

# Formularios usables: 60 Directrices de Usabilidad

## Artículos relacionados

[19-07-07] [Formulario con varios botones. Implementación usable ...](#)

[02-06-09] [Formularios accesibles según las WCAG 2.0](#)

Fuentes del artículo

## 60 Directrices para realizar formularios usables

### Generales

**1. Pida sólo la información absolutamente necesaria.**

**2. Infiera información a partir de otra disponible.**

Por ejemplo, la provincia se puede inferir del C.P.

**3. Reutilice los campos cuando sea posible.**

Por ejemplo, el email puede servirnos en ocasiones como nombre de usuario.

**4. No pida la información dos veces.**

Por ejemplo, si el usuario ha rellenado la dirección de facturación, no le obligue a volver a rellenar la dirección de envío si no es necesario, pregúntele si quiere que sea la misma.

### Textos

- 5. Proporcione un título al formulario** que exprese claramente su función.
  - 6. Si necesita instrucciones, que sean breves y comprensibles.**
  - 7. Utilice una nomenclatura clara y familiar**, sin tecnicismos ni extranjerismos.
  - 8. Sea consistente en el uso de los términos.**
- Es decir, use siempre las mismas palabras para los mismos conceptos.
- 9. No utilice preguntas complejas** ni haga pensar al usuario.
  - 10. Redacte siempre las opciones de forma afirmativa.**

Por ejemplo, junto a un *check* escriba "Deseo recibir el boletín" en vez de "No deseo recibir el boletín".

## Organización

- 11. Organice los campos en una sola columna de datos.**

Sin entrar en las razones de accesibilidad que lo justifican (ver "[75 Directrices de accesibilidad de Jakob Nielsen](#)") me cuesta muchas discusiones hacer comprender que, apelotonando los datos en la misma línea o colocándolos en varias columnas, se pierde tanto a nivel de usabilidad que no merece la pena el espacio vertical que se gana.

Como siempre, hay muchos contextos de uso y excepciones justificables, como los formularios que se rellenan de forma repetitiva y constante, pero la excepción nunca puede convertirse en norma.

*Una sola columna funciona mejor. Los formularios con dos columnas tienen más probabilidades de que los usuarios pasen por alto algunos campos, dado que crean un orden*

*ambiguo de lectura. Sus ojos se moverán hacia donde espera encontrar el próximo campo, que será habitualmente hacia abajo, en vertical. No esperan a que se les indique mediante el parpadeo del cursor dónde mirar.*

*[En "[Formularios largos: ¿una pantalla con scroll o varias páginas?](#)" de Usolab (resumen en español del artículo "[Caroline's Corner - Long Forms: Scroll or Tab?](#)" de Caroline Jarrett)]*

**12. Organice los campos en grupos lógicos**, utilizando para ello la mínima cantidad de elementos visuales (evitando así ruido visual).

**13. Agrupe, si es posible, los campos obligatorios al comienzo del formulario.**

**14. Evite fragmentar la petición de información.**

Por ejemplo, no pida por separado el tipo de vía, la calle, el número, etc. si no es estrictamente necesario.

**15. Proporcione un diseño ordenado, alineando verticalmente todas las etiquetas y todos los campos entre sí.**

Todos los campos deben estar verticalmente alineados entre sí a la izquierda.

**¿Cómo alinear las etiquetas entre sí:** a la derecha, a la izquierda o las colocamos encima del campo?

- Si tenemos que rellenar datos que son familiares (y no son muchos): Etiquetas en vertical encima del campo.
- Cuando necesitemos ajustar el espacio vertical: Etiquetas a la izquierda del campo, alineadas a la derecha.
- Hay que ajustar el espacio vertical, y los datos no nos son familiares o son complejos: Etiquetas al lado del campo, alineadas a la izquierda.

"Consejos para el diseño de formularios": resumen en español de las conclusiones de [Luke Wroblewski](#)

**16. Sitúe las respuestas de los campos *radio buttons* y *check box* después de los mismos.**

De esta manera se favorece la alineación vertical de todos los controles.

**17. Utilice etiquetas estándar para agrupar campos** y hacer más manejable la información(OPTGROUP, FIELDSET)

**18. Si se utilizan *radio buttons* o *checks box*** agrupe visualmente de forma clara y unívoca los distintos grupos de opciones.

**19. Distinga visualmente los campos deshabilitados siguiendo las normas de facto** (poniéndolos en gris claro)

## **Tipos de campos**

**20. El tamaño visible de los campos de texto** debe corresponderse con la longitud del contenido que ha de introducir el usuario.

**21. Homogeneice los anchos de los campos de texto** cuando estos sean similares (evitando así ruido visual).

**22. Dote a los campos de texto de flexibilidad** para que admitan los datos en cualquier formato.

Por ejemplo, un campo para introducir el número teléfono debería admitir paréntesis, guiones, espacios; un campo para introducir importes debería admitir decimales con punto o con coma, etc.

**23. Evite el uso de combos.**

No las use por ejemplo para seleccionar el país, fecha o profesión a no ser que sea estrictamente necesario, en cuyo caso incluya una opción del tipo “Otros” que pueda englobar casos no recogidos en

las opciones proporcionadas.

**24. Evite que las combos recarguen la página para rellenar otros campos,** pero cuando así sea, asegúrese de que el formulario conserva el mismo estado que tenía antes de recargar la página: con los mismos campos visibles o activos, y con todos los campos rellenos con los mismos datos que antes de la recarga.

**25. Si se utilizan combos o *radio buttons* seleccione siempre una opción por defecto,** asegurándose de que sea la más probable.

De lo contrario, el usuario no puede volver al estado inicial del formulario; si es necesario incluya una opción "Ninguna".

**26. Si se utiliza un *check box* para presentar una única opción que no es obligatoria** (recibir publicidad, aceptar unas cláusulas) no la marque por defecto.

**27. Si se utilizan *radio buttons* asegúrese de que todas las opciones son claramente excluyentes.**

- No los utilice cuando las respuestas sean más de tres y complejas, o más de cinco y simples.
- Siempre que se cumpla la regla anterior, utilice *radio buttons* en vez de combos

**28. Si un *radio button* tiene más de dos respuestas, colóquelas en vertical,** unas debajo de otras alineadas a la izquierda.

## Funcionamiento

**29. Valore la posibilidad de evitar, mediante JavaScript, que en determinados campos se pueda introducir determinados caracteres.**

Por ejemplo, que en el campo DNI sólo se puedan introducir números y letras, haciendo que el resto de

caracteres no se puedan teclear en el campo.

[A mí, personalmente, no me gusta esta práctica]

### **30. No implemente saltos automáticos del foco del formulario.**

Por ejemplo, en los campos de cuenta, no haga que el foco se mueva sólo al siguiente campo cuando se ha rellenado el anterior.

*Un error típico es introducir el salto automático entre campos de texto consecutivos y hacer innecesario el uso del tabulador.*

*Aunque este comportamiento puede parecer que facilita la tarea de introducción de datos, no es adecuado porque quita control a los usuarios, no es un funcionamiento estándar y es necesario mirar la pantalla para saber en que campo se está.*

*Todo ello puede provocar fácilmente errores, como por ejemplo, introducir datos pertenecientes a un campo en el siguiente cuando no se introduce el formato esperado por el salto automático.*

*[En ["Controles de formularios en diseño web, radio buttons, check-boxes..."](#) de Eduardo Manchón]*

Además es una práctica prohibida en las WCAG 2.0 "3.2.2 On Input: Changing the setting of any user interface component does not automatically cause a change of context unless the user has been advised of the behavior before using the component. (Level A)" aunque admiten que se implemente si se avisa antes al usuario de este comportamiento.

**31. Asegúrese de que la tecla "Intro" realiza la acción principal.**

**32. Evite, mediante JavaScript, que el usuario pueda impacientarse y enviar dos veces el formulario.**

**33. Al implementar la validación de los formularios (o al limitar el tamaño de los**

**campos)** piense si su formulario puede ser utilizado por usuarios de otros países.

Por ejemplo, el C.P. o el teléfono no tienen la misma longitud en unos países que en otros; por ejemplo, en España hay usuarios que no tienen DNI sino tarjeta de residencia.

## Ayudas

**34. Identifique claramente los campos obligatorios y los opcionales** mediante el literal (Obligatorio) u (Opcional), según si se van a marcar los obligatorios o los opcionales, colocando dicho literal detrás de la etiqueta del campo y por tanto antes del campo.

**Para saber si marcar los obligatorios o los opcionales** seguir las directrices de [Luke Wroblewski](#):

- Indique los campos obligatorios cuando sean menos que los opcionales.
- Indique los campos opcionales cuando sean menos que los obligatorios.

Para saber por qué poner el texto (Obligatorio) u (Opcional) después del literal y no después del campo, o por qué **se debe indicar mediante un texto y no mediante un asterisco**, leer "[75 Directrices de accesibilidad de Jakob Nielsen](#)"

**35. Incluir ayudas breves o ejemplos junto a los campos**, pero sólo cuando sea realmente necesario para saber cómo ingresar un dato.

**Asegúrese de que al leer en línea estas ayudas o ejemplos se lean antes que el campo**, por ello, un buen lugar para colocarlas es encima del campo. Para comprender por qué colocarlas en esta posición leer: "[75 Directrices de accesibilidad de Jakob Nielsen](#)"

## Botones

**36. No incluya un botón "Reset"** (es decir, de Limpiar o Borrar el formulario)

**37. En los formularios de un sólo paso evite tener un botón "Cancelar"** cuya función sea en realidad volver a la página anterior.

**38. Distinga entre las acciones primarias y secundarias** (volver, imprimir etc.) de su formulario.

Evite las secundarias, pero si ha de incluirlas distíngalas visualmente de forma inequívoca, destacando visualmente las primarias.

Por ejemplo, poniendo las acciones primarias como botones y las secundarias como enlaces.

**39. Coloque los botones o enlaces que realizan las acciones primarias** (por ejemplo el botón "Enviar") lo más cerca posible del último campo del formulario. No los separe del formulario mediante, por ejemplo, una línea.

**40. Dé un nombre adecuado a los botones del formulario**, relacionado con su acción y no de carácter general.

Por ejemplo, use "Buscar" en vez de un genérico "Aceptar".

## Errores

**41. Cuando se produzca un error al rellenar el formulario** proporcione en la parte superior del mismo, y con suficiente contraste, un listado de los errores. Por cada error indique qué campo lo ha provocado, por qué motivo, cómo solucionarlo y un enlace al campo.

**42. Destaqué los campos que han dado error pero no se base para ello únicamente en el**



**color.**

Acompáñelos de un icono de error que aparezca también en el resumen del comienzo de la página.

Repita el mensaje de error al lado del campo para no tener que volver a la lista inicial para saber qué error lo provocó. [Ver un ejemplo de cómo mostrar los errores de un formulario.](#)

**43. Cuando se produzca un error, el formulario no debe resetearse,** es decir, los campos no erróneos deben seguir manteniendo la información en ellos introducida.

**44. Redactar claramente los textos de error mediante términos claros, sencillos y no técnicos.**

No utilizar mensajes genéricos del tipo “No se ha podido enviar el formulario”.

**45. Evite validar los campos uno a uno, cuando pierden el foco,** mostrando inmediatamente un mensaje de error al usuario. A los usuarios les incomoda esta práctica.

## Feedback

**46. Cuando el usuario envíe el formulario, infórmele del resultado de su acción:** indíquele si se ha realizado correctamente, qué datos se han enviado, cómo puede ponerse en contacto con los responsables del sitio si ha habido problemas o para hacer un seguimiento del mismo, o cómo puede modificar los datos enviados.

**47. Si el proceso de envío es lento,** incluya en la página un mensaje de "enviando datos".

## Respuesta

**48. Informe a los usuarios** de por qué deben rellenar el formulario y cuándo y a través de que

medio recibirán una respuesta.

**49. Si es un formulario de contacto** envíe un email automático confirmado que se ha recibido.

**50. Si es un formulario de contacto**, asegúrese de que la empresa tenga los mecanismos necesarios para responder de forma rápida y adecuada al mismo.

## Legalidad

**51. Incluya las cláusulas de protección de datos** cuando sea pertinente.

## Accesibilidad

**52. Asocie explícitamente las etiquetas con sus controles** mediante LABEL y su atributo "for".

**53. Compruebe que el tabulador permite acceder a todos los campos** en el mismo orden que el visual.

**54. Mejore la experiencia del usuario mediante JavaScript y AJAX pero asegúrese que el formulario funcione correctamente sin ellos.**

**55. No establezca un límite de tiempo ajustado** para complementar el formulario.

## Formularios extensos

**56. Si los formularios son muy extensos** la solución no son las columnas, sino la división en páginas bien rotuladas que indiquen al usuario en que paso está del proceso (por ejemplo Paso 3 de 4).

- 57. Si el formulario se presenta en varias páginas** hay que seguir el lema 1 tema = 1 página.
- 58. El usuario debe poder volver a los pasos anteriores.**
- 59. No solicite información externa** en medio del proceso mediante la abertura de una ventana nueva del navegador.
- 60. Evite la utilización de pestañas** para crear formularios de varias páginas.

## Fuentes

- ["Best Practices for Form Design"](#) de Luke Wroblewski
- ["Formularios usables"](#) de Javier Cabrera
- ["Usabilidad al diseñar formularios de contacto"](#) de Reynier Matos Padilla
- ["Formularios"](#) en desarrolloweb.com
- ["Directrices de usabilidad para el diseño de formularios"](#) de Josep Casanovas
- ["Controles de formularios en diseño web, radio buttons, check-boxes..."](#) de Eduardo Manchón
- ["El principio de Visibilidad y el de Guiar en lugar de controlar: ¿son incompatibles?"](#) de Josep M. Junoy
- ["Formularios en la web, recomendaciones generales de diseño"](#) de Eduardo Manchón
- ["Usability and HTML Forms"](#) de Andrew Starling
- ["Reset and Cancel Buttons"](#) de Jakob Nielsen
- ["Checkboxes vs. Radio Buttons"](#) de Jakob Nielsen
- ["Should I use a drop-down? Four steps for choosing form elements on the Web"](#) de Sarah Miller and Caroline Jarrett
- ["Caroline's Corner - Long Forms: Scroll or Tab?"](#) de Caroline Jarrett
- ["Simplified Form Errors"](#) de Adam Kalsey
- ["Developing an Online Form"](#) de usability.org
- ["Usabilidad y Accesibilidad de sitios web. Lista de comprobaciones"](#) del Laboratorio Aragonés de Usabilidad
- Las [Elmer 2: User interface guidelines for governmental forms on the Internet \(PDF\)](#), guía del Gobierno de Noruega sobre la creación de formularios usables y accesibles.

## Artículos relacionados

[19-07-07] [Formulario con varios botones. Implementación usable ...](#)

[02-06-09] [Formularios accesibles según las WCAG 2.0](#)