



**UTN.BA**  
UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL BUENOS AIRES

**Centro de  
e-Learning**  
Secretaría de Cultura y Extensión Universitaria

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso

(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148

[www.sceu.frba.utn.edu.ar/e-learning](http://www.sceu.frba.utn.edu.ar/e-learning)

**Experto universitario:**

# **Diseño de experiencia de usuario e interacciones (UX y IxD)**



Módulo 1:

# Definiciones y base teórica

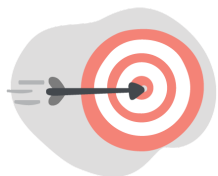
Unidad 4:

## Heurísticas



## Presentación

En esta unidad veremos a partir de tres autores diferentes, cómo se puede realizar un testeo de usabilidad a una interfaz con el uso de reglas heurísticas. Haremos una agrupación entre ambas, y se verá un ejemplo de aplicación de las mismas.



# Objetivos

**Que los participantes logren...**

- Conozcan qué son las reglas heurísticas
- Aprendan cómo medir la usabilidad
- Comprendan las diferentes reglas heurísticas.



## Bloques temáticos

1. Usabilidad y reglas Heurísticas.
2. Nielsen, Tognazzini y Norman.
3. Comparación e integración de reglas.

# Usabilidad y Reglas Heurísticas

Palabras que hoy se asocian a la experiencia de usuario en la actualidad son las de usabilidad y heurísticas. En la unidad 1, se definió a la "usabilidad" como un atributo cuali-cuantitativo por el cual se puede medir la facilidad de uso de un producto o servicio. La palabra "heurística" ( de origen griego) tiene muchas interpretaciones. Su literal traducción a "encontrar o inventar".

Herbert A. Simon propone la heurística como un sistema de técnicas utilizadas para resolver problemas. Justamente es esta acepción la que utiliza el estudio de la experiencia de usuario: una evaluación heurística será el medio para medir la usabilidad de una interfaz por medio de la inspección. Basado en el diseño centrado en el usuario, autores como Nielsen, Norman y Tognazzini proponen una serie de reglas útiles al momento de evaluar interfaces. En sí mismo, un análisis heurístico no se trata de una técnica cerrada, sino que es tarea de quién esté a cargo de la evaluación determinar diferentes parámetros o técnicas para cada sistema o producto de forma particular. Como justamente la usabilidad es un parámetro medible, una técnica útil es la utilización de la escala de Likert<sup>1</sup> dónde se puede dar un valor de cumplimiento absoluto a no cumplimiento de los puntos que se estén evaluando. La idea es plantear una escala de 0 a 5 para cada punto a analizar, y (a pesar de que la medición es subjetiva) pueden obtenerse conclusiones en función del análisis para generar mejoras. El proceso es conveniente repetirlo con diferentes analistas y así verificar de manera grupal coincidencias y puntos de vista

---

<sup>1</sup> La Escala de tipo Likert es una escala psicométrica (disciplina que se encarga de la medición en psicología) comúnmente utilizada en cuestionarios, y es la escala de uso más amplio en encuestas para la investigación, principalmente en ciencias sociales. Un elemento de tipo Likert es una declaración que se le hace a los sujetos para que éstos lo evalúen en función de su criterio subjetivo; generalmente se pide a los sujetos que manifiesten su grado de acuerdo o desacuerdo mediante unos niveles.

## ¿Por qué usar el análisis heurístico?

Ante todo es importante remarcar que este tipo de testeos no cuenta con la experiencia del usuario, sino que se trata exclusivamente de la medición de la usabilidad en la que no puede medirse cómo el usuario "se siente" al interactuar con el sistema. Una de las ventajas principales de las evaluaciones heurísticas es su bajo costo, suelen ser económicas en cuanto a recursos, ya que no necesitan de usuarios reales. Se utilizan también como un primer filtro para optimizar las interfaces antes de que estén en contacto con el usuario, y además pueden emplearse en instancias tempranas del diseño. Una gran ventaja es que pueden aplicarse desde el comienzo de prototipado, puliendo así cuestiones de diseño, estructura y errores. Es importante saber que pese a su simplicidad, es necesario tener evaluadores especializados, porque su análisis depende de la interpretación de principios sobre todo al tratarse de parámetros subjetivos.

## Proceso: Evaluación heurística:

1. Se define el objetivo, escenario y flujos de interacción. Además se define qué reglas heurísticas serán analizadas.
2. Los evaluadores analizan detalles de interacción de la interfaz (con un repaso de las mismas) y confeccionan una lista indicando problemas encontrados según reglas heurísticas con las que estén trabajando.
3. Los evaluadores rotan sus listados e indican los índices de gravedad de cada listado.

# Nielsen, Tognazzini y Norman.

Dentro del ámbito de la experiencia de usuario, podemos destacar a Jakob Nielsen, Don Norman y Bruce Tognazzini. Todos son parte de Nielsen Norman Group y son hoy las figuras de referencia en lo que al campo de estudio refiere.

## Heurísticas de Jakob Nielsen<sup>2</sup>:

Las primeras heurísticas fueron definidas por Nielsen a partir de un estudio que realizó sobre 249 problemas de usabilidad. A partir de los resultados que obtuvo de este análisis diseñó lo que denominó reglas generales para identificar posibles problemas de usabilidad:

1. Visibilidad de estado del sistema
2. Hablar el lenguaje de los usuarios
3. Control y libertad del usuario
4. Consistencia y estándares
5. Prevenir y ofrecer manejo de errores
6. Reducir la carga de la memoria del usuario
7. Flexibilidad y eficiencia de uso
8. Estética y Minimalismo
9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores
10. Ayuda y documentación

---

<sup>2</sup> Jakob Nielsen es una de las personas más respetadas en el ámbito mundial sobre usabilidad en la web. Este ingeniero de interfaces obtuvo su doctorado en diseño de interfaces de usuario y ciencias de la computación en la Universidad Técnica de Dinamarca. Su desempeño profesional le ha hecho pasar por empresas como



Bellcore, IBM y Sun Microsystems. Actualmente figura como co-fundador de Nielsen Norman Group con Donald Norman, otro experto en usabilidad.

Su trayectoria se inició en 1997 cuando escribió dos breves artículos sobre cómo preparar los textos. Los títulos de estos artículos fueron ¡Sea breve! (escribir para la web) y Cómo leen los usuarios en la web. Las ideas de los artículos de Nielsen se citan en muchos otros artículos que ofrecen pautas sobre cómo escribir para la web y mejorar su usabilidad.



Visibility of system status



Match between system and the real world



User control and freedom



Help users recognize, diagnose, and recover from errors



Help and documentation



Error prevention



Flexibility and efficiency of use



Aesthetic and minimalist design



Recognition rather than recall



Consistency and standards

Fuente: Tösabiten. Nielsens 10 Heuristics for usability:

<http://tosabiten.blogspot.hk/2011/03/nielsens-10-heuristik-for-usability.html>

## 1. Visibilidad de estado del sistema:

Informar al usuario sobre lo que está haciendo el sistema, mostrando información y dando feedback de progreso. Hacer que los objetos, las acciones (botones y links) y las opciones sean visibles.

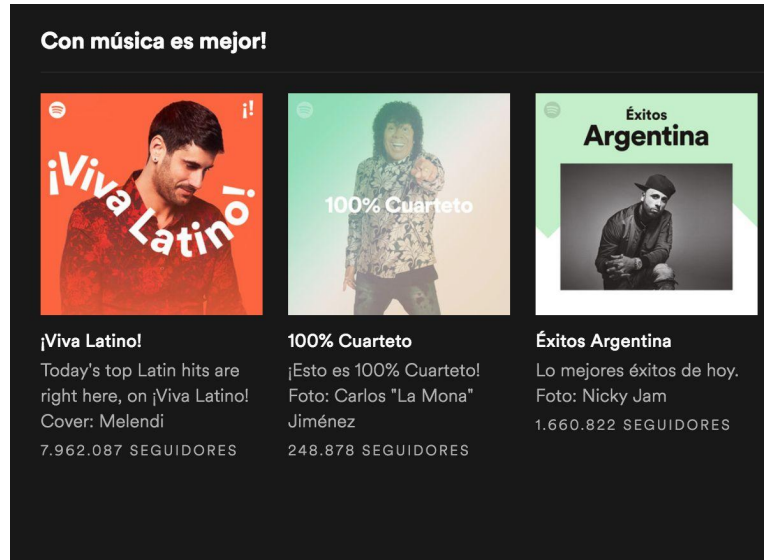


Fuente: We Transfer.

*Ejemplos: Mostrar tiempo, acción, espacio, compleción, cambios, próximos pasos.*

## 2. Hablar el lenguaje de los usuarios

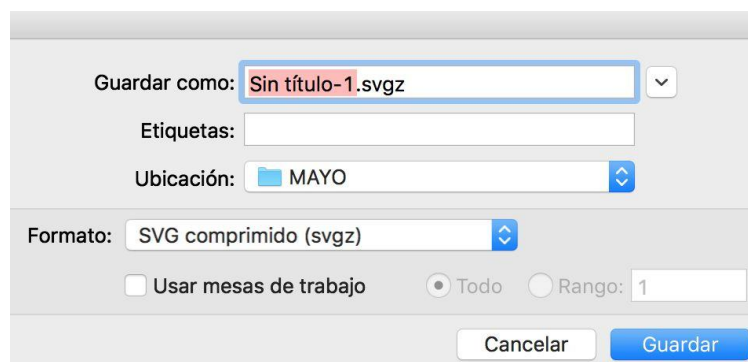
Usar un lenguaje que el usuario pueda entender. Usar la jerga propia de los usuarios. Redactar desde el punto de vista del usuario.



Fuente: Spotify

### 3. Control y libertad

Los usuarios deben sentir que tienen el control en todo momento, no deben sentirse atrapados por la aplicación. Deben iniciar las acciones en lugar de responderlas.

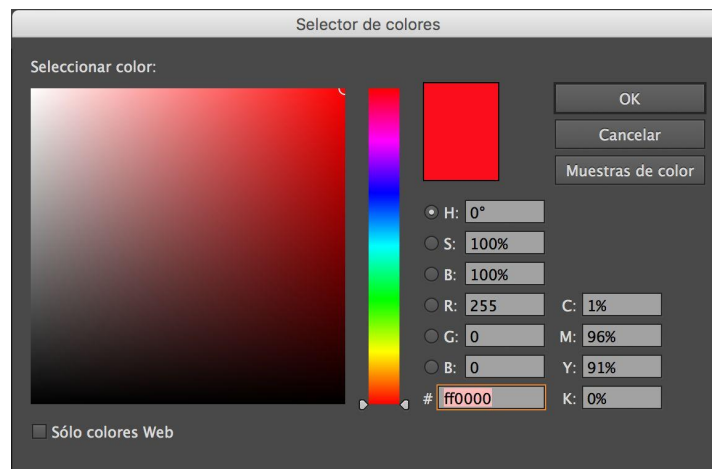


Fuente: Adobe Illustrator

*Ejemplos: Cancelar en todos los diálogos. Deshacer y escape, cuando sea posible. Salir del programa en cualquier momento. Permitir la interrupción de tareas en operaciones mayores a 10 seg. Ir hacia atrás en el navegador. Eliminar el carrito en aplicaciones e-commerce.*

## 4. Consistencia:

Los usuarios deben poder aprender una secuencia de acciones en una parte del sistema y aplicarla en otras para obtener resultados similares.

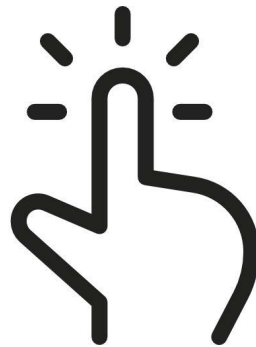


Fuente: Adobe Illustrator

*Ejemplos: No usar distintos nombres para las mismas acciones. Desde lo visual los elementos deben tener una correspondencia entre aspecto y comportamiento. Usar el mismo estilo (color y forma) o palabras para las mismas cosas. Respetar convenciones y estándares de la plataforma. (sistema operativo, web tradicional, web 2.0, herramientas de uso frecuente).*

## 5. Prevenir y ofrecer manejo de errores

Diseñar el sistema de manera que los usuarios no cometan errores. Si se comete un error el sistema debería ayudar a detectar el error y ofrecer una manera simple de resolverlo.



Fuente: [https://www.flaticon.es/icono-gratis/click-con-un-dedo\\_84590](https://www.flaticon.es/icono-gratis/click-con-un-dedo_84590)

Permitir las acciones correctas. Inhabilitar las acciones no válidas. Preferir la selección antes que el tipeo libre. Completar en forma automática. Impedir que el usuario continúe en un camino erróneo. Realizar validaciones. Advertir sobre situaciones que impliquen riesgo o inusuales. Wizards

## 6. Reducir la carga de la memoria del usuario:

La limitación del proceso de información de memoria a corto plazo requiere de una muestra simple de información.



Fuente: Wordpress

*Sugerencias: No obligar a un usuario a recordar cosas de una acción o de la siguiente. Los usuarios no están para hacer tareas repetitivas, para eso están las computadoras.*

*Ejemplos: fechas DD/MM/YY / Ofrecer valores por defecto. Texto sugerido. Tooltips en íconos. Funcionalidades para recordar valores.*

## 7. Flexibilidad y eficiencia de uso:

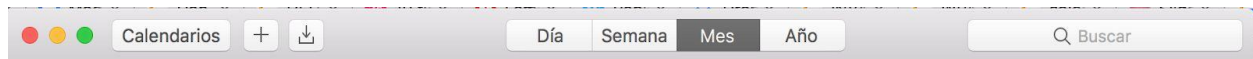
Archivo	Edición	Objeto	Texto	Sele
Nuevo...				⌘N
Nuevo de plantilla...				⇧⌘N
Abrir...				⌘O
Abrir archivos recientes				▶
Explorar en Bridge...				⇧⌘O
Cerrar				⌘W
Guardar				⌘S
Guardar como...				⇧⌘S
Guardar una copia...				⇧⌘S
Guardar como plantilla...				⇧⌘S
Guardar para Web...				⇧⌘S
Guardar sectores seleccionados...				⇧⌘S
Volver				⇧⌘Z
Colocar...				
Guardar para Microsoft Office...				
Exportar...				
Compartir en Behance...				

Al aumentar la frecuencia de uso, los usuarios desean aumentar la frecuencia de la interacción. Ofrecer alternativas de navegación teniendo en cuenta los distintos tipos de usuarios.

Ejemplos: Atajos del teclado, mnemónicos, teclas estándar, rueda del mouse. Comandos avanzados. Iconos estándar. Menús contextuales. Copiar, cortar, pegar. Navegación histórica. Proveer personalización para ver las cosas según la preferencia del usuario.

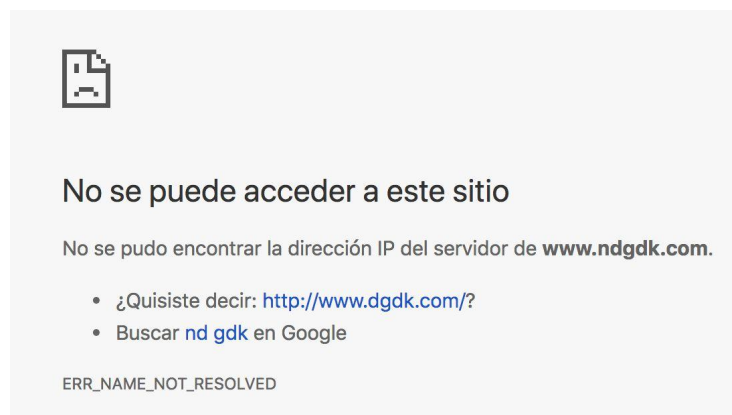
## 8. Estética y Minimalismo:

El diseño debe ser simple y práctico. Evitar información de más que pueda provocar distracciones y disminuir la velocidad del usuario. Decir lo mismo con menos palabras.



## 9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores:

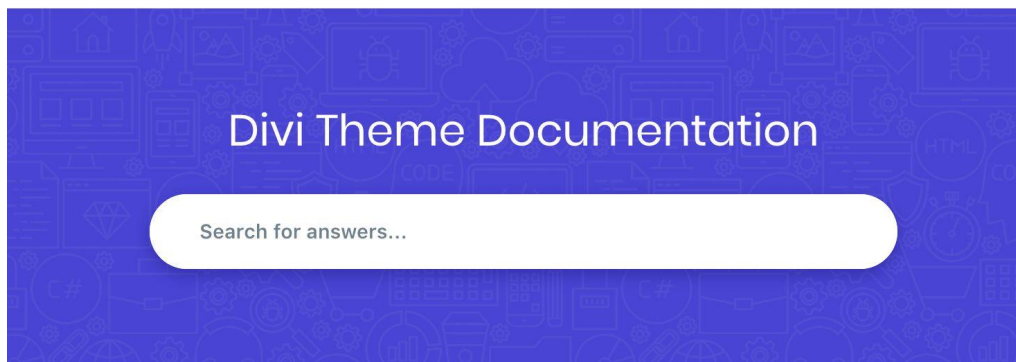
Cada mensaje de error debe decir qué ocurrió (en el lenguaje del usuario) y qué puede hacer para resolverlo.



Fuente: Google

## 10. Ayuda y documentación:

Lo ideal es que el usuario no requiera de documentación o ayuda, pero es importante aún así proveerla. Ésta debe ser fácil de encontrar, no debe ser extensa y debe ser simple en cuanto a su comprensión.



Fuente: Divi Theme

## Heurísticas de Bruce Tognazzini<sup>3</sup>

1. Anticipación.
2. Autonomía
3. Daltonismo
4. Consistencia

---

<sup>3</sup> Tognazzini es consultor y socio en Nielsen Norman Group, y es especialista en la interacción de los sistemas con el humano. Trabajó para Apple por 14 años, con Sun Microsystems durante 4 años y luego en WebMD. Escritor de dos libros "Tog on Interfaceland" y "Tog on Software Design", y en la revista digital Akstog, bajo su línea de "Soluciones de diseño de interacciones para el mundo real".



5. Opciones predeterminadas
6. Eficiencia del usuario
7. Interfaces explorables
8. Ley de Fitts
9. Objetos de interfaz humana
10. Reducción de demoras
11. Aprendizaje (learnability)
12. Uso de metáforas
13. Protección del trabajo del usuario
14. Legibilidad
15. Registro del estado
16. Navegación visible



Fuente: <https://medium.muz.li/10-tips-on-how-to-conduct-a-perfect-heuristic-evaluation-ae5f8f4b3257>

## **1. Anticipación:**

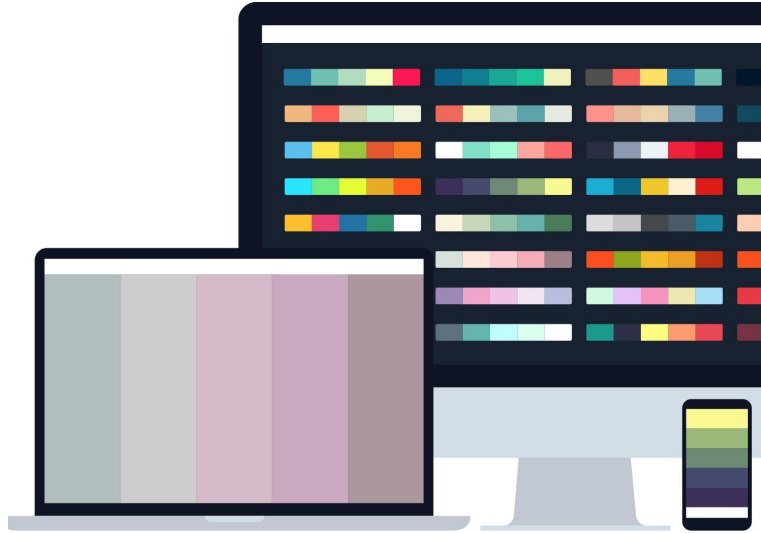
El sistema deberá intentar anticiparse a los deseos y a las necesidades que el usuario pueda tener. No hay que esperar que el usuario busque o agrupe información, sino que ante cada acción del mismo se debe presentar la información y herramientas necesarias para el proceso.

## **2. Autonomía:**

Los usuarios suelen aprender de forma rápida y cuando tienen control del sistema es cuando ganan confianza. El sistema no debe ser muy restrictivo, pero eso no implica que no tendrá reglas y límites, dando visibilidad de la información de forma fácil y actualizada.

## **3. Daltonismo:**

El recurso de color para transmitir información puede ser complejo para un 10% de hombres adultos que sufren daltonismo. Es por eso que es importante dar elementos complementarios para distinguir esta información.



Fuente: <https://coolers.co/>

#### **4. Consistencia:**

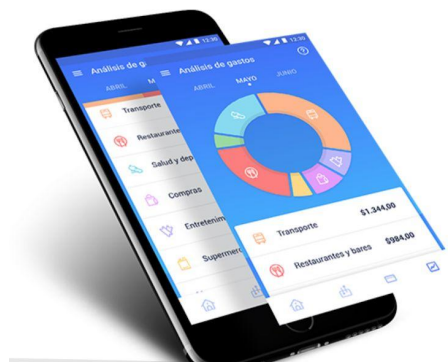
Es importante mantener la consistencia del sistema en todos los aspectos y dimensiones de la interacción y las interfaces. Esto implica desde estructuras invisibles, estructuras visibles (íconos, tamaños de las cajas, scrolls, ubicaciones, etc.), diseño visual, etc. Es también importante ser consistente con los objetos que tienen el mismo tipo de comportamientos, ya que ésto debe estar alineado con las expectativas del usuario.

#### **5. Opciones predeterminadas:**

Las opciones predeterminadas ( default , o por defecto) deben ser acordes al uso dado al sistema por la mayoría de los usuarios, ésto quiere decir que deben estar adecuadas al general, pero que a su vez, deben poder ser descartados con facilidad y rapidez, para que el usuario introduzca los propios.

## 6. Eficiencia del usuario:

El diseño debe estar centrado en la productividad del usuario y no del sistema. El sistema debe presentarle un modo de operación que lo mantenga ocupado, sin demoras ni esperas innecesarias. La redacción de los textos y mensajes debe ser clara y directa, para ayudar a que el usuario avance y resuelva situaciones rápidamente. Las opciones de los menús, las etiquetas de los botones y otros controles deben comenzar con la palabra más importante, ya que los usuarios no suelen leer sino escanear rápidamente el texto.



### Controlá

Te notificamos en el acto cada vez que usás tu tarjeta.

Organizamos tus gastos, para que entiendas cómo gastás por mes, por categorías, por colores, por lo que quieras.

Ver más

Fuente: <https://www.uala.com.ar/>

## 7. Interfaces explorables:

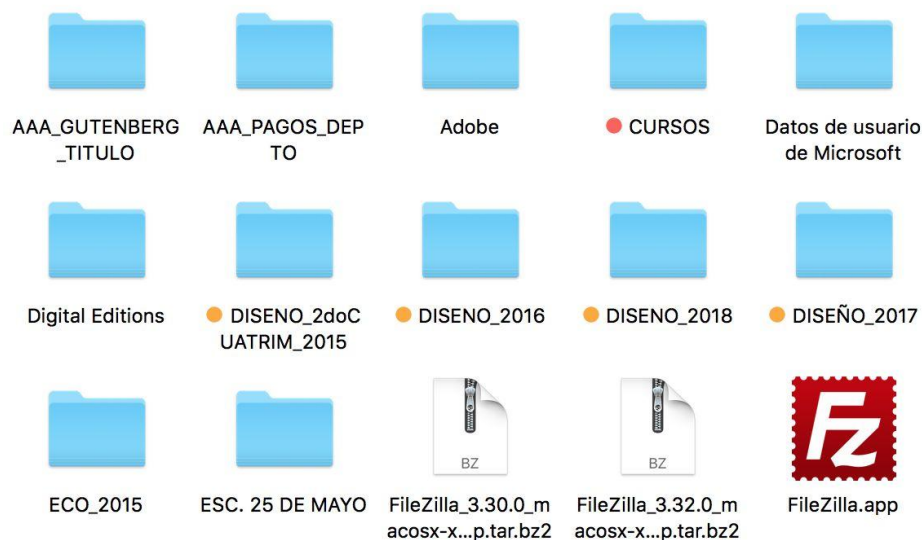
Se deben proporcionar caminos claros para que el usuario pueda completar las tareas de forma rápida. Al mismo tiempo el sistema debe dar apertura para que el usuario pueda explorar, como así también deben existir rutas de salida y la posibilidad de revertir las acciones.

## 8. Ley de Fitts:

“El tiempo para alcanzar un objetivo es una función de la distancia y el tamaño del objeto a alcanzar”. Esto significa que se deben utilizar áreas-objetivo grandes para funciones importantes y que las opciones relacionadas deben estar cercanas entre sí.

## 9. Objetos de interfaz humana:

Los elementos de una interfaz deben corresponderse con el mundo real de las personas. Estos objetos (como carpetas, documentos, la Papelera de Reciclaje, etc.) deben mantener una interacción y un comportamiento estandarizados, al tiempo que deben ser comprensibles.



Fuente: <https://www.apple.com/la/macbook/>

## **10. Reducción de demoras:**

Siempre que sea posible, se deben eliminar las esperas en la respuesta del sistema, incluso si esto implica eliminar características o funcionalidades que causen demora. Cuando eso no sea posible, se debe informar al usuario de la demora, su duración estimada, la evolución de la espera y avisarle claramente cuando haya finalizado.

## **11. Aprendizaje (“learnability”):**

Idealmente un sistema debería poder ser utilizado sin necesidad de aprendizaje pero, en la práctica, no siempre es posible. Es fundamental en esos casos proveer herramientas para que los usuarios principiantes superen la curva de aprendizaje, al mismo tiempo que no se compromete la facilidad de uso para los usuarios avanzados.

## **12. Uso de metáforas:**

El uso de metáforas para la interfaz, debe ser pensado y estar orientado a que los usuarios puedan comprender rápidamente hasta los detalles más finos del modelo conceptual en el que se basa. Las metáforas elegidas deben resultar familiares para los usuarios.

### 13. Protección del trabajo del usuario:

Se debe garantizar que el usuario nunca pierda su trabajo en el sistema como resultado de un error de su parte, de la pérdida de conectividad o de otro tipo de problemas.

### 14. Legibilidad:

Se debe utilizar texto de alto contraste (la mejor opción suele ser negro sobre blanco), en un tamaño de letra que sea legible en cualquier tipo de monitor, y dando mayor importancia visual a la información y datos que sea necesario destacar en lugar de los nombres de los campos e instrucciones. Hay que prestar especial atención a las necesidades de personas mayores y con dificultades visuales.



[Inicia sesión](#) [Registrarse](#)

**Para autónomos  
y trabajadores  
independientes**

[Obtén Dropbox](#)

**Para equipos y  
empresas de  
todos los  
tamaños**

[Obtén Dropbox Business](#)

Fuente: <https://www.dropbox.com/es/>

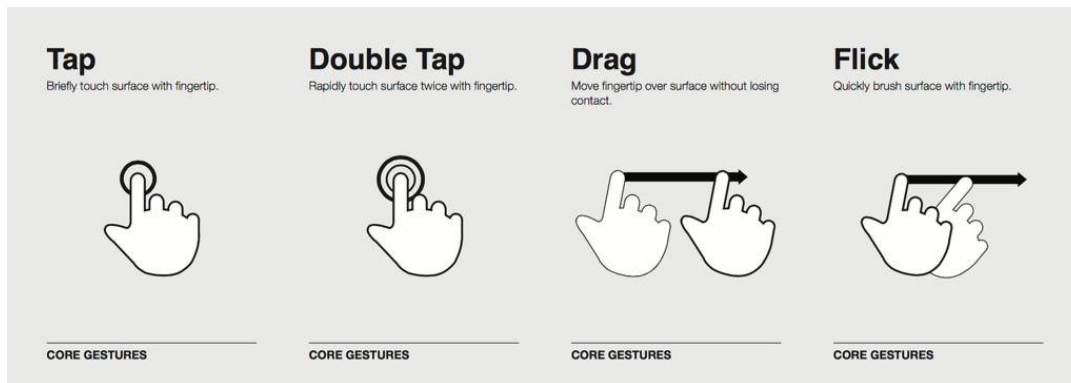
### 15. Registro del estado:

El sistema debe guardar toda la información que permita brindar una mejor experiencia al usuario (si es la primera vez que usa el sistema, dónde está, hacia

donde está yendo, dónde ha estado durante la sesión, dónde estaba la última vez que dejó el sistema, etc.). Los usuarios deberían ser capaces de cerrar una sesión, volver a conectarse desde cualquier otro sitio y seguir con su trabajo en donde lo dejaron.

## 16. Navegación visible:

La mayoría de los usuarios no pueden mantener mapas mentales complejos y se perderán o frustrarán si el sistema no les brinda información clara acerca de dónde están y a dónde pueden ir en el sistema. En lo posible se debe presentar al usuario la ilusión de que está siempre en el mismo lugar, para darle dominio y autonomía.



Fuente: Guía visual de gestos. Gráfico de Luke Wroblewski



# Heurísticas de Donald Norman: Seven Principles for Transforming Difficult Tasks into Simple Ones

1. Uso de conocimiento en el mundo y en la cabeza del usuario
2. Simplificación de la estructura de las tareas
3. Visualización de las cosas
4. Mapeo natural
5. Explotación del poder de las restricciones
6. Diseño para el error



Fuente: <http://www.winhelponline.com/blog/>

## 1. Uso de conocimiento en el mundo y en la cabeza del usuario:

Se debe diseñar un modelo que sea para el usuario. La utilización de la información que existe en el mundo y la que posee el usuario deben estar alineadas y

complementarse. Todo lo que se diseña, debe ser consistente con el modelo conceptual apropiado.

## **2. Simplificación de la estructura de las tareas:**

Las tareas deben ser simples, minimizando la cantidad de problemas que requieren. Esto se puede lograr con ayudas mentales, hacer visible lo que normalmente es invisible, cambiar la naturaleza de la tarea o por medio de la automatización.

## **3. Visualización de las cosas:**

Las acciones del sistema deben estar directamente correspondidas con las intenciones y las expectativas que el usuario tiene, como así también el estado del sistema debe ser visible. El usuario debe tener noción de lo que es posible y cómo hacerlo.

## **4. Mapeo natural:**

La asociación de controles debe ser análoga a lo que el sistema ofrece. La posición y relación espacial entre sistemas y objetos debe estar alineada. Si se utilizan controles análogos, éstos deben responder de forma similar en el sistema, como también debe darse la retroalimentación adecuada para comprender los resultados de las acciones que se generan.



Fuente: <https://ih1.redbubble.net/>

## **5. Explotación del poder de las restricciones:**

Se deben usar las limitaciones de modo que el usuario sienta que sólo hay una acción posible: la adecuada.

## **6. Diseño para el error:**

Es máxima entender que todo posible error que pueda cometerse, va a suceder. El usuario no debe combatirse, sino explicar dónde se equivocó y cómo puede recuperarse. Volver atrás debe ser una acción fácil.

## **7. Estandarización, cuando todo falle:**

Estandarizar las acciones, las diagramaciones, las interfaces o un sistema suele ser una solución cuando el sistema es muy complejo o tiene dificultades. Lo práctico de lo estándar, es que sólo debe ser aprendido una vez.

## Comparación e integración de reglas.

A modo de conclusión entonces, se puede decir que una evaluación heurística es una de las múltiples técnicas que se utilizan para poder medir y optimizar la usabilidad. Existen otras, pero es un buen comienzo y punto de partida antes de enfrentar un sistema al testeo con usuarios reales. Es fundamental recordar que cuando se diseña una interfaz el target es el usuario y no un experto en usabilidad, y serán finalmente los usuarios quienes van a interactuar con la misma.

### **Correlato:**

- Hablar el lenguaje de los usuarios (Jakob Nielsen)
- Uso de conocimiento en el mundo y en la cabeza del usuario (Donald Norman)
- Objetos de interfaz humana (Bruce Tognazzini)
- Uso de metáforas (Bruce Tognazzini)

### **Consistencia:**

- Consistencia y estándares (Jakob Nielsen)
- Consistencia (Bruce Tognazzini)

### **Libertad y control:**

- Control y libertad del usuario (Jakob Nielsen)
- Autonomía (Bruce Tognazzini)
- Interfaces explorables (Bruce Tognazzini)

### **Flexibilidad:**

- Flexibilidad y eficiencia de uso (Jakob Nielsen)
- Aprendizaje ("learnability") (Bruce Tognazzini)

### **Eficiencia del usuario:**

- Eficiencia del usuario (Bruce Tognazzini)
- Reducción de demoras (Bruce Tognazzini)
- Registro del estado (Bruce Tognazzini)

### **Anticipación:**

- Anticipación (Bruce Tognazzini)
- Ayuda y documentación (Jakob Nielsen)

### **Visibilidad:**

- Visibilidad del estado del sistema (Jakob Nielsen)
- Registro del estado (Bruce Tognazzini)
- Navegación visible (Bruce Tognazzini)
- Visualización de las cosas (Donald Norman)

### **Diseño minimalista:**

- Estética y diseño minimalista (Jakob Nielsen)

### **Simplificación:**

- Simplificación de la estructura de las tareas (Donald Norman)
- Reducir la carga de la memoria del usuario (Jakob Nielsen)
- Opciones predeterminadas (Bruce Tognazzini)
- Explotación del poder de las restricciones (Donald Norman)
- Mapeo natural (Donald Norman)
- Ley de Fitts (Bruce Tognazzini)
- Legibilidad (Bruce Tognazzini)

### **Diseño para el error:**

- Diseño para el error (Donald Norman)
- Prevención de errores (Jakob Nielsen)
- Reconocimiento, diagnóstico y recuperación de errores (Jakob Nielsen)
- Protección del trabajo del usuario (Bruce Tognazzini)
- Daltonismo (Bruce Tognazzini)

### **Estandarización:**

- Estandarización, cuando todo falle (Donald Norman)

Ejemplo (utilizando escala Likert):

### **Calificación de gravedad**

#### **Problema:**

Al completar el formulario no se indica si se envió o no.

#### **Índice de gravedad:**

3

#### **Heurística no aplicada:**

"Visibilidad de estado del sistema"

#### **Descripción:**

Al ingresar en un formulario de registro de cuenta una vez que se completa la información y se clickea el botón "enviar" el sitio no hace nada, no se sabe si se envió o no.

#### **Ejemplo de ESCALA:**

- 0: No es un problema de usabilidad
- 1: Es un problema mínimo, que responde a diseño visual.
- 2: Problema de usabilidad menor
- 3: Problema de usabilidad importante
- 4: Problema de usabilidad grave



## Bibliografía utilizada y sugerida

**Nielsen, J.** (1995). 10 Usability Heuristics for User Interface Design. Recuperado de:  
<https://www.nngroup.com/articles/ten-usability-heuristics/>

**Norman, D.** (1988). Design of everyday things. New York, USA: Basic Books, Inc.

**Tognazzini, B.** (2014) . First Principles of Interaction Design. Recuperado de:  
<http://www.asktog.com/basics/firstPrinciples.html>