



# Evaluaciones Heurísticas

David Baeza

01

# ¿Qué son?



Las evaluación heurísticas son métodos de inspección de usabilidad sin usuarios. Consisten en examinar un sistema de interfaz a través de distintas reglas preexistentes.

A man with glasses and a blue and white striped shirt is sitting on a red armchair. He is smiling and has his hands clasped. There are two other red armchairs on either side of him. The background is a dark grey wall.

# Heurísticas Jakob Nielsen

01

# Visibilidad del estado del sistema



El sistema debe mantener informado al usuario sobre lo que está ocurriendo en todo momento.

## 02

# Consistencia entre el sistema y el mundo real.



El sistema debe hablar el lenguaje del usuario, con idioma, palabras, frases y conceptos que le sean familiares, en lugar de términos orientados al sistema.

## 03

# Control y libertad del usuario



Es necesario tener control sobre el sistema. Siempre debe existir una “salida de emergencia” para situaciones no queridas por el usuario como la activación de funciones por error, sin tener que pasar por extensos diálogos.

## 04

# Consistencia y estándares



Los usuarios no deben verse obligados a discernir si diferentes palabras, acciones o situaciones significan la misma cosa.



## 05

# Prevención de errores



Se debe contar con buenos mensajes de error, pero es mejor lograr un diseño que prevenga en primera instancia que los errores ocurran.

# Reconocimiento en lugar de memoria.



Se debe minimizar el uso de memoria haciendo que los objetos, acciones y opciones resulten claramente visibles. El usuario no debería tener que recordar información al desplazarse de una parte a otra del sistema.

# Flexibilidad y eficiencia de uso



Los usuarios deben poder utilizar atajos y aceleradores para ejecutar sus operaciones habituales.

# Estética y diseño minimalista



Las interfaces no deberían contener información irrelevante o que sea raramente utilizada. Cada unidad adicional de información en una interfaz compite con las más importantes y disminuye su visibilidad relativa.

# Reconocimiento, diagnostico y recuperación de errores.



Los mensajes de error deben estar expresados en lenguaje llano (sin códigos), indicar con precisión cuál es el problema, y sugerir constructivamente una solución.

# Ayuda y documentación



Lo ideal es que todo sistema se auto explique, pero algunos casos es necesario generar un espacio de preguntas frecuentes para evitar posibles problemas.