Las 8 reglas del diseño de interfaces

1. Esforzarse por la consistencia: Es una de las reglas más infringidas, Esta regla indica que se debe utilizar los mismos patrones de diseño y las mismas secuencias de acciones para situaciones similares.
2. Atender las diferencia universales: Las diferencias de las personas nos da una infinidad de posibilidades en el desarrollo, debemos utilizar estas para crear algo buenos para todos: por ejemplo crear instrucciones simples para los usuarios menos experimentados y teclas de función o comandos ocultos para los usuarios mas experimentados
3. Retroalimentación informativa: Es necesario mantener a los usuarios informados de lo que está ocurriendo en cada etapa de su proceso. Esta retroalimentación debe ser significativa, relevante, clara y adaptada al contexto. Mientras las Acciones sean mas importantes respuestas deben ser sustanciales y mientras las Acciones son poco importantes las respuestas deben ser modestas
4. Diseñar acciones secuenciales: Las secuencias de acciones necesitan tener un principio, medio y final. Una vez que una tarea sea completada, es bueno otorga un poco de tranquilidad al usuario proporcionándole comentarios informativos y opciones bien definidas.
5. Prevenir errores: Una buena interfaz debe estar diseñada para evitar errores tanto como sea posible. Pero cuando los errores ocurren, el sistema necesita hacer que sea fácil para el usuario entender el problema y saber cómo resolverlo.
6. Permitir una fácil reversión de acciones: Las acciones que se cometan deberían ser fácilmente reversibles, como colocar un botón para retroceder a la ventana anterior, esto lograra que los usuarios se sientan más tranquilos al momento de explorar todas las opciones del programa.
7. Soporta un punto de control interno: Es importante proporcionar control y libertad a los usuarios para que ellos puedan sentir que están a cargo del sistema, y ​​no al revés. Es bueno Evitar sorpresas, interrupciones o cualquier cosa que no haya sido solicitada por los usuarios.
8. Reducir la carga de memoria a corto plazo Nuestra capacidad de atención es limitada y cualquier cosa que podamos hacer para facilitar el trabajo de los usuarios, es mejor. Es mucho más fácil reconocer la información que recordarla

PREVENCION ERRORES

Una frase importante para esta parte es: No hay medicina contra la muerte y contra el error no se ha encontrado ninguna regla. Una forma de reducir la pérdida de productividad debido a errores es mejorar los mensajes de error proporcionados por la interfaz ya que mejores mensajes de error pueden aumentar las tasas de éxito en la reparación de errores, reducir las tasas de error futuras y aumentar la satisfacción subjetiva. En lugar de utilizar frases como syntax error o signos de interrogación, es mucho mejor utilizar frases como por favor prenda la impresora

Es más eficaz prevenir que ocurran errores y aunque parezca imposible la verdad es que es bastante factible.

El primer paso es entender la naturaleza del error.

Una perspectiva es que las personas comenten errores en los que los diseñadores pueden ayudar a evitarlos creando menús y pantallas bien organizadas y haciendo que sea difícil que los usuarios cometan acciones irreversibles

Los productos exitosos deben ser diseñados de tal manera que se evite cometer errores graves por su incorrecto uso como los carros a los que no se puede poner en reversa mientras van a una alta velocidad.

Cuando una acción requiere muchos pasos para completarse, las personas pueden olvidar algún paso, es mejor juntar todas las secuencias de acciones en una sola. Como por ejemplo algunos programas ejecutan con frecuencia los comandos de establecer parámetros de comunicación, iniciar sesión y cargar archivos. Afortunadamente, la mayoría de los paquetes de software de comunicaciones permiten a los usuarios especificar estos procesos una vez y luego ejecutarlos simplemente seleccionando el nombre de proceso apropiado.

**Integración de la automatización mientras se preserva el control humano**

Los usuarios quieren simplificar las tareas, estos pueden evitar las rutinas, las tareas tediosas y los errores en las tareas, con la automatización. Incluso con el incremento de la automatización, los diseñadores pueden ofrecer interfaces controlables donde si bien existen varios procesos automatizados internamente, el usuario puede controlar varios aspectos de la aplicación a su antojo. Aun con la automatización el juicio humano es necesario en eventos impredecibles. **<DECIR EJEMPLO>**

**Las máquinas son buenas en situaciones en las cuales se requiere realizar tareas monótonas y extensas. Mientras que los humanos son mejores en situaciones imprevistas de toma de decisiones. La meta del diseño de un sistema en muchas aplicaciones, es dar a los operadores la suficientes información acerca de la actividad actual. Hay una controversia sobre si crear interfaces similares a herramientas o buscar agentes autónomos adaptativos que se anticipan a las necesidades.**

**Otra resolución de la controversia es aceptar el control del usuario en la interfaz, pero considere la programación de agentes o de múltiples agentes para automatizar procesos internos como la asignación de espacio en disco o el enrutamiento de red en función de las cargas actuales. Sin embargo, se trata de adaptaciones basadas en características del sistema, no perfiles de usuario.**

**<LEER DIAPOSITIVAS>**

El nivel conceptual es el "modelo mental" del usuario del sistema interactivo. Dos modelos mentales para la creación de imágenes son los programas de pintura que manipulan píxeles y los programas de dibujo que operan en objetos. Los usuarios de programas de pintura piensan en términos de secuencias de acciones en píxeles y grupos de píxeles, mientras que los usuarios de programas de dibujo aplican operadores para alterar y agrupar objetos

Nivel semántico: Por ejemplo, la eliminación de un objeto en un programa de dibujo podría lograrse deshaciendo una acción reciente o invocando una acción de eliminación de objeto. Cualquiera de las dos acciones debe eliminar un solo objeto y dejar el resto intacto

El nivel sintáctico se define cómo las acciones del usuario que transmiten semántica se ensamblan en oraciones completas que instruyen a la computadora para realizar ciertas tareas. Por ejemplo, la acción de eliminar archivos podría invocarse mediante una selección de varios objetos, seguida de una pulsación de tecla, seguida de una confirmación.

El nivel léxico se ocupa de las dependencias de los dispositivos y de los mecanismos precisos mediante los cuales los usuarios especifican la sintaxis (por ejemplo, una tecla de función o un doble clic del mouse en 200 milisegundos

1. Escriba una de las 8 reglas de oro del diseño de interfaz.
2. No hay medicina contra \_\_\_\_\_\_\_\_ y contra \_\_\_\_\_\_\_\_ no se ha encontrado ninguna regla.
3. La muerte, el error (Correcto)
4. El dolor, el error
5. El error, la muerte
6. Con la automatización de los sistemas ya no es necesario el juicio humano.
7. Verdadero
8. False (Correcto)
9. El nivel semántico describe el significado transmitido por la entrada del usuario y por la salida de la pantalla de la computadora
10. Verdadero (Correcto)
11. Falso