



UNIVERSIDAD NACIONAL DE EDUCACION A DISTANCIA

ESCUELA TECNICA SUPERIOR DE INGENIERIA INFORMATICA

Proyecto de Fin de Carrera de Ingeniero Informático

**MEDIDA DE LA USABILIDAD EN APLICACIONES DE
ESCRITORIO. UN MÉTODO PRÁCTICO.**

MARIO LORENZO ALCALA

Dirigido por: JESUS MARIA MINGUET MELIAN

Curso: 2006-07 (convocatoria de octubre)



MEDIDA DE LA USABILIDAD EN APLICACIONES DE ESCRITORIO.
UN MÉTODO PRÁCTICO.

Proyecto de Fin de Carrera de modalidad *oferta específica*

MARIO LORENZO ALCALA

Dirigido por: JESUS MARIA MINGUET MELIAN

Tribunal calificador:

Presidente: D./D^a.....
(Firma)

Secretario: D./D^a.....
(Firma)

Vocal: D./D^a.....
(Firma)

Fecha de lectura y defensa:.....

Calificación:.....

RESUMEN

La interfaz de usuario es una parte importantísima en el desarrollo de aplicaciones informáticas y es clave en el éxito o fracaso a la hora de la puesta en producción de la aplicación en el mundo real. A través de ella, el usuario interactúa con la aplicación con más o menos dificultad. Esta mayor o menor dificultad es determinante cuando una empresa elige sus aplicaciones empresariales porque de ellas dependerá de forma muy importante su productividad y costes.

Se presentan una serie de heurísticas muy conocidas en el mundo de la calidad del software y que son debidas a J. Nielsen. Estas heurísticas son ampliadas y comentadas con algunos ejemplos, sirviendo de forma general para la mejora de las interfaces de usuario.

Se desarrolla una métrica para la medida de la facilidad de uso (usabilidad) de una interfaz de usuario y poder realizar comparaciones entre diferentes soluciones para la misma aplicación en desarrollo.

La métrica está basada en las heurísticas de J. Nielsen y cada una de ellas está formada por una lista de elementos verificables extraídos de documentos de la empresa Xerox. Estas listas de elementos se han modificado para adaptarlas a los tiempos actuales y a estándares de facto.

Establecido el método, se ha validado este empíricamente, midiendo la facilidad de uso de una serie de aplicaciones de escritorio actualmente en funcionamiento en multitud de empresas de todo el mundo.

Se presentan los resultados de las mediciones, comentándolos y estableciendo unas conclusiones finales.

PALABRAS CLAVE

Usabilidad

Facilidad de uso

Diseño centrado en el usuario

ISO/IEC 9126

Heurísticas de usabilidad

Proceso de evaluación

Métrica

Medida de la interfaz

Facilidad de aprendizaje

Interfaz de usuario

Principios de diseño centrados en el usuario

Calidad del software

Ingeniería de usabilidad.

**DESK APPLICATIONS USABILITY MEASUREMENT.
A PRACTICAL METHOD.**

ABSTRACT

User's interface is a very important part in the development of computer applications and it is key in the success or failure when it's setting in production of the application in the real world. Through it, the user interacts with the application with more or less difficulty. This bigger or smaller difficulty is decisive when a company chooses its managerial applications because of them it will depend in a very important way its productivity and costs.

They are presented a series of heuristic very well-known in the world of the quality of the software and that they are due to J. Nielsen. These heuristic ones are enlarged and commented with some examples, serving in a general way for the improvement of user's interfaces.

A metric one is developed for the measure of the use easiness (usability) of user's interface and power to carry out comparisons among different solutions for the same application in development. The metric method is based on the heuristic of J. Nielsen and each one of them is formed by a list of elements or extracted confirmations of documents of the company Xerox. These lists of elements have modified to adapt them at the current times and to standard of facto.

Established the method, this has been validated empirically, measuring the easiness of use of a series of desk applications at the moment in operation in multitude of companies from all over the world.

The results of the mensurations are presented, commenting them and establishing some final conclusions.

KEYWORDS

Usability

Use easiness

User-centred design

ISO/IEC 9126

Heuristic of usability

Evaluation process

Metric

Measure of the interface

Learning easiness

User's interface

Design principles user-centred

Quality of the software

Usability engineering.



INDICE

1. INTRODUCCION.....	13
1.1 Objetivos.....	16
2. USABILIDAD.....	17
2.1 Definición.....	18
2.2 Los beneficios de usabilidad.....	19
2.3 ISO/IEC 9126.....	22
2.4 Modelo de proceso de la Ingeniería de usabilidad.....	26
3. PRINCIPIOS DE DISEÑO CENTRADOS EN EL USUARIO.....	31
3.1 Heurísticas de usabilidad.....	32
4. MEDICION DE LA INTERFAZ GRAFICA DE USUARIO.....	41
4.1 El proceso de evaluación de ISO 9126.....	41
4.2 Métodos de evaluación.....	44
4.2.1 User Testing.....	44
4.2.2 Usability Inspection.....	45
4.2.3 Métodos de mejora de la usabilidad.....	47
4.2.4 Criterios cuantificables de usabilidad de Constantine y Lockwood.....	47
4.2.5 Características de un método de evaluación.....	51



5. MEDIDA DE LA USABILIDAD.....	53
5.1 Metodología.....	53
5.1.1 Algoritmo de cálculo para cada heurística.....	54
5.1.1.1 Expresión matemática.....	55
5.2 Medida de las observaciones.....	56
5.3 Criterios de asignación de pesos a las heurísticas.....	57
5.3.1 1 ^{er} Criterio.....	57
5.3.2 2 ^o Criterio.....	58
5.3.3 3 ^{er} Criterio.....	59
5.3.4 4 ^o Criterio.....	60
5.3.5 Resumen de pesos y sus medias.....	61
5.4 Expresión matemática de la métrica de usabilidad.....	63
5.5 Aplicación de la métrica a todas las aplicaciones.....	63
5.6 Heurísticas detalladas de usabilidad de Xerox.....	65
5.7 Establecimiento de la escala de medida.....	65
5.8 Aplicación de la medida a las aplicaciones reales.....	68
5.8.1 Selección de las aplicaciones a medir.....	68
5.8.2 Resultados de las medidas realizadas.....	69
5.9 Aplicación de la métrica al resto de aplicaciones.....	71
5.10 Estudio de los resultados.....	72
5.11 Análisis de las heurísticas del apéndice 3.....	79
6. CONCLUSIONES.....	81
7. APENDICE 1. Heurísticas originales de Xerox.....	88
8. APENDICE 2. Heurísticas traducidas al castellano modificadas.....	99



9. APENDICE 3. Gráficos de los resultados por cada heurística.....	116
10. APENDICE 4. Alternativa a los criterios de valoración de pesos.....	121
11. ANEXO 1. Análisis y descripción de la aplicación “Usable”. Disco contraportada.	
12. BIBLIOGRAFIA.....	126
13. REFERENCIAS WEB.....	128
14. ABREVIATURAS.....	130



INDICE DE FIGURAS

2.1 Relaciones de calidad.....	23
2.2 Características de calidad del software ISO/IEC 9126.....	24
2.3 Modelo de calidad para Calidad en Uso ISO/IEC 9126.....	25
2.4 Modelo de Proceso de la Ingeniería de la Usabilidad.....	27
3.1 Visibilidad del estado del sistema.....	32
3.2 Funciones deshacer y rehacer.....	33
3.3 Información de error.....	35
3.4 Como en el mundo real.....	36
3.5 Reconocer antes que recordar.....	37
3.6 Flexibilidad de uso.....	38
3.7 Mensaje de error mal formateado, sin información.....	39
3.8 Ayuda con pasos concretos y claros.....	40
4.1 Proceso de evaluación de ISO/IEC 9126.....	42
5.1 Los cuatro criterios y su media.....	62
5.2 Escala de usabilidad.....	67
5.3 Gráfico de resultados de la aplicación de la métrica.....	71
5.4 Reparto de aplicaciones buenas y malas.....	73
5.5 Dispersión de las observaciones.....	74
5.6 Explicación del gráfico boxplot.....	74
5.7 Relación de los porcentajes de valores menores de 80 y de 40.....	76
5.8 Rendimientos decrecientes.....	77
5.9 Resultados globales.....	79



INDICE DE TABLAS

4.1 Métodos de mejora de la usabilidad.....	47
5.1 Medida de las observaciones.....	56
5.2 Pesos del 1 ^{er} criterio.....	57
5.3 Mezcla de aplicaciones buenas y malas.....	58
5.4 Pesos del 2 ^o criterio.....	59
5.5 Pesos del 3 ^{er} criterio.....	59
5.6 Aplicaciones con medias de observaciones mayores de 80.....	60
5.7 Pesos del 4 ^o criterio.....	61
5.8 Resumen de asignación de pesos.....	61
5.9 Tabla definitiva de medición de métricas.....	64
5.10 Medida máxima de usabilidad.....	66
5.11 Selección de aplicaciones a medir	69
5.12 Resultados de la aplicación de la métrica.....	70
5.13 Desviación típica y variabilidad de las observaciones.....	75
5.14 Datos y porcentajes resumidos.....	75
5.15 Correlaciones de Pearson entre observaciones.....	78



1. INTRODUCCION

En la mayoría de las empresas de desarrollo de software de tamaño medio o pequeño, la principal preocupación a la hora de desarrollar una aplicación encargada por un cliente es que tal desarrollo tenga los costes mas reducidos posibles, por lo que se realiza de una forma precipitada y no completa, dando muy poca importancia al estudio de la interfaz de usuario y al estudio de la usabilidad. Ha sido demostrado por estudios de conocidos investigadores en el área de usabilidad que la aplicación de la ingeniería de la usabilidad al proceso de desarrollo en las empresas decremanta de manera dramática los costes de producción y aumenta los beneficios.

Refiriendo algunas estadísticas para ver la importancia que la interfaz de usuario tiene en el total del desarrollo de un sistema se puede mencionar [Sanz 1996]:

- La interfaz de usuario es aproximadamente el 60% de las líneas de código del total de un sistema de información interactivo.
- Una interfaz de usuario gráfica supone como mínimo el 29% del presupuesto de desarrollo de un sistema de información interactivo.
- El 80% de los costes en el ciclo de vida del desarrollo del software se producen después de que el sistema se haya liberado y a su vez el 80% de esos costes se deben a requisitos no cumplidos.



Las empresas de desarrollo raramente emplean métodos de la ingeniería de usabilidad en sus proyectos reales por la creencia de que la aplicación de la usabilidad a sus desarrollos incrementaría mucho los costes.

En sus estudios sobre 31 proyectos, Jacob Nielsen [1] estima que el esfuerzo necesario para aplicar técnicas y métodos de la ingeniería de usabilidad al proceso de desarrollo es de dos personas-año, y como mucho de cuatro personas-año.

En muchas ocasiones es el usuario el que se adapta a la aplicación. Esto último es impensable en estos tiempos, en los que la competitividad en el mundo de las tecnologías de la información está a tan alto nivel.

La interfaz de usuario es un componente crítico de la aplicación, ya que a través de ella, el usuario, interactúa con el sistema y para él el sistema es lo que ve y lo que toca. Una interfaz de usuario debe transmitir al usuario de la aplicación todo de lo que es capaz de hacer el sistema. El usuario sólo está interesado en la realización de una serie de tareas, para lo cual utiliza la aplicación en cuestión, y esta debe de poder realizar las tareas de la forma más rápida, intuitiva y eficaz posible.

Una interfaz de usuario mal diseñada desencadena una serie de problemas relacionados con los tres adjetivos del párrafo anterior:

- Rápida: reducción de la productividad, incrementando los tiempos en la realización de las tareas.
- Intuitiva: aumentando el tiempo de aprendizaje necesario para utilizar la aplicación.
- Eficaz: alcanzando unos niveles de errores muy altos en la realización de las tareas.



Un adjetivo a añadir a los tres anteriores sería lo atractivo del diseño. Un producto atractivo se vende mejor y se usa más que otro sin atractivo.

Todos los adjetivos anteriores son contemplados en el Modelo de calidad ISO/IEC 9126 a través del concepto de usabilidad. Esta, mide el grado de utilidad, facilidad de uso, facilidad de aprendizaje y la apreciación del producto en un contexto determinado de uso.

Un producto usable tiene las siguientes características:

- Es fácil de aprender
- Es eficiente en el uso
- Facilita una rápida recuperación de errores
- Es fácil de recordar
- Es fácil de usar y amigable en el uso
- Es visualmente agradable

La usabilidad reduce costes y proporciona una ventaja competitiva a la empresa desarrolladora de la aplicación. Las actividades de la usabilidad son simples de realizar y se pueden realizar en la propia empresa.

Teniendo en cuenta los beneficios que proporciona la usabilidad, esta debería ser el eje central en el proceso de diseño de la aplicación.

Un equipo de desarrollo se debería de centrar en el diseño de la interfaz gráfica de usuario siguiendo unos principios de diseño que muchas empresas e investigadores realizaron en base a su experiencia en este campo y en su evaluación para corregir posibles errores habidos en el periodo de diseño.



1.1 Objetivos

Se pretende en este trabajo la realización de una cuantificación de la usabilidad a través de las heurísticas de usabilidad de Jakob Nielsen, basada en los principios de diseño centrados en el usuario. Para ello se utilizan listados detallados de la empresa Xerox que desmenuzan las heurísticas en elementos directamente verificables. Estos listados serán reformados para adaptarlos a los tiempos actuales, eliminando, modificando o añadiendo nuevos elementos a las heurísticas.

A partir de las heurísticas modificadas se dará diferente peso a cada una y se establecerá un modelo para la medición de la usabilidad a través de una escala de valores.

La validación del modelo se realizará empíricamente a través de la medición de una serie de aplicaciones. Unas de probado “prestigio” que se pueden considerar estándar y otras menos conocidas. También formarán parte del grupo de aplicaciones a medir algunas que se desarrollaron a medida y están funcionando en varias empresas.

Por supuesto, el valor obtenido de las aplicaciones consideradas estándar debe ser superior al de otras no tan estándar. Esto se considerará en las conclusiones al final del trabajo.

En aplicación de la métrica de usabilidad propuesta, se desarrollará una aplicación informática para su utilización en la práctica.



2. USABILIDAD

La usabilidad es una característica que está relacionada con la medida de la calidad de los sistemas interactivos usados por usuarios específicos en un contexto de uso, para conseguir objetivos específicos con efectividad, eficiencia y satisfacción, en términos de utilidad, facilidad de uso, facilidad de aprendizaje y apreciación.

- *Efectividad*: la precisión y completitud con la que el usuario alcanza sus objetivos. Relacionados con este concepto está la facilidad de aprendizaje, la tasa de errores cometidos y la facilidad de recordar sus funcionalidades y procedimientos.
- *Eficiencia*: los recursos empleados relacionados con la precisión y completitud con que el usuario alcanza sus objetivos. Relacionados con este concepto está la facilidad de aprendizaje, la tasa de errores cometidos y la facilidad de la aplicación para ser recordada.
- *Satisfacción*: la ausencia de incomodidad y una actitud positiva hacia el uso de la aplicación. La aplicación debe ser agradable de usar de forma que el usuario esté satisfecho al utilizarla.

Utilidad: capacidad de la aplicación para ayudar en la realización de tareas.

Facilidad de uso: está relacionada con la eficiencia o efectividad con que se realizan las tareas. En una aplicación fácil de usar se realizarán las tareas más rápidamente.



Facilidad de aprendizaje: es una medida del tiempo requerido para usar la aplicación con cierta eficiencia y llegar a recordar los procedimientos después de no usar la aplicación durante un tiempo determinado.

Apreciación: es una medida de la percepción, opinión, sentimiento y actitud generada en el usuario por el uso de la aplicación. Es una medida subjetiva pero muy importante.

2.1 Definiciones de usabilidad

El término usabilidad es un anglicismo que significa “facilidad de uso”.

Tiene su origen en la expresión “user friendly”.

Se pueden encontrar en la literatura sobre el tema varias definiciones sobre la usabilidad:

Según ISO 9241-11: “La medida en la que un producto se puede usar por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso específico”.

Según ISO 9126-1: “La capacidad que tiene un producto software para ser atractivo, entendido, aprendido, usado por el usuario cuando es utilizado bajo unas condiciones específicas”.

Esta norma incluye la usabilidad como un parámetro de calidad del software junto con otros cinco parámetros: funcionalidad, fiabilidad, eficiencia, mantenibilidad y portabilidad.



Según J. Nielsen: La definición de este autor, indica que un sistema usable debe poseer los atributos: capacidad de aprendizaje, eficiencia en el uso, facilidad de memorizar, tolerante a errores y satisfactorio.

Según J. Preece: Esta autora propone la siguiente definición: “desarrollo de sistemas fáciles de usar y de aprender”.

Todos coinciden en que un producto usable debe ser fácil de usar y de aprender.

2.2 Beneficios de la usabilidad

La ingeniería de la usabilidad aporta importantes beneficios referentes a los costes de desarrollo, la calidad del producto y la satisfacción del cliente.

Se pueden enumerar [Sanz 1996] algunos de los más importantes, entre los que se encuentran:

- Incremento de la productividad de los usuarios de la aplicación, al reducir los tiempos para completar sus tareas y cometer menos errores que deberían ser corregidos mas tarde. Esta productividad puede ser cuantificada por la propia empresa que implanta productos usables en su organización.
- Incremento del uso de la aplicación.
- Reducción del coste en documentación. Se necesitarán menos manuales de la aplicación.



- Reducción de los costes de soporte a la aplicación ya que resulta un producto fácil de instalar, de aprender y de usar.
- Reducción de los costes y tiempos de desarrollo, acertando a la primera con los requerimientos de los usuarios. Parece una contradicción, pero la realidad es que existe tal reducción evitando sobrediseños y reduciendo el número de cambios posteriores en la aplicación.

Una regla [10] de coste que siguen las organizaciones que tienen en cuenta la aplicación de ingeniería de usabilidad en sus desarrollos es que el ratio coste-beneficio por usabilidad es 1\$:10\$-100\$. Significa que una vez que el sistema está en desarrollo, corregir un problema de usabilidad cuesta 10 veces más que si el problema se hubiera corregido en el diseño. Corregir el problema cuando el sistema ya ha sido liberado cuesta 100 veces más que si se hubiera corregido en el diseño.

- Detección temprana de fallos, aplicando usabilidad en las fases tempranas del desarrollo. En las primeras fases de desarrollo, el coste [10] de los errores y omisiones es mucho menor que cuando la aplicación está casi terminada.
- Reducción de los costes de mantenimiento de la aplicación.



- Aumento de la satisfacción del cliente reduciendo el esfuerzo de uso por parte del usuario y mejorando la calidad de vida de los usuarios.
- Aumento de la calidad del producto final con el consiguiente ahorro en control de calidad. Aumento de la competitividad en el mercado.
- Aumento de la calidad de desarrollo de la organización con la adquisición de buenas prácticas a base de la aplicación de la usabilidad en todo el proceso de desarrollo lo que redundará en el desarrollo de futuros proyectos.
- Aumento de la reputación de la organización en el desarrollo de productos usables. Creación de imagen de marca y fidelización de clientes.
- Proporciona una ventaja competitiva para la organización. Una ventaja competitiva se consigue por reducción de costes o por diferenciación del producto. La aplicación de ingeniería de usabilidad proporciona una diferenciación del producto de los productos de la competencia que los clientes perciben rápidamente. Lo más fácil de usar es más sencillo de vender.



2.3 ISO/IEC 9126

La norma ISO/IEC 9126 nació como consecuencia de los intentos de estandarización de las medidas de calidad del software. Esta norma se basa en la definición de calidad de una norma anterior, la ISO 8402, la cual define la calidad como un conjunto de características del producto que afectarán a su habilidad para satisfacer las especificaciones del usuario.

La norma ISO/IEC consta de cuatro partes [Minguet 2003], de las cuales, sólo la primera está reconocida como un estándar:

1. Modelo de calidad
2. Métricas externas
3. Métricas internas
4. Métricas de calidad en uso

La primera parte, Modelo de calidad, a su vez está dividida en dos partes más:

- Calidad externa e interna
- Calidad en uso

La calidad externa está definida como la totalidad de las características del producto software desde un punto de vista externo, esto es, la medida en que un producto software satisface las necesidades especificadas cuando es utilizado en determinados contextos de uso. Son por lo tanto medidas obtenidas del comportamiento del sistema durante su funcionamiento.



La calidad interna, a su vez, está definida como la totalidad de las características del producto software desde un punto de vista interno, esto es, la medida de conceptos tales como productividad, reusabilidad, densidad de defectos, fiabilidad,...Son por lo tanto medidas que se realizan sobre componentes software no ejecutables: código fuente, documentación,...

La calidad de uso está definida como la capacidad del software para obtención de objetivos específicos con efectividad, productividad, satisfacción y seguridad. Por lo tanto es la medida en la que un producto software alcanza las especificaciones del usuario en términos de efectividad, seguridad, productividad y satisfacción.

Las relaciones entre la calidad externa, interna y calidad en el uso se pueden ver claramente en la figura siguiente.

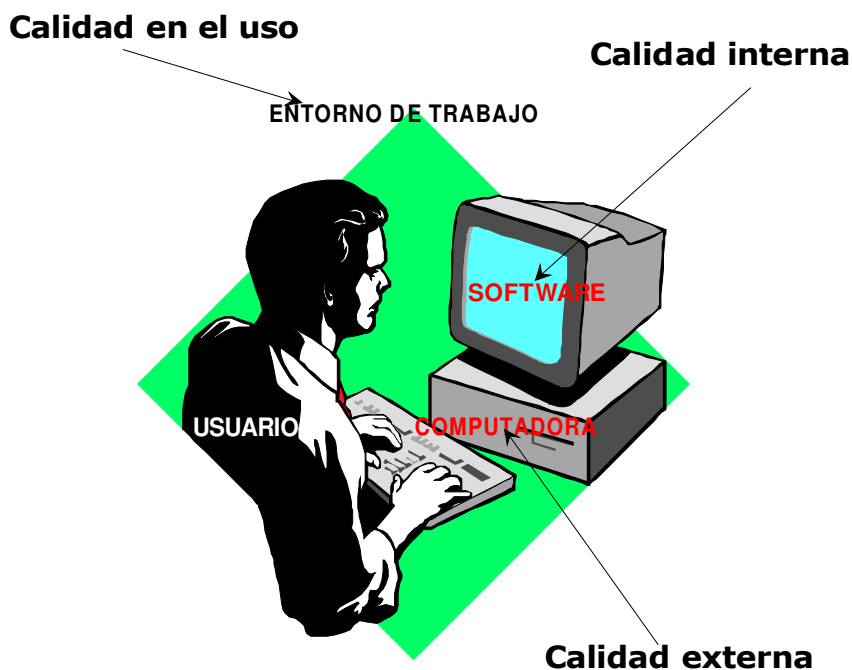


Figura 2.1 Relaciones de calidad



La calidad interna mide las propiedades estáticas del código, normalmente por inspección.

La calidad externa mide las propiedades dinámicas del código cuando es ejecutado.

La calidad en el uso mide cómo se satisfacen las necesidades del usuario en el entorno de trabajo.

El modelo de calidad para calidad externa e interna está definido en ISO/IEC 9126 de la forma mostrada en la siguiente figura:



Figura 2.2. Características de calidad del software ISO/IEC 9126

La norma ISO/IEC 9126 divide la calidad externa e interna en seis características: funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

A su vez, cada característica está descompuesta en otras subcaracterísticas.



A los efectos de este trabajo se explicarán solamente las subcaracterísticas de Usabilidad.

Capacidad para ser entendido: es la capacidad del producto software que permite al usuario entender si el software es adecuado y cómo puede ser usado para unas tareas específicas en unas condiciones de uso determinadas.

Capacidad para ser aprendido: es la capacidad del producto software que permite al usuario aprender sobre su aplicación.

Capacidad para ser operado: es la capacidad del producto software que permite al usuario usar la aplicación y controlarla.

Capacidad de atracción: es la capacidad del producto software de ser atractivo al usuario.

Cumplimiento de la usabilidad: es la capacidad del producto software de adherirse a normas, convenciones, guías de estilo y regulaciones existentes relacionadas con la usabilidad.

El modelo de calidad para calidad en uso está definido en ISO/IEC 9126 de la forma mostrada en la siguiente figura:

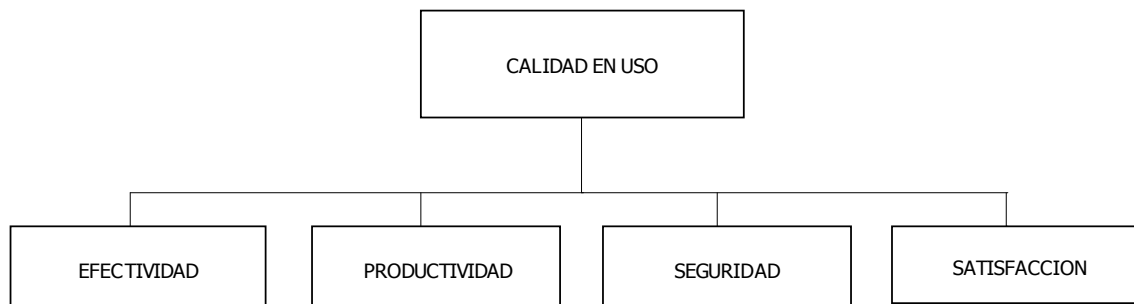


Figura 2.3. Modelo de Calidad para Calidad en Uso ISO/IEC 9126.



Efectividad: es la capacidad del producto software que permite a los usuarios alcanzar objetivos especificados con exactitud y completitud en un determinado contexto de uso.

Productividad: es la capacidad del producto software que permite a los usuarios emplear una cantidad adecuada de recursos en relación con la efectividad alcanzada en un determinado contexto de uso.

Seguridad: es la capacidad del producto software de alcanzar niveles aceptables de riesgo hacia las personas, negocio, software, propiedad o medio ambiente en un determinado contexto de uso.

Satisfacción: es la capacidad del producto software de satisfacer al usuario en un determinado contexto de uso.

2.4 Modelo de proceso de la Ingeniería de la usabilidad [12]

En función del tamaño de la organización se aplicará un modelo del ciclo de desarrollo más o menos complejo y más o menos costoso. En este trabajo se pretende establecer un método para la medición de la usabilidad en organizaciones pequeñas o medianas que no se pueden permitir un departamento de usabilidad diferenciado del resto del equipo de desarrollo.

Es clave para el proceso comenzar con el ciclo de diseño lo antes posible para tener una menor probabilidad de que la versión que se libere contenga errores importantes. En los prototipos es mucho más barato y rápido realizar las modificaciones.

El Modelo de Proceso de la Ingeniería de la Usabilidad especifica una metodología de desarrollo para guiar al equipo y conseguir un alto nivel de



usabilidad en el producto final. Este modelo es un diseño iterativo como el de la siguiente figura:

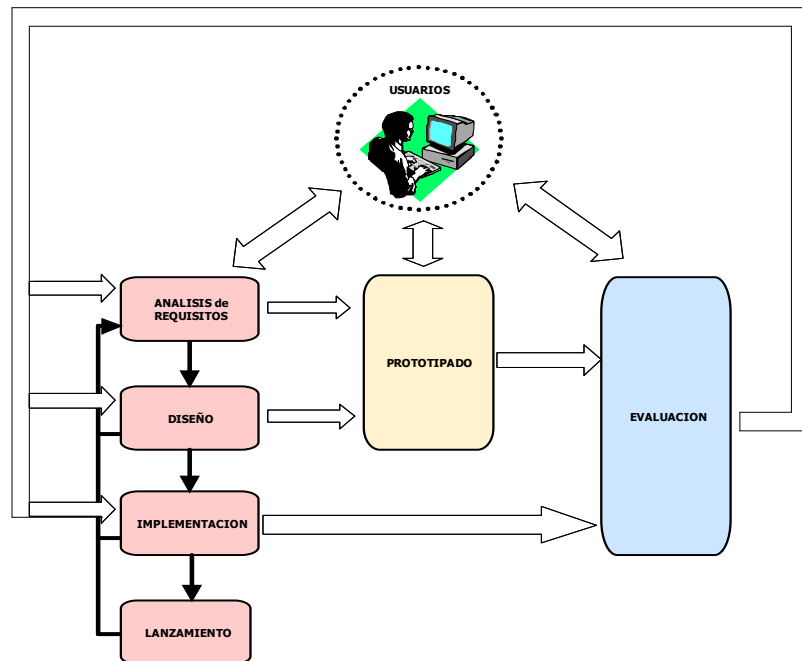


Figura 2.4. Modelo de Proceso de la Ingeniería de la Usabilidad

Análisis de requisitos.

1. Conocer al usuario.

Observación del usuario en su trabajo: realizar visitas al sitio de trabajo, realizar entrevistas, formularios,...

Conocer las características de los usuarios individuales: clasificarlos por su nivel de experiencia, por su nivel de educación, por edades, por limitaciones físicas,... Establecer perfiles de usuario.

Entender las motivaciones de los usuarios: entretenimientos, trabajo, aprender,...

Para que la aplicación sea usable se debe conocer y entender al usuario en su lugar de trabajo.



2. Establecer los objetivos de usabilidad.

3. Establecer listas de tareas

Especificar las funciones a realizar por cada tipo de usuario.

Diseño.

1. Análisis de las tareas.

Determinar los objetivos globales de los usuarios y cómo se llevan a cabo actualmente.

Definir el marco de trabajo conceptual para diseñar la aplicación con el conocimiento y la experiencia del usuario objetivo.

Desarrollo del modelo conceptual con diagramas y notaciones formales.

Adquisición del modelo mental del usuario.

2. Análisis de requerimientos de la aplicación.

Razón funcional de las tareas.

Establecer por qué es necesario realizar las tareas. Establecer el flujo de trabajo de las tareas.

3. Evolución del usuario y el trabajo.

De qué forma cambiarán los usuarios el usar el nuevo sistema.

Mejorará su calidad de vida?, Aumentará su productividad con la utilización del nuevo sistema?,...

4. Generación de posibles metáforas.

Aplicar heurísticas de usabilidad durante el diseño.

Aplicar estándares al diseño.



De qué forma se puede adaptar la aplicación a la tarea que representa.

Analizar los tipos de posibles diálogos.

5. Organización de la información y su apariencia.

Composición de la presentación: diseño de pantallas, representación de la información, formato de la realimentación, formato de los mensajes,...

Formato de pantalla: balanceado, simétrico, regularidad y acentuación, agrupamiento y alineamiento, proporción.

Prototipado

1. Generación de prototipos.

Desarrollar prototipos de forma paralela para el estudio de diferentes alternativas de diseño y escoger la más idónea. Realizar esto tan pronto como sea posible para empezar el ciclo iterativo de desarrollo.

Se pueden construir de distintos tipos: maquetas, papel, software, viñetas,...

Evaluación

1. Inspección e indagación.

Evaluación heurística, aplicación de estándares, observación de campo, entrevistas, cuestionarios,...

2. Planificar la medición



Seleccionar evaluadores, definir las medidas para la evaluación heurística (se propone la medida de usabilidad de este trabajo), escoger la escala (también propuesta en este trabajo),...

3. Realización de las medidas.

4. Conclusiones

Analizar los datos y los resultados.

Elaborar los informes pertinentes conteniendo también las recomendaciones para corregir las deficiencias encontradas y proceder a una nueva iteración del ciclo.



3. PRINCIPIOS DE DISEÑO CENTRADOS EN EL USUARIO

Diversos autores, investigadores y empresas han publicado una serie de principios de diseño centrados en el usuario. Entre los más importantes se pueden citar a: IBM, Ben Shneiderman, Jakob Nielsen, Mandel, Simpson, Preece, Dix, SUN,...

El diseño centrado en el usuario nos dice que los diseñadores de interfaz de usuario deben comprender el contexto de uso en el que funcionará la aplicación. Esto significa tener un profundo conocimiento del usuario, un profundo conocimiento del entorno y la realización de un estudio detallado de las tareas del usuario.

Que el diseño esté centrado en el usuario implica que este participará en el proceso de desarrollo de diseño de la interfaz de usuario.

Se presentan a continuación una serie de heurísticas, extraídas fundamentalmente del sitio web de Jacob Nielsen [1] y ampliadas con ejemplos, que se deben aplicar en el desarrollo de la interfaz gráfica de usuario del sistema. Los principios de diseño son fundamentales para diseñar e implementar interfaz gráficas de usuario efectivas y usables.

Las interfaces gráficas de usuario efectivas tienen tal apariencia visual que hacen sentir al usuario que este tiene el control del sistema.



3.1 Heurísticas de Usabilidad

H1. Visibilidad del estado del sistema. La aplicación debe mantener siempre informado al usuario del estado del sistema así como de los caminos que este pueda tomar con una retroalimentación visual apropiada en un tiempo razonable. El sistema ofrecerá al usuario una respuesta que le indique lo que está sucediendo en cada una de las operaciones que realiza. Por ejemplo, en una aplicación que imprime diferentes páginas, mostraría al usuario la actividad que está realizando con la posibilidad de deshacer la operación. En la siguiente ventana, la aplicación informa al usuario del formateado de las páginas y del envío de estas a la impresora con movimiento de dos círculos.

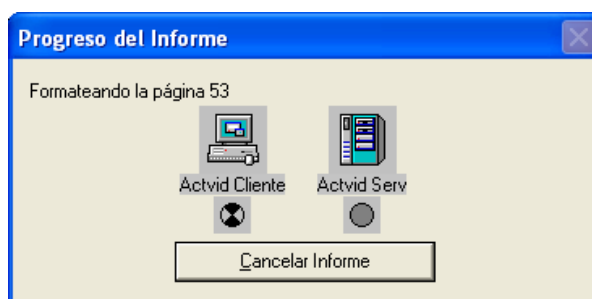


Figura 3.1. Visibilidad del estado del sistema

H2. Control y libertad del usuario. La interfaz debe ser diseñada de tal manera que el control de la interacción con el sistema lo tenga el usuario de manera que interactúe directamente con los objetos de la pantalla. De esta forma, este, se sentirá más cómodo y no se sentirá un módulo más de la aplicación. Esto se consigue cuando el usuario manipula los objetos como si fueran objetos físicos.



Se le proporcionarán salidas de emergencia sin tener que pasar por un montón de diálogos. Se deberá proporcionar también acciones de deshacer y rehacer sin perder el trabajo realizado hasta el momento.



Figura 3.2. Funciones deshacer y rehacer

Se debe tener en cuenta que existen diferentes tipos de usuarios, con diferentes niveles de conocimientos informáticos y percepción de la aplicación. Los desarrolladores deben crear grupos de usuarios con características comunes y para cada uno de los grupos crear un nivel de uso de la aplicación.

Por ejemplo se pueden tener niveles de uso para usuarios noveles, intermedios y avanzados. La presentación de la interfaz será diferente para cada grupo de usuarios.

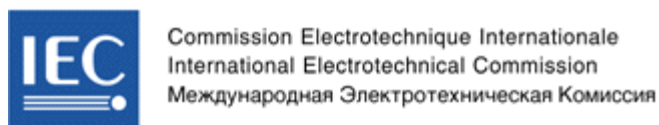
H3. Consistencia y estándares. Una buena interfaz contribuye al aumento de la productividad si es consistente en todos los diálogos que desarrolla, basándose en el conocimiento que el usuario ha adquirido con otras aplicaciones y en la aplicación propia. Se debe mantener la consistencia en todas las aplicaciones relacionadas. Deberán implementar las mismas reglas de diseño para mantener la consistencia en toda la interacción.

El usuario debe ser capaz de saber en cada momento en qué contexto está trabajando, de donde viene y a donde va. Esto se puede realizar con indicadores gráficos como iconos o colores diferentes para cada situación.



Existen dos tipos de estándares: estándares de *iure* y estándares de *facto*.

Los estándares de *iure* son aquellos generados por comités con status legal y avalados por gobiernos o instituciones. Requieren una elaboración y proceso complejo. En el mundo informático, entre los comités más importantes dedicados a la realización de normas están:



Los estándares de *facto* son aquellos que nacen de productos de la industria con mucho éxito en el mercado informático o a partir de trabajos de investigadores y que han tenido una gran difusión.

La utilización de estándares en el diseño de interfaz gráfica de usuario produce una serie de beneficios, como pueden ser:

- Una terminología común: permitiendo a los diseñadores hablar de los mismos conceptos y poder realizar comparativas sobre ellos.
- Una identidad común: resultando sistemas fáciles de reconocer.



- Reducción de la formación: el conocimiento ‘se hereda’ de una sistema a otro con la consiguiente reducción de costes de formación y productividad.
- Mantenimiento y evolución: al tener estructuras y estilos comunes se reducen costes por este concepto.
- Salud y seguridad: los sistemas con controles de estandarización tienen pocos comportamientos no esperados.

En la organización nos debemos asegurar que los estándares son seguidos por los desarrolladores para mantener la consistencia.

H4. Prevención de errores. El mejor tratamiento de los errores es prevenirlos con un buen diseño de los diálogos desde el primer momento en que ocurren, minimizando los riesgos de que puedan ocurrir. Se debe realizar un buen diseño de mensajes de error que den la posibilidad al usuario de retraerse antes de que se realice la acción y se comprometan los datos.

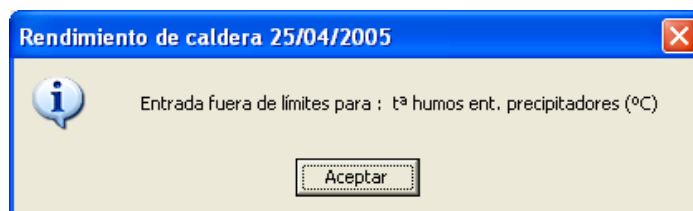


Figura 3.3. Información de error

La ventana del ejemplo anterior está informando al usuario que está introduciendo un valor incorrecto en la celda “*tª humos ent. Precipitadotes (°C)*”



para que lo corrija. El sistema no realizará los cálculos hasta que los datos de entrada sean correctos.

H5. Correspondencia entre el sistema y el mundo real. El sistema debe hablar el lenguaje de los usuarios, con palabras, frases y conceptos familiares para el usuario, siempre en el contexto de la aplicación. Se debe hacer que la información aparezca en un orden lógico y natural.

El usuario no tiene por qué conocer los términos técnicos utilizados en el mundo informático. La aplicación debe interactuar con el usuario de forma que este perciba las palabras y frases cotidianas de la metáfora de la aplicación.

La aplicación debe ser lo más parecida posible al objeto del mundo real que representa.

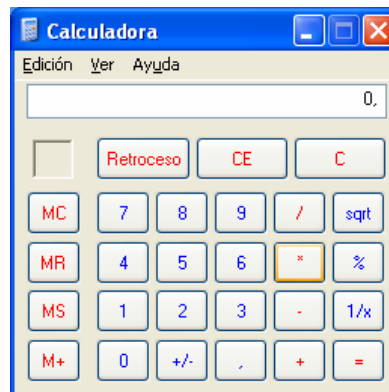


Figura 3.4. Como en el mundo real

El programa de la calculadora del sistema operativo Windows XP es un buen ejemplo de lo expuesto anteriormente. Para el usuario, a la vista de la ventana anterior, es inmediata la realización de cálculos como lo haría con una calculadora real.



H6. Reconocer antes que recordar. Reducir la carga de memoria del usuario para reducir la propensión a errores en su interacción con el sistema. El usuario no debería tener que recordar información desde una parte de un diálogo a otro.

Las instrucciones para usar el sistema deben ser visibles o fácilmente accesibles.

Se deben establecer unos valores por defecto para la aplicación, con la posibilidad de que el usuario pueda especificar sus preferencias. También se debe tener la opción de reinicializar los valores por defecto.

Otra manera de reducir la carga cognitiva del usuario es cuando se utiliza la mnemónica para la realización de algunas acciones, como por ejemplo en las aplicaciones de Microsoft siempre se guarda el trabajo con Ctrl+G y siempre se imprime con Ctrl+P.

Otra forma de reducir la carga cognitiva del usuario es que la interfaz de usuario sea un metáfora del mundo real. Por ejemplo, una aplicación para control del volumen de sonido imitará los controles de una cadena de música:

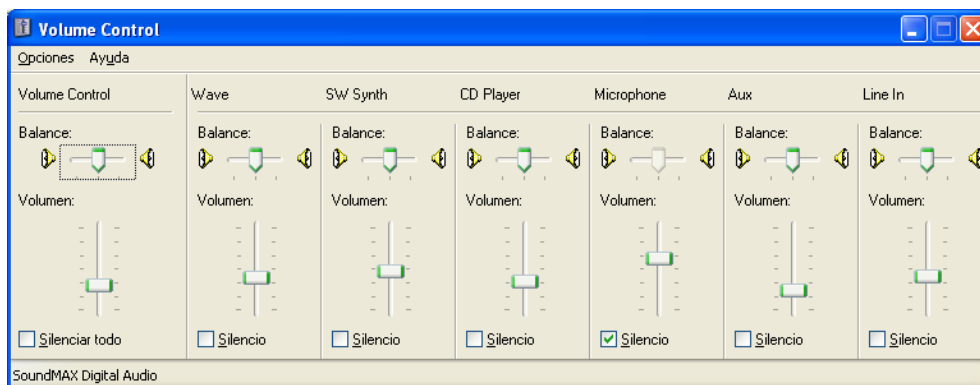


Figura 3.5. Reconocer antes que recordar



H7. Flexibilidad y eficiencia de uso. El sistema se debe diseñar para que lo puedan manejar diferentes tipos de usuarios, en función de su experiencia con la aplicación. De esta manera se aumentará la productividad del usuario y se ganará en usabilidad.

Una innovación muy importante en el desarrollo de interfaz de usuario es proporcionar al sistema técnicas de adaptabilidad. De esta forma la interfaz se adecua de forma automática a las características del usuario. Un sistema adaptativo puede estar basado en la experiencia del usuario con la aplicación o en la observación de las tareas que el usuario realiza repetidamente.

Establecer aceleradores para aumentar la velocidad de la interacción para los usuarios expertos. Esto se puede realizar con pulsaciones de teclas rápidas, iconos,...

El usuario deberá poder adaptar la interfaz a su conveniencia, por ejemplo, tendrá la posibilidad de quitar o añadir iconos, menús o barras de iconos.

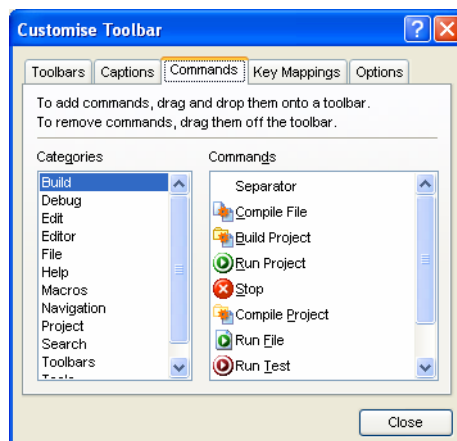


Figura 3.6. Flexibilidad de uso



H8. Estética y diseño minimalista. Los diálogos no deben contener información que sea irrelevante para la tarea que está realizando el usuario. Debe ser una interfaz simple, fácil de aprender y de usar y con fácil acceso a las funcionalidades que ofrece la aplicación. La información extra no necesaria disminuye la visibilidad al usuario causando errores en la interacción y distrayendo al usuario en la realización de la tarea.

H9. Ayudar a los usuarios a reconocer, diagnosticar y recuperación de errores. Los mensajes de error deben estar expresados en lenguaje que el usuario entienda y no con códigos de error, indicando el problema y sugiriendo la solución al problema que causa el error.

El siguiente ejemplo muestra un mensaje de error que no sigue la línea del párrafo anterior. Sólo dice que no es correcto, pero no dice el qué y cómo solucionarlo. Esto es un ejemplo de lo que no se debe hacer.

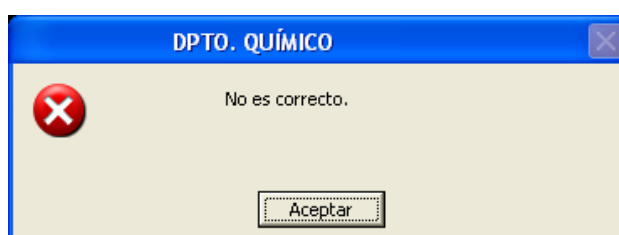


Figura 3.7. Mensaje de error mal formateado, sin información

H10. Ayuda y documentación. El mejor sistema es el que no necesita ningún tipo de documentación, pero de todas formas hay que proporcionar al usuario ayuda y documentación. Esta debe ser fácil de encontrar y enfocada a la tarea



que el usuario realiza. Se deben listar sólo los pasos necesarios para la realización de la tarea.

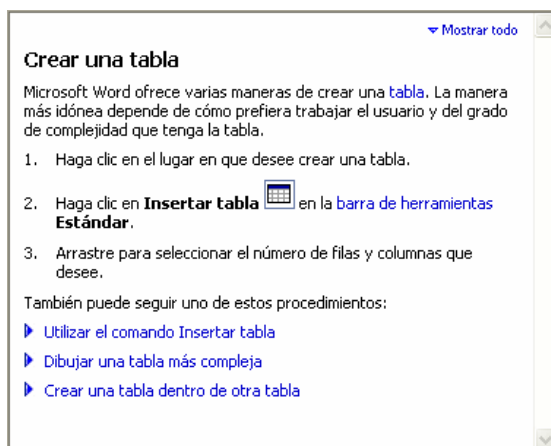


Figura 3.8. Ayuda con pasos concretos y claros

El anterior ejemplo es la ayuda de Microsoft Word para crear una tabla. Como se ve en la imagen, la ayuda está estructurada en pasos muy concretos y claros.



4. MEDICION DE LA INTERFAZ GRAFICA DE USUARIO

La medición de la interfaz de usuario es esencial para asegurar la correcta interacción entre el sistema desarrollado y el usuario final. La interfaz es el medio por el cual el usuario “habla” con el sistema. Una interfaz bien diseñada evita muchos errores en el uso y aumenta la productividad del usuario.

La evaluación tiene como misión principal asegurar que los sistemas se han diseñado teniendo en cuenta las necesidades de los usuarios. La evaluación proporcionará información valiosísima que permite comprobar si la interacción del sistema y el usuario se ha diseñado correctamente, detectando deficiencias y proponiendo mejoras.

4.1 El proceso de evaluación de ISO 9126

El proceso de evaluación según ISO/IEC 9126 es como se describe en el siguiente gráfico. En él se pueden apreciar en la parte derecha las tres fases principales: Definición de Requerimientos de Calidad, Preparación de la Evaluación y Procedimiento de Evaluación.

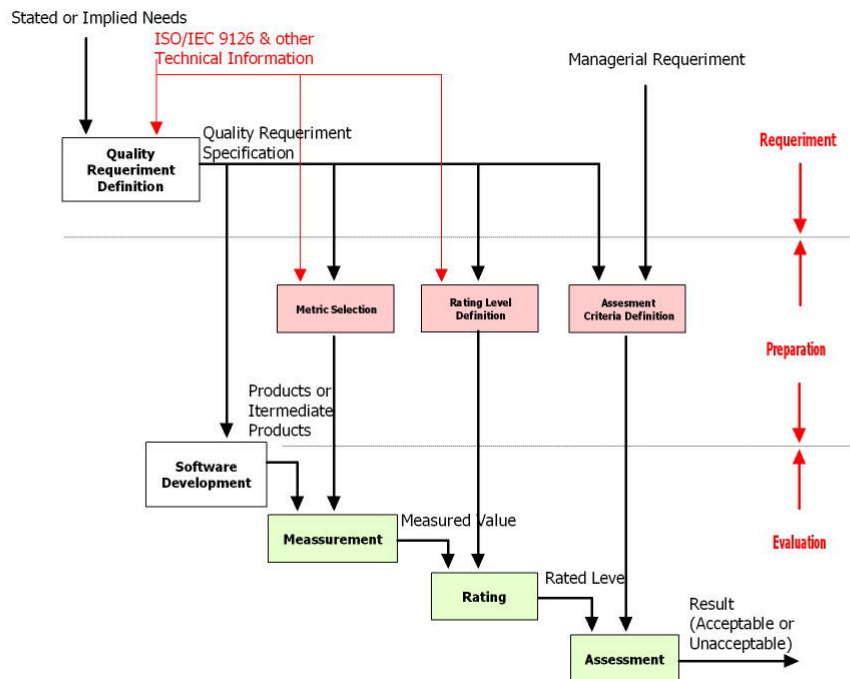


Figura 4.1. Modelo de proceso de evaluación de ISO/IEC 9126

En la fase de “Definición de Requerimientos de Calidad” se especifican los requerimientos en términos de las características y posibles subcaracterísticas prescritas en el estándar y vistas en el punto 2.3. Sirven de entrada al proceso junto con las explícitas e implícitas del usuario. Produce como salida una Especificación de Requerimientos de Calidad.

En la fase de “Preparación de la Evaluación” se especifican los siguientes pasos:

- Selección de las Métricas de Calidad. Los evaluadores deben decidir y seleccionar las métricas que se utilizarán durante la fase de evaluación y estas deben correlacionarse con las características respectivas.
- Definición de los Niveles de Puntuación. Se deben definir rangos de valores que definan los niveles de satisfacción de los requerimientos. El valor de una métrica se lleva sobre una escala que no expresa el nivel



de satisfacción. Por esta razón el estándar ISO indica que se deben establecer rangos de valores.

- Definición de Criterios de Valoración. En esta etapa, los evaluadores deben preparar procedimientos que especifiquen cómo resumir los resultados de las diferentes características.

En la fase de “Procedimiento de Evaluación” se especifican los siguientes pasos:

- Medición. Tiene como entrada el software desarrollado y las métricas seleccionadas. Aquí, se aplican las métricas seleccionadas el producto desarrollado. Como salida se obtiene un valor para la métrica seleccionada.
- Puntuación. A partir del valor dado en el paso anterior se determina la puntuación conseguida en la escala de rangos de valores y niveles de satisfacción.
- Valoración. Tiene como entrada el nivel de puntuación del paso anterior y como salida, el resultado como una declaración de la calidad del producto desarrollado. Con este resultado ya es posible la toma de decisiones sobre el producto.



4.2 Métodos de evaluación

En la literatura sobre el tema se pueden encontrar multitud de métodos para evaluar la usabilidad ya que este concepto es crucial para asegurar un buen diseño de la interfaz de usuario y de la interacción entre el sistema y el usuario final.

La relación de métodos que se presenta está clasificada en función de la participación del usuario en el proceso de evaluación (user testing) [Piattini 2002] o la evaluación es realizada por expertos en usabilidad (usability inspection) [1].

4.2.1 User Testing

Estos métodos se caracterizan por la alta participación del usuario en el proceso de evaluación. Con este tipo de pruebas se identifican problemas de usabilidad que se le presentan al usuario durante el uso del sistema.

Este tipo de pruebas resulta costoso por el hecho de involucrar al usuario y por la utilización de prototipos o producto final. Requiere mucha planificación y preparación.

- SUMI (Software Usability Measurement Inventory). Desarrollado por “*The human factors research group*”, University College Cork, 1990. Este método consiste en un cuestionario que el usuario responde mientras interactúa con el sistema. Tiene tres tipos de respuestas: sí, no, no sabe. Requiere de la participación de muchos usuarios para que el resultado sea representativo.



- MUMMS (Measuring the Usability of Multi-Media Systems). Desarrollado también por “*The human factors research group*”. Parecido al anterior pero para desarrollo multimedia.
- WAMMI (Website Analysis and Measurement Inventory). De los mismos autores que los anteriores métodos, se utiliza para evaluar sitios web. Está basado también en un cuestionario que el usuario rellena según va navegando e interactuando en el sitio web a medir.
- SUS (System Usability Scale). Desarrollado por “User Information Architecture A/D Group”, Digital Equipment Co. Ltd. Es un test para rellenar de manera sencilla y que permite realizar comparaciones cruzadas entre productos.

4.2.2 Usability Inspection

Este tipo de pruebas se diferencia del anterior en la poca o nula participación del usuario final y en la participación de expertos en usabilidad y la consulta de guías de estilo y heurísticas.

- Evaluación heurística. Es el método más informal e involucra a especialistas en usabilidad que examinan si los elementos de los diálogos siguen unos principios de usabilidad establecidos.
- Estimación heurística. Es una variante del anterior en el que los inspectores son preguntados para estimar la usabilidad relativa de dos o más diseños en términos cuantitativos.
- Cognitive walkthrough. Usa un procedimiento explícitamente detallado para simular cómo el usuario resuelve problemas en cada paso a través



del diálogo, chequeando si las metas del usuario simuladas y el contenido de memoria pueden ser asumidos para .

- Pluralistic walkthrough. Utiliza encuentros de grupos donde usuarios, desarrolladores y dan pasos a través de un escenario, discutiendo cada elemento del diálogo.
- Inspección de características. Secuencias de listas de características son usadas para ejecutar tareas típicas, chequear secuencias largas, pasos engorrosos o pesados, pasos que no serían naturales probar por los usuarios y pasos que requieren un gran conocimiento o experiencia en orden a evaluar un conjunto de características propuesto.
- Inspección de consistencia. Diseñadores que de otros proyectos inspeccionan la interfaz para ver si la cosas son como en sus propios diseños.
- Inspección estándar. Un experto sobre una interfaz estándar inspecciona la interfaz para conformidad.
- Inspección de usabilidad formal. Combina inspecciones individuales y en grupo en un procedimiento de seis pasos con papeles estrictamente definidos para cada elemento de una inspección heurística y una forma simplificada de cognitive walkthroughs.

La evaluación heurística, estimación heurística, cognitive walkthrough, inspección de características e inspección estándar se suele realizar con un solo evaluador cada vez. Los otros métodos se realizan por un grupo de evaluadores.



4.2.3 Métodos de mejora de la usabilidad

Se presenta a continuación una tabla [Piattini 2002] que muestra una serie de métodos aplicables en las diferentes fases del ciclo de desarrollo del sistema y que permiten alcanzar mayores cotas de usabilidad y las fases del ciclo en que pueden utilizarse:

<i>Fase del ciclo de desarrollo</i>	<i>Método de mejora</i>
Recopilación de requisitos y análisis	<ul style="list-style-type: none"> - Indagación - Observación (estudio etnográfico) - Tormenta de ideas - Encuestas, cuestionarios, entrevistas - Categorización (por tarjetas)
Diseño e implementación	<ul style="list-style-type: none"> - Prototipado - Técnicas de composición de interfaz - Diseño paralelo - Secuencias de escenarios - Construcción de escenarios - Cuadros de asignación de tareas - Análisis de tareas - Matriz de funcionalidad
Evaluación de la interfaz	<ul style="list-style-type: none"> - Guías de estilo, heurísticas, patrones - Inspecciones formales de usabilidad - Evaluación heurística - Revisiones cognitivas - Guías de comprobación - Cuestionarios, entrevistas - Pensando en alto

Tabla 4.1. Métodos de mejora de la usabilidad

4.2.4 Criterios cuantificables de usabilidad de Constantine & Lockwood

Las métricas [Constantine 1999] que se presentan a continuación y propuestas por Constantine & Lockwood se utilizan para medir la complejidad y la adecuación del diseño de la interfaz de usuario así como la cohesión de los datos. La información dada por estas métricas es útil para hacer una estimación de la facilidad de aprendizaje o de memorización de la interfaz, con



la consiguiente disminución de errores en la utilización del interfaz por parte del usuario.

Essential Efficiency. (Eficiencia esencial). Esta métrica pretende medir la simplicidad de la interfaz. Asume que los casos de uso esenciales expresan el número mínimo de pasos de usuario para una tarea. Los pasos mencionados son los que los usuarios experimentan como acciones discretas, como son seleccionar, mover, entrar, borrar.

$$EE = \frac{S_{essential}}{S_{enacted}} * 100$$

Donde:

$S_{essential}$: número de pasos ideal para completar la tarea del caso de uso.

$S_{enacted}$: número de pasos reales para completar la tarea del caso de uso.

Task Concordance. (Concordancia de tareas). Pretende medir la eficiencia y la simplicidad. Mide cómo las frecuencias esperadas de las tareas se corresponden con su dificultad. Favorece un diseño donde las tareas más frecuentes son hechas fáciles. ¿ Las tareas más frecuentemente ejecutadas son las más fáciles ?.

$$TC = \frac{D}{P} * 100$$

Donde:

$$P = N(N - 1) / 2$$

N = número de pares de tareas totales.



D = número de pares de tareas en orden – número de pares de tareas fuera de orden.

Task Visibility. (Visibilidad de la tarea). La visibilidad de la tarea a realizar. Mide la proporción de objetos del interfaz o elementos necesarios para completar una tarea que son visibles al usuario.

$$TV = \frac{\sum_{\forall i} V_i}{S_{total}} * 100$$

Donde:

V_i = característica de visibilidad para cada paso, que puede ser:

- 0 si es necesaria la llamada al elemento (oculto).
- 0.5 si está disponible pero el elemento necesita ser expuesto o si el contexto de interacción está cambiado (suspendido).
- 1 si el elemento es visible (directo).

S_{total} = número total de pasos necesarios para completar la tarea.

Si todos los pasos son visibles, la puntuación es 100.

Layout Uniformity. (Uniformidad del esquema). Mide la uniformidad y regularidad del esquema de diseño de la interfaz. Cómo de similares son los elementos de la interfaz.



$$LU = \left(1 - \frac{N_h + N_w + N_t + N_l + N_b + N_r}{6 * N_c - M} \right) * 100$$

Donde:

N_h = altura del elemento (height).

N_w = anchura del elemento (weight).

N_t = distancia de alineamiento a arriba (top).

N_l = distancia de alineamiento a la izquierda (left).

N_b = distancia de alineamiento abajo (bottom).

N_r = distancia de alineamiento a la derecha (right).

N_c = número total de elementos en la pantalla.

M = valor de ajuste para que LU esté comprendido entre 0 y 100.

Visual Concordance. (Coherencia visual). Trata de medir la comprensibilidad de la interfaz. Trata de medir si los elementos similares semánticamente están agrupados.

$$VC = \left(\frac{\sum_{\forall k} G_k}{\sum_{\forall k} N_k * \frac{(N_k - 1)}{2}} \right) * 100$$

$$G_k = \sum_{\forall i, j \neq i} R_{i,j}$$



Donde:

N_k = número de elementos en una agrupación de elementos k .

$R_{i,j}$ = valor entre 0 y 1 asociado con la relación semántica entre los elementos i y j en la agrupación k .

4.2.5 Características de un método de evaluación

Constantine y Loockwood proponen una serie de características [Piattini 2002] que debe cumplir un método de evaluación para que sea aplicable de manera práctica:

- Ser fácil de interpretar y de calcular.

El método propuesto tiene una fácil interpretación de los resultados y sus cálculos se realizan de forma sencilla, simplemente marcando un 1 en la casilla correspondiente. Después se realiza una media ponderada.

- Poder aplicarse a diferentes fases del desarrollo, a prototipos y a modelos.

El método propuesto se puede aplicar en cualquier momento del desarrollo del producto.

- Tener una base conceptual simple, junto con un análisis razonado fuerte. Conceptualmente es sencillo y fácil de entender.
- Tener el suficiente grado de detalle y la posibilidad de discriminar entre diseños.

Cada heurística del método está detallada en un conjunto de elementos.

- Dar soporte a actividades de diseño.



Cuando se termina la evaluación con el método propuesto, la información obtenida sugiere dónde se encuentran los problemas de usabilidad y dónde se puede mejorar el diseño.

- Permitir una predicción eficiente de la usabilidad.

Es el objetivo del método, cuantificar la usabilidad.

- Permitir indicar la calidad relativa de los diseños realizados.

El método suministra un valor que es comparable entre diferentes diseños de interfaces de usuario.



5. MEDIDA DE LA USABILIDAD

5.1 Metodología

El centro de mayor importancia de este trabajo es la metodología propuesta en este punto para hallar una medida de usabilidad que permita decidir si una aplicación es fácil de usar.

El método es una evaluación heurística basado en las heurísticas de Jakob Nielsen. Cada heurística se compone de una serie de elementos directamente verificables. Estos elementos se puntuarán con un porcentaje en el intervalo de 0 a 100, en función de su grado de cumplimiento en la aplicación a medir. Con la expresión matemática del punto 5.1.1.1 se obtendrá su valor para la aplicación a examen. La métrica final se obtendrá al realizar la media ponderada de los valores obtenidos. Aplicando la medida a una escala de valores se podrá afirmar su mayor o menor grado de cumplimiento de las heurísticas.

El método también devuelve un valor para cada heurística para poder estudiar cada una por separado y recomendar su mejora con consejos concretos.

Como anexo al proyecto se han desarrollado dos herramientas para facilitar el cálculo de la medida de usabilidad: 1) Aplicación desarrollada en Java, 2) Libro de Microsoft Excel. Los puntos que siguen, están referidos a la hoja de cálculo mencionada.

Los puntos a seguir para hallar un valor de usabilidad son los siguientes:

1. Ejecutar a la vez la aplicación a medir y la hoja de cálculo.



2. Para cada heurística, y dentro de esta para cada fila, marcar con un “1” el grado de certeza que consideremos más conveniente, siendo “0%” el total desacuerdo y “100%” el total acuerdo. La última columna de la hoja marca “ERROR” si hay más de una marca o ninguna. En el caso de no ser aplicable a la aplicación a medir, marcar con un “1” la columna “N/A”.
3. Después de rellenar cada heurística, en la hoja “Resumen” están los resultados como aplicación del algoritmo de cálculo del punto 5.1.2.
4. En la columna “Total” de la mencionada hoja “Resumen” están los valores en una escala de 0 a 100 para poder comparar entre heurísticas. La columna “Total ponderado” es la cifra anterior ponderada por su peso. La medida de la usabilidad es la media simple de los valores “Total ponderado”.

5.1.1 Algoritmo de cálculo para cada heurística

Para cada heurística, el algoritmo a seguir es el siguiente:

```

t = 0
para cada k = 1 hasta 5
    sk = 0
    para cada i = 1 hasta n
        sk = sk + vi
    fin para
    t = t + sk * pk
fin para
    
```

Siendo:



k: el número de grados de certeza

i: el número de fila

n: el número total de filas

s: la suma de unos, {1}, de cada columna

t: total de puntuación de la heurística en función del grado de certeza de cada fila

v: valor de cada cuadro. Se escribe un 1 en el lugar correspondiente en función del grado de certeza de la fila.

P: grados de certeza: {0, 25, 50, 75, 100}

5.1.1.1 Expresión matemática

Para cada heurística se puede considerar:

$$t = \sum_{k=1}^5 \left[\sum_{i=1}^n v_i \right] p_k$$



5.2 Medida de las observaciones

<i>Aplicación</i>	<i>Visibilidad del estado del sistema</i>	<i>Control y libertad del usuario</i>	<i>Consistencia y estándares</i>	<i>Prevención de errores</i>	<i>Correspondencia entre el sistema y el mundo real</i>	<i>Reconocer antes que recordar</i>	<i>Flexibilidad y eficiencia de uso</i>	<i>Estética y diseño minimalista</i>	<i>Ayuda a los usuarios a reconocer, ...</i>	<i>Ayuda y documentación</i>	<i>Media</i>
2Microsoft Word	94,53	83,33	93,75	76,09	96,87	81,45	72,50	88,16	94,56	82,41	86,37
AutoCAD LT	84,38	84,87	91,87	75,00	92,86	76,67	72,73	93,75	92,05	94,64	85,88
SAP R3	82,50	75,00	90,00	81,25	86,11	72,22	96,00	100,00	92,71	80,36	85,61
NetBeans	92,19	75,64	94,44	83,33	93,33	78,79	63,16	97,37	88,64	82,14	84,90
Lotus Notes	79,55	75,00	94,39	80,68	93,33	81,25	73,53	98,68	85,23	81,25	84,29
Microsoft Outlook	85,61	68,75	96,23	79,76	91,67	72,58	77,94	94,74	95,46	78,57	84,13
Windows Draw	89,06	67,57	95,41	86,67	98,21	67,86	76,47	92,11	81,82	82,14	83,73
Microsoft Excel	93,75	85,29	97,87	75,00	80,56	75,00	75,00	81,58	84,78	83,33	83,22
Windows Media Player	88,46	73,08	87,50	84,62	92,86	75,00	69,64	98,08	81,82	78,57	82,96
Gel	92,24	69,70	85,71	96,43	100,00	79,17	75,00	94,74	78,57	50,89	82,24
WS_FTP Pro	85,35	69,12	90,48	92,50	97,92	75,89	61,54	94,74	84,52	68,27	82,03
HTML-Kit	87,10	74,32	86,77	67,11	93,33	78,33	82,35	93,42	79,55	78,57	82,08
CoffeeCup HTML Editor	93,94	74,29	88,33	78,57	91,07	81,88	63,89	93,42	72,73	76,79	81,49
OpenOffice Calc	86,29	72,37	88,54	69,05	96,67	77,68	65,00	89,47	80,68	80,36	80,61
Nero Smart	92,31	55,00	85,12	100,00	94,23	68,27	57,14	93,42	82,61	81,48	80,96
WinZip	85,83	60,00	95,35	77,78	83,93	66,67	80,00	93,42	75,00	72,12	79,01
Minitab	90,91	74,34	91,18	62,50	82,35	62,12	72,06	88,16	86,36	74,11	78,41
VirusScan Enterprise	95,37	50,83	84,88	68,42	91,07	74,17	57,81	92,11	89,47	76,92	78,11
MSN Messenger	75,83	55,88	95,92	59,09	90,00	71,55	75,00	94,74	80,44	71,00	76,94
WinRar	88,79	66,94	86,36	80,00	92,31	63,33	76,92	86,84	82,14	45,19	76,88
Norton Antivirus	78,45	47,22	89,67	72,06	87,50	67,24	61,54	86,84	85,71	78,85	75,51
WinDVD	84,00	44,57	83,72	83,33	91,67	68,52	50,00	98,68	76,14	74,00	75,46
GEMA	84,17	36,21	89,09	59,52	78,33	72,86	71,25	90,00	78,75	92,86	75,30
Renta 2005	85,29	57,69	87,74	68,48	76,47	78,79	59,52	86,84	75,00	68,75	74,46
FTP Commander Pro	69,44	43,55	80,68	75,00	97,73	70,19	54,55	90,79	65,48	47,12	69,45
Steam Pro	53,23	47,66	79,35	65,00	91,67	80,47	58,82	69,74	72,73	69,64	68,83
Gestión Medioambiental CT Aboño	81,45	38,28	73,26	76,32	86,67	66,67	37,50	94,74	57,96	18,75	63,16
ArgoUML	61,29	48,39	81,67	41,18	78,33	64,06	59,72	97,22	53,75	7,69	59,33
SIO Sist. de información de operación	56,82	37,50	71,67	53,75	70,00	46,30	62,50	97,37	47,22	36,61	57,97
Sistema Gestión de Planos	45,69	37,93	75,58	40,48	92,31	46,77	35,29	85,53	53,75	7,69	52,10
ORCA	33,33	22,92	51,35	27,63	56,82	26,00	28,57	71,88	58,75	32,00	40,92



La tabla anterior muestra las observaciones realizadas según la expresión del punto anterior y ordenadas de mayor a menor. La muestra está realizada sobre las 31 primeras aplicaciones de la tabla del punto 5.5.1.

5.3 Criterios de asignación de pesos a las heurísticas

Para la medida última de usabilidad se ha dado peso a cada heurística considerando que no todas tienen la misma importancia.

Se han considerado cuatro criterios para hallar los pesos definitivos y poder aplicarlos a las medias de las observaciones calculadas con la expresión del punto 5.1.1.1.

5.3.1 1^{er} criterio

Para el primer criterio se ha utilizado la mayor o menor cantidad de elementos verificables de cada heurística según la expresión:

$$w_h = \frac{\sum_{i=1}^n v_i}{\sum_{h=1}^{10} \sum_{i=1}^n v_i} ; \quad \text{siendo } \sum_{h=1}^{10} w_h = 1$$

Los pesos obtenidos según este criterio son:

Num. (k)	Peso (w _i)	HEURISTICAS
1	0,104	Visibilidad del estado del sistema
2	0,129	Control y libertad del usuario
3	0,186	Consistencia y estándares
4	0,082	Prevención de errores
5	0,066	Correspondencia entre el sistema y el mundo real
6	0,117	Reconocer antes que recordar
7	0,082	Flexibilidad y eficiencia de uso
8	0,066	Estética y diseño minimalista
9	0,079	Ayuda a los usuarios a reconocer, diagnosticar...
10	0,088	Ayuda y documentación

Tabla 5.2 Pesos del 1^{er} criterio



5.3.2 2º criterio

Para la aplicación de este segundo criterio se han seleccionado aplicaciones subjetivamente consideradas en una primera estimación entre buenas y malas.

Estas aplicaciones son las siguientes:

Aplicación	Función	Fabricante	Versión	Año
Microsoft Word	Editor de texto	Microsoft	2002	2001
Microsoft Excel	Hoja de cálculo	Microsoft	2002	2001
ORCA	Gestión de análisis de combustibles	propio		1997
Lotus Notes	Ciente de correo y coordinación	IBM	6.5.1	2004
SAP R3	Gestión integral empresa	SAP AG	4.6C	2001
NetBeans	Entorno de desarrollo	Sun Microsystems	5	2006
Coffee Cup HTML Editor	Desarrollo web	Coffee Software	9.6	2004
SIO Sistema de información operación	Gráficos de variables de planta	propio		2003
Renta 2005	Cálculo del IRPF y patrimonio	Agencia Tributaria	1.0	2005
AutoCad LT	Diseño CAD mecánico	AutoDesk	2000i	2000

Tabla 5.3 Mezcla de aplicaciones buenas y malas

Aplicando un sistema de 10 ecuaciones con 10 incógnitas y las medias como los términos independientes:

$$\left\{ \begin{array}{l} 94,53x_1 + 83,33x_2 + 93,75x_3 + 76,09x_4 + 86,87x_5 + 81,45x_6 + 72,50x_7 + 88,16x_8 + 94,56x_9 + 82,41x_{10} = 86,37 \\ 93,75x_1 + 85,29x_2 + 97,87x_3 + 75,00x_4 + 80,56x_5 + 75,00x_6 + 51,47x_7 + 81,58x_8 + 84,78x_9 + 83,33x_{10} = 83,22 \\ 33,33x_1 + 22,92x_2 + 51,35x_3 + 27,63x_4 + 56,82x_5 + 26,00x_6 + 28,57x_7 + 71,76x_8 + 58,75x_9 + 32,00x_{10} = 40,92 \\ 79,54x_1 + 75,00x_2 + 94,39x_3 + 80,68x_4 + 93,33x_5 + 81,25x_6 + 73,53x_7 + 98,68x_8 + 85,23x_9 + 82,25x_{10} = 84,29 \\ 82,50x_1 + 75,00x_2 + 90,00x_3 + 81,25x_4 + 86,11x_5 + 72,22x_6 + 96,00x_7 + 100,00x_8 + 92,71x_9 + 80,36x_{10} = 85,61 \\ 92,19x_1 + 75,64x_2 + 94,44x_3 + 83,33x_4 + 93,33x_5 + 78,79x_6 + 63,16x_7 + 97,16x_8 + 88,64x_9 + 82,14x_{10} = 84,90 \\ 93,94x_1 + 74,29x_2 + 88,33x_3 + 78,57x_4 + 91,07x_5 + 81,90x_6 + 63,89x_7 + 93,42x_8 + 72,73x_9 + 76,79x_{10} = 81,49 \\ 56,82x_1 + 37,50x_2 + 71,67x_3 + 53,75x_4 + 70,00x_5 + 46,30x_6 + 62,50x_7 + 97,37x_8 + 47,22x_9 + 36,61x_{10} = 57,97 \\ 85,29x_1 + 57,69x_2 + 87,74x_3 + 68,48x_4 + 76,47x_5 + 78,79x_6 + 59,52x_7 + 86,84x_8 + 75,00x_9 + 68,75x_{10} = 74,46 \\ x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} = 1 \end{array} \right.$$

con las condiciones siguientes:

$$x_1 > 0; x_2 > 0; x_3 > 0; x_4 > 0; x_5 > 0; x_6 > 0; x_7 > 0; x_8 > 0; x_9 > 0; x_{10} > 0$$



Los pesos obtenidos son los siguientes:

Num. (k)	Peso (w _i)	HEURISTICAS
1	0,098	Visibilidad del estado del sistema
2	0,144	Control y libertad del usuario
3	0,120	Consistencia y estándares
4	0,075	Prevención de errores
5	0,059	Correspondencia entre el sistema y el mundo real
6	0,094	Reconocer antes que recordar
7	0,089	Flexibilidad y eficiencia de uso
8	0,124	Estética y diseño minimalista
9	0,095	Ayuda a los usuarios a reconocer, diagnosticar...
10	0,102	Ayuda y documentación

Tabla 5.4 Pesos del 2º criterio

5.3.3 3^{er} Criterio

Para el tercer criterio se ha formado un sistema de 10 ecuaciones con 10 incógnitas como el del punto anterior pero con las 10 mejores aplicaciones de de la tabla de observaciones del punto 5.2, obteniéndose los siguientes resultados:

Num. (k)	Peso (w _i)	HEURISTICAS
1	0,104	Visibilidad del estado del sistema
2	0,102	Control y libertad del usuario
3	0,101	Consistencia y estándares
4	0,101	Prevención de errores
5	0,101	Correspondencia entre el sistema y el mundo real
6	0,099	Reconocer antes que recordar
7	0,099	Flexibilidad y eficiencia de uso
8	0,098	Estética y diseño minimalista
9	0,098	Ayuda a los usuarios a reconocer, diagnosticar...
10	0,097	Ayuda y documentación

Tabla 5.5 Pesos del 3er criterio



Como se puede apreciar en la tabla anterior, los pesos son bastante parecidos entre ellos debido a que se ha realizado con aplicaciones consideradas buenas.

5.3.4 4º Criterio

Para este cuarto criterio se han tenido en cuenta las aplicaciones de la tabla de observaciones que han obtenido una media mayor de 80 por ser considerada esta medida como aceptable en la escala del punto 5.7. Estas aplicaciones son las siguientes:

<i>Aplicación</i>	<i>Media</i>
Microsoft Word	86,37
AutoCAD LT	85,88
SAP R3	85,61
NetBeans	84,90
Lotus Notes	84,29
Microsoft Outlook	84,13
Windows Draw	83,73
Microsoft Excel	83,22
Windows Media Player	82,96
Gel	82,24
WS_FTP Pro	82,03
HTML-Kit	82,08
CoffeeCup HTML Editor	81,49
OpenOffice Calc	80,61
Nero Smart	80,96

Tabla 5.6 Aplicaciones con media > 80

Los pesos se han obtenido midiendo la contribución de cada heurística a su media:

$$w_i = \frac{1}{n} \sum_{i=1}^n \left(\frac{h_i}{\sum_{k=1}^{10} h_k} \right)$$



donde:

w_i los pesos a hallar.

n el número de aplicaciones, en este caso 16.

k el número de heurísticas, 10.

h_k el valor de la observación de cada heurística.

Los pesos obtenidos son los siguientes:

Num. (k)	Peso (w_i)	HEURISTICAS
1	0,106	Visibilidad del estado del sistema
2	0,088	Control y libertad del usuario
3	0,109	Consistencia y estándares
4	0,098	Prevención de errores
5	0,112	Correspondencia entre el sistema y el mundo real
6	0,091	Reconocer antes que recordar
7	0,086	Flexibilidad y eficiencia de uso
8	0,112	Estética y diseño minimalista
9	0,102	Ayuda a los usuarios a reconocer, diagnosticar...
10	0,094	Ayuda y documentación

Tabla 5.7 Pesos del 4º criterio

5.3.5 Resumen de los pesos y su media

	1 ^{er} criterio Por cantidad de valoraciones	2º criterio Aplicaciones escogidas buenas y malas	3 ^{er} criterio Las 10 mejores aplicaciones	4º criterio Contribución de las medias > 80	Media
Visibilidad del estado del sistema	0,104	0,098	0,104	0,106	0,103
Control y libertad del usuario	0,129	0,144	0,102	0,088	0,116
Consistencia y estándares	0,186	0,120	0,101	0,109	0,129
Prevención de errores	0,082	0,075	0,101	0,098	0,089
Correspondencia entre el sistema y el ...	0,066	0,059	0,101	0,112	0,085
Reconocer antes que recordar	0,117	0,094	0,099	0,091	0,100
Flexibilidad y eficiencia de uso	0,082	0,089	0,099	0,086	0,089
Estética y diseño minimalista	0,066	0,124	0,098	0,112	0,100
Ayuda a los usuarios a reconocer, diag....	0,079	0,095	0,098	0,102	0,093
Ayuda y documentación	0,088	0,102	0,097	0,094	0,095

Tabla 5.8 Resumen de asignación de pesos



Los pesos definitivos obtenidos son los datos en rojo de la última columna, obtenidos como la media de los pesos según los cuatro criterios.

En el gráfico siguiente se puede apreciar el suavizado de la media en aplicación de los cuatro criterios anteriores.

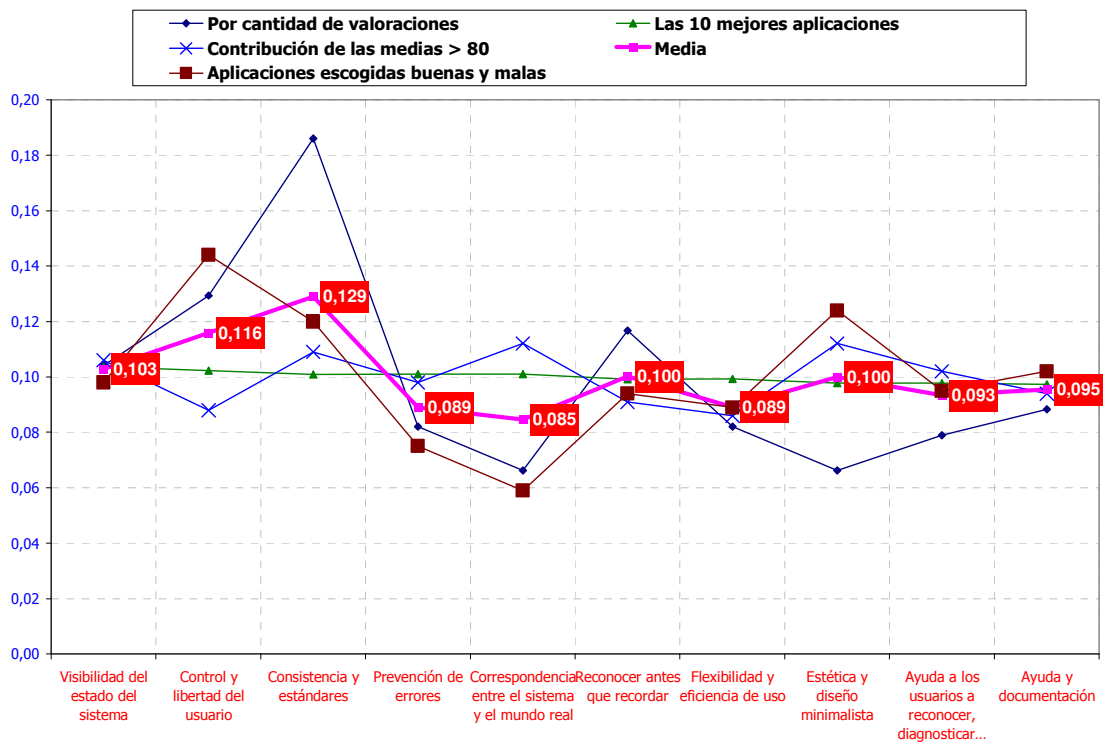


Figura 5.1 Los cuatro criterios y su media



5.4 Expresión matemática de la métrica de usabilidad

Como resultado de la obtención de los pesos y pasando estos a números enteros para un manejo más fácil y dividiendo por 10000 para la entrada en la escala de 0 a 10, se ha obtenido la siguiente expresión como métrica de la usabilidad:

$$u = \frac{103h_1 + 116h_2 + 129h_3 + 89h_4 + 85h_5 + 100h_6 + 89h_7 + 100h_8 + 94h_9 + 95h_{10}}{10000}$$

Siendo h_i la medida de las observaciones de cada heurística.

Se ha realizado una regresión lineal simple con la media de las observaciones como variable explicativa de la métrica obteniéndose una recta de regresión $y = 0.04025 + 0.09977x$ con un coeficiente de determinación de 100% y una desviación típica de 0.0218.

Una buena aproximación para aplicar en un primer vistazo a una observación entre 0 y 100 puede ser:

$$u = \frac{\text{observación}}{10}$$

5.5 Aplicación de la métrica a todas las aplicaciones

Después de proponer la métrica se ha aplicado esta a la totalidad de las aplicaciones del punto 5.8.1, obteniéndose los valores de la siguiente tabla:



<i>Aplicación</i>	<i>H1</i>	<i>H2</i>	<i>H3</i>	<i>H4</i>	<i>H5</i>	<i>H6</i>	<i>H7</i>	<i>H8</i>	<i>H9</i>	<i>H10</i>	<i>Med.</i>	<i>Métr.</i>
Microsoft Word	94,53	83,33	93,75	76,09	96,87	81,45	72,50	88,16	94,56	82,41	86,37	8,663
AutoCAD LT	84,38	84,87	91,87	75,00	92,86	76,67	72,73	93,75	92,05	94,64	85,88	8,611
SAP R3	82,50	75,00	90,00	81,25	86,11	72,22	96,00	100,0	92,71	80,36	85,61	8,547
NetBeans	92,19	75,64	94,44	83,33	93,33	78,79	63,16	97,37	88,64	82,14	84,90	8,517
Oracle jDeveloper	91,67	67,57	93,75	97,50	94,23	84,48	63,24	100,0	73,81	83,65	84,99	8,502
PartitionMagic	94,83	58,93	88,64	93,75	100,0	85,71	73,33	94,74	85,23	77,68	85,28	8,484
Eclipse	90,91	73,61	92,22	81,25	92,31	85,83	77,94	96,05	76,14	79,81	84,61	8,474
Lotus Notes	79,55	75,00	94,39	80,68	93,33	81,25	73,53	98,68	85,23	81,25	84,29	8,445
Microsoft Outlook	85,61	68,75	96,23	79,76	91,67	72,58	77,94	94,74	95,46	78,57	84,13	8,420
Microsoft Excel	93,75	85,29	97,87	75,00	80,56	75,00	75,00	81,58	84,78	83,33	83,22	8,392
Windows Draw	89,06	67,57	95,41	86,67	98,21	67,86	76,47	92,11	81,82	82,14	83,73	8,368
Adobe Photoshop	89,84	75,64	89,13	85,53	96,43	73,28	76,47	96,25	77,27	73,21	83,31	8,331
Windows Media Player	88,46	73,08	87,50	84,62	92,86	75,00	69,64	98,08	81,82	78,57	82,96	8,296
HTML-Kit	87,10	74,32	86,77	67,11	93,33	78,33	82,35	93,42	79,55	78,57	82,08	8,214
Gel	92,24	69,70	85,71	96,43	100,0	79,17	75,00	94,74	78,57	50,89	82,24	8,201
WS_FTP Pro	85,35	69,12	90,48	92,50	97,92	75,89	61,54	94,74	84,52	68,27	82,03	8,201
CoffeeCup HTML Editor	93,94	74,29	88,33	78,57	91,07	81,88	63,89	93,42	72,73	76,79	81,49	8,177
OpenOffice Calc	86,29	72,37	88,54	69,05	96,67	77,68	65,00	89,47	80,68	80,36	80,61	8,078
Nero Smart	92,31	55,00	85,12	100,0	94,23	68,27	57,14	93,42	82,61	81,48	80,96	8,054
MySQL Query Builder	85,48	57,26	92,71	72,83	98,33	82,81	72,62	96,05	82,95	52,88	79,39	7,942
WinZip	85,83	60,00	95,35	77,78	83,93	66,67	80,00	93,42	75,00	72,12	79,01	7,919
Minitab	90,91	74,34	91,18	62,50	82,35	62,12	72,06	88,16	86,36	74,11	78,41	7,891
VirusScan Enterprise	95,37	50,83	84,88	68,42	91,07	74,17	57,81	92,11	89,47	76,92	78,11	7,799
ModelMaker	90,63	60,42	90,22	70,45	92,31	72,58	78,13	92,11	69,32	61,61	77,78	7,789
MSN Messenger	75,83	55,88	95,92	59,09	90,00	71,55	75,00	94,74	80,44	71,00	76,94	7,719
WinRAR	88,79	66,94	86,36	80,00	92,31	63,33	76,92	86,84	82,14	45,19	76,88	7,690
Norton Antivirus	78,45	47,22	89,67	72,06	87,50	67,24	61,54	86,84	85,71	78,85	75,51	7,541
PHP Designer 2005	89,84	61,36	85,71	68,18	96,43	85,00	78,95	89,47	71,59	25,96	75,25	7,536
WinDVD	84,00	44,57	83,72	83,33	91,67	68,52	50,00	98,68	76,14	74,00	75,46	7,519
GEMA	84,17	36,21	89,09	59,52	78,33	72,86	71,25	90,00	78,75	92,86	75,30	7,517
Renta 2005	85,29	57,69	87,74	68,48	76,47	78,79	59,52	86,84	75,00	68,75	74,46	7,483
MySQL Administrator	82,76	41,94	91,49	66,67	96,43	76,67	62,50	94,74	76,19	52,88	74,23	7,421
Camedia MASTER	90,00	34,82	75,00	68,18	93,75	86,36	46,15	90,63	75,00	69,00	72,89	7,243
FTP Commander Pro	69,44	43,55	80,68	75,00	97,73	70,19	54,55	90,79	65,48	47,12	69,45	6,918
Steam Pro	53,23	47,66	79,35	65,00	91,67	80,47	58,82	69,74	72,73	69,64	68,83	6,853
Gestión Medioambiental CT Aboño	81,45	38,28	73,26	76,32	86,67	66,67	37,50	94,74	57,96	18,75	63,16	6,315
ArgoUML	61,29	48,39	81,67	41,18	78,33	64,06	59,72	97,22	53,75	7,69	59,33	6,001
SIO Sist. de información de operación	56,82	37,50	71,67	53,75	70,00	46,30	62,50	97,37	47,22	36,61	57,97	5,803
Sistema Gestión de Planos	45,69	37,93	75,58	40,48	92,31	46,77	35,29	85,53	53,75	7,69	52,10	5,246
ORCA	33,33	22,92	51,35	27,63	56,82	26,00	28,57	71,88	58,75	32,00	40,92	4,090

Tabla 5.9 Tabla definitiva de medición de métricas



5.6 Heurísticas detalladas de usabilidad de Xerox

Las heurísticas mostradas en el punto 3.1 se conocen como Heurísticas de Jakob Nielsen. Son de sobra conocidas en el campo de la usabilidad.

La empresa Xerox, en el año 1995, para cada una de ellas las detalló en una serie de elementos directamente aplicables al examen de las aplicaciones. El método propuesto en este trabajo se basa en el grado de certeza de cada uno de los elementos. En el apéndice 1 se muestran los checklist originales de Xerox.

En este trabajo se han actualizado las listas de Xerox, apéndice 2: se han modificado algunos, eliminados otros y añadidos otros nuevos (en rojo) con el fin de adaptarlo a las aplicaciones actualmente en producción.

5.7 Establecimiento de la escala de medida

Xerox estableció la siguiente escala de cinco puntos para sus evaluaciones de usabilidad. En función de esta escala se establecerá la escala de trabajo para el método propuesto.

1 **Cosmético**: valores de 8 a 10. No afectará a la usabilidad del sistema. Existen errores que no impactan en el diseño, pero si en su apreciación.

2 **Menor**: valores de 6 a 8. Los usuarios pueden fácilmente trabajar con algunos problemas.

3 **Medio**: valores de 4 a 6. Los usuarios tropiezan con problemas, pero se adaptan rápidamente a ellos.



4 **Mayor**: valores de 2 a 4. Los usuarios tienen dificultades, pero son capaces de encontrar la forma de trabajar a pesar de ellas. Puede poner en peligro el éxito de la tarea.

5 **Catastrófico**: valores de 0 a 2. Los usuarios son incapaces de hacer su trabajo. Imposible terminar la tarea con éxito.

Teniendo en cuenta que si se coloca un 1 en todos los grados de certeza del 100% en los elementos a medir se obtiene como máximo de medida de usabilidad un 10, se puede establecer la escala entre 0 y 10 como se muestra en la siguiente figura:

Num. (k)	Peso (w _i)	HEURISTICAS	Grado de certeza							Normalizado para el gráfico						Total	
			0%	25%	50%	75%	100%	N/A	Σ1	Σ2	0%	25%	50%	75%	100%		N/A
1	103	Visibilidad del estado del sistema	0	0	0	0	34	0	34	34	0,00	0,00	0,00	0,00	###	0,00	100,000
2	116	Control y libertad del usuario	0	0	0	0	41	0	41	41	0,00	0,00	0,00	0,00	###	0,00	100,000
3	129	Consistencia y estándares	0	0	0	0	59	0	59	59	0,00	0,00	0,00	0,00	###	0,00	100,000
4	89	Prevención de errores	0	0	0	0	26	0	26	26	0,00	0,00	0,00	0,00	###	0,00	100,000
5	85	Correspondencia entre el sistema y el mundo real	0	0	0	0	21	0	21	21	0,00	0,00	0,00	0,00	###	0,00	100,000
6	100	Reconocer antes que recordar	0	0	0	0	37	0	37	37	0,00	0,00	0,00	0,00	###	0,00	100,000
7	89	Flexibilidad y eficiencia de uso	0	0	0	0	26	0	26	26	0,00	0,00	0,00	0,00	###	0,00	100,000
8	100	Estética y diseño minimalista	0	0	0	0	21	0	21	21	0,00	0,00	0,00	0,00	###	0,00	100,000
9	94	Ayuda a los usuarios a reconocer, diagnosticar...	0	0	0	0	25	0	25	25	0,00	0,00	0,00	0,00	###	0,00	100,000
10	95	Ayuda y documentación	0	0	0	0	28	0	28	28	0,00	0,00	0,00	0,00	###	0,00	100,000
Medida de usabilidad																10	

Tabla 5.10 Medida máxima de usabilidad



Se propone la siguiente escala:

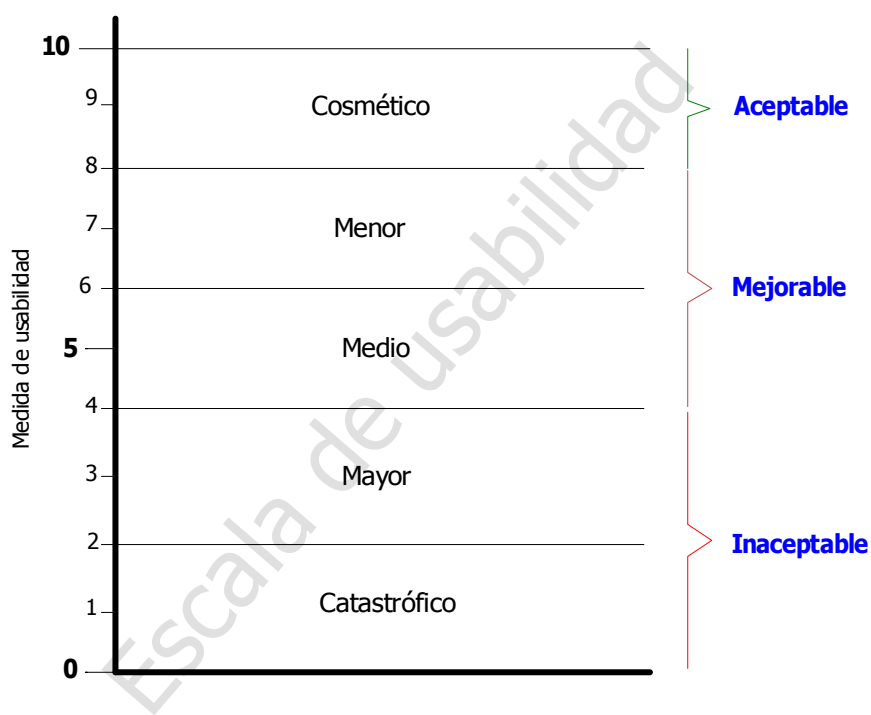


Figura 5.2 Escala de usabilidad

El significado de los tres puntos de la escala es el siguiente:

- Aceptable [8 – 10]: la interfaz de usuario es aceptable y no necesita mejorar para la versión actual. En una nueva versión se podría intentar su posible mejora.
- Mejorable [4 – 8]: la interfaz de usuario hay que mejorarla para su liberación dentro de la versión actual. Se debe mejorar hasta que su medida resulte aceptable.
- Inaceptable [0 – 4]: la interfaz de usuario es del todo inaceptable. Se debería comenzar totalmente de nuevo su diseño.



5.8 Aplicación de la medida a aplicaciones reales

5.8.1 Selección de aplicaciones a medir

Se han escogido las aplicaciones que aparecen en la siguiente tabla. Unas son aplicaciones de sobra conocidas, otras desarrolladas internamente en una empresa y otras desarrolladas por encargo.

Se ha intentado escoger aplicaciones de diferentes campos de actuación como puedan ser: edición de texto, hoja de cálculo, antivirus, entorno de programación, comunicaciones ftp, reproductor de video,...

Nº	Aplicación	Función	Fabricante	Versión	Año
1	Microsoft Word	Editor de texto	Microsoft	2002 SP3	2001
2	Microsoft Excel	Hoja de cálculo	Microsoft	2002 SP3	2001
3	OpenOffice SreetSheet	Hoja de cálculo	Comunidad OpenOffice.org	1.1.4	2004
4	ORCA	Gestión de análisis de carbón sobre Microsoft Access	Propia		1997
5	GEMA	Gestión de mantenimiento sobre Oracle Developer	Thales (a medida)		
6	Windows Draw	Gráficos vectoriales y dibujo	Micrografx	6 LE	1998
7	Windows Media Player	Reproductor y creador de archivos multimedia. Radio y video	Microsoft	10	2004
8	WinDVD	Reproductor de video	InterVideo	4.0	2003
9	MiniTab	Estadística	MiniTab, Inc	14.12.0	2004
10	Lotus Notes	Correo, internet, planificación, coordinación	IBM	6.5.1	2004
11	Nero Start Smart	Grabación CD - DVD	Ahead Software, Inc	6.3.0.3	2003
12	SAP R3	Gestión integral de la empresa	SAP AG	4.6C	2001
13	NetBeans	Entorno de desarrollo java	Sun Microsystems, Inc	5	2006
14	Gel	Entorno de desarrollo java	Gexpert, Inc	1.0.0.0RC 28b	2003
15	VirusScan Enterprise	Antivirus	McAfee Security	7.0.0	2003
16	Norton Antivirus	Antivirus	Simantec	10.0.13.3	2004
17	Microsoft Outlook	Correo y planificación	Microsoft	2002	2001
18	Gestión Medioambiental de CT Aboño	Sistema de control y conexión de analizadores ambientales	Ingenieros Asesores		
19	WinRar	Compresor/descompresor	Alexander Roshal	3.41	2004
20	WinZip	Compresor/descompresor	WinZip Computing, Inc	10	2005
21	WS_FTP Pro	Comunicaciones por ftp	IPSWITCH	5.0 XP	2003
22	FTP Commander Pro	Comunicaciones por ftp	InternetSoft	7.8	2004



			Corporation		
23	Renta 2005	IRPF y patrimonio	Agencia Tributaria	1.0	2005
24	HTML-Kit	Desarrollo web	Chami	292	2005
25	CoffeeCup HTML Editor	Desarrollo web	Coffee Software	9.6	2004
26	SIO Sistema de información de la operación	Gráficos de variables de proceso	A medida		2003
27	ArgoUML	Diagramas UML	Universidad de California	0.18.1	2005
28	Steam Pro	Diseño y simulación de plantas térmicas de producción de energía	Thermoflow	15	2004
29	MSN Messenger	Mensajería instantánea	Microsoft	7.0	2005
30	Sistema Gestión de Planos	Gestión de planos	propio	2.5.10b	2005
31	AutoCAD LT	Diseño CAD	AutoDesk	2000i	2000
32	Camedia MASTER	Gestión de fotografías	Olimpos Optical	4.10	2004
33	PartitionMagic	Particionado de discos	PowerQuest	8.0	2002
34	Adobe Photoshop	Edición de fotografías	Adobe	8.0.1	2005
35	ModelMaker	Diseño en UML	ModelMaker Tools BV	6.20	2006
36	Oracle jDeveloper	Desarrollo en java	Oracle	10g	2006
37	Eclipse	Desarrollo en java	Fundación Eclipse	3.1.1	2005
38	MySQL Administrator	Administración de base de datos	MySQL AB	1.1.9	2005
39	MySQL Query Builder	Editor de scrips para bases de datos	MySQL AB	1.1.20	2005
40	PHP Designer 2005	Entorno de desarrollo en PHP	MPSOFTWARE	3.0.6	2005

Tabla 5.11 Selección de aplicaciones a medir

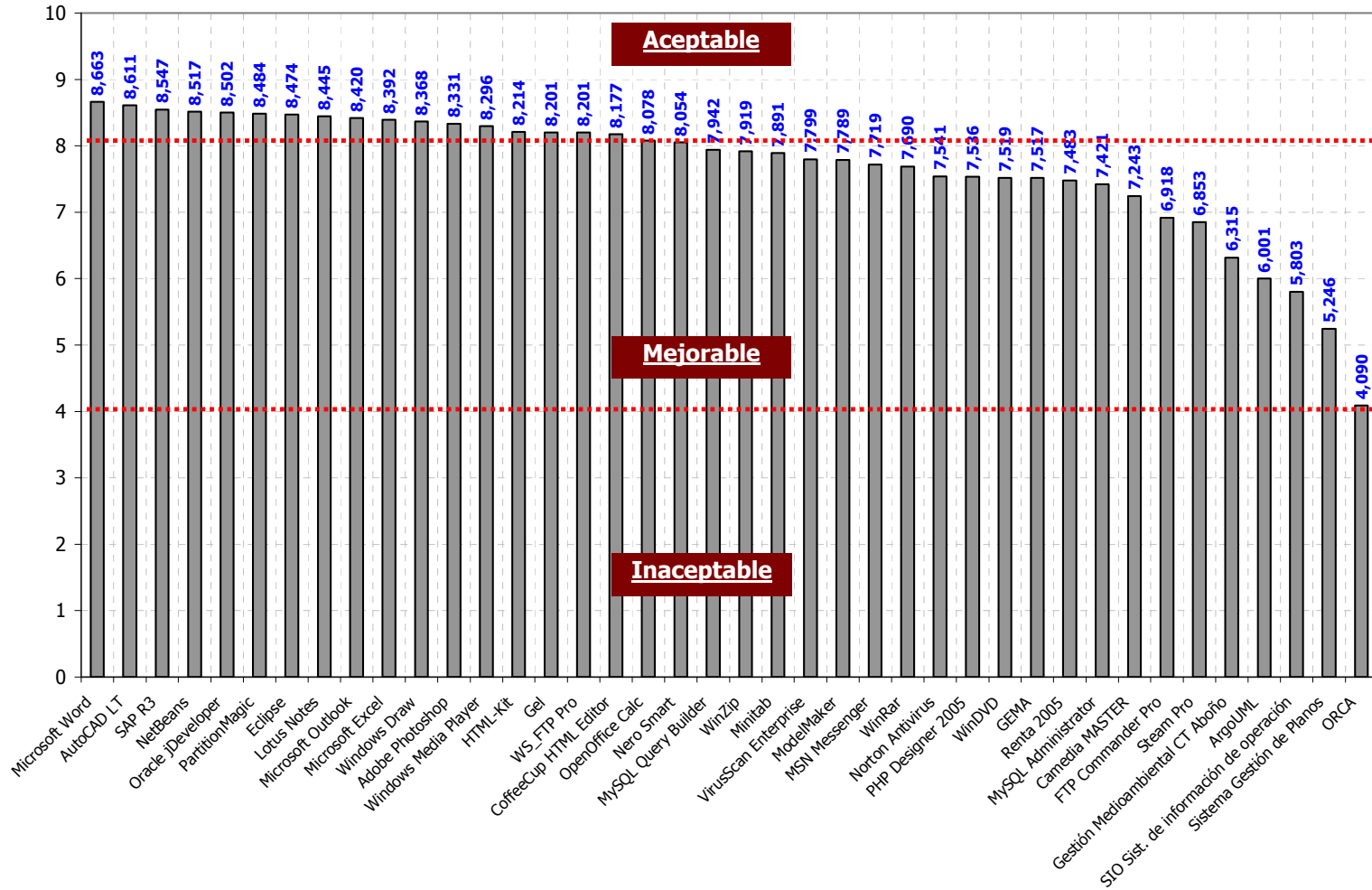
5.8.2 Resultados de las medidas realizadas

En la siguiente tabla se pueden ver los resultados de las medidas efectuadas con la métrica propuesta a las aplicaciones de la tabla anterior. Están ordenas descendentemente por valor de la métrica.

Las medidas obtenidas para cada heurística están referidas a los grados de certeza reales sin introducir los datos de los valores de la columna N/A (no aplicable).



Métricas





5.9 Aplicación de la métrica al resto de aplicaciones

Se miden con la métrica propuesta las aplicaciones de la lista que no se utilizaron en su desarrollo mostrando los resultados en el gráfico siguiente:

Figura

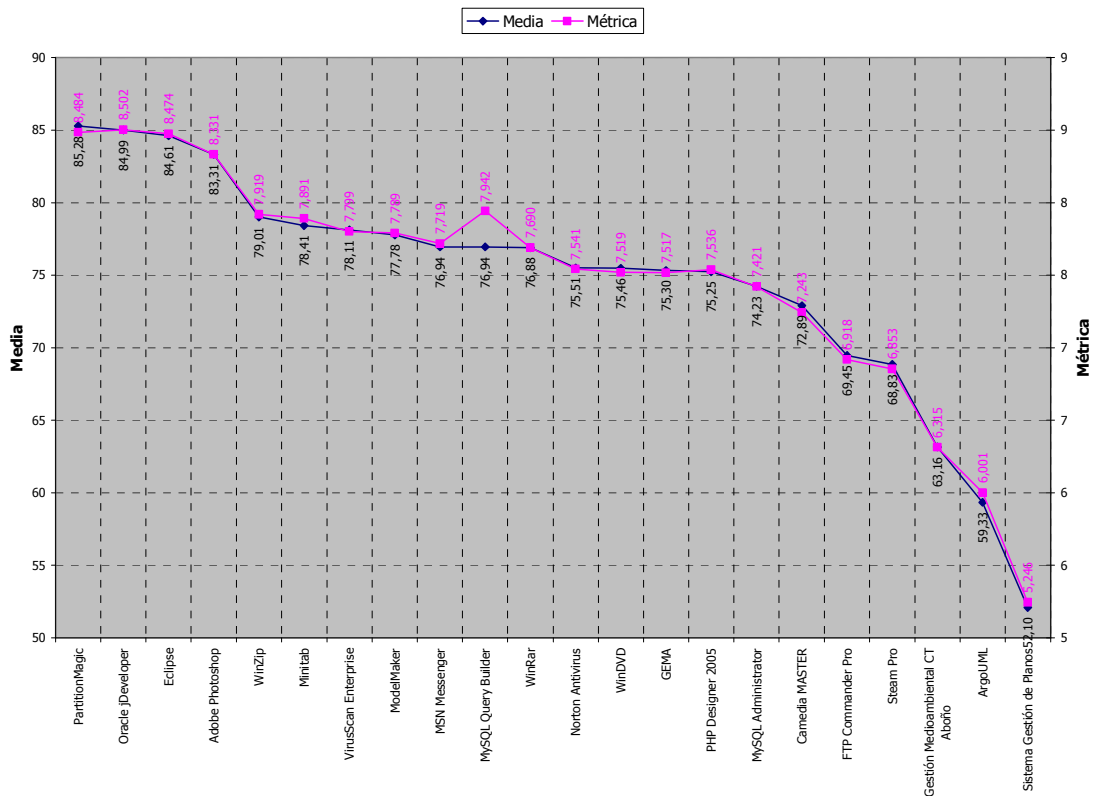


Figura 5.3 Gráfico de resultados de la aplicación de la métrica

Se puede apreciar en el gráfico la fuerte correlación entre la media de las observaciones y la métrica. Concretamente la correlación de Pearson es de 0,997 y su P-Valor de 0,000.



5.10 Estudio de los resultados

Como se puede apreciar en el gráfico anterior, hay un grupo de aplicaciones con una métrica bastante alta, todas por encima de 8,300. Estas aplicaciones son:

- Microsoft Word
- Microsoft Excel
- Windows Draw LE
- PartitionMagic
- jDeveloper
- Lotus Notes
- NetBeans
- Eclipse
- Adobe Photoshop
- Microsoft Outlook
- Autocad LT

El factor común de todas ellas es que están fabricadas por empresas que dedican una parte muy importante de sus recursos a cuidar la usabilidad de sus aplicaciones. Son empresas como: Microsoft, Micrografx, IBM, Sun Microsystems, PowerQuest, Oracle, Adobe y Autodesk.

Un caso especial es Eclipse. Esta aplicación empezó desarrollándose por un conjunto de empresas muy conocidas y actualmente se desarrolla como Open Source.

El porcentaje de aplicaciones con una métrica superior a 8 es del 48% e inferior a 8 es del 52%:

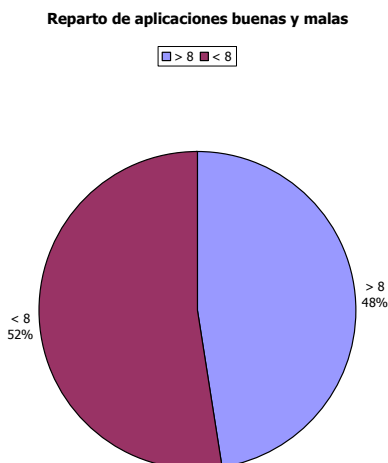


Figura 5.4 Reparto de aplicaciones buenas y malas

La aplicación ORCA está desarrollada bajo Microsoft Access y tiene un valor de la métrica inaceptable. Fue desarrollada por personas que casi nada tienen que ver con el desarrollo de aplicaciones, pero que asombrosamente siguen funcionando en una empresa con un sistema de calidad establecido y certificado.

Hay una serie de aplicaciones con una puntuación cercana al 8 que con pequeñas modificaciones y correcciones serían aceptables. Para su mejora habría que estudiar la puntuación para cada heurística en su gráfico y hacer las correcciones necesarias.

Hay dos aplicaciones, Gestión Medioambiental y Steam Pro, con una puntuación baja, pero que están desarrolladas por empresas de reconocido prestigio en su entorno tecnológico pero no en el desarrollo informático. Estas dos aplicaciones funcionan muy bien internamente pero deben mejorar su usabilidad para alcanzar el nivel de aceptables.

Existe una aplicación de código abierto, ArgoUML, que debe su baja puntuación a la total falta de ayuda.



Hay también otros dos sistemas con puntuación muy baja, SIO y Sistema de Gestión de Planos. Estas aplicaciones están actualmente funcionando en una empresa. Internamente funcionan aceptablemente pero su interfaz de usuario deja mucho que desear.

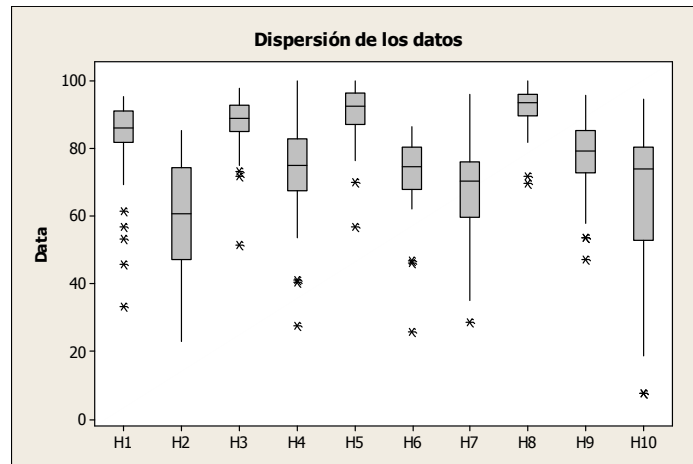


Figura 5.5 Dispersión de las observaciones

Estadísticamente, se puede apreciar que la mayoría de los datos están localizados entre el primer y tercer cuartil, quedando muy pocos datos fuera del rango intercuartílico.

La heurística que presenta una mayor dispersión es “Ayuda y documentación”.

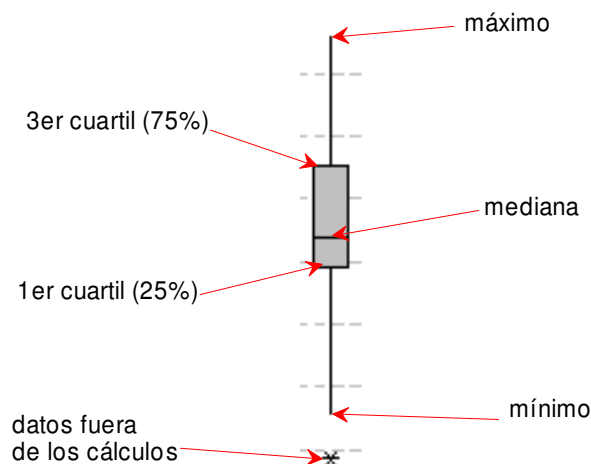


Figura 5.5 Explicación del gráfico boxplot



Media	Desviación típica	Coficiente de variación
82,58	14,00	16,96
60,12	15,81	26,30
87,07	8,65	9,94
73,52	15,17	20,64
90,16	8,74	9,69
72,51	11,81	16,28
66,31	13,55	20,44
91,97	6,47	7,04
77,55	11,25	14,51
65,74	22,27	33,88

Tabla 5.13 Desviación típica y variabilidad de las observaciones

La tabla de resultados de la figura 5.12 queda de forma resumida de la siguiente forma:

Nº Heurística	1	2	3	4	5	6	7	8	9	10
Nº de valores < 80 y > 40	8	31	6	26	5	29	35	2	21	23
Nº de valores < de 40	1	6	0	1	0	1	3	0	0	6
Nº de valores > 80	31	3	34	13	35	10	2	38	19	11
% de valores entre 80 y 40	20,00	77,50	15,00	65,00	12,50	72,50	87,50	5,00	52,50	57,50
% de valores < 40	2,50	15,00	0,00	2,50	0,00	2,50	7,50	0,00	0,00	15,00
% de valores > 80	77,50	7,50	85,00	32,50	87,50	25,00	5,00	95,00	47,50	27,50

Tabla 5.14 Datos y porcentajes resumidos

En cuanto a las heurísticas se puede apreciar en siguiente gráfico el tratamiento dado por los fabricantes de las aplicaciones en su conjunto a cada una de ellas.

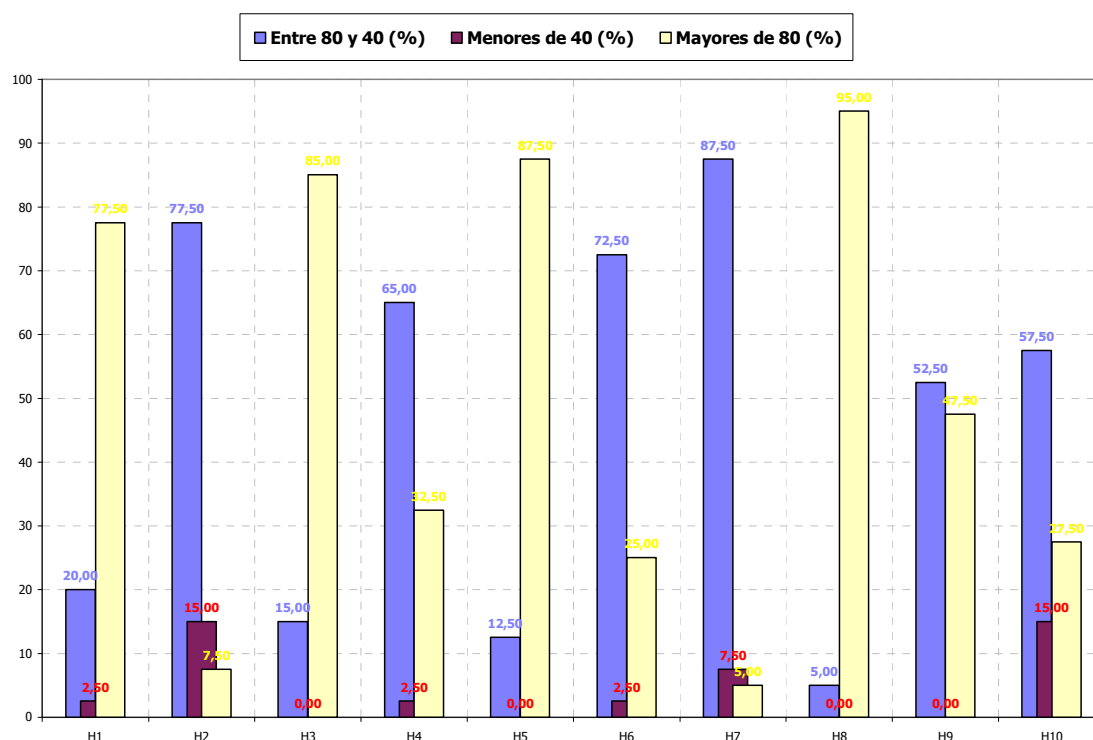


Figura 5.7 Relación de los porcentajes

Las heurísticas que mejor se cumplen por tener una media más alta y desviación típica más reducida y que también queda reflejado en el gráfico anterior son:

- Visibilidad del estado del sistema
- Consistencia y estándares
- Correspondencia entre el sistema y el mundo real
- Estética y diseño minimalista

La heurística peor parada en el estudio es “Control y libertad del usuario”, ya que tiene un porcentaje de valores menores de 40 del 16,16% conjuntamente con un porcentaje de valores entre 40 y 80 del 74,19%. En esta



heurística fallan todas las aplicaciones menos tres: Microsoft Word, Microsoft Excel y Autocad LT.

Otra heurística con fallo general es “Flexibilidad y eficiencia de uso”. Suele ser debido a la falta de observación en las aplicaciones de diferentes niveles de usuarios y también a la falta de elección de botones. En esta heurística en cambio destacan aplicaciones como SAP R3, HTML Kit y WinZip.

La heurística “Ayuda y documentación” también falla de forma general, con un porcentaje de valores menores de 40 del 16,13% y por tanto una variabilidad relativa muy alta de 35.60 debido a que varias aplicaciones carecen totalmente de ayuda de algún tipo. Destacan en cambio con una puntuación muy alta las aplicaciones GEMA y Autocad LT.

También es de destacar en el gráfico de resultados que ninguna aplicación llega a la puntuación de 9 posiblemente debido a la ley de rendimientos decrecientes, ya que cuanto más refinadas están las aplicaciones más difícil es conseguir un aumento de la usabilidad. Las mayores mejoras se pueden conseguir cuanto más baja sea su métrica.

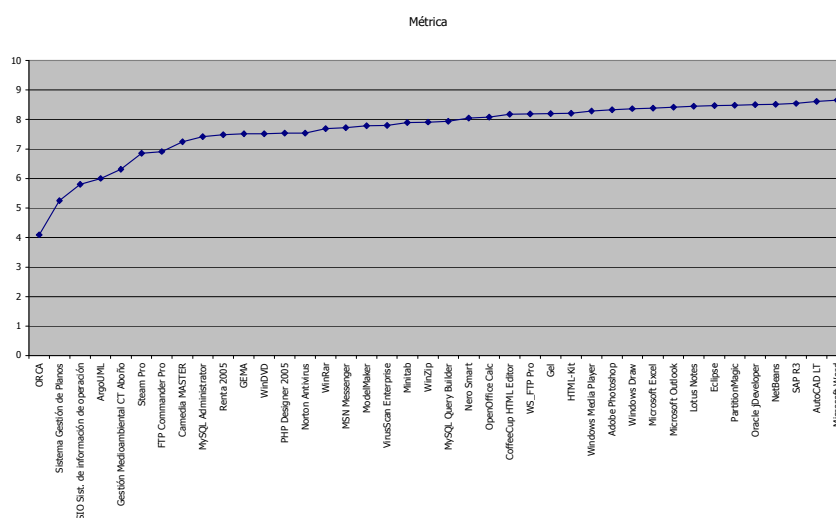


Figura 5.7 Rendimientos decrecientes



Con los datos de las observaciones se realiza una tabla con las correlaciones de Pearson existentes entre las diferentes heurísticas:

	H1	H2	H3	H4	H5	H6	H7	H8	H9
H2	0,629								
H3	0,719	0,727							
H4	0,752	0,553	0,575						
H5	0,566	0,422	0,535	0,666					
H6	0,737	0,516	0,657	0,624	0,698				
H7	0,565	0,712	0,746	0,424	0,322	0,485			
H8	0,485	0,292	0,439	0,490	0,379	0,350	0,365		
H9	0,663	0,655	0,694	0,558	0,441	0,532	0,570	0,158	
H10	0,594	0,553	0,621	0,547	0,256	0,507	0,483	0,161	0,784

Tabla 5.15 Correlaciones de Pearson entre observaciones

En la tabla de correlaciones se pueden apreciar algunas correlaciones medianamente fuertes resaltadas en color rojo. La correlación más fuerte se da entre “Ayuda a los usuarios a reconocer, diagnosticar y recuperarse desde errores” y “Ayuda y documentación”, lo cual es bastante lógico teniendo en cuenta el contenido de los elementos verificables de sus heurísticas. La heurística que parece tener más relación con las demás es “Visibilidad del estado del sistema” ya que es la que presenta más cantidad de correlaciones mayores de 0,7.

Las tres primeras heurísticas concentran la mayor cantidad de correlaciones grandes confirmándose en ellas su mayor peso en el cálculo de la métrica.

La heurística “Estética y diseño minimalista” parece tener poco relación con todas las demás ya que ninguna de sus correlaciones con las demás pasa de 0,5.



5.11 Análisis de las heurísticas del apéndice 3

Los porcentajes en que las aplicaciones sobrepasan la medida de 80 de la escala propuesta son los siguientes:

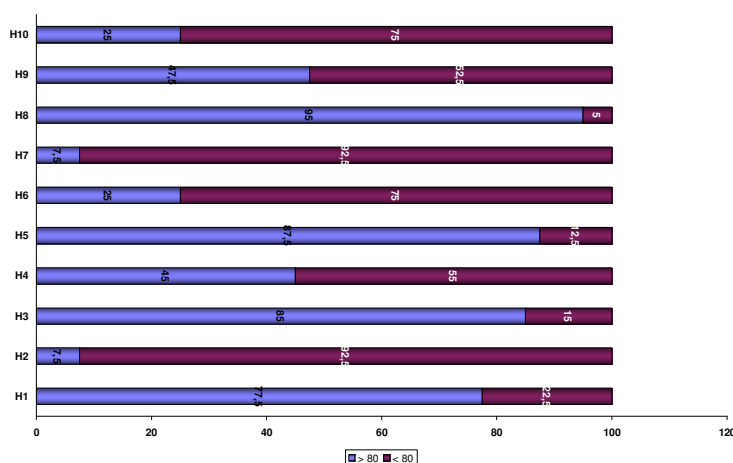


Figura 5.8 Resultados globales

H1. Visibilidad del estado del sistema.

Tiene un 77,5% de aplicaciones que superan el 80 en la escala propuesta, frente a un 22,5% de aplicaciones que no lo superan. Se puede decir que en general si se cumple esta heurística.

H2. Control y libertad del usuario.

Esta es la heurística en la que menos aplicaciones llegan a la puntuación de 80. Concretamente sólo pasan esta puntuación el 7,5%, frente al 92,5% de aplicaciones que no lo hacen. En general se puede decir que esta heurística casi no se cumple. Todas la empresas deben mejorar esta heurística.

H3. Consistencia y estándares

Se cumple en un 85% frente a un 15% que no la cumplen. En general si se cumple. Esta heurística la cumplen todas la empresas grandes.



H4. Prevención de errores

Esta heurística se cumple aproximadamente a partes iguales. Debe notarse que hay compañías grandes que no llegan a la puntuación de 80.

H5. Correspondencia entre el sistema y el mundo real

Se puede decir que se cumple en general, con un 87,5% mayores de 80 y un 12,5% menores.

H6. Reconocer antes que recordar

Esta heurística se puede decir que, en general, no se cumple ya que el 75% no pasa de 80. Sólo cuatro empresas grandes sobrepasan esta medida..

H7. Flexibilidad y eficiencia de uso

En esta heurística sólo sobrepasa la medida de 80 una empresa de las consideradas grandes. Con un 92,5% de aplicaciones por debajo de 80 y un 7,8% por encima es la heurística peor tratada por todas las empresas.

H8. Estética y diseño minimalista

Esta es la heurística mejor tratada por todas las compañías con un 95% de aplicaciones que sobrepasan la medida de 80. Sólo dos aplicaciones están por debajo de 80, pero no muy lejos.

H9. Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de errores.

Esta heurística se cumple aproximadamente a partes iguales. Siendo las que sobrepasan claramente algunas empresas grandes.

H10. Ayuda y documentación

Esta heurística no se cumple en general. Un 75% no pasa de la medida de 80. Dos aplicaciones sobrepasan con claridad esta medida, AutoCAD LT y GEMA.



6. CONCLUSIONES

A la vista de los resultados obtenidos como consecuencia de la aplicación de la métrica, se puede concluir que la métrica propuesta es válida para su aplicación en el desarrollo de sistemas interactivos y cuantificar la usabilidad de estos sistemas de una forma más económica que las típicas evaluaciones heurísticas donde se emplea a diferentes evaluadores y usuarios. Esta métrica realiza la medida de la usabilidad a un nivel de detalle que desmenuza las heurísticas en elementos directamente verificables, valorando su inclusión con porcentajes de 0 a 100%.

En función de los resultados se puede deducir que las grandes compañías producen mejores aplicaciones informáticas como consecuencia, seguramente, de la aplicación de más recursos a su desarrollo, que por otra parte resulta bastante lógico.

Como norma general es necesario mejorar en las heurísticas: "Control y libertad del usuario", "Flexibilidad y eficiencia de uso" y "Ayuda y documentación". Estas heurísticas parecen ser las más difíciles de cumplir en el desarrollo de las aplicaciones.

Las heurísticas que mejor se cumplen son: "Estética y diseño minimalista", "Correspondencia entre el sistema y el mundo real" y



“Consistencia y estándares”. Resulta obvio que estas heurísticas son las más fáciles de cumplir por las organizaciones.

Como trabajo futuro de este proyecto se podría realizar una aplicación informática que automatice lo máximo posible la medida con la métrica propuesta, ya que la realización a mano lleva aproximadamente unos cuarenta y cinco minutos de media para cada aplicación, en función de la experiencia en la aplicación de la métrica y de la experiencia en la aplicación a medir.



7. APENDICE 1

Checklist originales de la compañía Xerox.

Visibility of System Status

The system should always keep user informed about what is going on, through appropriate feedback within reasonable time.

#	Review Checklist	Yes	No	N/A	Comments
1.1	Does every display begin with a title or header that describes screen contents?	0	0	0	
1.2	Is there a consistent icon design scheme and stylistic treatment across the system?	0	0	0	
1.3	Is a single, selected icon clearly visible when surrounded by unselected icons?	0	0	0	
1.4	Do menu instructions, prompts, and error messages appear in the same place(s) on each menu?	0	0	0	
1.5	In multipage data entry screens, is each page <input type="checkbox"/> abelled to show its relation to others?	0	0	0	
1.6	If oertype and insert mode are both available, is there a visible indication of which one the user is in?	0	0	0	
1.7	If pop-up windows are used to display error messages, do they allow the user to see the field in error?	0	0	0	
1.8	Is there some form of system feedback for every operator action?	0	0	0	
1.9	After the user completes an action (or group of actions), does the feedback indicate that the next group of actions can be started?	0	0	0	
1.10	Is there visual feedback in menus or dialog boxes about which choices are selectable?	0	0	0	
1.11	Is there visual feedback in menus or dialog boxes about which choice the cursor is on now?	0	0	0	
1.12	If multiple options can be selected in a menu or dialog box, is there visual feedback about which options are already selected?	0	0	0	
1.13	Is there visual feedback when objects are selected or moved?	0	0	0	
1.14	Is the current status of an icon clearly indicated?	0	0	0	
1.15	Is there feedback when function keys are pressed?	0	0	0	
1.16	If there are observable delays (greater than fifteen seconds) in the system's response time, is the user kept informed of the system's progress?	0	0	0	
1.17	Are response times appropriate to the task?	0	0	0	
1.18	Typing, cursor motion, mouse selection: 50-1 50 milliseconds	0	0	0	
1.19	Simple, frequent tasks: less than 1 second	0	0	0	
1.20	Common tasks: 2-4 seconds	0	0	0	



1.21	Complex tasks: 8-12 seconds	0	0	0	
1.22	Are response times appropriate to the user's cognitive processing?	0	0	0	
1.23	Continuity of thinking is required and information must be remembered throughout several responses: less than two seconds.	0	0	0	
1.24	High levels of concentration aren't necessary and remembering information is not required: two to fifteen seconds.	0	0	0	
1.25	Is the menu-naming terminology consistent with the user's task domain?	0	0	0	
1.26	Does the system provide <i>visibility</i> : that is, by looking, can the user tell the state of the system and the alternatives for action?	0	0	0	
1.27	Do GUI menus make obvious which item has been selected?	0	0	0	
1.28	Do GUI menus make obvious whether deselection is possible?	0	0	0	
1.29	If users must navigate between multiple screens, does the system use context labels, menu maps, and place markers as navigational aids?	0	0	0	



Match Between System and the Real World

The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

#	Review Checklist	Yes	No	N/A	Comments
2.1	Are icons concrete and familiar?	0	0	0	
2.2	Are menu choices ordered in the most logical way, given the user, the item names, and the task variables?	0	0	0	
2.3	If there is a natural sequence to menu choices, has it been used?	0	0	0	
2.4	Do related and interdependent fields appear on the same screen?	0	0	0	
2.5	If shape is used as a visual cue, does it match cultural conventions?	0	0	0	
2.6	Do the selected colors correspond to common expectations about color codes?	0	0	0	
2.7	When prompts imply a necessary action, are the words in the message consistent with that action?	0	0	0	
2.8	Do keystroke references in prompts match actual key names?	0	0	0	
2.9	On data entry screens, are tasks described in terminology familiar to users?	0	0	0	
2.10	Are field-level prompts provided for data entry screens?				
2.11	For question and answer interfaces, are questions stated in clear, simple language?	0	0	0	
2.12	Do menu choices fit logically into categories that have readily understood meanings?	0	0	0	
2.13	Are menu titles parallel grammatically?	0	0	0	
2.14	Does the command language employ user jargon and avoid computer jargon?	0	0	0	
2.15	Are command names specific rather than general?	0	0	0	
2.16	Does the command language allow both full names and abbreviations?	0	0	0	
2.17	Are input data codes meaningful?	0	0	0	
2.18	Have uncommon letter sequences been avoided whenever possible?	0	0	0	
2.19	Does the system automatically enter leading or trailing spaces to align decimal points?	0	0	0	
2.20	Does the system automatically enter a dollar sign and decimal for monetary entries?	0	0	0	
2.21	Does the system automatically enter commas in numeric values greater than 9999?	0	0	0	
2.22	Do GUI menus offer activation: that is, make obvious how to say "now do it"?	0	0	0	
2.23	Has the system been designed so that keys with similar names do not perform opposite (and potentially dangerous) actions?	0	0	0	
2.24	Are function keys <input type="checkbox"/> labelled clearly and distinctively, even if this means breaking consistency rules?	0	0	0	



User Control and Freedom

Users should be free to select and sequence tasks (when appropriate), rather than having the system do this for them. Users often choose system functions by mistake and will need a clearly marked “emergency exit” to leave the unwanted state without having to go through an extended dialogue. Users should make their own decisions (with clear information) regarding the costs of exiting current work. The system should support undo and redo.

#	Review Checklist	Yes	No	N/A	Comments
3.1	If setting up windows is a low-frequency task, is it particularly easy to remember?	0	0	0	
3.2	In systems that use overlapping windows, is it easy for users to rearrange windows on the screen?	0	0	0	
3.3	In systems that use overlapping windows, is it easy for users to switch between windows?	0	0	0	
3.4	When a user’s task is complete, does the system wait for a signal from the user before processing?	0	0	0	
3.5	Can users type-ahead in a system with many nested menus?	0	0	0	
3.6	Are users prompted to confirm commands that have drastic, destructive consequences?	0	0	0	
3.7	Is there an “undo” function at the level of a single action, a data entry, and a complete group of actions?	0	0	0	
3.8	Can users cancel out of operations in progress?	0	0	0	
3.9	Are character edits allowed in commands?	0	0	0	
3.10	Can users reduce data entry time by copying and modifying existing data?	0	0	0	
3.11	Are character edits allowed in data entry fields?	0	0	0	
3.12	If menu lists are long (more than seven items), can users select an item either by moving the cursor or by typing a mnemonic code?	0	0	0	
3.13	If the system uses a pointing device, do users have the option of either clicking on menu items or using a keyboard shortcut?	0	0	0	
3.14	Are menus broad (many items on a menu) rather than deep (many menu levels)?	0	0	0	
3.15	If the system has multiple menu levels, is there a mechanism that allows users to go back to previous menus?	0	0	0	
3.16	If users can go back to a previous menu, can they change their earlier menu choice?	0	0	0	
3.17	Can users move forward and backward between fields or dialog box options?	0	0	0	
3.18	If the system has multipage data entry screens, can users move backward and forward among all the pages in the set?	0	0	0	
3.19	If the system uses a question and answer interface, can users go back to previous questions or skip forward to later questions?	0	0	0	
3.20	Do function keys that can cause serious consequences have an undo feature?	0	0	0	
3.21	Can users easily reverse their actions?	0	0	0	
3.22	If the system allows users to reverse their actions, is there a retracing mechanism to allow for multiple undos?	0	0	0	
3.23	Can users set their own system, session, file, and screen defaults?	0	0	0	



Consistency and Standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

#	Review Checklist	Yes	No	N/A	Comments
4.1	Have industry or company formatting standards been followed consistently in all screens within a system?	0	0	0	
4.2	Has a heavy use of all uppercase letters on a screen been avoided?	0	0	0	
4.3	Do abbreviations not include punctuation?	0	0	0	
4.4	Are integers right-justified and real numbers decimal-aligned?	0	0	0	
4.5	Are icons labeled?	0	0	0	
4.6	Are there no more than twelve to twenty icon types?	0	0	0	
4.7	Are there salient visual cues to identify the active window?	0	0	0	
4.8	Does each window have a title?	0	0	0	
4.9	Are vertical and horizontal scrolling possible in each window?	0	0	0	
4.10	Does the menu structure match the task structure?	0	0	0	
4.11	Have industry or company standards been established for menu design, and are they applied consistently on all menu screens in the system?	0	0	0	
4.12	Are menu choice lists presented vertically?	0	0	0	
4.13	If "exit" is a menu choice, does it always appear at the bottom of the list?	0	0	0	
4.14	Are menu titles either centered or left-justified?	0	0	0	
4.15	Are menu items left-justified, with the item number or mnemonic preceding the name?	0	0	0	
4.16	Do embedded field-level prompts appear to the right of the field label?	0	0	0	
4.17	Do on-line instructions appear in a consistent location across screens?	0	0	0	
4.18	Are field labels and fields distinguished typographically?	0	0	0	
4.19	Are field labels consistent from one data entry screen to another?	0	0	0	
4.20	Are fields and labels left-justified for alpha lists and right-justified for numeric lists?	0	0	0	
4.21	Do field labels appear to the left of single fields and above list fields?	0	0	0	
4.22	Are attention-getting techniques used with care?	0	0	0	
4.23	Intensity: two levels only	0	0	0	
4.24	Size: up to four sizes	0	0	0	
4.25	Font: up to three	0	0	0	
4.26	Blink: two to four hertz	0	0	0	
4.27	Color: up to four (additional colors for occasional use only)	0	0	0	
4.28	Sound: soft tones for regular positive feedback, harsh for rare critical conditions	0	0	0	



4.29	Are attention-getting techniques used only for exceptional conditions or for time-dependent information?	0	0	0	
4.30	Are there no more than four to seven colors, and are they far apart along the visible spectrum?	0	0	0	
4.31	Is a legend provided if color codes are numerous or not obvious in meaning?	0	0	0	
4.32	Have pairings of high-chroma, spectrally extreme colors been avoided?	0	0	0	
4.33	Are saturated blues avoided for text or other small, thin line symbols?	0	0	0	
4.34	Is the most important information placed at the beginning of the prompt?	0	0	0	
4.35	Are user actions named consistently across all prompts in the system?	0	0	0	
4.36	Are system objects named consistently across all prompts in the system?	0	0	0	
4.37	Do field-level prompts provide more information than a restatement of the field name?	0	0	0	
4.38	For question and answer interfaces, are the valid inputs for a question listed?	0	0	0	
4.39	Are menu choice names consistent, both within each menu and across the system, in grammatical style and terminology?	0	0	0	
4.40	Does the structure of menu choice names match their corresponding menu titles?	0	0	0	
4.41	Are commands used the same way, and do they mean the same thing, in all parts of the system?	0	0	0	
4.42	Does the command language have a consistent, natural, and mnemonic syntax?	0	0	0	
4.43	Do abbreviations follow a simple primary rule and, if necessary, a simple secondary rule for abbreviations that otherwise would be duplicates?	0	0	0	
#	Review Checklist	Yes	No	N/A	Comments
4.44	Is the secondary rule used only when necessary?	0	0	0	
4.45	Are abbreviated words all the same length?	0	0	0	
4.46	Is the structure of a data entry value consistent from screen to screen?	0	0	0	
4.47	Is the method for moving the cursor to the next or previous field consistent throughout the system?	0	0	0	
4.48	If the system has multipage data entry screens, do all pages have the same title?	0	0	0	
4.49	If the system has multipage data entry screens, does each page have a sequential page number?	0	0	0	
4.50	Does the system follow industry or company standards for function key assignments?	0	0	0	
4.51	Are high-value, high-chroma colors used to attract attention?	0	0	0	



Help Users Recognize, Diagnose, and Recover From Errors

Error messages should be expressed in plain language(NO CODES).

#	Review Checklist	Yes	No	N/A	Comments
5.1	Is sound used to signal an error?	0	0	0	
5.2	Are prompts stated constructively, without overt or implied criticism of the user?	0	0	0	
5.3	Do prompts imply that the user is in control?	0	0	0	
5.4	Are prompts brief and unambiguous.	0	0	0	
5.5	Are error messages worded so that the system, not the user, takes the blame?	0	0	0	
5.6	If humorous error messages are used, are they appropriate and inoffensive to the user population?	0	0	0	
5.7	Are error messages grammatically correct?	0	0	0	
5.8	Do error messages avoid the use of exclamation points?	0	0	0	
5.9	Do error messages avoid the use of violent or hostile words?	0	0	0	
5.10	Do error messages avoid an anthropomorphic tone?	0	0	0	
5.11	Do all error messages in the system use consistent grammatical style, form, terminology, and abbreviations?	0	0	0	
5.12	Do messages place users in control of the system?	0	0	0	
5.13	Does the command language use normal action-object syntax?	0	0	0	
5.14	Does the command language avoid arbitrary, non-English use of punctuation, except for symbols that users already know?	0	0	0	
5.15	If an error is detected in a data entry field, does the system place the cursor in that field or highlight the error?	0	0	0	
5.16	Do error messages inform the user of the error's severity?	0	0	0	
5.17	Do error messages suggest the cause of the problem?	0	0	0	
5.18	Do error messages provide appropriate semantic information?	0	0	0	
5.19	Do error messages provide appropriate syntactic information?	0	0	0	
5.20	Do error messages indicate what action the user needs to take to correct the error?	0	0	0	
5.21	If the system supports both novice and expert users, are multiple levels of error-message detail available?	0	0	0	



Error Prevention

Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

#	Review Checklist	Yes	No	N/A	Comments
6.1	If the database includes groups of data, can users enter more than one group on a single screen?	0	0	0	
6.2	Have dots or underscores been used to indicate field length?	0	0	0	
6.3	Is the menu choice name on a higher-level menu used as the menu title of the lower-level menu?	0	0	0	
6.4	Are menu choices logical, distinctive, and mutually exclusive?	0	0	0	
6.5	Are data inputs case-blind whenever possible?	0	0	0	
6.6	If the system displays multiple windows, is navigation between windows simple and visible?	0	0	0	
6.7	Are the function keys that can cause the most serious consequences in hard-to-reach positions?	0	0	0	
6.8	Are the function keys that can cause the most serious consequences located far away from low-consequence and high-use keys?	0	0	0	
6.9	Has the use of qualifier keys been minimized?	0	0	0	
6.10	If the system uses qualifier keys, are they used consistently throughout the system?	0	0	0	
6.11	Does the system prevent users from making errors whenever possible?	0	0	0	
6.12	Does the system warn users if they are about to make a potentially serious error?	0	0	0	
6.13	Does the system intelligently interpret variations in user commands?	0	0	0	
6.14	Do data entry screens and dialog boxes indicate the number of character spaces available in a field?	0	0	0	
6.15	Do fields in data entry screens and dialog boxes contain default values when appropriate?	0	0	0	



Recognition Rather Than Recall

Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

#	Review Checklist	Yes	No	N/A	Comments
7.1	For question and answer interfaces, are visual cues and white space used to distinguish questions, prompts, instructions, and user input?	0	0	0	
7.2	Does the data display start in the upper-left corner of the screen?	0	0	0	
7.3	Are multiword field labels placed horizontally (not stacked vertically)?	0	0	0	
7.4	Are all data a user needs on display at each step in a transaction sequence?	0	0	0	
7.5	Are prompts, cues, and messages placed where the eye is likely to be looking on the screen?	0	0	0	
7.6	Have prompts been formatted using white space, justification, and visual cues for easy scanning?	0	0	0	
7.7	Do text areas have "breathing space" around them?	0	0	0	
7.8	Is there an obvious visual distinction made between "choose one" menu and "choose many" menus?	0	0	0	
7.9	Have spatial relationships between soft function keys (on-screen cues) and keyboard function keys been preserved?	0	0	0	
7.10	Does the system gray out or delete labels of currently inactive soft function keys?	0	0	0	
7.11	Is white space used to create symmetry and lead the eye in the appropriate direction?	0	0	0	
7.12	Have items been grouped into logical zones, and have headings been used to distinguish between zones?	0	0	0	
7.13	Are zones no more than twelve to fourteen characters wide and six to seven lines high?	0	0	0	
7.14	Have zones been separated by spaces, lines, color, letters, bold titles, rules lines, or shaded areas?	0	0	0	
7.15	Are field labels close to fields, but separated by at least one space?	0	0	0	
7.16	Are long columnar fields broken up into groups of five, separated by a blank line?	0	0	0	
7.17	Are optional data entry fields clearly marked?	0	0	0	
7.18	Are symbols used to break long input strings into "chunks"?	0	0	0	
7.19	Is reverse video or color highlighting used to get the user's attention?	0	0	0	
7.20	Is reverse video used to indicate that an item has been selected?	0	0	0	
7.21	Are size, boldface, underlining, color, shading, or typography used to show relative quantity or importance of different screen items?	0	0	0	
7.22	Are borders used to identify meaningful groups?	0	0	0	
7.23	Has the same color been used to group related elements?	0	0	0	
7.24	Is color coding consistent throughout the system?	0	0	0	
7.25	Is color used in conjunction with some other redundant cue?	0	0	0	



7.26	Is there good color and brightness contrast between image and background colors?	0	0	0	
7.27	Have light, bright, saturated colors been used to emphasize data and have darker, duller, and desaturated colors been used to de-emphasize data?	0	0	0	
7.28	Is the first word of each menu choice the most important?	0	0	0	
7.29	Does the system provide mapping: that is, are the relationships between controls and actions apparent to the user?	0	0	0	
7.30	Are input data codes distinctive?	0	0	0	
7.31	Have frequently confused data pairs been eliminated whenever possible?	0	0	0	
7.32	Have large strings of numbers or letters been broken into chunks?	0	0	0	
7.33	Are inactive menu items grayed out or omitted?	0	0	0	
7.34	Are there menu selection defaults?	0	0	0	
7.35	If the system has many menu levels or complex menu levels, do users have access to an on-line spatial menu map?	0	0	0	
7.36	Do GUI menus offer affordance: that is, make obvious where selection is possible?	0	0	0	
7.37	Are there salient visual cues to identify the active window?	0	0	0	
7.38	Are function keys arranged in logical groups?	0	0	0	
7.39	Do data entry screens and dialog boxes indicate when fields are optional?	0	0	0	
7.40	On data entry screens and dialog boxes, are dependent fields displayed only when necessary?	0	0	0	



Flexibility and Minimalist Design

Accelerators-unseen by the novice user-may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions. Provide alternative means of access and operation for users who differ from the “average” user (e.g., physical or cognitive ability, culture, language, etc.)

#	Review Checklist	Yes	No	N/A	Comments
8.1	If the system supports both novice and expert users, are multiple levels of error message detail available?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.2	Does the system allow novices to use a keyword grammar and experts to use a positional grammar?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.3	Can users define their own synonyms for commands?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.4	Does the system allow novice users to enter the simplest, most common form of each command, and allow expert users to add parameters?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.5	Do expert users have the option of entering multiple commands in a single string?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.6	Does the system provide function keys for high-frequency commands?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.7	For data entry screens with many fields or in which source documents may be incomplete, can users save a partially filled screen?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.8	Does the system automatically enter leading zeros?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.9	If menu lists are short (seven items or fewer), can users select an item by moving the cursor?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.10	If the system uses a type-ahead strategy, do the menu items have mnemonic codes?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.11	If the system uses a pointing device, do users have the option of either clicking on fields or using a keyboard shortcut?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.12	Does the system offer “find next” and “find previous” shortcuts for database searches?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.13	On data entry screens, do users have the option of either clicking directly on a field or using a keyboard shortcut?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.14	On menus, do users have the option of either clicking directly on a menu item or using a keyboard shortcut?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.15	In dialog boxes, do users have the option of either clicking directly on a dialog box option or using a keyboard shortcut?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8.16	Can expert users bypass nested dialog boxes with either type-ahead, user-defined macros, or keyboard shortcuts?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	



Aesthetic and Minimalist Design

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

#	Review Checklist	Yes	No	N/A	Comments
9.1	Is only (and all) information essential to decision making displayed on the screen?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.2	Are all icons in a set visually and conceptually distinct?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.3	Have large objects, bold lines, and simple areas been used to distinguish icons?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.4	Does each icon stand out from its background?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.5	If the system uses a standard GUI interface where menu sequence has already been specified, do menus adhere to the specification whenever possible?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.6	Are meaningful groups of items separated by white space?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.7	Does each data entry screen have a short, simple, clear, distinctive title?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.8	Are field labels brief, familiar, and descriptive?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.9	Are prompts expressed in the affirmative, and do they use the active voice?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.10	Is each lower-level menu choice associated with only one higher level menu?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.11	Are menu titles brief, yet long enough to communicate?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9.12	Are there pop-up or pull-down menus within data entry fields that have many, but well-defined, entry options?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	



Help and Documentation

Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

#	Review Checklist	Yes	No	N/A	Comments
10.1	If users are working from hard copy, are the parts of the hard copy that go on-line marked?	0	0	0	
10.2	Are on-line instructions visually distinct?	0	0	0	
10.3	Do the instructions follow the sequence of user actions?	0	0	0	
10.4	If menu choices are ambiguous, does the system provide additional explanatory information when an item is selected?	0	0	0	
10.5	Are data entry screens and dialog boxes supported by navigation and completion instructions?	0	0	0	
10.6	If menu items are ambiguous, does the system provide additional explanatory information when an item is selected?	0	0	0	
10.7	Are there memory aids for commands, either through on-line quick reference or prompting?	0	0	0	
10.8	Is the help function visible; for example, a key <input type="checkbox"/> labelled HELP or a special menu?	0	0	0	
10.9	Is the help system interface (navigation, presentation, and conversation) consistent with the navigation, presentation, and conversation interfaces of the application it supports?	0	0	0	
10.10	Navigation: Is information easy to find?	0	0	0	
10.11	Presentation: Is the visual layout well designed?	0	0	0	
10.12	Conversation: Is the information accurate, complete, and understandable?	0	0	0	
10.13	Is the information relevant?	0	0	0	
10.14	Goal-oriented (What can I do with this program?)	0	0	0	
10.15	Descriptive (What is this thing for?)	0	0	0	
10.16	Procedural (How do I do this task?)	0	0	0	
10.17	Interpretive (Why did that happen?)	0	0	0	
10.18	Navigational (Where am I?)	0	0	0	
10.19	Is there context-sensitive help?	0	0	0	
10.20	Can the user change the level of detail available?	0	0	0	
10.21	Can users easily switch between help and their work?	0	0	0	
10.22	Is it easy to access and return from the help system?	0	0	0	
10.23	Can users resume work where they left off after accessing help?	0	0	0	



Skills

The system should support, extend, supplement, or enhance the user's skills, background knowledge, and expertise ---- not replace them.

#	Review Checklist	Yes	No	N/A	Comments
11.1	Can users choose between iconic and text display of information?	0	0	0	
11.2	Are window operations easy to learn and use?	0	0	0	
11.3	If users are experts, usage is frequent, or the system has a slow response time, are there fewer screens (more information per screen)?	0	0	0	
11.4	If users are novices, usage is infrequent, or the system has a fast response time, are there more screens (less information per screen)?	0	0	0	
11.5	Does the system automatically color-code items, with little or no user effort?	0	0	0	
11.6	If the system supports both novice and expert users, are multiple levels of detail available.	0	0	0	
11.7	Are users the initiators of actions rather than the responders?	0	0	0	
11.8	Does the system perform data translations for users?	0	0	0	
11.9	Do field values avoid mixing alpha and numeric characters whenever possible?	0	0	0	
11.10	If the system has deep (multilevel) menus, do users have the option of typing ahead?	0	0	0	
11.12	When the user enters a screen or dialog box, is the cursor already positioned in the field users are most likely to need?	0	0	0	
11.13	Can users move forward and backward within a field?	0	0	0	
11.14	Is the method for moving the cursor to the next or previous field both simple and visible?	0	0	0	
11.15	Has auto-tabbing been avoided except when fields have fixed lengths or users are experienced?	0	0	0	
11.16	Do the selected input device(s) match user capabilities?	0	0	0	
11.17	Are cursor keys arranged in either an inverted T (best for experts) or a cross configuration (best for novices)?	0	0	0	
11.18	Are important keys (for example, ENTER , TAB) larger than other keys?	0	0	0	
11.19	Are there enough function keys to support functionality, but not so many that scanning and finding are difficult?	0	0	0	
11.20	Are function keys reserved for generic, high-frequency, important functions?	0	0	0	
11.21	Are function key assignments consistent across screens, subsystems, and related products?	0	0	0	
11.22	Does the system correctly anticipate and prompt for the user's probable next activity?	0	0	0	



Pleasurable and Respectful Interaction with the User

The user's interactions with the system should enhance the quality of her or his work-life. The user should be treated with respect. The design should be aesthetically pleasing- with artistic as well as functional value.

#	Review Checklist	Yes	No	N/A	Comments
12.1	Is each individual icon a harmonious member of a family of icons?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.2	Has excessive detail in icon design been avoided?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.3	Has color been used with discretion?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.4	Has the amount of required window housekeeping been kept to a minimum?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.5	If users are working from hard copy, does the screen layout match the paper form?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.6	Has color been used specifically to draw attention, communicate organization, indicate status changes, and establish relationships?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.7	Can users turn off automatic color coding if necessary?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.8	Are typing requirements minimal for question and answer interfaces?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.9	Do the selected input device(s) match environmental constraints?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.13	If the system uses multiple input devices, has hand and eye movement between input devices been minimized?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.14	If the system supports graphical tasks, has an alternative pointing device been provided?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.15	Is the numeric keypad located to the right of the alpha key area?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.16	Are the most frequently used function keys in the most accessible positions?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
12.17	Does the system complete unambiguous partial input on a data entry field?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	



Privacy

The system should help the user to protect personal or private information- belonging to the user or the his/her clients.

#	Review Checklist	Yes	No	N/A	Comments
13.1	Are protected areas completely inaccessible?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
13.2	Can protected or confidential areas be accessed with certain passwords.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
13.3	Is this feature effective and successful.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	



8. APENDICE 2

Heurísticas traducidas al castellano y ampliadas y modificadas.

VISIBILIDAD DEL ESTADO DEL SISTEMA

1.1	Empieza cada pantalla con un título o cabecera que describe su contenido.
1.2	Hay un esquema de diseño de iconos consistente y un tratamiento estilístico por todo el sistema.
1.3	Está un solo icono seleccionado claramente visible cuando está rodeado por iconos no seleccionados. Se ilumina o resalta al pasar el puntero sobre él.
1.4	En los menús, las instrucciones, avisos y mensajes de error, aparecen siempre en la misma ubicación.
1.5	En las pantallas multipágina de entrada de datos, está cada página etiquetada para mostrar su relación con otras.
1.6	Si el modo de sobrescritura e inserción están ambos disponibles, hay una indicación visible del modo que está activo.
1.7	Hay alguna indicación del estado de las teclas de Bloq. Num, Bloq. Mayús y Bloq. Despl.
1.8	Cambia el cursor de aspecto en función del objeto, o lugar del objeto, sobre el que está posicionado, sugiriendo lo que se puede hacer con el objeto.
1.9	Si las ventanas pop-up (saltan de pronto) son usadas para visualizar mensajes de error, permite al usuario ver el campo que causa el error.
1.10	Hay alguna forma de realimentación del sistema para cada una de las acciones del operador.
1.11	Después de que el usuario completa una acción (o grupo de acciones), indica la realimentación que el siguiente grupo de acciones puede ser empezado.
1.12	Hay realimentación visual en menús o cajas de diálogo sobre qué opciones son seleccionables.
1.13	Hay realimentación visual en menús o cajas de diálogo sobre qué opción está apuntando el cursor.
1.14	Si múltiples opciones pueden ser seleccionadas en un menú o caja de diálogo, hay realimentación visual sobre qué opciones están ya seleccionadas.
1.15	Hay realimentación visual cuando los objetos son seleccionados o movidos.
1.16	El estado actual de un icono está claramente indicado.
1.17	Hay realimentación cuando son presionadas las teclas de función.
1.18	Si hay esperas importantes (mayores de 15 segundos) en los tiempos de respuesta del sistema, se mantiene informado al usuario del progreso del sistema.
1.19	Los tiempos de respuesta son apropiados para las tareas.
1.20	Tecleando, movimiento del cursor, selección del ratón: 50 – 150 milisegundos.
1.21	Tareas simples o frecuentes: menos de un segundo.
1.22	Tareas comunes: 2 – 4 segundos.
1.23	Tareas complejas: 8 – 12 segundos.
1.24	Los tiempos de respuesta son apropiados para el proceso cognitivo de los usuarios.



1.25	La continuidad de pensamiento es requerida y la información debe ser recordada a lo largo de varias respuestas: menos de 2 segundos.
1.26	No son necesarios altos niveles de concentración y no se requiere estar recordando la información: de 2 a 15 segundos.
1.27	La ventana principal de la aplicación tiene barra de estado.
1.28	La barra de estado está dividida en varias áreas, una para cada función.
1.29	Es la terminología de denominación de menú consistente con el dominio de tareas del usuario.
1.30	En los sistemas MDI (Interfaz de Documento Múltiple), está claramente visible la ventana activa.
1.31	Provee el sistema visibilidad: esto es, mirando, puede el usuario discernir el estado del sistema y las alternativas para actuar.
1.32	Son obvios en los menús del GUI los ítems que han sido seleccionados.
1.33	Es obvio en los menús del GUI si es posible la desección.
1.34	Si el usuario debe navegar entre múltiples pantallas, hace el sistema uso de etiquetas de contexto, mapas de menús, y coloca marcadores como ayuda a la navegación.



CONTROL Y LIBERTAD DEL USUARIO

2.1	Si la configuración de ventanas (ver, acoplamiento, colocación,...) es una tarea frecuente, es particularmente fácil de recordar.
2.2	En los sistemas que usan solapado de ventanas, es fácil para el usuario reestructurar las ventanas sobre la pantalla.
2.3	En los sistemas que usan solapado de ventanas, es fácil para el usuario cambiar entre ventanas.
2.4	Cuando una tarea del usuario está completa, espera el sistema por una señal del usuario antes de procesar.
2.5	Es posible cambiar el aspecto (look and feel) del sistema.
2.6	Se avisa al usuario para confirmar órdenes que tienen efectos drásticos, con consecuencias destructivas.
2.7	Hay una función "desahacer" al nivel de una acción simple, una entrada de dato, y un grupo completo de acciones.
2.8	El usuario puede cancelar las operaciones en progreso.
2.9	El usuario puede reducir el tiempo de entrada de datos copiando o modificando datos existentes.
2.10	Si la aplicación es MDI (Interfaz de Documento Múltiple), sólo la ventana padre es modal. Las ventanas hijo deben presentarse dentro de la ventana padre.
2.11	Si la aplicación es MDI (Interfaz de Documento Múltiple), las ventanas hijo se pueden colocar en cascada o composición.
2.12	Está permitida la corrección de caracteres en los campos de entrada de datos.
2.13	Si las listas de menús son largas (más de siete elementos), el usuario puede seleccionar un elemento, bien moviendo el cursor, bien tecleando un código mnemotécnico.
2.14	Si el sistema usa un dispositivo apuntador, tiene el usuario ambas opciones de pulsar sobre los elementos del menú o usar atajos de teclado.
2.15	Existe un menú de control en cada ventana de la aplicación
2.16	Son los menús extensos (muchos elementos en un menú) más que profundos (muchos niveles de menú).
2.17	Si el sistema tiene muchos niveles de menú, hay un mecanismo que permita al usuario volver a los menús previos.
2.18	Las teclas de método abreviado son totalmente configurables.
2.19	El sistema de menús es adaptativo, esto es, aparecen en los menús las opciones más usadas por el usuario.
2.20	El usuario puede elegir entre diferentes menús, en función del trabajo que realiza.
2.21	El usuario puede seleccionar las barras de herramientas que quiere ver.
2.22	En las barras de herramientas, es posible ver los botones con icono más texto.
2.23	Para cada barra de herramientas el usuario puede elegir los comandos (botones) que quiere ver.
2.24	El trabajo que realiza la aplicación es configurable (fuentes, signos de puntuación, colores, líneas,...)
2.25	En los menús, hay tres o menos de tres niveles de menú.



2.26	Puede el usuario moverse hacia delante y hacia atrás entre campos u opciones en las cajas de diálogo.
2.27	Si el sistema tiene pantallas multipágina de entrada de datos, el usuario puede moverse hacia delante o hacia atrás entre todas las páginas del conjunto.
2.28	Si el sistema usa una interfaz de pregunta y respuesta, el usuario puede volver a preguntas previas o pasar a cuestiones posteriores.
2.29	Las teclas de función que pueden tener consecuencias serias, tienen una característica deshacer.
2.30	El usuario puede fácilmente retroceder en sus acciones.
2.31	El usuario puede salir de la aplicación desde cualquier parte de ella.
2.32	La aplicación se muestra en el idioma del usuario. En caso contrario el usuario puede cambiar el idioma por defecto de los menús, mensajes y etiquetas
2.33	Si el sistema soporta tareas gráficas ha sido proporcionada una alternativa al dispositivo apuntador.
2.34	Las tareas más complejas están divididas en tareas más simples
2.35	El usuario puede cambiar el directorio por defecto de salida de su trabajo
2.36	Las secuencias largas están divididas en pasos separados
2.37	Si el sistema permite al usuario retroceder en sus acciones, hay un mecanismo de trazabilidad para permitir deshacer múltiples veces.
2.38	El usuario puede configurar su sistema, sesión, archivo, y pantallas por defecto.
2.39	El usuario puede apagar el color codificado automáticamente si fuera necesario.
2.40	El usuario puede cambiar el tamaño de los botones de la barra de herramientas.
2.41	El usuario puede colocar las diferentes barras de herramientas en el lugar que prefiera.



CONSISTENCIA Y ESTANDARES

3.1	Los estándares estructurados que tiene la compañía han sido seguidos consistentemente en todas las pantallas dentro del sistema.
3.2	Ha sido evitado el pesado uso de letras mayúsculas en la pantalla.
3.3	Están los enteros justificados a la derecha y los números reales alineados por decimales.
3.4	Está disponible la ayuda pulsando la tecla de función F1
3.5	En la barra de menú general, la última opción es el menú "Ayuda" o "?".
3.6	Existe un menú "Archivo"
3.7	Existe un menú "Ayuda"
3.8	Existe un menú "Ventana"
3.9	Es posible realizar el despliegue de los menús pulsando la tecla "Alt" más una letra del menú.
3.10	Las opciones de "Nuevo", "Abrir", "Guardar", "Imprimir", "Configurar", si son necesarias, están en el menú "Archivo"
3.11	Las opciones de "Cortar", "Copiar" y "Pegar", si son necesarias, están en un menú "Edición".
3.12	Los botones son todos de tamaño y forma similar a lo largo de la aplicación.
3.13	Están los iconos etiquetados.
3.14	No hay más de entre doce o veinte tipos de iconos.
3.15	Hay señales visuales para identificar la ventana activa.
3.16	Cada ventana tiene un título que la identifica claramente.
3.17	Los elementos de los menús que llevan a abrir un submenú contienen el símbolo ► en la parte derecha.
3.18	En los botones de comando que contienen iconos y texto, este está colocado a la derecha o debajo del icono.
3.19	Los títulos del menú principal constan exactamente de una palabra.
3.20	Los títulos de los elementos de los submenús constan de una palabra o frase corta.
3.21	Es posible el scrolling vertical y horizontal en cada ventana, si fuera necesario.
3.22	Corresponde la estructura del menú a la estructura de la tarea.
3.23	Los estándares que tiene la compañía han sido establecidos para el diseño de menús, y están aplicados consistentemente sobre todos los menús en las pantallas del sistema.
3.24	Las listas de opciones de menú están presentadas verticalmente.
3.25	Si "Salir" es una opción del menú, aparece siempre en la parte baja de la lista del menú "Archivo".
3.26	Los títulos de los menús están centrados o justificados a la izquierda.
3.27	Aparece la invitación a introducir datos en un campo en la parte derecha del mismo.
3.28	Aparecen las instrucciones en línea en una ubicación consistente en todas las pantallas.
3.29	Las etiquetas de los campos y los campos están distinguidas tipográficamente. (diferentes fuentes)
3.30	Las etiquetas de los campos son consistentes desde una pantalla de entrada de datos a otra.



3.31	Los campos y las etiquetas están justificados a la izquierda para listas alfabéticas y justificadas a la derecha por listas numéricas.
3.32	Las etiquetas de los campos aparecen a la izquierda de un sólo campo y sobre los campos de lista.
3.33	Las técnicas de atraer la atención son usadas de forma adecuada.
3.34	Intensidad: sólo dos niveles
3.35	Tamaño de fuentes: hasta cuatro tamaños
3.36	Conjunto de caracteres: hasta tres tipos
3.37	Parpadeo: de dos a cuatro cambios
3.38	Color: hasta cuatro (colores adicionales sólo para uso ocasional)
3.39	Sonido: tonos suaves para una realimentación positiva regular, tono áspero para condiciones críticas raras.
3.40	Las técnicas de atraer la atención son usadas sólo para condiciones excepcionales o para información dependiente del tiempo.
3.41	No hay más que cuatro a siete colores, y están separados a lo largo del espectro.
3.42	Se proporciona una leyenda si los códigos de colores son numerosos o no es obvio su significado
3.43	Han sido evitados los emparejamientos de colores de alto valor cromático y los extremos del espectro.
3.44	Son evitados los azules saturados para pequeño texto u otros símbolos de línea delgados.
3.45	Las acciones de usuario son nombradas consistentemente a través de todos los prompts del sistema.
3.46	Los objetos son nombrados consistentemente a través de todos los prompts del sistema.
3.47	Para las interfaces de pregunta y respuesta, se listan las entradas válidas para una pregunta.(entradas posibles)
3.48	Los nombres de las opciones de menú son consistentes, para cada menú y a través del sistema, en estilo gramatical y terminología.
3.49	Casan los nombres de las opciones de la estructura de menús con sus correspondientes títulos de menú. (Si están en el menú apropiado)
3.50	Las órdenes son usadas de la misma forma, y significan lo mismo en todas las partes del sistema.
3.51	El lenguaje de comandos tiene una sintaxis consistente, natural y mnemotécnica.
3.52	Son todas las palabras abreviadas de la misma longitud.
3.53	La estructura del valor de un campo de entrada de datos es consistente de una pantalla a otra.
3.54	El método para mover el cursor al siguiente o anterior campo es consistente a lo largo del sistema.
3.55	Si el sistema tiene pantallas con entrada de datos multipágina, tienen todas las páginas el mismo título.
3.56	Si el sistema tiene pantallas con entrada de datos multipágina, tiene cada página un número de página secuencial.
3.57	La asignación de teclas de función es consistente a lo largo de las pantallas, subsistemas y productos relacionados.
3.58	El sistema sigue los estándares de la compañía para la asignación de las teclas de función.



3.59 Los colores de alto valor cromático son usados para atraer la atención del usuario.



PREVENCIÓN DE ERRORES

4.1	Si la base de datos incluye grupos de datos, los usuarios pueden introducir más de un grupo en una sola pantalla.
4.2	Son usados puntos o subrayado para indicar la longitud del campo.
4.3	Las opciones de menú son lógicas, distintivas, y mutuamente excluyentes.
4.4	Está iluminada la entrada de datos siempre que sea posible.
4.5	Las áreas protegidas son completamente inaccesibles.
4.6	Las áreas protegidas o confidenciales son accedidas con passwords.
4.7	Si el sistema muestra múltiples ventanas, es la navegación entre ventanas simple y visible.
4.8	Están las teclas de función que pueden causar las consecuencias más serias en una posición de difícil alcance.
4.9	Están las teclas de función que pueden causar las consecuencias más serias, localizadas lejos de las de funciones con baja consecuencia y alto uso.
4.10	Ha sido minimizado el uso de teclas cualificadas. (Ctrl, Alt, Shift y command key en Apple)
4.11	Si el sistema usa teclas cualificadas, son usadas consistentemente a lo largo del sistema.
4.12	Previene el sistema a los usuarios de cometer errores siempre que es posible.
4.13	Advierte el sistema a los usuarios sobre la producción de un potencial error serio.
4.14	Interpreta el sistema inteligentemente variaciones en las órdenes de usuario.
4.15	Indican las pantallas de entrada de datos y cajas de diálogo el número de espacios disponibles en los campos.
4.16	En los mensajes, la información más importante está colocada al principio. Sigue la secuencia de acciones del usuario.
4.17	Contienen las pantallas de entrada de datos y cajas de diálogo valores por defecto cuando es apropiado.
4.18	En las entradas de datos, se da al usuario información del formato esperado de los datos.
4.19	Están las teclas de función reservadas para funciones importantes, genéricas de mucha frecuencia.
4.20	El sistema se anticipa correctamente y sugiere la siguiente actividad de los usuarios.
4.21	El sistema completa la entrada parcial inequívoca en un campo de entrada de datos.
4.22	Ha sido evitado el uso de auto-tab excepto cuando los campos tienen una longitud fija o los usuarios son experimentados.
4.23	Informa el sistema al usuario cuando se intenta salir o cerrar una ventana, que hay trabajo sin guardar
4.24	El método de mover el cursor al campo anterior o campo siguiente es simple y visible.
4.25	Están sólo iluminados los botones necesarios en función de la tarea a realizar.
4.26	Informa el sistema al usuario ante un error de entrada de datos para realizar la corrección.



CORRESPONDENCIA ENTRE EL SISTEMA Y EL MUNDO REAL

5.1	Los iconos son concretos y familiares al usuario.
5.2	Las opciones de menú están ordenadas de la manera más lógica, según el tipo de usuario, los nombres de los ítems, y las variables de las tareas.
5.3	Si hay una secuencia natural para las opciones de menú, se ha usado.
5.4	Campos relacionados e interdependientes aparecen en la misma pantalla.
5.5	Si la forma es usada como una indicación visual, se corresponde con formas convencionales.
5.6	Corresponden los colores seleccionados a expectativas comunes sobre códigos de colores.
5.7	Cuando los avisos implican la necesidad de realizar una una acción, las palabras del mensaje son consistentes con esa acción.
5.8	El nombre de las teclas que se referencian en los avisos se corresponden con el nombre actual de las teclas.
5.9	En las pantallas de entrada de datos, las tareas están descritas en terminología familiar al usuario.
5.10	En las pantallas de entrada de datos, se proporcionan campos para esta tarea.
5.11	Para la interfaces de preguntas y respuestas, las cuestiones son expuestas en lenguaje claro y simple.
5.12	Encajan las opciones de menú lógicamente dentro de categorías de modo que rápidamente se comprende su significado.
5.13	Están los títulos de los menús en los mismos tiempos gramaticales.
5.14	Emplean las órdenes la jerga del usuario y evita la jerga informática.
5.15	Son los nombres de las órdenes, específicas más bien que generales.
5.16	El lenguaje de comandos permite tanto los nombres completos y como las abreviaciones.
5.17	El sistema introduce automáticamente la alineación de los puntos decimales.
5.18	El sistema introduce automáticamente un signo de euro y los decimales para las entradas monetarias.
5.19	El sistema introduce automáticamente puntos en los valores numéricos mayores de 9999.
5.20	El sistema ha sido diseñado de forma que las teclas con nombres similares no realicen acciones opuestas (y potencialmente peligrosas).
5.21	Las teclas de función están etiquetadas clara y distintivamente, incluso si ello conlleva romper reglas de consistencias.



RECONOCER ANTES QUE RECORDAR

6.1	Para interfaces para preguntas y respuestas, las señales visuales y los espacios en blanco son usados para distinguir preguntas, avisos, instrucciones y entradas de usuario.
6.2	La muestra de datos empieza en la esquina arriba-izquierda de la pantalla.
6.3	Las etiquetas multipalabra de los campos están colocadas horizontalmente (no verticalmente apilados).
6.4	Todos los datos que el usuario necesita son mostrados en cada paso en una secuencia de transacción.
6.5	Los avisos, señales y mensajes están colocados donde probablemente el ojo mirará sobre la pantalla.
6.6	Los campos han sido formateados usando espacios en blanco, justificación y señales visuales para examinar fácilmente.
6.7	Tienen las áreas de texto “espacios de pausa” alrededor de ellas.
6.8	Las teclas de función inactivas en pantalla aparecen en gris o sin etiqueta.
6.9	Los espacios en blanco son usados para crear simetría y guiar al ojo en la dirección apropiada.
6.10	Los ítems han sido agrupados dentro de zonas lógicas y han sido utilizados encabezamientos para distinguir entre zonas.
6.11	Las zonas no están con más de doce a catorce caracteres de ancho y de seis a siete líneas de alto.
6.12	Las zonas han sido separadas por espacios, líneas, colores, letras, títulos en negrita, líneas de regla o áreas sombreadas.
6.13	Las etiquetas de los campos están cerca de los campos, pero separadas al menos, por un espacio.
6.14	Los campos en columnas largas están partidos en grupos de cinco, separados por una línea en blanco.
6.15	Los campos de entrada de datos optativos están claramente marcados.
6.16	El video inverso o el color altamente iluminado es usado para conseguir la atención de los usuarios.
6.17	El video inverso es usado para indicar que un elemento ha sido seleccionado.
6.18	Son usados tamaño, negrita, subrayado, color, sombreado o tipografía para mostrar la cantidad relativa o importancia de los diferentes elementos de pantalla.
6.19	Los bordes son usados para identificar los grupos significativos.
6.20	Ha sido usado el mismo color para los grupos de elementos relacionados.
6.21	El color está codificado consistentemente a lo largo del sistema.
6.22	Ha sido usado el color en conjunto con alguna otra señal redundante.
6.23	Hay buen color y contraste de brillo entre las imágenes y el color de fondo.
6.24	Ha sido usada la luz, brillo y colores saturados para enfatizar los datos y han sido usados colores oscuros, densos y no saturados para no enfatizar datos.
6.25	Es la primera palabra de cada opción de menú la más importante.
6.26	El texto de los elementos de los menús que llevan a la apertura de un diálogo terminan en puntos suspensivos sugiriendo que hay un paso intermedio.
6.27	Cada elemento final de los menús tiene un icono indicativo de su función.



6.28	Provee el sistema trazabilidad: esto es, están las relaciones entre controles y las acciones, claras para el usuario.
6.29	Han sido eliminados los pares de datos que frecuentemente confunden, siempre que sea posible.
6.30	Los elementos inactivos de los menús han sido puestos en gris o han sido omitidos.
6.31	Hay menús con selecciones por defecto.
6.32	Si el sistema tiene muchos niveles de menú o niveles de menú complejos, tiene el usuario acceso en línea a un mapa espacial del menú.
6.33	El GUI ofrece acomodación: esto es, es obvio dónde es posible la selección.
6.34	Hay señales resaltadas para identificar la ventana activa.
6.35	Están ordenadas las teclas de función en grupos lógicos.
6.36	Indican las pantallas de entradas de datos y las cajas de diálogo cuándo los campos son opcionales.
6.37	En las pantallas de entrada de datos y cajas de diálogo, los campos dependientes son sólo mostrados cuando es necesario. (p.e. una opción hace aparecer un campo o no)



FLEXIBILIDAD Y EFICIENCIA DE USO

7.1	Si el sistema soporta usuarios noveles y expertos, están disponibles niveles múltiples de detalle de mensajes de error.
7.2	Los usuarios pueden definir sus propios sinónimos para los comandos.
7.3	Permite el sistema a los usuarios noveles introducir la forma más simple y común de cada comando, y permite a los usuarios expertos añadir parámetros.
7.4	Los elementos de los submenús están agrupados conceptualmente y separados por líneas.
7.5	Tienen los usuarios expertos la opción de entrar múltiples comandos en una sola cadena.
7.6	Provee el sistema teclas de función para comandos frecuentes.
7.7	Para pantallas de entrada de datos con muchos campos o el que los documentos fuente pueden estar incompletos, pueden los usuarios salvar una pantalla parcialmente rellena.
7.8	Introduce el sistema automáticamente ceros.
7.9	Si las listas de menús son cortas (siete elementos o menos), los usuarios pueden seleccionar un elemento moviendo el cursor. (teclas del cursor)
7.10	Los elementos de los menús tienen códigos mnemotécnicos (elementos de los menús con una letra subrayada para ejecutarlos)
7.11	Si el sistema usa dispositivo apuntador, tienen los usuarios la opción de pinchar sobre campos o usar un atajo de teclado (pasar a otro campo pulsando una tecla).
7.12	Ofrece el sistema caminos cortos "Siguiente" y "Anterior" para búsquedas en la base de datos.
7.13	En <i>pantallas de entrada de datos</i> , tiene el usuario ambas opciones de pinchar directamente sobre cada campo o usar un atajo de teclado (pasar a otro campo pulsando una tecla).
7.14	En los <i>menús</i> , tiene el usuario ambas opciones de pinchar directamente sobre un elemento del menú o usar un atajo de teclado (pasar a otro campo pulsando una tecla).
7.15	En las <i>cajas de diálogo</i> , tiene el usuario ambas opciones de pinchar directamente sobre una opción de la caja de diálogo o usar un camino corto de teclado.
7.16	Si el sistema tiene menús profundos (multiniveles), el usuario tiene la opción de teclear la orden.
7.17	Cuando el usuario debe introducir datos en una pantalla o caja de diálogo, se posiciona el cursor en el campo en el cual el usuario necesitará.
7.18	Los menús cambian en función de las tareas que el usuario esté realizando.
7.19	Los usuarios pueden ir hacia delante o hacia atrás dentro de un campo.
7.20	Las teclas de función mas frecuentemente usadas están en la posición más accesible.
7.21	Los campos de valores evitan la mezcla de caracteres alfabéticos y numéricos cuando es posible.
7.22	Si el sistema soporta usuarios noveles y expertos, hay múltiples niveles de detalle disponibles.
7.23	El dispositivo de entrada seleccionado es parejo con las capacidades del usuario.
7.24	El sistema realiza las traducciones de los datos para los usuarios.
7.25	Los usuarios pueden escoger entre visualización de información textual o icónica



7.26 Un usuario experto puede bипasar cajas de diálogo anidadas con teclado, macros definidas por el usuario o atajos de teclado.



ESTETICA Y DISEÑO MINIMALISTA

8.1	Está sólo la información esencial (y toda) para la toma de decisión, mostrada sobre la pantalla.
8.2	Están todos los iconos visualmente conjuntados y conceptualmente diferentes.
8.3	Ha sido evitado el excesivo detalle en el diseño de los iconos.
8.4	El color ha sido usado con discreción.
8.5	El color ha sido usado específicamente para atraer la atención, comunicar organización, indicar cambios de estado y establecer relaciones.
8.6	Si el sistema usa múltiples dispositivos de entrada, el movimiento del ojo y mano ha sido minimizado entre los dispositivos de entrada.
8.7	Cada icono individual es miembro armonioso de una familia de iconos.
8.8	Han sido usados los objetos grandes, líneas en negrita y áreas simples para diferenciar iconos.
8.9	Destaca cada icono de su fondo.
8.10	Si el sistema usa una interfaz GUI estándar donde la secuencia de menú ya ha sido especificada, se adhieren los menús a la especificación cuando es posible. (Access)
8.11	Están los grupos más importantes de elementos separados por espacios en blanco.
8.12	Los objetos están distribuidos uniformemente en las cajas de diálogo.
8.13	Los objetos conceptualmente similares están distribuidos en grupos.
8.14	Los botones comunes de los diálogos aparecen siempre colocados en el mismo lugar.
8.15	Existe siempre en las cajas de diálogo los botones "Aceptar" y "Cancelar".
8.16	Cada pantalla de entrada de datos tiene un título corto, simple, claro y distintivo.
8.17	Las etiquetas de los campos son concisas, familiares y descriptivas.
8.18	Los avisos están expresados de forma afirmativa y hacen uso de la voz activa. (el sujeto ejerce una acción sobre el objeto)
8.19	Está cada opción de menú de más bajo nivel asociada con sólo un menú del más alto nivel. No hay repeticiones de opciones.
8.20	Los títulos breves de los menús son informativos, comunican bastante bien para lo que sirven.
8.21	Hay menús pop-up o pull-down dentro de los campos de entrada de datos que tienen diferentes opciones de entrada. (p.e. pulsando dos veces en un campo aparecen las entradas posibles en una lista)



AYUDA A LOS USUARIOS A RECONOCER, DIAGNOSTICAR Y RECUPERAR DESDE ERRORES.

9.1	El sonido es usado como señal de error.
9.2	El aviso (prompt) está declarado constructivamente, sin criticar abierta o implícitamente al usuario.
9.3	El aviso (prompt) implica que el usuario tiene el control.
9.4	Los avisos (prompts) son concisos y no ambiguos.
9.5	Los mensajes de error están formulados para que el sistema tenga la culpa, y no el usuario.
9.6	Si son usados mensajes de error cómicos, son apropiados e inofensivos para la población de usuarios.
9.7	Los mensajes de error son gramaticalmente correctos.
9.8	Los mensajes de error evitan el uso de puntos de exclamación. (escandalizando al usuario)
9.9	Los mensajes de error evitan el uso de palabras violentas y hostiles.
9.10	Los mensajes de error evitan un tono antropomórfico. (atribución de cualidades humanas a cosas naturales o artificiales : ej. "El motor es el corazón del automóvil").
9.11	Todos los mensajes de error en el sistema usan un estilo gramatical consistente, en forma, terminología, y abreviaciones.
9.12	Los mensajes sitúan a los usuarios sobre el control del sistema.
9.13	El lenguaje de comandos usa una sintaxis normal acción–objeto. (ej. "xcopy /etc/lista.txt")
9.14	Si un error es detectado en un campo de datos de entrada, el sistema coloca el cursor en ese campo o ilumina el error.
9.15	Los mensajes de error alarmantes son evitados (p.e. "error fatal", "ejecución abortada", "error catastrófico")
9.16	Los mensajes de error ambiguos son evitados.
9.17	Los mensajes de error están escritos en el idioma del usuario
9.18	Los mensajes de error informan al usuario de la severidad de los errores.
9.19	Los mensajes de error sugieren la causa del problema.
9.20	Los mensajes de error proveen apropiada información semántica. (significado sobre el propio error)
9.21	El sistema utiliza archivos log
9.22	Los mensajes de error proveen apropiada información sintáctica. (significado en el contexto de uso)
9.23	Los mensajes de error indican qué acción del usuario se necesita para corregir el error.
9.24	El sistema guarda el trabajo en curso de manera automática y regular ante una caída del sistema.
9.25	Si el sistema soporta niveles de usuario noveles y expertos, están disponibles múltiples niveles de mensajes de error.



AYUDA Y DOCUMENTACION

10.1	Las instrucciones en línea son visualmente distintas.
10.2	Las instrucciones siguen la secuencia de acciones del usuario.
10.3	Existe algún tipo de ayuda externa a la aplicación: internet, intranet, teléfono de asistencia, help desk, ...
10.4	Se apoyan las pantallas de entrada de datos y cajas de diálogo, en instrucciones de navegación y cumplimentación.
10.5	Al pasar con el ratón sobre los elementos finales de los menús, se muestra ayuda en la barra de estado sobre la función que realizan.
10.6	Si los elementos de menú son ambiguos, provee el sistema información explicativa adicional cuando un elemento es seleccionado.
10.7	Hay ayudas de memoria para los comandos, a través de rápidas referencias en línea o avisando.
10.8	Es la función de ayuda visible; por ejemplo, una botón etiquetado AYUDA o un menú especial.
10.9	El sistema de ayuda tiene un asistente como solucionador de problemas.
10.10	El sistema de ayuda utiliza figuras o gráficos frecuentemente.
10.11	Está la ayuda disponible pulsando la tecla F1 en cualquier momento.
10.12	Es la interfaz de ayuda del sistema (navegación, presentación y conversación) consistente con la interfaz de navegación, presentación y conversación de la aplicación.
10.13	El contenido de la ventana de ayuda es hipertextual.
10.14	Navegación: es fácil de encontrar la información.
10.15	Presentación: el esquema visual está bien diseñado.
10.16	Conversación: la información es exacta, completa y entendible.
10.17	La información es relevante.
10.18	Orientada al objetivo (Qué puedo hacer con este programa).
10.19	Descriptiva (Para qué es esta cosa).
10.20	Procesal (Cómo hago esta tarea).
10.21	Interpretativo (Por qué ocurrió eso).
10.22	Navegacional (Dónde estoy).
10.23	Hay ayuda sensitiva al contexto.
10.24	El usuario puede cambiar el nivel de detalle disponible.
10.25	El usuario puede fácilmente intercambiar entre la ayuda y su trabajo.
10.26	Es fácil acceder y retornar desde el sistema de ayuda.

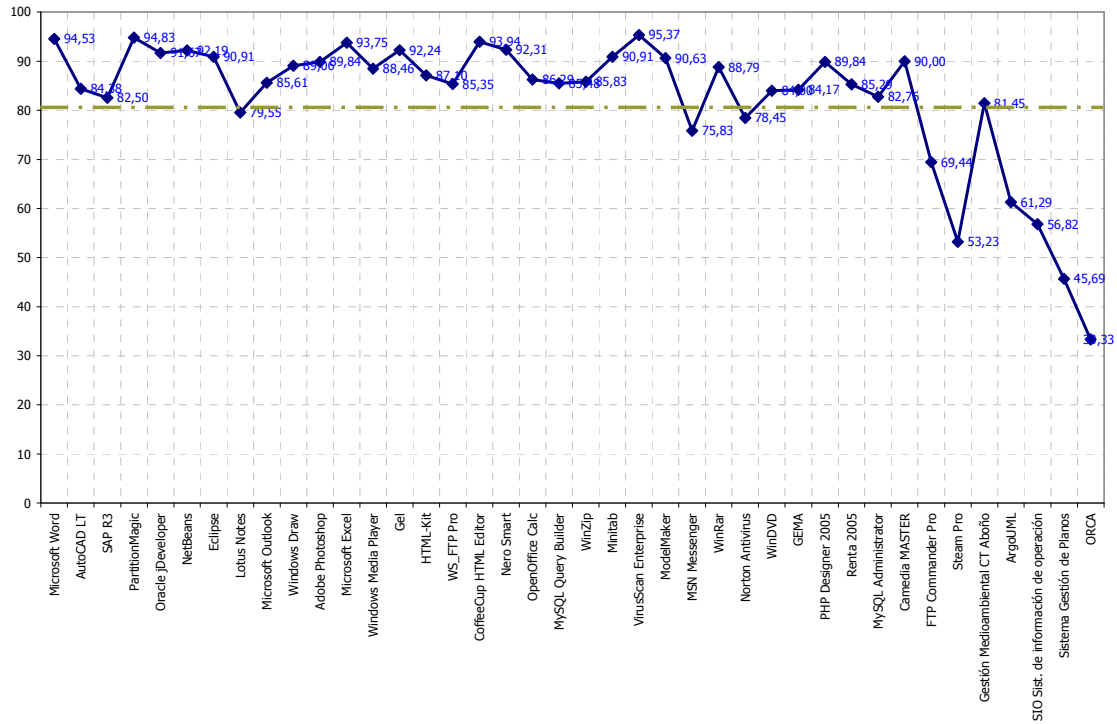


10.27	El usuario puede continuar el trabajo donde lo dejó después de acceder a la ayuda.
10.28	Al colocar el apuntador sobre un elemento gráfico se muestra alguna pequeña explicación sobre él. (tooltip)

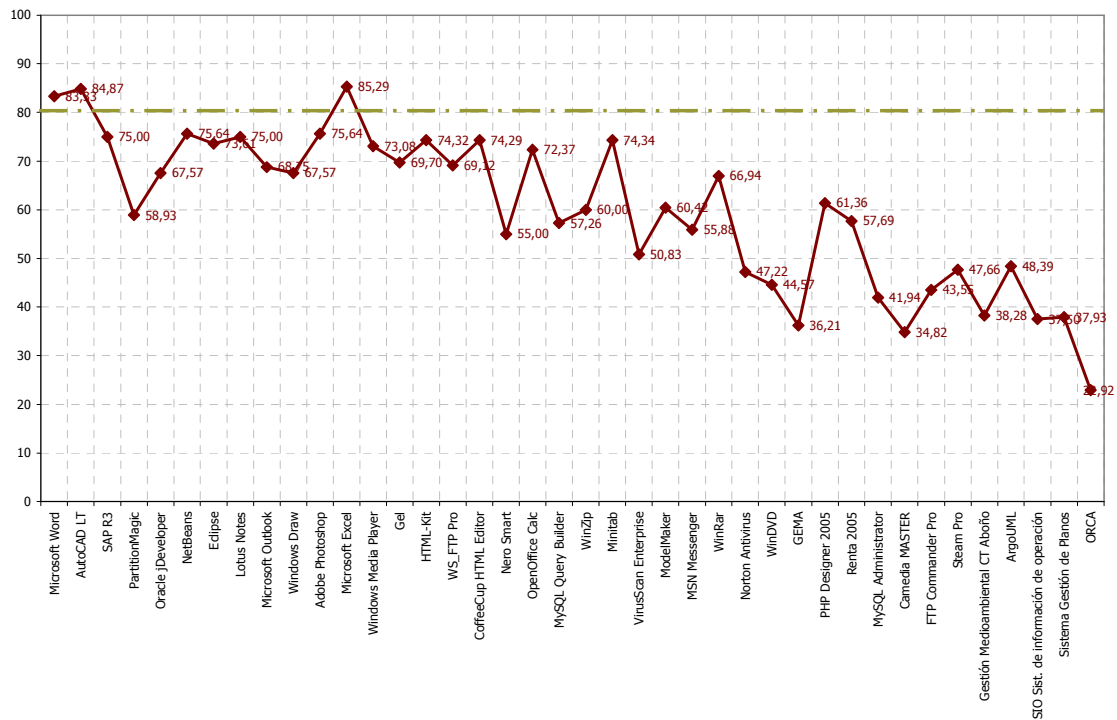


9 APENDICE 3. Gráficos de los resultados por cada heurística.

Visibilidad del estado del sistema

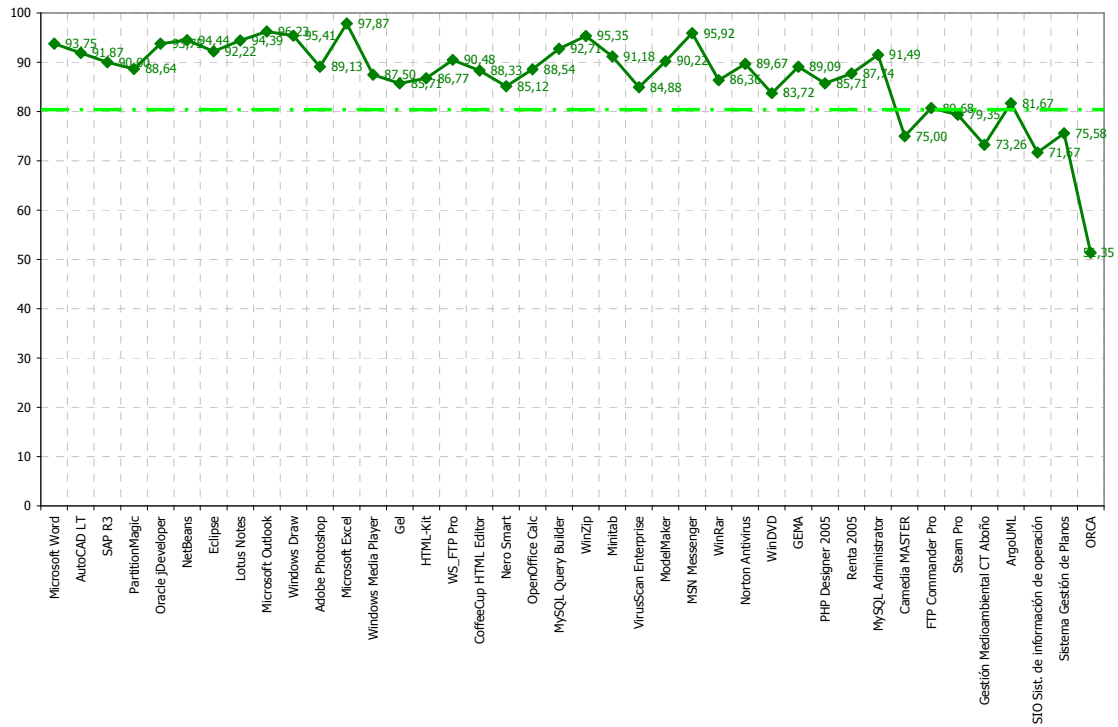


Control y libertad del usuario

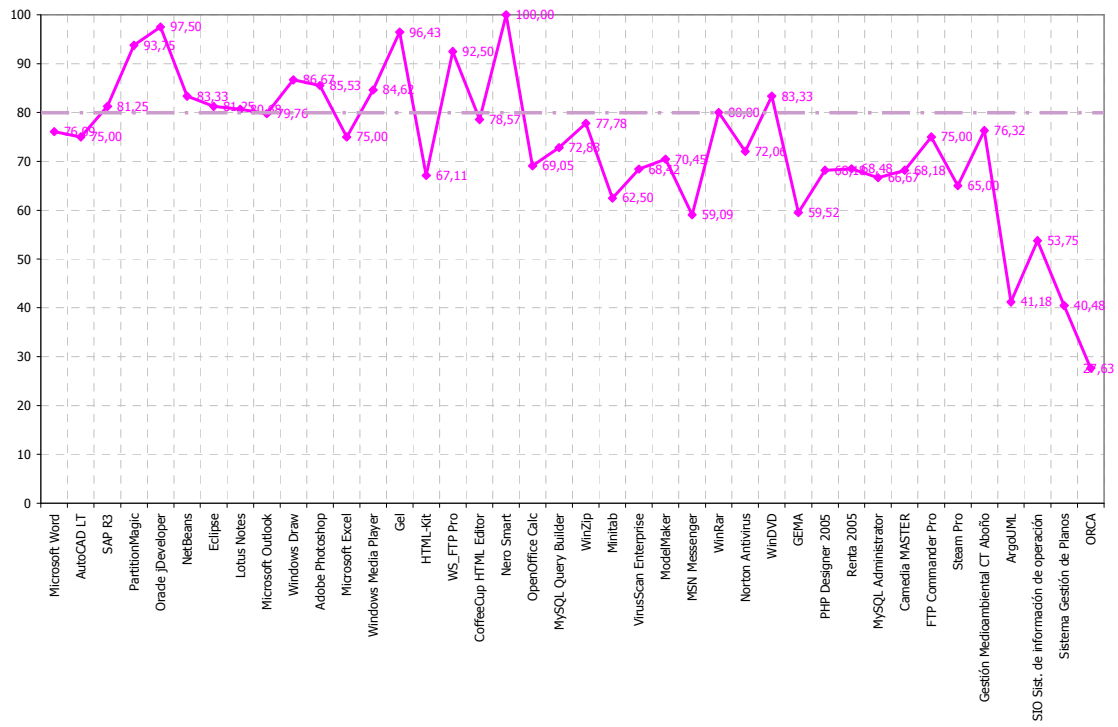




Consistencia y estándares

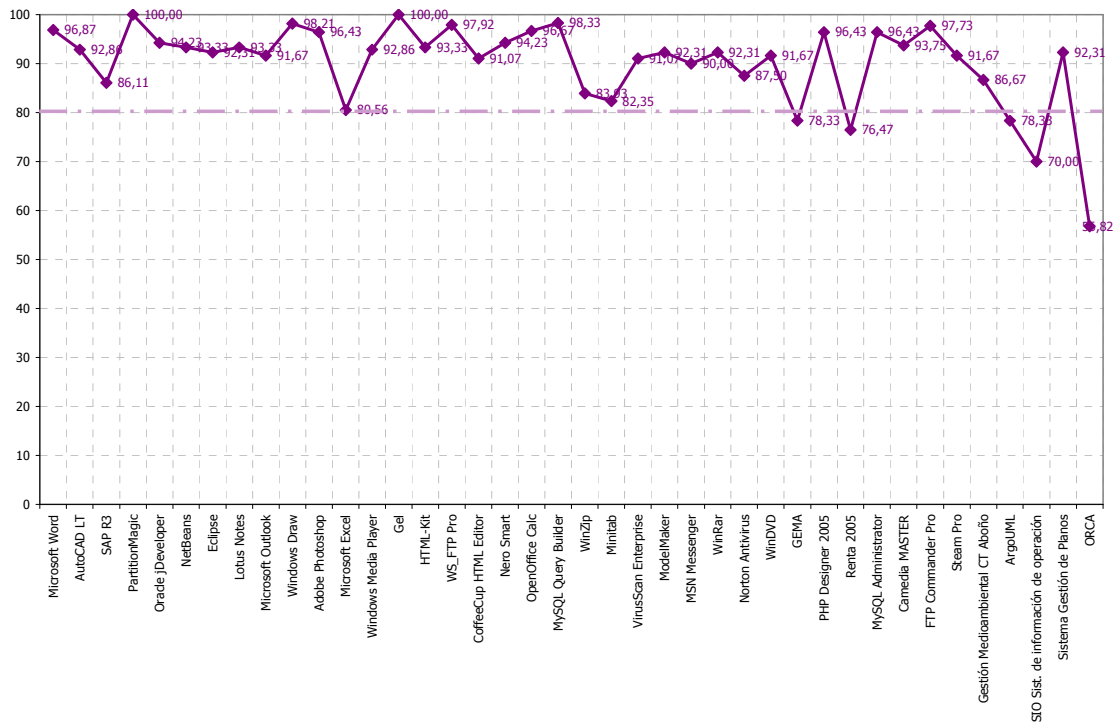


Prevención de errores

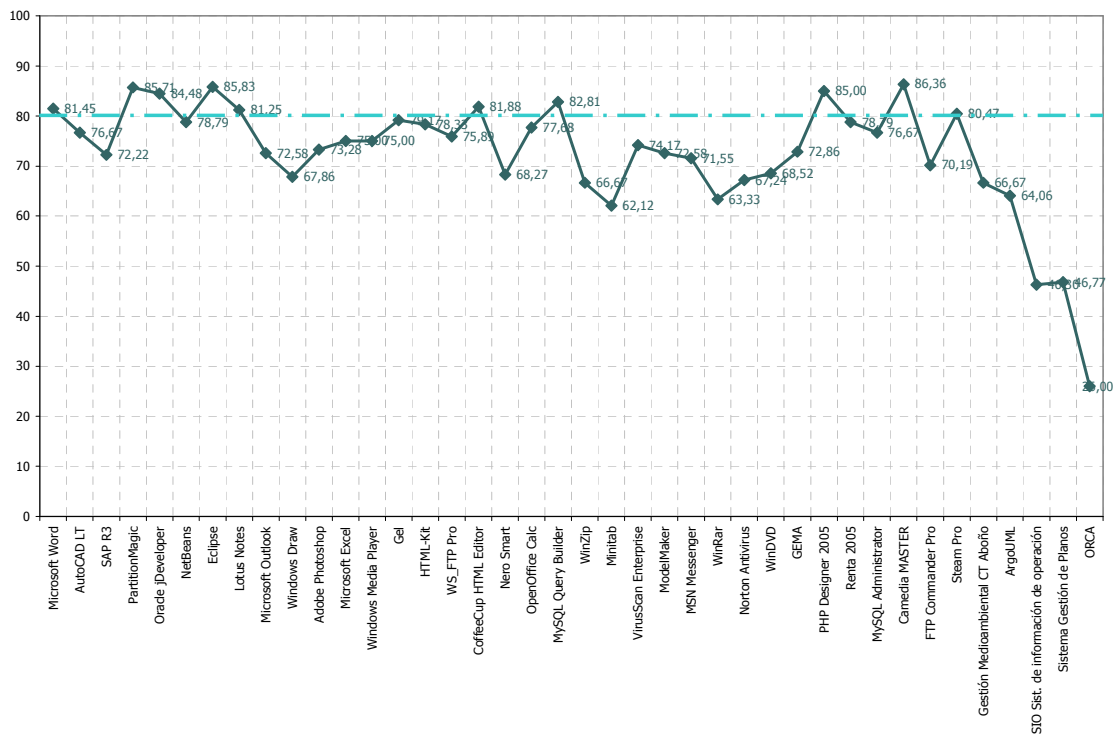




Correspondencia entre el sistema y el mundo real

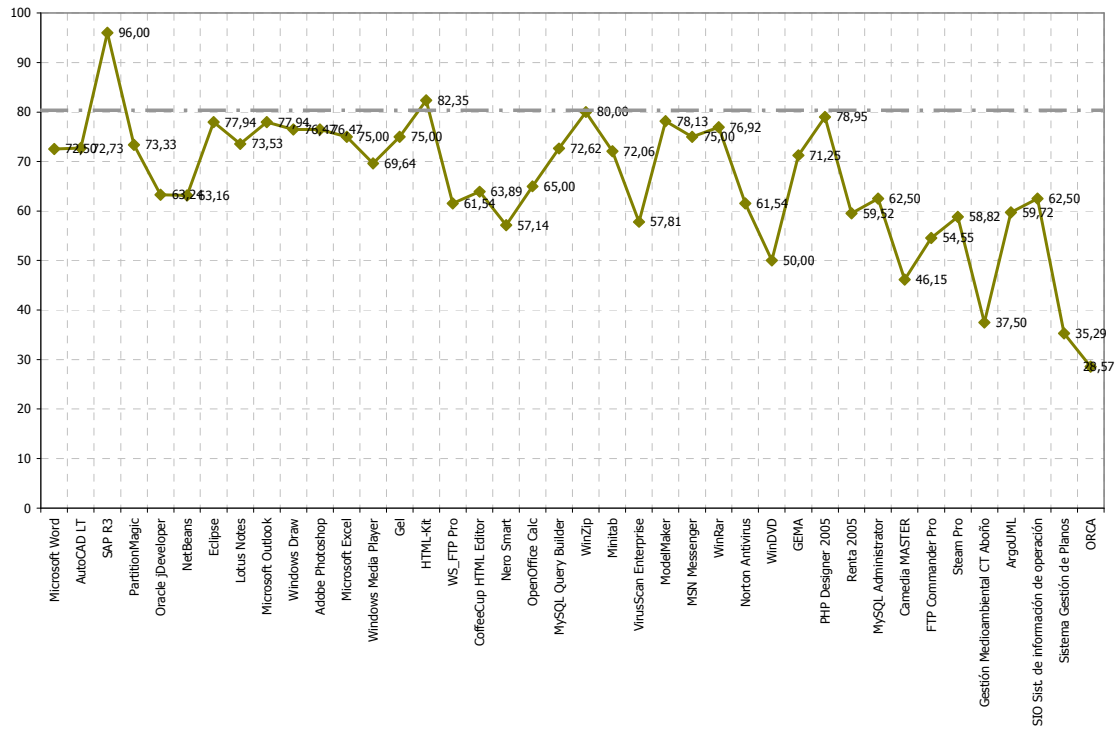


Reconocer antes que recordar

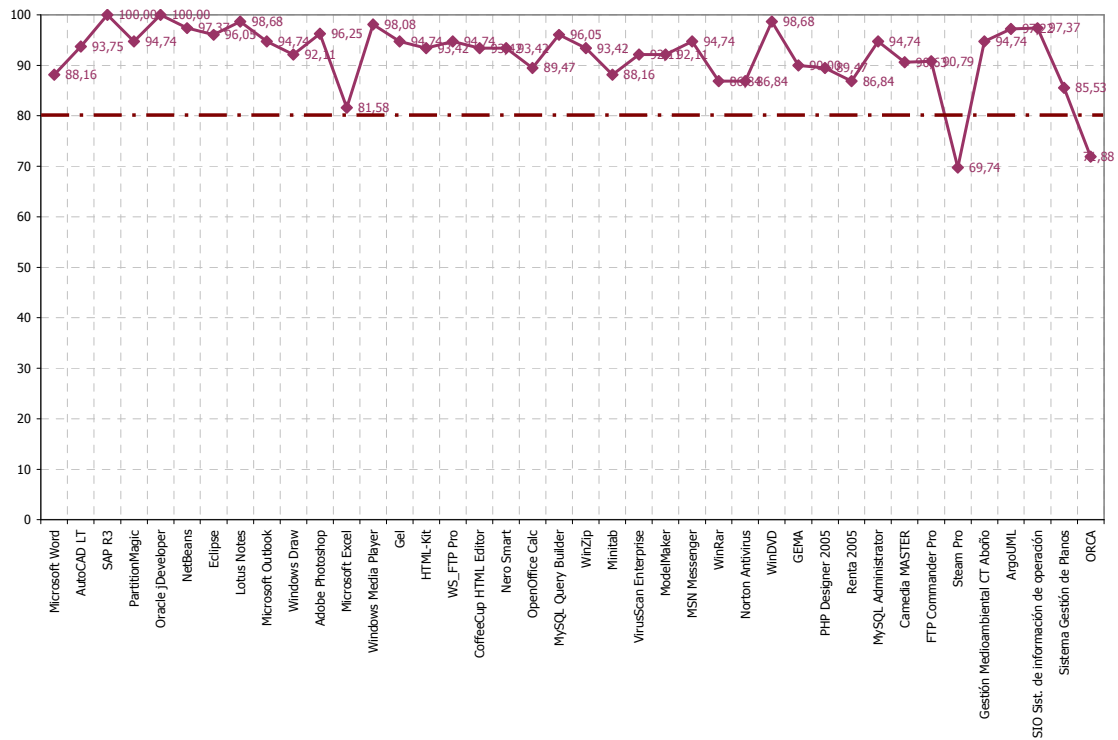




Flexibilidad y eficiencia de uso

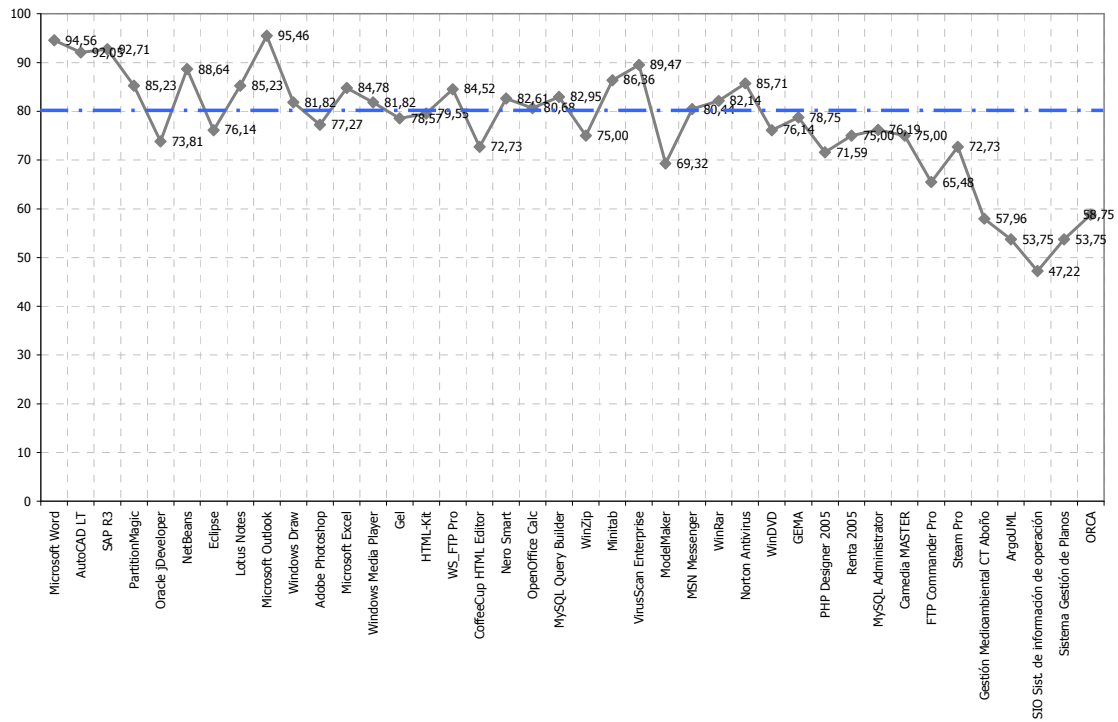


Estética y diseño minimalista

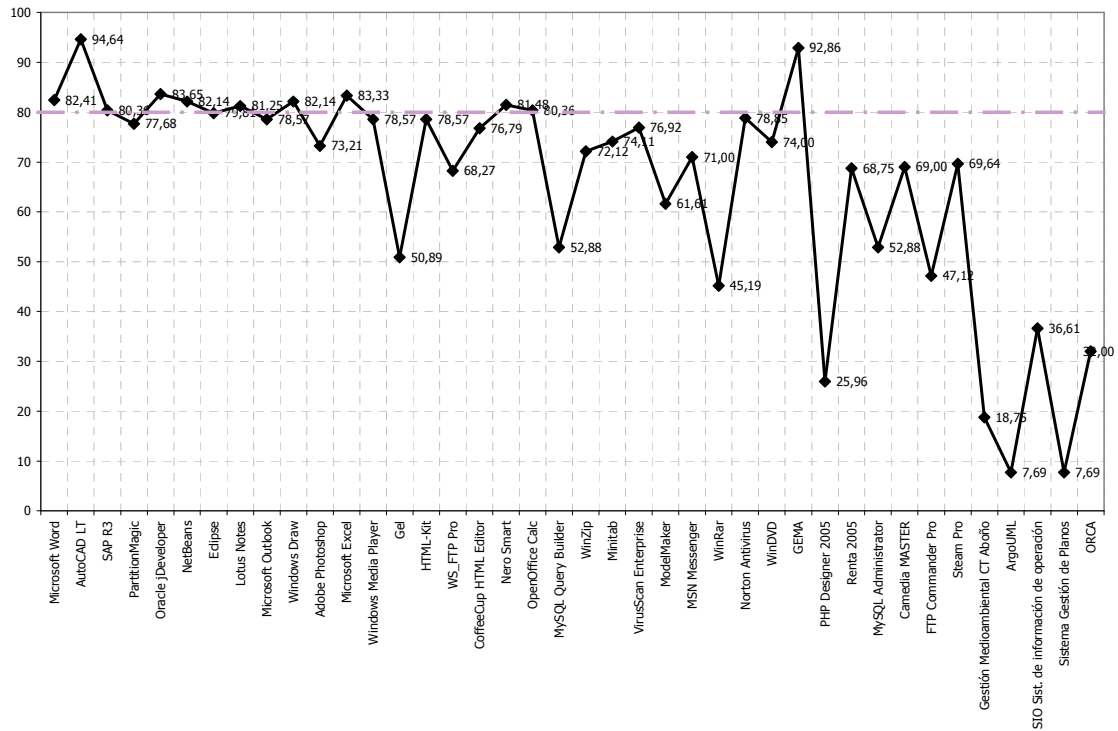




Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de errores



Ayuda y documentación





10. ALTERNATIVA A LOS CRITERIOS DE VALORACION DE PESOS

Se ha considerado otra posible forma de valorar los pesos a dar a cada heurística valorando cada elemento verificable de cada una de ellas en una escala de 0 a 100, en pasos de 10 en 10 (10, 20,...,100) y en función de su importancia relativa al contexto de la heurística.

El procedimiento seguido ha sido el siguiente:

Para cada heurística, se valora cada elemento verificable en función de su importancia relativa en el contexto de la heurística. Realizando lo anterior se han obtenido las siguientes valoraciones:



Nº	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10
1	30	20	30	30	80	80	40	90	10	10
2	40	20	20	40	40	10	10	70	20	60
3	20	30	30	40	60	20	60	30	30	60
4	20	60	50	30	70	60	70	20	30	70
5	30	10	50	70	80	40	20	90	20	80
6	10	80	50	70	20	10	90	70	10	80
7	10	90	50	40	60	10	90	20	20	80
8	30	100	50	60	60	10	10	10	20	20
9	50	30	50	60	90	20	10	20	20	20
10	80	20	40	60	50	70	70	10	20	10
11	40	50	40	90	80	10	10	20	80	90
12	40	60	70	90	70	80	10	90	90	60
13	40	80	10	90	80	50	20	90	70	90
14	50	40	20	30	90	10	20	90	50	70
15	50	10	40	70	60	60	20	100	40	50
16	30	80	40	80	30	70	40	80	90	90
17	10	30	50	70	40	70	20	70	90	90
18	80	50	70	90	10	90	20	20	90	30
19	80	40	60	100	10	90	10	30	90	40
20	70	40	60	80	90	50	30	60	50	20
21	60	80	70	90	40	60	80	50	20	20
22	50	30	30	90		20	80		50	70
23	40	80	60	100		20	30		100	80
24	60	50	70	10		80	30		50	10
25	90	90	90	70		70	30		90	90
26	90	60	40	90		80	70			90
27	20	90	10			30				90
28	10	90	70			90				90
29	90	40	50			10				
30	20	70	20			80				
31	70	20	50			80				
32	20	90	20			20				
33	10	40	70			20				
34	90	80	10			20				
35		60	30			20				
36		80	40			60				
37		50	10			80				
38		40	50							
39		10	50							
40		80	60							
41		90	70							
42			60							
43			40							
44			20							
45			30							
46			30							
47			90							
48			90							
49			20							
50			80							
51			90							
52			10							
53			80							
54			10							
55			50							
56			50							
57			90							
58			90							
59			20							



Seguidamente se han contado el número de elementos para cada puntuación obteniéndose la siguiente tabla:

Puntuación máxima	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10
10	5	3	6	1	2	7	6	2	2	3
20	5	4	7	0	1	8	6	5	7	4
30	4	4	6	3	1	1	4	2	2	1
40	5	6	7	3	3	1	2	0	1	1
50	4	4	13	0	1	2	0	1	4	1
60	2	4	4	3	4	4	1	1	0	3
70	2	1	7	5	2	4	3	3	1	3
80	3	8	2	2	4	7	2	1	1	4
90	4	6	6	7	3	3	2	5	6	8
100	0	1	0	2	0	0	0	1	1	0

A continuación se realizan los cálculos del total conseguido por cada heurística y su porcentaje sobre su puntuación máxima posible.

$$p_o = \sum_{i=1}^{10} c_i * d_i$$

donde:

$$d_i = 10, 20, \dots, 100$$

c_i = cantidad de elementos de cada puntuación

$$p_{\max} = n * 100$$

donde:

n = número de elementos verificables de cada heurística

P_{\max} = puntuación máxima posible de cada heurística



$$p_i = \frac{\frac{p_o}{p_{\max}}}{\sum_{i=1}^{10} \frac{p_o}{p_{\max}}}$$

donde:

p_i = peso de cada heurística

$i = 1, 2, \dots, 10$

$$w_i = p_i * 1000$$

donde:

w_i = peso aplicable al modelo de la métrica

Dividiendo ese porcentaje sobre la suma de todos ellos y multiplicando por 1000 para su adecuación a la escala de la métrica, da como resultado los siguientes pesos:

Puntuación obtenida total	1530	2260	2740	1740	1210	1750	990	1130	1250	1660
Puntuación máxima	3400	4100	5900	2600	2100	3700	2600	2100	2500	2800
% / Puntuación máxima	0,450	0,551	0,464	0,669	0,576	0,473	0,381	0,538	0,500	0,593
Peso obtenido	0,087	0,106	0,089	0,129	0,111	0,091	0,073	0,104	0,096	0,114
Σ de % / Puntuación máxima =	5,196									

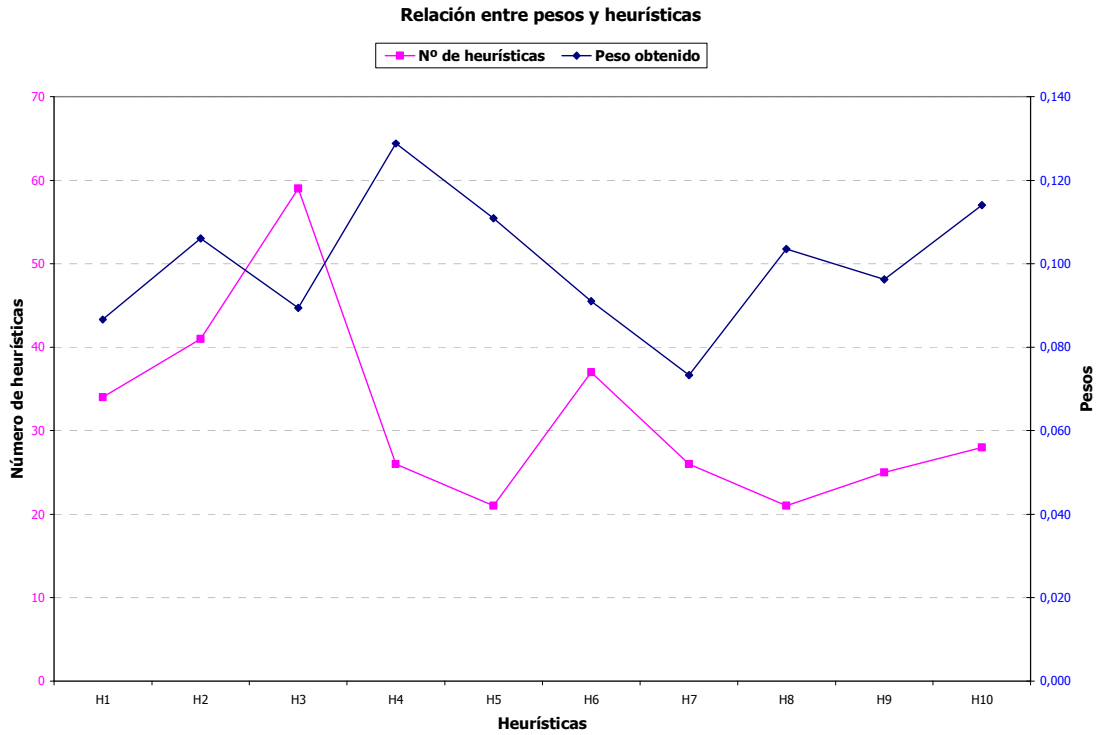
Los pesos obtenidos por este método y en el orden de las heurísticas [H1..H10]

son:

87, 106, 89, 129, 111, 91, 73, 104, 96, 114



Como demostración de que los pesos no están en relación con el número de elementos verificables de cada heurística, se presenta el siguiente gráfico:





11. BIBLIOGRAFIA

[Aedo 2004]

Aedo y otros

Sistemas multimedia: análisis, diseño y evaluación.

UNED

[Cuevas 2003]

Cuevas

Gestión del proceso software

Cerasa

[Duarte 2001]

Duarte Nuno Jadim Numes

Object Modeling for User-Centered Development and User Interface Design:

The Wisdom Approach

Tesis Doctoral

Universidad de Madeira

[Constantine 1999]

L. Constantine & L. Lockwood

Software for use. A practical guide to the models and methods of usage-centered design.

Addison-Wesley

[Minguet 2003]

Minguet, J.M y Hernandez, J.F.

La calidad del software y su medida

Cerasa

[Nielsen 1993]

Nielsen, J.

Usability Engineering

Academia Press: Boston, MA

[Piattini 2002]

Piattini, M. y García, F.M.

Calidad en el desarrollo y mantenimiento del software

RA-MA

[Pressman 2002]

Pressman, R.S.

Ingeniería del software: un enfoque práctico

5ª edición

McGraw-Hill



[Sanz 1996]

Marcos F. Sanz, Enrique J. Gómez, Francisco del Pozo

Boletín Factores Humanos nº 10, Artículo 7

ETSI Telecomunicación

Universidad Politécnica de Madrid

[Alba 2005]

María Elena Alba Obeso

Metodología de Medición y Evaluación de la Usabilidad en Sitios Web Educativos

Tesis Doctoral

Universidad de Oviedo



12. REFERENCIAS WEB

- [1] Jacob Nielsen
Uno de los sitios sobre usabilidad más importantes del mundo.
<http://www.useit.com>
- [2] James Hobart
Presidente de la empresa Classic System Solutions, Inc.
Principles of Good GUI Design
<http://www.classicsys.com>
- [3] Gina Ellis
Sitio web de la empresa Information & Design.
<http://www.infodesign.com.au>
- [4] IBM
Sitio web de la empresa IBM sobre el diseño de interfaz gráfica de usuario.
<http://www.ibm.com/ibm/easy/>
- [5] Turka Keinonen
One-dimensional usability – influence of usability on consumer’s product preference. Cap. 2
Helsinki 1998
<http://www2.uiah.fi/project/metodi/258.htm>
- [6] Alejandro Floría
Usabilidad y diseño centrado en el usuario
<http://www.usabilidad.org>
- [7] Eduardo Mercovich
Ponencia sobre Diseño de Interfaces y Usabilidad: cómo hacer productos más útiles, eficientes y seductores.
<http://planeta.gaiasur.com.ar/infoteca/disenio-de-interfaces-y-usabilidad.html>
- [8] ERGINOMIC GUIDELINES USER-INTERFACE DESIGN
Universidad de Cornell
<http://ergo.human.cornell.edu/ahtutorials/interface.html>



[9] A Checklist of Common GUI Errors Found in Windows, Child Windows and Dialog Boxes.

<http://www.csst-technologies.com/guichk.htm>

[10] Human Computer Interaction Laboratory

Ben Shneiderman

University of Maryland

<http://www.cs.umd.edu/~ben/>

[11] Usability Professionals' Associations

http://www.usabilityprofessionals.org/usability_resources/usability_in_the_real_world/roi_of_usability.html

[12] GRIHO

Universidad de Lerida

<http://griho.udl.es/mpiua/>



13. ABREVIATURAS

ISO	International Organization for Standardization
IEC	International Electrotechnical Commission
W3C	World Wide Web Consortium
ANSI	American National Standards Institute
IEEE	Institute of Electrical and Electronic Engineers
CEN	European Committee for Standardization
SUMMI	Software Usability Measurement Inventory
MUMMS	Measuring the Usability of Multi-Media Systems
WAMMI	Website Analysis and Measurement Inventory
SUS	System Usability Scale