

DESARROLLO DE INTERFACES

TÉCNICO EN DESARROLLO DE APLICACIONES MULTIPLATAFORMA

Usabilidad **concepto y normativa**

Existe una **extensa normativa** que recoge **reglas y estándares** para la aplicación de la usabilidad en multitud de entornos, definida por la **Organización Internacional de la Normalización** y cuyos fundamentos principales veremos aquí.

Concepto de usabilidad

Facilidad o dificultad de uso de un sitio web, de aplicaciones y herramientas o de cualquier otro entorno software que implique la interacción con un usuario.

Conjunto de parámetros que permiten establecer las **características deseables** sobre una mejor usabilidad. Se trata de diseñar sistemas **eficientes, efectivos, seguros, útiles, fáciles** de aprender y fáciles de recordar.

Jakob Nielsen: «Si no lo haces fácil, los usuarios se marcharán de tu web».

Características de la usabilidad:

- **Eficiencia de uso:** Esta característica hace referencia al **tiempo** que se requiere para **completar una acción** determinada. No resulta muy conveniente que un usuario tarde mucho tiempo en encontrar botones, menús, accesos... O en comprender un funcionamiento excesivamente complejo de la aplicación.
- **Facilidad de aprendizaje:** En relación con la anterior característica, el **tiempo empleado en conocer el funcionamiento** de una aplicación no puede ser demasiado amplio, ya que provocará que el usuario quiera **buscar otras** que **satisfagan** sus necesidades de forma **inmediata**.
- **Retención del tiempo:** En el caso de aplicaciones utilizadas de manera intermitente, es aconsejable que el usuario precise de un **menor tiempo de aprendizaje** que la **primera vez que accedió** a la herramienta.
- **Satisfacción:** Una de las características más **subjetivas**, puesto que indica el grado de satisfacción del usuario con respecto al sistema.
- **Tasa de error:** Será deseable que el número de errores cometidos por los usuarios de una aplicación sea lo menor posible. En concreto, aquellos errores que puedan estar **provocados por una excesiva complejidad** de la herramienta.

Principios para el diseño de sistemas interactivos

Se pueden establecer un conjunto de **principios de diseño** relativos a la **interacción** persona-computador. Según el libro **User Engineering Principles for Interactive Systems de Hansen** (1971), estos principios para el diseño de sistemas interactivos son, entre otros:

- **Conocer al usuario:** Es necesario analizar el comportamiento de los '**usuarios tipo**' que van a acceder a la aplicación que se desarrolla, puesto que, en función de estos, se implementarán unas **características** u otras.
- **Minimizar la memorización:** Suele ser aconsejable sustituir las entradas de datos por la **selección de ítems**. Por ejemplo, el uso de nombres aporta una mejor facilidad de uso que el empleo de números. De esta forma, se conseguirá un **comportamiento predecible** y proporcionará **acceso rápido** a la información práctica, y útil, del sistema.
- **Optimizar** las operaciones mediante la **rápida ejecución de operaciones comunes:** La **consistencia** de la interfaz y organizando y **reorganizando la estructura** de la información basándose en la observación del **uso del sistema**.
- **Facilitar buenos mensajes de error,** crear **diseños que eviten** los **errores** más comunes: Haciendo posible **deshacer** acciones realizadas y garantizar la **integridad** del sistema en caso de un fallo de software o hardware (¿backup?).

La **interacción Persona - Ordenador (IPO)** o, en inglés, **Human-Computer Interaction (HIC)** es «la disciplina que estudia el intercambio de información entre las personas y los ordenadores», cuyo objetivo es que el **intercambio y acceso a la información sea lo más eficiente posible**.

Normativa

ISO

La **Organización Internacional de la Normalización** (International Organization for **Standardization**, ISO), se encarga de la creación de normas y estándares cuyo objetivo principal es conseguir asegurar que **servicios y productos** presenten ciertos niveles de **calidad, eficiencia y seguridad**.

La ISO define la **usabilidad** como:

*La capacidad de un producto para ser **entendido, aprendido, usado** y resultar **atractivo** para el usuario cuando se usa bajo determinadas condiciones. Es decir, la usabilidad no antepone la facilidad de uso al diseño, ni el diseño a la facilidad para comprender su funcionamiento por parte del usuario, será una **combinación de diferentes aspectos clave**, que **podrán variar** en función del **tipo de aplicación** y del **grupo de usuarios** al que va dirigida.*

A continuación, vamos a hacer un recorrido sobre las principales **normas y estándares** enunciados por la ISO en relación a la **consecución de buenos parámetros de usabilidad**, con respecto al diseño y desarrollo de interfaces que favorecen la interacción entre el usuario y la aplicación.

En cuanto al **uso del producto, servicio o herramienta desarrollado**:

- **ISO/IEC 9126-1**: Estándar internacional relativo a la **ingeniería del software** en cuanto a la **calidad** del producto.

Se distinguen **subestándares**: **modelo** de calidad y **métricas** (de calidad de uso, **externas e internas**).

- **ISO/IEC 9241**: Se trata de una **guía de usabilidad** donde se recogen los **beneficios** relativos a las medidas de usabilidad evaluadas. En el siguiente apartado, la analizaremos más en detalle.

En cuanto al **diseño de la interfaz y la interacción** generado entre el usuario y la aplicación son:

- **ISO/IEC 14915**: Estándar internacional relativo a la **ergonomía del software** elaborado con respecto a la interfaz multimedia.
- **IEC TR 61997**: Se recogen un conjunto de **guías de interfaz multimedia** para **usuarios** en equipos de uso general.

Normas y estándares de usabilidad. ISO 9241

El estándar ISO/IEC 9241 es uno de los **más amplios** y recoge múltiples subapartados que organizan de forma clara todas las **normas** relativas a la **usabilidad** en el desarrollo de **sistemas interactivos**.

ISO 9241-**10**: Principios de **diseño de los diálogos** que se han de generar entre el usuario y el sistema.

ISO 9241-**11**: **Ergonomía** de la interacción entre el usuario y el sistema.

ISO 9241-**12**: **Organización y disposición** de la información.

ISO 9241-**13**: **Modelado y diseño** de las **ayudas** implementadas en la aplicación para el usuario.

ISO 9241-**14**: Diseño de los **diálogos** producidos por los diferentes elementos que modelan un **menú**: cómo se han de modelar y qué texto poner en botones, casillas, etc.

ISO 9241-**15**: **Diálogos** de lenguaje de **órdenes**.

ISO 9241-**16**: **Requisitos ergonómicos** para el correcto diseño de aplicaciones para trabajos de **oficina** con pantallas de **visualización de datos**.

ISO 9241-**17**: Diseño de **formularios**.

Otros estándares propios de este tipo de desarrollo:

- **ISO 13407**: **Diseño de aplicaciones e interfaces** en sistemas interactivos, poniendo el foco en el **usuario**.
- **ISO/TR 16982**: **Métodos ergonómicos** presentes en las distintas **fases del ciclo de diseño y desarrollo**.

Diferencia entre usabilidad y accesibilidad

Accesibilidad: especifica en qué grado un producto, o en el caso del software una interfaz, está **disponible** para ser usada por el **mayor número de personas** posible. Facilidad para **acceder a una interfaz**. Suele enfocarse a personas con **discapacidades** o limitaciones en su uso, y en los **derechos de uso y acceso**. En realidad, se debe recordar que una interfaz que sea accesible **beneficia a todas las personas**.

Usabilidad: Grado en que una interfaz puede ser **utilizada para lograr objetivos** concretos con **eficacia, eficiencia y satisfacción** en un determinado contexto de utilización. De esta manera una interfaz será usable si se logra que su **utilidad** sea **mayor** que el **esfuerzo** necesario para aprender su manejo y el tiempo que será necesario invertir en su uso (fácil de usar).

Medidas de usabilidad: Satisfacción, efectividad y eficiencia

Son **herramienta clave para la evaluación** de este concepto en el desarrollo de interfaces adecuadas para la aplicación y los usuarios a los que va dirigida.

Los **test de usabilidad** evalúan:

- La **facilidad de uso** por parte de un usuario (no experto).
- Que la funcionalidad desarrollada **cumple con la finalidad** de la aplicación. Si el desarrollo resulta intuitivo para una persona, pero no cumple sus expectativas en cuanto al objeto de desarrollo, tampoco estará cumpliendo los criterios de usabilidad.

Los test de usabilidad **no son** herramientas de opinión **subjetiva**, ya que se deben desarrollar de forma exhaustiva para que, de **manera objetiva**, se evalúen todos los **parámetros deseados**.

Por lo tanto, en el diseño de este tipo de test, se deben contemplar ciertas **métricas comunes** que devuelven de manera fiable la usabilidad presenten en el sistema evaluado.

Satisfacción

Es clave para la evaluación de la aplicación. La obtención de un nivel bajo de satisfacción, unido a un buen diseño de cuestionario, permitirá añadir las modificaciones oportunas al sistema.

Métricas de satisfacción:

- ➔ Calificación de **satisfacción** del usuario sobre la aplicación.
- ➔ Frecuencia de **reutilización** de la aplicación.
- ➔ Calificación relativa a la **facilidad** de aprendizaje.
- ➔ Medida de **uso voluntario** de la aplicación.

Efectividad

Grado de éxito de una aplicación, si **cumple con la funcionalidad** para la cual ha sido desarrollada. Este atributo está estrechamente ligado a la facilidad de aprendizaje de la herramienta. Será necesario contemplar distintos escenarios.

Métricas de efectividad:

- Cantidad de **tareas relevantes completadas** en cada uno de los intentos.
- Número de acceso a la **documentación**, al **soporte** y a la **ayuda**.
- Cantidad de **funciones aprendidas**.
- Número de **usuarios capaces de aprender** las características del producto.
- Cantidad y tipos de **errores tolerados** por los usuarios.
- Cantidad o porcentaje de **palabras leídas correctamente**.

Eficiencia

Tiempo que es necesario **para completar una tarea** con el software desarrollado (a más eficiencia más tareas completadas).

Las métricas definidas para este atributo están **basadas**, sobre todo, en el **primero de los intentos**, puesto que, en ese momento, aún no se tiene demasiado conocimiento sobre la aplicación. Si en este intento los **valores son buenos**, en los **restantes ofrecerán grandes resultados**, que se valoran en las métricas de efectividad.

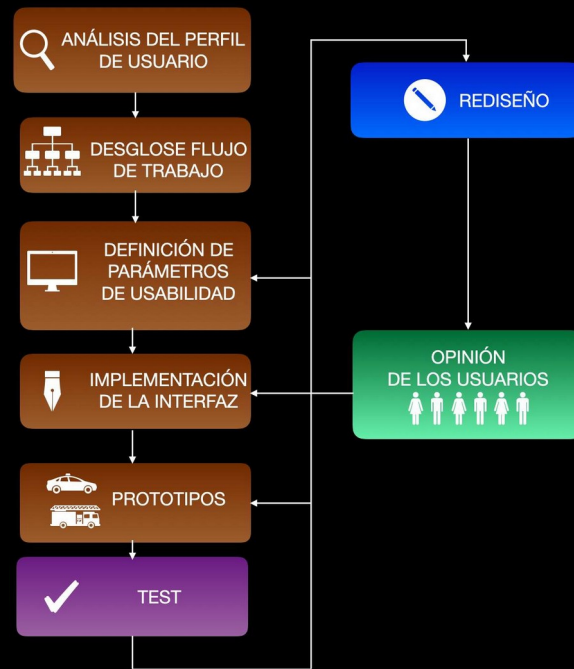
Métricas de eficiencia:

- Tiempo **productivo**.
- **Tiempo** para **aprender el funcionamiento**.
- **Tiempo** requerido en el **primer intento** para completar la **funcionalidad** evaluada.
- **Eficiencia** relativa al **primer intento**.
- **Errores persistentes**.
- Tiempo necesario para **aprender de nuevo** la funcionalidad del producto **pasado un tiempo** desde su anterior uso.

Diseño y realización de pruebas de usabilidad

Requiere de un **algoritmo**.

El **diseño de los test de usabilidad** cumple un papel activo **en el proceso de implementación** de la aplicación.



El flujo no solo se ejecuta una vez, es decir, cuando se completa el primer diseño y se realizan las pruebas y test necesarios, puede extenderse todo lo necesario.

Estos **pasos necesarios** para un desarrollo completo son:

1. Se realiza un **análisis** completo del **perfil** de **usuario** de la aplicación.
2. Se desglosan los diferentes **flujos de trabajo** del usuario de la herramienta (funcionalidades).
3. Se diseñan los **parámetros de usabilidad** necesarios.
4. Se comienza a implementar el **diseño de la aplicación**, tanto a nivel de **funcionalidad** como de **interfaz**.
5. Tras este primer desarrollo, se obtienen los llamados **prototipos**, sobre los cuales se comienza a aplicar los **test de usabilidad**, creados **en base a los parámetros** definidos en el punto 3.
6. En base a estos resultados, se irá **rediseñando y readaptando** las diferentes versiones del software, hasta obtener un **resultado final** que puede llevarse a producción, donde será distribuida de la forma pertinente.

Pruebas con expertos (evaluación heurística)

Las **pruebas realizadas por expertos** realizan una inspección del software basada en su **conocimiento** de la aplicación, así como en un listado de todos los **posibles escenarios** que se pueden contemplar.

Las **pruebas realizadas por usuarios** no llevan a cabo una evaluación tan profesional, ya que se basa en otra serie de criterios.

Los **expertos** realizan la conocida **evaluación heurística** o **método de inspección**: analizan toda la aplicación e identifican los **problemas existentes** o algunos **que pudieran ocurrir** antes de llevar a producción una aplicación. El coste de **corrección después** de implantar cualquier software o herramienta es mucho **más elevado** que si se realiza antes.

La **evaluación heurística se divide en dos partes**:

- Evaluación **detalle**: Se realiza un análisis **exhaustivo**.
- Evaluación de **alto nivel**: Se analiza el funcionamiento de forma **general**.

Número de expertos: Suelen ser **cuatro**, que actuarán de forma **independiente**, evaluando cada **funcionalidad** y analizando todos los posibles **problemas** surgidos.

Finalmente, se realiza un **informe de manera conjunta**. Es deseable que estos tengan bastante **experiencia**, tanto en la detección de **errores de usabilidad** como en el manejo de la propia herramienta; de esta forma, **detectarán más problemas** que personal ajeno al producto y al análisis de estos.

Estas pruebas se realizan en **cualquier momento** del proceso de **desarrollo**.

Es aconsejable que se realicen **antes** de las **pruebas con usuarios**, puesto que, cuando se hacen estas, el prototipo tiene que estar lo más avanzado posible para que la **versión probada** por usuarios sea **lo más cercana a la realidad**.

Pruebas con usuarios

Se basan en el análisis y evaluación de una herramienta o aplicación software mediante un grupo de usuarios reales que pueden detectar **errores** que los **expertos no han sido capaces de encontrar**.

Los métodos de test con usuarios se basan en el uso de cuestionarios tipo. Según el Diseño Centrado en el Usuario (DCU), los test de usuario se basan en pruebas que **observan la forma de interacción** de los usuarios con el **producto objeto del test**.

Número de usuarios que participan en este test: Aconsejable **al menos, 15**, para poder garantizar una tasa de **detección cercana al 100%**. La elección de estos debe basarse en los **perfiles** a los que está **dirigida la aplicación**. No tendrá sentido probar una aplicación para la gestión logística de un almacén con un grupo de usuarios que no tienen ninguna vinculación a este tipo de áreas.

Las **pruebas** se realizan **de forma individual** y se deben tener en cuenta todas las observaciones que se tomen, desde la primera toma de contacto, hasta la realización de la prueba completa.

Algunos **criterios de diseño** son:

- Pruebas **razonables**, es decir, que un **usuario real** realizará.
- Pruebas **específicas**, es aconsejable realizar pruebas concretas y **no muy genéricas**.
- Pruebas **factibles**, que **pueden realizarse**, no se trata de un examen que los usuarios no deben superar.
- **Tiempo de realización razonable**.

Tipos de test de usabilidad

Cuatro tipos de test de usabilidad. La elección del más adecuado se hará en base al tipo de estudio:

- Test de uso **pautado**: En este caso, un responsable se encarga de **monitorizar todas las pruebas** que se hacen en base a un listado previamente diseñado sobre un **prototipo muy cercano** a la versión real.
- Test de uso **descontextualizado**: Implementa un proceso similar al anterior, sobre un **prototipo no real**.
- Uso **natural**: Estas pruebas se realizan sobre la **versión final** y con una interacción también similar a la real, **sin ningún moderador** que pauté las acciones.
- **Híbridos**: Combinación entre cualquiera de las descritas.

Pautas para el diseño de las pruebas de usabilidad:

1. Se definen los **objetivos** de las pruebas.
2. Se diseña el **formato** y **tipo de datos** que se tomarán y analizarán en el estudio (cuantitativos, cualitativos, mixtos, presencial o en remoto...).
3. Se realiza el **diseño de las pruebas** que se van a realizar.
4. Se determina el **número de personas** que participan en el estudio. Estos se escogen adecuando sus perfiles al **tipo de producto** que se va a evaluar. En el caso de los **expertos**, será deseable que tengan cierta **experiencia** en evaluaciones de usabilidad.
5. Se escogen las **métricas** que se van a recoger en los **test de usabilidad**..
6. Se **implementa** el plan de test: La guía de las pruebas que se van a realizar, el tiempo que se va a destinar a cada una de ellas, etc. Todos los **aspectos relativos** a estas pruebas tienen que estar cuidadosamente escogidos para abarcar el **mayor número posible de casuísticas** y prever cualquier tipo de **contratiempo** que pudiera **entorpecer la satisfacción** del usuario con respecto a la aplicación.