

# 1.- Concepto de Usabilidad.

Cuando diseñamos aplicaciones multiplataforma, la interfaz que planteemos será fundamental para que el usuario se sienta cómodo trabajando con ella. Un mal diseño puede provocar que los clientes abandonen nuestra aplicación.

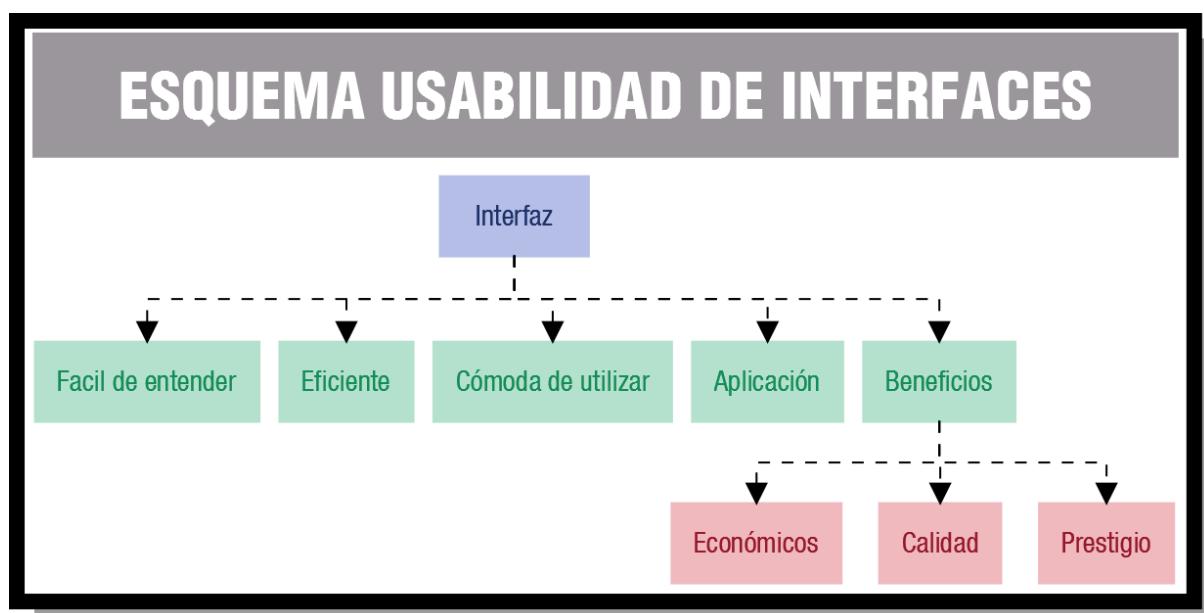
Dentro del contexto de aplicaciones informáticas y desarrollo de software, se define **usabilidad** como la disciplina que estudia la forma de diseñar sitios webs y aplicaciones para que los usuarios puedan interactuar con ellos de la forma más fácil, cómoda e intuitiva posible.

Este concepto tiene en la actualidad gran relevancia, ya que cuanto más sencillo le resulte a un usuario navegar por una aplicación, más posibilidades tendrá de realizar de forma eficiente su trabajo y mayor éxito tendrá nuestro software.

Por todo esto, vamos a tratar en la presente unidad de introducirnos en el campo del diseño usable de aplicaciones y entender la necesidad de invertir tiempo en determinar criterios de usabilidad que garanticen, en la medida de lo posible, la aceptación del software que diseñemos.

Veremos qué normativa es de aplicación en este campo, qué principios básicos garantizarán la usabilidad y qué parámetros de diseño de la interfaz contribuirán a facilitar el uso de nuestro software a los usuarios.

Definir adecuados parámetros de usabilidad en el diseño de nuestros productos, se traducirá en beneficios económicos, de calidad y de imagen de nuestra empresa.



## 1.1.- Características.

Las características más importantes de una interfaz usable son las siguientes:

- **Útil**: que sea capaz de cumplir las tareas específicas para la cual se ha diseñado.
- **Fácil de usar**: que sea eficiente, veloz y con la menor cantidad de errores posibles.
- **Fácil de aprender**: que no se necesite excesivo tiempo en aprender a trabajar con la aplicación y que sea sencillo recordar su funcionamiento.
- **Elegante en su diseño**: para favorecer la percepción del usuario y sus emociones.

**En el diseño de la interfaz de usuario debemos tener en cuenta que los usuarios deben ser capaces de alcanzar sus objetivos con el mínimo de esfuerzo y con máximos resultados.**

Otras características también importantes, que no siempre son tenidas en cuenta en el diseño de interfaces, son:

- Previsión de errores de los usuarios al navegar por la interfaz: debemos proveer de mecanismos de recuperación de errores y de reversibilidad de acciones.
- Retroalimentación de la interfaz ante acciones del usuario, para evitar que se sienta perdido.
- Simplicidad de diseño de la interfaz: evitar sobrecargarla será esencial para un buen uso de la misma.

En definitiva, hay que tener en cuenta los siguientes aspectos de una interfaz usable:

## DOCUMENTO RESUMEN CARACTERÍSTICAS INTERFACES USABLES

Interfaz  
usuable



## **1.2.- Atributos.**

En el diseño de interfaces gráficas, usamos elementos como la tipografía, símbolos, iconos, color... para representar hechos, conceptos y emociones. Es necesario saber que su correcta manipulación puede hacer que se cumplan o no adecuadamente los objetivos planteados por el sistema. Por ello, los principios básicos que no debes olvidar son:

- Organizar.
- Economizar.
- Comunicar.

**El diseño de la interfaz debe caracterizarse por su simplicidad y estética haciéndola accesible a todas las personas y generando un entorno agradable que contribuya al entendimiento de los usuarios de una forma efectiva.**

Los cinco **atributos de la usabilidad** son:

1. Facilidad de aprendizaje: El sistema debe ser fácil de aprender.
2. Eficiencia: Una vez que sepa manejar la aplicación, el usuario debe adquirir un nivel alto de productividad.
3. Recuerdo de utilización en el tiempo: El uso de la aplicación debe ser fácil de recordar.
4. Tasas de error: Cuántos errores comete el usuario por unidad de tiempo.
5. Satisfacción: Qué grado de satisfacción tiene la utilización de la aplicación para el usuario.

Beneficios de la usabilidad:

- Reducción de los costes de aprendizaje y ayuda al usuario.
- Optimización de los costes de diseño, rediseño y mantenimiento.
- Disminución de la tasa de errores cometidos por el uso en el uso cotidiano de la aplicación.
- Aumento de la satisfacción de los usuarios y del prestigio de la marca.

## **2.- Normas relacionadas con la usabilidad.**

### **ISO 25000 .. Calidad del software y Métricas de Evaluación.**

Esta norma sustituye a la norma ISO 9126 y contribuye a la calidad del software construido en diversas categorías, entre las cuales está la usabilidad. Se define usabilidad como "La capacidad del componente para ser entendido, comprendido, usado y atractivo para el usuario cuando se usa bajo unas determinadas condiciones".

Esta definición se centra en los requerimientos del producto, los cuales le dan funcionalidad y eficiencia. La usabilidad no sólo depende del producto, sino también del usuario, por ello, se entiende dentro de un contexto determinado y con usuarios particulares.

#### **ISO 9241.- Requisitos del software en relación a la calidad de su uso.**

"Usabilidad es la eficacia y satisfacción con la que un producto permite alcanzar objetivos específicos a usuarios específicos en un contexto de uso específico".

Esta es una definición centrada en el concepto de calidad en el uso, es decir, se refiere a cómo el usuario realiza tareas específicas en escenarios específicos con efectividad.

#### **ISO 14915.- Ergonomía del software para interfaces de usuarios multimedia.**

En general, hace recomendaciones acerca del diseño de controles y navegación (por ejemplo, controles de audio con las funciones "play", "stop"...). Asimismo, proporciona recomendaciones sobre medios específicos y dominios específicos como la formación asistida por ordenador.

### **3.- Medida de usabilidad de aplicaciones.**

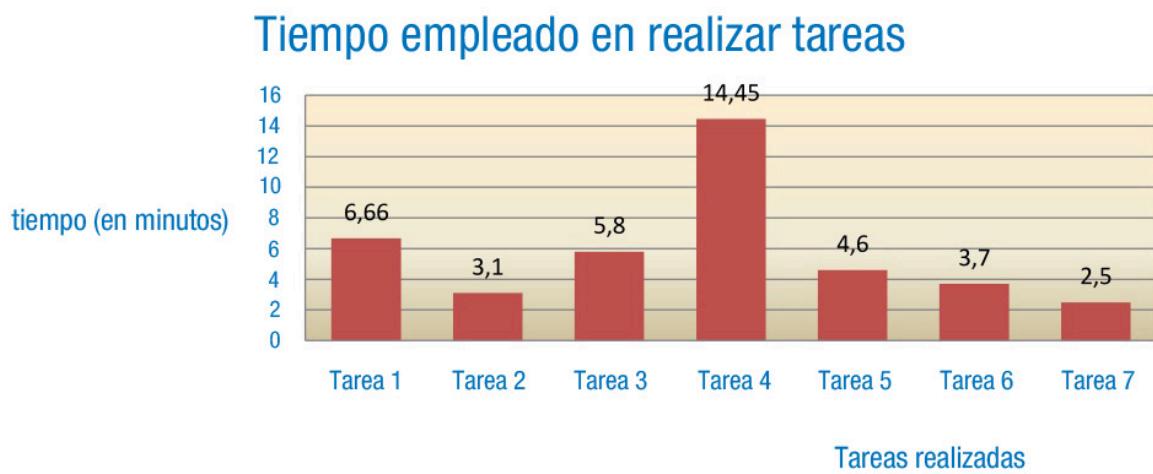
Se definen las métricas de usabilidad, cómo aquellas características de la interfaz que son medibles de forma objetiva. Es decir, dejando a un lado interpretaciones personales, se trata de encontrar una forma que evalúe la usabilidad de manera cualitativa y cuantitativa. Estas características se suelen dividir en tres grupos:

1. **Efectividad.** - Mide la plenitud con la que se alcanza un objetivo concreto. Algunas de las variables que se emplean para medir la efectividad son: tanto por ciento (%) de tareas completadas; tanto por ciento (%) de tareas completadas en el primer intento; tanto por ciento (%) de usuarios que completan la tarea en el primer intento, etc.
2. **Eficiencia.** - Mide el esfuerzo para conseguir un objetivo. Algunas de las variables típicas que se emplean para medir este valor son: tiempo empleado en completar una tarea; tanto por ciento (%) de errores cometidos; tiempo empleado en recuperarse de los errores producidos; número de clics necesarios para realizar una tarea, etc.
3. **Satisfacción.** - Mide el grado de satisfacción del usuario. Algunas de las variables que se utilizan para medir la satisfacción son: tanto por ciento (%) de usuarios que recomendarían la aplicación a un amigo; tanto por ciento (%) que califican el producto como fácil de usar, etc.

Para medir las métricas anteriores, es necesario contar con un grupo numeroso de usuarios y seguir las siguientes pautas:

1. **Definir las tareas** que van a realizar los usuarios y usuarias. Para ello, lo más conveniente es centrarse en aquellas tareas que suponemos más complejas o que requieren más tiempo para ejecutarlas.

2. **Establecer los objetivos** para las tareas seleccionadas. Es decir, qué objetivo concreto pretendemos que alcance el usuario.
3. **Definir qué variables se van a medir** durante el proceso. Normalmente, se mide el tiempo que requiere la realización de una tarea o grupo de tareas, el porcentaje de error al realizar las tareas, el porcentaje de tiempo en que los usuarios siguen una ruta de acciones de forma óptima y el número de veces que es necesario retroceder en la aplicación por encontrarse desubicado.
4. **Planificar cómo se van a recoger los datos.**



Una vez recogidos los resultados, hay que proceder a su análisis. Para ello, lo más normal es utilizar gráficos donde se mide el tiempo necesario para ejecutar una determinada tarea, ya que su interpretación es directa y sencilla.

Para sacar conclusiones, exploraremos los gráficos en busca de problemas de usabilidad. Una vez detectados, se procede al rediseño de la interfaz (en aquellos puntos más conflictivos) y se vuelven a realizar las pruebas.

## 4.- Pruebas de expertos.

Los usuarios expertos contribuyen a las pruebas de usabilidad detectando errores del sistema, basando sus opiniones en su propia experiencia.

**Su objetivo principal es detectar elementos de diseño de la interfaz que pueden confundir al usuario y restar calidad a la aplicación.**

Tipos de pruebas con usuarios expertos:

- **Evaluación heurística.** Verificación frente a heurísticas de diseño.
- **Revisión de normas:** Se revisa la interfaz para asegurarnos de que cumple con las normas establecidas.
- **Inspección de consistencia:** Se hace a través de una familia de interfaces que conforman la aplicación.

- **Inspección formal de usabilidad:** Técnica de evaluación formada por un grupo de expertos que realizan una especie de juicio, con uno de los participantes actuando como moderador, destacando las fortalezas y debilidades de la aplicación. En concreto, se pretenden encontrar problemas de
  - Diseño: color, vocabulario, presentación...
  - Navegación: controles, menús, búsquedas...

El estudio examina la interfaz y su cumplimiento con los principios clásicos de usabilidad.

También existe otra técnica de evaluación llamada **caminata cognitiva**, en la cual un grupo de expertos simula la manera en la cual un usuario caminaría por la interfaz al realizar tareas particulares.

**Las pruebas con expertos suelen ser una manera rápida y económica de evaluar el diseño de la interfaz y encontrar aquellos problemas que un experto puede detectar desde la perspectiva de un usuario.**

Los expertos, usan en sus evaluaciones, unos principios generales (heurísticos) obtenidos de la experimentación sistemática en IPO/HCI (Interacción Persona-Ordenador).

## 5.- Pruebas con usuarios.

Las pruebas con los usuarios nos van a permitir obtener una valiosa información que será la base para lograr un diseño centrado en el usuario, sobre la que se construirán las etapas posteriores de explotación y mantenimiento.

Metodologías de pruebas con usuarios:

a. **Reunión**

Son encuentros que se hacen durante distintas etapas del proceso de desarrollo de software. Es aconsejable que en su realización siempre haya presente un experto actuando como moderador.

b. **Entrevistas y encuestas**

Son contactos personalizados con usuarios y se pueden realizar de manera oral o escrita. Según se realice, obtendremos información cualitativa o cuantitativa, respectivamente.

Las encuestas deben ser escritas y revisadas por especialistas para asegurarnos que se van a evaluar los aspectos más complejos de la interfaz. Si la encuesta es on-line pueden ser colocadas en un sitio web, enviadas por correo directamente al grupo elegido de usuarios o colocada en algún grupo de noticias.

c. **Diseño de escenarios**

Es una variante de la encuesta donde se les pide a los usuarios que definan el orden de las acciones que realizan para lograr algún objetivo específico. Su objetivo no es otro que obtener las secuencias lógicas de acción en la consecución de

alguna tarea.

d. **Diseño participativo**

Se trata de una reunión entre los productores y una muestra de usuarios potenciales del producto final. El objetivo es que los usuarios participen en el diseño de la interfaz.

Para ello, se toma nota de sus necesidades y expectativas y los productores les mostrarán la idea general de cómo quedaría el producto final.

Se pueden aplicar metodologías clásicas de debate, como son la mesa redonda, la tormenta de ideas...

**Una muestra de 5-10 usuarios es, en la mayoría de los casos, suficiente para detectar muchos problemas de usabilidad, pero los resultados no tienen ninguna validez estadística.**

La prueba con usuarios permitirá que detectemos cuáles son las tareas que les resultan más difíciles, así como los elementos de la interfaz que son menos comprensibles.

Toda esta información será analizada y se obtendrá una lista de prioridades que permitan un rediseño de la interfaz con las correcciones que mejor ayuden a hacer que la aplicación sea lo más usable posible.

## 6.- Pautas de diseño de la interfaz de usuario.

Los diseñadores de interfaces siempre dicen que la interfaz de usuario debería ser la primera cosa a diseñar a la hora de desarrollar una aplicación, ya que el esfuerzo y conocimiento sobre nuestra interfaz mejorará la usabilidad de la aplicación de manera radical.

**Las pautas de diseño pueden resumirse en:**

- **Organizar todos los elementos que conforman la interfaz de una manera clara y consistente.**
- **Economizar elementos y contenidos, para comunicar lo máximo con la mínima cantidad de elementos.**
- **Comunicar, ajustando la presentación de contenidos a las capacidades del usuario.**



Francisco Javier Cabrerizo (Elaboración propia)

Pautas de diseño que deben tenerse en cuenta:

- **Diseño visual.**  
Un buen diseño visual está centrado en aumentar la comunicación.
- **Color.**  
El color debe ser considerado como una herramienta adicional en el diseño, no una necesidad básica. Se recomienda aplicar un conjunto limitado de colores, siendo los colores apagados, sutiles y complementarios los más apropiados en el diseño de interfaces de corte empresarial y académicas.
- **Matiz, contraste y resplandor.**  
Los usuarios con desórdenes visuales requieren alternativas de colores por defecto de una aplicación. Una buena interfaz de usuario se anticipa a estas necesidades, proporcionando una opción para la personalización de las preferencias de color.
- **Disposición de las ventanas tipo formulario.**  
La localización visual de los componentes es importante porque la relación entre los componentes es indicada por su posición. Una disposición limpia es crucial para crear un flujo visual de información sin problemas para el usuario.
- **Redacción de texto en la interfaz.**  
Hay que tener en cuenta:
  - Brevedad: Es más probable que los usuarios lean bloques cortos de textos que bloques largos.
  - Lenguaje: Utiliza la gramática estándar, un lenguaje claro y coherente en la interfaz mejora la facilidad de uso de la aplicación. Escribe frases positivas. Retroalimenta los errores informando brevemente al usuario cuál es el problema y qué debe hacer para resolverlo.

## 6.1.- Estructura de la interfaz de usuario.

Los principios básicos para el diseño de la estructura de interfaces de usuario son:

- **Familiaridad del usuario.**  
El ordenador, la interfaz y el entorno de trabajo deben estar a disposición del usuario. Se debe dar al usuario el ambiente flexible para que pueda aprender rápidamente a usar la aplicación. Se utilizarán términos y conceptos que se toman de la experiencia de las personas que más utilizan el sistema.
- **Consistencia.**  
Siempre que sea posible, la interfaz debe ser consistente. Esto implica que operaciones similares tendrán que activarse de la misma forma. Además, se debe tener presente la ergonomía mediante menús, barras de acciones e iconos de fácil acceso e identificación.

- **Legibilidad.**

Para que la interfaz favorezca la usabilidad de la aplicación, la información que se muestra debe ser fácil de ubicar y leer. Es importante hacer una clara presentación visual (colocación/agrupación de objetos, evitar la presentación de excesiva información...). Las interacciones se basarán en acciones físicas sobre elementos de código visual o auditivo (iconos, botones, imágenes, mensajes de texto, barras de desplazamiento...) y en selección con sintaxis y órdenes. En la interfaz del ejemplo observamos un diseño con una estructura completamente contraria a la usabilidad.

- **Mínima sorpresa.**

El comportamiento de la aplicación no debe provocar sorpresa a los usuarios.

- **Recuperabilidad.**

La interfaz debe incluir mecanismos para permitir a los usuarios recuperarse de los errores.

**Una interfaz coherente debe integrar la interacción del usuario y la presentación de la información.**

Para encontrar la mejor presentación de la información es necesario conocer a los usuarios y la forma en que utilizarán la aplicación. Factores a considerar:

- ¿El usuario está interesado en información precisa o en las relaciones entre los diferentes valores de los datos?
- ¿Qué rápido cambian los valores de la información? ¿Se indicarán de forma inmediata al usuario los cambios de un valor?
- ¿El usuario debe llevar a cabo una acción en respuesta a los cambios de la información?
- ¿El usuario necesita interactuar con la información desplegada vía una interfaz de manipulación directa?

En las siguientes páginas vamos a tratar de los elementos que conforman la estructura de la interfaz. Se trata de los menús, las ventanas, los cuadros de diálogo y los atajos de teclado. Seguramente ya los conoces de sobra, pero quizás no te has parado a pensar de qué forma hay que emplearlos para que tu aplicación sea más usable. Es lo que se pretende en los siguientes apartados.

### **6.1.1.- Menús.**

Un menú es una lista de opciones que se muestran en la pantalla o en una ventana de la pantalla para que los usuarios elijan la opción que deseen.

Los menús permiten dos cosas:

- Navegar dentro de un sistema, presentando rutas que llevan de un sitio a otro.

- Seleccionar elementos de una lista, que representan propiedades o acciones que los usuarios desean realizar sobre algún objeto.

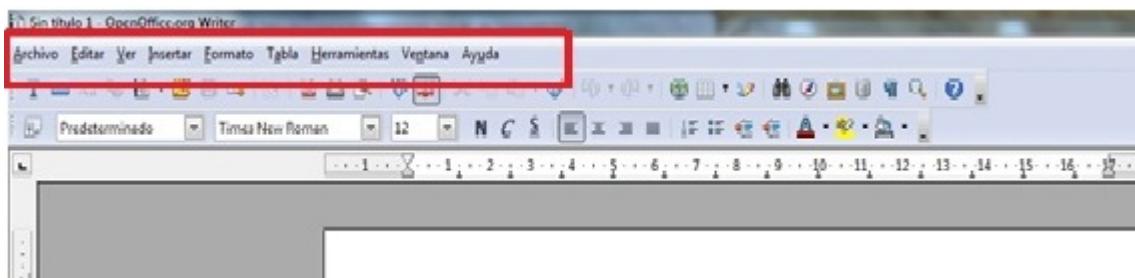
En el diseño de un menú hay que especificar su título, las opciones que se mostrarán y qué acciones van a tener lugar cuando el usuario elija esa opción.

Ventajas de uso:

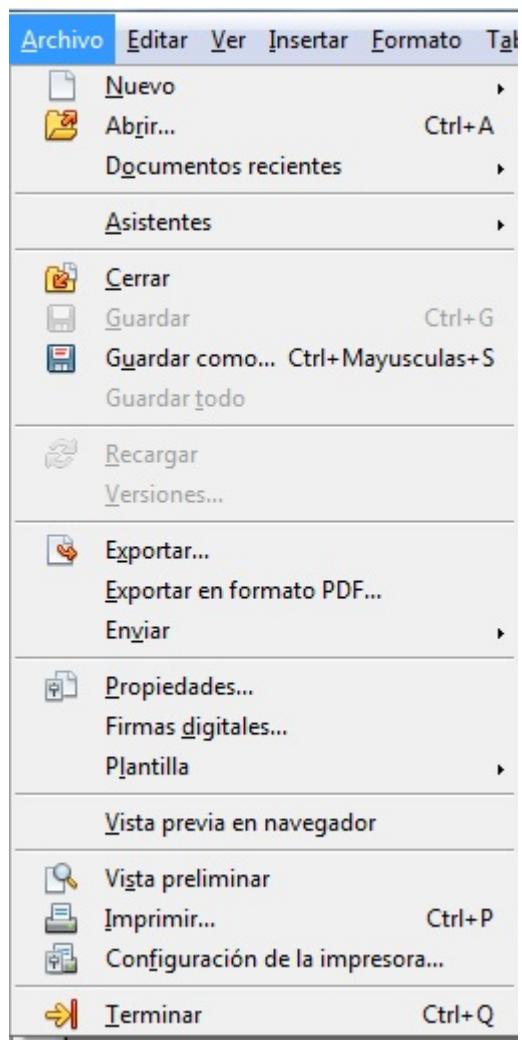
- Permiten que la pantalla se mantenga despejada.
- Evitan que el usuario siga viendo aquellas opciones del menú que ya no son de su interés.
- Permiten un movimiento rápido del usuario a lo largo del programa.

En el diseño de interfaces de usuario usables, podemos definir distintos tipos de menús. Los más usados son los siguientes:

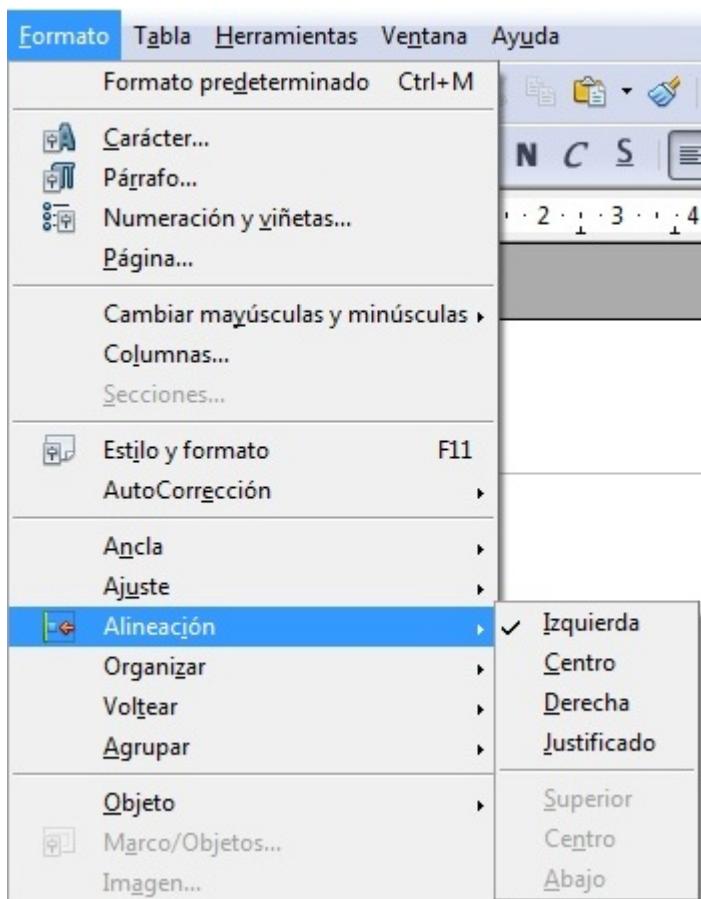
**a) Los menús de barras:** situados en la parte superior de la pantalla, son muy utilizados en las aplicaciones actuales. Contienen una lista de acciones genéricas que dan paso a menús desplegables dónde se concretan.



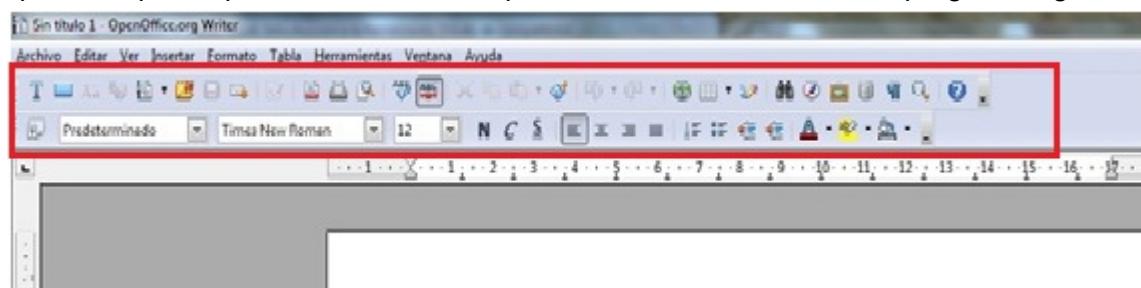
**b) Menús de barras y menú despegables:** son menús que presentan una lista de opciones que a su vez presentan otros menús. Se denominan menús en cascadas.



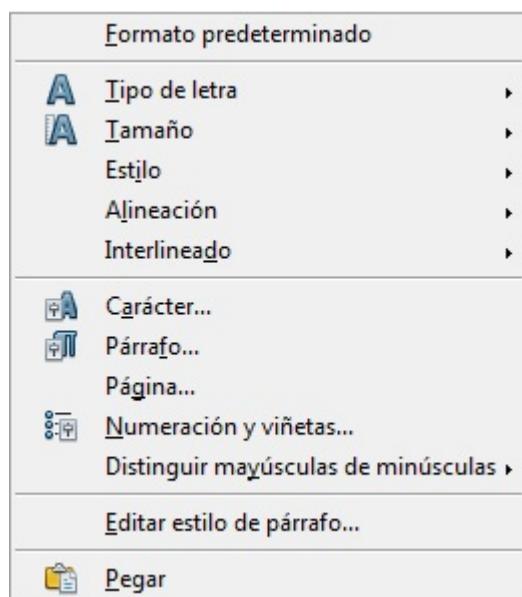
Pueden cambiar dinámicamente y deshabilitar opciones que no estén disponibles en un momento dado (mostrándose normalmente de color gris).



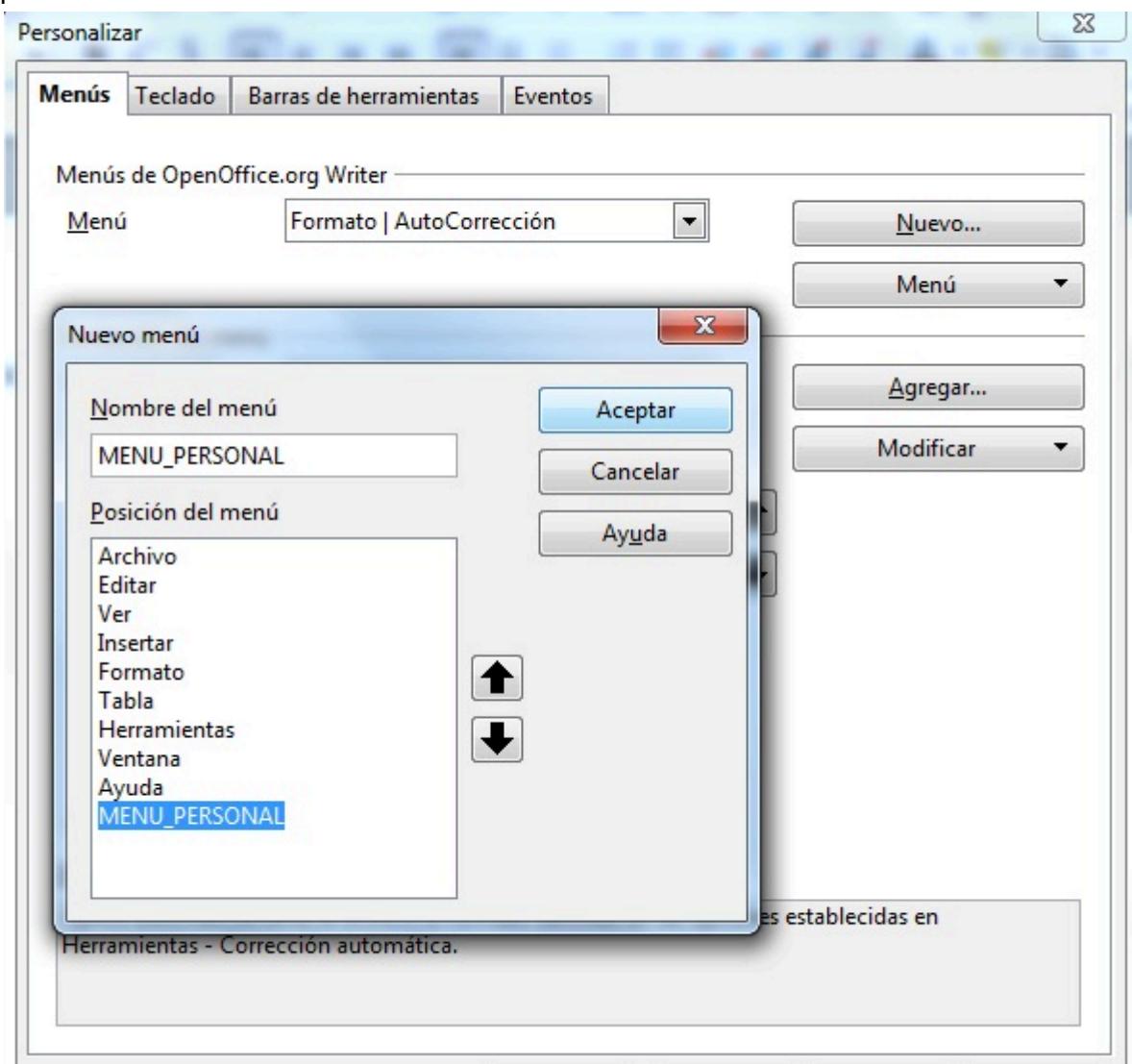
c) **Paletas o barras de herramientas:** son menús gráficos con acciones, herramientas y opciones que se pueden colocar en la pantalla. Se utilizan mucho en programas gráficos.



d) **Menús contextuales o menús pop-up:** el contenido del menú depende del contexto de trabajo del usuario. Contienen únicamente las opciones que son aplicables al objeto seleccionado, más algunas de uso frecuente que también son accesibles desde el menú de barra.



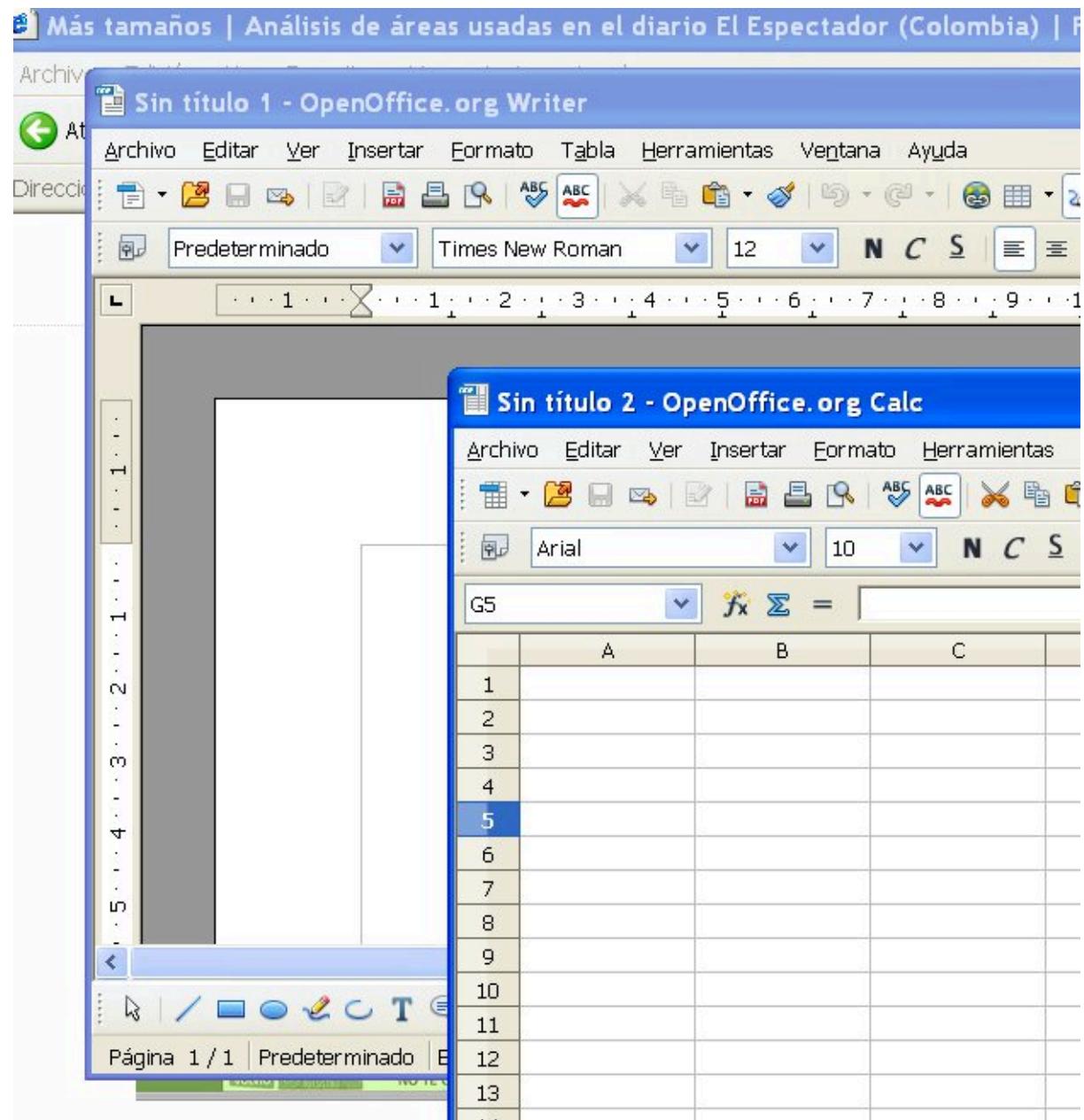
e) **Personalización:** creación de menús personalizados o modificación de los existentes. Es muy interesante que el usuario tenga la posibilidad de intervenir en el diseño de los menús, personalizando los menús de acuerdo a sus necesidades.



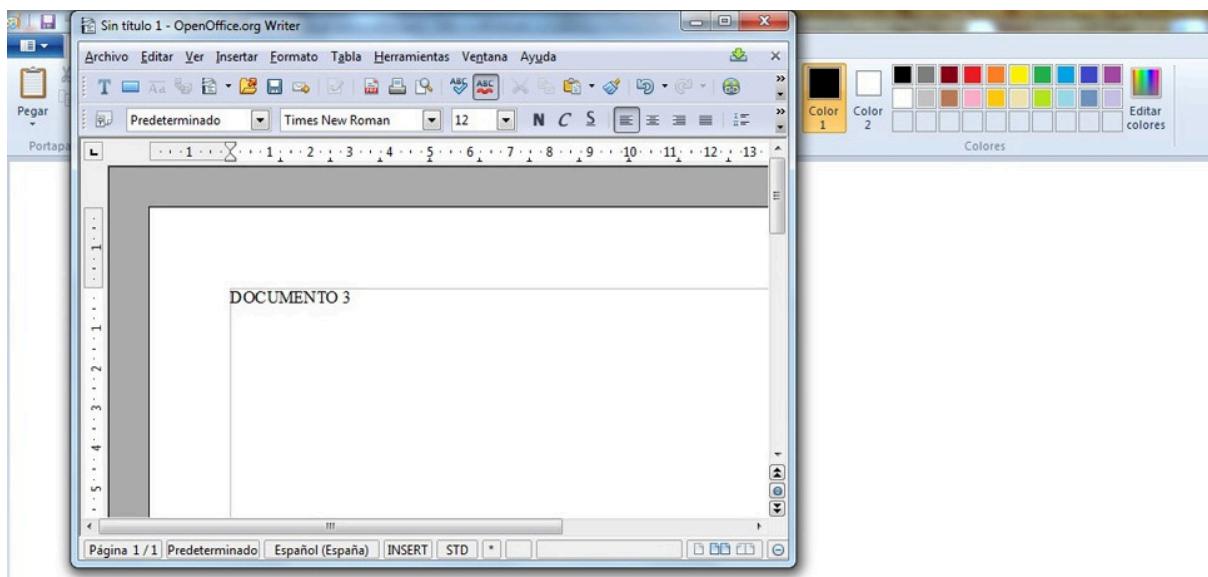
## 6.1.2.- Ventanas.

La ventana es un área particular de la pantalla dedicada a un propósito particular. Las ventanas pueden moverse, ocultarse, minimizarse,... permitiendo al usuario o usuaria controlar la apariencia de la interfaz: puede abrir, cerrar, restaurar y maximizar los espacios que contienen la información.

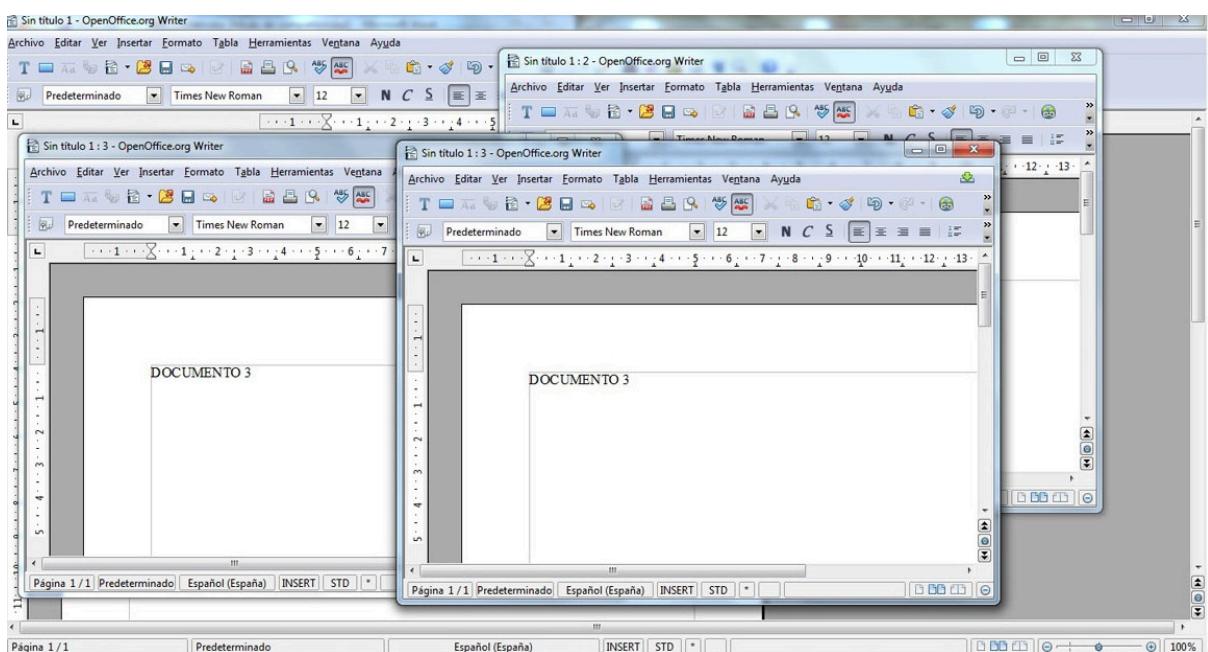
La ventaja más notable de usar pantallas en la interfaz es que facilita una captura rápida y correcta.



En el diseño de interfaces de usuario usables, es vital el uso que delimitan los espacios dedicados a una tarea concreta.



Si aplicamos los criterios de usabilidad al uso de ventanas, debemos de tener claro que utilizar un número excesivo de ventanas en la aplicación producirá confusión y molestias innecesarias al usuario:

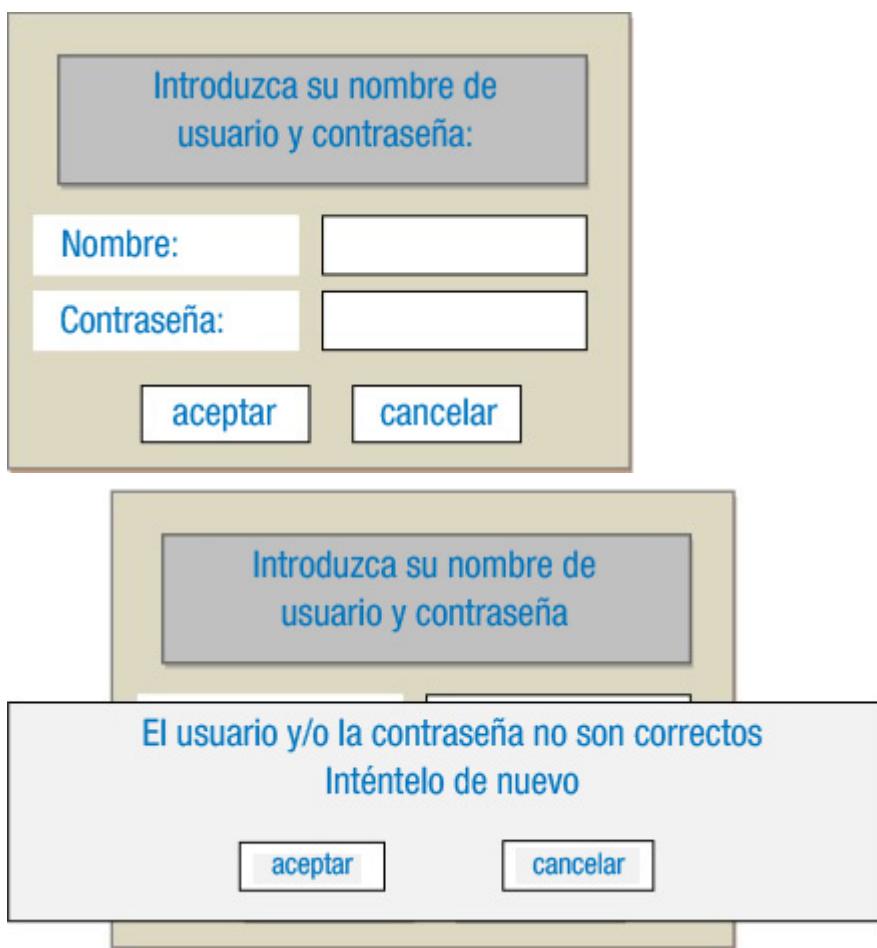


### 6.1.3.- Cuadros de diálogo.

Son mensajes producidos por el sistema en respuesta a las acciones del usuario. También se utilizan para el sistema de ayuda en línea y como parte de la documentación suministrada con el sistema.

Los factores a tener en cuenta a la hora de su diseño son:

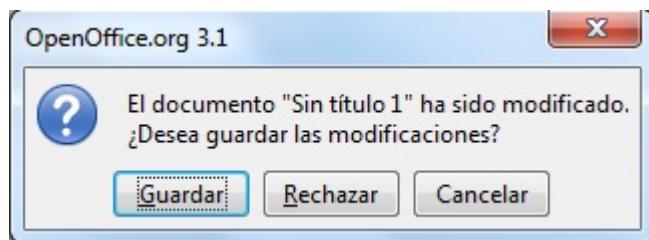
- Nivel de habilidad del usuario.  
Conocer al usuario y sus habilidades implica adecuar los mensajes a la terminología que el usuario utiliza. (Por ejemplo, los usuarios principiantes no comprenderán los mensajes demasiado concisos).
- Estilo.
- Los mensajes deben ser positivos y activos.
- Cultura. (reconocer la cultura del país evitará mala interpretaciones del contexto del mensaje).



En el diseño de interfaces de usuario usables, los cuadros de diálogo son muy importantes, ya que contribuyen a la retroalimentación de información entre el sistema y el usuario. Hay que tener en cuenta que, ya que son los responsables de informar al usuario sobre el éxito o fracaso de las acciones que pretende ejecutar, la sintaxis debe de ser clara, breve y sencilla, expresada en un lenguaje natural y favoreciendo la positividad. No debemos de olvidar que si el cuadro pretender informar al usuario sobre un error, debería de informar también al usuario sobre la manera de resolverlo.

Existen diferentes tipos de cuadros de diálogo:

a) Para informar del éxito o fracaso de una acción:



b) Para motores de búsqueda de la aplicación:



c) Para ayuda en línea de la aplicación:

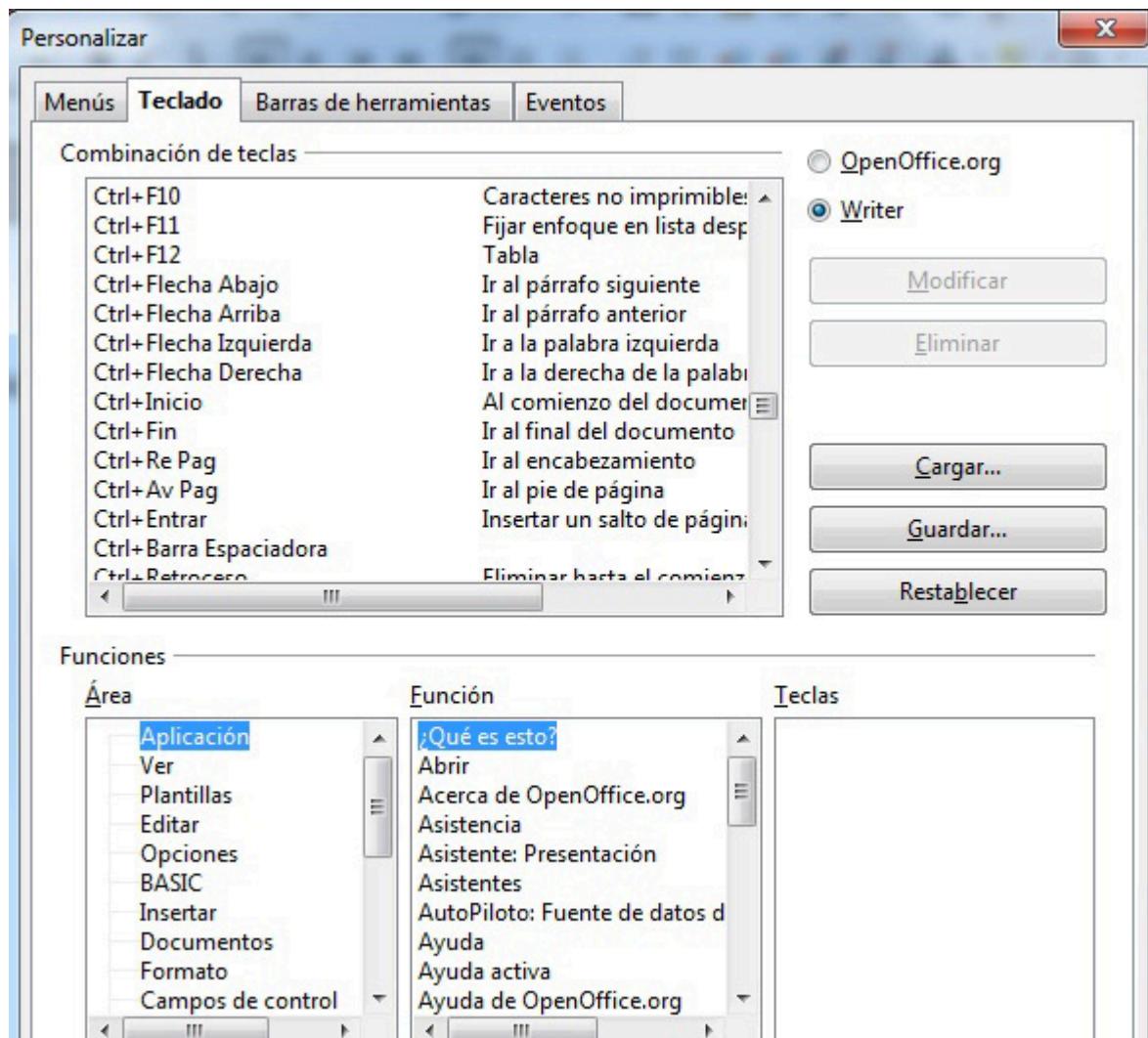


## 6.1.4.- Atajos de teclado.

Los atajos de teclado son combinaciones de teclas que realizan una acción determinada, sustituyendo al uso del puntero del ratón sobre el elemento gráfico de la interfaz.

Es conveniente definir siempre atajos de teclado para aumentar la velocidad de trabajo del usuario en aquellas acciones que realiza con más frecuencia.

Por ejemplo, cuando queremos copiar un bloque de texto, siempre es más rápido pulsar Ctrl+c que buscar con el puntero del ratón sobre el menú correspondiente.



## 6.2.- Aspecto de la interfaz de usuario.

El aspecto que toma la interfaz tiene que ver con los colores utilizados en el diseño, el tipo de fuente empleada para el contenido textual, los iconos que representarán acciones diversas y la distribución de todos estos elementos en el espacio de la interfaz.

### 6.2.1.- Colores.



Francisco Javier Cabrerizo (Elaboración propia)

El color es un elemento de gran importancia en la determinación del éxito de la interfaz. Utilizado adecuadamente, constituye una **potente herramienta de comunicación**.

Su uso inadecuado puede provocar molestias visuales o asociaciones erróneas que impidan la correcta percepción del mensaje.

Otra función es **resaltar información que se considera más importante** para que el usuario pueda identificar más rápidamente los datos más relevantes (que serán a los que antes se les preste atención).

Nuestro objetivo en el manejo de colores en la interfaz es agrupar los diferentes elementos de la pantalla en conjuntos que quedarán asociados espacialmente, distanciados de los demás, haciendo más comprensible el espacio y disminuyendo errores de interpretación.

#### **Ventajas del uso de colores:**

- Puede comunicar ideas de forma más rápida y estética al usuario, aumentando la eficiencia del sistema.
- Hacen que ciertos mensajes se presenten de forma más llamativa al usuario, aumentando su interés en leerlos.
- Crea un efecto psicológico en el ser humano, ya que llama su atención, se memoriza visualmente y crea emociones, persuadiendo al observador.

#### **Precauciones:**

- No hay que abusar del color.
- Cuando hay demasiados colores compitiendo entre sí para captar la atención del usuario se produce confusión.

Los principios básicos, en la utilización de colores en las interfaces de usuario usables son:

- 1-. Limitar el número de colores utilizados: se recomienda no más de 4 o 5 colores diferentes en una misma ventana y no más de 7 en la interfaz total de la aplicación.
- 2-. Utilizar un cambio de color para mostrar un cambio en el estado del sistema.
- 3-. Utilizar el código de colores para apoyar la tarea que los usuarios están tratando de llevar a cabo: se puede utilizar un color para resaltar una situación anómala y otro para similitudes.
- 4-. Utiliza el código de colores en una forma consciente y consistente: si usamos el rojo para mostrar una alarma, mantendremos esta premisa para toda la aplicación.
- 5-. Ser cuidados al utilizar pares de colores: el ojo humano no puede enfocar el rojo y el azul simultáneamente.
- 6-. Se debe de utilizar en todo el espacio de la interfaz un color consistente, dando la apariencia de homogeneidad.

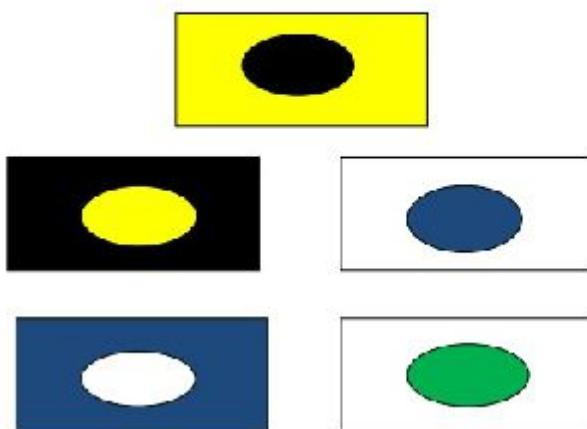
7-. Seleccionar combinaciones de colores más legibles entre el fondo y lo que presentamos en el frente:



Combinación de colores legible

Combinación de colores poco legible

### Combinaciones de color



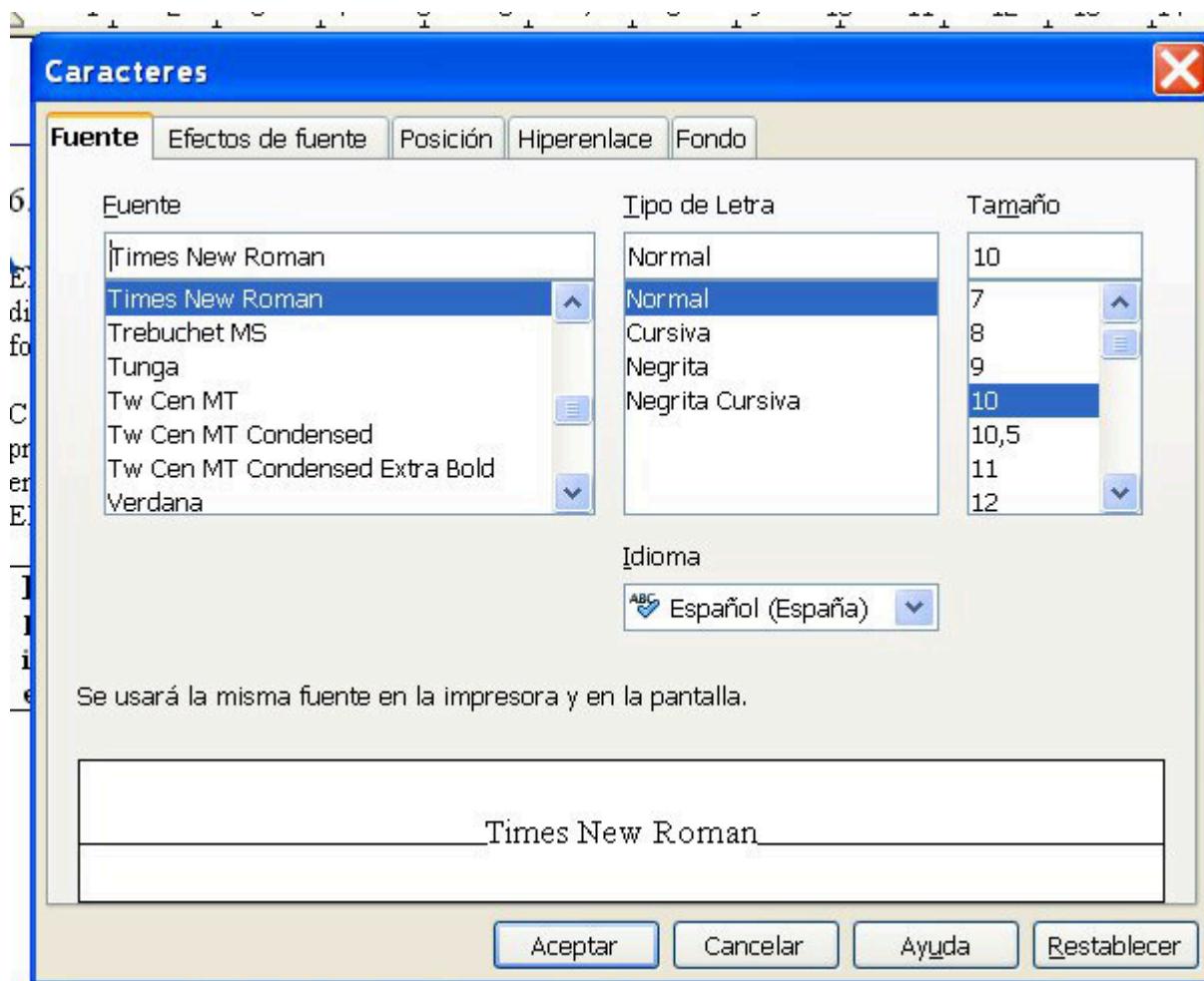
Estas son las cinco combinaciones de color que resultan más legibles entre frente y fondo de una interfaz usable.

## 6.2.2.- Fuentes.

Elegir adecuadamente la tipografía de la interfaz gráfica contribuirá al equilibrio y la interacción entre los distintos caracteres de la pantalla, así como de una igualdad verbal y visual que ayude al lector a entender la forma que adoptan los contenidos de la misma.

Con una buena selección de las fuentes se puede establecer una jerarquía visual que facilite la lectura al proporcionar tanto interrupciones visuales como énfasis gráficos que ayuden al lector a entender la relación entre el texto y la imagen, entre los titulares y los bloques de texto subordinados.

El tamaño de las fuentes debe ser adecuado: legible y proporcional a la resolución de la pantalla.



### 6.2.3.- Iconos.

Los iconos son imágenes que representan acciones o conceptos. Su ventaja fundamental es que se reconocen más rápido que las palabras y ocupan menos espacio en pantalla.

El equilibrio y organización general de los iconos es fundamental en la creación de interfaces gráficas. Una pantalla con sólo texto provoca rechazo: sólo encontramos una pantalla monótona sin saber cómo se estructura la información.

Sin embargo, una página con demasiados elementos gráficos distraerá al usuario.

Por ello, se debe encontrar el equilibrio entre la necesidad de atraer al usuario mediante el contraste visual y la necesidad de ofrecer una clara organización.



## **6.2.4.- Distribución de elementos.**

La organización espacial de los elementos cobra su sentido porque puede dirigir las intenciones del usuario y hacer que la interacción sea mucho más amena y eficaz.

Usar imágenes superpuestas, bloques de contenidos dispuestos de forma heterogénea y encabezados difusos dificultan al lector a comprender la estructura en que se organiza la información.

Es por ello que debemos prestar atención a la distribución de elementos. Una forma muy útil de hacerlo es empleando retículas base y estilos predefinidos para los elementos gráficos del espacio.

Esto ayudará a diseñar una interfaz ordenada y agradable al determinar claramente el espacio de texto y gráficos y el espacio en blanco.

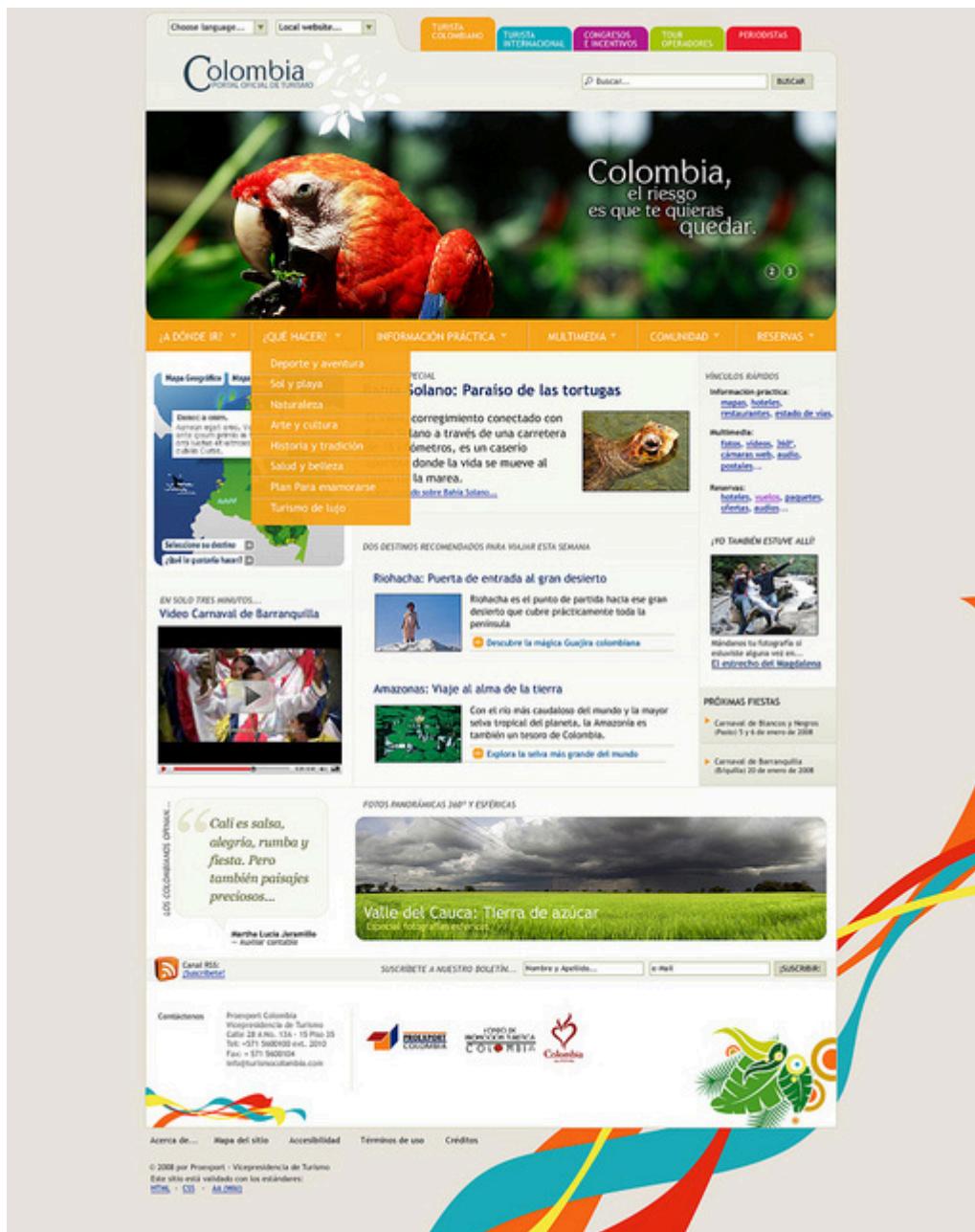
## **6.3.- Elementos interactivos de la interfaz de usuario.**

En el diseño interactivo de interfaces, todos los elementos que fomenten la realimentación son fundamentales, ya que el usuario debe estar informado en cada momento de las acciones que realiza. Si el sistema tarda más tiempo de lo normal en procesar un comando se debe mostrar un mensaje al usuario informándole del proceso que está teniendo lugar para no provocarle incertidumbre. Asimismo, es necesario informar de los errores, efectos y confirmación de acciones que se lleven a cabo en cada momento.

La realimentación debe ser fácil de leer y entender. Para ello, se debe fomentar su consistencia y el almacenamiento de información adicional para asegurar el feed-back.

**Para el diseño de elementos interactivos en interfaces, hay que tener en cuenta el tipo de acciones que se van a llevar a cabo y cuál es la interacción más apropiada de cada**

una con el usuario.



En este caso, observamos una interfaz que carece por completo de elementos interactivos, siendo su finalidad puramente informativa (página web estática).

## 6.3.1.- Botones de comando.

Un botón es un objeto de control sobre la interfaz que nos permite introducir un dato de confirmación al sistema.

En las interfaces de usuario se suelen utilizar distintos tipos de botones:

**Materiales formativos de FP Online propiedad del Ministerio de Educación y Formación Profesional.**

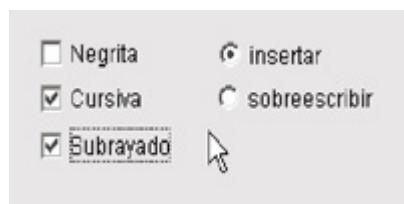
- Botones en relieve.
- Botones de confirmación.
- Botones de forma de radio.

Se trata de elementos de las interfaces gráficas que nos permiten escoger una opción entre varias alternativas. Según la acción que represente, podemos encontrar botones mutuamente excluyentes (sólo se puede seleccionar uno del grupo) o tener la posibilidad de seleccionar varios botones a la vez. Un botón no seleccionado permanecerá en blanco y el seleccionado, relleno.

Al lado de cada botón usualmente aparece una etiqueta que describe la opción que representa.

Los botones de comando se caracterizan por ser muy intuitivos. Los usuarios saben cómo usarlos. Deberían limitarse a una opción (editar, guardar, salir, etc.). Son botones de acción.

Un factor de diseño importante en los botones de comando es su ubicación en la interfaz. Es interesante que su selección sea lo más rápida posible.



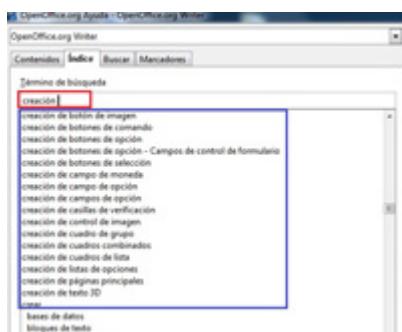
## 6.3.2.- Listas desplegables.

Son controles para manejar una lista de posibles elecciones.

El usuario puede empezar a escribir en un campo de texto mientras se va mostrando una lista de opciones, que dependen directamente del texto que está escribiendo el usuario. Estas opciones pueden ser textos que el usuario escribió con anterioridad o autorelleno del contenido que se escribe, a modo de sugerencia.

Las listas desplegables son una combinación de cuadros de texto y menús desplegables y un claro ejemplo de las mismas es la barra de direcciones de un navegador web o ventanas de ayuda de las aplicaciones, como podemos ver en la imagen. En ella, empezamos a escribir en el campo de texto la palabra "creación" y, automáticamente, nos aparece una lista donde podemos seleccionar la opción deseada.

El punto clave a tener en cuenta es que el usuario debe navegar la lista para encontrar lo que desea. Además, las opciones disponibles deben ser limitadas.



### 6.3.3.- Etiquetas.

Al hablar sobre los botones de comando, hemos mencionado las etiquetas.

Las etiquetas (label) son textos estáticos utilizados en las interfaces gráficas de usuario que generalmente se usan para identificar otros elementos.

Son, por tanto, elementos de información entre el usuario y la interfaz que sirven para describir los demás controles y aumentar la legibilidad y usabilidad de la aplicación.

Son imprescindibles, por ejemplo, para todos los campos de los formularios, ampliamente utilizados y aceptados en el campo de diseño de interfaces.

## 6.4.- Presentación de datos.

El espacio de lectura de la interfaz es la pantalla, por tanto, será fundamental determinar cómo se van a presentar los datos al usuario para conseguir la mayor eficacia y eficiencia en nuestra interfaz; esto es, para que la interfaz tenga el mayor grado posible de usabilidad.

El significado de una imagen puede ser más fácilmente percibido por el usuario si posee claridad visual. La claridad visual afecta a la impresión general de la interfaz. Al reforzarla, promovemos las relaciones lógicas entre los distintos elementos (un ejemplo es que minimizamos el movimiento ocular para obtener información de la interfaz).

La información que se presenta en pantalla el usuario no la lee de forma necesariamente secuencial, sino que va navegando por la interfaz, explorando la información a través de los menús, los enlaces asociativos, recorriendo ciertas rutas predefinidas o saltando de un bloque a otro por medio de mapas de navegación, o bien yendo directamente a la información que le interesa a través de los buscadores u otras herramientas de ayuda y navegación.

**La usabilidad en la presentación de datos de la interfaz es la medida en que los elementos pueden ser utilizados para explotar al máximo todas las posibilidades de la aplicación.**

Algunas reglas de diseño para una efectiva presentación de los datos en la interfaz son:

- Enrejillado. Consiste en organizar el espacio por áreas separadas según función.
- Balanceado. Consiste en buscar el equilibrio en la presentación de los datos entre los ejes horizontal y vertical de la interfaz.
- Simetría. Consiste en duplicar la imagen a lo largo de alguno de los ejes. Con ella aseguramos que el diseño está balanceado.

Algunos consejos para diseñar la presentación de datos en pantalla son:

Ser conciso en la información: al usuario le desagrada leer trozos extensos de texto.

Permitir una fácil navegación por el sitio: el usuario suele explorar globalmente toda la interfaz antes de decidir si le interesa su contenido o no.

Diseñar los contenidos por una estructura piramidal: las ideas más importantes deben ir al principio de los mensajes.

Lenguaje: el lenguaje empleado debe ser cercano al usuario.

Presentación separada de información: los mensajes deben aparecer de manera aislada, no siendo recomendable transmitir varios mensajes en la misma ventana o el mismo párrafo.

## **6.5.- Diseño de la secuencia de control de la aplicación.**

Para decidir cómo va a funcionar realmente la aplicación y qué acciones van a estar asociadas a cada uno de los elementos de la interfaz, tenemos que seguir una serie de pasos.

El diseño de la secuencia de control de la aplicación se puede resumir en los siguientes puntos:

1. Se crea una interfaz dibujando los controles y objetos sobre un formulario. Para que su código sea más fácil de leer le ponemos un nombre.
2. Después de añadir los objetos al formulario, establecemos las propiedades de los objetos.
3. Escribimos el código que se va a ejecutar en respuesta a los eventos que se produzcan cuando se active un control u objeto determinado.
4. Guardamos el proyecto y generamos el ejecutable. Por último, creamos el instalador.

**En las aplicaciones dirigidas por eventos la ejecución no sigue una ruta prefijada. Según las acciones que invoque el usuario, se llevarán a cabo los eventos correspondientes.**

Es recomendable seguir las siguientes consideraciones en cuanto al diseño:

- Establecer los objetivos de cada componente de la interfaz y sus intenciones.
- Hacer corresponder cada objetivo con una secuencia de interacción.
- Especificar la secuencia de acciones de las tareas.
- Indicar el estado del sistema.
- Definir mecanismos de control.
- Mostrar la manera de cómo los mecanismos de control afectan al estado del sistema.
- Indicar la forma en que los usuarios interpretan el estado del sistema a partir de la información de la interfaz.

Concluyendo, se dibuja los controles y objetos sobre un formulario en la interfaz. Posteriormente, se establecen las propiedades de esos controles y objetos; es decir, el código correspondiente a la acción o acciones que se llevarán a cabo cuando el objeto sea invocado por el usuario. Probar y depurar la aplicación, crear un archivo ejecutable y distribuir la aplicación.

## **6.6.- Aseguramiento de la información.**

El aseguramiento de la información es la base sobre la que se construye la toma de decisiones de una organización.

**Sin aseguramiento, no tenemos la certeza de que la información que manejamos en la interfaz sea segura, confiable y esté disponible cuando se necesite.**

Se define aseguramiento de la información como la utilización de información y de diferentes actividades operativas, con el fin de proteger la información, los sistemas de información y las redes de forma que preservemos la integridad, disponibilidad, autenticación y confidencialidad ante el riesgo de amenazas ya sean locales o remotas (a través de una red de comunicaciones).

El aseguramiento de la información que se maneja en una aplicación informática debe ser tenido en cuenta en cuatro ámbitos diferentes:

1. Aseguramiento de Datos.
2. Aseguramiento de Procesos.
3. Aseguramiento de Comportamiento.
4. Aseguramiento del Sistema de Gestión de la empresa.

La plataforma de programación que escojamos para desarrollar interfaces debe ofrecer herramientas que garanticen la seguridad en la información que se maneja. Por ejemplo, la plataforma JDK ofrece las siguientes características:

- Control de acceso: El JDK introduce el llamado "applet firmado digitalmente", con total acceso a los recursos.
- Políticas de seguridad: definen un conjunto de permisos disponibles que pueden ser configurados por el administrador del sistema. Cada permiso especifica un acceso permitido a un recurso concreto.
- Servicios de criptografía.
- Manejo de certificados y claves.

## 6.7.- Aplicaciones multimedia.

Las aplicaciones multimedia son, como ya sabrás, aquellas que además de texto, pueden presentar elementos como sonido, imágenes, animaciones, vídeos, etc.

Multimedia mejora las interfaces tradicionales basadas sólo en texto y proporcionan beneficios importantes que atraen al usuario y mantienen su atención.

**El uso de elementos multimedia en las interfaces mejora la retención de la información que se presenta.**

En el diseño de contenidos multimedia se debe mantener un equilibrio entre los contenidos que aprovechan las posibilidades de hipertexto y multimedia y los que no. En caso contrario, el contenido de la interfaz sería caótico debido a un exceso de elementos multimedia.

En definitiva, de lo que se trata es de diseñar contenidos relacionados y vinculados que faciliten el uso de la aplicación manteniendo siempre una coherencia visual y comunicativa.

Los elementos multimedia utilizados en el diseño de interfaces gráficas de usuario son:

- **Gráficos:** Representaciones visuales que mantienen relación con el concepto al que representa. Puede ser bidimensional o tridimensional y con su uso conseguimos facilitar la expresión y comprensión de ideas.
- **Imágenes:** Representaciones visuales estáticas, digitales, ampliamente utilizadas para transmitir información.
- **Animaciones:** Presentaciones muy rápidas de una secuencia de imágenes que dan al observador una sensación de movimiento. Aportan a la interfaz veracidad y expresividad.
- **Vídeos:** Se utilizan cuando es primordial mostrar los atributos de algún concepto. Se diseñan de forma que el observador puede interrumpir, reiniciar y parar la secuencia tantas veces como desee.
- **Audio:** Mensajes de naturaleza acústica que aportan sonoridad a la interfaz.