

# Diseño: los usuarios al poder (I)

Por Daniel Mordecki  
12 de junio de 2001

Artículos relacionados

[Segunda parte de este artículo](#)

[Los objetivos del diseño](#)

[¿Qué es diseñar?](#)

**Cuando los que estamos en el negocio de e-business seamos capaces de hacernos una auto crítica sincera, de las primeras cosas que tendremos que reconocer es que fuimos absolutamente incapaces de generar una disciplina de diseño focalizada en los usuarios. Tal vez esa sea la causa para que más del 90% de los sitios Web no sean visitados por nadie.**

## Lo que entendemos por diseño

Cuando escribí el artículo "Algunos apuntes sobre diseño", en agosto de 2000, recibí numerosos comentarios. Por un lado, los amigos y conocidos vinculados al diseño, que criticaron duramente el llamado a no utilizar a discreción las nuevas tecnologías, el llamado a un diseño casi Franciscano. Por otro lado, recibí algunos mensajes de aliento y algún comentario del estilo: "usted me ayudó a decir que no a esa introducción Flash que mi diseñador le quería poner al sitio. Realmente yo no me animaba a decirle que no, y su artículo me dio los argumentos y la fuerza".

Después de mucho buscar llegué a 2 libros que recomiendo leer, y que son la base del presente artículo: Presos de la Tecnología, de Alan Cooper y Usabilidad. Diseño de sitios Web, de Jakob Nielsen.

Esa dicotomía en los comentarios, me llevó a tratar de encontrar más información y elaboración sobre el tema. Como todo en Internet, por cada hoja de información útil, existen mil hojas llenas de lugares comunes, frases hechas e información absolutamente descartable. Después de mucho buscar llegué a 2 libros que recomiendo leer, y que son la base del presente artículo: Presos de la Tecnología, de Alan Cooper y Usabilidad. Diseño de sitios Web, de Jakob Nielsen. El primero analiza el problema del proceso de diseño y el segundo analiza el problema de como analizar el uso de un diseño. Ambos tienen una premisa en común: en el mundo informático en general y en la Web en particular, los diseños son en su mayoría pésimos. Y ambos coinciden al tratar de identificar las causas: el usuario es el principal olvidado.

Diseñar es una actividad que nació mucho antes que la informática y que se ha dividido en numerosas ramas del saber y de la actividad profesional. El diseño en informática, no coincide con ninguna de ellas. No se trata de diseño artístico en ninguna de sus facetas. Tal vez en algún aspecto secundario el diseño artístico sea importante para el diseño informático, pero el tronco, el corazón del diseño en informática, no se trata de diseño artístico. Tampoco es diseño gráfico, ni es diseño industrial. El diseño informático es el proceso de análisis y creación de la interacción y la de los sistemas de computación con los seres humanos que los usan y la experiencia de éstos al utilizarlos. Es el diseño de la experiencia de la persona que usa el sistema, no es el diseño de la apariencia del sistema.

El diseño informático es el proceso de análisis y creación de la interacción y la de los sistemas de computación con los seres humanos que los usan y la experiencia de éstos al utilizarlos.

Cuando escribí el artículo anterior, no conocía esta definición, que me parece mucho más exacta y útil que la que utilicé anteriormente.

## La experiencia de los usuarios

Es muy difícil negar que la experiencia de los usuarios con las computadoras es nefasta. A los programadores (y yo me cuento con gusto en las huestes de los programadores) nos duele asumir esta realidad.

Pero que en el mundo moderno muchas tareas los obliguen a utilizar computadoras, no quiere decir que la tarea sea gratificante. Las pantallas están llenas de decenas de botones que el 90% de los usuarios no sabe para que sirven, ni sabe porque están, ni sabe como sacarlos. Y a medida que los distintos equipos industriales y aparatos domésticos incorporan microprocesadores, comienzan a comportarse de la misma forma desagradable y arrogante que se comporta el software. Así el microondas, el video, el lavarropas, la el despertador y otros aparatos otrora intuitivos y sencillos se han transformado en verdaderos desafíos. El 90% de los botones y funciones no se usan para nada. No solo el grupo de los capaces de dejar el video pronto para grabar un programa es muy pequeño con respecto a los usuarios de videograbadoras, dentro de ese grupo, los que jamás grabaron un programa equivocado, o un día equivocado, son casi un conjunto vacío.

Es muy difícil negar que la experiencia de los usuarios con las computadoras es nefasta.

## El cajero automático

¿Conoce usted algún aparato más tosco y arrogante que el cajero automático?

Comencemos por el principio: si yo no tengo cuenta corriente, ¿porqué tengo que contestar cada vez que lo utilizo la opción entre caja de ahorro y cuenta corriente? Y sin embargo el muy maldito lo sabe: si yo me equivoco, él se da cuenta, porque conoce perfectamente que yo no tengo cuenta corriente y me castigará duramente: como en el Juego de la Oca, caí en la casilla roja, que me manda nuevamente al comienzo. ¿Y su caja de ahorro es en pesos o en dólares?

¿Cuanto dinero quiere retirar? Recuerde que tiene que ser múltiplos de cien. Primero, si tiene que ser múltiplos de cien, podría por ejemplo, adelantar los dos ceros y no obligarme a poner hasta los centésimos (y si me equivoco, otra vez casilla roja!!!). Segundo: ¿porqué no me indica cuanto puedo sacar? el maldito lo sabe, jamás me dejará sacar un centésimo más de lo que tengo, pero no me lo dice. No me lo dice ni siquiera cuando me informa: "Ha excedido el limite de retiro". Castigo!!! Casilla roja!!! Vuelva a empezar. El sabe cuanto es, pero se lo guarda.

Nadie pensó en la interacción del usuario con el cajero cuando diseñó el sistema. Nadie se preguntó porque esa persona está allí, qué busca, cuáles son sus objetivos, y cómo se siente al interactuar con el cajero automático.

Tal vez esa sea la causa de que a pesar de la inversión y los años que han pasado, los cajeros sigan en la mayoría aplastante de los casos, desempeñando la misma y única función que el día que los introdujeron: retiros, sin haber cumplido la promesa de permitir que las personas realicen la mayoría de las transacciones fuera de las sucursales.

## La alarma del auto

Debe haber pocas situaciones que lo hacen sentir a uno tan tonto como cuando abre la puerta de su propio auto y comienza a sonar la alarma. Una de estas situaciones que lo hacen sentir a uno aún más tonto es dejar las llaves dentro del auto y cerrarlo, y sin duda la situación culminante es la combinación de ambas: la alarma activada y las llaves con el control dentro del auto, y éste cerrado.

La combinación bloqueo automático+alarma con control remoto, en su funcionamiento básico podrían evitar prácticamente en su totalidad estas situaciones. Cuando se trancan las puertas se prende la alarma, cuando se destrancan las puertas se apaga la alarma y ambas operaciones se realizan solamente con las puertas cerradas y con el control remoto de la alarma. Lo desafío a encontrar una situación en la que la alarma suene por otro motivo que no sea su fin específico, detectar un robo. Es fácil prever un estacionamiento de supermercado donde no suenen permanentemente las alarmas.

Sin embargo la realidad es muy distinta. Los programadores, como tenemos un microprocesador disponible, nos empecinamos en agregar "inteligencia" que lo único que hace es comprender

situaciones de borde, que se dan una de cada cien mil veces, y que tienen como consecuencia inmediata un estacionamiento lleno de alarmas sonando y una cantidad de conductores humillados.

Si la interacción es oscura, dura, frustrante, agregar funcionalidad hará que en vez de ignorar el 75% de las funciones, ahora ignore el 85% de las funciones, pero además que el 15% que se usa, sea aún más oscuro.

Tal vez, la solución más acertada que un programador pueda proponer es llenar la alarma de carteles que digan ¿Está Usted seguro?

## Diseño y Funcionalidad

Está ampliamente difundida la idea de que cuanto más funciones tiene un sistema, más útil será para los usuarios. Esta afirmación es en esencia falsa y es fácil comprobarlo. Por ejemplo, la mayoría de las funciones disponibles en el software de productividad personal (procesadores de texto, planillas electrónicas, etc.) no solo no son utilizadas por la mayoría de los usuarios. Ni siquiera sabe que existen. ¿Cuántas personas conoce que sepan que son los estilos de su procesador de texto, o que alguna vez hayan creado una tabla de contenidos utilizando la función del menú y no a mano?

Otro problema de las funciones es que en muchos casos, agregar funcionalidad no mejora sino que empeora el problema. Si la interacción es oscura, dura, frustrante, agregar funcionalidad hará que en vez de ignorar el 75% de las funciones, ahora ignore el 85% de las funciones, pero además que el 15% que se usa, sea aún más oscuro, los botones más chiquitos, los menús más largos, el área de la pantalla utilizable más pequeño.

## La función Guardar

Analicemos por ejemplo la función "Guardar". Esta función responde a un problema de diseño interno de la mayoría de las computadoras. Existen dos tipos de memoria: la memoria RAM, de acceso rápido y volátil, y el almacenamiento en discos, de acceso lento pero persistente. Debido a esto, desde el principio de los lenguajes de programación y sistemas operativos, éstos proveen funciones al programador para trasladar información de un sistema de memoria al otro de modo de poder hacer que los programas ejecuten eficientemente, y cumplan con la necesidad de guardar la información almacenada en memoria RAM para que se transforme en información persistente al apagado del equipo. El corazón de las funciones que manejan internamente esta problemática, tienen desde el comienzo los mismos nombres que hoy utilizan habitualmente los usuarios: Abrir, Guardar y Cerrar. (Open, Save or Flush, Close).

Sin embargo, desde el punto de vista del usuario, que internamente el equipo tenga memoria RAM, ROM, discos ópticos, magnéticos, o ranas y gatos, es exactamente lo mismo. Si bien fue aceptable que al grupo selectos de técnicos que comenzaron hace 50 años a utilizar computadoras, en el curso de capacitación se les instruyera sobre esta limitación interna de los equipos, no tiene ningún asidero cincuenta años después seguir obligando a los usuarios a entender esta diferencia para poder sobrevivir. Si alguna vez intentó explicar la noción de Guardar a un simple mortal, sabe de que hablo. El resultado es obvio, aunque pocos hablen de él y los que nos animamos lo hagamos con vergüenza: los usuarios se sienten humillados porque sistemáticamente pierden su trabajo. "Guardan" uno sobre otro, destruyendo inadvertidamente cosas que no querían o llenan sus discos de decenas de archivos sin después saber cual es cual.

La solución de los programadores tiene 2 líneas de trabajo. La primera es llenar de diálogos que preguntan ¿Está usted seguro? que en realidad quiere decir: Mire que si borra todo será su culpa. La segunda es agregar más funcionalidad. Ahora cuando utilizo la función guardar, puedo borrar otros archivos, crear otras carpetas, cambiar cosas de nombre, mover, y una cantidad de opciones que tal vez resuelvan otros problemas, pero no el específico: Guardar es un problema interno de la computadora y no un objetivo del usuario. El diseño se debería encargar de eliminarlo, no de perpetuarlo y agrandarlo.

La primera solución es llenar la interfaz de diálogos que preguntan ¿Está usted seguro? que en realidad quiere decir: Mire que si borra todo será su culpa.

## Cuando menos es más

La competencia en el mercado está basada en la cantidad de funciones que un producto o sitio Web tiene para el usuario. Esto es consecuencia de la carencia de diseño de la interacción, orientado al usuario, pero además es causa de que este problema se profundice. Nadie se anima a lanzar un procesador de texto que desafíe el dominio de Word bajo el lema: Solo 15 funciones que usted adorará. Todos creemos que para desbancar a Word, necesitaremos incluir cientos de funciones más que las que Word incorpora, y en mi opinión esto no es correcto. Word no escapa a la carencia de diseño de la interacción orientado al usuario, tiene una interacción recargada, inconsistente, llena de funcionalidad impresionante pero inútil para la gran mayoría de los usuarios. Estoy convencido que un producto que este diseñado realmente desde los objetivos de quienes lo usan, tal vez con muchísima menos funcionalidad, pero la exacta, que no haga sentir inútil a quien lo usa, podría realmente ser exitoso en el mercado.

Veamos el ejemplo de los puestos de autoconsulta de los Shopping. No creo que haya nadie que sea capaz de encontrar un local determinado a partir del mapa con la flechita, que invariablemente utilizan para contestar la pregunta de la ubicación de locales. Técnicamente es una solución interesante, muy cercano al desarrollo de un Sistema de Información Geográfica. Cualquier programador lo mostraría con orgullo, y sin embargo, es absolutamente inútil desde el punto de vista de los usuarios: no cumple con sus objetivos. La realidad es que los dos proyectos de autoconsultas en centros comerciales que conozco, fueron condenados al olvido después de fracasar con todas y cada una de sus promesas. Para mí la razón es sencilla: tenían una enorme funcionalidad, pero carecían completamente de diseño de la experiencia del usuario. Si se hubieran invertido los términos, hubieran sido mucho más exitosos con muchísima menos funcionalidad.

## Ese fantasma el usuario

Si quiere comenzar a desarrollar software o sitios Web realmente orientados al usuario, lo primero que debe hacer es eliminar la palabra "usuario" de su vocabulario y del de todo el equipo de trabajo. El usuario es un concepto "chicle", ambiguo, maleable, que no ayuda a identificar a quien va dirigido el trabajo, sino que ayuda a confundirlo todo. Se le puede atribuir en cada ocasión conocimientos distintos, atributos distintos e inclusive objetivos distintos, haciendo que en realidad el proyecto no esté destinado a nadie.

Para eliminar completamente la palabra usuario, y evitar este tipo de problemas, póngale nombre y apellido a esos usuarios, e intente describirlos con detalle.

La consecuencia más inmediata es lo que se llama "diseño autorreferenciado". Es decir, el diseño hecho a la medida de uno mismo. El programador desarrolla con su propia imagen en la imagen del usuario. La universalidad y omnipresencia de los botoncitos "Abrir", "Salvar", "Guardar" son un ejemplo de esta práctica. Es muy fácil detectar el diseño autorreferenciado: le presentan a usted (¡que no es un usuario promedio!), un nuevo sitio. El programador se para al lado, y cada vez que usted se tranca, porque no descubre como seguir, él le explica con naturalidad la lógica. El fracaso está asegurado, salvo que usted sea capaz de enviar al programador a que se pare al lado de cada usuario.

Para eliminar completamente la palabra usuario, y evitar este tipo de problemas, póngale nombre y apellido a esos usuarios, e intente describirlos con detalle. Por ejemplo, supongamos que usted está haciendo un portal médico. En vez de decir el usuario, podríamos llamar a este individuo Mario del Buono

Mario del Buono es Gastroenterólogo. Es un profesional destacado, y es además docente grado 4 en la universidad. Tiene 54 años, es casado, tiene 3 hijos, dos nietos y uno en camino. Tiene un Nissan Sentra Super Saloon azul oscuro. Adora el silencio del auto, y que hay que llevarlo poco al mecánico. Mario es un estudioso, tiene una enorme biblioteca, y ha publicado 2 libros. Tiene una computadora en su casa, que ya casi no usa y un notebook, que lleva a todos lados. A pesar de que hace mucho que utiliza computadoras, el 70% del trabajo lo realiza en un procesador de texto, donde reconoce que usa solo las funciones elementales. El resto del tiempo lo ocupa el correo electrónico y una planilla de cálculo para manejar sus gastos, honorarios y ahorros.

Ahora es mucho más fácil imaginar lo que Mario quiere, lo que necesita, lo que no quiere y lo que no necesita. Mario es tangible, y si no lo es, cuelgue una foto de un médico en bata que diga "Mario del Buono, Gastroenterólogo"

## El reparto

El resultado final de esta técnica es contar con un reparto, tal como en una obra de teatro o película de cine, antes de digitar la primera línea de código. Encontrar el reparto no será fácil. Debe limitarse a un personaje principal para cada área del sistema.

Por ejemplo, el portal tendría 3 personajes principales:

Además de Mario, podríamos incluir a Teresa Hernández, administradora del portal, con una larga experiencia como operadora de sistemas, pero en su primer proyecto Web. Es Analista de Sistemas, acostumbrada a manejar complejos sistemas de respaldo, controlar el desempeño de servidores y medir el tiempo de respuesta. Teresa tiene 24 años, y un novio que insiste en que se casen lo antes posible. Ella está esperando a recibirse de Ingeniera antes de la boda.

El tercer personaje es Fabio Daremberg, Web Master. Fabio tiene 32 años, es soltero y trasnochador, siempre con una compañía agradable y distinta. Se viste de negro, recita a Neruda de memoria y escribe solo con tinta verde, como él. Ama el PhotoShop, y es capaz de describir cada una de las cosas que se puede hacer con cada una de las versiones. Solo usa Mac, una cuestión de principios.

El reparto no puede tener más de 7 a 10 personajes y solo 1 a 3 principales. No intente diseñar para todo el mundo. El resultado es una de esas palas de campamento que son además martillo, hacha, remo y destornillador a la vez: no hacen ninguna de las tareas bien, ni siquiera son útiles como pala.

Tener un reparto permite darse cuenta con claridad que es lo que se debe hacer y que no. Está claro que Mario no quiere problemas, quiere ir directo al grano y poco le importa la técnica. Por más funciones que pongamos, usará solo algunas, tal como lo hace desde hace años con su procesador de texto.

La pregunta: ¿Necesitará el usuario una búsqueda con operadores lógicos? es completamente distinta si se plantea ¿Necesitará Mario una búsqueda con operadores lógicos?. Un programador contestará probablemente a la primera con un si, dado que al final, la lógica es la base de su tarea, y el jamás utilizaría un sistema que no le permita programar funciones que no incluyan and, not, or, xor y si quiere realmente posar de nerd, exigirá un operador nand. Mario contestará que apenas entiende como funciona la búsqueda. Que en general pone una palabra y salen cientos de miles de documentos, y que no va a gastar un segundo de su tiempo en complicarse más la vida. Evidentemente lo que Mario necesita es que mejoren la interacción en la pagina de resultados y no que agreguen funcionalidad a la búsqueda.

La pregunta: ¿Necesitará el usuario una búsqueda con operadores lógicos? es completamente distinta si se plantea ¿Necesitará Mario una búsqueda con operadores lógicos?.

## Una razón imperativa para actuar

A pesar de que la publicidad insista en que los mortales debemos navegar por Internet, después de la experiencia de sentarse ante un navegador y preguntarse ¿Ahora qué?, todos nos damos cuenta de que navegar en internet es una tarea, o una herramienta, no un objetivo.

Las personas hacemos cosas cuando tenemos una razón imperativa para actuar. Cuando existe un conjunto de causas que generan una razón imperativa, nuestro usuario, cliente, Mario, o como quiera que lo llame, actuará. El diseño debe entender el objetivo, de modo de analizar y crear una interacción que acerque al usuario a su objetivo. La funcionalidad crea herramientas. El diseño crea experiencias. Si usted busca un destornillador determinado, cuantas más herramientas tenga la caja de herramientas, más difícil será la búsqueda. Y si las herramientas son todas destornilladores, peor aún. Tener mucha funcionalidad, no tiene relación directa con los objetivos de los usuarios. Estos se sentirán más satisfechos cuando encuentren exactamente la cantidad necesaria, ni una más, ni una menos, de funciones que los ayuden a alcanzar sus objetivos.

Si bien para alcanzar los objetivos hay que desarrollar tareas, no hay que confundir uno con otro. Cuando el padre deja sin televisión a su hijo, su objetivo es educarlo para que pueda ser feliz en su edad adulta, su tarea es mostrarle que una determinada actitud tuvo consecuencias

Focalizarse en el objetivo

que tendrá que asumir. Los objetivos tienden a permanecer, a ser estables en el tiempo. Las tareas tienden a cambiar. El padre durante toda la niñez de su hijo, intentará educarlo para que sea feliz en el futuro. Las tareas que desarrollará para ello serán múltiples y variadas.

Focalizarse en el objetivo de sus personajes, entender este objetivo y concentrarse en él, le permitirá abordar con creatividad y libertad el análisis de las tareas y la funcionalidad que las implementa. Podrá libremente eliminarlas, crear nuevas, o cambiarlas completamente.

de sus personajes, entender este objetivo y concentrarse en él, le permitirá abordar con creatividad y libertad el análisis de las tareas y la funcionalidad que las implementa

En el ejemplo de la función "Guardar", el objetivo del usuario es crear, administrar y probablemente imprimir sus documentos. "Abrir", "Guardar" y "Cerrar" archivos no está dentro de su lista de objetivos. Podría por ejemplo eliminar completamente el menú de archivos y hablar de Versiones. Crear una vista de versiones por fecha, y una vista de versiones por documento, y almacenar automáticamente las versiones que se generan cada vez que se trabaja sobre un documento. Esta solución dista mucho de ser adecuada, pero está mucho más orientada a los objetivos del usuario, y seguro será mucho más apreciada por él.

En la Web el problema de encontrar la razón imperativa que el cliente tiene para actuar es aún más importante. El cliente tiene que decidir venir a nuestro sitio. Tiene que tener un motivo, y después que vino a mi sitio, yo tengo que darme cuenta de ese motivo, ese objetivo y ayudarlo lo más rápidamente a resolverlo. Todos los sitios que tienen una presentación al comienzo, en general desarrollada en Flash, violan este criterio. Ignoran los objetivos de sus clientes y los condenan a esperar 1 minuto a que se cargue y otro minuto a que se despliegue, una presentación siempre igual, que no dice absolutamente nada útil para los objetivos de sus clientes. Se arriesgan además a mensajes horribles como "Su sitio no soporta Flash 5", un enorme cuadrado negro o que la presentación cuelgue la maquina del cliente, lo que determinará automáticamente la condena del sitio al olvido. Un cartelito que diga "Saltar la presentación" es apenas una forma errónea de sacarse un poco de culpa. Flash es un producto fantástico, que genera herramientas poderosas, que ayudan a resolver algunos de los objetivos de los clientes. Eso no lo exime a usted de encontrar los objetivos de sus clientes, las razones imperativas que lo llevaron a su sitio, y hacer que su experiencia sea placentera, y que resuelvan lo mejor y más rápido posible sus problemas e inquietudes.

Todos los sitios que tienen una presentación al comienzo, en general desarrollada en Flash, violan el criterio de la razón imperativa para actuar. Ignoran los objetivos de sus clientes.

## Los escenarios

Todo buen elenco necesita un escenario para actuar. Su reparto también. Para entender como hacen los personajes para cumplir con los objetivos, es necesario describir con detalle los escenarios en los que se está actuando. Una vez más esto ayuda a entender como es la experiencia del usuario y como se realiza la interacción hombre-sistema, a los efectos de desarrollar un diseño que haga exitoso su proyecto.

Por ejemplo, en el portal médico, un escenario posible es que Mario haya tenido un paciente con una dolencia poco común y esté buscando dos cosas: material para tratar de diagnosticar mejor la enfermedad, así como otros médicos que lo puedan ayudar.

Otro escenario posible es que Mario reciba como usuario registrado una consulta y entre al portal a contestarla, ya que sabe que la participación activa en la comunidad le traerá beneficios a la hora de pedir ayuda.

## Escenarios de uso diario

En todo sistema existen 2 o 3 escenarios que representan las interacciones básicas que realizará el usuario con frecuencia. Se trata de apenas 2 o 3 escenarios. Inclusive muchas veces un solo escenario es suficiente. Nunca son 10.

Independientemente de la cantidad de funciones que su software o sitio Web incluya, el 80% del



tiempo del usuario se desarrollará en esos 2 o 3 escenarios por lo que el 80% del tiempo debe destinarse a diseñar la interacción en ellos. La distribución homogénea del esfuerzo de diseño es un error tan frecuente como nocivo. Concéntrese en encontrar sus personajes principales, identificar sus objetivos y diseñar la interacción de los dos o tres escenarios de uso diario. También allí está escondido el 80% del éxito.

## Escenarios de uso necesario

Los escenarios de uso necesario abarcan las interacciones imprescindibles, pero poco frecuentes. Si bien el diseño de esas interacciones debe ser cuidado, si usted enamoró al personaje de su sistema o sitio Web en los escenarios de uso diario, el usuario entenderá que es imprescindible desarrollar una determinada tarea una vez cada tanto, y estará dispuesto a una interacción no tan agradable. Sumado a ello, el hecho de que sean poco frecuentes, hará muy difícil que el usuario recuerde lo aprendido de una interacción a la siguiente. Como corolario, ya hemos gastado en los escenarios de uso diario el 80% del esfuerzo de diseño, por lo que nos queda solo un 20%.

Mientras que el objetivo de los escenarios de uso diario debe ser fascinar al cliente, los de uso necesario deben tener como objetivo que éste realice su tarea sin frustrarse.

## Escenario de caso límite

Quienes alguna vez programamos, sabemos que los buenos programadores son aquellos que son capaces de prever todos los casos límite, de modo que su software no falle jamás. Cuando el sistema operativo se cuelga, es porque se excedió alguna condición de borde que algún programador no previó y esto habla muy mal de ese programador.

Su sistema o sitio no tendrá éxito o fracasará en función del mensaje de error cuando los resultados de la búsqueda exceden 216 ocurrencias.

Sin embargo a pesar de que los programadores tengan que seguir previendo el 100% de los casos límites, desde el punto de vista del usuario, es prácticamente innecesario el diseño de interacción en estos casos. Su sistema o sitio no tendrá éxito o fracasará en función del mensaje de error cuando los resultados de la búsqueda exceden 216 ocurrencias. Esto le sucederá a un usuario por año, y es prácticamente innecesario abarcar estos problemas desde el punto de vista del diseño de la interacción, a pesar de que consuma gran parte del tiempo de programación.

## Diseño e Interfaz, el orden de los factores

El diseño de la interacción y la experiencia es al diseño de la interfaz, lo que la Arquitectura es al Diseño de Interiores. Mientras que la Arquitectura **precede** a la construcción de la vivienda, la decoración de interiores **la sigue**.

El diseño de la interfaz es efectivamente la parte del diseño cercana al diseño gráfico y al diseño artístico, pero ni reemplaza ni se opone al diseño de la interacción. La realizan personas distintas, con conocimiento distinto, con objetivos distintos y en momentos distintos.

Cuando usted le habla a un programador experiente de diseño de la interacción, automáticamente contestará que se trata del diseño de la interfaz del usuario. Según Alan Cooper, es como llamar a los carpinteros para que den forma al encofrado, después que se vertió y fraguó el cemento. Una tarea tan inútil como improductiva.

El diseño de la interacción y la experiencia es al diseño de la interfaz, lo que la Arquitectura es al Diseño de Interiores. Mientras que la Arquitectura precede a la construcción de la vivienda, la decoración de interiores la sigue.

El **orden** es vital: **diseño de la interacción, codificación, diseño de la interfaz, testeo y depuración, entrega del producto**. Este orden tiene una lógica y consecuencias favorables a su proyecto

## El producto terminado

Si su mecanismo es **definición de la lista de características de su aplicación, codificación,**

**lanzamiento de un beta, vuelta al principio**, usted carece al comenzar de una idea de cual es el producto terminado. Esto determina durante el proceso de desarrollo de su aplicativo o sitio Web de una negociación interminable entre el gerente de desarrollo, los programadores, el área de Marketing y el area financiera que paga las cuentas. Los usuarios no tienen representante en esa negociación. A medida que los plazos se acortan, los features se recortan, no en función de su importancia, o lo vital que son para el usuario y su experiencia con el producto, sino en función del tiempo que lleve implementarlos y del costo que tengan asociados, estos son los dos parámetros más importante bajo la presión de los plazos.

También se sustituirán determinadas formas de interactuar por otras más fáciles de codificar, pero no en función de la interacción del usuario. Por ejemplo, existen además de las API específicas de Windows para manejar la apertura, grabación y cierre de archivos, cientos o miles de objetos, código o controles para realizar esa tarea. Todos comparten la idea de hace 50 años de obligar al usuario a entender qué es un disco duro y qué es la memoria RAM, y entender las funciones open, save, close, etc. del lenguaje de programación. Tal vez no sea capaz de encontrar código reutilizable para manejar los documentos desde el punto de vista de las versiones, o del que usted definió como óptimo. El cambio acorta los plazos, pero frustrará una y otra vez a su usuario.

El orden es vital: diseño de la interacción, codificación, diseño de la interfaz, testeo y depuración, entrega del producto. Este orden tiene una lógica y consecuencias favorables a su proyecto.

El diseño de la interacción y la experiencia del usuario, la documentación amplia y detallada de este diseño por escrito, utilizando texto, imágenes, storyboards, etc. Dará una idea acabada antes de empezar de como es el producto terminado. Esto es una herramienta invaluable para planificar la programación, para evaluar los tiempos y eliminar las disputas interminables sobre las características a incluir o eliminar.

## El proyecto más barato

La parte más cara del proyecto es sin duda la programación. Probablemente con un equipo de dos diseñadores de la interacción, se diseñe una aplicación que involucre a una decena de programadores. Es mucho más económico planificar antes, que intentar modificar durante. No es solamente un problema de experiencia del usuario. Las idas y venidas, los cambios sobre la marcha, rehacer lo ya hecho, es mucho más caro.

El mecanismo de lanzar los productos sin terminar al mercado, para que los gritos pidiendo piedad de los usuarios indiquen el camino a seguir, es una estrategia descabellada. Recuerde que Microsoft tiene cifras de dinero casi ilimitadas, y una aparente inmunidad a las críticas de los usuarios, aún las mas duras. El resultado en una interacción pobre, una experiencia frustrante y un código de dudosa calidad, con productos como Windows 98 que conservan en su interior el código escrito 18 años antes para DOS. Recuerde que una vez que hay código en el proyecto, el concreto empieza a fraguar y es ya imposible cambiar su forma.

Interactuar dos, tres, cinco veces con los usuarios, desnudando la falta total de consideración de criterios de diseño de interacción no solo es malo, sino que además es muy caro.

## El proyecto más exitoso

La guerra de features entre los productos y sitios del mercado solo es una demostración de lo que en Marketing se llama miopía. Si estoy en el negocio del software, estoy en el negocio de la programación. Cuanto más programe, cuanto más difícil sean esos programas, mejor estará posicionado mi programa. Dicho en otras palabras: cuanto más features tenga, cuantos más cuadros de dialogo tenga, mejor será el programa. Definitivamente falso.

El proyecto más exitoso será el que cumpla mejor con los objetivos de sus clientes. Todo lo demás es adjetivo. Esto no quiere decir que lo que hay que hacer es eliminar features. Esto quiere decir que es un error quedar atrapado en la guerra de los features. Esta forma de pensar va en contra de la concepción de casi todos los desarrolladores de productos de software y sitios Web. Pero está



profundamente basada en la realidad, y en la práctica de los usuarios.

## Los usuarios al poder

Si no nos guiamos por las declaraciones de intención y los folletos publicitarios, los usuarios no están representados en el proceso de desarrollo, salvo que se incluya una fase de diseño de la interacción y la experiencia del usuario antes de que se codifique la primera línea.

En un mercado tan competitivo, tan exigente, con procesos de desarrollo tan demandantes, quienes no están fuertemente representados, no cuentan. Y los usuarios son sin duda los parias de este proceso de desarrollo. Después que está todo terminado, los invitan a 2 o 3 focus groups, para analizar su comportamiento, y a partir de ello cambiar algunas cosas de la interfase, que milagrosamente solucionará todos los problemas que surgen de haber obviado la fase de diseño de la interacción.

Centrar el proyecto en el cliente, partiendo del desarrollo de la interacción y la experiencia del usuario, a través de la creación de personajes y un reparto, la identificación de sus objetivos, así como de los escenarios en los que tendrán que actuar para cumplir esos objetivos, es una apuesta a un éxito más económico y más probable.

Artículos relacionados

[Segunda parte de este artículo](#)

[Los objetivos del diseño](#)

[¿Qué es diseñar?](#)

# Diseño: los usuarios al poder (II)

Por Daniel Mordecki  
12 de junio de 2001

Artículos relacionados:

[Primera parte de este artículo](#)

[Los objetivos del diseño](#)

[¿Qué es diseñar?](#)

**Cuando los que estamos en el negocio de e-business seamos capaces de hacernos una auto crítica sincera, de las primeras cosas que tendremos que reconocer es que fuimos absolutamente incapaces de generar una disciplina de diseño focalizada en los usuarios. Tal vez esa sea la causa para que más del 90% de los sitios Web no sean visitados por nadie.**

## Un sitio bien diseñado

En la parte I de esta nota, hicimos hincapié en las características más importantes que hacen a un buen diseño. Fundamentalmente, el hecho de que el diseño es un problema que debe ser resuelto antes de generar la primera línea de código, y que el fin del diseño debe ser que la experiencia del usuario al interactuar con nuestro sitio sea óptima: es decir, que nuestros usuarios y clientes consigan sus objetivos con la mayor rapidez y el mayor grado de satisfacción posible.

Las posibilidades que brinda hoy Internet, y que brindará por algunos años más, son mucho más limitadas que las promesas que brinda Internet.

Lamentablemente, el mundo real dista mucho del mundo ideal. Las posibilidades que brinda hoy Internet, y que brindará por algunos años más, son mucho más limitadas que las promesas que brinda Internet. A eso se suma que la base de hipertexto y las modalidades de despliegue y visualización de sitios Web generan modalidades de lectura e interacción del usuario con la información absolutamente particulares.

Internet es a nivel de interacción del usuario un medio completamente nuevo y requiere un formato completamente nuevo.

Se insiste en compararlo con la televisión porque tiene pantalla y con la prensa porque publica textos con fotos. Son comparaciones extremadamente pobres. Si tomamos como ejemplo el cine y la televisión, ambos comparten una cantidad enorme de características que derivan de que ambos son la conjunción de imagen en movimiento y sonido. Sin embargo el formato de los productos cinematográficos es distinto que el de los productos televisivos. El cine tiende a generar productos más extensos que la televisión (del orden de las 2 horas), de un solo episodio en el que se completa el ciclo introducción-desarrollo-cierre. Los usuarios los consumen en un recinto cerrado, a oscuras, sin ningún otro foco de atención que la pantalla y el audio. La televisión produce productos más breves (del orden de los 45 minutos) que se separan en bloques entre los cuales se despliega publicidad. Si el tema se extiende, entonces el producto se puede dividir en episodios. Los usuarios los consumen en ambientes no dedicados, con otros focos de atención activos. Incluso muchas veces la emisión televisiva es un foco de atención secundario y no primario. Ni hablamos todavía de informativos, video clips, etc. En resumen, dos medios aparentemente iguales presentan formatos completamente distintos. Es imprescindible diseñar sitios que respeten a la vez el formato de Internet y las limitaciones que la tecnología disponible impone a ese formato.

## Los criterios de Usabilidad

Una vez que el sitio se pone al aire, podemos llamar Usabilidad a la facilidad con que los usuarios del sitio puedan cumplir con los objetivos

Podemos llamar

que se proponen. El objetivo del diseño es entonces alcanzar el mayor grado de Usabilidad posible.

La Usabilidad es extremadamente importante y no es frecuente encontrar en los proyectos referencias a ella. No se trata de que sea fácil de usar, ni que el contenido sea atractivo, ni de que se puedan "hacer muchas cosas". Eso no alcanza: se trata de que los usuarios alcancen sus objetivos, que es mucho más importante.

Usabilidad a la facilidad con que los usuarios del sitio puedan cumplir con los objetivos que se proponen.

Si su sitio Web no alcanza un grado suficiente de Usabilidad, sus usuarios no podrán alcanzar sus objetivos, es decir: sus clientes no podrán comprar los productos que necesitan, sus lectores no encontrarán las noticias que buscan y sus empleados no encontrarán la información que tienen que incluir en la propuesta.

Si los usuarios no pueden alcanzar sus objetivos, usted jamás será exitoso, independientemente de lo fascinante del look de su sitio y de lo relevante del contenido. Lo más triste, es que muchas veces, las fallas de Usabilidad pueden ser detectadas y corregidas con un costo bajísimo si se encara el problema del diseño de la forma adecuada.

## El tiempo es tirano

El criterio de Usabilidad rey en Internet es el tiempo de descarga de las páginas. Internet es lenta. Por cada usuario VIP con una conexión realmente rápida, hay 100 que tienen anchos de banda lamentables. Esto conduce a la realidad de que Internet es lenta.

Ya en 1968 se estudiaba la relación de la Usabilidad con el tiempo en que reacciona un sistema, en aquel momento trabajando sobre los sistemas centrales con terminales y los tiempos de respuesta. Los resultados son los siguientes:

- **Menos de 0.1 segundos:** Por debajo de una décima de segundo el usuario percibe funcionamiento instantáneo. Es lo que sucede por ejemplo al digitar en un procesador de texto. La percepción es que la computadora responde instantáneamente cada vez que se presiona una tecla.
- **Menos de 1 segundo:** Por debajo de un segundo, el usuario percibe que la computadora trabaja o "piensa" para responder al estímulo, pero la interacción es fluida. El usuario percibe la demora, pero mantiene el 100% de atención en la tarea que estaba realizando.
- **Menos de 10 segundos:** Por debajo de los 10 segundos, la interacción sigue siendo fluida, pero el usuario abre un foco secundario de atención. La atención primaria sigue estando en la tarea que está haciendo, pero durante la espera es muy alta la probabilidad de que preste atención a otra cosa, generando un segundo foco.
- **Menos de 1 minuto:** A medida que nos alejamos de la marca de los diez segundos, se incrementa la posibilidad de que el usuario genere un nuevo foco de atención y que este se convierta en primario, relegando la tarea que estaba realizando a un segundo plano. Podemos decir que todavía existe un flujo de interacción, pero este es entrecortado con fuerte probabilidad de que se interrumpa definitivamente
- **Más de 1 minuto:** si los tiempos de respuesta superan el minuto, podemos decir que es imposible mantener un flujo de interacción.

De este estudio puede concluirse que para que un sitio tenga un gran flujo de navegación por parte de los usuarios, debería tener tiempos de respuesta inferiores al segundo. En nuestras latitudes, esto es sencillamente una Utopía. Es necesario entonces tener como norte el escalón siguiente: menos de 10 segundos. Cada página de su sitio debe estar concebida para descargarse en menos de 10 segundos, inclusive si el usuario tuviera una conexión lenta (por ejemplo un ancho de banda real entre 20kbps y 28kbps) Esto no es imposible: Yahoo! se descarga en menos de 8 segundos en una conexión de este tipo. Y esto es una gran herramienta: exija a sus técnicos y diseñadores a que cada página que construyen se descargue en un tiempo no mayor que el doble del que se descarga Yahoo! en cualquier equipo.

Cada página de su sitio debe estar concebida para descargarse en menos de 10 segundos, inclusive si el usuario tuviera una conexión lenta

Me animo a afirmar que prácticamente cualquier concesión es válida si permite que una página se descargue en menos de 10 segundos. Eliminar un gráfico, bajar la cantidad de colores de un logotipo,

cambiar dibujos por textos, eliminar animaciones, son todas decisiones que los diseñadores gráficos odian, pero que son imprescindibles en el mundo de hoy.

El resultado de no tomar en cuenta el problema del tiempo genera sitios Web no navegables. No importa cuanto contenido valioso tenga, cuanta publicidad se haga o cuanto dinero se gaste en hosting y sistemas. Si el sitio es no navegable, el fracaso está asegurado.

## A la búsqueda de la obviedad

Usted no tendrá acceso a capacitar a sus usuarios, ni podrá pedirles que lean la ayuda. Probablemente tenga uno o dos minutos en los cuales el usuario decida si su sitio es o no útil para él, y si la decisión es negativa, jamás tenga chance de revertirla. A esto se suma el hecho de que su capacidad de comunicar es mínima y cara. La búsqueda de la obviedad, la capacidad de hacer que las cosas sean obvias cuando se puede, le ayudará a concentrar esa pequeña capacidad de comunicación y capacitación de los usuarios en los temas que realmente importan.

En muchas actividades, lo obvio es cursi, o irrelevante, o contraproducente. En Internet ocurre todo lo contrario.

## Los links

El comportamiento default de los links es: tener color azul y estar subrayados cuando no fueron visitados, cambiar el color a violeta cuando fueron visitados. Esto es lo obvio. Existirán en cada proyecto un millón de argumentos para eliminar el subrayado, para cambiar el color, o para hacer una enorme serie de modificaciones. Sin embargo, cualquier alteración a este criterio requerirá que el usuario deba darse cuenta del cambio y que gaste unos segundos de esos 2 minutos que dedicará para evaluar nuestro sitio a comprender que los links son blancos y los links visitados son verdes. Si yo tuviera que tomar la decisión, cambiaría todo lo demás, para conservar esos preciosos segundos y poder utilizarlos para un mejor fin que el de comprender un comportamiento no obvio.

Otra cosa importante relativa a los links es que el comportamiento obvio es que conduzcan a donde deben conducir. Un link de dice "Diseño: los usuarios al poder (II)" no debe conducir a ningún otro lado que no sea a la página que contiene el artículo. No es válido que conduzca a la home page del sitio, ni a la parte I del artículo, ni a ningún otro lado. Descargar una página lleva tiempo, y no es una gran idea defraudar al usuario alejándose de lo obvio.

## Los títulos y titulares

En la prensa es necesario desarrollar titulares que atraigan a los clientes a comprar el periódico. En internet es distinto. Por ejemplo "La desaparición del Príncipe" puede ser un tratado de historia sobre la muerte de algún noble, una nota sobre Enzo Francescoli y el momento en que abandonó el fútbol profesional, o un cuento de hadas para niños. Si se trata de un link a una nota, el usuario no sabrá el contenido hasta que no clickee el link y descargue la página, y con un título tan ambiguo, muy pocos se formarán una expectativa acertada de lo que les espera.

En la prensa es necesario desarrollar titulares que atraigan a los clientes a comprar el periódico. En internet es distinto.

El usuario tiene que ser capaz de generarse una expectativa acertada con la información que dispone en la página actual, el resultado tiene que ser obvio. Si desea mantener el titular, agregue un acápite, un colgado o un subtítulo que incorpore la información necesaria para generar la obviedad. Hay muchos diseñadores que piensan que obligar a los usuarios a navegar mucho para obtener la información es un gran logro, porque generan muchos "page views". Para mí lo único que consiguen es frustrar a los usuarios y que estos no vuelvan nunca más al sitio.

## El entorno y la navegación

Para el usuario tiene que ser obvio el entorno o contexto en que se encuentra la página con respecto al sitio y las opciones de navegación que de ella se desprenden. En un libro, la pagina 87 precede a la 88 y sigue a la 86, no hay otra opción. El libro, salvo casos excepcionales

En un sitio Web, por más

como Rayuela de Cortazar, se navega de principio a fin. En un sitio Web, por más definiciones que se hagan en cuanto a la navegación deseada, el camino de navegación lo definirá cada usuario en el momento. Esto se refuerza con los buscadores, tanto externos como internos.

definiciones que se hagan en cuanto a la navegación deseada, el camino de navegación lo definirá cada usuario en el momento.

Un buscador puede conducir a un usuario a una página muy profunda en el árbol del sitio y que esa sea la puerta de entrada al sitio. Es frecuente encontrar una pagina así, que ni siquiera indica a que sitio pertenece. Es un error. En cada pagina del sitio hay que dotar al usuario de las herramientas para que comprenda dónde está ubicado en ese instante y hacia dónde puede seguir navegando. Y estas herramientas tienen que ser obvias, no pueden requerir explicaciones.

Muchas veces es muy fácil mejorar enormemente la navegabilidad de un sitio, con soluciones extremadamente sencillas. Por ejemplo, la solución llamada Migas de Pan, en alusión a las migas que dejaban Hansel y Grettel en el cuento para recordar el camino sería algo así:

Home >> Productos >> Especificaciones >> Video 3248

En apenas un renglón de texto usted tiene una idea exacta de la profundidad de la página, de la rama del árbol del sitio en la que está inserta y una opción excelente de navegabilidad para ascender a los niveles superiores con un solo click. Complementando esta solución con menús que den idea de la amplitud en cada nivel se puede dotar al usuario de un marco obvio de navegabilidad que genere una agradable Usabilidad de su sitio.

## Los que no pueden

Dado que la renovación de navegadores es lenta, la migración de las tecnologías de comunicación hacia anchos de banda más elevados está lejos, la proliferación de versiones distintas de navegadores y plug-ins y otros factores generan un panorama extremadamente variado de opciones para los equipos de sus clientes, para toda solución siempre existirán "los que no pueden". Los que no pueden bajarlo, los que no pueden instalarlo, los que no pueden leerlo.

Revise las estadísticas de su sitio y verificará que "los que no pueden", representan un porcentaje muy importante de sus visitantes.

Revise las estadísticas de su sitio y verificará que los que no pueden, sumados representan un porcentaje muy importante de sus visitantes. Si suma los que ven la pantalla en 256 colores, a los que utilizan versiones 3 o anteriores de navegador, a los que leen el correo en modo texto, a los que no tienen instalada una plug-in determinada, a los que no soportan JavaScript, a los que no soportan VisualBasicSript y sigue una larga lista de etcéteras, comprobará que los que no pueden son en general muchos más de los que usted imagina.

Las actividades de diseño gráfico requieren cantidades enormes de memoria y capacidad de procesamiento. En las máquinas de los diseñadores todo se ve perfecto y rápido. Es necesario desde el principio ver los diseños en el entorno en que serán utilizados, si fuera posible en el propio sitio. Y es necesario verlo también en las condiciones de los que no pueden. Vea el sitio con Netscape 3 y con Explorer 3 y descubrirá cuadros negros dónde debrían ir esos botones con un fantástico roll-over, interminables mensajes de error, textos corridos. Véalo desde un equipo Linux y descubrirá otros cambios. Cambie la resolución a 640 x 480 con 256 colores y terminará de horrorizarse

Sólo la utilización de un Font no disponible en el equipo del usuario, puede cambiar completamente el look and feel de la página y con ello estropear totalmente la Usabilidad y la capacidad del usuario de interactuar con el sitio. Un diseñador profesional debería estar más preocupado por estos temas que por el iconito de la búsqueda.

## El mito de llamar la atención

Otra característica vinculada a la Usabilidad tiene que ver con la atención de los usuarios. Los publicistas gastan su vida tratando de atraer la atención de los posibles consumidores. Los grandes publicistas lo consiguen y lo hacen de una forma tan eficiente y agradable que recordamos por muchos años con mucho cariño sus piezas publicitarias.

En su sitio Web usted no tiene necesidad de llamar la atención de sus visitantes, tiene la necesidad de mantenerla. Quien está en su sitio es porque conscientemente puso la URL de su sitio o porque clickeó en un link en un buscador u otro sitio que lo condujo a su sitio. Ese usuario ya decidió prestarle atención a su sitio. No hay que llamar su atención, hay que darle lo que vino a buscar de la manera más rápida y eficiente y así hacer que valga para él la pena mantener la atención en nuestro sitio.

En Internet no hay vidriera, no hay zapping. No hay ningún menú de todos los sitios posibles donde llamar la atención, ni siquiera los buscadores. Porque inclusive el zapping recorre una serie limitada de opciones y en ese caso es válida la idea de llamar la atención. Internet es muy distinto. Y como lo más parecido a un comercial televisivo en Internet es una presentación animada, los malos publicistas insisten con incluir presentaciones animadas que "capturen la atención del usuario". Lo único que consiguen es alejar al usuario de su objetivo, y bajar fuertemente el grado de satisfacción que tendrá con el sitio.

## Las animaciones en pantalla

Otro problema vinculado a la atención son las animaciones en la pantalla. El hombre desciende del mono, y desde aquellas instancias conserva en su instinto la reacción a lo que se mueve en un entorno estático. Eso le ayudaba a sobrevivir: si se movía una hoja en el arbusto, era que había alguien o algo atrás tratando de comérselo Nada más básico e instintivo.

En su sitio Web usted no tiene necesidad de llamar la atención de sus visitantes, tiene la necesidad de mantenerla.

Es imposible leer atentamente si hay elementos de la pantalla en constante movimiento. Eso responde a la naturaleza del ser humano. Las animaciones deben ser utilizadas con prudencia, y cuando se utilizan, deben comenzar y terminar, para luego quedar estáticas.

## Aprovechar las reglas no escritas

Existen algunas convenciones no escritas sobre navegabilidad. Una de ellas es por ejemplo que en la esquina superior izquierda va el logotipo de la empresa y que está linkeado a la home page. Otra de ellas es que abajo de la página va un menú principal en modo texto y en letra pequeña los créditos, Copyrights, sentencias de privacidad, etc.

Respetar estas reglas no escritas y utilizarlas a nuestro favor, agrega mucho a la Usabilidad del sitio.

## Los buscadores

Una buena característica de Usabilidad es incluir un buscador en su sitio. Los hay gratuitos, tanto como servicio así como en la forma de programas o scripts para instalar. Los hay baratos y los hay caros. En general, no hay muchos argumentos para no tener buscador.

Pero no alcanza con poner un buscador. El buscador tiene que contribuir y no restar. Es tan común encontrar buscadores absolutamente impredecibles e incomprensibles en los sitios que asusta.

No busque en toda la Web. Existen sitios especializados y archiconocidos para eso, y sus usuarios seguramente tienen preferencias bien marcadas al respecto. Si quisieran buscar en la Web, seguramente no hubieran entrado en su sitio, hubieran ido directamente al buscador de su preferencia.

No busque en toda la Web. Existen sitios especializados y archiconocidos para eso, y sus usuarios seguramente tienen preferencias bien marcadas al respecto.

La lista de resultados debe ser autoexplicativa. El link no dice nada, es bueno que tenga el título de la página, una descripción corta del contenido, y si fuera posible la línea que contiene la palabra buscada.

Explique el orden de los resultados. Indique claramente como están ordenados los resultados y en



caso de que se use un sistema de score combinado para el orden, deje que el usuario acceda a la información sobre como se calcula ese score.

Evite las opciones de lógica booleana y los operadores lógicos. Es bueno brindar opciones, pero lamentablemente, solo los programadores y los filósofos conocen en profundidad la lógica booleana. Unos por necesidad y otros porque la inventó el famoso Aristóteles. Los simples mortales son incapaces de utilizar los operadores and, not, nor, xor, etc. Utilice opciones de menú simples y autoexplicativas, obvias.

Por ejemplo en vez de and se puede utilizar un check box que diga "que contenga todas las palabras" y en vez de or uno que diga "que contenga al menos una de las palabras". Seguramente no es tan flexible como brindar la posibilidad de incluir operadores en el texto de búsqueda, pero tal vez eso al 99% de sus visitantes le tenga sin cuidado.

Permita refinar la búsqueda. Si fuera posible, permita volver a buscar sobre el resultado de una búsqueda. Es el camino más natural para llegar a un resultado. Por supuesto que (primera búsqueda) .and. (segunda búsqueda) provee el mismo resultado, pero la opción de refinar la búsqueda es más intuitiva y obvia.

## Yahoo! un diseño contra sensus

Yahoo! como empresa de Internet tiene muchas particularidades. Una de ellas es que hace 3 años que es una empresa rentable. Otra de ellas es que se trata según varias estadísticas, del sitio de mayor tráfico del mundo. Y se podrían listar muchas otras. Resulta muy interesante analizar el diseño de Yahoo! Si usted junta 100 libros acerca de diseño para Internet, probablemente ninguno de ellos contenga recomendaciones que conduzcan a un diseño como el de Yahoo!. Pero la tozuda realidad muestra que el diseño de Yahoo! es excelente.

Yahoo! no llama la atención. La pagina principal es siempre idéntica o casi idéntica. Apenas cambian las noticias que se ven a la derecha de la página. Y desde el comienzo que es casi idéntica.

Yahoo! no tiene animaciones, ni gráficos vistosos, salvo en los avisos publicitarios, y los espacios en Yahoo! se cuentan entre los más caros de la Web. Prácticamente todo está hecho en base a texto y tablas, algunas ocultas y algunas visibles. Los gráficos son de no más de 64 colores en su mayoría. Los íconos son pequeños y de 2 o 4 colores. A veces parece que fueran tacaños en el diseño gráfico.

Yahoo! es un sitio extremadamente usable, o dicho de otra forma, con un diseño obsesionado por dar Usabilidad a sus navegantes.

Yahoo! se ve igual en todos los navegadores. Salvo las pequeñas diferencias de despliegue de un navegador a otro, Yahoo! se ve exactamente igual en cualquier máquina, independientemente del sistema operativo, del navegador, de las plug-ins instaladas y de la versión de Java. Los diseñadores de Yahoo! sencillamente eliminaron todas las características que atentaran contra esta característica, sin piedad. Ninguna virtud o posibilidad los tentó a generar un sitio que interactuará con los usuarios de una forma distinta que la prevista en el diseño original.

Yahoo! es increíblemente rápido. Existen buscadores más rápidos, servicios de noticias más rápidos y en general sitios más rápidos que Yahoo!, pero con 20kbps de ancho de banda Yahoo! descarga cualquiera de sus páginas, en cualquier parte de su sitio en 8 segundos. Con una conexión de 56kbps, el tiempo baja a 5 segundos.

En resumen, Yahoo! es un sitio extremadamente usable, o dicho de otra forma, con un diseño obsesionado por dar Usabilidad a sus navegantes. Sin duda, esa obsesión tiene responsabilidad fundamental a la hora de encontrar las causas del éxito.

Artículos relacionados

[Primera parte de este artículo](#)

[Los objetivos del diseño](#)

[¿Qué es diseñar?](#)

