

Interfaces de usuario

Tabla de contenido

Introducción	2
Usabilidad	2
Definición.....	2
Características.....	2
UI (User Interface).....	2
UX (User Experience).....	2
Principios de diseño universales.....	3
¿Cómo diseñamos una aplicación?	3
Proceso de diseño	3
Plan.....	3
Diseño.....	4
¿Cómo se verá la aplicación?	4
Workflow	4
Wireframes de baja	4
Wireframes	4
Pruebas de usuario.....	5
Desarrollo del diseño.....	5
Diseño visual	5
Test de usuario.....	5
Especificaciones de los Mockups	5
Es un proceso iterativo.....	5
Pruebas de usuario	6
Herramientas	6
Nosotros también podemos probar.....	6
Casos conocidos (y no tanto).....	6
Tips al crear interfaces de usuario	7

Introducción

Vamos a introducir unos conceptos básicos:

Usabilidad

Definición

La **usabilidad** es la **medida de la calidad** de la **experiencia** que tiene un **usuario** cuando **interactúa** con un **producto o un sistema**. También podríamos decir que usabilidad es la eficacia, eficiencia y satisfacción con la que un producto permite alcanzar objetivos específicos a usuarios específicos en un contexto de uso específico.

[V / F] Usabilidad es la eficacia, eficiencia y satisfacción con la que un producto permite alcanzar objetivos específicos a usuarios específicos en un contexto de uso también específico.

Características

- **Facilidad de aprendizaje:** ¿Cuánto tarda un usuario nuevo en aprender a usar bien una interfaz y realizar operaciones básicas?
- **Facilidad y eficiencia de uso:** ¿Con qué rapidez se pueden desarrollar las tareas una vez aprendido a usar el sistema?
- **Facilita de recordar cómo funciona:** ¿Qué tan simple es recordar las características y formas de uso de un sistema para volver a utilizarlo en un futuro?
- **Frecuencia y gravedad de errores:** ¿Se ayuda al usuario cuando cometen errores al usar el sistema?
- **Satisfacción subjetiva:** ¿Queda satisfecho el usuario al emplear el sistema gracias a la facilidad y simplicidad de uso de sus pantallas?

¿Cuál o cuáles de las siguientes opciones con características de la usabilidad?

- ☒ **Cuán fácil es recordar la utilización y funcionamiento de un sistema**
- ☒ **Con que rapidez el usuario desarrolla las tareas dentro del sistema**
- ☒ **Cuánto tarda un nuevo usuario en aprender a utilizar correctamente la interfaz para realizar operaciones básicas**
- ☐ **Facilidad para detectar errores del usuario**

UI (User Interface)

La **interfaz de usuario** es el medio con el que el **usuario** puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el **usuario** y el equipo. Aquí podemos hacer referencia desde una radio con 2 botones hasta la interfaz de una web de compras.

[V / F] La interfaz de usuario abarca todos los puntos de contacto entre *los usuarios*.¹

UX (User Experience)

La **experiencia de usuario** es el conjunto de factores y elementos relativos a la **interacción del usuario**, con un entorno o dispositivo concretos, cuyo resultado es la generación de una **percepción positiva o negativa** de dicho servicio, producto o dispositivo. Esto es, básicamente, luego de interactuar con el producto, ¿me queda una sensación positiva o negativa de la experiencia?

[V / F] La experiencia de usuario se refiere a la sensación tanto positiva como negativa del usuario luego de interactuar con el producto o servicio.

¹ La interfaz de usuario abarca todos los puntos de contacto entre *el usuario y el equipo*.

Principios de diseño universales

Hay algunos principios de diseño universales:

- ◆ Que sea **usable por todos**: Es decir, que el diseño sea útil para personas con distintas capacidades.
- ◆ Que tenga **flexibilidad en el uso**: Es decir, que el diseño se acomoda a distintas preferencias y capacidades. Por ejemplo, cuando van a un cajero automático donde uno puede utilizar el audio como un recurso auxiliar.
- ◆ Que sea **simple e intuitivo**: El diseño es fácil de entender sin importar la experiencia, el conocimiento, habilidades comunicacionales o capacidad de concentración del usuario. Un ejemplo que suelo utilizar en este punto en particular es el caso de itunes connect. Cualquier persona que tenga un ipod, por ejemplo, va a poder corroborar lo complicado que es agregar música al dispositivo, mientras que por ser un simple reproductor de MP3, uno esperaría poder hacer un simple drag & drop de archivos.
- ◆ La **información** debe ser **efectiva**: Es decir, nuestro diseño comunica la información efectivamente.
- ◆ **Tolerancia al error**: Tenemos que minimizar los peligros o efectos adversos de accidentes sin intención durante el uso. Por ejemplo, cuando realizamos una compra, uno esperaría que al estar por completar una acción que genere una salida de la cual me pueda arrepentir, tenga confirmación: ¿desea borrar tal producto? ¿desea comprar tal producto?
- ◆ Que el **uso requiera el menor esfuerzo**: El diseño puede ser utilizado de manera eficiente. Ejemplo: Que las funcionalidades sean de simple acceso. Si quiero llegar a realizar una compra y el flujo del pago me incurre en 15 minutos ingresando datos, eso no es óptimo.

Algunos principios de diseño son:

- ☒ **Minimizar los peligros o efectos adversos de acciones sin intención**
- ☒ **Comunicar de forma efectiva**
- ☒ **Flexibilidad en el uso**
- ☒ **Requerir un esfuerzo mínimo**

¿Cómo diseñamos una aplicación?

Teniendo en cuenta todo lo que vimos hasta el momento, podemos concentrarnos ahora en cómo diseñamos una aplicación. Para diseñar una aplicación tenemos que tener en cuenta muchos aspectos: Partiendo del público al cual va a estar dirigida la aplicación, podemos analizar la cultura, idiomas, edades, modelos mentales, interfaces, colores, gestos, etc. ¿Cuál es el proceso de sugerido para diseñar una aplicación?

Proceso de diseño

Vamos a tener cuatro etapas dinámicas:

- 1) Plan
- 2) Diseño: Workflow, wireframes
- 3) Desarrollo: diseño, mockups
- 4) Especificaciones de los Mockups

Plan

Primer paso es el plan:

- ❖ Conceptualización y definición
- ❖ Alcance del proyecto

Podemos comenzar a planificar una aplicación, analizando cuál es el contenido mínimo que vamos a incluir, pensando en la forma en la que el usuario va a poder acceder a los contenidos, qué pantallas va a haber, cómo va a

llegar a las mismas. Al principio arrancamos con nombres y una breve descripción de las secciones que luego iremos refinando.

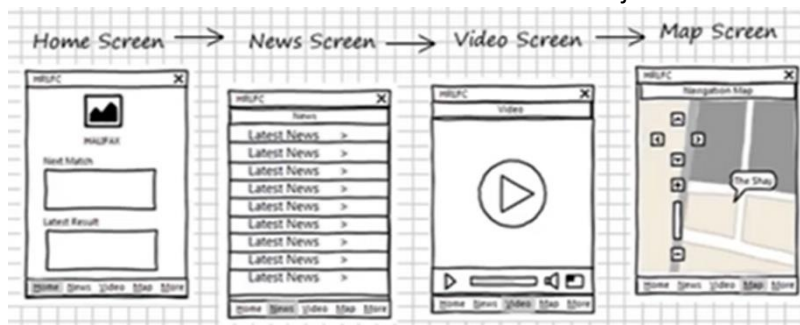
Diseño

¿Cómo se verá la aplicación?

Luego viene el diseño:

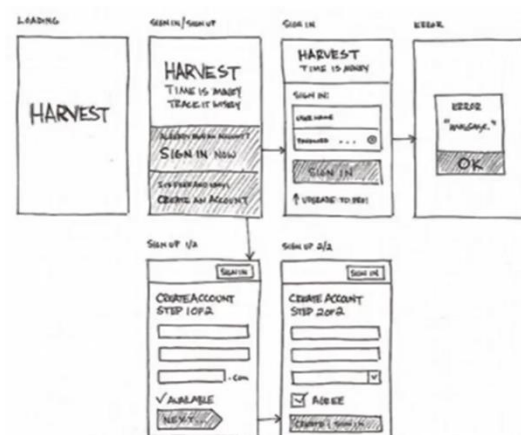
- ❖ ¿Cómo se va a ver la aplicación?
- ❖ ¿Qué información va a contener?
- ❖ ¿Qué flujo de pantallas tendrá?

Una vez que sabemos aproximadamente cómo va a estar compuesta la aplicación, podemos comenzar a armar un bosquejo de las pantallas/secciones definidas anteriormente. Cuando estamos armando un look & feel en baja definición (como se ve en la imagen) noten que no hay colores, no hay imágenes reales en este diseño. Hay que destacar que la finalidad es conocer cómo sería el supuesto contenido para poder realizarlo y mejorarlo, y luego aplicar un diseño más detallado. La intención de un bosquejo de bajo nivel, o sea, un **wireframe** de baja, es validar que el contenido mínimo esté ahí y nos permita entender qué se va a hacer en cada parte.



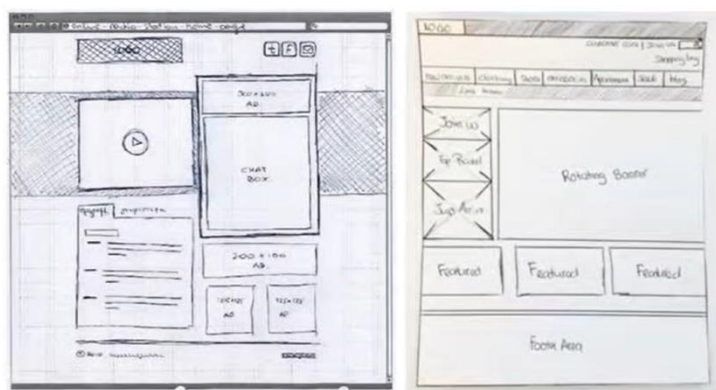
Workflow

Estos wireframes de baja también nos ayudan a ver cómo va a ser el *flujo de la aplicación*. De esta forma, podemos analizar y optimizar el diseño de forma tal que el resultado sea óptimo a la experiencia de usuario. Al ver la información abierta de esta forma, podríamos detectar que estamos mostrando la información en varios lugares distintos, lo cual nos permitiría reorganizar el contenido para evitar confusiones. Otro aspecto que podemos profundizar es la *navegabilidad*. Si jugamos con los wireframes, podemos navegar a través de los mismos, y así, podríamos detectar que nos falta incluir formas de navegar de una pantalla a otra. Por ejemplo, me olvide el botón de back en una aplicación IOS.



Wireframes de baja

Así como hay herramientas para armar estos wireframes, también podemos realizarlos a mano alzada en una hoja, en una servilleta. Todo depende de la intención del diseño y del esfuerzo que quiera dedicarle; también de la etapa del diseño. Recordar que en esta etapa estamos validando alto nivel: podríamos dedicar muchas horas a pensar aquí y que el cliente nos pida cambiarlo en gran parte.



Wireframes

Según la herramienta que utilizemos, vamos a poder armar wireframes que simulen contenido web, mobile, tablets, etc. Hay muchas herramientas de ahí fuera que nos permiten ahorrarnos el proceso de diseño, incluyendo botones predefinidos según la plataforma.

Como mencionaba, el beneficio de investigar estas herramientas del momento es que nos proveen assets actualizados, incluso hay herramientas que nos permiten definir acciones en los wireframes y exportarlos de forma tal, de facilitar las pruebas de usuario.

Pruebas de usuario

Y nos preguntamos, ¿qué son las pruebas de usuario? Son pruebas que personas ajenas al diseño realizan sobre los diseños que creamos. Las pruebas de usuario son importantes en todo momento. Pueden realizarse pruebas sobre una hoja, hasta sobre una pantalla mostrándoles una imagen, o incluso sobre wireframes con interacción. Pero nos preguntamos, ¿cómo prueba el usuario sobre una imagen? Nosotros en teoría conocemos las interacciones que estamos definiendo en las aplicaciones, por lo cual pedimos al usuario ejecutar una acción específica, y según lo que el mismo toque en la imagen, simulamos un cambio de pantalla reemplazando una imagen por otra. ¿Qué información nos brindan este tipo de pruebas? Nos permite saber a qué información le presta más atención el usuario; entender patrones de acción; observar cómo intenta ejecutar un pedido. Por ahí detectamos que el 80% de los usuarios que ejecutan estas pruebas intuitivamente intentan realizar una acción de determinada forma contraria a nuestros diseños, y esto nos permite realizar ajustes correctivos.

Desarrollo del diseño

En la tercera etapa comenzamos a desarrollar el diseño.

Diseño visual

Una vez que los wireframes están avanzados, podemos comenzar a aplicar el diseño: colores de la marca, el branding, gráficos, iconos, etc. Cabe aclarar, cuando el wireframe pasa a tener diseño, se convierte en un **mockup**. Un mockup es cómo se vería la aplicación realmente. ¡Ojo! Todavía no desarrollamos ninguna funcionalidad, sólo nos estamos enfocando en el diseño.

Test de usuario

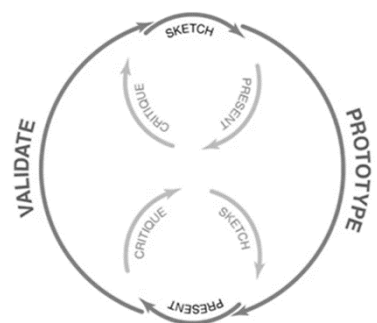
En la medida que seguimos avanzando con el diseño visual, podemos repetir los test de usuario. Estos se vuelven a realizar todas las veces que se considere necesario. Una herramienta que aquí es conocida es "InVision", donde uno puede, en una plataforma web, ir colocando las imágenes de la aplicación y definir interacciones entre las mismas, es decir, puedo marcar áreas clickeables en una imagen y según el área donde clickeé, me llevan hacia la siguiente imagen simulando una transición dentro de la aplicación.

Especificaciones de los Mockups

Una vez que tenemos el diseño prácticamente terminado, procedemos a especificar el mismo. Este output/salida le va a servir a los desarrolladores para poder replicar el diseño por medio del código. Es decir, que tendrán que aplicar determinada fuente, con determinado tamaño, determinado formato, determinado color.

Es un proceso iterativo

Este proceso es un proceso iterativo, por el cual armamos el bosquejo, prototipamos, presentamos y discutimos, validamos y nuevamente comenzamos hasta considerar que las iteraciones no son más necesarias.



Las etapas dinámicas del proceso iterativo de diseño son:

- ☒ **Planificar, flujo de trabajo, desarrollo, especificación**
- ☐ Alcance, flujo de pantallas, bosquejos, pruebas, diseño visual
- ☐ Planificar, navegabilidad, validación, desarrollo, especificación

Pruebas de usuario

Herramientas

Para realizar pruebas de usuario, hay muchísimas herramientas. Entre las tantas que existen, puedo mencionar:

- 1) UserTesting.com: Es un sitio donde uno puede subir una aplicación, solicitar a los usuarios que realicen determinadas acciones, solicitar que los filmen y también obtener feedback de los mismos sobre qué cosas mejorarían o no. Uno puede pedir lo que necesite en las pruebas.
- 2) Amazon Mechanical Turk: Donde se pueden crear HITs (Human Intelligence Task) donde se solicitan a personas, también con determinadas características, que realicen acciones que nosotros solicitemos.
- 3) Ejemplo AB Testing (Spotify): Implica que una misma aplicación muestra 2 formas distintas de una misma funcionalidad, por lo cual 2 usuarios, utilizando la misma aplicación, podrán realizar determinada acción de 2 formas diferentes. Esto se lo mide y en base a las métricas, según lo que se esté midiendo, se decide cuál de las 2 formas queda efectivamente productiva y disponible al 100% de los usuarios.
- 4) Deploys en etapas: Yo tengo una aplicación y la voy liberando en un determinado porcentaje; según el feedback y las métricas que voy obteniendo de ese pequeño porcentaje, lo sigo liberando en etapas a mayor porcentaje de gente hasta llegar al 100%.
- 5) Deploy internos con TestFairy: TestFairy es una forma de repartir aplicaciones mobile donde esta herramienta nos permite grabar vídeos de las pruebas de usuario y exportarlas directamente a YouTube. Es decir, que yo le doy a un cliente esta aplicación y puedo ver como se desempeña dentro de la misma, y obtener estos videos directamente.
- 6) Y más... Les recomiendo una lectura: Rocket Surgery es un libro muy interesante sobre pruebas de usuario.

Nosotros también podemos probar

Es común que a veces nos provean el diseño de una aplicación. Nos pueden dar un Photoshop, por ejemplo, y una actividad muy interesante es imprimir este diseño y simular usar la aplicación. De esta forma vamos a poder entender mejor el flujo, optimizarlo, detectar fallas en el mismo, funcionalidades ausentes, mejoras, etc.

[V / F] Las pruebas de usuario son realizadas por personas ajenas al diseño de interfaces.

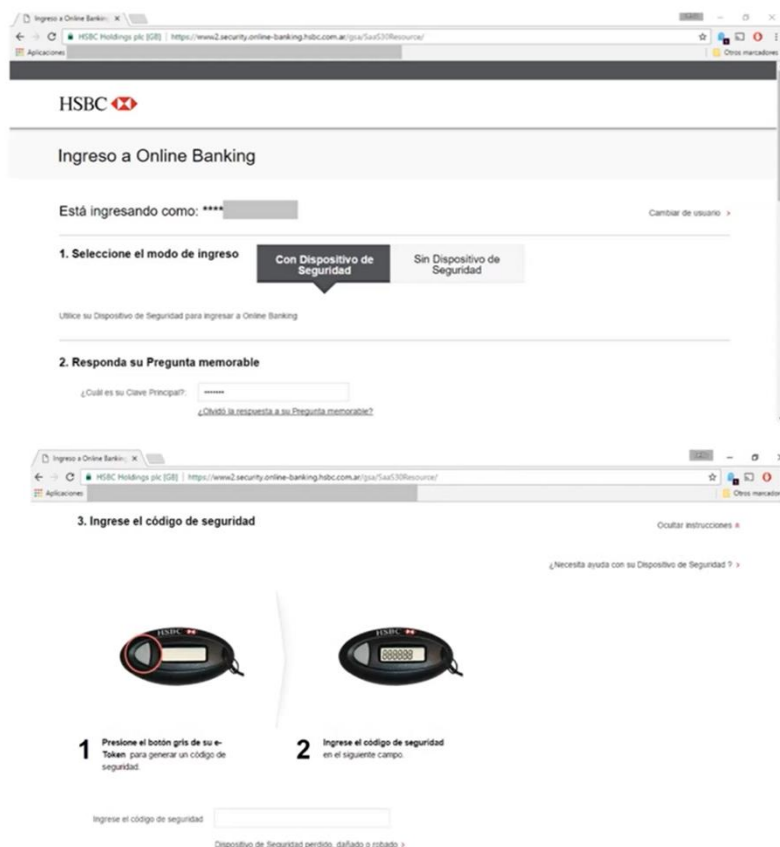
Casos conocidos (y no tanto)

Para finalizar con pruebas de usuario, quiero mencionar algunos casos conocidos, otros no tanto, donde las pruebas de usuario fallaron:

- 👉 iPhone 4: Este caso fue famoso, según cómo tomabas el celular durante una llamada, el dispositivo se quedaba sin señal. ¿Qué sucedería? Que la antena estaba ubicada de manera tal que al tomar el teléfono con la mano, la señal se veía directamente afectada, provocando que se corten las llamadas.
- 👉 Google Chromecast: Aquellos que lo tengan podrán apreciar un pequeño prolongador de cable. Uno pensaría, siendo tan pequeño, ¿cuál es la utilidad de este cable? Y la realidad es que el mismo permite conectar el dispositivo en televisores donde no hay espacio para conectar de forma directa el Chromecast. ¿Por qué? Porque el espacio detrás de la televisión es como un cuadrado sumergido y el tamaño del Chromecast no permite ser enchufado. Claramente en el diseño de la derecha, resolvieron eso con la versión 2 del mismo.



📌 En este famosos home banking, tenemos un ejemplo donde la usabilidad es óptima: Uno espera que el usuario ingrese un conjunto de datos y lejos de facilitar esta acción, aquí nos la complican. A simple vista, tenemos que agregar usuario y contestar una pregunta memorable. Yo me pregunto, ¿qué es una pregunta memorable? Si en todos los sitios se conoce como usuario y contraseña, ¿por qué no poner directamente eso? Usualmente uno ingresaría aquí usuario y password y luego apretaría enter. En este caso particular, no va a funcionar, dado que si notamos a la derecha hay una gran barra de scroll que nos oculta la siguiente información (imagen de abajo). Ahora podemos apreciar que había un paso número 3. Al scrollear hacia abajo de todo notamos que tenemos que ingresar el código de seguridad. Nuevamente nos queda fuera de vista los 3 campos básicos que tenemos que ingresar, que son usuario, contraseña y código de seguridad. Luego de ingresar este valor, ahora sí podemos apretar enter y va a funcionar. Pero toda la información está distribuida en la pantalla de forma tal que no está fácilmente accesible para el usuario.



Tips al crear interfaces de usuario

Ahora pasamos a mencionar algunos tips/consejos para crear buenas interfaces de usuario:

👍 Es muy recomendable a la hora de utilizar algún sistema particular, validar la existencia de **guidelines** (guías de desarrollo) para el sistema en mención. Cada sistema suele tener sugerencias sobre cómo diseñar aplicaciones para dicho sistema.

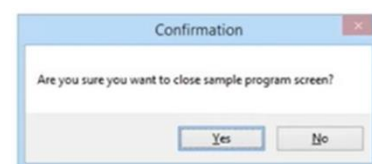
👍 Respeto a la **navegabilidad**, es recomendable indicar la ubicación parcial actual del usuario dentro de la aplicación de forma tal que el usuario siempre pueda volver en sus pasos, sepa dónde está ubicado y no necesite comenzar de cero.



👍 Siempre es de gran ayuda **indicar el progreso del usuario en un proceso**. De esta forma le damos la opción de elegir permanecer en el proceso o realizarlo en otro momento.



👍 **Confirmar acciones:** Siempre ofrezcamos la opción de confirmar acciones, más aún si pueden tener un efecto negativo no deseado. Respetar la ubicación de las respuestas positivas y negativas. Pueden apreciar en las imágenes que ambas imágenes tienen invertidas los botones positivos y negativos. El usuario tiene más probabilidades de presionar el botón de la derecha, es por eso que se recomienda que la acción menos dañina se ubique a la derecha. Por ejemplo: “¿Quiere borrar este dato?” El “No” debiera estar a la derecha, de forma tal que, si efectivamente quería borrar tal información y se equivoca, el usuario tenga que volver conscientemente a presionar el “Sí”, y en caso de error, intuitivamente presionaría el “No”.



- Siempre **indicarle al usuario si el sistema está procesando información**. Con una spinning wheel, una progress bar. Un usuario que mira más de cierta cantidad de tiempo en la misma pantalla sin feedback, puede pensar que el sistema se congeló o no funciona.



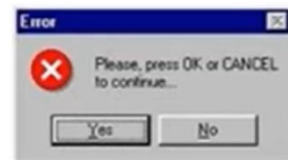
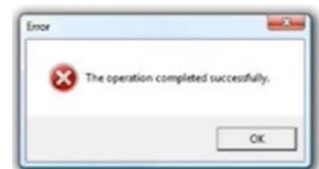
- Podemos utilizar la cantidad de recursos que sea necesario para transmitir la información. También tenemos **sistema de colores**: Generalmente se utiliza el verde para algo que está considerado OK/bien, el rojo para lo que está considerado mal, y podemos ayudar esta información con íconos como un check (✓) para lo que está OK y una cruz (X) para lo que está mal.



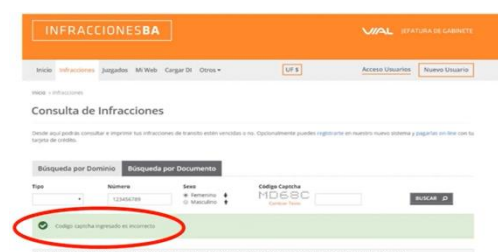
- Tener en cuenta la **ubicación por default de las acciones positivas Vs. negativas**.
- Considerar las distintas resoluciones**: ¿cómo vamos a mostrar la información? ¿de forma vertical? ¿Portrait? ¿De forma horizontal? ¿Landscape? ¿Cómo podemos optimizar el uso de la información según la orientación?
- Siempre estar al día respecto a los **beneficios que nos brinda la tecnología** de forma tal de mejorar la experiencia de usuario. Si estamos construyendo un sistema, obtener lo mejor del mismo:
 - ¿Tenemos disponible el Mouse o pantalla táctil?
 - ¿Podemos utilizar los sensores del positivo como giróscopo, acelerómetro, etc.?
- Si se puede aprender a utilizar nuestro producto, sugiriendo pasos o ayuda (**hints/tutoriales**), permitamos al usuario disponer de esa información de manera opcional.
- Respetar la terminología**: Tengamos en cuenta respetar el uso común de la misma. No alteremos el uso de términos comunes a nuestro gusto.
- Mostrar mensajes de resultado de operación** al usuario: Ser claros sobre la efectividad de la ejecución de una acción.
- Siempre recordemos la importancia de **transmitir un mensaje correctamente**. Se deben transmitir mensajes **claros y seguros**.

Por ejemplo:

- “La operación se completó exitosamente” pero lo informo con un título de error, una Cruz Roja.
- Ser consistentes entre el mensaje y la acción: “Presiona OK o cancel para continuar”, y las opciones son “yes o “no”.
- Un mensaje informativo cuyo contenido es “mensaje”... ¿OK, gracias?



- Otro ejemplo de mensaje incorrecto: “El código Captcha ingresado es incorrecto” en verde y con una tilde verde. Para mí no hay error, si estoy mirando esto en un sistema.



- Acá tenemos un ejemplo de por qué las interfaces de usuario son importantes en todos los ámbitos: 2 simples tarjetas, 2 diseños completamente distintos, y la efectividad en la comunicación de ambos es notoriamente distinta. Siempre prestemos atención a qué es particularmente lo que queremos comunicar con claridad. Acá podemos apreciar a la izquierda, la famosa tarjeta donde se anunció la mejor película durante la ceremonia de los oscars. La categoría está indicada en tamaño ínfimo, debajo de todo y lo que destaca a la vista tiene mismo tamaño, misma tipografía es actriz y película. Tener en cuenta que una persona promedio comienza a leer de arriba hacia abajo, de izquierda a derecha. Mientras que a la derecha podemos ver que un diseño sugerido, se indica claramente que se trata de la categoría “mejor actriz”, el nombre de la misma destacado por sobre la película. Con seguridad, este diseño hubiera ayudado a evitar la confusión.



- 👉 Siempre **seamos consistentes con lo que hacemos**: Si hacemos algo de la misma forma siempre, el usuario se acostumbra y espera que sigamos haciendo todo de la misma forma.
- 👉 Probar lo que diseñamos es importante.

Independientemente de que muchas de estas imágenes nos pueden resultar graciosas, recuerden que somos profesionales y que solemos trabajar en equipo. Si bajo alguna circunstancia notan que los diseños presentan algún inconveniente, siempre es bienvenido compartir ese feedback con el resto y resolver los inconvenientes antes de poner manos a la obra.

Algunos tips al momento de diseñar interfaces de usuario son:

- ☒ **Que los mensajes sean claros y seguros**
- ☐ La opción menos dañina se ubique a la *izquierda*²
- ☒ **Estar al día respecto a los beneficios que brinda la tecnología**
- ☒ **Seguir los lineamientos o guías de diseño cuando dichas plataformas o sistemas los proveen**
- ☒ **Que la ayuda respecto a la usabilidad del sistema sea opcional**

² La opción menos dañina se ubique a la *derecha*