

TEMA 54

DISEÑO DE INTERFACES DE USUARIO. CRITERIOS DE DISEÑO. DESCRIPCIÓN DE INTERFACES. DOCUMENTACIÓN. HERRAMIENTAS PARA LA CONSTRUCCIÓN DE INTERFACES.

ÍNDICE

1. INTRODUCCIÓN
2. FACTORES QUE CONDICIONAN EL DISEÑO
 - 2.1. Tipos de usuarios
 - 2.2. Áreas de trabajo
 - 2.3. Factores Humanos
3. DIRECTRICES PARA EL DISEÑO
 - 3.1. Interacción general
 - 3.2. Visualización de la información
 - 3.3. Entrada de datos
4. ESTÁNDARES DE INTERFAZ
5. HERRAMIENTAS PARA EL DISEÑO DE INTERFACES
6. BIBLIOGRAFÍA

1. INTRODUCCIÓN

Para muchos usuarios de sistemas de ordenador, la frustración y la ansiedad forman parte de la vida diaria. Se esfuerzan por aprender un lenguaje de órdenes o un sistema de selección de menús que, se supone, les ayudará en su trabajo. Algunas personas sufren casos tan serios de shock con el ordenador, de terror al terminal o de neurosis de red, que evitan utilizar sistemas de ordenador.

Los problemas a los que se alude son reales. Todos hemos encontrado "interfaces" difíciles de aprender, complicadas de utilizar, confusas, imperdonables y, en muchos casos, totalmente frustrantes. Aun así, alguien dedicó tiempo y energía construyendo cada una de esas interfaces y es casi seguro que el constructor no creó esos problemas intencionadamente.

En este tema se considera el diseño de la interfaz de usuario; un tema que se ha hecho cada vez más importante, a medida que ha aumentado el uso de los ordenadores. Encontramos interfaces "inteligentes" cuando utilizamos una fotocopidora, un horno de microondas, un procesador de textos o un sistema de diseño asistido por ordenador. Desde el punto de vista del usuario, lo que permite a un piloto pilotar un avión moderno, a un radiólogo interpretar la salida de un "scanner" CAT o a un banquero transferir millones de dólares a través de continentes, es la interfaz. Esta es, en muchos sentidos, el "envoltorio" del software de ordenador. Si es fácil de aprender, simple de utilizar, directo y no muy estricta, el usuario podrá hacer buen uso de lo que hay dentro. Si, por el contrario, no tiene ninguna de estas características, aparecerán problemas invariablemente.

El diseño está relacionado con las partes internas del software. El "diseño" considerado de esa manera está, en gran parte, oculto al usuario final aunque es muy importante para la calidad global del software. El diseño de la interfaz es diferente. Si es muy bueno, el usuario se adaptará a un ritmo normal de interacción. El o ella pueden incluso olvidar que se está llevando a cabo una comunicación con la máquina. Pero si no lo es, el usuario lo sabrá inmediatamente y se sentirá a disgusto con un modo de interacción poco amigable.

El diseño de la interfaz de usuario tiene tanto que ver con el estudio de la gente como con aspectos de la tecnología. ¿Quién es el usuario? ¿Cómo aprende el usuario a interaccionar con un sistema nuevo basado en ordenador? ¿Qué espera el usuario del sistema? Estas son sólo unas pocas de las muchas preguntas que deben ser planteadas y respondidas como parte del diseño de la interfaz de usuario.

2. FACTORES QUE CONDICIONAN EL DISEÑO

Antes de dar normas de diseño hay que tener presentes qué factores afectan, y a veces condicionan, al diseño:

- Tipos de usuarios.
- Áreas de trabajo del sistema.
- Factores humanos medibles.

2.1. Tipos de usuarios

Aunque la categorización de usuarios podría ampliarse tanto como usuarios posibles de nuestro sistema existan, los agrupamos en tres categorías principales atendiendo al nivel de conocimiento de la ordenador y de la tarea. **Grupos de usuarios:**

- Novatos.
- Intermitentes.
- Frecuentes.

Novatos

No tienen conocimientos sintácticos y pocos semánticos sobre la tarea a desempeñar.

Llegan a la máquina con ansiedad.

Intentan relacionar su conocimiento con los objetos de la tarea. Los conceptos (semántica) y la sintaxis son una sobrecarga.

Qué hacer en el diseño de nuestro sistema:

- Restringir vocabulario.
- Pequeño número de opciones y posibilidades.
- Conducir las tareas.
- Evitar respuestas rápidas por parte del sistema.
- Retroalimentación constante.
- Mensajes de error constructivos.
- Amplia documentación en papel.
- Tutoriales en línea.

Intermitentes

Pueden mantener conocimientos semánticos de la tarea y ordenador, pero olvidan los sintácticos.

Qué hacer en el diseño de las interfaces de nuestro sistema:

- Importante la consistencia para facilitar el reconocimiento (es mejor reconocer que recordar).
- Protección contra daños causados por una exploración no metódica.
- Ayuda en línea para recordar sintaxis.
- Manuales de referencia organizados por contenidos, nunca organizados sintácticamente.

Frecuentes

Dominan la semántica y la sintaxis.

Quieren aprender rápidamente la tarea.

Qué hacer para el diseño de las interfaces:

- Tiempos de respuesta breves.
- Poca realimentación.
- Posibilidad de realizar las acciones con pocos pasos.
- Incluir un lenguaje de macros para la realización de tareas comunes y frecuentes.

Conclusiones:

- Ningún diseño simple puede satisfacer a todos los usuarios y situaciones.
- Antes de comenzar el diseño hay que caracterizar a usuarios y situaciones.
- Nadie está en contra de nuestro diseño, pero suponen que ya lo conocen.
- Sin embargo, la caracterización nunca acaba, por la magnitud en cuanto al tipo de usuarios y porque estos usuarios cambian.

Si el usuario tiene experiencia preferirá un grado de libertad y control mayor sobre los procesos, mientras que si el usuario es novato u ocasional deseará una interfaz simple y que proceda según una estructura totalmente dirigida.

En general la solución más cercana a la realidad es la que incluya elementos en las interfaces que sirvan para los tres tipos de usuarios. Ej.: Para acceder a una tarea se podrá “navegar” por una serie de menús desplegables o se podrá ir directamente pulsando una combinación de teclas preestablecida.

2.2. Áreas de Trabajo

Aunque sucede lo mismo que con los usuarios (que se pueden establecer calificaciones muy amplias) hemos agrupado todas las áreas de trabajo en cuatro grandes bloques:

- **Sistemas críticos para la vida humana.**
- **Industria y comercio.**
- **Oficina, hogar, entretenimiento.**
- **Sistemas exploratorios, creativos y colaborativos.**

Sistemas críticos

Control de Tráfico Aéreo, bomberos, unidades de vigilancia intensiva, etc. Se esperan altos costes, pero deben tener una efectividad y confiabilidad elevadas. Es aceptable emplear períodos largos de entrenamiento para lograr una ejecución del sistema rápida y libre de errores. La satisfacción subjetiva de los usuarios no es tan importante ya que están muy motivados. Y por último, se consigue una retención de los conocimientos a causa del uso frecuente del sistema. Se tienen que incluir en el sistema prácticas de simulación para situaciones de emergencia.

Industria y comercio

Inventarios, reservas, banca, etc. Los costes se tienen muy en cuenta para las decisiones de diseño. A menudo se prefieren costes bajos aunque esto implique menor confiabilidad y prestaciones. El entrenamiento, que siempre es caro (por necesidad de tiempo y recursos), debe estar implícito en el uso frecuente del sistema (nuestro sistema ha de ser fácil de aprender). Los costes también influyen en la resolución de los compromisos entre velocidad de ejecución (objetivo principal) y las cantidades de errores que pueden cometer los operadores. La satisfacción subjetiva incide de forma modesta (ya que son prioritarias las directrices de empresa), mientras que la retención se consigue con el uso.

Oficinas, hogares y entretenimiento

Procesadores de texto, juegos, paquetes educativos, gestión de pequeños negocios, etc. Aquí lo más importante es la facilidad de aprendizaje, la poca posibilidad de errores del usuario y la satisfacción subjetiva (ha de gustar nuestro producto para ser vendido). La competencia en esta área es muy grande por lo que es importante obtener un coste pequeño.

Sistemas exploratorios

Enciclopedias electrónicas, ayudas para la toma de decisiones, escritura de trabajos en colaboración, diseño de automóviles, composición de música, etc. En estos sistemas el usuario es experto en su área de trabajo, pero es novel en los conceptos informáticos. Tienen motivaciones altas y expectativas elevadas, a veces demasiado elevadas. El tipo de usuario puede ser ocasional o frecuente, por tanto es difícil diseñar y evaluar estos sistemas.

2.3. Factores Humanos

Cuando consideramos un sistema interactivo basado en software, la frase "factores humanos" toma una gran variedad de significados. En un nivel fundamental, debemos comprender la percepción visual, la psicología cognitiva de la lectura, la memoria humana y el razonamiento deductivo e inductivo. En otro nivel, debemos comprender al usuario y su comportamiento. Finalmente, debemos comprender las tareas que el sistema basado en software realiza para el usuario y las tareas que pide el usuario como parte de la interacción entre el hombre y la máquina.

La interfaz de usuario es el mecanismo a través del cual se establece un diálogo entre el programa y el humano. Si se han tenido en cuenta los factores humanos, el diálogo será fluido y se establecerá un ritmo entre el usuario y el programa. Si estos factores se han ignorado, el sistema será casi siempre visto como "poco amigable".

Fundamentos de la percepción humana

Un ser humano percibe el mundo a través de un sistema sensorial que comprendemos razonablemente bien. Cuando se considera una **interfaz hombre-máquina (IHM)**, predominan los sistemas visual, táctil y auditivo. Esto permite al usuario de un sistema basado en ordenador percibir información, almacenarla en la memoria (humana) y procesarla utilizando un razonamiento inductivo o deductivo.

La neurofisiología de la percepción sensorial está más allá del ámbito de este tema. En esta sección se presenta una breve revisión con el fin de proporcionar un conocimiento básico sobre factores humanos.

La mayoría de las IHM se llevan a cabo a través de un medio visual (p. ej.: informes impresos o gráficos, CRT o pantallas planas). El ojo y el cerebro trabajan conjuntamente para recibir e interpretar la información visual basándose en el tamaño, la forma, el color, la orientación, el movimiento y otras características. La comunicación visual tiene la cualidad del "paralelismo". Al humano se le presentan simultáneamente muchos elementos de información distintos para que los asimile. La especificación apropiada de la comunicación visual es el elemento clave de una interfaz "amigable".

Aunque existe una tendencia definida hacia la comunicación gráfica en el diseño de IHM, todavía se presenta mucha información visual en forma de texto. La lectura -el proceso de extraer información de un texto- es una actividad muy importante en muchas interfaces. El humano debe decodificar patrones visuales para extraer el significado de las palabras y las frases. La velocidad de este proceso se controla mediante un patrón de movimiento del ojo que barre el texto utilizando desplazamientos de alta velocidad, denominados sacádicos. El tamaño del texto, el tipo de letra, la longitud de la línea, las mayúsculas, el sitio y el color influyen en la facilidad con la que se realiza la extracción de información.

La información, según se extrae de la interfaz, debe ser almacenada para ser usada posteriormente. Además, el usuario puede tener que recordar órdenes, secuencias de operación, alternativas, situaciones de error y otra información tediosa. Toda esta información debe ser almacenada en la memoria humana; un sistema extremadamente complejo del cual se cree actualmente que está compuesto por una memoria a corto plazo (MCP) y una memoria a largo plazo (MLP). La información sensorial (visual, auditiva y táctil) adquirida se mantiene en un "buffer" para ser posteriormente almacenada en la MCP desde donde puede ser reutilizada en cualquier momento. El tamaño del buffer y la cantidad de tiempo durante la que esta información se puede reutilizar es limitado. El conocimiento se mantiene en la MLP y constituye la base para nuestras respuestas cuando utilizamos una IHM. La información semántica y sintáctica (conocimiento) es almacenada en la MLP. Si el ingeniero de sistemas especifica una interfaz hombre-máquina con exigencias desproporcionadas sobre la MCP y/o la MLP, el rendimiento del elemento humano sobre el sistema se verá degradado.

La mayoría de la gente no aplica un razonamiento inductivo o deductivo formal cuando se enfrenta a un problema. En lugar de esto, aplicamos un conjunto de heurísticas (consejos, reglas y estrategias) basadas en nuestra comprensión de problemas similares. De hecho, las heurísticas que utilizamos tienden a ser específicas de un sector. Esto es, un problema idéntico, encontrado en contextos completamente diferentes puede ser resuelto aplicando diferentes heurísticas. Una IHM debe especificarse de tal manera que permita al humano desarrollar heurísticas para la interacción. En general, estas heurísticas deben ser consistentes en diferentes ámbitos de interacción.

Nivel de habilidad humana y comportamiento

Además de los elementos básicos de la percepción humana, es importante tener en cuenta las diferencias individuales de habilidad, personalidad y comportamiento entre usuarios de un sistema basado en ordenador. Una interfaz completamente aceptable para un ingeniero puede ser inadecuada para un trabajador sin preparación. Una interfaz utilizada por dos individuos con la misma educación y

preparación, pero con personalidades completamente diferentes puede ser vista como "amistosa" por uno y como "poco amigable" por el otro.

El nivel de habilidad del usuario final tendrá un impacto significativo sobre su habilidad para extraer información significativa de la IHM, para responder eficientemente a tareas exigidas por la interacción y para aplicar de forma efectiva heurísticas que crean un ritmo de interacción. En la mayoría de los casos, más importante que la cultura general o la inteligencia es el conocimiento específico del terreno o del contexto. Por ejemplo, un mecánico que utilice un sistema de diagnóstico de automóvil basado en ordenador, comprende el ámbito del problema y puede interactuar efectivamente mediante una interfaz diseñada específicamente para usuarios con una formación en mecánica. Esta misma interfaz podría confundir a un físico; aún cuando un físico tiene una educación formal considerablemente mayor y se siente, de hecho, más cómodo con las ordenadores.

Cada usuario de ordenador tiene una personalidad única. En la mayoría de los casos, la personalidad de un individuo está fuertemente ligada a su estilo cognitivo. De esta forma, la IHM debería ser diseñada para adaptarse a las diferencias de personalidad o, alternativamente, para adaptarse a una personalidad "típica" de entre una clase de usuarios finales. Shneiderman en su libro "Diseño de Interfaces de Usuario" propone las siguientes escalas psicológicas: acepta riesgos/evita riesgos, de control interno/externo, reflexivo/impulsivo, convergente/divergente, ansiedad alta/baja, tolerancia al estrés alta/baja, tolerancia a la ambigüedad alta/baja, dependencia/ independencia del campo de especialización, activo/pasivo, motivación alta/baja, compulsividad alta/baja, orientación del cerebro a la izquierda/derecha. Existen pocos datos empíricos que puedan ayudar a los diseñadores de IHM a crear una interfaz que se adapte a un tipo de personalidad específica. Sin embargo, no se cuestiona que ciertas interfaces serán aceptadas antes por usuarios con un tipo de personalidad determinada que por usuarios con un tipo diferente.

La proliferación de sistemas basados en ordenador ha desencadenado un fenómeno denominado "tecnomiedo" -un miedo irracional hacia sistemas y productos de alta tecnología-. Es muy probable que el nivel de tecnomiedo se reduzca en gran medida según vayamos ampliando nuestro conocimiento del elemento humano en sistemas basados en ordenador y se especifiquen IHM que se adapten mejor a las necesidades humanas.

Tareas y factores humanos

Un sistema interactivo basado en ordenador raramente permite a un usuario hacer algo enteramente nuevo. En la mayoría de los casos, el sistema se construye para automatizar (y por lo tanto mejorar) ciertas tareas que se realizaban antes a mano o utilizando algún otro procedimiento. Idealmente, la nueva tecnología permite a un usuario realizar tareas mejor, más deprisa, más eficientemente, con más precisión y a un coste menor. Pero las tareas subyacentes siguen siendo las mismas y una IHM debe dotar al usuario final de un entorno fácil y natural para realizarlas.

Aunque para cada aplicación las tareas son diferentes, es posible una categorización global. Tanto si se hacen las cosas "a la antigua usanza", como si se utiliza un procedimiento nuevo que use un sistema interactivo basado en computador, casi siempre se realizan las siguientes *tareas genéricas*:

Tareas de comunicación. Actividades que permiten que la información sea transferida desde el productor hacia el consumidor.

Tareas de diálogo. Actividades que permiten al usuario dirigir y controlar la interacción con el sistema basado en ordenador.

Tareas cognitivas. Actividades que se realizan una vez que se ha obtenido la información; actividades asociadas con el funcionamiento del sistema.

Tareas de control. Actividades que permiten al usuario controlar la información y el conocimiento y ordenar el proceso a través del cual se desarrolla el resto de las tareas.

Para desarrollar instancias específicas de estas tareas generales, se utiliza una técnica de diseño de interfaces de usuario llamada **análisis y modelización de tareas**. Esta técnica se explicará en una sección posterior.

3. DIRECTRICES PARA EL DISEÑO

El diseño de interfaces de usuario recae fuertemente en la experiencia del diseñador y en la experiencia anecdótica que aparece en cientos de artículos técnicos y docenas de libros. Muchas referencias bibliográficas (p. ej.: "Diseño de interfaces de usuario para Software" de J.S. Dumas) presentan un conjunto de directrices para el diseño de IHM que conducen al desarrollo de una interfaz eficiente y "amigable". En esta sección se presentan algunas de las directrices más importantes.

Se sugieren tres *categorías de directrices para el diseño de IHM*:

- Interacción general,
- Visualización de información y
- Entrada de datos.

3.1. Interacción General

Las directrices para la interacción general sobrepasan, a menudo, las fronteras que la separan de la visualización de información, la entrada de datos y el control global del sistema. Son, por lo tanto, complementarias y se corre un gran riesgo ignorándolas. Las siguientes directrices se centran en la interacción general:

- Ser consistente. Se debe utilizar un formato consistente para la selección de menús, la entrada de órdenes, la visualización de datos y cientos de otras funciones que incorpora una IHM.
- Ofrecer una realimentación significativa. Se debe proporcionar al usuario una realimentación visual y auditiva para asegurar que se establece una comunicación bidireccional (entre el usuario y la interfaz).
- Preguntar por la verificación de cualquier acción destructiva no trivial. Si un usuario pide que se borre un archivo, indica que se escriba sobre información ya existente o desea concluir la ejecución de un programa, debe aparecer un mensaje del tipo "¿Está seguro ... ?".
- Permitir una vuelta atrás fácil en la ejecución de la mayoría de las acciones. Las funciones deshacer o invertir han salvado a decenas de miles de usuarios finales de millones de horas de frustración. La vuelta atrás debe estar disponible en todas las aplicaciones interactivas.
- Reducir la cantidad de información que debe ser memorizada entre acciones. No se debe esperar que el usuario recuerde una lista de números o nombres para que pueda reutilizarlos como entrada a la siguiente función. La carga de memorización debe ser minimizada.
- Buscar la eficiencia en el diálogo, el movimiento y el pensamiento. El número de pulsaciones debe ser minimizado, la distancia que un ratón debe recorrer entre dos pulsaciones también debe ser tenida en cuenta al diseñar el formato de presentación, y el usuario debe encontrar pocas veces una situación en la que tenga que preguntar, "¿Qué significa esto?".
- Perdonar los errores. El sistema debe protegerse de los errores del usuario que pudiesen afectarle causándole un fallo.
- Categorizar las actividades en base a su función y organizar la geografía de la pantalla convenientemente. Uno de los beneficios claves del menú desplegable es su habilidad para organizar las órdenes de acuerdo a su tipo. En esencia, el diseñador debe hacer hincapié en la ubicación coherente de órdenes y acciones.
- Proporcionar facilidades de ayuda sensibles al contexto.

- Utilizar verbos de acción simples o frases verbales cortas para nombrar las órdenes. Un nombre de orden largo es más difícil de reconocer y de recordar. También ocupa un espacio innecesario en las listas de menús.

3.2. Visualización de la Información

Si la información que presenta la IHM es incompleta, ambigua o ininteligible, la aplicación no satisfará las necesidades del usuario. La información puede ser presentada de formas diferentes: con texto, dibujos y sonidos; por posición, movimiento y tamaño; utilizando colores, resolución; e incluso por omisión. Las siguientes directrices se centran en la visualización de la información:

- Mostrar sólo aquella información que sea relevante en el contexto actual. El usuario no debe tener que buscar a través de datos extraños, menús y gráficos para obtener información relevante a una función concreta del sistema. No abrumar al usuario con datos -utilizar un formato de presentación que permita una asimilación rápida de la información-. Los gráficos o esquemas deben reemplazar a las tablas de datos.
- Utilizar etiquetas consistentes, abreviaciones estándar y colores predecibles. El significado de una pantalla debe ser obvio, evitando referencias a otras fuentes externas de información.
- Permitir al usuario mantener el contexto visual. Si se están reescalando gráficos en una pantalla, debe mantenerse la imagen original (en forma reducida en la esquina de la pantalla) de tal forma que el usuario no pierda la posición relativa de la porción de imagen que está observando.
- Producir mensajes de error significativos.
- Utilizar mayúsculas y minúsculas, tabulaciones y agrupaciones de texto para ayudar a la comprensión. La mayoría de la información visualizada en una IHM es textual, por lo que la presentación y formato del texto tiene una incidencia importante en la facilidad con que la información es asimilada por el usuario.
- Utilizar ventanas (si están disponibles) para modularizar los diferentes tipos de información. Las ventanas mantienen accesibles al usuario muchos tipos diferentes de información.
- Utilizar representaciones "analógicas" para mostrar la información que es más fácil de asimilar bajo este tipo de representación. Por ejemplo, una representación digital del nivel de presión mantenido en un depósito de una refinería de petróleo tendría poco impacto. Sin embargo, si se visualiza mediante un dispositivo de tipo termómetro, se pueden utilizar cambios de color y movimientos verticales para indicar condiciones de presión peligrosas. Esto proporcionaría al usuario información relativa y global simultáneamente.
- Considerar la geografía disponible en la pantalla y utilizarla eficientemente. Cuando se utilizan muchas ventanas, debe haber espacio disponible para mostrar, al menos, una porción de cada una. Además, el tamaño de la pantalla (un aspecto de la ingeniería de sistemas) debe ser seleccionado de acuerdo con el tipo de aplicación que se va a desarrollar.

3.3. Entrada de Datos

El usuario emplea la mayor parte de su tiempo seleccionando órdenes, introduciendo datos y, de una u otra forma, suministrando datos al sistema. En muchas aplicaciones, el teclado sigue siendo el medio de entrada de datos más importante, pero el ratón, los digitalizadores e incluso los sistemas de reconocimiento de voz ya son alternativas eficientes. Las siguientes directrices se centran en la entrada de datos:

- Minimizar el número de acciones de entrada de datos que debe realizar el usuario. Ante todo se debe reducir la cantidad requerida de pulsaciones de teclas. Esto se puede conseguir utilizando el ratón para seleccionar un elemento de entre un conjunto predefinido; utilizando una "escala

deslizante" para seleccionar el valor de un dato dentro de un determinado rango de valores; utilizando "macros" que permitan convertir una pulsación en una secuencia de datos de entrada.

- Mantener la consistencia entre la información visualizada y los datos de entrada. Las características visuales de la pantalla (p. ej.: tamaño del texto, color, situación) deben ser mantenidas a lo largo de todo el campo de adquisición.
- Permitir al usuario personalizar la entrada de datos. Un usuario experto podría decidir crear órdenes personalizadas y respuestas con determinados tipos de mensajes de aviso y de verificación de acciones. La IHM debería permitirlo.
- La interacción también debe ser flexible y estar ajustada al modelo de entrada preferido por el usuario. El modelo de usuario debe guiar la determinación del modo de entrada preferido. Un trabajador con experiencia puede encontrarse muy cómodo con una entrada por teclado, mientras que un director podría sentirse más cómodo utilizando un ratón.
- Desactivar órdenes que sean inapropiadas en el contexto actual. Esto evita que el usuario realice acciones que podrían conducir a error.
- Permitir al usuario controlar el flujo interactivo. El usuario debe poder evitar acciones innecesarias, cambiar el orden de acciones requeridas (cuando sea posible en el contexto de una aplicación) y salir de situaciones de error sin tener que abandonar el programa.
- Proporcionar ayuda en todas las acciones de entrada de datos.
- Eliminar las entradas innecesarias. No se debe obligar al usuario a especificar las unidades en entradas de datos de ingeniería (a menos que pueda existir ambigüedad). No se debe obligar al usuario a escribir ,00 en cantidades enteras de dinero, se deben proporcionar valores por defecto siempre que sea posible y nunca obligar al usuario a introducir información que pueda ser adquirida automáticamente o calculada dentro del programa.

4. ESTÁNDARES DE INTERFAZ

La mayoría de los constructores de aplicaciones modernas de software tienden a desarrollar interfaces con ventanas y dispositivos de señalización y selección. Aun así, la creación de estas interfaces no es una tarea sencilla. Por esta razón, se están adoptando estándares de interfaces de usuario en toda la industria. La estandarización de las interfaces beneficia al constructor y al usuario de una IHM.

El diseñador puede reutilizar módulos y objetos existentes de otra IHM creando, de esta forma, la nueva interfaz más rápidamente y con una mayor calidad. Se pueden utilizar las herramientas SDIU para crear un prototipo de interfaz de usuario en mucho menos tiempo del requerido para desarrollarlo partiendo de cero.

El usuario se familiariza con la presentación y el ritmo de la IHM y, de esta forma, aprende mucho antes cualquier aplicación nueva que utilice la misma interfaz estándar. Después de un tiempo, la utilización de la interfaz se hace intuitiva y mucho más productiva para el usuario final.

Aunque hay en desarrollo varios estándares de interfaz, el que más se sigue es el sistema **X Window**. El sistema X Window define una sintaxis y una semántica para el diseño de IHM y proporciona herramientas para la creación de pantallas, ventanas y gráficos, así como un protocolo para manejar recursos, interacciones con dispositivos y manejo de excepciones. Se han desarrollado varias extensiones y variaciones del sistema X Window estándar que se pueden utilizar sobre PC y estaciones de trabajo, bajo Unix y otros sistemas operativos. Otro estándar de interfaz muy importante es el utilizado por los programas basados en **Microsoft Windows**, que ofrecen un aspecto similar independientemente del tipo de aplicación que se utilice. Además se pueden configurar diversos aspectos de la interfaz: colores, tamaño de las ventanas, etc.

5. HERRAMIENTAS PARA EL DISEÑO DE INTERFACES

El proceso de diseño de la interfaz de usuario es interactivo. Esto es, se crea un modelo de diseño, se implementa como un prototipo -debe tenerse en cuenta que en algunos casos (p. ej. pantallas en cabinas de aeronaves) el primer paso puede ser simular la interfaz en un dispositivo de visualización en lugar de desarrollar un prototipo en hardware-; los usuarios lo examinan (que son los que desarrollan el modelo de usuario descrito antes) y se modifica en base a sus comentarios. Para facilitar este enfoque interactivo de diseño, se ha desarrollado una amplia gama de herramientas de diseño y prototipo. Estas se denominan herramientas para la interfaz de usuario o **sistemas de desarrollo de interfaces de usuario (SDIU)**, y proporcionan módulos u objetos que facilitan la creación de ventanas, menús, interacción con dispositivos, mensajes de error, órdenes y otros muchos elementos de un entorno interactivo.

Al utilizar paquetes de software que pueden ser aprovechados directamente por el diseñador y por el programador, o bien por la interfaz de usuario, un SDIU proporciona mecanismos para:

- Control de los dispositivos de entrada (como ratones y teclados).
- Validación de la entrada de datos del usuario,
- Manejo de errores y visualización de mensajes de error.
- Realimentación (p. ej.: eco automático de la entrada).
- Ayudas e indicaciones.
- Manejo de ventanas y campos, controlando el movimiento del texto dentro de las ventanas.
- Establecimiento de conexiones entre el software de aplicación y la interfaz.
- Aislamiento de la aplicación de las funciones de manejo de la interfaz.

Estas funciones pueden ser implementadas utilizando un sistema gráfico o un sistema basado en un lenguaje de interacción.

Incluso con la evolución de los estándares de interfaces de usuario, el diseño y el desarrollo de interfaces hombre/máquina sigue siendo un reto para los ingenieros de software. Estudios en la industria han mostrado que entre el 50 y el 80 por ciento de todo el código generado para aplicaciones interactivas está dedicado a la gestión y mantenimiento de la interfaz hombre/máquina.

Las herramientas de diseño y desarrollo de interfaces son, en realidad, un conjunto de componentes de software, tales como menús, botones, estructuras de ventanas, iconos, mecanismos de visualización, controladores de dispositivos y otros elementos de este tipo. Sin embargo, estos conjuntos de herramientas están siendo reemplazados por herramientas para desarrollar prototipos que permiten la creación rápida en pantalla de interfaces sofisticadas ajustadas al estándar elegido para el software (p. ej.: X-Windows, de Motif).

6. BIBLIOGRAFÍA

Pressman, Roger S.
Ingeniería del Software
 Mc Graw-Hill, 1993

López, Antonio
Metodologías de Desarrollo
 Ra-ma, 1990