

Sistemas de diseño, UI Kits y guidelines en el diseño de interfaces digitales: Una revisión sistemática

Autor

Lic. Rodrigo Iver Romero Frias

Instituto de Investigaciones del Hábitat, Facultad de Arquitectura y Cs. del Hábitat,
Universidad de San Francisco Xavier, Carrera de Arte y Diseño Gráfico.

Resumen

En la actualidad se convive con numerosas interfaces gráficas digitales en diferentes dispositivos y con diferentes tecnologías, el proceso de diseño e implementación de estas interfaces exige un flujo de trabajo ágil y eficiente; al mismo tiempo que se deben cumplir criterios de calidad de diseño de interfaces y experiencia de usuario.

En el presente artículo se realiza una revisión literaria sistemática para explorar el UI Kit como herramienta en el proceso de diseño de interfaces gráficas digitales, su relación con los guidelines o pautas de los sistemas de diseño; y se analiza su impacto en principios de diseño de interfaces y en el trabajo con el equipo de desarrollo.

Los resultados obtenidos indican que el establecimiento de un UI Kit pueden mejorar en la eficiencia, comunicación del equipo de trabajo, la coherencia visual y de interacción, aumentando la satisfacción.

Adicionalmente, se discute sobre los desafíos y limitaciones de aplicar los sistemas de diseño, el establecimiento de UI Kits para proponer interfaces consistentes, intuitivas, que brinden una UX satisfactoria y que faciliten su desarrollo, implementación y uso, según buenas prácticas y guidelines de estándares de sistemas de diseño y librerías (frameworks). Finalmente se explora sobre las perspectivas futuras de los sistemas de diseño.

Introducción

En la era de la tecnología y la comunicación digital, el diseño de interfaces digitales es cada vez más relevante. La calidad de una interfaz digital puede ser determinante en el éxito de una aplicación o sitio web, y se requiere que sea intuitiva y atractiva. Para facilitar esta tarea, se han creado herramientas esenciales como los UI Kits y las Guidelines, que proporcionan recursos para diseñar interfaces digitales coherentes y adaptables.

Desarrollo

Los sistemas de Diseño

En 1960 las computadoras se hicieron rápidas y económicas, sin embargo, existía una “crisis del software” por su costo, lentitud y dificultad en mantenimiento. Con lo que en 1968 en la conferencia de la OTAN sobre ingeniería de software, Douglas McIlroy propuso el desarrollo basado en componentes, pudiendo reutilizarse código para mejorar la eficiencia y escalabilidad. Posteriormente, desde 1950 se experimentó un reto similar en el diseño, ya que se requiere una solución a medida específica para cada problema (Suarez et al., 2019).

Un Design System o Sistema de Diseño no se limita a ser una guía de estilo o una biblioteca de patrones, sino que es un modelo para el desarrollo de productos (Treder & Cao, 2017), un sistema de diseño establece de manera estandarizada pautas o directrices (guidelines), componentes y herramientas que fundamentan buenas prácticas de diseño de GUI; tanto en su apariencia, interacción y experiencia.

Los sistemas de diseño se documenta en un manual de diseño (manual de sistema de diseño visual) “guidelines” para su comunicación, consulta y aplicación, de tal forma que los diseñadores y desarrolladores puedan desarrollar productos de manera colaborativa, ágil y eficiente; así también, se brinde a los usuarios una UI y UX uniforme.

Problemas frecuentes en el diseño de GUI

De acuerdo con Treder & Cao (2017) en su trayectoria lograron identificar cuatro problemas frecuentes en el diseño y desarrollo de productos digitales:

- Inconsistencias entre productos y plataformas, se genera cuando los equipos no trabajan con las mismas plataformas.
- La falta de activos centralizados y de control de versiones, tiene que establecerse flujos de trabajo coordinados y protocolos de versionamiento.
- Ampliación de las brechas de conocimiento entre los equipos de productos, puede provocar falta de entendimiento entre equipos.
- Los procesos ineficientes conducen a un trabajo redundante o desperdiciado, como las reuniones extensas.

Adicionalmente, según Suarez et al. (2019). Cuando se realiza una auditoría de UI se identifican diversos problemas de diseño y buenas prácticas.

La deuda de diseño

La deuda de diseño se adquiere cuando se diseña a corto plazo, debido a la heterogeneidad de estilos, convenciones de elementos o acciones inconsistentes o no reutilizables, los cuales a largo plazo se van acumulando; el interés es la tarea de no poder brindar mantenimiento a esos elementos (Suarez et al., 2019).

El diseñar con perspectiva a corto plazo puede originarse principalmente por la urgencia que se tiene para dar una solución de diseño, falta de coordinación y acuerdo en el equipo, escasa claridad en las tareas asignadas, mala praxis en control de versionamiento y modificaciones injustificadas.

Figura 1. Deuda de diseño acumulada por la heterogeneidad de los elementos de diseño.



Fuente: (Suarez et al., 2019).

El valor de los sistemas de diseño

Stamas (2017) sugiere aplicar sistemas de diseño para mitigar los problemas mencionados, ya que pueden ofrecer una mejora significativa a nivel de equipo, proporcionando un proceso más eficaz y previsible que minimice el tiempo empleado en el diseño y el desarrollo, permitiendo a los diseñadores y desarrolladores enfocarse en tareas de mayor nivel.

El valor que brindan aplicar sistemas de diseño en las GUI es que permite que el diseño sea escalable, consistente, permite gestionar la “deuda de diseño”, agiliza el proceso de prototipado, facilita la iteración, incrementa la usabilidad y facilita construir con accesibilidad (Suarez et al., 2019).

Así también, según Stamas (2017), un sistema de diseño también puede contribuir a mejorar la calidad de la experiencia de usuario al ofrecer consistencia y previsibilidad en la interfaz.

Objeciones o limitaciones

En la aplicación de sistemas de diseño pueden surgir las siguientes objeciones: puede parecer que es limitante, que restringe la creatividad, o que después de estar diseñado y construido un sistema, el trabajo ha culminado.

Los UI Kits

Ulkit proviene de la abreviatura en inglés de user interface kit (Hernández Castro & Monge Fallas, 2014) en español se traduce como kits de interfaz de usuario, los UI Kits son un conjunto de componentes de diseño de interfaces normalizados de manera uniforme que se utilizan para diseñar prototipos de interfaces digitales coherentes y consistentes.

Estos kits normalmente incluyen elementos como botones, menús, formularios, iconos, tipografía, paletas de colores y otros elementos de diseño (SosaaTzec & Siegel, 2014; Zúñiga-Saldaña & Montiel-Cisneros, 2020) que se usan comúnmente en el diseño de GUI.

Los elementos de un UI Kit pueden variar dependiendo del kit en particular, pero generalmente incluyen una variedad de elementos de diseño predefinidos para ayudar a los diseñadores a crear interfaces de usuario coherentes. Los cuales se describen a continuación:

Botones: Los botones son elementos básicos en la mayoría de las interfaces de usuario. Los UI Kits pueden contener diferentes estados y tamaños de botones para adaptarse a diferentes necesidades de diseño. Los principales estados de los botones son:

- Normal, estado en reposo de un botón que puede ser interactivo.
- Hover o Sobre, estado aplicado cuando el cursor está posado sobre el botón, sin haber sido presionado.
- Active o Activo es el estado cuando el botón está siendo presionado.
- Disabled o Deshabilitado, por convención es grisáceo para dar entender que el botón no es interactivo. Se recomienda dar retroalimentación de porque no está activo y en caso de no ser necesario es mejor no mostrarlo.
- Focused o enfocado, estado que suele emplearse para sugerirse una acción dentro de otras como en llamadas a la acción o CTA (del inglés Call To Acción).

La diferencia de estados permite al usuario tener certeza si el botón es interactivo, puede insinuar realizar una acción y brinda retroalimentación sobre las acciones realizadas por el usuario.

Iconos: Los iconos son gráficos o símbolos utilizados para representar acciones, objetos o ideas. Los UI Kits pueden incluir una variedad de iconos comunes para su uso en las interfaces de usuario, los iconos tienen que mantener el mismo estilo gráfico.

Campos de formulario o inputs: Los formularios son una parte frecuente en el diseño de interfaces de usuario, y los UI Kits pueden incluir diferentes estados, estilos y diseños de campos de formulario para ayudar a los diseñadores a crear formularios coherentes, atractivos y cumpliendo pautas de diseño según el tipo de dispositivo.

Menús de navegación (Navbar y sidebar): Los menús de navegación se utilizan comúnmente en sitios web y aplicaciones para ayudar a los usuarios a encontrar el contenido que están buscando, brinda jerarquías y separa los diferentes apartados o secciones que puede tener la interfaz. Los UI Kits pueden proporcionar diferentes estilos y diseños de menús de navegación para ayudar a los diseñadores a crear una navegación clara y fácil de usar.

Gráficos: Los gráficos, que pueden incluir gráficos de barras, de líneas y circulares, son empleados para representar información de manera clara.

Tipografía: Se enfoca en el estilo y diseño del texto en la interfaz, con diferentes fuentes y tamaños para lograr una apariencia coherente y legible.

En general, un UI Kit puede incluir una amplia gama de elementos de diseño para ayudar a los diseñadores a crear interfaces de usuario coherentes y atractivas de manera rápida y eficiente.

Además de los componentes individuales, los UI Kits también pueden construirse o encontrarse pre-elaborados en sistemas de diseño, incluyendo diseños de pantalla completos y elementos interactivos, como carruseles, menús desplegables, pestañas y más. Estos elementos preconstruidos ayudan a los diseñadores a prototipar rápidamente interfaces coherentes y profesionales, sin tener que crear cada elemento desde cero.

Cómo se utilizan los UI Kits

Los UI Kits se emplean comúnmente en el diseño de interfaces digitales mediante programas de diseño gráfico como Figma, Sketch, Adobe XD, JustinMind, UXPin, InVision Studio, MockFlow, Axure RP, ProtoPie, Origami Studio, Fluid UI, Framer, Marvel, Proto.io, Principle o Balsamiq (Cardello, 2022; Geysler, 2022; Myre, 2022; Tidwell et al., 2022) otros diseñadores también usan Adobe Illustrator, Adobe Photoshop; aunque Photoshop al ser más un software para el tratamiento de imágenes raster, ya no se recomienda usar como software principal para diseñar UI Kits, sino más bien, como un optimizador de imágenes raster para web.

En el caso de Adobe Illustrator es empleado para generar gráficos vectoriales, los elementos del UI Kit se importan al programa de prototipado y se pueden ajustar y personalizar según sea necesario para diseñar la interfaz.

Además, numerosos UI Kits están disponibles en línea tanto de forma gratuita como de pago para que los diseñadores puedan acceder a ellos y utilizarlos en sus proyectos. Algunos UI Kits incluso vienen con archivos de demostración que se pueden descargar y emplear como plantilla para el diseño de interfaces, un buen ejemplo es la comunidad de Figma (Figma, s. f.) .

Para la creación de un UI Kit personalizado, se tiene que considerar el sistema donde

funcionará la interfaz en Android, iOS, Mac OS, Windows. Linux, sitio web, u otro, ya que los sistemas pueden establecer pautas de diseño o Guidelines, los cuales pueden exigir o sugerir para poder generar una GUI según sus estándares.

En el caso de realizar para un sitio web se puede optar por aplicar algún framework, los cuales son agilizan el desarrollo.*** La modificación de los elementos de diseño, como la paleta cromática, paleta tipográfica, y otros elementos de identidad; pueden realizarse basandose en la identidad visual del proyecto, comunmente normalizados en un manual de marca.

¿Qué son las Guidelines?

Las Guidelines o pautas de diseño, son un conjunto de reglas y principios que definen cómo debe ser diseñada una interfaz digital para garantizar la coherencia y consistencia en todo el producto o sitio web. Estas guías proporcionan instrucciones detalladas sobre la tipografía, los colores, los tamaños de los botones, la disposición de los elementos y otros aspectos del diseño.

Las guidelines suelen ser desarrolladas por los propios desarrolladores o diseñadores de un producto o sitio web, y se basan en los principios de diseño y las mejores prácticas que se han demostrado efectivas en el pasado. Las Guidelines también pueden ser específicas de una plataforma o sistema operativo, como iOS o Android, y proporcionar directrices sobre cómo diseñar interfaces coherentes y adaptables para esa plataforma en particular.

Principales sistemas de diseño

Los principales sistemas de diseño de interfaces de usuario (UI) y experiencia de usuario (UX) son los siguientes:

Human Interface Guidelines (Apple):

Son las directrices de diseño de Apple para sus dispositivos y sistemas operativos. Se centran en la creación de interfaces intuitivas, coherentes y atractivas que proporcionen una experiencia de usuario consistente y satisfactoria en el ecosistema de Apple, en sus diferentes plataformas: iOS, iPadOS, macOS, tvOS, wathcOS (Apple, 2019).

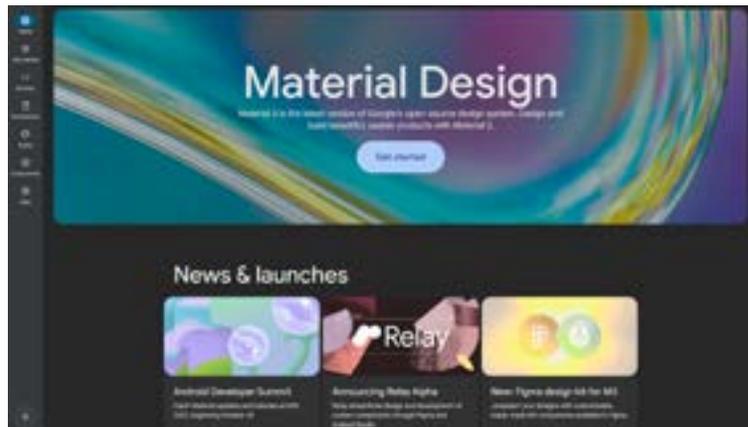
Figura 2. Vista de Human interface guidelines



Fuente: (Apple, 2019).

Material Design Guidelines (Google):

Figura 3. Vista de Material design 3 guideline.



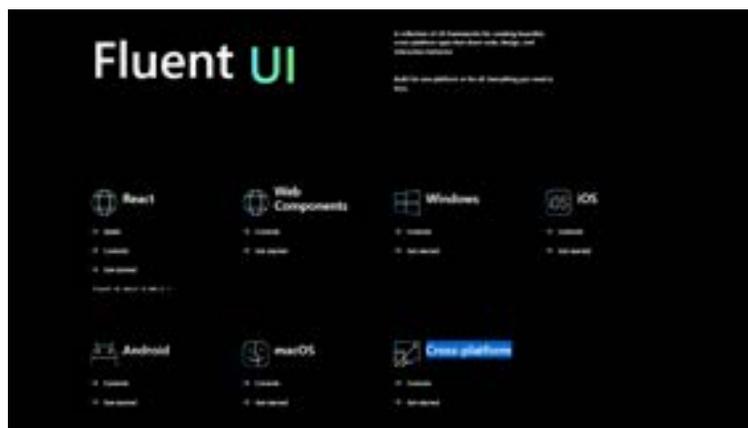
Fuente: (Google, 2021).

Son las directrices de diseño de Google para sus productos y servicios en línea. Se centran en la creación de interfaces visualmente atractivas y adaptables, con énfasis en la tipografía, la paleta de colores y los elementos de diseño (Google, 2021; Mew, 2015).

Fluent Design System (Microsoft):

Es una colección de (frameworks) o marcos de trabajo de UX multiplataforma de código abierto elaborado y usado por Microsoft para sus productos y servicios (Microsoft, 2020); el sistema de diseño comparte código, diseño e interacciones para diversas tecnologías como React, Web, como diversos sistemas operativos como Windows, iOS, Android y macOS.

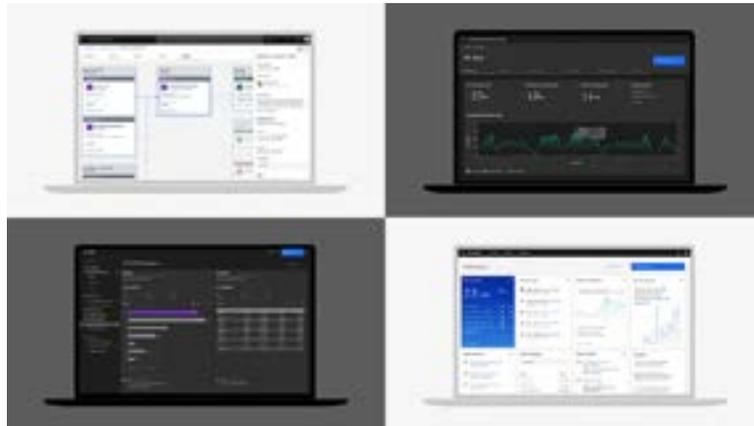
Figura 4. Fluent UI



Fuente: (Microsoft, 2020).

Carbon desing system (IBM):

Figura 5. Products using the Carbon Design System (IBM Cloud Pak for Data, IBM Watson Knowledge Catalog, IBM EPM, and IBM Cloud).



Fuente: (Rice, 2022)

Carbon Design system es un sistema de diseño de código abierto diseñado, patrocinado, mantenido y aplicado por IBM para productos y experiencias digitales. Con una comunidad activa de colaboradores (IBM, 2018). El 2022 fueron ganadores con el premio de diseño “Red Dot Design Award” el cual es el más prestigioso reconocimiento en la industria de diseño dentro de tres disciplinas: Diseño de Producto, Diseño de Marcas y Comunicación; y Diseño de Concepto (Rice, 2022).

Para conocer más sistemas de diseño se recomienda la galería de sistemas de diseño (Limcaco, 2021).

En general, el diseñador es necesario que conozca los diferentes sistemas de diseño y sus directrices, debido a que cada conjunto de directrices tiene sus propias características y se adapta a diferentes necesidades y requisitos de diseño. Un sitio recomendado es el repositorio de guidelines, el cual enlista diversos guidelines que podrían coadyuvar al equipo de diseño y desarrollo. En el repositorio de sistemas de diseño se puede encontrar una variedad amplia de sistemas de diseño (Limcaco, 2021).

Frameworks o librerías

Los framework front-end son una colección de código estandarizado pre-escrito para el desarrollo de aplicaciones web o móviles (moweb, 2022). Los framework más populares según lenguaje de programación son (Bravo, 2018; Patel, 2021).

- JavaScript: Angular, Ember, Vue.js, React, jQuery, Bootstrap, Qooxdoo
- PHP: Laravel, CodeIgniter, Symfony
- Java: Spring MVC, JSF, Struts
- Python: Django

La relación entre los sistemas de diseño y los frameworks está en que ambos facilitan el diseño de interfaces consistentes y eficientemente (Constant, 2021), proporcionan un lenguaje y estructura para la implementación de la UI (Mendesaltaren, 2023), contienen elementos reutilizables y personalizables que se adaptan a las necesidades y contextos (Bravo, 2018; Constant, 2021).

La diferencia entre los sistemas de diseño y los framework principalmente son las siguientes (Castillo, 2021; M, 2022):

- Los sistemas de diseño son pautas visuales y conceptuales que especifican la identidad y coherencia de una aplicación (Castillo, 2021). Los frameworks front-end son herramientas técnicas que facilitan la aplicación del código del diseño.
- Los sistemas de diseño tienen componentes reutilizables, aunque no es necesario que contengan código (Castillo, 2021). Por otra parte, las librerías son paquetes de código que proporcionan componentes predefinidos y personalizables.
- Mientras que las librerías contienen código prescrito, a menudo se centran en una sola tecnología o lenguaje, los sistemas de diseño se pueden aplicar en una variedad de plataformas y tecnologías.

Ventajas y desventajas de aplicar librerías (Bravo, 2018)

Ventajas:

- Los marcos brindan a los desarrolladores un enfoque organizado, lo que puede resultar en períodos de desarrollo más cortos y menos errores.
- Los marcos suelen proporcionar componentes preconstruidos, lo que puede ayudar a los desarrolladores a ahorrar tiempo y esfuerzo al crear funciones estándar.
- Los marcos han desarrollado comunidades que pueden ofrecer recursos y soporte a los desarrolladores.
- Los marcos pueden facilitar el proceso de actualización y mantenimiento del código a lo largo del tiempo.
- Los marcos pueden ayudar a garantizar que un equipo de desarrolladores escriba código de forma coherente.

Desventajas:

- Los marcos pueden ser constrictivos, lo que reduce la capacidad de los desarrolladores para modificar su código.
- Adquirir el conocimiento de un nuevo marco puede llevar mucho tiempo y esfuerzo.
- Los marcos pueden agregar complejidad innecesaria a un proyecto, lo que podría resultar en un código inflado y un rendimiento deficiente.
- Es posible que los marcos no siempre reflejen los desarrollos tecnológicos más recientes y se vuelvan anticuados con el tiempo.
- Los marcos pueden desarrollar riesgos de compatibilidad o seguridad al depender de un código externo.

Metodología de diseño de sistemas de diseño

Una metodología ampliamente reconocida y aplicada para el diseño de sistemas de diseño es Atom Design, de Frost (2016) que consiste en construir un sistema en cinco niveles jerárquicos:

Átomos, Moléculas, Organismos, Plantillas (Templates) y Páginas (Pages). Inicialmente, los átomos son los elementos básicos de las interfaces, incluidos los elementos HTML básicos, como etiquetas (labes) de formulario, entradas, botones y otros que si se desintegran dejan de ser funcionales (Frost, 2016). Es decir, los átomos de diseño están constituidos por tokens de diseño.

Figura 6. Metodología de diseño atómico (Atom design)



Tokens de diseño o valores de diseño

Basados en lo anterior

Los tokens son valores básicos de estilo de diseño estandarizados, centralizados y aprovechables de manera multiplataforma por el equipo transdisciplinario, principalmente diseñadores y desarrolladores. Permite al equipo trabajar de manera consistente. Los tokens ayudan a los equipos de producto a tomar decisiones de diseño con rapidez y confianza (Holgado, s. f.; Justin, 2021; Vele, 2023).

Diferencias entre variables regulares y tokens reside en que no se establecen valores individuales “hard coded values”(Justin, 2021; Salesforce Developers, 2016) sino que se establecen categorías. En vez de colocar el número de píxeles de determinado botón, se coloca la categoría del botón.

Algunos tipos de tokens de diseño:

Aspectos de composición

- Espaciado (píxeles)

Atributos de objetos

- Tipo de formas
- Opacidad (Porcentaje)
- Dimensión (ancho y alto)
- Estilos de borde (continua, segmentado)
- Radio de esquinas (Píxeles)
- Altura de la línea (puntos)

Valores de paleta tipográfica

- Nombre de fuentes tipográficas
- Familias de fuentes/estilos/pesos
- Interlineado

Valores de paleta cromática

- Color de fondo
- Color de texto
- Color de trazo
- Gradientes

Efectos

- Sombras de cuadro/texto (shadow)
- Desenfocado (Blur)

Otros.

- Duración de animación
- Z-indexes
- Media Queries
- Etcétera.

Cómo se aprecia en la figura 7, la variable regular `$blue-400` se puede aplicar en un preprocesador CSS como SASS. Sin embargo, el nombre de las variables no reflejan claramente el propósito de cada opción, su contexto de uso, “el para qué, dónde y cuándo se va a aplicar”. Con los tokens de diseño los pueden saber qué opción aplicar en determinado contexto.

El token: `$button-cta-background-color = $blue-400`; especifica que será aplicado como color de fondo para el botón de llamada a la acción (CTA) (Justin, 2021; Salesforce Developers, 2016). En otras palabras se establecen atributos globales para cada categoría de componente y no de individualmente a cada elemento.

Figura 7. Ilustración de la diferencia entre una variable (`blue-400`) y un token (`button-cta-background-color`).



Fuente: Adobe (2022).

Ventajas de aplicar tokens de diseño, Los tokens permiten un diseño flexible, Los tokens crean una apariencia unificada y Los tokens simplifican el proceso de desarrollo. Este último debido a que los tokens de diseño se pueden integrar en el código a través del paquete npm y así mantener el diseño actualizado sin modificar el código.

Consejos para administrar tokens

El futuro de los sistemas de diseño

Según Suarez et al. (Suarez et al., 2019), en el futuro, la aplicación de los sistemas de diseño va a ser más común. ***

Según la perspectiva de Thorsten Jankowski (2022) de Volkswagen, el diseñador podrá solicitar a la IA a través de solicitudes “prompts” como rastrear tokens o items de diseño aplicados hasta el momento y la IA identificará automáticamente los elementos de diseño de sitios web públicos y privados, de manera que los diseñadores de sistemas, no tengan que crear manualmente desde cero.

Así también, se incrementará el empleo de patrones y componentes estandarizados

según plataformas, por consecuencia se preocuparán 85% más en resolver los problemas y sus procesos; y 15% en la implementación de la UI. Las marcas aplicarán en sus UI 99% tecnologías estándar para generar aplicaciones desechables con la marca y el diseño para casos de uso específico.

Resultados

De la revisión realizada se pueden rescatar los siguientes resultados.

Resultado 1, Implicaciones al diseñar un sistema de diseño propio

Diseñar un sistema de diseño con visión a largo plazo puede conllevar bastante tiempo y recursos invertidos. Esto debido a la gran variedad de criterios y contextos que se tienen que tener en cuenta. Así también, llegar a un consenso en el equipo puede ser complejo. Por lo tanto, se debe valorar la conveniencia de usar un sistema de diseño existente o diseñar uno propio.

Resultado 2, Limitaciones al aplicar un sistema de diseño

El aplicar un sistema de diseño condiciona las soluciones de diseño, que si bien puede brindar consistencia, en casos concretos puede limitar la solución de diseño. El diseñador puede conformarse con dar soluciones genéricas a problemas particulares. Al aplicar soluciones convencionales o predeterminadas, puede dejar de lado el explorar nuevas posibilidades de dar solución específica a problemas particulares que sean más efectivas para el usuario. Probablemente en casos concretos, se tiene que decidir entre la eficiencia del diseño y desarrollo; y la eficiencia del usuario. El trabajar con sistemas de diseños y personalizarlos puede ser considerado como luchar contra el sistema de diseño.

Resultado 2, la aplicación de frameworks o librerías

Entre las ventajas que tiene el

Beneficios de aplicar Sistemas de Diseño

La aplicación de Design Systems en el diseño de GUI puede proporcionar una serie de beneficios, entre los que se incluyen:

Coherencia y consistencia: Los Design Systems ayudan a mantener una apariencia visual coherente y una experiencia de usuario consistente en toda la aplicación o producto, lo que puede mejorar la comprensión y la usabilidad de los usuarios.

Ahorro de tiempo y recursos: Los Design Systems pueden reducir el tiempo y los recursos necesarios para diseñar y desarrollar interfaces de usuario, ya que proporcionan componentes y directrices predefinidos que pueden ser reutilizados y adaptados para diferentes casos de uso.

Mejora de la calidad: Al proporcionar directrices claras y componentes bien diseñados, los Design Systems pueden mejorar la calidad de las interfaces de usuario y la experiencia del usuario en general.

Facilitación de la colaboración: Los Design Systems pueden ayudar a los equipos de diseño y desarrollo a colaborar más eficazmente al proporcionar un conjunto común de herramientas y directrices que todos pueden seguir.

Facilidad de mantenimiento: Los Design Systems pueden hacer que sea más fácil mantener y actualizar las interfaces de usuario, ya que todos los componentes y directrices se encuentran en un lugar centralizado.

En resumen, los Design Systems pueden mejorar la coherencia, ahorrar tiempo y recursos, mejorar la calidad, facilitar la colaboración y hacer que sea más fácil mantener y actualizar las interfaces de usuario en el diseño de GUI.

Resultado 2, Beneficios del uso de UI Kits basados en Guidelines***

El uso de UI Kits y Guidelines tiene varios beneficios importantes para los diseñadores y desarrolladores de interfaces digitales. En inicialmente, ayuda a garantizar la coherencia y la consistencia en todo el producto o sitio web, lo que puede mejorar la experiencia del usuario y aumentar la eficacia de la interfaz. Además, la utilización de UI Kits y Guidelines puede acelerar el proceso de diseño, ya que los elementos preconstruidos y las directrices pueden ahorrar tiempo y esfuerzo al diseñador.

Otro beneficio relevante es que el empleo de UI Kits y Guidelines puede ayudar a reducir los errores y los problemas de usabilidad. Al utilizar componentes preconstruidos y seguir directrices específicas de diseño, los diseñadores pueden evitar errores comunes y garantizar que la interfaz sea fácil de usar y accesible para todos los usuarios.

Adicionalmente, el emplear UI Kits es que se pueden adaptar a diferentes dispositivos y tamaños de pantalla. Al diseñar una interfaz para diferentes tamaños de pantalla, es importante asegurarse de que los elementos de diseño se ajusten de manera óptima a la pantalla y sean fáciles de aplicar en cada dispositivo. Los UI Kits proporcionan una solución para esto al ofrecer elementos de diseño adaptables que se pueden aprovechar en diferentes tamaños de pantalla.

Resultado 2, La inteligencia artificial generativa en el proceso de diseño

El proceso de diseño y desarrollo de sistemas de diseño como de las GUI va a ser acompañado con el uso de inteligencia artificial. Se podrá aplicar inteligencia artificial generativa para generar aspectos estandarizados de la interfaz, y se pondrá más atención en la experiencia de usuario, en la resolución de problemas del usuario como en el proceso que el usuario tiene que atravesar.

Las tecnologías permitirán la optimización en tiempos y recursos en aspectos generales.

La ideación será asistida por inteligencia artificial generativa (IAG), facilitando explorar múltiples soluciones en menor tiempo, visualización multiplataforma y posteriormente aplicar personalizaciones.

Conclusiones

En conclusión, los Sistemas de Diseño y los UI Kits son herramientas esenciales para el diseño de GUI consistentes, de manera eficiente y escalable.

Los Design Systems pueden mejorar la coherencia, ahorrar tiempo y recursos, mejorar la calidad, facilitar la colaboración y hacer que sea más fácil mantener y actualizar las interfaces de usuario en el diseño de GUI. Los UI Kits pueden proporcionar componentes de diseño preestablecido y directrices detalladas, estos recursos pueden ayudar a los diseñadores a crear interfaces más eficaces, coherentes y fáciles de usar.

Es responsabilidad de los diseñadores, el ser crítico y revisar las posibles personalizaciones que se tengan que realizar a partir de las necesidades del usuario y casos de uso; así también, para una comunicación eficiente entre el diseñador y el equipo de desarrollo, se sugiere que el diseñador UX/UI al menos conozca sobre HTML y CSS. Finalmente, la tecnología como la inteligencia artificial está y continuará transformando el diseño y desarrollo de GUI.

BIBLIOGRAFÍA

Adobe. (2022). Spectrum, Adobe's design system. <https://spectrum.adobe.com/>

Apple. (2019). Designing for iOS - Platforms—Human Interface Guidelines—Design—Apple Developer. <https://developer.apple.com/design/human-interface-guidelines/platforms/designing-for-ios/>

Bravo, L. (2018). Framework o librerías: Ventajas y desventajas. tiThink Technology. <https://www.tithink.com/es/2018/08/29/framework-o-librerias-ventajas-y-desventajas/>

Cardello, J. (2022, junio 15). 14 best prototyping tools for UI/UX designers in 2023. <https://webflow.com/blog/prototyping-tools>

Castillo, M. (2021, enero 10). ¿Cómo elegir Diseño UX/UI vs Desarrollo Front End? Desafío Latam. <https://blog.desafiolatam.com/disenio-ux-ui-o-desarrollo-front-end/>

Constant, I. (2021). ¿Qué es un sistema de diseño y cómo puedo usarlo en mis proyectos digitales? | Domestika. <https://www.domestika.org/es/blog/6334-que-es-un-sistema-de-disenio-y-como-puedo-usarlo-en-mis-proyectos-digitales>

Figma. (s. f.). Figma. Recuperado 21 de febrero de 2023, de https://www.figma.com/login_iframe

Frost, B. (2016). Atomic design. Brad Frost Pittsburgh. <http://atomicdesign.bradfrost.com/chapter-1/>

Geyser, W. (2022, marzo 31). Top 10 UX/UI Design Tools in 2023. Influencer Marketing Hub. <https://influencermarketinghub.com/ux-ui-design-tools/>

Google. (2021). Material Design. Material Design. <https://m3.material.io/>

Hernández Castro, F., & Monge Fallas, J. (2014). Dibujando 2D en iOS (Nota técnica). Tecnología en Marcha, 27(Extra 6), 9.

Holgado, J. L. (s. f.). ¿Qué son los tokens de diseño? Recuperado 5 de enero de 2023, de <https://www.atsistemas.com/es/blog/que-son-tokens-disenio>

IBM. (2018). Carbon Design System. <https://www.carbondesigsystem.com>

Jankowski, T. (Director). (2022, julio 2). The Future of Design Systems—Thorsten Jankowski Volkswagen. https://www.youtube.com/watch?v=eIS_7mMCcFQ

Justin, M. (2021). What Are Design Tokens? A Design Systems Tool | Adobe XD Ideas. Ideas. <https://xd.adobe.com/ideas/principles/design-systems/what-are-design-tokens/>

Limcaco, J. (2021). Design Systems Repo. Design Systems Repo. <https://designsystemsrepo.com/>

M, L. (2022, agosto 3). Mejores Frameworks FrontEnd: Descubre qué framework usar. BitDegree.org Online Learning Platforms. <https://es.bitdegree.org/tutoriales/frameworks-frontend/>

Mendesaltaren. (2023). Handbook. <https://www.mendesaltaren.com/handbook>

Mew, K. (2015). Learning Material Design. Packt Publishing Ltd.

Microsoft. (2020). Fluent UI - Get started—Fluent UI. <https://developer.microsoft.com/en-us/fluentui#/>

Moweb. (2022). 5 Best Front-End Frameworks of 2023. Moweb. <https://www.moweb.com/blog/best-front-end-frameworks>

Myre, M. (2022, diciembre 13). 11 Best UX Prototyping Tools for UX/UI Designers. <https://designlab.com/blog/best-prototyping-tools-for-UX-UI-designers/>

Patel, J. (2021, noviembre 24). 10 Best Front end Frameworks for Web Development in 2023. <https://www.monocubed.com/blog/best-front-end-frameworks/>

Rice, L. (2022, agosto 23). Carbon Design System wins 2022 Red Dot Design Award. _carbondesign. <https://medium.com/carbondesign/carbon-design-system-wins-2022-red-dot-design-award-e200808c33ab>

Salesforce Developers (Director). (2016, octubre 26). Using Design Tokens with the Lightning Design System. <https://www.youtube.com/watch?v=wDBEc3dJJV8>

SosaaTzec, O., & Siegel, M. (2014). Principios de diseño visual para HCI. Ann Arbor, 1001, 48109.

Stamas, N. (2017, febrero 13). Selling a Design System at Your Company. Medium. <https://uxdesign.cc/selling-a-design-system-at-your-company-74cb2bc97195>

Suarez, M., Anne, J., Saylor-Miller, K., Mounter, D., & Stanfield, R. (2019). Design Systems Handbook. Design Systems Handbook - DesignBetter. <https://www.designbetter.co/design-systems-handbook>

Tidwell, J., Brewer, C., & Valencia, A. (2022, septiembre 30). 14 Best UI Design Tools and Software [Free, Paid + Open Source]. <https://www.hotjar.com/ui-design/tools/>

Treder, M., & Cao, J. (2017). Why Build a Design System? <https://www.uxpin.com/studio/ebooks/design-systems-why-build-one/>

Vele, M. (2023, enero 26). Diseño UI - Tokens. <https://es.linkedin.com/pulse/dise%C3%B1o-ui-tokens-mario-velez>

Zúñiga-Saldaña, N., & Montiel-Cisneros, J. M. (2020). Diseño gráfico, UI y UX de sistema de monitoreo ambiental.