

# Análisis Heurístico para UX - Cómo Ejecutar una Evaluación de Usabilidad

Hay algunas maneras que la utilidad de un producto se puede probar: un método de inspección llamado análisis heurístico es uno de ellos. Esto generalmente significa ejecutar una evaluación heurística en un producto, ya sea existente o nuevo.

**Los productos bien diseñados tienen una gran facilidad de uso. La gran usabilidad es un contribuyente significativo a la calidad del producto y una experiencia de usuario sin fisuras.**

## ¿Qué es un análisis heurístico?

**Un análisis heurístico se utiliza para identificar problemas comunes de usabilidad de un producto para que los problemas puedan resolverse, y así mejorar la satisfacción del usuario y su experiencia y aumentar las posibilidades de éxito de un producto digital**

Centrándose en la usabilidad, un análisis heurístico es un método de evaluación en el que uno o más expertos comparan el diseño de un producto digital con una lista de principios de diseño predefinidos e identifican dónde el producto no sigue estos principios.

**La evaluación heurística implica que un pequeño grupo de evaluadores examine la interfaz y juzgue su cumplimiento con los principios de usabilidad reconocidos (la heurística). — Jakob Nielsen, The Nielsen Norman Group**

El proceso implica identificar las metas del usuario y llegar a una lista de tareas para lograr esos objetivos. Los evaluadores luego señalan los problemas que los usuarios pueden tener al usar el producto.

Es útil tener en cuenta que a pesar de que un solo experimentado profesional del UX es generalmente experto en identificar los problemas de usabilidad más críticos, un grupo de evaluadores es generalmente la mejor opción. Entre 5 y 8 individuos es el punto dulce: deben ser capaces de señalar más del 80% de problemas de usabilidad. Sin embargo, el uso de más de 10 evaluadores heurísticos no producirá mejores resultados.

## ¿Por qué hacerlo?

La razón principal para realizar un análisis heurístico es mejorar la usabilidad de un producto digital. Otra razón es la eficiencia (en este contexto, “eficiencia” es la velocidad con que un producto puede ser utilizado como consecuencia directa de una mejor usabilidad). “Usabilidad” se refiere a componentes de calidad tales como capacidad de aprendizaje, detección, memorización, flexibilidad, satisfacción del usuario y manejo de errores. El UX de un producto se mejora mucho cuando estos componentes se entregan con una alta calidad.

## ¿Cuándo hacerlo?

No hay reglas duras y rápidas. Un análisis heurístico se puede realizar en cualquier etapa avanzada del proceso de diseño (Obviamente, no sería productivo hacerlo demasiado pronto). Con los nuevos productos, un análisis heurístico suele realizarse más tarde en la fase de diseño, después de hacer *wireframes* y prototipos, y antes de que comience el desarrollo de la interfaz de usuario. Hacerlo demasiado tarde y hacer cambios se hará costoso. Los productos existentes que tienen mala usabilidad a menudo tendrán un análisis heurístico ejecutado antes de que comience un rediseño.

### **Paso 1: Definir el alcance:**

Para sitios muy grandes de comercio electrónico, el análisis puede tomar las áreas más importantes del sitio. Puede tener la capacidad de centrarse en flujos específicos de usuario y funcionalidades, como inicio de sesión/registro, búsqueda y navegación, páginas de detalles del producto, carrito de la compra, y checkout.

### **Paso 2: Conocer los requisitos del negocio y los usuarios**

Por ejemplo, aunque las heurísticas debían funcionar como estándares universales de usabilidad, tal vez se debe hacer especial hincapié en la accesibilidad para una audiencia más vieja -o tal vez se necesite tener en cuenta diversas audiencias multiculturales-.

### **Paso 3: Decide qué herramientas de reporte y heurísticas utilizar**

Es importante decidir qué conjunto de heurísticas los evaluadores van a utilizar. Un conjunto seleccionado de heurísticas proporcionará pautas comunes para que cada uno de los expertos pueda hacer su evaluación. Como parte del plan de evaluación heurística, debe acordarse un sistema, un formato y las herramientas a utilizar. Esto podría ser Google Docs, hojas de cálculo y presentaciones, o alguna otra herramienta común de informes que todos puedan utilizar ya la que el “observador” tendrá acceso fácil.

**Para que las pruebas de usabilidad sean valiosas, los resultados del estudio deben identificar claramente los**

## problemas y ayudar al equipo a avanzar hacia soluciones de diseño. – The Nielsen Norman Group

### Usabilidad en dispositivos móviles:

Si bien muchos conceptos tradicionales de usabilidad se han aplicado con éxito en casos reales sobre aplicaciones para dispositivos móviles, el entorno rápidamente cambiante y las particulares del medio han hecho una necesidad imperativa del desarrollo de métodos propios al dominio

Para Zang et la usabilidad en móviles incluye nuevos retos, como: **contexto, conectividad, tamaño de pantalla, diferentes resoluciones, capacidad de procesamiento y poder limitados, y métodos de ingreso de datos**. Si bien los conceptos de usabilidad en móviles son independientes de la plataforma, tanto Google como Apple, para Android y iOS respectivamente, proveen restricciones para obtener mejor control de la usabilidad en sus aplicaciones.

### Ventajas:

1. **Puede proveer feedback rápido y a bajo costo a los diseñadores.**
2. **Se puede realizar temprano en el proceso de diseño.**
3. **Asignar la heurística correcta puede ayudar a sugerir la mejor medida correctiva.**
4. **Se puede utilizar junto a otras metodologías de pruebas de usabilidad.**
5. **Se puede identificar una gran cantidad de problemas.**

### Las 10 heurísticas de Nielsen (1994) son:

**1) Visibilidad del estado del Sistema:** El software debe informar el estado del sistema al usuario de una manera entendible, y en el tiempo adecuado.

**2) Relación entre el Sistema y el Mundo Real:** El sistema debe utilizar un lenguaje que sea conocido y familiar para el usuario, para que no se dificulte su uso.

**3) Control y libertad del usuario:** Usuarios que seleccionan opciones por error deben poder salir de ese estado “no deseado” del sistema sin tener que usar diálogos extendidos. Soporte para deshacer y rehacer.

**4) Consistencia y Estándares:** Usuario no deberían preguntarse si diferentes opciones en el sistema realizan la misma tarea. Seguir convenciones de la plataforma.

**5) Prevención de Errores:** El sistema es diseñado de tal forma que los errores se pueden prevenir antes de que ocurran. Eliminar condiciones predispuestas a errores, o validarlas y presentar al usuario con mensajes de confirmación antes de ejecutar la tarea. Página 25 de 83

**6) Reconocimiento antes que recuerdo:** Disminuir la carga de memoria en el usuario al hacer objetos, acciones y opciones visibles. El usuario no debería tener que recordar información desde un diálogo hacia otro. Instrucciones para el uso del sistema deben ser visibles o fáciles de acceder cuando sea necesario.

**7) Flexibilidad y eficiencia de uso:** Aceleradores de acciones para usuarios expertos, que no afecten el desempeño para usuarios sin experiencia con el sistema. Permitir al usuario configurar acciones frecuentes.

**8) Estética y diseño minimalista:** Diálogos no deberían contener información irrelevante que puedan desviar (o distraer) del mensaje principal.

**9) Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores:** Mensajes de error deben ser expresados en lenguaje natural, especificar el problema y sugerir una solución.

**10) Ayuda y Documentación:** La documentación debe ser fácil de buscar, enfocada en tareas del usuario, listar pasos concretos para ser realizados, y no ser muy grande.

## Heurísticas para prueba de interfaces en dispositivos móviles

Aunque las heurísticas de Nielsen son las más utilizadas en la actualidad para evaluaciones heurísticas, éstas no contemplan las particularidades que presentan aplicaciones móviles, por lo que podrían dejar pasar algunos problemas en la interfaz. El trabajo de Olibário Machado Neto y Maria da Graça Pimentel en “Heuristics for the Assessment of Interfaces of Mobile Devices”, propone heurísticas específicas para esta labor, después de compilación y validación de las mismas. Las heurísticas propuestas por dicho trabajo son:

**i. Uso del tamaño de la pantalla:** La interfaz debería ser diseñada para que los elementos no estén muy distantes, ni muy cerca. Espacios de márgenes no deberían ser muy largos en pantallas pequeñas para mejorar la visibilidad de la información.

**ii. Consistencia y estándares:** La aplicación debe mantener los componentes en el mismo lugar, y mantener consistencia visual (look & feel) a lo largo de la aplicación, para facilitar el aprendizaje y estimular la memoria a corto plazo del usuario. La misma funcionalidad debe ser realizada por similares interacciones. La metáfora de cada componente o funcionalidad debe ser única en toda la aplicación, para evitar confusiones.

**iii. Visibilidad y acceso fácil a la información:** Toda la información debe ser visible y legible, tanto en modo portrait (orientación vertical) como landscape (orientación horizontal). También aplica a multimedia, que debe ser completamente mostrada, a menos que el usuario opte por ocultarla. Los elementos en la pantalla deben ser adecuadamente alineados y contrastados.

**iv. Correspondencia del componente con su funcionalidad:** El usuario debería saber exactamente qué información debe ingresar en determinado componente, sin ambigüedades o dudas. Metáforas de funcionalidades deben ser entendidas sin dificultad.

**v. Correspondencia del mensaje sobre funcionalidad hacia el usuario:** La aplicación debe utilizar lenguaje que sea natural para el usuario, para evitar presionarlo. Instrucciones para llevar a cabo las tareas deben ser claras y objetivas.

**vi. Prevención de errores y rápida recuperación al último estado estable:** El sistema debe ser capaz de anticipar una situación que lleva a un error (por intervención del usuario). Cuando un error ocurre, la aplicación debería advertir al usuario rápidamente y regresar al último estado estable de la aplicación. En casos en los que ésta última opción es difícil, el sistema debe transferir el control al usuario, quien debe decidir qué hacer a dónde ir.

**vii. Facilidad de ingreso de información:** La forma en la que el usuario ingresa los datos puede estar basada en tecnología de asistencia ("assistive technologies"), pero la aplicación debe permitir que la información ingresada sea siempre fácil de leer, para otorgar control completo al usuario. La información debe ser ingresada de manera práctica.

**viii. Facilidad de acceso a todas las funcionalidades:** Las principales funcionalidades de la aplicación deben ser fácilmente encontradas/accedidas por el usuario, de preferencia mediante una sola interacción. Funcionalidades utilizadas con más frecuencia deberían poder ser realizadas fácilmente mediante atajos o rutas alternativas más rápidas. Ninguna funcionalidad debería ser difícil de encontrar en la interfaz de la aplicación. Todos los componentes de entrada deberían ser fácilmente asimilados.

**ix. Retroalimentación inmediata y visible:** Información para el usuario en respuesta a acciones ("Feedback"), debe ser fácilmente identificado y entendido, para que el usuario conozca el estado del sistema. Pequeñas actualizaciones de sectores de la pantalla deben ser preferidos a actualizaciones globales, porque ellas mantienen el estado de la interacción. La interfaz debe brindar la opción de esconder mensajes que aparecen repetitivamente. Actividades que toman mucho tiempo deben permitir al usuario realizar otras tareas en paralelo. La información para el usuario debe ser amable, ser positivo y no redundante u obvio.

**x. Ayuda y documentación:** La aplicación debe tener una opción de ayuda donde problemas comunes y sus formas de resolución están especificadas. Los problemas considerados en esta opción deberían ser fáciles de encontrar. xi. Reducción a la carga de memoria del usuario: El usuario no debería tener que recordar información de una pantalla a otra para completar una tarea. La información de la interfaz debe ser clara y suficiente para que el usuario pueda completar su tarea actual.

