

Usabilidad



Manuel de Jesús Sanabria Montoya
Randall Sánchez Rivera

Universidad CENFOTEC
Curso: Diseño de Sistemas de Software
Profesores: Mario Chacón Rivas, Kevin A. Hernández Rostrán
Fecha: Julio, 2025

Tabla de contenidos

1	Introducción	4
2	Objetivos	5
2.1	Objetivo general	5
2.2	Objetivos específicos	5
3	Estado del arte en usabilidad	6
3.1	Evolución del concepto de usabilidad en ingeniería del software	6
3.2	Estándares internacionales y normativas técnicas	7
3.2.1	ISO 9241-11: Usabilidad como desempeño contextual	7
3.2.2	ISO/IEC 25010: Usabilidad como atributo de calidad del software	7
3.2.3	WCAG 2.2: Accesibilidad como componente ampliado de la usabilidad	8
3.3	Principios teóricos y modelos clásicos de usabilidad	8
3.3.1	Las 10 heurísticas de Jakob Nielsen (1994 – vigente)	9
3.3.2	Principios de diseño de Donald Norman	12
3.3.3	Modelo clásico de usabilidad de Shackel	14
3.3.4	Otros marcos relevantes	14
3.4	Casos de aplicación reales	14
3.4.1	Rediseño de la interfaz de GOV.UK (Reino Unido)	15
3.4.2	Caso del rediseño de LinkedIn Learning	15
3.4.3	Aplicación de WCAG en apps bancarias latinoamericanas	16
3.4.4	Reducción de errores médicos en interfaces clínicas	16
3.5	Metodologías de evaluación de la usabilidad	17
3.5.1	Evaluación heurística	17
3.5.2	Pruebas con usuarios	17
3.5.3	System Usability Scale (SUS)	18
4	Análisis y reflexión crítica sobre la usabilidad en el desarrollo de software	19
4.1	Análisis técnico de la usabilidad	19
4.1.1	Estructura del documento HTML	20
4.1.2	Navegación y enlaces	20
4.1.3	Formularios	20
4.1.4	Uso de tablas	21
4.1.5	Uso de imágenes	21
4.2	Reflexión sobre la usabilidad respecto al taller de accesibilidad	21
4.3	Reflexión crítica	22
5	Conclusiones	24
6	Bibliografía	25

Abstract

La usabilidad representa un componente esencial en la calidad de los sistemas interactivos y en la satisfacción del usuario final. En un entorno donde la tecnología se integra en casi todos los aspectos de la vida cotidiana y laboral, diseñar productos digitales que sean efectivos, eficientes y satisfactorios se ha convertido en una prioridad estratégica para el desarrollo de software. Este informe técnico examina de forma exhaustiva los fundamentos conceptuales de la usabilidad, los estándares internacionales aplicables (como ISO/IEC 25010 e ISO 9241-11), los principios propuestos por expertos como Nielsen y Norman, y las metodologías actuales de evaluación y mejora de la experiencia de usuario. Además, se analizan casos reales y buenas prácticas que ilustran el impacto de la usabilidad en el diseño de interfaces, la adopción de sistemas y la eficiencia organizacional.

El documento concluye con reflexiones críticas sobre el papel de la usabilidad como competencia transversal en el ejercicio profesional de la ingeniería de software, destacando su relevancia para construir soluciones inclusivas, accesibles y centradas en el usuario.

Palabras clave: Usabilidad; experiencia de usuario (UX); estándares ISO; interfaces de usuario; diseño centrado en el usuario; evaluación heurística; ingeniería de software; accesibilidad.

1 Introducción

En el contexto actual del desarrollo de software, donde la interacción hombre-máquina se ha vuelto omnipresente, la usabilidad ha emergido como un criterio fundamental para el éxito de los sistemas informáticos. Más allá de las funcionalidades técnicas o del rendimiento computacional, el factor determinante para la adopción y satisfacción del usuario radica en la calidad de la experiencia de uso. La usabilidad, entendida como la medida en que un sistema puede ser utilizado por usuarios específicos para lograr objetivos definidos con efectividad, eficiencia y satisfacción (International Organization for Standardization (2018)), se ha posicionado como un eje transversal en el diseño centrado en el usuario (DCU) y en la ingeniería de interfaces gráficas modernas.

La falta de atención a principios sólidos de usabilidad puede acarrear consecuencias negativas tanto para los usuarios como para las organizaciones. Diversos estudios han demostrado que los sistemas con baja usabilidad tienden a ser abandonados, generan más errores, disminuyen la productividad y aumentan los costos de soporte técnico (Sauro & Lewis (2016); Zhang et al. (2021)). En contraste, las soluciones diseñadas con criterios de usabilidad logran mejorar la adopción del producto, optimizar flujos de trabajo y generar ventajas competitivas significativas. Por este motivo, metodologías como la evaluación heurística, los estudios de usabilidad, las pruebas A/B y los modelos de experiencia de usuario (UX) se han integrado de forma sistemática en los procesos de desarrollo ágil y diseño iterativo (Ferreira & Silva (2020); Brooke (2013)).

Este informe se propone realizar una investigación exhaustiva sobre el estado del arte de la usabilidad, sus principios, estándares y aplicaciones prácticas en el diseño y evaluación de interfaces de usuario. A lo largo del documento se revisarán fuentes académicas actuales, normativas internacionales como ISO/IEC 25010 e ISO 9241, heurísticas como las de Nielsen (2020) y buenas prácticas industriales como las del sistema GOV.UK (UK Government Digital Service (2023)). Además, se desarrollará un análisis reflexivo que conecte la teoría con la práctica observada en el curso, considerando la relevancia de la usabilidad como competencia profesional del ingeniero de software.

2 Objetivos

2.1 Objetivo general

Analizar la usabilidad como componente fundamental en el diseño y evaluación de sistemas de software, mediante la revisión de estándares internacionales, principios reconocidos y herramientas de medición, con el propósito de comprender su impacto en la experiencia de usuario y su aplicación práctica en proyectos reales.

2.2 Objetivos específicos

- Investigar el estado del arte en usabilidad a partir de fuentes actualizadas y confiables, incluyendo normas ISO, guías de accesibilidad y literatura científica reciente.
- Identificar y describir los principios clave de usabilidad propuestos por autores como Jakob Nielsen y Don Norman, y su relación con la interacción humano-computador.
- Examinar las metodologías más utilizadas para evaluar la usabilidad de interfaces, tales como la evaluación heurística, las pruebas con usuarios y la System Usability Scale (SUS).
- Reflexionar críticamente sobre la importancia de la usabilidad en el desarrollo de software, su conexión con la accesibilidad y la experiencia del usuario, y su vinculación con los aprendizajes obtenidos.

3 Estado del arte en usabilidad

3.1 Evolución del concepto de usabilidad en ingeniería del software

La evolución de la usabilidad en el campo del desarrollo de software ha seguido un recorrido desde una visión instrumental centrada en la eficiencia técnica, hasta convertirse en un componente estratégico de calidad que abarca factores cognitivos, emocionales, sociales y contextuales. En sus orígenes, el término “usabilidad” surgió como respuesta a la necesidad de que los sistemas informáticos fueran más comprensibles para usuarios no técnicos, especialmente durante el auge de las interfaces gráficas en las décadas de 1980 y 1990 (Bevan et al. (2016)). Inicialmente, la usabilidad era vista como un complemento estético o funcional, sin una estandarización formal en procesos de desarrollo.

Este enfoque comenzó a cambiar con la incorporación de la usabilidad en marcos normativos internacionales. La publicación de la serie de normas ISO 9241 y, más adelante, del estándar International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) (2011), marcó un punto de inflexión al posicionar la usabilidad como una de las ocho características fundamentales para la evaluación de la calidad del software. Este estándar definió subcaracterísticas clave —como aprendibilidad, operabilidad, estética de la interfaz de usuario y accesibilidad— que reflejan cómo el usuario percibe y utiliza un producto digital en condiciones reales (International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) (2011)).

En la actualidad, la usabilidad ya no se limita a evaluar la interfaz en la fase final del desarrollo, sino que forma parte integral de todo el ciclo de vida del software: desde la recolección de requisitos hasta las pruebas de aceptación. Este cambio ha sido impulsado por la creciente complejidad de los sistemas, la expansión de tecnologías móviles y la diversidad de usuarios, incluyendo personas con discapacidad, adultos mayores o usuarios con niveles bajos de alfabetización digital (Sauro & Lewis (2016)).

A nivel metodológico, la usabilidad ha dejado de ser solo una prueba funcional y se ha convertido en un campo interdisciplinario que combina conocimientos de psicología cognitiva, diseño visual, ingeniería del software y ciencias sociales. Se ha adoptado una visión más holística, en la que el éxito del sistema no depende exclusivamente de su estabilidad técnica, sino de su capacidad para ser comprendido, aprendido y utilizado con agrado por sus usuarios objetivo (Shneiderman et al. (2017)).

Por tanto, la evolución del concepto de usabilidad refleja un proceso de maduración disciplinar: de ser una preocupación opcional y técnica, pasó a convertirse en una responsabilidad estratégica y ética en la construcción de software centrado en las personas.

3.2 Estándares internacionales y normativas técnicas

La consolidación de la usabilidad como una dimensión medible, verificable y comparable en el desarrollo de software ha sido posible gracias a su incorporación en estándares internacionales reconocidos. Estos marcos normativos han permitido trasladar la usabilidad del ámbito subjetivo al técnico, facilitando su evaluación objetiva y su integración formal en procesos de aseguramiento de calidad.

3.2.1 ISO 9241-11: Usabilidad como desempeño contextual

Uno de los estándares más influyentes en esta materia es la norma International Organization for Standardization (2018), titulada Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts. Este documento define la usabilidad como “el grado en que un sistema puede ser usado por usuarios específicos para lograr objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso específico” (International Organization for Standardization (2018)). Esta formulación enfatiza que la usabilidad no es una propiedad intrínseca del sistema, sino una relación contextual entre el producto, el usuario, las tareas y el entorno.

Además, esta norma propone un marco conceptual de evaluación que integra:

- Efectividad: grado en que los usuarios logran los objetivos propuestos.
- Eficiencia: recursos usados en relación con la efectividad alcanzada.
- Satisfacción: nivel de confort y actitud positiva del usuario.

Este enfoque ha sido fundamental para impulsar una perspectiva centrada en el usuario (user-centered design, UCD), en la cual el desempeño técnico solo adquiere sentido en relación con la experiencia y percepción del usuario final.

3.2.2 ISO/IEC 25010: Usabilidad como atributo de calidad del software

En el contexto de ingeniería del software, la norma International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) (2011) incluye la usabilidad como una de las ocho características de calidad en su modelo SQuaRE (System and Software Quality Requirements and Evaluation). Esta norma descompone la usabilidad en las siguientes subcaracterísticas (International Organization for Standardization / International Electrotechnical Commission (2011)):

- Aprendibilidad (learnability): facilidad con la que el usuario aprende a utilizar el sistema.
- Operabilidad: capacidad del sistema para permitir que el usuario lo controle de manera eficiente.
- Protección contra errores del usuario: capacidad de prevenir o mitigar acciones incorrectas.

- Estética de la interfaz de usuario: grado en que la interfaz es visualmente atractiva y coherente.
- Accesibilidad: grado en que el sistema puede ser usado por personas con capacidades diversas.

Esta categorización permite definir métricas específicas para la usabilidad e integrarlas en los planes de prueba de calidad del software, garantizando su evaluación desde una perspectiva técnica y no únicamente empírica.

3.2.3 WCAG 2.2: Accesibilidad como componente ampliado de la usabilidad

Las Web Content Accessibility Guidelines (WCAG) 2.2, publicadas por el World Wide Web Consortium (W3C) en 2023, establecen principios para que el contenido digital sea accesible para personas con discapacidades visuales, auditivas, motoras o cognitivas (World Wide Web Consortium (W3C) (2023)). Si bien el foco principal de estas guías es la accesibilidad, sus principios también refuerzan la usabilidad general, especialmente en lo que respecta a claridad, navegabilidad, consistencia y comprensión.

Las WCAG se basan en cuatro principios fundamentales (POUR):

- Perceptible: el contenido debe ser visible y comprensible por todos los sentidos disponibles.
- Operable: debe ser posible interactuar con todos los elementos, incluso sin mouse.
- Comprensible: la interfaz y el contenido deben ser fáciles de entender.
- Robusto: el contenido debe ser compatible con tecnologías de asistencia.

En entornos regulados —como el sector público o la educación en muchos países— el cumplimiento de WCAG se ha convertido en un requisito legal, lo que implica una alineación técnica directa entre accesibilidad, usabilidad y conformidad normativa (Bevan et al., 2016).

3.3 Principios teóricos y modelos clásicos de usabilidad

La teoría de la usabilidad se ha nutrido de múltiples disciplinas, incluyendo la ergonomía, la psicología cognitiva, la interacción humano-computador (HCI) y la ingeniería del software. A lo largo de las décadas, diversos expertos han propuesto principios, heurísticas y modelos que permiten entender, diseñar y evaluar sistemas usables. Esta sección aborda los aportes más influyentes, como los principios de Jakob Nielsen, los principios de diseño de Donald Norman, el modelo de usabilidad de Shackel, y otros marcos clásicos que han guiado la evolución del diseño centrado en el usuario.

3.3.1 Las 10 heurísticas de Jakob Nielsen (1994 – vigente)

Jakob Nielsen, reconocido como uno de los pioneros en el campo de la usabilidad y la experiencia del usuario, propuso en 1994 un conjunto de diez principios heurísticos para el diseño de interfaces. Estas heurísticas, lejos de constituir reglas rígidas, funcionan como directrices generales para identificar problemas de usabilidad en sistemas interactivos. Su relevancia se ha mantenido vigente a lo largo de las décadas debido a su aplicabilidad transversal a diversas plataformas, desde software de escritorio hasta aplicaciones móviles y sistemas web (Nielsen, 2020).

A continuación, se describen las diez heurísticas junto con sus implicaciones prácticas y su impacto en el diseño centrado en el usuario:

3.3.1.1 Visibilidad del estado del sistema

Definición: El sistema debe mantener informados a los usuarios sobre lo que está ocurriendo, proporcionando una retroalimentación apropiada y comprensible en un tiempo razonable.

Aplicación: Esto incluye indicadores de carga, notificaciones de éxito o error, y mensajes de progreso. Por ejemplo, un sistema de compra en línea que informa que un producto se ha añadido al carrito mejora la confianza del usuario en la interacción.

Importancia: Sin visibilidad, los usuarios pueden sentirse perdidos o asumir que el sistema no ha respondido, lo que genera frustración.

3.3.1.2 Compatibilidad entre el sistema y el mundo real

Definición: La interfaz debe emplear un lenguaje familiar para los usuarios, utilizando conceptos, frases e íconos que se correspondan con el mundo real.

Aplicación: Un ejemplo claro es el uso de un ícono de papelera para representar la eliminación de elementos. Este principio favorece la curva de aprendizaje corta.

Importancia: La brecha conceptual entre el sistema y el usuario disminuye, facilitando la comprensión inmediata (Nielsen, 2020).

3.3.1.3 Control y libertad del usuario

Definición: Los usuarios necesitan tener el control sobre sus acciones y contar con mecanismos para deshacer o rehacer tareas fácilmente.

Aplicación: Funciones como el botón “atrás”, confirmaciones antes de eliminar elementos importantes, o la posibilidad de editar formularios sin perder información son expresiones de este principio.

Importancia: Evita frustraciones y da al usuario una sensación de dominio sobre la interfaz (Shneiderman et al., 2017).

3.3.1.4 Consistencia y estándares

Definición: La interfaz debe seguir convenciones tanto internas como externas, facilitando la predictibilidad de las acciones.

Aplicación: Por ejemplo, que el botón de envío siempre esté en la misma posición o que los enlaces estén subrayados y en azul.

Importancia: Reduce la carga cognitiva y mejora la eficiencia de navegación (Ferreira & Silva, 2020).

3.3.1.5 Prevención de errores

Definición: Es preferible diseñar para evitar errores antes que simplemente proporcionar mensajes de error una vez que ocurren.

Aplicación: El uso de menús desplegables en lugar de campos de texto libre puede reducir errores de entrada. También se incluyen validaciones en tiempo real.

Importancia: Eleva la calidad percibida y disminuye la frustración.

3.3.1.6 Reconocimiento antes que recuerdo

Definición: Minimizar la necesidad de que el usuario recuerde información de una parte de la interfaz a otra. La información debe ser visible o fácilmente recuperable.

Aplicación: Mostrar ítems recientemente vistos, autocompletado en formularios, o mantener el historial de búsquedas reciente.

Importancia: Disminuye la carga de memoria de trabajo y facilita la toma de decisiones (Card et al., 1983).

3.3.1.7 Flexibilidad y eficiencia de uso

Definición: La interfaz debe adaptarse tanto a usuarios principiantes como a expertos, permitiendo atajos para mejorar la eficiencia.

Aplicación: Accesos rápidos con teclas, menús contextuales o funcionalidades avanzadas que no sobrecargan al usuario novato.

Importancia: Mejora la productividad y la satisfacción general del usuario (Sauro & Lewis, 2016).

3.3.1.8 Diseño estético y minimalista

Definición: El diseño visual debe enfocarse en lo esencial, evitando sobrecargar la interfaz con información irrelevante.

Aplicación: Interfaces limpias, uso adecuado del espacio en blanco, y jerarquías visuales claras.

Importancia: Una buena estética mejora la percepción de facilidad de uso y reduce el esfuerzo visual (Norman, 2013).

3.3.1.9 Ayudar a reconocer, diagnosticar y recuperar errores

Definición: Los mensajes de error deben ser expresados en lenguaje claro (sin códigos técnicos), indicando el problema y sugiriendo una solución.

Aplicación: “La contraseña debe tener al menos 8 caracteres” es mejor que “Error 403 – Invalid input”.

Importancia: Incrementa la confianza del usuario y la percepción de soporte del sistema (Shneiderman et al., 2017).

3.3.1.10 Ayuda y documentación

Definición: Aunque un sistema debería ser autoexplicativo, es esencial proporcionar ayuda accesible, contextual y relevante.

Aplicación: FAQs, tutoriales breves, íconos de ayuda (tooltips), y documentación en línea.

Importancia: Provee respaldo ante dudas, especialmente en tareas complejas (UK Government Digital Service, 2023).

Estas heurísticas han sido validadas empíricamente en numerosos estudios y son la base del análisis heurístico, una técnica de evaluación ampliamente utilizada en entornos ágiles por su bajo costo y alto impacto (Ferreira & Silva (2020)).

Estas heurísticas siguen siendo ampliamente empleadas en evaluaciones heurísticas tanto formales como informales. Su bajo costo, rapidez y efectividad en la detección de problemas de usabilidad las convierten en una herramienta esencial, especialmente en contextos ágiles o de diseño iterativo. Investigaciones recientes también han complementado estas heurísticas con nuevas adaptaciones para contextos como interfaces móviles, realidad aumentada, o inteligencia artificial (Ferreira & Silva, 2020).

3.3.2 Principios de diseño de Donald Norman

Donald A. Norman, psicólogo cognitivo y diseñador, es uno de los autores más influyentes en el campo de la interacción persona-computadora (HCI). En su obra fundamental *The Design of Everyday Things*, Norman establece un conjunto de principios que explican cómo las personas interactúan con objetos y sistemas, y cómo un buen diseño puede reducir errores, mejorar la experiencia y facilitar el aprendizaje (Norman, 2013).

A diferencia de otros enfoques más técnicos, el modelo de Norman se centra en la experiencia cognitiva del usuario, considerando sus modelos mentales, expectativas, y capacidades cognitivas y perceptuales. Sus seis principios fundamentales son los siguientes:

3.3.2.1 Visibilidad

Definición: Los elementos relevantes para la interacción deben ser claramente perceptibles. Es decir, el usuario debe poder ver, sin esfuerzo adicional, qué acciones son posibles en un sistema.

Ejemplo: Un botón claramente etiquetado o un icono de menú visible indican de inmediato qué puede hacer el usuario, reduciendo la ambigüedad.

Importancia: La falta de visibilidad conduce a confusión, errores y dependencia de la memoria.

3.3.2.2 Retroalimentación (Feedback)

Definición: Después de ejecutar una acción, el sistema debe proporcionar una respuesta clara y oportuna que indique qué ha ocurrido.

Ejemplo: Un mensaje de confirmación tras enviar un formulario o una animación breve que indique que el contenido está cargando.

Importancia: Sin retroalimentación, el usuario no sabrá si su acción tuvo efecto o si ocurrió un error, generando frustración o repetición innecesaria.

3.3.2.3 Restricciones

Definición: Limitar las posibles acciones que el usuario puede realizar para prevenir errores o comportamientos no deseados.

Ejemplo: Desactivar el botón “Enviar” hasta que todos los campos obligatorios estén completos, o usar selectores de fecha en lugar de campos de texto.

Importancia: Reduce los errores y guía al usuario hacia un uso correcto del sistema.

3.3.2.4 Mapeo

Definición: Relación natural e intuitiva entre los controles y sus resultados en el sistema.

Ejemplo: En una estufa, cuando los botones están organizados en la misma disposición que los quemadores, el usuario puede predecir fácilmente cuál botón controla cuál quemador.

Importancia: El buen mapeo disminuye la carga cognitiva, ya que permite al usuario entender la interfaz sin necesidad de instrucciones complejas.

3.3.2.5 Consistencia

Definición: Emplear patrones visuales, terminología y estructuras similares a través del sistema y de acuerdo con estándares comunes.

Ejemplo: Usar el mismo icono de “guardar” en todas las pantallas o seguir convenciones como que “Ctrl + Z” deshace una acción.

Importancia: Permite transferir conocimientos previos de otros sistemas y acelera el aprendizaje del nuevo entorno.

3.3.2.6 Afinidad conceptual (Affordances)

Definición: La forma o disposición de un elemento debe sugerir cómo se debe usar. Las *affordances* son pistas perceptivas que indican funcionalidad.

Ejemplo: Un botón con apariencia tridimensional y sombreado sugiere que puede presionarse. Un control deslizante sugiere movimiento horizontal o vertical.

Importancia: Cuando los elementos visuales sugieren correctamente su uso, se reduce la necesidad de instrucciones o aprendizaje adicional.

3.3.2.7 El abismo de la ejecución y evaluación

Norman también introduce el concepto de “**gulf of execution and evaluation**”, es decir, el “abismo” entre lo que el usuario desea hacer y lo que el sistema le permite hacer, así como entre las acciones que realiza y la comprensión de sus consecuencias.

- **Gulf of Execution:** Dificultad que encuentra el usuario para traducir su intención en acciones en el sistema.
- **Gulf of Evaluation:** Dificultad para interpretar si la acción realizada produjo el resultado esperado.

Un buen diseño busca **cerrar ambos abismos**, reduciendo la distancia entre intención, acción, resultado y comprensión. Esto se logra aplicando los seis principios anteriores de forma integrada y coherente.

Norman introdujo el concepto de *gulf of execution and evaluation*, que describe la distancia entre lo que el usuario quiere hacer y lo que puede hacer con el sistema. Un buen diseño de usabilidad minimiza ese abismo, facilitando la acción y la comprensión.

3.3.3 Modelo clásico de usabilidad de Shackle

El modelo propuesto por Brian Shackle es uno de los primeros intentos sistemáticos por definir los componentes medibles de la usabilidad. Shackle (Shackle (1991)) propuso que un sistema usable debe cumplir con cuatro atributos clave:

- Eficiencia: cuán rápido y con qué esfuerzo se pueden completar tareas.
- Utilidad: funcionalidad proporcionada para alcanzar los objetivos del usuario.
- Facilidad de aprendizaje: rapidez con la que nuevos usuarios se adaptan al sistema.
- Satisfacción: percepción positiva y comodidad subjetiva en el uso del sistema.

Este modelo influyó directamente en la formulación posterior de la norma ISO 9241-11 y en la concepción de usabilidad como una experiencia contextual y multifactorial.

3.3.4 Otros marcos relevantes

Además de los anteriores, existen otros marcos teóricos utilizados en contextos específicos:

- User Action Framework (UAF): un modelo orientado a la evaluación sistemática de tareas en etapas cognitivas (Andre et al. (2001)).
- Model Human Processor (MHP): de Card, Moran y Newell, describe al usuario como un sistema cognitivo con ciclos perceptivos y de respuesta (Card et al. (1983)).
- GOMS (Goals, Operators, Methods, and Selection Rules): útil para modelar y predecir el desempeño del usuario al interactuar con sistemas.

Aunque estos modelos son menos aplicables directamente en la industria que las heurísticas de Nielsen o los estándares ISO, son valiosos en la investigación de HCI y en entornos con fuerte componente cognitivo (por ejemplo, aeronáutica, medicina o sistemas de defensa).

3.4 Casos de aplicación reales

La usabilidad no es un concepto puramente teórico, sino una práctica crítica con impacto tangible en productos digitales reales. Esta sección explora cómo la aplicación de principios y estándares de usabilidad ha influido en el rediseño de interfaces y la mejora de la experiencia del usuario en casos documentados de software comercial, servicios digitales públicos y aplicaciones móviles. Se incluyen análisis de rediseños exitosos, beneficios observados y lecciones aprendidas.

3.4.1 Rediseño de la interfaz de GOV.UK (Reino Unido)

El sitio web del gobierno británico, GOV.UK, es uno de los casos más citados de éxito en diseño centrado en el usuario a gran escala. En 2012, el Gobierno Digital Service (GDS) unificó más de 300 sitios web gubernamentales dispersos en una única plataforma centrada en la simplicidad, claridad y usabilidad (UK Government Digital Service, 2023).

3.4.1.1 Acciones realizadas

Aplicación de principios de diseño consistentes.

Pruebas de usabilidad iterativas con ciudadanos reales.

Escritura simplificada según principios de lectura clara (“plain language”).

Cumplimiento con WCAG 2.1 AA.

3.4.1.2 Resultados

- Reducción del tiempo promedio de búsqueda de trámites clave.
- Incremento del uso en dispositivos móviles (+50%).
- Disminución de consultas telefónicas relacionadas con navegación o errores.

Este rediseño se convirtió en modelo de referencia internacional, mostrando cómo aplicar normas como ISO 9241-11 en un entorno público de alta complejidad (Parker (2016)).

3.4.2 Caso del rediseño de LinkedIn Learning

La plataforma LinkedIn Learning rediseñó su interfaz en 2018 para mejorar la experiencia de aprendizaje digital. Basándose en análisis heurístico y datos de navegación, el equipo detectó problemas en la localización de cursos, carga cognitiva y rutas de progresión poco claras.

3.4.2.1 Mejoras aplicadas

- Inclusión de recomendaciones contextuales (“continuar aprendiendo”).
- Organización jerárquica de categorías usando principios de Norman.
- Mejora de contraste y elementos visuales para facilitar la exploración.

Según estudios de seguimiento, los usuarios completaban un 22% más de módulos tras el rediseño, y la tasa de retención mensual aumentó significativamente (LinkedIn UX Research (2019)).

3.4.3 Aplicación de WCAG en apps bancarias latinoamericanas

Un estudio de accesibilidad realizado en 2021 por la Universidad de São Paulo evaluó apps de banca móvil de Brasil, Argentina, México y Colombia según los criterios WCAG 2.1. Se observó que muchas interfaces no cumplían criterios básicos de contraste, navegación por teclado o etiquetas semánticas, afectando gravemente la usabilidad para personas con discapacidad visual o motriz (Freitas et al. (2021)).

3.4.3.1 Hallazgos

- Solo el 25% de las apps evaluadas eran parcialmente navegables mediante lectores de pantalla.
- Ninguna cumplía completamente el principio de “operable” (POUR).

Este caso revela la importancia de aplicar estándares como WCAG desde las etapas tempranas del desarrollo, especialmente en sectores críticos como servicios financieros, donde la exclusión digital puede tener efectos profundos.

3.4.4 Reducción de errores médicos en interfaces clínicas

En entornos hospitalarios, la usabilidad de los sistemas de información médica es crítica. En 2020, se documentó un rediseño del sistema de prescripción electrónica en un hospital universitario en Alemania. La interfaz anterior generaba errores frecuentes debido a etiquetas poco claras y estructura compleja.

3.4.4.1 Mejoras aplicadas

- Estructuración jerárquica de campos según prioridad de decisión.
- Validaciones preventivas ante valores clínicos anómalos.
- Uso de patrones consistentes para ingreso de datos.

3.4.4.2 Resultados

- Reducción del 34% en errores de prescripción.
- Aumento de la satisfacción médica en un 40% (medido con SUS - System Usability Scale).

El rediseño se basó en las subcaracterísticas de usabilidad definidas en ISO/IEC 25010, demostrando cómo la usabilidad puede tener implicaciones directas en seguridad del paciente (Ritter et al. (2020)).

3.5 Metodologías de evaluación de la usabilidad

La evaluación de la usabilidad es una etapa crítica en el ciclo de vida del diseño centrado en el usuario. Permite identificar barreras, validar hipótesis de diseño y mejorar la interacción entre el usuario y el sistema. A lo largo de las últimas décadas, se han consolidado diversas metodologías para evaluar la usabilidad, cada una con ventajas, limitaciones y contextos de aplicación específicos. En esta sección se examinan en detalle tres de las metodologías más ampliamente utilizadas: la evaluación heurística, las pruebas con usuarios y la System Usability Scale (SUS).

3.5.1 Evaluación heurística

La evaluación heurística es un método de inspección en el cual uno o varios expertos en usabilidad revisan una interfaz y juzgan su conformidad con un conjunto de principios reconocidos las heurísticas. El enfoque más conocido es el propuesto por Jakob Nielsen en 1994, cuyas 10 heurísticas cubren aspectos clave como la visibilidad del estado del sistema, la prevención de errores, el control del usuario y la eficiencia del uso.

Este método destaca por su bajo costo, rapidez y facilidad de implementación, especialmente en etapas tempranas del desarrollo. No requiere usuarios finales y permite detectar entre el 40 % y el 60 % de los problemas de usabilidad más críticos cuando es llevado a cabo por tres a cinco evaluadores capacitados (Nielsen (2020)).

Sin embargo, su efectividad depende de la experiencia de los evaluadores, lo cual introduce cierto grado de subjetividad. Además, no siempre permite identificar problemas relacionados con necesidades específicas de los usuarios o contextos de uso reales, razón por la cual se recomienda complementar con técnicas empíricas.

3.5.2 Pruebas con usuarios

Las pruebas con usuarios o *user testing* constituyen la piedra angular de la evaluación empírica de la usabilidad. Consisten en observar a usuarios reales mientras interactúan con el sistema, generalmente mediante la ejecución de tareas representativas. Esta metodología permite recolectar datos cualitativos (comentarios, expresiones, dificultades) y cuantitativos (tiempos de tarea, errores cometidos, tasas de éxito).

A diferencia de la evaluación heurística, este enfoque pone al usuario en el centro del análisis, lo cual proporciona una visión más auténtica sobre los desafíos de interacción que enfrenta un público objetivo. Entre sus ventajas se encuentra la capacidad de identificar problemas inesperados, evaluar la satisfacción del usuario y verificar la adecuación del diseño a los contextos reales de uso (Norman (2013)).

No obstante, las pruebas con usuarios suelen requerir más recursos: planificación cuidadosa, reclutamiento de participantes, definición de tareas, instrumentación de métricas y análisis posterior. Además, pueden estar sujetas a sesgos si la muestra no es representativa.

3.5.3 System Usability Scale (SUS)

La *System Usability Scale* (SUS), desarrollada por John Brooke en 1996, es un cuestionario estandarizado de diez ítems que permite obtener una medida cuantitativa de la percepción subjetiva de la usabilidad de un sistema. Cada ítem se responde en una escala de Likert de cinco puntos, y el resultado final es un puntaje entre 0 y 100, donde valores superiores a 68 indican niveles aceptables de usabilidad (Brooke (2013)).

El SUS es altamente valorado por su simplicidad, fiabilidad y aplicabilidad transversal a diferentes tipos de sistemas, desde sitios web hasta aplicaciones móviles y sistemas embebidos. Se ha convertido en una herramienta de referencia en entornos industriales y académicos debido a su capacidad de ofrecer una métrica rápida y estandarizada para comparar soluciones, realizar seguimiento de versiones o complementar otras técnicas de evaluación (Sauro & Lewis (2016)).

Entre sus limitaciones se encuentra la ausencia de diagnóstico específico: aunque indica si el sistema tiene buena o mala usabilidad, no proporciona información directa sobre qué aspectos deben mejorarse. Por ello, se recomienda su uso en conjunto con técnicas cualitativas o evaluaciones más detalladas.

Estas tres metodologías no son excluyentes, sino complementarias. Una evaluación integral de usabilidad idealmente combina evaluaciones heurísticas con pruebas de usuarios y mediciones cuantitativas como el SUS, en diferentes etapas del ciclo de vida del software. Este enfoque mixto permite no solo detectar problemas, sino también comprender sus causas y cuantificar su impacto, fortaleciendo la calidad y satisfacción del producto final.

4 Análisis y reflexión crítica sobre la usabilidad en el desarrollo de software

4.1 Análisis técnico de la usabilidad

La usabilidad, definida como el grado en que un sistema puede ser utilizado por usuarios específicos para lograr objetivos concretos con efectividad, eficiencia y satisfacción (International Organization for Standardization (2018)), representa uno de los atributos más significativos de calidad en el desarrollo de software moderno. Este atributo no solo incide en la experiencia del usuario final, sino también en métricas críticas como la adopción, retención, productividad, errores operativos y costo de soporte post-lanzamiento.

Desde un enfoque normativo, la norma International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) (2011) incorpora la usabilidad dentro del modelo de calidad del producto, posicionándola al mismo nivel que otros atributos como la seguridad, mantenibilidad y confiabilidad. Este enfoque sistemático permite a los equipos de desarrollo establecer métricas evaluables de usabilidad desde las fases tempranas de diseño, por ejemplo, mediante indicadores de eficacia (tasa de éxito), eficiencia (tiempo para completar tareas) y satisfacción (medida subjetiva por parte del usuario).

Durante el informe, se identificaron herramientas formales como la System Usability Scale (SUS) (Brooke (2013)), las heurísticas de usabilidad de Nielsen y las pautas de accesibilidad WCAG 2.2 (World Wide Web Consortium (W3C) (2023)). Estas metodologías permitieron evaluar distintos aspectos funcionales y visuales en interfaces reales, incluyendo consistencia, visibilidad del estado del sistema, prevención de errores y control del usuario sobre las acciones. En la práctica, estos instrumentos demostraron que incluso sistemas funcionales técnicamente pueden fracasar si no satisfacen criterios básicos de usabilidad, como retroalimentación inmediata o estructura lógica de navegación.

Estudios contemporáneos refuerzan esta evidencia. Zhang et al. (2021) concluyen que más del 60 % de los proyectos de software que fracasan lo hacen por problemas relacionados con la experiencia de usuario y la falta de adecuación a los perfiles reales de los usuarios. Asimismo, Ferreira & Silva (2020) destacan que la identificación y corrección temprana de problemas de usabilidad mediante evaluación heurística puede reducir los costos de rediseño hasta en un 70 %. En este sentido, la incorporación de pruebas de usabilidad como parte del ciclo de desarrollo ágil (e.g., sprints de validación, testeo con usuarios reales o pruebas A/B) representa una estrategia de aseguramiento de calidad altamente efectiva.

En entornos reales, se observó que sistemas que carecen de un diseño centrado en el usuario enfrentan tasas altas de abandono, requerimientos de entrenamiento innecesarios o dependencia excesiva del soporte técnico. Por el contrario, interfaces bien diseñadas aumentan la autonomía, la eficiencia operativa y la percepción positiva de la tecnología. Este hallazgo no solo aplica al software comercial, sino también a sistemas internos de gestión, portales académicos, aplicaciones móviles y plataformas de servicios públicos.

Desde una perspectiva centrada en la usabilidad del software, se llevó a cabo un análisis exhaustivo de diversas páginas HTML utilizadas en el sitio del expositor del taller de accesibilidad. Estas páginas abarcan componentes clave del diseño de interfaces digitales: estructura general del documento, navegación mediante enlaces, uso de formularios, despliegue de tablas informativas y manejo de imágenes. La evaluación se fundamenta en principios de diseño centrado en el usuario, particularmente aquellos propuestos por Nielsen (2020), Norman (2013), las normas International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) (2011) y las mejores prácticas en usabilidad y experiencia de usuario (UX).

4.1.1 Estructura del documento HTML

Las páginas presentan una estructura semánticamente clara, organizada a través de secciones como encabezados, navegación y áreas de contenido principal. Esta disposición se alinea con los principios de consistencia estructural y afinidad conceptual, al ofrecer al usuario una delimitación visual coherente de las funcionalidades y contenidos. El uso correcto de etiquetas `<header>`, `<nav>`, `<main>`, `<section>` y `<footer>` promueve una comprensión fluida de la jerarquía de la interfaz, facilitando la exploración del sistema sin sobrecarga cognitiva.

No obstante, se identifican oportunidades de mejora en la jerarquización de los encabezados (`<h1>` a `<h6>`), ya que en algunos casos se rompe la secuencia lógica esperada. Esta disrupción puede obstaculizar tanto la navegación visual como el uso eficiente por parte de tecnologías de asistencia, afectando directamente la predictibilidad y la visibilidad del estado del sistema Shneiderman et al. (2017).

4.1.2 Navegación y enlaces

El análisis reveló un uso generalizado de enlaces mediante la etiqueta `<a>`, con textos que en su mayoría describen adecuadamente su destino o función. Esta práctica promueve el principio de reconocimiento antes que recuerdo, ya que permite a los usuarios identificar de inmediato las acciones disponibles sin requerir memorización previa Nielsen (2020).

Sin embargo, también se encontraron enlaces con descripciones ambiguas como “clic aquí” o sin contexto suficiente, lo cual constituye una violación al principio de prevención de errores y puede generar incertidumbre. Asimismo, en algunas páginas no se proporciona retroalimentación visual clara al pasar el cursor sobre un enlace, lo que debilita la visibilidad de la interacción y el control del usuario sobre su navegación.

4.1.3 Formularios

Las HTMLs incluyen formularios con etiquetas `<form>` correctamente estructuradas y campos acompañados por `<label>` asociadas a sus respectivos elementos de entrada. Este enfoque promueve la correspondencia entre el sistema y el mundo real, facilitando que los usuarios comprendan intuitivamente el propósito de cada campo.

No obstante, se detectó la ausencia de validación inmediata de datos, retroalimentación en caso de errores y ayudas contextuales como mensajes emergentes o sugerencias de formato. Estas omisiones comprometen los principios de diagnóstico y recuperación de errores y reducen la eficiencia para usuarios expertos, quienes esperarían mayor flexibilidad y funcionalidad avanzada.

4.1.4 Uso de tablas

Las tablas HTML analizadas están correctamente construidas mediante las etiquetas `<table>`, `<thead>`, `<tbody>` y `<th>`, lo que proporciona una estructura robusta para organizar información. Desde el enfoque de usabilidad del software, esta implementación contribuye a la escaneabilidad visual, permitiendo al usuario comparar datos de forma rápida y eficiente.

No obstante, se observó que las tablas carecen de mecanismos de interacción como ordenamiento de columnas, filtros dinámicos o paginación, elementos esperados en sistemas interactivos actuales. La falta de estas funciones limita la eficiencia de uso y la flexibilidad para tareas complejas, reduciendo el valor informativo del componente en contextos donde la manipulación de datos es crucial.

4.1.5 Uso de imágenes

En las páginas del expositor se implementa correctamente la etiqueta `` para la inserción de imágenes. Sin embargo, se identificaron casos en los que los atributos `alt` son vagos, genéricos o directamente inexistentes. Desde una perspectiva centrada en usabilidad del software, las imágenes no solo cumplen un rol decorativo, sino que deben integrarse funcionalmente al flujo de interacción para reforzar la comprensibilidad y la afinidad conceptual del contenido Norman (2013).

Además, se observó la ausencia de funcionalidades complementarias como `aria-label` o `title` en imágenes con valor interactivo, lo cual disminuye la accesibilidad semántica del sistema. En entornos donde las imágenes representan acciones o estados del sistema, estos atributos son fundamentales para garantizar una experiencia consistente y fluida.

4.2 Reflexión sobre la usabilidad respecto al taller de accesibilidad

El análisis detallado de las HTMLs de las páginas del expositor del taller de accesibilidad permite extraer conclusiones relevantes sobre la implementación práctica de principios de usabilidad en contextos reales de desarrollo. Si bien el sistema demuestra una comprensión general de los fundamentos del diseño centrado en el usuario, como la estructuración semántica de los contenidos y el etiquetado básico de formularios, se evidencia una brecha importante entre el cumplimiento mínimo de estándares y la excelencia en experiencia de usuario.

Desde el enfoque de usabilidad del software, se identifican áreas donde la interfaz carece de elementos que potencien la eficiencia, la flexibilidad y la adaptabilidad a distintos perfiles de usuario. La ausencia de validaciones dinámicas, retroalimentación contextual, interacción enriquecida con datos tabulares y descripciones funcionales de imágenes refleja un diseño que prioriza la presentación estática por encima de la optimización funcional de la interacción.

En un entorno digital moderno, donde los usuarios interactúan con múltiples sistemas altamente refinados, las expectativas sobre eficiencia, control y claridad son elevadas. Por tanto, la usabilidad del software no puede abordarse únicamente desde el cumplimiento estructural del HTML, sino que debe comprenderse como una disciplina integral que conecta la arquitectura de información, la ingeniería de interfaces, la psicología cognitiva y las buenas prácticas de desarrollo.

La mejora de estos aspectos no solo beneficiaría la experiencia individual del usuario, sino que elevaría la calidad general del sistema, reduciendo errores, incrementando la retención, y fortaleciendo la percepción de profesionalismo y confiabilidad. En consecuencia, se recomienda incorporar ciclos iterativos de evaluación heurística, pruebas de usuario y revisión por expertos en usabilidad, a fin de construir sistemas no solo accesibles, sino también altamente utilizables, sostenibles y centrados en las verdaderas necesidades cognitivas y operativas de sus usuarios.

4.3 Reflexión crítica

Desde una perspectiva crítica, la experiencia obtenida en el curso permitió reconocer que la usabilidad es mucho más que un conjunto de principios de diseño o listas de verificación. Se trata de una filosofía de construcción de software centrada en las personas, en sus capacidades cognitivas, emocionales y contextuales. En lugar de asumir un “usuario promedio”, el diseño centrado en el usuario invita a comprender las limitaciones, necesidades y preferencias de una diversidad de perfiles, incluyendo personas con discapacidades, adultos mayores o usuarios con baja alfabetización digital (World Wide Web Consortium (W3C) (2023)).

El enfoque de la investigación promovió la incorporación de estos valores a través de prácticas colaborativas y reflexivas. Por ejemplo, al realizar análisis heurísticos de interfaces reales, se revelaron patrones de exclusión involuntaria como botones sin contraste, textos no escalables, ausencia de etiquetas para lectores de pantalla que hubieran pasado desapercibidos en una revisión funcional estándar. Estos hallazgos evidencian la importancia de adoptar una ética del diseño, donde la accesibilidad y la equidad no sean criterios opcionales, sino inherentes al desarrollo.

Asimismo, la reflexión condujo a reconocer que muchas decisiones de diseño se toman sin evidencia empírica, basadas en intuición o estética subjetiva. Este sesgo puede generar soluciones elegantes pero ineficientes o incluso contraproducentes. En este sentido, la integración de herramientas como Quarto y metodologías reproducibles fortalece la disciplina técnica del diseño, permitiendo argumentar y justificar decisiones con datos, estándares y retroalimentación del usuario.

Otro aspecto clave que emergió de la investigación fue la interdependencia entre usabilidad y otras dimensiones de calidad. Por ejemplo, una aplicación con alta eficiencia técnica pero baja usabilidad puede ser percibida como poco confiable o difícil de usar. Esta percepción negativa afecta la satisfacción general y puede repercutir en la imagen institucional del sistema. En este sentido, las normas ISO promueven una visión holística de la calidad, donde la usabilidad se convierte en una mediadora entre los objetivos de negocio y las experiencias del usuario final (International Organization for Standardization (2011)).

Finalmente, la investigación realizada resalta la importancia de formar profesionales que no solo dominen lenguajes de programación o estructuras de datos, sino que también comprendan profundamente la dimensión humana del software. La usabilidad, en este contexto, se revela como un puente entre la ingeniería y la empatía, entre el algoritmo y la experiencia. Este enfoque integral será clave para enfrentar los desafíos futuros del desarrollo tecnológico con responsabilidad social y sostenibilidad.

5 Conclusiones

La usabilidad se ha consolidado como un eje transversal en el diseño de sistemas de software, al garantizar que las soluciones tecnológicas sean accesibles, eficientes, intuitivas y satisfactorias para los usuarios finales. A lo largo de este informe se ha evidenciado cómo la usabilidad no solo implica un conjunto de buenas prácticas estéticas o superficiales, sino que está profundamente vinculada con estándares internacionales, principios de diseño centrado en el usuario y metodologías formales de evaluación como la escala SUS, la evaluación heurística y las pruebas con usuarios reales.

Desde la conceptualización histórica hasta su aplicación práctica en entornos reales, queda claro que la usabilidad incide directamente en el éxito de los productos digitales, influyendo en indicadores clave como la adopción del sistema, la fidelización del usuario, la reducción de errores y la satisfacción general. Las normas ISO 9241-11 e ISO/IEC 25010 proporcionan marcos robustos para comprender y evaluar este atributo, mientras que referentes como Jakob Nielsen y Don Norman aportan principios esenciales que siguen vigentes en la práctica profesional actual.

Asimismo, se ha destacado la importancia de integrar la usabilidad desde las etapas iniciales del ciclo de vida del software, favoreciendo un enfoque proactivo en lugar de correctivo. Este enfoque reduce significativamente los costos asociados a la corrección de errores de diseño y mejora la alineación entre los objetivos de negocio y las necesidades del usuario final (Ritter et al. (2020)).

Finalmente, la reflexión crítica ha permitido identificar cómo los conceptos estudiados pueden aplicarse en escenarios reales de desarrollo, así como reconocer el valor de una formación técnica sólida que incorpore herramientas como Quarto, estándares de calidad y pensamiento crítico en torno a la experiencia del usuario.

6 Bibliografía

- Andre, T. S., Hartson, J., & Williges, A. (2001). The usability methods toolbox: An overview of usability evaluation methods. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 45, 1739-1743. <https://journals.sagepub.com/doi/10.1177/154193120104502611>
- Bevan, N., Carter, J., & Harker, S. (2016). ISO standards for measuring usability and user experience. En *Lecture Notes in Computer Science* (Vol. 9731, pp. 268-278). https://link.springer.com/chapter/10.1007/978-3-319-39510-4_25
- Brooke, J. (2013). SUS: A retrospective. *Journal of Usability Studies*, 8(2), 29-40. <https://uxpajournal.org/sus-a-retrospective/>
- Card, S. K., Moran, T. P., & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates.
- Ferreira, A. R., & Silva, A. R. (2020). Heuristic evaluation revisited: Prioritizing usability problems. *Journal of Systems and Software*, 169, 110703. https://www.researchgate.net/publication/381455092_Towards_a_Refined_Heuristic_Evaluation_Incorporating_Hierarchical_Analysis_for_Weighted_Usability_Assessment3
- Freitas, J. D., Ribeiro, A. L., & Rocha, A. da. (2021). Accessibility evaluation of mobile banking applications in Latin America: Compliance with WCAG 2.1. *Journal of Web Engineering*, 20(7), 2175-2196. <https://www.sciencedirect.com/science/article/pii/S1877050915030896>
- International Organization for Standardization. (2011). *ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*.
- International Organization for Standardization. (2018). *ISO 9241-11:2018 Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts*.
- International Organization for Standardization / International Electrotechnical Commission. (2011). *ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*.
- International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC). (2011). *Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models (ISO/IEC 25010:2011)*. International Organization for Standardization. <https://www.iso.org/standard/35733.html>
- LinkedIn UX Research. (2019). *Enhancing learner engagement through UX redesign*.
- Nielsen, J. (2020). *10 Usability Heuristics for User Interface Design*. <https://www.nngroup.com/articles/ten-usability-heuristics/>.
- Norman, D. A. (2013). *The Design of Everyday Things* (Revised and expanded edition). MIT Press.
- Parker, D. (2016). Designing for Digital Government: Lessons from GOV.UK. *UXPA Journal*, 15(3), 28-35.
- Ritter, F. E., Baxter, G. D., & Churchill, E. F. (2020). *Foundations for Designing User-Centered Systems: What System Designers Need to Know About People* (2nd ed.). Springer. https://www.cur.ac.rw/mis/main/library/documents/book_file/2014_Book_

- Sauro, J., & Lewis, J. R. (2016). *Quantifying the User Experience: Practical Statistics for User Research* (2nd ed.). Morgan Kaufmann.
- Shackel, B. (1991). Usability—Context, Framework, Definition, Design and Evaluation. En B. Shackel & S. Richardson (Eds.), *Human Factors for Informatics Usability* (pp. 21-38). Cambridge University Press.
- Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2017). *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (6th ed.). Pearson.
- UK Government Digital Service. (2023). *GOV.UK Design System*.
- World Wide Web Consortium (W3C). (2023). *Web Content Accessibility Guidelines (WCAG) 2.2*. <https://www.w3.org/TR/WCAG22/>.
- Zhang, S., Zeng, W., Huang, Z., & Wang, C. (2021). Usability evaluation methods and their applications in software engineering. *ACM Computing Surveys*, 54(9), 1-36. <https://dl.acm.org/doi/10.1145/3453474>