

Ingeniería de la Usabilidad

Desarrollo de Sistemas Interactivos

Contenido

1. Usabilidad
2. Experiencia de usuario
3. Ingeniería del software y usabilidad
4. Ingeniería de la usabilidad

1.1 Usabilidad

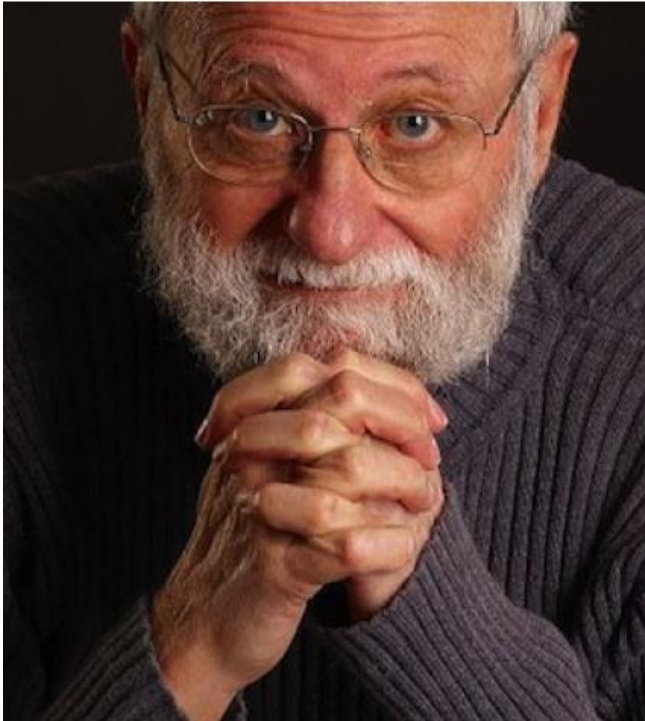
Según el estándar [ISO 9241 \(parte 11: Orientaciones sobre Usabilidad\)](#), este término se puede definir como:

“la cualidad por la que un producto, sistema o servicio puede ser usado por un *usuario específico* para conseguir unos *objetivos específicos* con ***eficiencia, efectividad y satisfacción en un contexto de uso específico***”.

Los tres atributos que miden la usabilidad:

- Efectividad:** Es la precisión y el grado de completitud con la que el usuario es capaz de satisfacer sus objetivos.
- Eficiencia:** Los recursos y el tiempo empleados en relación con el grado de precisión y completitud con el que se han satisfecho los objetivos.
- Satisfacción:** Actitud positiva hacia el uso del producto, ausencia de frustración o incomodidad al usarlo.

1.2 Don Norman



Es el autor de uno de los libros más influyentes en el campo del diseño y la usabilidad: “[The Design of Everyday Things](#)”. En él se acuña términos como la **experiencia de usuario** (del que hablaremos más adelante) o **diseño centrado en el usuario**, en el que habla sobre una forma de diseñar centrada en las necesidades del usuario, dejando de lado otros elementos secundarios. Algunas de las pautas que da (y de las que hablaremos más adelante) son:

- Simplificar la estructura de las tareas.
- Hacer las cosas visibles.
- Hacer “mapeos” comprensibles
- Utilizar el poder de las restricciones
- Diseñar aprendiendo de los errores

1.3 Alan Cooper

Aunque es arquitecto, [Alan Cooper](#) ha sido conocido como un gran diseñador de software y programador.

Es el padre del Visual Basic y, en el campo que nos interesa, fue muy crítico con el diseño de software. Echaba en falta que durante el diseño no nos preguntáramos **cómo va a interactuar el usuario con el software**.

Tras trabajar en Microsoft fundó su propia consultoría de software donde comenzó con el diseño de software más dirigido al usuario y donde comenzó a establecer metodologías de diseño como **el diseño dirigido por objetivos** y el uso de personas como herramientas de diseño. Fue también de los primeros en acuñar el concepto **diseño de interacciones** (interaction design).



1.4 Jakob Nielsen

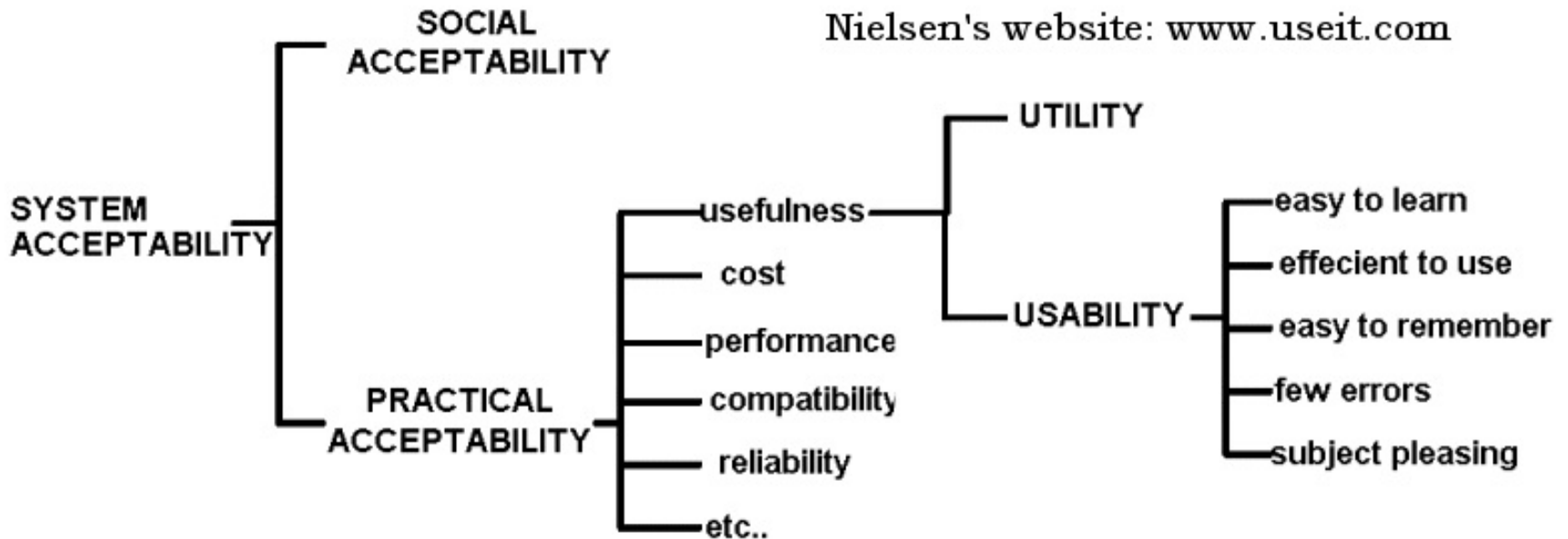
[Jakob Nielsen](#) es una de las personas más respetadas en el ámbito de la usabilidad y cofundador, junto con Don Norman, del Nielsen Norman Group. Ha trabajado para Sun e IBM, entre otras. Es uno de los mayores expertos en usabilidad, sobre todo en páginas web, y ha definido lo que se conoce como **evaluación heurística de interfaces**, una de las técnicas que estudiaremos y que nos ayudan a evaluar la usabilidad de los sistemas interactivos.



1.4 Usabilidad (Nielsen's)

La usabilidad no existe de manera aislada ni implica la aceptación general de un sistema.

Nielsen's website: www.useit.com



Nielsen's Taxonomy of System Acceptability

1.4 Usabilidad (Nielsen's)

Nielsen define hasta 5 atributos para la usabilidad:

- **Facilidad de aprendizaje** (Learnability): Cómo de fácil es para un usuario novato realizar una tarea en el sistema.
- **Eficiencia** (Efficiency): Cómo de rápido puede realizar una tarea un usuario experto.
- **Memorabilidad** (Memorability): Cómo de fácil es recordar cómo se usa un sistema tras haber pasado un tiempo sin usarlo.
- **Errores** (Errors): Cuántos errores comete el usuario, cómo de graves son y cómo de fácil es recuperarse de ellos.
- **Satisfacción** (Satisfaction): Cómo de agradable es usar el sistema desde el punto de vista del usuario.

1.5 Usabilidad

La usabilidad está relacionada con los buenos diseños por lo que no tener en cuenta los conceptos de usabilidad a la hora de diseñar un sistema tiene consecuencias: tiempo, dinero (sobre todo el de la empresa que vende el sistema) e, incluso, vidas.

La usabilidad afecta a cómo un usuario percibe un sistema:

- un sistema usable “vende”;
- un sistema no usable hace que el usuario lo “deseche”.

Pero hay que tener en cuenta que es un criterio subjetivo y que depende del tipo de usuario al que va dirigido. Distintos usuarios valorarán más unos atributos que otros:

- Los usuarios novatos valoran la facilidad de aprendizaje.
- Los usuarios esporádicos valoran la memorabilidad.
- Los usuarios expertos valoran la eficiencia.

2. Experiencia de usuario

User Experience (UX) es el conjunto de todas las interacciones que un usuario tiene con un producto.

Pretende ir un paso más allá de la usabilidad: no solo se centra en los **aspectos objetivos del sistema** (su funcionalidad) sino también con todos los demás **aspectos relacionados con los sentimientos** que genera la interacción con el sistema.

El término fue acuñado por Don Norman cuando era el Vicepresidente del Grupo de Tecnologías Avanzadas de Apple, ya que consideraba que el término usabilidad no cubría ciertos aspectos relacionados con las emociones que el usuario podía experimentar al interactuar con un producto.

El diseño centrado en la experiencia del usuario va más allá de la usabilidad y tiene en cuenta otros elementos.

USER EXPERIENCE DESIGN DIAGRAM



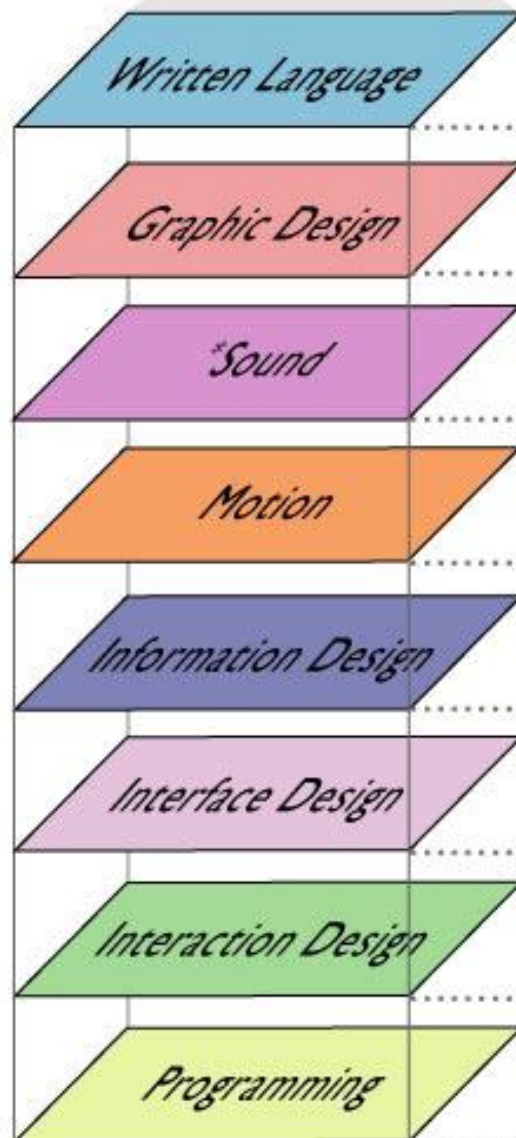
Language Interaction Motion Graphic Interface Design



code>
<object>



- Conscious Awareness +



English, Spanish, Mandarin, etc.

Shape, symbolism, line, color, spacial composition, texture, dimension, and other facets of visual rendering.

Music or spoken word (a.k.a Voice Over / V.O.) audio.

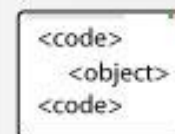
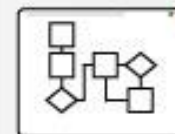
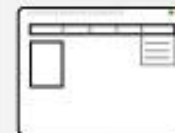
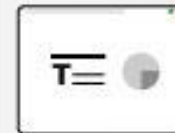
Animation, change, motion, time, rhythm, calculus.

Textual style, graphics, and composition for information structure, meaning, relationship and user comprehension.

Graphical and information design elements utilized to indicate controls for data manipulation.

Task flow, system flow/behavior, and human comprehensibility of controls provided by the user interface.

"Front-end" (client executed) or "back-end" (server executed) code for data input, processing, and retrieval.



The mind's eye does not naturally distinguish between individual elements that comprise an interactive system. Parts of the interactive communications / software vernacular are not experienced separately by the user, but as a complete synthetic language which is apprehended and used as a unified whole.

User Experience Design is the art and science of integrating all of the various elements that comprise an interactive system so that

- 1) The user's needs, limitations, goals, desires, and expectations are served
- 2) The publishing organization's objectives are served as a result of serving the user's (#1)
- 3) The whole is greater than the sum of its parts

*Sound is seldom used in contemporary web app's but is a multimedia element common to other types of web-based software, thus important to UX.

2.1 Experiencia de usuario

La experiencia de usuario habla de hacer que el usuario se sienta a gusto con el sistema, que le divierta (incluso [que le haga feliz](#)) y que la experiencia de interactuar con él sea excitante.

El *fluir* (o [flow](#)) es un concepto de psicología que se está volviendo popular dentro del HCI. Es un estado mental muy intenso en el que el usuario se encuentra completamente concentrado en la tarea en la que está realizando y disfrutando. Algunos diseños pretenden inducir al usuario en esta experiencia de fluir.

En ocasiones, la experiencia de usuario no va de la mano de la usabilidad sino que son opuestas e incluso pueden ser incompatibles. Por ejemplo, la mayoría jugadores hardcore de videojuegos disfrutan más con la experiencia de un mando de PlayStation o Xbox (que no son fáciles de usar) que con las acciones intuitivas que se pueden realizar con el mando de la Wii o con una Kinect.

2.2 Experiencia de usuario

La experiencia de usuario puede ir [más allá de la interfaz de usuario](#). Es la relación entre el usuario y el producto completo, la marca.



© Apple, Inc. All rights reserved.



Image courtesy of Jason Garber on Flickr.



© Apple, Inc. All rights reserved.



Image courtesy of YU-TA LEE on Flickr.



© source unknown.
All rights reserved.

3 Ingeniería del software y usabilidad

Los procesos de Ingeniería del Software tradicionales se basan en modelos de proceso orientados a conseguir programas a partir de un conjunto de especificaciones formales requeridas (habitualmente) por un cliente.

Típicamente la Ingeniería del Software incluye los siguientes pasos:

1. Captura de requisitos y especificación formal (planificación)
2. Análisis
3. Diseño (de la arquitectura)
4. Implementación
5. Pruebas
6. Mantenimiento

3.1 Ingeniería del software y usabilidad

“Los procesos de Ingeniería del Software se centran en resolver los problemas de los ingenieros, no de los usuarios.”

Los procesos de Ingeniería del Software se centran en aliviar los problemas típicos que pueden hacer fracasar un proyecto, tales como la falta de comunicación con los clientes, los costes de desarrollo, el diseño de arquitecturas mantenibles y la capacidad de evaluación de un proyecto.

Todos estos procesos son muy valiosos y necesarios para que los ingenieros sean capaces de desarrollar grandes aplicaciones software. Pero si queremos que estas aplicaciones sean usables y aceptadas por los usuarios es necesario corregir los errores que incluyen estos modelos de proceso.

3.2 Errores de la I.S. frente al usuario

Las tareas no son objetivos:

En general, estos procesos se centran en la viabilidad y la calidad de la tecnología desarrollada pero dejan de lado el conocimiento del usuario. Describen qué tareas se han de realizar (casos de uso) y desarrollan procesos para completarlas. Pero no suelen tener en cuenta cómo se realizan las tareas, quién las realiza ni cuáles son los objetivos al realizar dicha tarea:

- El objetivo de un sistema es el propósito por el que lo usamos.
- Las tareas son las distintas formas en las que conseguimos objetivos.

3.2 Errores de la I.S. frente al usuario

Los casos de uso no consideran de verdad al usuario:

Los casos de uso describen en detalle una actividad con todos sus pasos y sirven de contrato de qué debería hacer la aplicación. Desde el punto de vista de la usabilidad el análisis basado en casos de uso adolece de los siguientes problemas:

- Se limitan a cubrir tareas, replicar los procesos existentes en la empresa y no contemplan los objetivos del usuario final.
- El diseño basado en casos de uso da lugar a flujos lineales de ejecución, violando los principios de libertad y control del usuario, de los que hablaremos más adelante.

3.2 Errores de la I.S. frente al usuario

Los modelos de proceso formales dan lugar a malas interfaces

Los procesos formales requieren partir de requisitos formales, que dan lugar a especificaciones formales sobre las que aplicar procesos formales de diseño. Esto invita de forma natural a los análisis basados en casos de uso.

A partir de dichos casos de uso, se diseñan los flujos, y a partir de los flujos se diseña la arquitectura. Sobre la arquitectura se van implementando los puntos de función, y para cada punto de función se coloca un elemento de control en la interfaz.

Los modelos de proceso formales dan lugar a diseños centrados en la implementación.

4 Ingeniería de la usabilidad

La Ingeniería del Software tradicional sigue siendo necesaria, pero resulta insuficiente. Los problemas generales de usabilidad se deben comúnmente a tres razones:

- Ignorancia sobre el usuario.
- Conflicto de intereses entre cubrir las necesidades del usuario y las prioridades de desarrollo.
- Falta de un proceso para comprender las necesidades del usuario.

La **ingeniería de la usabilidad** (o usability engineering) es la disciplina que define el proceso ingenieril para crear software usable (de acuerdo a los criterios de usabilidad que hemos tratado anteriormente), que nos ayuda a comprender a los usuarios y a diseñar sistemas interactivos siguiendo un diseño centrado en el usuario.

4.1 Ingeniería de la usabilidad

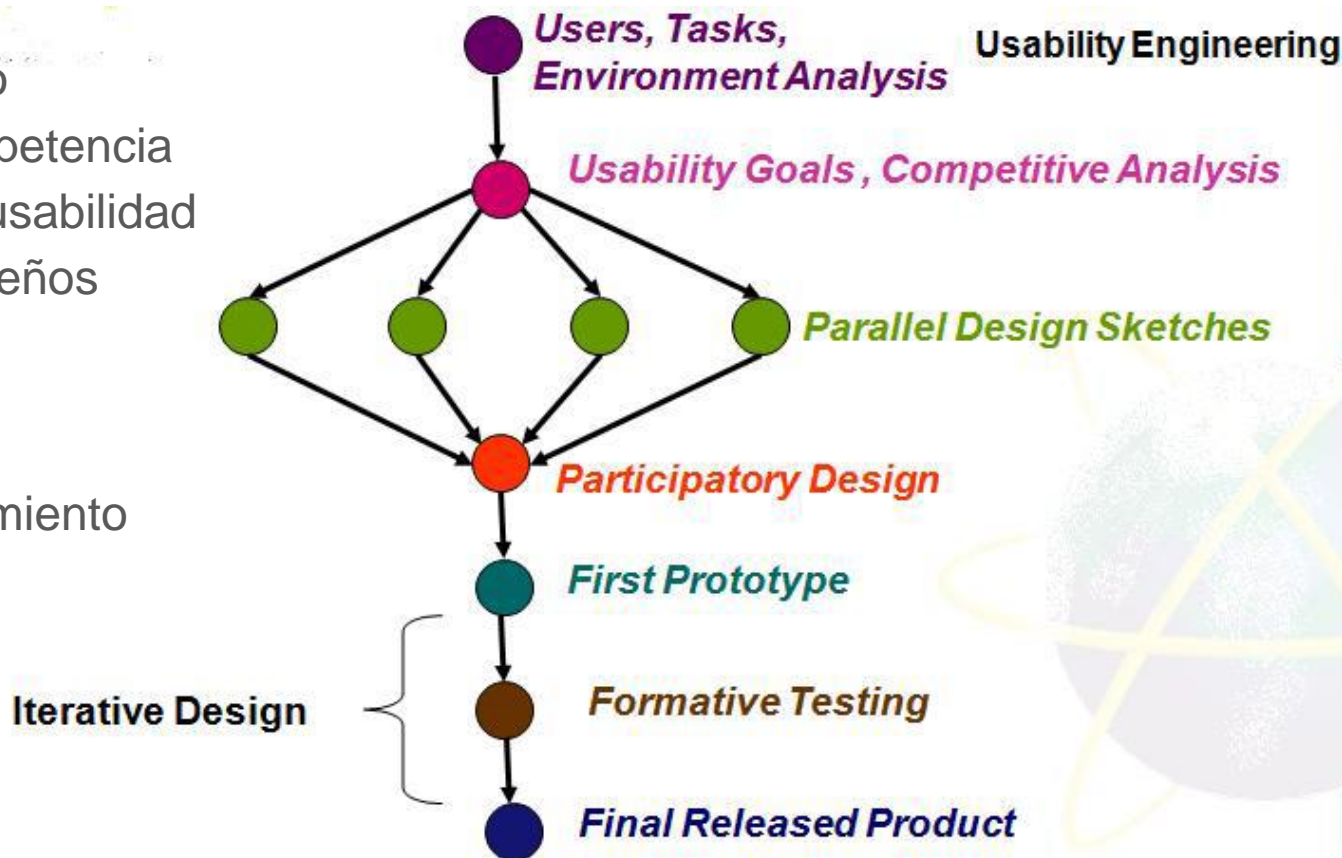
A grandes rasgos, la ingeniería de la usabilidad se caracteriza por las siguientes afirmaciones (según Nielsen):

- Tu mejor suposición no es suficiente
- El usuario siempre tiene razón
- El usuario no siempre tiene razón
- Los usuarios no son diseñadores
- Los diseñadores no son usuarios
- Los vicepresidentes no son usuarios
- Menos es más
- Los detalles importan
- La ayuda no es tan importante
- La ingeniería de la usabilidad es un proceso

4.2 Ingeniería de la usabilidad

La ingeniería de usabilidad define un proceso de diseño iterativo y centrado en el usuario. El ciclo de vida de la ingeniería de la usabilidad, según Nielsen, se divide en las siguientes etapas:

- Conocer al usuario
- Análisis de la competencia
- Fijar objetivos de usabilidad
- Realización de diseños
- Diseño iterativo
 - Prototipado
 - Evaluación
- Estudios de seguimiento



4.2.1 Conocer al usuario

Consiste en estudiar y analizar a los usuarios finales para conocer a quién va dirigido el sistema. Es importante que no sea un proceso “boca a boca” con intermediarios sino que sean los propios diseñadores quienes conozcan de primera mano al usuario.

El diseño centrado en el usuario parte de las siguientes tres preguntas:

- ¿Quién va a usar el sistema interactivo?
- ¿Con qué objetivo va a ser usado?
- ¿Cuándo y en qué contexto se va a usar?

4.2.1 Conocer al usuario

Conocer al usuario implica identificar los siguientes elementos:

- En qué somos buenos/malos los humanos (que serán los usuarios de nuestro sistema)
- Cómo podemos ayudarles a hacer una tarea en la forma en la que actualmente la realizan.
- Qué puede producirles experiencias de calidad.

4.2.1 Conocer al usuario

Una forma de conocer al usuario consiste en entender las tareas que realiza. El análisis de tareas consiste en observar a los usuarios durante la realización de una tarea y entrevistarles para extraer información sobre:

- ¿Qué objetivos persigue?
- ¿Qué información necesita para completar su objetivo?
- ¿Cómo realiza la tarea?
- ¿Cómo se comporta ante situaciones excepcionales?
- ¿Cuáles son los resultados obtenidos?

4.2.1 Conocer al usuario

Una forma de conocer al usuario consiste en entender las tareas que realiza. El análisis de tareas consiste en observar a los usuarios durante la realización de una tarea y entrevistarles para extraer información sobre:

- ¿Qué objetivos persigue?
- ¿Qué información necesita para completar su objetivo?
- ¿Cómo realiza la tarea?
- ¿Cómo se comporta ante situaciones excepcionales?
- ¿Cuáles son los resultados obtenidos?

4.2.2 Análisis de la competencia

Un sistema ya implementado es el mejor prototipo que se puede usar desde el punto de vista de la usabilidad. El análisis de la competencia se centra en analizar cómo los usuarios perciben y usan un producto que ya existe.

Un análisis de la competencia nos ayuda a conocer qué características de un producto son útiles para los usuarios y cuáles son los problemas existentes en el producto.

Si hay varios sistemas en el mercado se pueden realizar estudios comparativos para extraer ideas para el nuevo diseño. Las revisiones de sistemas hechos por terceras partes (artículos en revistas especializadas, blogs, etc.) son también una buena fuente de información de cara al nuevo diseño. Es importante destacar que este análisis no supone robar una idea sino analizar sus puntos fuertes y sus debilidades de cara a realizar un futuro diseño.

4.2.3 Fijar objetivos de usabilidad

En esta fase se definen cuáles son los objetivos de usabilidad que queremos llegar a cumplir durante el diseño de un sistema interactivo.

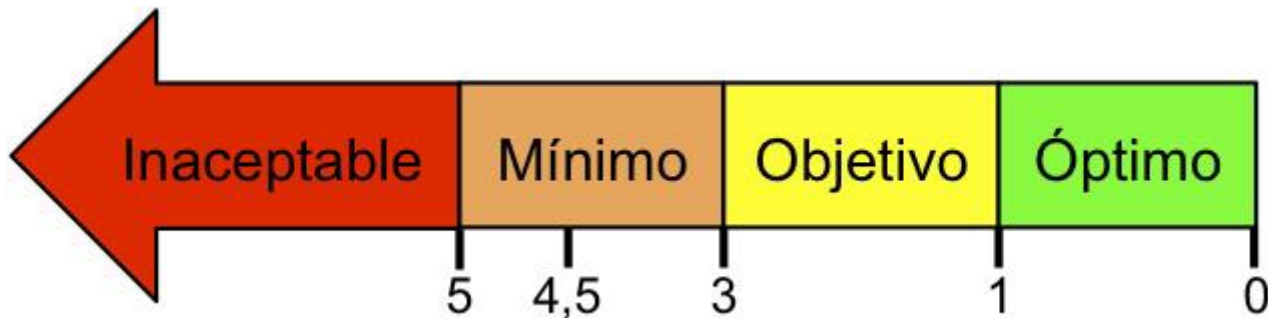
Para ello se definen cuáles son los atributos de usabilidad que queremos destacar en el sistema, su prioridad y, opcionalmente, la métrica de usabilidad utilizada para medir que se ha alcanzado dicho objetivo.

Si ya existe un sistema que realiza esta misma tarea los objetivos serán más fáciles de fijar ya que estos consistirán en, al menos, conseguir los mismos objetivos que en el otro sistema.

4.2.3 Fijar objetivos de usabilidad

Por ejemplo, podemos establecer una tasa de errores mínima como objetivo de usabilidad relacionado con los errores que el usuario puede llegar a cometer al interactuar con el sistema.

En el esquema de la figura se ha representado que un tipo de sistema tiene actualmente una tasa media de 4,5 errores a la hora. El objetivo se ha establecido entre 1-3 errores a la hora, mientras que una tasa de errores de entre 3-5 errores a la hora indica que se está al límite de no cumplir este objetivo de usabilidad.



4.2.3 Fijar objetivos de usabilidad

Generalmente en esta fase también se puede realizar un análisis del impacto financiero de la usabilidad del sistema de cara a justificar los objetivos propuestos. Suele consistir en estimaciones realizadas sobre el coste que conlleva la realización de una tarea teniendo en cuenta el número de usuarios que utilizan el sistema, el coste por usuario y el tiempo medio que pasa cada usuario interactuando con el sistema.

4.2.4 Realización de diseños

Esta etapa de diseño está relacionada con la forma en la que el usuario interactúa con el sistema. Según Nielsen, la fase de diseño está dividida en distintas etapas aunque en esta asignatura haremos una simplificación y la trataremos como una sola.

- Primera etapa de **diseños en paralelo**.
- Segunda etapa de **diseño participativo**.

Durante el diseño es necesario coordinar los distintos diseños para que guarden consistencia entre ellos y con posibles versiones anteriores del sistema. Así mismo puede ser necesario tener que cumplir ciertos estándares de diseño o que la terminología empleada en todas las interfaces sea la misma por lo que es fundamental que exista al menos una persona responsable de esta coordinación.

4.2.4 Diseño paralelo

Etapa se recomienda realizar diseños en paralelo, de modo que se exploren diferentes alternativas a la vez antes de centrarnos en una única idea de diseño.

Podemos tener 3-4 diseñadores trabajando individualmente y en paralelo durante esta fase. Esta primera etapa será corta: no más de 1-2 días de trabajo. Es importante que cada diseñador trabaje individualmente para obtener la máxima diversidad en los diseños. Una vez que se han completado los diseños se estudiarán para quedarnos con el menor número de diseños posibles. Si es necesario, se pueden combinar ideas de diseño para formar uno solo. El diseño en paralelo reduce el coste ya que se exploran varias alternativas de diseño simultáneamente aunque la mayoría de estas ideas se descarten incluso antes de que el sistema se haya implementado.

4.2.4 Diseño paralelo

Una forma alternativa de diseño en paralelo es aquél en el que se pide a cada diseñador que se centre en un aspecto concreto del sistema (por ejemplo, un diseñador se centra en la interfaz para novatos mientras que otro se centra en la interfaz para expertos).

Mientras que la ventaja principal es que en cada diseño saldrán ideas que no aparecerán en otros diseño, el principal inconveniente es que posteriormente habrá que realizar un gran esfuerzo en unificar todo a una única interfaz.

4.2.4 Diseño participativo

La etapa de diseño participativo consiste en realizar trabajo de diseño junto con usuarios reales. Estas sesiones de diseño ayudan a acercar el modelo de la interfaz al modelo mental del usuario. A los usuarios que participan en estas sesiones se les conoce como expertos de materia (subject matter experts o SME).

No se les pide que traigan sus propias ideas de diseño ni se les preguntará qué es lo que quieren (recordemos que el usuario no sabe lo que quiere) sino que se les pedirá que trabajen sobre las ideas ya desarrolladas por los diseñadores.

No se les proporcionan especificaciones de diseño sino prototipos de diseño en forma de bocetos, mockups y prototipos en papel para que lo evalúen. Es recomendable que en diseños largos no se trabaje siempre con lo mismos usuarios ya que, después de varias reuniones, estos usuarios pueden quedar sesgados hacia las ideas de los diseñadores. Nuevos usuarios que no conozcan la evolución del diseño pueden aportar ideas frescas.

4.2.4 Realización de diseños

También durante el diseño será importante definir y utilizar unas directrices o guidelines a seguir durante el desarrollo. Las directrices contienen información sobre principios de diseño conocidos y probados que hay que tener en cuenta de cara a los distintos atributos de usabilidad. Estas directrices pueden ser de carácter general (p.e. proporcionar feedback al usuario), para una categoría de interfaz específica (p.e. para la navegación hipermedia se recomienda que se informe al usuario de que se está produciendo una transición de un nodo a otro) o específicas para un producto (p.e. algunas apps para móviles como la de los periódicos –El País, Marca, El mundo.es . . . – hace un desplazamiento lateral de la página para indicar que se ha cambiado de página, en lugar de cargar directamente el contenido).

4.2.5 Prototipado

La ingeniería de la usabilidad describe un proceso iterativo de diseño y evaluación basada en prototipos. No requiere que el proceso de ingeniería haya concluido (haya un sistema implementado y probado) sino que se realiza sobre prototipos que ejemplifican la interacción con el sistema. Esto supone una agilización y una reducción de los costes en el desarrollo del software.

El proceso es iterativo e incremental en cuanto a fidelidad con el sistema final. Así, los primeros prototipos son sencillos y desarrollados de manera rápida y a un bajo coste. De esta manera también tendremos prototipos que son fácilmente modificables incluso desechables, hasta que tengamos un conocimiento más claro del diseño final del sistema interactivo que se va a desarrollar.

4.2.5 Prototipado

El bajo coste de los prototipos no solo se consigue con la reducción de las funcionalidades del sistema sino que también se pueden tener en cuentas otros aspectos:

- Reducir la fidelidad del prototipo con respecto a la interfaz final.
- Usar prototipos en papel y mockups, en lugar de prototipos en computador.
- Usar prototipos verbales.
- Usar datos falsos y contenido que no es real.
- Usar simplificaciones que no tienen en cuenta casos especiales.
- Dejar la eficiencia a un lado.
- Aceptar la existencia de errores y “cuelgues”
- Usar a personas que hagan “por detrás” el trabajo (Mago de Oz).
- Usar una plataforma distinta a la final en la que se va a ejecutar el sistema.

4.2.6 Evaluación

Los diseños realizados en las fases anteriores han de sufrir evaluaciones que nos ayuden a cumplir los objetivos de usabilidad fijados con anterioridad.

Las evaluaciones se realizan en cada iteración y sobre cada uno de los diseños. Las evaluaciones pueden ser distintas en cada iteración, teniendo en cuenta distintos aspectos de la interfaz. Los dos tipos de evaluaciones fundamentales que se realizarán:

- **Evaluaciones heurísticas.** Evalúan la interfaz en términos de usabilidad. En este caso no serán los usuarios finales los que la evalúen sino que serán los expertos en usabilidad.
- **Evaluaciones con usuarios.** La forma más directa de entender cómo el usuario final va a interactuar con la interfaz es permitiendo a los usuarios trabajar directamente con el programa y observar hasta qué punto son capaces de realizar determinados conjuntos de tareas.

4.2.6 Evaluación

Las evaluaciones óptimas (y las más costosas) se realizan con implementaciones completas del sistema y con usuarios reales.

Para optimizar el uso de evaluaciones se propone usar los prototipos descritos anteriormente. Estos, además, permiten realizar cambios ante problemas encontrados durante estas evaluaciones.

Generalmente, el resultado de estas evaluaciones será una lista de problemas de usabilidad junto con una lista de características necesarias para dar soporte a las estrategias de interacción del usuario. Como no todos los problemas se pueden resolver a la vez se establecerán puntuaciones de severidad (o severity ratings) para cada uno de los problemas de usabilidad detectados.

4.2.6 Evaluación

Estas puntuaciones se pueden establecer de acuerdo a dos escalas:

- Una escala numérica 0-4 que va desde la inexistencia del problema (0) a la usabilidad catastrófica (4), que indica que es imprescindible resolver ese problema.
- Una combinación de dos dimensiones:
 - Cuántos usuarios han tenido/van a tener el problema de usabilidad detectado.
 - Cuál es el impacto del problema en el usuario

		Proportion of users experiencing the problem	
		<i>Few</i>	<i>Many</i>
Impact of problem on the users who experience it	<i>Small</i>	Low severity	Medium severity
	<i>Large</i>	Medium severity	High severity

4.2.6 Evaluación

Los resultados de las evaluaciones son analizados antes de iniciar una nueva iteración del proceso iterativo de diseño: las deficiencias encontradas conducen a la creación/modificación del diseño anterior; este diseño será trasladado a un prototipo de menor o mayor fidelidad (dependiendo del estado en el que se encuentre el diseño); finalmente, se volverá a realizar una evaluación para garantizar la usabilidad del sistema.

Como norma general, en cada iteración iremos resolviendo los problemas de usabilidad del sistema. Sin embargo, en ocasiones, un cambio de diseño no resuelve un problema e, incluso, introduce nuevos problemas. El proceso iterativo garantiza la detección de estas situaciones y su fácil resolución. También en ocasiones un cambio de diseño resuelve un problema a algunos usuarios pero perjudica a otros (novatos vs. intermedios). En estos casos hay que analizar la compensación para decidir la solución con la que nos quedamos: a cuántos usuarios le surgió el problema con respecto a cuántos usuarios sufren problemas con la nueva solución, cuál es el coste en incluir la nueva solución. . .

4.2.7 Estudios de seguimiento

Una vez que el sistema ha sido diseñado y puesto en el mercado es hora de hacer estudios de seguimiento para ver cómo se comporta el sistema con los usuarios reales. Estos estudios sirven para preparar nuevas versiones o futuros productos. Además, estos sistemas en producción son prototipos perfectos para ser usados para conocer al usuario y la competencia, las primeras etapas que hemos visto dentro del ciclo de vida de la ingeniería de la usabilidad.

La información de seguimiento se puede sacar de diversas fuentes:

- Como parte de los estudios de marketing.
- Cuestionarios
- Entrevistas con usuarios
- Estudios de observación
- Información extraída del sistema (sistemas de logging o bitácora)
- Quejas de usuarios y sistemas de reporte de errores.
- Datos económicos sobre el impacto del uso del sistema en la calidad y el coste de los usuarios.