

Clase 18/05 Interfaz de Usuario

Overview

Interfaz

- Método que permite interactuar a los usuarios con máquinas, aplicaciones, dispositivos

Usabilidad

- Efectividad
 - Precisión y plenitud con que los usuarios alcanzan los objetivos deseados
- Eficiencia
 - En los recursos empleados para llegar a la efectividad
- Satisfacción
 - Actitud positiva en el uso de la aplicación y a ausencia de inconformidad

Jakob Nielsen

- Doctorado en diseño de Interfaz de Usuario y Ciencias de la Computación

Heurísticas

1. Visibilidad del estado del sistema

- El diseño debería siempre mantener informado al usuario de lo que está sucediendo, otorgando feedback apropiado en intervalos de tiempo razonables.

2. Relación entre el sistema y el mundo real

- El diseño debería comunicarse con el lenguaje que entiende el usuario. Usar conceptos familiares para el usuario y no jerga técnica. Apoyarse en convenciones para mostrar la información de forma lógica y natural.

3. Control y libertad del usuario

- Los usuarios pueden cometer errores, por lo cual es conveniente que haya una forma clara de salir de la situación indeseada sin tener que atravesar un

proceso arduo.

4. Consistencia y estándares

- Seguir estándares de industria y plataforma para no dejar a los usuarios preguntándose qué significa cada cosa.

5. Prevención de errores

- Hay que hacer buenos mensajes de error, pero aún mejor es que no se den estos mensajes y que el sistema se encargue de comunicarle al usuario que la acción puede cometer errores.

6. Reconocer antes que recordar

- Minimizar las cosas que el usuario deba memorizar, haciendo las cosas lo más visibles posibles. El usuario no debe memorizar información de una parte de la interfaz a la otra.

7. Flexibilidad y eficiencia de uso

- Accesos directos pueden acortar el tiempo necesario para interacciones realizadas por un usuario experto.

8. Diseño estético y minimalista

- Las interfaces deben contener lo único e indispensable.

9. Ayudar a los usuarios a reconocer, diagnosticar y corregir los errores

- Errores deben de ser fáciles de comprender (usando lenguaje natural), indicando la totalidad del problema y sugerir una solución de forma constructiva.

10. Ayuda y documentación

- Cuanta menos documentación mejor, pero en cualquier caso es útil que se haga.

Bruce Tognazzini

Heurísticas

1. Anticipación

Equipa a tu usuario con toda la información y herramientas que pueda necesitar durante cada etapa del proceso. Para ello necesitas tener un alto conocimiento del

servicio o producto en el que trabajas y también de la persona que interactúa con aquello que has diseñado.

Por ejemplo, facilita al usuario información adicional que pueda necesitar, los recursos para obtener fácilmente datos que pueda necesitar en ese momento, etc.

2. Autonomía

La persona que interactúa con tu dispositivo, web o aplicación tiene que sentir que tiene el control para tomar las decisiones que considere oportunas. Esto no significa que tengas que ceder todo el control, sino que establezcas un marco dentro del cual el usuario pueda operar libremente. Si se siente *encajonado* puede sentirse frustrado y enfadado.

Por ejemplo, déjale cambiar el tamaño de tipografía que desee, permítele ordenar el escritorio o la pantalla principal como quiera o evita que la interfaz tome decisiones por sí misma: es lo que sucede cuando en algunas interfaces el contenido cambia de forma automática.

3. Daltonismo

Este principio está muy relacionado con la accesibilidad: ten cuidado a la hora de seleccionar los colores de la interfaz y el uso que haces de ellos. Asegúrate de ofrecer alternativas para quienes no pueden identificar o diferenciar bien algunos colores de la paleta cromática, como por ejemplo iconos.

4. Consistencia

La consistencia tiene sus pros y sus contras pero lo que no se puede es negar su utilidad. Tognazzini recomienda mantener la consistencia con los estándares existentes y entre los diferentes diseños de interfaces con los que pueda interactuar un usuario dentro de la misma compañía. Por ejemplo, en este último caso hablamos de la consistencia entre iconos, tamaño de la tipografía, espaciados, diseño de los elementos de la interfaz, diseño visual, etc.

Piensa por ejemplo en la *suite* de Office o la de Google: todos los productos respiran la misma familiaridad y tienen el mismo aspecto.



La consistencia de la suite de Google (con sus pros y sus contras)

5. Valores por defecto

Los valores estándar tienen que poder sustituirse fácilmente por el contenido que quiera escribir el usuario. En este caso concreto hablamos de los *placeholders* de los campos de texto, que quedan sustituidos por el contenido que se escribe «encima».

Con este principio Tognazzini también hace referencia a las «opciones predeterminadas» que pueda tener una interfaz: asegúrate siempre de que el usuario sabe a qué se está haciendo referencia, qué es lo que modificará y, como indica la heurística de Nielsen, que tiene la posibilidad de deshacer un error en caso de cometerlo.

6. Eficiencia del usuario

El diseño debe estar centrado en la productividad del usuario y no en la del sistema. El sistema debe ayudar al usuario a conseguir el objetivo que se había marcado de forma rápida, sin esperas innecesarias, con textos claros y directos, una buena arquitectura y una buena redacción de las opciones de los menús, botones y otros controles.

Recuerda que generalmente al navegar escaneamos más que leemos y tenemos que promover una rápida comprensión de qué está pasando y de donde hacer clic.

7. Interfaces explorables

Ligado a la autonomía: según Tognazzini se deben proveer flujos de navegación bien delimitados para que el usuario pueda alcanzar sus objetivos, pero deben existir alternativas para que los usuarios exploren. Por ejemplo, tienen que existir rutas de salida y otras formas de alcanzar el objetivo utilizando diferentes flujos.

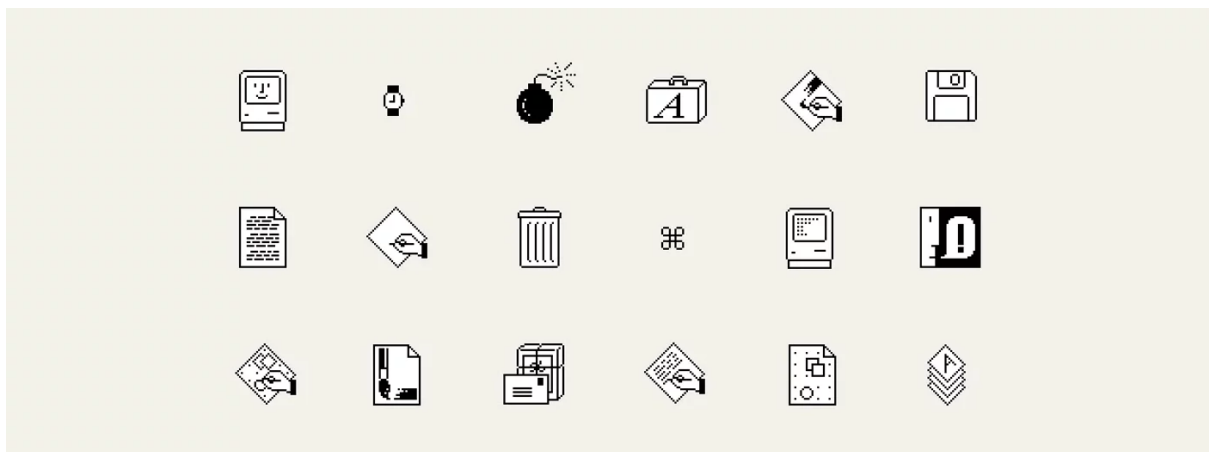
Esto puedes trabajarlo de forma fácil y clara con los diagramas de flujo.

8. Ley de Fitts

De esta ley te hablé en [Principios de UX: Ley de Hick y Ley de Fitts](#), te dejo el enlace directamente 😊

9. Objetos de interfaz humana

En mi opinión este principio quizás dejará de tener sentido en un futuro no muy lejano. Tognazzini afirmaba que los objetos de una interfaz deben corresponderse con elementos que existan en el mundo real: una carpeta para guardar diferentes documentos, una papelerera para descartar archivos que se quieran eliminar, un disquete para guardar, etc.





Primeros iconos de Apple (Susan Kare)

10. Reducción de demoras

Cuando sea posible se debe trabajar para reducir demoras en la interacción con el sistema y la interfaz. Este principio quizás te suena de Google, que siempre insiste en que la carga tiene que ser rápida: el buscador penaliza en el ranking las páginas que cargan lento. Más allá de esto, una interfaz lenta empeora la experiencia del usuario.

No te confundas, este principio no es solo tarea de los equipos de desarrollo. A la hora de diseñar tu propuesta puedes afectar la carga.

Tognazzini define estos intervalos que hay que tener en cuenta y cómo dar feedback adecuadamente:

| Expected Delay | Indication |
|------------------|--|
| 1/2 to 2 seconds | Use animated mouse cursor or other "busy" indicator  |
| > 2 seconds | Tell them potential length of wait |
| > 5 seconds | Use an animated progress indicator  Process must end by the time indicator is full! |
| > 10 seconds | Keep users a) informed & b) entertained |
| > 15 seconds | Same as >10 plus add at end a noticeable sound & strong visual indication so users know to return |

Reducción de demoras y feedback (Bruce Tognazzini)

11. Aprendizaje (learnability)

Está relacionado con el principio de «Flexibilidad y eficiencia de uso» de la heurística de Nielsen. En este caso, Tognazzini afirma que idealmente los productos no deberían tener curva de aprendizaje: los usuarios deberían poder utilizarlos correctamente desde el primer momento y dominarlos rápidamente.

Aquí puede ser útil un buen *onboarding*, pero también un buen diseño que sea usable, se entienda y permita al usuario ser eficiente (léete de nuevo el sexto principio de Tognazzini 😊).

12. Uso de metáforas

Diseña utilizando metáforas que permitan al usuario entender rápidamente un modelo conceptual.

El objetivo es generar (y utilizar) conexiones mentales creadas en el mundo «real» en interfaces digitales: el uso de la papelera que comentaba en el noveno principio es un buen ejemplo.

Tognazzini habla de metáforas que evoquen algo familiar pero que a la vez añadan un nuevo giro. Por ejemplo, si tienes que diseñar un periódico online no necesitas copiar exactamente todos los elementos de un periódico físico: puedes mantener solo algunos elementos que evoquen el objeto «original» sin llegar a copiarlo literalmente.

13. Protección del trabajo del usuario

Asegúrate, en la medida de lo posible, que el usuario nunca pierde su trabajo como resultado de un error suyo, un error en el sistema, un apagón o los problemas que puedan existir en la conexión.

14. Legibilidad

Ligado al tercer principio (contraste), Tognazzini señala la importancia de que los textos puedan leerse fácilmente, que el tamaño de la tipografía es adecuado a cada dispositivo y que las etiquetas que identifican la navegación, botones y otros elementos con los que se interactúa se escriban con palabras clave en mente.

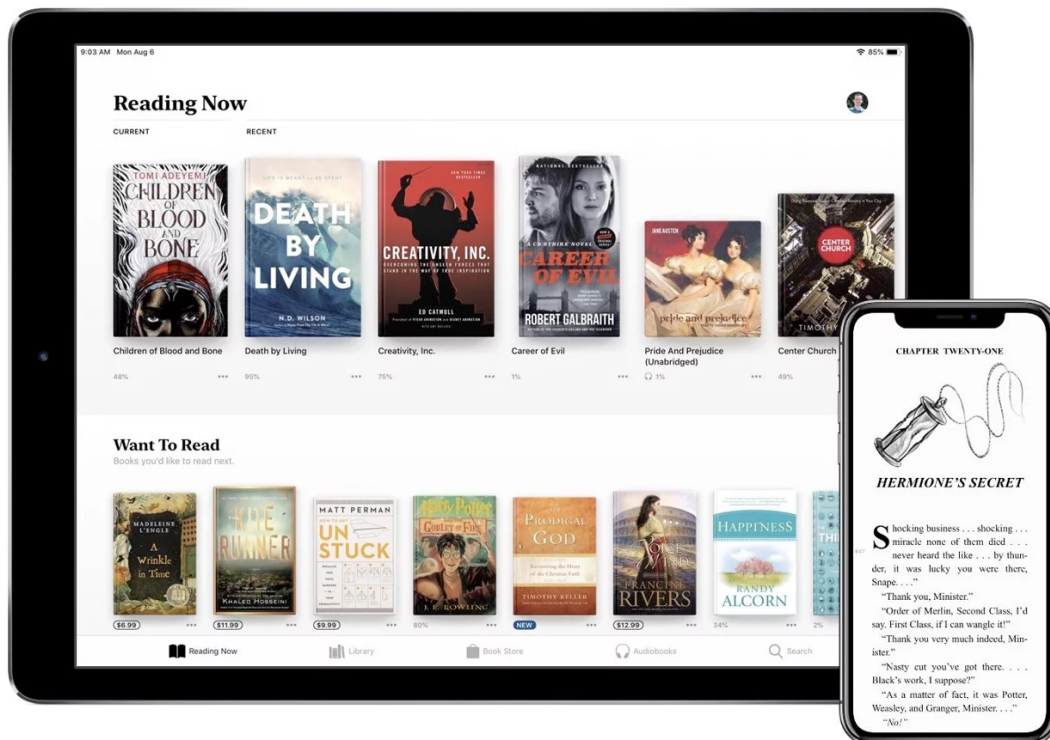
Realiza tests de usabilidad también con los usuarios más mayores que puedan estar utilizando tu interfaz: muchas veces tenemos tendencia a hacer tests con individuos de edad media o jóvenes, dejando de lado la población más envejecida.

15. Registro del estado

Es importante que el sistema guarde toda la información que permita mejorar la experiencia al usuario. Por ejemplo:

- Si es la primera vez que utiliza un sistema
- Dónde está el usuario
- Dónde quiere ir el usuario
- Dónde ha estado el usuario en esta sesión
- Dónde abandonó el usuario la última sesión

Si buscas un ejemplo concreto, piensa en la aplicación Books de Apple (o cualquier otra que te permita leer libros electrónicos): siempre saben indicarte en qué página te quedaste la última vez, te informan de qué página es, cuantas te quedan y si «cierras» el libro puedes recuperar tu anterior posición.



La aplicación «Books» de Apple (MacStories)

Otro ejemplo serían las plataformas que te mantienen identificado aunque hayas cerrado la pestaña. O el propio YouTube, que te indica cuanto te queda por ver de un vídeo y tiene una página con tu histórico que te permite recuperar cualquier vídeo visualizado previamente.

16. Navegación visible

Este principio es especialmente importante si tu producto o servicio es complicado. Si estás en España, piensa por ejemplo en la página web de la Agencia Tributaria: tiene muchísimas páginas y a veces es difícil hacerse un mapa mental correcto.

Reduce la navegación al máximo y ofrece la mínima imprescindible de forma clara y natural. Si el usuario tiene dudas o no se siente cómodo, se irá y probablemente no volverá.