

## **Aspectos lingüísticos y comunicativos del interfaz de usuario de un software basado en la tecnología de la Web**

*Uta Weis*

(Channel One GmbH, Hamburgo, Alemania)

El interfaz de usuario ha ocupado desde siempre una posición marginal dentro del desarrollo de *software*. Este hecho se explica, en primer lugar, por el dominio de la realización técnica. En efecto, por un lado se encuentra el cliente que necesita una aplicación determinada, y, por otro lado, el programador que es responsable de la realización técnica. Entre los dos hay a menudo una distancia demasiado grande. Este espacio tendría que ser llenado con soluciones que respondiesen a la pregunta de cómo los hábitos humanos coinciden con las restricciones técnicas del ordenador. Si la distancia entre el sistema técnico y el usuario no está cubierta adecuadamente se observa el fenómeno frecuente según el cual, si bien se dispone de una solución técnicamente potente, sin embargo no resulta útil.

El objetivo de este trabajo es contribuir a la investigación de esta distancia. El punto de partida es la reflexión sobre el hecho de que el trabajo con el ordenador representa una acción lingüística que se distingue de las acciones lingüísticas tal y como las conocemos entre los seres humanos.

En la primera parte de este trabajo se define la expresión *interfaz de usuario* con la perspectiva de la acción lingüística en primer plano. En la segunda parte se presta atención a los elementos del interfaz de usuario, por así decirlo, con la lengua entre el usuario y el sistema. En una tercera parte, y a modo de conclusión, se sugieren unas propuestas de normas para el diseño del interfaz de usuario independientemente del contenido del *software*.

### **1) El interfaz de usuario: definición y tema**

El interfaz de usuario son las partes del *software* que el usuario puede utilizar a través de la pantalla. La enciclopedia *Webopedia* precisa en este

sentido:

*"The junction between a user and a computer program. An interface is a set of commands or menus through which a user communicates with a program. A command-driven interface is one in which you enter commands. A menu-driven interface is one in which you select command choices from various menus displayed on the screen.*

*The user interface is one of the most important parts of any program because it determines how easily you can make the program do what you want. A powerful program with a poorly designed user interface has little value. Graphical user interfaces (GUIs) that use windows, icons, and pop-up menus have become standard on personal computers."*

(Webopedia, [http://www.pcwebopaedia.com/TERM/U/user\\_interface.html](http://www.pcwebopaedia.com/TERM/U/user_interface.html)).

Esta definición acentúa la comunicación entre el ser humano y la máquina mediante la entrada de órdenes, menús, ventanas, botones y "pop-up-menús"<sup>1</sup>. Las disciplinas que entran en concurso son la Psicología (percepción, interacción entre ser humano y máquina), la Lingüística (lenguaje profesional o tecnolecto, terminología), el Diseño gráfico, y, naturalmente, la Informática. A continuación, nos vamos a referir principalmente a los aspectos de la acción lingüística necesaria para el manejo del *software*: ¿Cómo comunican los usuarios entre sí? ¿Cómo se orienta el usuario dentro del *software*?

El trabajo de un usuario con un programa del ordenador es en primer lugar un diálogo entre dos interlocutores. A diferencia de un diálogo entre seres humanos hay algunas particularidades en la comunicación entre el ser humano y la máquina. Este tipo de diálogo deroga diversas reglas de la conversación humana y las sustituye por otras.

Una característica esencial de la comunicación humana es **la tolerancia con respecto a los errores**. Puesto que cada información contiene mucha redundancia y como a menudo está codificada más de una vez (por ejemplo, la

---

<sup>1</sup>De ahora en adelante se entiende por *menú* un índice digital que resume los contenidos y las funciones del *software*. Un menú normalmente está situado en la moldura superior de una ventana o en el margen izquierdo en forma de una lista que se deja abrir. Las *ventanas* sirven de pizarra para la representación de información. Los *iconos* son representaciones gráficas que se utilizan para caracterizar funciones. Los *menús desplegables de acciones* o menús que se abren en una nueva ventana son menús pequeños que se utilizan

frase: *"Me alegro"* posiblemente esté acompañada de una sonrisa), el proceso de entendimiento puede ser perturbado sin que la comunicación fracase. El comunicante puede recurrir a muchas ayudas para entender el significado (por ejemplo, conocimiento del contenido y del contexto de un tema); por eso un tren que pasa durante una conversación telefónica delante de la ventana no perturba necesariamente la conversación. Otra característica de la comunicación humana es **la negociación del significado**, i.e., el significado sólo se desarrolla por la interacción entre dos comunicantes. De esta manera es posible que puedan surgir algunos malentendidos, pero lo más probable es que también puedan ser eliminados. El proceso de la negociación cambia el conocimiento y la actitud de los comunicantes: o bien se adaptan uno al otro o la comunicación fracasa. Además, **las posibilidades de las acciones lingüísticas** son casi ilimitadas, ligadas siempre al conocimiento de los usuarios. El usuario puede ampliar su conocimiento en todo momento **aprendiendo** nuevos temas. Einon (2000) precisa en cuanto a la negociación del significado y a los estudios:

*"The negotiation of meaning is a productive process but negotiating meaning is not constructing it from scratch. Meaning is not pre-existing, but neither it is simple made up. Negotiated meaning is at once both historical and dynamic, contextual and unique"* (Einon 2000, extract from a slide show).

Este enunciado, fundamentado en el Constructivismo, explica que tanto el entendimiento como la comunicación humana se basan en procesos cuyos resultados sólo son el producto, la comprensión o el acto de habla (cf. Weis, 2000).

Si se transfieren los aspectos de la comunicación humana descritos en el párrafo anterior a la interacción entre el ser humano y la máquina, hay que constatar algunas diferencias: el ordenador es absolutamente intolerante en relación a los errores. El más pequeño error está sancionado: el programa no ejecuta la acción pedida. El usuario tiene que teclear su clave de acceso sin error, porque, si no lo hace así, el *software* no es capaz de identificarlo como usuario autorizado. El ser humano, en cambio, trata los errores de manera

---

en formularios o en una página Web para que el usuario pueda elegir entre varias alternativas.

mucho más flexible y, a pesar de todo, es capaz de decidir correctamente. Igualmente, no es posible negociar un significado con el ordenador. Hay sólo correcto o falso. Si bien hay programas que saben "aprender", no obstante también este "aprender" está basado en algoritmos lógicos. Numerosas investigaciones científicas han llevado a la conclusión de que el aprendizaje humano no se deja reducir a tales algoritmos. El radio de acción del ordenador está restringido a los datos introducidos por el ser humano y a las deducciones lógicas de estos datos. El significado está determinado a priori y no es negociable.

Estas explicaciones demuestran que la comunicación entre el ser humano y la máquina está restringida fuertemente en comparación con la comunicación entre seres humanos. Resulta necesario establecer convenciones con las que trabaja el *software* y que el usuario tiene que conocer para que el ordenador facilite el trabajo del usuario de verdad en vez de poner nuevos problemas. El siguiente apartado se ocupa de estas convenciones y elementos del diseño del interfaz de usuario para aplicaciones basadas en la Web.

## **2) Elementos del diseño del interfaz de usuario de un software basado en la tecnología de la Web**

Durante la historia del desarrollo de *software* se han impuesto algunas convenciones que facilitan al usuario el manejo del *software* a pesar de los inconvenientes descritos del ordenador. Se pueden distinguir los siguientes sectores:

### **A) La lengua**

El idioma es el medio de comunicación más importante con el usuario. Los contenidos lingüísticos para el programa de *software* están previstos por el programador. A través de la lengua, los contenidos son presentados al usuario y, a través de la lengua, se realizan muchas interacciones entre el programa y el usuario. Una parte importante de cada programa es **la terminología**. Ésta se compone de dos campos: del vocabulario técnico del ordenador y del vocabulario técnico del contenido de la aplicación. El vocabulario del ordenador

es conocido por los usuarios que están familiarizados con el medio. Se trata, por ejemplo, de palabras como "ventana", "hiperenlace", "navegar" etc. Se recomienda tener en cuenta las convenciones para que el usuario no esté obligado a aprender detalles que al fin y al cabo no le servirán mucho. En cuanto al segundo campo, el objeto del programa, se supone que el usuario conoce el vocabulario porque es su campo de trabajo (pongamos por caso, contabilidad). En cambio, hay que tener en cuenta que la terminología está integrada correctamente dentro del programa. Aparte lo ya señalado, son interesantes desde una perspectiva lingüística y psicológica las **invitaciones de entrada de datos**, las **instrucciones de acción** (por ejemplo, a través de formularios o editores de texto) y los **comentarios del programa** en referencia a acciones terminadas con éxito. Invitaciones, instrucciones y comentarios extienden el campo de la terminología porque comprenden frases enteras. Tendrían que ser formulados siempre en el mismo estilo claro. Los **textos** son el tercer campo al que se tendría que prestar atención. Hay que tener en cuenta los conocimientos de la producción textual en la Web: los textos tienen que ser compactos y representados en unidades concluidas. Textos largos y complicados no son aptos para el uso. En este sentido, las construcciones sintácticas no tendrían que contener frases subordinadas largas. Como el ordenador sólo entiende la lengua en un sentido muy limitado, todas las entradas del usuario son grabadas mediante un procedimiento determinado (por ejemplo, en bancos de datos y directorios). Estos procedimientos y, sobre todo, el fondo técnico no son siempre comprensibles al usuario. Razón por la cual es mayor la necesidad de que las informaciones sean formuladas claramente.

Los textos y la información lingüística son una parte importante de cualquier *software*. Gracias a ellos, el usuario y el *software* se comunican. Pese a ello, no se les considera un campo independiente. Las inconsistencias que resultan de este procedimiento afectan a la calidad del interfaz de usuario. Dos ejemplos concluyentes de dos programas de *software* deben ilustrar una vez más la problemática:

*"Desgraciadamente ocurrió un error desconocido. Probablemente esto no*

*significa nada bueno."*

(Cf.: <http://ourworld.compuserve.com/homepages/klartech/schmu12a.htm>: un error de mensaje traducido del alemán).

*"Error: Keyboard not attached. Press F1 to continue."*

(Cf.: <http://users.bart.nl/~jelsma/jokes/gPcjokes.htm> ).

El primer mensaje de error es completamente inútil porque no ofrece ninguna solución del problema al usuario. El segundo es una instrucción de acción no practicable e inaceptable: no es posible continuar trabajando. Es evidente que estos son dos ejemplos extremos, pero numerosos obstáculos ocurren cuando se pregunta a los usuarios o cuando se localiza el *software* (traducción del *software* a varios idiomas). Es en el momento en que se extraen todos los textos para la traducción del *software* (que antes estaban integrados dentro del código del programa) cuando se producen las inconsistencias.

## B) La representación de la información<sup>2</sup>

Otro punto importante es la manera como se presenta la información en la pantalla y la forma en que se organiza. Estas informaciones pueden ser **textos, formularios, listas, tablas, fotos, diagramas e ilustraciones**. Elementos como la **escritura** y los **colores** tienen una función de refuerzo para facilitar al usuario la orientación y la legibilidad de la información. Vale lo mismo para los **iconos**. Todos estos elementos deben tener un significado y hay que utilizarlos siempre con este significado. Para organizar la información se utilizan normalmente las **estructuras de árboles y listas**.<sup>3</sup>

Otro elemento importante de representación son las **ventanas**. Si el usuario trabaja con aplicaciones diferentes, es útil que tenga varias ventanas a

<sup>2</sup> La representación de la información está muy ligada con el tema de la visualización gráfica. Que se trata este tema aquí de una manera marginal no tiene nada que ver con que se lo considere secundario. Su tratamiento excede los límites del presente trabajo.

<sup>3</sup> Hay algunos principios que intentan presentar la idea de la red con las llamadas técnicas de "zooming". Se refieren a la suposición popular de que el conocimiento en la cabeza del ser humano está representado en una red y que por eso las informaciones que están representadas en el ordenador en forma de una red pueden ser comprendidas más fácilmente. En el contexto más amplio de las teorías del aprendizaje estas suposiciones están discutidas (cf. McKendree, J. Reader, W. Hammon, N. (1995)). Por esta razón y por los problemas técnicos ligados todavía con este tema, la perspectiva de zoom no se considera aquí.

su disposición para que no haya que arrancar cada aplicación cuando se la utiliza de nuevo. Para la organización del interfaz de usuario es importante definir reglas cuando una información o una aplicación aparece en una nueva ventana o cuando se recomiende un cambio de página.

### C) La navegación y la interacción

La conducción del usuario dentro del programa es una tarea fundamental. Se necesita una serie de elementos de navegación que el usuario tiene que conocer y comprender para trabajar con el programa. Estos elementos normalmente tienen aspectos lingüísticos y gráficos.

Hay que tener en cuenta igualmente que la evolución de un concepto consistente de navegación es inevitable. Además hace falta un plano para la conexión en red (reticulación) de los diferentes estados del *software*.

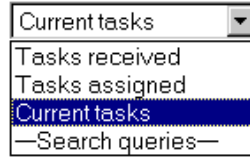
Un *software* basado en la tecnología de la Web funciona igual que un sitio Web, como un hipertexto, es decir, informaciones que están representadas en una red dentro de la cual el usuario puede navegar. En general, hay tres posibilidades de organización de la navegación:

**Hiperenlaces:** palabras azules subrayadas o frases conducen como en un sitio Web a una nueva página. Ejemplo: [Lista de las direcciones del proyecto](#)

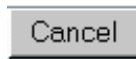
**Menú desplegable de acciones:** son menús que permiten una selección entre opciones diferentes y que pueden conducir a una nueva página. Las llamadas pestañas son una alternativa que se encuentran como ayuda de navegación y de estructuración. Ejemplo: El siguiente menú desplegable hace posible la navegación entre las listas de « tareas actuales », « tareas recibidas » y « tareas mandadas »<sup>4</sup>:

---

<sup>4</sup> Este y los siguientes ejemplos provienen del *software* Intranet « Intraline » de la Channel One GmbH, Alemania.



**Botones:** sirven para desencadenar órdenes que también pueden conducir a un cambio de página. Contienen textos o iconos. Ejemplos : «Cancel» and «Save» sirven para terminar una acción o almacenar los datos



El siguiente icono representa la acción de eliminar los datos seleccionados:



Una particularidad frecuente de un *software* basado en la tecnología de la Web es la manera de interacción entre el cliente y el servidor. Como no sólo los datos son calculados en la parte del servidor, sino también la presentación de la pantalla, se generan desde allí retrasos temporales que no son infrecuentes. Esto cuesta mucho tiempo de espera para el usuario que posiblemente se aumenta más por restricciones técnicas. Por eso es importante tener en cuenta que los cambios de página ocurran con la menor frecuencia posible. Evidentemente este hecho coincide con la exigencia de que no debe aparecer demasiada información en una página por razones de legibilidad. Encontrar una solución para este conflicto es otra tarea del concepto de navegación.



## D) La tramitación de datos

Las funciones estándares para la tramitación de datos como "nuevo", "almacenar", "tramitar", "eliminar" y "buscar" son realizadas a través de formularios y botones. Se añade a éstas la función "mover", por ejemplo, cuando se intenta mover un documento a otro archivo. Aplicaciones tradicionales de *desktop* ofrecen al usuario un "drag and drop" (esto es, que un elemento es movido por medio del ratón a otro sitio). Esta función no se realiza tan fácilmente en un *browser*, así que hay que recurrir a las funciones "cortar" e "insertar".

### 3) *Proposiciones de normas para el diseño del interfaz de usuario*

Las explicaciones que hemos realizado en los dos primeros apartados se pueden agrupar en doce proposiciones de normas que deben ayudar en el desarrollo de un interfaz de usuario que sea a la vez consistente y útil para el usuario.<sup>5</sup>

1. Atiende desde el principio a **un catálogo de terminología** que distinga entre terminología técnica del ordenador y terminología temática.
2. Las informaciones e instrucciones tienen que ser **inequívocas**. Escríbelas también en un catálogo para asegurar su consistencia estilística y de contenido.
3. Escribe **el menor texto posible** en una página. La legibilidad en la pantalla es mucho peor que en el papel. Sólo el texto que hubiera de imprimirse tendría que aparecer como texto largo. Por lo demás, los textos en la pantalla suelen ser cortos y precisos, y se utilizan hiperenlaces y marcas de salto, es decir, el principio de hipertexto, para vincular contenidos.
4. Preste atención a **una disposición adecuada de la pantalla**. La legibilidad se mejora si el usuario puede orientarse en la pantalla a primera vista. Elementos de navegación que se repiten continuamente tendrían que aparecer siempre en el mismo lugar.

<sup>5</sup>Las proposiciones se refieren a un *software* basado en la tecnología de la Web, pero la mayoría también es aplicable para *software* que no se basa en esta tecnología o para el desarrollo de páginas de Web.

5. **El significado y la utilización de símbolos**, botones, iconos, escritura y colores tienen que ser definidos. También se recomienda la confección de catálogos.
6. Decida a tiempo cómo **la información será representada** (listas, texto entero, árboles, tablas, gráficos, formularios, etc.). Piense, en el caso de sitios Web, si los gráficos animados son necesarios de verdad.
7. Haga la **navegación** en un compromiso entre longitud de camino, lo más pequeño posible, y la protección de las proposiciones de normas del interfaz de usuario. Considere que no es el camino más corto el que siempre es el mejor. Proponga un mapa que documente la coordinación entre las páginas.
8. Tenga cuidado con **el cambio de páginas** en una aplicación basada en la Web. Cada clic de botón del usuario tiene que ser mandado al servidor para que devuelva una respuesta. Esto cuesta tiempo.
9. Procure **un mecanismo de ventana** que sea uniforme. ¿Cuándo se abre una nueva ventana? ¿Cuándo la información aparece en la ventana actualmente utilizada?
10. Tenga en cuenta que el *software* **acepta las acciones del usuario**. Hay que evitar absolutamente una pérdida de datos.
11. Tenga en cuenta que el usuario recibe **respuestas comprensibles** del *software*.
12. Preste atención al **rendimiento del software (o capacidad práctica)**. Cuanto más tiempo el usuario tenga que esperar la reacción a su entrada, mayor es el peligro de que rechace el programa.

### **REFERENCIAS BIBLIOGRÁFICAS**

- Constantine, L.L., Lockwood, L.A.D. (1999), "Software for Use", Addison Wesley.
- Einon, G.S. (2000), "Communities of Practice":  
<http://telematics.open.ac.uk/res/eating/einon/> (The Open University).
- McKendree, J., Reader, W., Hammon, N. (1995), "'The homeopathic fallacy' in learning from Hypertext", in: *Interactions*, July 1995, S.82-94.
- Raskin, J. (2000), "The Humane Interface - New Directions for Designing

Interactive Systems", Addison Wesley, Boston.

Webopedia: Online-Lexikon zu Fragen aus der Computerwelt. Diccionario *on line* con preguntas sobre el mundo del ordenador:

[http://www.pcwebopaedia.com/TERM/U/user\\_interface.html](http://www.pcwebopaedia.com/TERM/U/user_interface.html))

Weis, U. (2000), "Lesen in der Fremdsprache Deutsch - Eine empirische Studie zum Lesen linearer Texte im Vergleich zu Hypertexten", Libri, Books on Demand, Hamburg.

### **OTRAS REFERENCIAS DE INTERÉS**

Mensajes de error curiosos en programas de *software* :

<http://users.bart.nl/~jelsma/jokes/gPcjokes.htm>

Referencia de *software*:

*Intraline*, un *software* estándar de intranet, desarrollado por la empresa Channel One GmbH. Véase [www.intraline.de](http://www.intraline.de) o [www.intraline.com](http://www.intraline.com) .<sup>6</sup>

---

<sup>6</sup> La idea de este trabajo con las proposiciones de normas es el resultado de un proyecto de desarrollo del *software Intraline*.