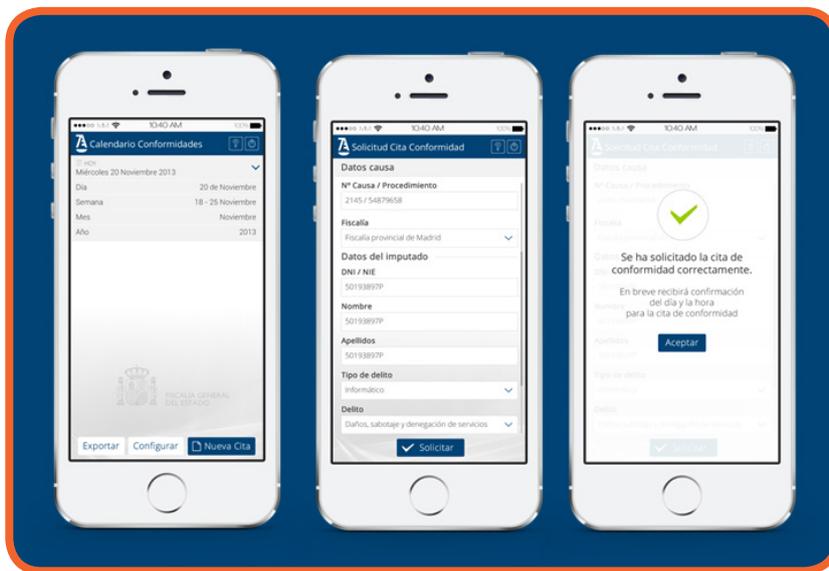
- En web móviles **es vital que estén optimizados los tiempos de carga** (ver apartado 1.11 Velocidad de carga) . En este medio no se tienen a menudo las mismas velocidades que en un equipo de escritorio, el aumento de ancho de banda puede suponer gastos adicionales en el usuario, y las conexiones son más inconsistentes (por una mala cobertura, por ejemplo)
- No es nada recomendable, tener cientos de funciones javascript, corriendo al mismo tiempo; si se puede resumir en una decena, es especialmente importante hacerlo.

- Hay que considerar que no todos los dispositivos móviles tienen webviews (navegadores propios de un dispositivo móvil) tan potentes como los propios de un equipo de escritorio.
- Cada acción que lleve a cabo el usuario debe conllevar un **feedback**, esto es, una respuesta del sistema informativo. Por ejemplo, **si un usuario aprieta un botón éste debe dar la sensación de haberse apretado**. Si un usuario realiza una búsqueda, infórmale de que se está realizando. Si no lo haces es muy probable que tus usuarios intenten apretar varias veces los mismos botones porque no saben si lo han hecho ya.
- **Rapidez**. Sólo interrumpir si es importante. Es mejor no hacer perder el tiempo a los usuarios con ventanas emergentes ni pantallas de bienvenida (splash-screen).
- Utilizar **estándares**. Algo que ya se conoce, facilita la comprensión. El gesto desplazar con el dedo, por ejemplo, ya es un estándar de gestualización básica en smartphones y puede ser un buen atajo para conseguir determinados objetivos al usuario.
- La **adaptación a la rotación y diferentes tamaños** de pantalla, adaptar de la mejor manera posible, y aprovechar el tener más espacio para mostrar barras de tareas –por ejemplo- que en versiones de pantalla más pequeñas estén ocultas para optimizar espacio.



- Utilizar **colores vivos con moderación y con propósito**. Utilizar colores de manera semántica y siguiendo estándares, siempre que sea posible.
- **Tamaños grandes**. Esto es especialmente importante en zonas activas que respondan ante el pulsado de un usuario.

- **Tamaños coherentes** en todas las pantallas. Si hay iconos de la aplicación que realizan la misma función en diferentes pantallas, no es recomendable cambiar el tamaño en ninguna otra, a no ser que exista una justificación muy concreta para dicho cambio.
- Si en maquetación web **no es recomendable** la utilización **frames o iframes**, en web móviles lo es aún menos, por su incompatibilidad.
- Se deben **minimizar** los peligros y las **consecuencias negativas de acciones erróneas** o accidentales del usuario.



- El uso del **scroll vertical es una característica habitual** en las webs móviles, dadas las limitaciones por el tamaño de pantalla, por lo que su uso es frecuente en web adaptadas a dispositivos móviles.
- La estructura recomendable para la maquetación sería la de **una columna**, para mejorar la claridad y adaptabilidad.
- La tipografía óptima para pantallas es la de tipografías **sans-serif** (consultar apartado 2.1. *Consideraciones tipográficas*), siendo esta consideración especialmente importante en dispositivos con pantallas de pequeño tamaño. Se utilizará la **Arial**, fuente que además está disponible en la mayoría de los sistemas operativos, para evitar el tiempo de carga adicional que supondría cargar otras fuentes adicionales.
- Debido a las limitaciones propias de dispositivos móviles para introducir datos escritos, este recurso debe relegarse a los casos de uso realmente imprescindibles. **Es mejor ver y seleccionar que recordar y escribir**. Por tanto, son preferibles recursos como listas de selección, radiobuttons y otros controles que no requieran que el usuario escriba.
- **Evitar el inicio de acciones que el usuario no haya iniciado** mediante su interacción de antemano.

- Siempre que sea posible, que los campos tengan **valores por defecto**.
- **Autocompletar** los campos siempre que sea posible. Por ejemplo, si se ha introducido el nombre en alguna ocasión, evitar que en otro proceso se tenga que volver a escribir.
- **Obtener los datos automáticamente**, siempre que resulte posible.
- **Indicar** en los campos **cómo debe introducirse la información** requerida (por ejemplo fecha 00/00/0000).
- Es recomendable poner las **etiquetas (labels) asociadas a los diferentes campos de formulario, encima de los mismos**, en vez de al lado, al contrario de como se muestra frecuentemente en dispositivos de mayor tamaño. Así mismo deben estar asociadas a dicho campo mediante el atributo **for** (ver el apartado 2.11.1 *Inputs y labels, consideraciones generales*).
- **Minimizar el número de campos**.



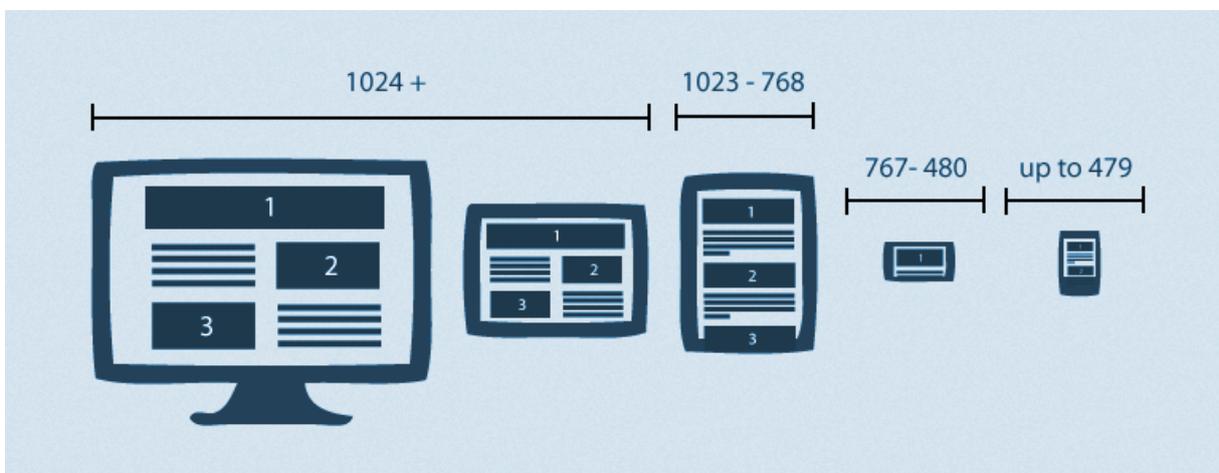
```
1 @media screen and (min-wi...
2   html, body {
3     -moz-background-size: 50% auto;
4     -webkit-background-size: 50% auto;
5     background-size: 50% auto;
6   }
7   body { opacity: 0; }
8 }

1 html { background: #000 url(chameleon2.jpg) 50% 100% no-repeat; }
2 body { background: url(chameleon.jpg) 50% 100% no-repeat; }

1 /* Portrait */
2 @media screen and (orientation:portrait) {
3   -portrait styles */
```

- Debe mostrarse siempre **cuál es la posición del cursor** en un formulario, siendo recomendable el uso del atributo **autofocus**, para que ya esté seleccionado por defecto el primer campo de un formulario a rellenar.
- **Evitar formularios multicolumna**. Si esto es aplicable en los formularios en general, en las aplicaciones y webs móviles es muy importante.
- **No es recomendable el uso de tablas** en aplicaciones o webs móviles. En ningún caso deben utilizarse para estructurar la maquetación. Ni siquiera para mostrar datos tabulares son recomendables, pues pueden provocar fácilmente scroll horizontal. Es recomendable estudiar en cada caso particular otra manera alternativa de presentar dichos datos. En caso de que sean realmente necesarias, debe evitarse en cualquier caso el anidamiento de tablas dentro de tablas, que los encabezados estén bien diferenciados y asociados correctamente con sus correspondientes datos.

- Los **campos con contraseña**, deben estar **protegidos**, enmascarándolos con asteriscos o puntos. En dispositivos móviles, además, es de bastante utilidad que exista la posibilidad de permitir al usuario desactivar esta opción para ver los caracteres, dado que la dificultad en la entrada de datos hace que sea más probable el que un usuario se equivoque. Debe contemplarse, además la opción de “mostrar caracteres”, para facilitar al usuario el desenmascarar los caracteres si así lo desea.
- **Rapidez para cerrar una aplicación.** No poder hacerlo es de las experiencias que mayor frustración provocan.
- Permitir a los usuarios **interrumpir la aplicación en proceso de la memoria del dispositivo.**



- Permitir la opción de **guardar la información en la memoria SD**, y no en la memoria del dispositivo para evitar sobrecargar la memoria propia de los móviles con menor capacidad.
- Dar **seguimiento** a las aplicaciones. Realizar actualizaciones y mejoras no sólo optimiza la aplicación, también aumenta su prestigio. Esto es aplicable siempre desde la prudencia: el exceso puede llegar a resultar molesto para el usuario si recibe actualizaciones a diario, por ejemplo.
- **Aprovechar las características del dispositivo** (geolocalización, cámara, etc) si las características de la app lo requieran y/o cuando permitan facilitar determinados procesos al usuario.
- Para guardar un estilo de diseño coherente entre las diferentes aplicaciones, se escogerá el **flat design o diseño plano, frente al esquemorfismo**. El flat design se basa en evitar el abuso de efectos o filtros gráficos que dan una apariencia 3d a los gráficos, por ejemplo sombreados, biselados, etc. La tendencia contraria se denomina esquemorfismo, y simula elementos reales en la pantalla.
- No es conveniente abusar de varias tipografías en una misma aplicación, preferiblemente utilizar las nativas (propias del sistema operativo). Tipografía de fábrica para Android: Roboto; tipografía de fábrica para IOS: Seoe UI;

3.4.2 Uso de la información: organización, simplificación.

- Debe facilitarse en todo momento el **rápido acceso a la información**. Hay que ser selectivo, reducir y mostrar lo más relevante. Esto ayuda tanto a la hora de su lectura como a la optimización del sitio. Los contenidos deben ser **fáciles de leer**, utilizando un lenguaje amigable, evitando tecnicismos y abreviaturas siempre que sea posible. Para facilitar su rápida asimilación, la información debe estar esquematizada y jerarquizada.
- Es especialmente importante en dispositivos móviles al utilizar el lenguaje, que éste sea, breve y directo, frente a uno divagador.
- Uso **simple e intuitivo**: el diseño debe ser fácil de utilizar y entender independientemente de la experiencia del usuario, su conocimiento, sus habilidades o su nivel de concentración.
- Debe comunicarse la información de manera efectiva, **independientemente de las condiciones ambientales o de las habilidades sensoriales** de los usuarios. Un dato importante a tener en cuenta es que los dispositivos móviles y su uso para interactuar con aplicaciones y web móviles están en pleno desarrollo, y se extiende cada vez más a sectores de población más amplios, como por ejemplo, la tercera edad.
- El propósito u objetivo de los contenidos debe ser claro y obvio: con un **rápido vistazo** los usuarios deben comprender **qué ofrece la aplicación o web**, y **con qué finalidad**.
- La **información más importante debe situarse en primer lugar**, evitando siempre que sea posible que para acceder a la misma el usuario tenga que hacer uso de la barra de desplazamiento o arrastrar la pantalla.



- Añadir una cabecera, un botón u otro elemento a la interfaz resta buena parte de la pantalla. Es una buena práctica preguntarse siempre si ese botón o esa cabecera son totalmente necesarios.
- La web o aplicación debe contener un **título corto pero descriptivo**, que la identifique fácilmente.
- Los **iconos** que se utilicen deben ser **comprensibles** para el usuario, para facilitar su interacción. Unos iconos fáciles de entender, más allá de consideraciones estéticas, son representativos de la buena accesibilidad de un sitio.
- **Dividir** las tareas complejas **en pasos** más pequeños que se pueden lograr fácilmente.
- **Dividir en orden de importancia las acciones** y facilitar el acceso a las más importantes antes.

3.4.3 Notificaciones

Se debe **informar al usuario en todo momento de su interacción con el dispositivo**, los efectos que tienen sus toques sobre la aplicación. Para ello se pueden utilizar diferentes recursos, siendo los más habituales el **cambio de color, aclaramiento u oscurecimiento al pulsar** sobre un elemento susceptible de ello. De este modo se le informa a un usuario de que, efectivamente, ha pulsado sobre un elemento sensible al tacto. Hay que tener en cuenta que en aplicaciones táctiles no existe el elemento hover, por lo que en app hay que evitarlo.

A cualquier usuario se le debe pedir un mensaje de confirmación ante situaciones que:

- Pueden hacerle **perder datos** de manera permanente.
- Requieran **que el sistema se reinicie**.
- Supongan **gastos económicos adicionales** al usuario.
- Impliquen una **acción que va a durar un tiempo considerable**.

Excepciones: a estas consideraciones se le pueden añadir excepciones, en los casos en los que la **acción sea muy obvia**. Por ejemplo, cuando se tiene que pulsar un elemento aislado y arrastrarlo a una papelera, puede considerarse que el proceso es lo suficientemente elaborado como para deducir que sabe perfectamente cuál es la repercusión sobre la acción que está realizando

3.4.4 Navegación.

Aunque muchas de las consideraciones sobre la navegación aquí indicadas también son válidas para otros medios, en dispositivos móviles son especialmente importantes:

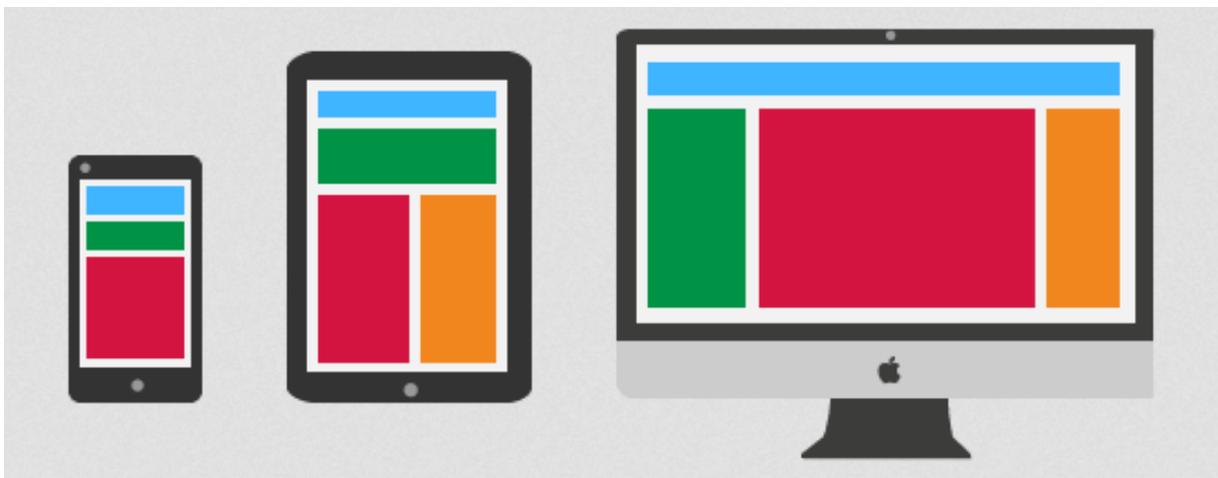
- **Simplificar** lo máximo posible.



- En webs adaptadas al móvil, la simplificación implica eliminar los contenidos irrelevantes. El ideal es **lograr un equilibrio entre la simplificación y la funcionalidad e interés**, sin perder estos al ofrecer una versión excesivamente simplificada. Por ejemplo, en un periódico online, es posible que un usuario prefiera arrastrar la pantalla hacia abajo para seguir leyendo más noticias, que no leer tan sólo las cinco noticias principales.

- En Webs móviles debe facilitarse la opción de **permitir al usuario ver la web en la versión normal** no optimizada para móviles, por si desea ver la versión completa de nuestro sitio.
- Reducir en lo posible los **niveles de navegación**.
- Debe **facilitarse la vuelta a la página principal (Home) o el retroceso a la etapa precedente**.

- Los procesos han de ser sencillos para el usuario. Para facilitar al usuario la realización de procesos, habrá que tener en cuenta los siguientes factores:
 1. Que esté indicado el número de **pasos del proceso** y los pasos restantes.
 2. Que se **evite el uso de información oculta** que requiera de una acción para su visualización.
 3. Que se **informe** de manera inmediata cuando una **acción** está aún **en proceso**.
 4. Que se **informe** de manera inmediata cuándo una **acción** ha sido realizada **con éxito o no**.
 5. Que se le permita al usuario **volver** al paso/s anterior/es **y modificar** los datos introducidos.
 6. Que se **permita salir del proceso** o cualquier estructura de información mediante un botón del tipo desconectar o cancelar, según el caso.
 7. Que las **mismas acciones** deriven en los **mismos resultados**.
 8. Que sea posible **deshacer una acción funcional y operativa**.



- Respecto a los **enlaces externos** o los que enlazan a archivos, deben **identificar claramente adónde se dirigen, o el tipo de documento que se va a descargar** o abrir, para que el usuario evalúe si está realmente interesado en pulsar sobre dicho enlace.

3.4.5 Botones y enlaces

Se aplican los mismos principios que los vistos en el apartado 2.10.2 Botones. Consideraciones Generales, excepto los que se sustituyan por las consideraciones siguientes:

Los **botones y enlaces**, a diferencia de otros dispositivos, en las aplicaciones y webs móviles, **no tienen el estado "hover"** (al situarse sobre el elemento, cambia de aspecto gráfico). En cambio, sí es recomendable que **se informe al usuario** de que, efectivamente, **ha pulsado** un botón de una manera gráfica (cambiando de color, aumentando el brillo, etc.) En CSS, este estado se corresponde con la pseudoclase (clases que definen los estados de las etiquetas que implican una acción) **"active"**.

Por ejemplo:

```
a:active {  
    color: red;  
}
```

Donde **a** hace referencia a la etiqueta html (hiperenlace) **:active** hace referencia al comportamiento que tendrá dicho enlace en el momento de ser pulsado y **color:red;** establece el estilo gráfico que adoptará el enlace en el momento de ser pulsado.

Es **mejor usar iconos que pecar en un exceso de botones**, los iconos reducen el esfuerzo cognitivo, y mejoran la sensación general del usuario al utilizarla.

Las **zonas activas**, susceptibles de interactuar con el usuario al ser tocadas, deben tener un **área mínima de 44x44 píxeles**, siendo esta norma aplicable para botones, elementos seleccionables, etc.

3.4.6 Tipografía

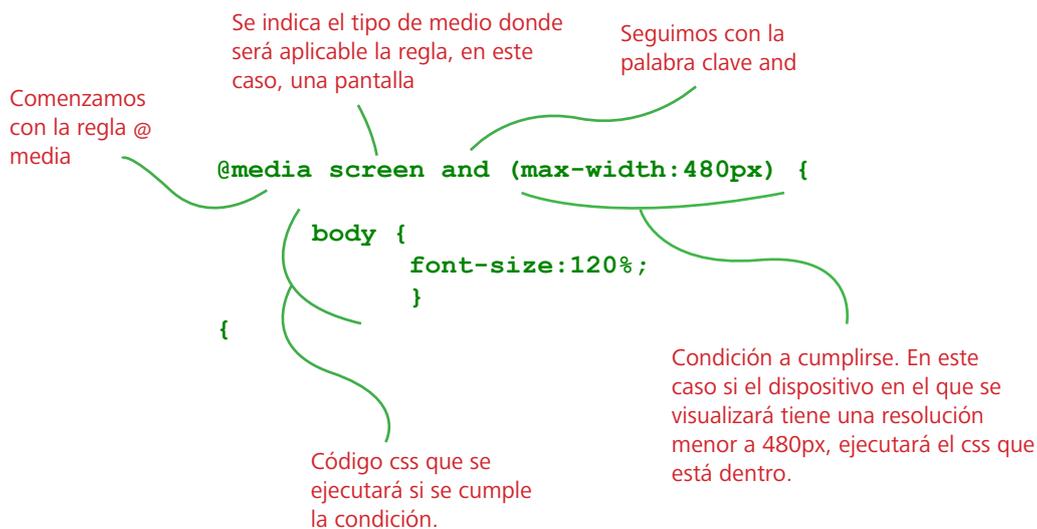
La variación entre los tamaños de texto siempre ha servido en diseño gráfico para **jerarquizar la información**. Esto es aplicable también al diseño orientado a dispositivos móviles, pero con una peculiaridad: al ser pantalla de pequeño tamaño abusar de demasiados tamaños tipográficos puede provocar el efecto contrario, por lo que debe utilizarse este recurso con cuidado. Una manera inteligente sería por ejemplo, reduciendo los textos a **cuatro tamaños jerárquicos** y no sobrepasar a más tamaños:

Text Size Micro	12sp
Text Size Small	14sp
Text Size Medium	18sp
Text Size Large	22sp

3.4.7 Uso de media query

Como ya se ha visto en el apartado 1.9 *Resolución y diseño adaptable*, se recomienda una maquetación basada en **medidas porcentuales**, adecuada para su adaptación en diferentes dispositivos con **diferentes tamaños y resoluciones**. Conjuntamente se utilizará el uso de las propiedades **media query** de CSS para controlar cómo se ve nuestra web en diferentes tamaños de pantalla.

Ejemplo:



Otra manera de utilizar los media query es **referenciar desde el archivo de origen**, de tal manera que **sólo se va a cargar la hoja de estilo en el caso de que se cumpla la condición**.

Ejemplo:

```
<link rel="stylesheet" type="text/css" media="screen and (max-device-width: 480px)" href="style.css">
```

Este método es más recomendable, para que sólo se carguen los contenidos en caso de que se detecte que el dispositivo tiene las condiciones que establezcamos, en este caso que la pantalla tenga una resolución de 480 píxeles como mínimo. En cualquier caso se pueden conjugar ambos métodos para conseguir una maquetación adecuada.

A continuación, se muestra una lista con ejemplos en el código css para identificar diferentes tipos de dispositivos:

```
/* Smartphones (portrait and landscape) ----- */
@media only screen
and (min-device-width : 320px)
and (max-device-width : 480px) {
/* Estilos */
}

/* Smartphones (landscape) ----- */
```

```
@media only screen
and (min-width : 321px) {
/* Styles */
}

/* Smartphones (portrait) ----- */
@media only screen
and (max-width : 320px) {
/* Styles */
}

/* iPads (portrait and landscape) ----- */
@media only screen
and (min-device-width : 768px)
and (max-device-width : 1024px) {
/* Styles */
}

/* iPads (landscape) ----- */
@media only screen
and (min-device-width : 768px)
and (max-device-width : 1024px)
and (orientation : landscape) {
/* Styles */
}

/* iPads (portrait) ----- */
@media only screen
and (min-device-width : 768px)
and (max-device-width : 1024px)
and (orientation : portrait) {
/* Styles */
}
}
/*****
iPad 3
*****/
@media
only screen
and (min-device-width : 768px)
and (max-device-width : 1024px)
and (orientation : landscape)
and (-webkit-min-device-pixel-ratio : 2) {
/* Styles */
}

@media
only screen
and (min-device-width : 768px)
and (max-device-width : 1024px)
and (orientation : portrait)
and (-webkit-min-device-pixel-ratio : 2) {
/* Styles */
}
}

/* Desktops and laptops ----- */
@media only screen
and (min-width : 1224px) {
/* Styles */
}

/* Large screens ----- */
@media only screen
and (min-width : 1824px) {
```

```
/* Styles */
}

/* iPhone 4 ----- */
@media
only screen
and (min-device-width : 320px)
and (max-device-width : 480px)
and (orientation : landscape)
and (-webkit-min-device-pixel-ratio : 2) {
/* Styles */
}

@media
only screen
and (min-device-width : 320px)
and (max-device-width : 480px)
and (orientation : portrait)
and (-webkit-min-device-pixel-ratio : 2) {
/* Styles */
}
```

Aparte, se debe utilizar la etiqueta **meta viewport** para controlar la visualización por defecto del dispositivo:

```
<meta name="viewport" content="width=device-width, minimum-scale=1.0,
maximum-scale=1.0" >
```

3.4.8 Uso de inputs personalizados HTML 5

HTML5 contempla **etiquetas personalizadas para diferentes tipos de entradas de información** (inputs). De esta manera se puede, entre otros, personalizar el tipo de teclado que aparece en la pantalla de un smartphone de pantalla táctil.

Algo positivo a considerar es que, si bien no todos los navegadores los soportan, los que no lo hacen sencillamente los muestran como el genérico `<input type="text">`, por lo que siempre es recomendable su utilización.

Normalmente el dispositivo detecta el tipo de campo que es al especificarlo nosotros con estas etiquetas, y muestran diferentes tipos de teclado que facilitan la rápida introducción de los diferentes datos. Por ejemplo, en un campo email, se facilita la rápida introducción del símbolo @, o en otros casos muestra sólo un teclado numérico para datos que así lo requieran. Antiguamente este tipo de teclados sólo se podían invocar con aplicaciones nativas, pero con html 5 es posible la misma funcionalidad con aplicaciones híbridas.

A continuación vemos los principales:

```
<input type="search">
```

search. Renderiza un campo de búsqueda. Al pulsar sobre él se activa el teclado para realizar dicha búsqueda.



```
<input type="email" name="email">
```

email. Renderiza un campo para introducir una dirección de email. Al pulsar sobre él se activa el teclado con la @ fácilmente disponible.



```
<input type="url" name="url">
```

url. Ofrece teclas fundamentales para escribir una dirección web como son el punto, la barra "/" o la tecla ".com".



```
<input type="number" name="num" min="0" max="50">
```

number. Para este input tenemos los atributos min y max para establecer el máximo y el mínimo que acepta el campo.



```
<input type="time" name="hora">
```

time. Para este input tenemos los atributos min y max para establecer el máximo y el mínimo que acepta el campo.

`<input type="month" name="mes">`

month. Como su nombre indica, especifica el mes.

`<input type="semana" name="semana">`

week. Como su nombre indica, especifica la semana.

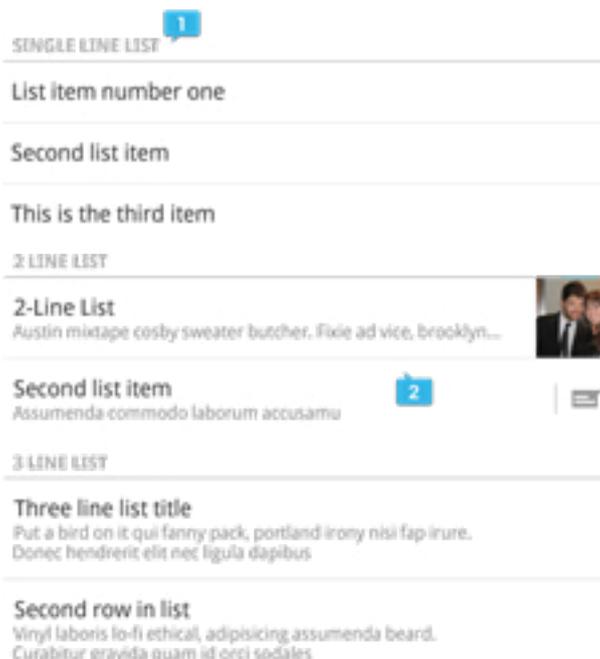
`<input type="range" name="rango" min="0" max="50">`

range. Muestra un control desplazable susceptible de seleccionar diferentes valores que podemos definir con los valores "min" y "max". 

3.4.9 Listas.

Las listas se pueden utilizar para una amplia gama de tipos de datos, pudiendo combinarse con otros elementos para facilitar su interacción con los usuarios, tales como botones de acción, casillas de verificación, etc...

Las listas es recomendable que sean mostradas siempre con una **disposición vertical**, y ordenarlas en **grupos lógicos** cuando sea necesario. Para esto, utilizar recursos gráficos, por ejemplo:



Para listas desplegables, utilizar un icono que indique gráficamente que dichas listas tienen más opciones que se desplegarán al ser pulsadas:



3.4.10 La pantalla de Preferencias o Configuración

Se siguen las mismas especificaciones que las vistas en el apartado 1.7 *Configuración*. Aparte para aplicaciones móviles hay que tener en cuenta que es especialmente importante permitir al usuario **gestionar las notificaciones**.

Puede resultar molesto que una aplicación esté enviando mensajes constantemente al dispositivo de un usuario. **Un usuario debe poder decidir siempre el número de notificaciones que recibe, y el grado de notoriedad de las mismas.** Un ejemplo de una aplicación mal diseñada sería aquella que enviara, por ejemplo, un mensaje sonoro y vibratorio ante cualquier nuevo evento, sin permitir al usuario modificar estos valores.

3.4.11 La ayuda

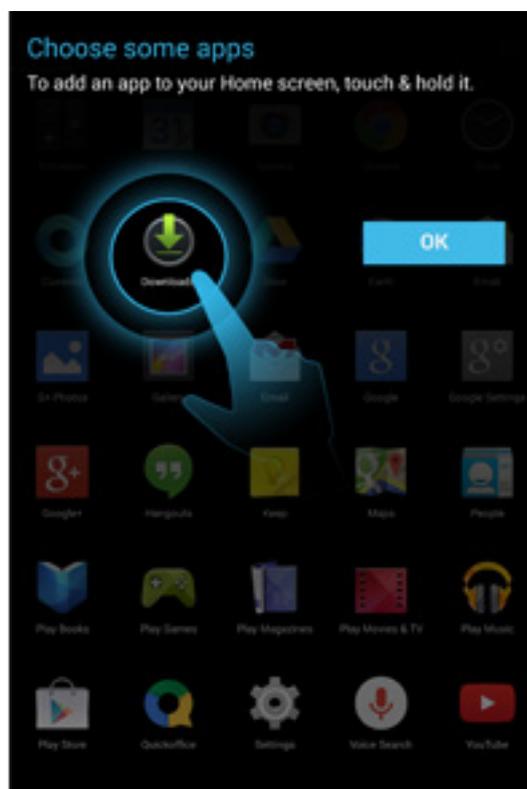
En una aplicación ideal debería no ser necesaria. No obstante, una aplicación puede ser bastante compleja, y el rango de usuarios que utilicen la aplicación, muy amplio, personas más o menos inteligentes, más o menos familiarizadas con los procesos propios de las aplicaciones, y otras variables que hacen que sea importante evaluar el integrar algún tipo de ayuda. En caso de aplicaciones móviles, dada su peculiaridad, es recomendable que la ayuda sea particularmente clara y concisa.

En caso de considerarse realmente necesaria una ayuda, se recomienda seguir la siguiente lista de recomendaciones:

- **Una imagen vale más que mil palabras.** Decantarse por explicaciones con fotografías, gráficos... en lugar de largas explicaciones con textos.
- Prescindir de los textos prescindibles, **no explicar lo que es obvio.**

- Los usuarios no leen: escanean la pantalla en busca de un indicio de información que resuelva rápidamente su problema. Si no lo encuentran rápidamente es probable que se marchen. Hay que ayudarles con diseños donde **los textos respiren: existan suficientes espacios en blanco, ayudados por listas, una jerarquización clara y gráficos y/o fotos que faciliten la búsqueda.**
- Preferiblemente **que la ayuda sea contextual**, es decir, explicar sobre la misma pantalla el uso de los elementos de dicha pantalla.
- El recurso de mostrar una pantalla de **ayuda al inicio de una aplicación debe evitarse** a no ser que se juzgue realmente necesario, y **en cualquier caso debe facilitarse al usuario de una manera muy rápida y clara el salir** de dicha ayuda.
- Son recomendables **comentarios y sugerencias sobre acciones**, incluso si son un brillo sutil.
- Es recomendable **añadir una ayuda para explicar una funcionalidad importante que esté disponible sólo a través de un gesto.**

Ejemplo:



3.4.12 Animaciones

Ayudan a dar una imagen de **dinamismo** a una aplicación, y **facilitan la usabilidad al fortalecer el uso metafórico**, por ejemplo, el concepto de pantallas siguiente y anterior, iconos de proceso en curso etc. No obstante no es recomendable su abuso, y en cualquier caso **no debe interrumpir el proceso normal de la aplicación de manera gratuita**.

Deben **seguir unos estándares**: por ejemplo, al deslizar una pantalla de abajo a arriba, se espera que dicha pantalla se mueva de abajo a arriba para seguir viendo el contenido de -por ejemplo- un scroll. Si la pantalla se moviera de manera diferente, prestaría a la confusión del usuario. En las animaciones hay que **evitar los efectos inesperados**.

Suavidad en los efectos, mostrar al usuario que efectivamente la aplicación reacciona ante el tacto de nuestros dedos en los elementos interactivos. Se recomienda evitar efectos que sobresalten o se muestren “a trompicones”.

4 Utilidades

4.1 Evaluación heurística

- **Facilidad de Uso.**
- **Utilidad.**
- **Estandarización.**
- **La página principal deja de indicar lo que puede hacer en el sitio.**
- **El acceso a los datos de la compañía** debe estar al alcance del usuario en cualquier sección o proceso de la aplicación.
- La información es **difícil de leer** o no responde a sus preguntas claves.
- Lenguaje **amigable, claro, y conciso.**
- **Tecnicismos no se recomienda el uso si no es imprescindible.**
- **Unificación en el trato (Tuteo o tratamiento de ud)** En formularios, lenguaje impersonal. Infinitivos en formularios, evitar tono imperativo.
- **Abreviaturas** no son recomendables. Si son necesarias, con tooltip explicativo.
- Gramática: primera letra irá en mayúsculas exceptuando las preposiciones (de, para, con, el, etc), contemplando las excepciones especificadas en el Libro de Usos (pág. 7)
- **Combinar creatividad y usabilidad eligiendo siempre la función sobre la forma.**
- Diseño no sobreenformativo, sin exceso de información, colores, movimiento y con zonas de descanso para la vista del usuario.
- Estructurar el contenido: jerarquización por medio de módulos, títulos y subtítulos.
- División por afinidad de datos. En formularios, estructurar en pasos.
- Elementos más importantes a la izquierda.
- El logotipo lleva al inicio de la web.
- Uso del Favicon.
- Navegación, debe saber dónde está en cualquier momento.
- Uso de Breadcrumbs si es necesario.
- Sección Mapa Web.
- Botón Volver.
- Uso correcto de la Ayuda: General, contextual, ayuda sobre un grupo lógico de formularios.
- Evaluar la claridad y lo intuitiva que es la ayuda. ¿Tiene buscador?
- Ayuda sobre elemento en concreto.

- Buscador.
- Evitar aspecto excesivamente publicitario.
- Banners (pág. 10).
- Prevención de errores.
- Heurísticas de usabilidad de Nielsen (pág. 11) :
 1. **Visibilidad del estado del sistema.**
 2. **Utilizar el lenguaje de los usuarios.**
 3. **Control y libertad para el usuario (deshacer y rehacer fácilmente).**
 4. **Consistencia y estándares.**
 5. **Prevención de errores.**
 6. **Minimizar la carga de la memoria del usuario.**
 7. **Flexibilidad y eficiencia de uso.** Uso de atajos de teclado.
 8. **Diálogos estéticos y diseño minimalista.**
 9. **Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores.**
 10. **Ayuda y documentación.**
- Evitarse el uso de **frames**.
- **Rapidez de carga (medir).**
- Limpieza de código. Uso de comentarios explicativos.
- **Bloques de texto** no se recomiendan líneas de más de 20 palabras.
- **Evitarse el justificado del texto**
- Uso de listas.
- **Diseño web adaptable (responsive web design)**
- Comprobar el **aumento de texto** sin que al hacerlo se pierda su legibilidad.
- En la versión web, comprobar que se ha minimizado la funcionalidad en su punto justo, y optimizado su peso.
- Comprobar soporte en navegadores.
- Nomenclatura.
- Tipografía: webs públicas y webs privadas. Uso de negrita y cursiva, mayúsculas, subrayado.
- Texto como imagen.
- Contraste textos vs color fondo.
- Comprobar etiqueta h en los títulos y subtítulos.

- Jerarquización semántica en HTML5.
- Optimización de imágenes. Uso de Alt y title. Viabilidad de sprites.
- Coherencia iconográfica.
- **Formularios (pág. 25)**
 1. Título descriptivo
 2. Información estrictamente necesaria
 3. Campos obligatorios diferentes, agrupados lógicamente y preferentemente en la parte superior del formulario
 4. No pedir los mismos datos
 5. Campos de textos, aceptar errores comunes. Mensaje informativo al pasar de un campo obligatorio sin rellenar a otro.
 6. Informar al usuario detalladamente de los **campos incorrectos, y no borrar la información que ya ha completado.**
 7. Evitar interrupciones visuales innecesarias.
 8. Evitar **pedir información que podemos obtener nosotros.**
 9. **Tabindex correcto**
 10. **Tecla Intro para la opción principal.**
 11. **Paso de campo**
 12. En los que sea factible por tener un número de caracteres fijo (DNI, código postal) , tras completarlos, pasar al siguiente campo automáticamente.
Uso de label en los elementos de formulario
 13. Contraseñas: control de fortaleza
 14. **Uso de teclas para facilitar las acciones** a los usuarios avanzados.
- Uso del botón destacado correctamente.
- Mensajes: estudiar su usabilidad. Errores significativos, reportar a teléfono.
- Mensajes error: Ser preciso: Hacer una descripción exacta del problema y su **solución**, siempre que sea posible.
- Mensaje de error de **protocolo estándar HTTP**: mostrar “página fuera de servicio”
- Mensajes de espera, el adecuado según lo especificado (pág. 32)
- Enlaces (pág. 33): tamaño y el estilo de la tipografía tiene que mantenerse.
- Los enlaces que no tengan un contenido textual claro deben llevar un **title** que especifique textualmente dónde enlaza el link.
- Evitar la **redundancia**, no enlaces tipo pulse aquí o más información.

- Listas de enlaces, sentido semántico más marcado.
- **Uso de enlaces de ancla** no recomendables.
- Enlaces rotos: comprobación.
- Coherencia elementos gráficos.
- Select: con múltiples opciones.
- Paginación.
- Tablas: evitar saltos de línea.
- Captchas.
- Uso de buscadores

4.2 Ubicación de los archivos en red

Logotipos, isotipo, Manual breve de identidad corporativa:

\\192.168.20.1\11.- Mkting y Comunicacion\LOGOS

Libro de usos:

\\192.168.20.1\3.- Desarrollo\Usabilidad-Diseno\Libro-de-Usos

Iconografía Portal Privado:

\\192.168.20.1\3.- Desarrollo\Usabilidad-Diseno\Libro-de-Usos\Maquetacion\material-adjunto\Iconos-Intranet

Css y fuentes optimizadas para la web y compatibles con la mayor parte de navegadores:

\\192.168.20.1\3.-Desarrollo\Usabilidad-Diseno\Libro-de-Usos\Maquetacion\maquetacion-intranet\css

Imágenes y otros iconos utilizados en el Portal Privado:

\\192.168.20.1\3.-Desarrollo\Usabilidad-Diseno\Libro-de-Usos\Maquetacion\maquetacion-intranet\css

4.3 Enlaces de interés

Comprobar la **compatibilidad de los diferentes navegadores con html5:**

<http://html5test.com>

Web del World Wide Web Consortium (W3C):

<http://www.w3c.es>

Compresor de archivos javascript y css:

<http://javascript-minifier.com/>

Analítica web:

<http://www.google.com/analytics/>

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

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

<http://clicky.com/>

<http://www.w3counter.com/>

<http://www.screamingfrog.co.uk/seo-spider/>

Medición de la velocidad de carga:

<https://developers.google.com/speed/pagespeed/>