

Juan esteban Grajales mesa

Consulta de saberes básicos

1 un sistema de diseño es un conjunto de componentes reutilizables que se pueden usar para facilitar el desarrollo de web y reduciendo el tiempo de producción.

El propósito principal es el de reducir el tiempo, los costos en la producción del producto y estandarizar los diseños para una mejor comprensión

2

- Patrones de diseño: Soluciones reutilizables para problemas de diseño comunes (botones, formularios, tarjetas, etc.).
- Guía de estilo: Reglas para tipografía, color, espaciado y otros elementos visuales.
- Biblioteca de componentes: Conjunto de componentes prediseñados y reutilizables.
- Documentación: Explicación detallada de cómo utilizar el sistema.

3 Una guía de estilo se enfoca en los aspectos visuales, mientras que un sistema de diseño abarca un conjunto más amplio de reglas y componentes, incluyendo la estructura y la interacción.

4 algunas Populares son:

- **Frameworks de diseño:** Bootstrap, Material Design, Ant Design.
- **Herramientas de diseño:** Figma, Sketch, Adobe XD.
- **Bibliotecas de componentes:** React, Vue, Angular.

5

- **Modularidad:** Divide el sistema en componentes pequeños y reutilizables.
- **Versionamiento:** Utiliza un sistema de control de versiones para rastrear cambios.
- **Documentación clara:** Mantén la documentación actualizada.
- **Retroalimentación:** Involucra a diseñadores y desarrolladores en la evolución del sistema.

6

- **Herramientas de diseño colaborativo:** Utiliza plataformas como Figma o Sketch.
- **Lenguaje común:** Define un vocabulario compartido para facilitar la comunicación.
- **Workshops y reuniones:** Organiza sesiones para alinear expectativas.

7

- **Directrices de accesibilidad:** Sigue pautas como WCAG.
- **Pruebas de accesibilidad:** Utiliza herramientas para evaluar la accesibilidad.
- **Consideraciones en el diseño:** Asegúrate de que los componentes sean accesibles desde el principio.

8 los tokens de diseño son variables que almacenan valores como colores, espaciados y tipografías. Facilitan la personalización y la coherencia en todo el sistema.

9

- **Velocidad de desarrollo:** Mide la reducción en el tiempo de desarrollo de nuevas características.
- **Consistencia:** Evalúa la uniformidad visual y de experiencia de usuario.
- **Reutilización de código:** Calcula el porcentaje de código reutilizado.
- **Satisfacción del equipo:** Realiza encuestas para conocer la opinión del equipo.

10 el box model es un contenedor en el CSS que es fundamental para la modelación de los elementos dentro de este

El modelo de caja es fundamental para el diseño web, ya que permite, Controlar el tamaño y la posición de los elementos, Crear espacios entre elementos, Crear diseños complejos, Resolver problemas de diseño

11

El atributo display en CSS controla cómo se muestran los elementos en la página. Cada valor tiene un comportamiento diferente:

- **display: block**
Características:
 Ocupa todo el ancho disponible de su contenedor.
 Crea una nueva línea antes y después de sí mismo.
 Puede tener márgenes, rellenos y bordes.
Uso:
 Para elementos que deben ocupar todo el ancho de la pantalla o de su contenedor, como encabezados, párrafos o divs.
- **display: inline**
Características:
 No ocupa todo el ancho disponible.
 Se muestra en línea con otros elementos.

No puede tener márgenes ni rellenos.

Uso:

Para elementos que deben aparecer en línea con el texto, como enlaces, imágenes o spans.

- **display: inline-block**

Características:

Combina las características de block e inline.

Ocupa solo el ancho necesario.

Se muestra en línea con otros elementos.

Puede tener márgenes, rellenos y bordes.

Uso:

Para elementos que deben aparecer en línea, pero también tener márgenes, rellenos o bordes, como botones o iconos.

- **display: flex**

Características:

Crea un contenedor flexible que puede distribuir sus elementos de manera dinámica.

Permite controlar la alineación, la distribución y el orden de los elementos.

Uso:

Para crear diseños complejos y responsivos, como tarjetas, encabezados, o pies de página.

12

¿Qué es Flexbox?

Flexbox es un módulo CSS que proporciona un mecanismo flexible para organizar elementos en una línea o columna. Es especialmente útil para crear diseños responsivos que se adaptan a diferentes tamaños de pantalla.

Cuando usar Flexbox:

- **Diseños responsivos:** Cuando necesitas que tus diseños se ajusten automáticamente a diferentes tamaños de pantalla.
- **Alineación y distribución:** Cuando deseas controlar la alineación, la distribución y el orden de los elementos dentro de un contenedor.
- **Creación de layouts complejos:** Para crear layouts más complejos, como tarjetas, encabezados o pies de página.

Ventajas de Flexbox:

- **Flexibilidad:** Permite una gran flexibilidad en la disposición de los elementos.
- **Simplicidad:** Es relativamente fácil de aprender y usar.

- **Compatibilidad:** Es ampliamente compatible con los navegadores modernos.

Cuando no usar Flexbox:

- **Diseños estáticos:** Si no necesitas que tus diseños sean responsivos, puede ser más sencillo usar técnicas tradicionales como float.
- **Layouts extremadamente complejos:** Para layouts extremadamente complejos, puede ser necesario combinar Flexbox con otras técnicas.

13

¿Cómo funciona CSS Grid?

CSS Grid es un módulo CSS que proporciona un sistema de cuadrícula para organizar elementos en una página. A diferencia de Flexbox, que organiza elementos en una sola dirección (fila o columna), Grid permite organizar elementos en filas y columnas simultáneamente.

Componentes de CSS Grid:

- **Contenedor de cuadrícula:** El elemento raíz que define la estructura de la cuadrícula.
- **Ítems de cuadrícula:** Los elementos que se colocan dentro de la cuadrícula.
- **Líneas de cuadrícula:** Las líneas verticales y horizontales que forman la estructura de la cuadrícula.
- **Celdas de cuadrícula:** Los espacios entre las líneas de cuadrícula donde se colocan los ítems.

Ventajas de CSS Grid:

- **Flexibilidad:** Permite crear layouts complejos y flexibles con facilidad.
- **Control preciso:** Proporciona un control preciso sobre la colocación y el tamaño de los elementos.
- **Eficiencia:** Puede ser más eficiente que Flexbox para ciertos tipos de layouts.
- **Compatibilidad:** Es ampliamente compatible con los navegadores modernos.

14

¿Qué son los Media query?

La media query son reglas CSS que permiten aplicar estilos diferentes según las características del dispositivo o del navegador del usuario. Esto incluye factores como el ancho de pantalla, la resolución, la orientación (horizontal o vertical) y la densidad de píxeles.

Cómo se utilizan para crear un diseño responsivo

Identificar los puntos de ruptura: Determina los diferentes tamaños de pantalla a los que deseas adaptar tu diseño. Por ejemplo, podrías definir puntos de ruptura para dispositivos móviles, tabletas y computadoras de escritorio.

Crear estilos específicos: Define estilos CSS para cada punto de ruptura. Estos estilos pueden incluir cambios en el diseño, el tamaño de los elementos, la disposición y otros aspectos visuales.

Utilizar Media query: Aplica los estilos específicos a los puntos de ruptura correspondientes utilizando media query.

15

1. Uso adecuado de colores y contraste:

- **Contraste mínimo:** Asegúrate de que el contraste entre el texto y el fondo sea suficiente para que las personas con discapacidad visual puedan leerlo. Utiliza herramientas en línea para evaluar el contraste.
- **Evita combinaciones de colores que sean difíciles de distinguir:** Evita combinaciones de colores similares, como verde y amarillo, o azul y morado.

2. Tamaño de fuente y tipografía:

- **Tamaño de fuente:** Utiliza un tamaño de fuente legible para todos los usuarios, especialmente para aquellos con discapacidad visual.
- **Tipografía:** Elige fuentes sans-serif que sean fáciles de leer y legibles en diferentes tamaños.

3. Espaciado y legibilidad:

- **Líneas de texto:** Evita líneas de texto demasiado largas o cortas.
- **Espaciado entre líneas:** Utiliza el espaciado adecuado entre las líneas de texto para mejorar la legibilidad.

4. Encabezados y etiquetas:

- **Encabezados:** Utiliza los encabezados `<h1>`, `<h2>`, `<h3>`, etc., para estructurar el contenido y ayudar a los lectores con discapacidad visual a navegar por la página.
- **Etiquetas:** Utiliza etiquetas semánticas como `<label>`, `<input>`, `<button>`, etc., para describir los elementos del formulario y ayudar a los usuarios con lectores de pantalla.

5. Alternativas de texto para imágenes:

- **Atributo `alt`:** Proporciona una descripción textual de cada imagen utilizando el atributo `alt`. Esto es esencial para los usuarios con discapacidad visual que utilizan lectores de pantalla.

6. Tabulación y navegación:

- **Orden de tabulación:** Asegúrate de que el orden de tabulación sea lógico y fácil de seguir para los usuarios que navegan con el teclado.
- **Enlaces salteables:** Proporciona enlaces "saltar al contenido" para que los usuarios puedan omitir la navegación y acceder directamente al contenido principal.

7. Media query para dispositivos móviles:

- **Adaptación a diferentes tamaños de pantalla:** Utiliza Media query para ajustar el diseño y el tamaño de los elementos según el dispositivo del usuario. Esto puede mejorar la accesibilidad en dispositivos móviles.

8. Pruebas de accesibilidad:

- **Herramientas de evaluación:** Utiliza herramientas como WAVE, A11y Checker o Lighthouse para evaluar la accesibilidad de tu sitio web.
- **Pruebas con usuarios:** Realiza pruebas con usuarios con diferentes discapacidades para identificar cualquier problema de accesibilidad.

16

¿Qué es el cascading?

El cascading en CSS se refiere a la forma en que los estilos se aplican a los elementos HTML. Cuando múltiples reglas CSS se aplican a un mismo elemento, el navegador determina el orden en que se aplican esas reglas. Este proceso se conoce como "cascading" o "cascada".

Factores que afectan el orden de prioridad:

1. **Especificidad:** La especificidad de una regla CSS se determina por el número de elementos, clases e IDs que utiliza. Cuanto más específica sea una regla, mayor será su prioridad.
2. **Importancia:** Las reglas CSS definidas en hojas de estilo externas tienen una prioridad más alta que las reglas definidas en el elemento HTML.
3. **! important:** ¡Si se utiliza la declaración! important después de una propiedad CSS, esa propiedad tendrá la máxima prioridad, independientemente de otros factores.

17

¿Qué son los preprocesadores CSS?

Los preprocesadores CSS son lenguajes que extienden las capacidades de CSS al agregar nuevas características como variables, funciones, mixins, anidamiento y más. Estos lenguajes son compilados a CSS estándar que puede ser utilizado por los navegadores.

SASS y LESS: Comparación

- **SASS:**
 - Utiliza una sintaxis más similar a Ruby.
 - Ofrece funciones avanzadas como listas, mapas y control de flujo.
 - Tiene dos sintaxis: SCSS (similar a CSS) y Sass (indentación-based).
- **LESS:**
 - Tiene una sintaxis más similar a CSS.
 - Ofrece funciones y variables, pero con una sintaxis menos compleja que SASS.

Beneficios de usar preprocesadores CSS:

1. **Organización y mantenimiento:** Los preprocesadores permiten escribir CSS de manera más organizada y modular, lo que facilita el mantenimiento y la colaboración en proyectos grandes.
2. **Variables y funciones:** Las variables y funciones permiten definir valores y reutilizar código, lo que mejora la eficiencia y la consistencia.
3. **Anidamiento:** El anidamiento de selectores hace que el código CSS sea más legible y fácil de entender.
4. **Mixins:** Los mixins permiten crear conjuntos de reglas CSS que pueden ser reutilizados en diferentes lugares del código.
5. **Extensibilidad:** Los preprocesadores ofrecen características adicionales como operadores matemáticos, funciones de color y más, lo que amplía las posibilidades de diseño.

18

- **Minimizar archivos CSS:** Elimina comentarios y espacios en blanco innecesarios.
- **Utilizar preprocesadores CSS:** SASS o LESS pueden mejorar la organización y la eficiencia.
- **Reducir el número de peticiones HTTP:** Combina archivos CSS.
- **Evitar reglas CSS excesivamente específicas.**

19 El diseño mobile-first consiste en diseñar primero para dispositivos móviles y luego adaptar el diseño para pantallas más grandes. Se implementa utilizando media queries para aplicar estilos específicos según el tamaño de la pantalla.

20 Estas son tendencias de diseño que buscan crear interfaces más realistas y tridimensionales. El neomorfismo simula elementos físicos, mientras que el glassmorfismo crea la ilusión de vidrio. Se logran mediante el uso de sombras, degradados y efectos de desenfoque.

