Curso de Layouts Responsivos.

Clase 1

Emmet: Herramienta que viene por defecto en la gran mayoría de editores de código. Permite, al revés de escribir línea por línea de código, abreviar y generar los códigos más rápido.

Signo >: Deja claro que dentro de una etiqueta vendrá otra nueva. Siendo la correlación padre > hijo.

Signo +: Deja claro que dentro de una etiqueta hija, vendrán añadidas otras etiquetas siendo la correlación padre>hijo+hermano1+hermano2+hermano3.

Signo \*: Multiplica la cantidad de elementos que tendrá una etiqueta dentro de sí.

Medida REM: REM es una unidad relativa a la propiedad font-size de la etiqueta <html>, por lo tanto si la etiqueta <html> tiene 16px de font-size 1 REM equivale a 16px. REM proviene de “Root ephemeral” que en traducción técnica es la “variable de la raíz”, o sea, es una variable en relación de la propiedad font-size de la etiqueta raiz (HTML). Tiene el mayor nivel de jerarquía y será adaptable a todos los tamaños del navegador.

Medida EM: EM es relativa a la propiedad font-size de la etiqueta madre, por lo tanto, si la etiqueta madre es un font-size de 16px, 1 EM equivale a 16px. EM proviene de “ephemeral” que significa “variable” y es una variable de la propiedad font-size de la etiqueta madre.

Imagen PNG:

* Poco compacto
* Se mantiene la calidad de la imagen
* Tamaño elevado: 29.6Kb

Imagen JPG:

* Muy compacto
* La calidad de la imagen es inferior
* Tamaño leve: 9.29Kb

Imagen GIF:

* La calidad de la imagen es baja
* Uso limitado
* Tamaño depende de la calidad de la imagen: 199Kb

Imagen SVG:

* Al acercarse no se pierde tanto la calidad
* Imágenes vectoriales en dos dimensiones
* Tamaño variable: 4.26Kb,
* Algunas imágenes pueden perder color
* El SVG no mantiene detalles de una imagen compleja

¿Cuándo utilizar SVG?

* Logotipos
* Íconos
* Ilustraciones sencillas

Clase 2

Mobile First: Concepto en donde la versión móvil de una página web debe ser la correcta.

**Ventajas:**

* Se concentra en lo esencial.
* La información es de más fácil acceso.
* El rendimiento se potencia al máximo.
* Se reduce el código fuente.
* Se hace el uso de HTML 5, dejando de lado parcialmente JavaScript.

**Para saber más:**

Para seguir un concepto de desarrollo, sea mobile-first o sea desktop-first, puedes seguir lo que prefieras porque no existe una regla que indique cual sea primero. Cada uno de ello tiene sus ventajas y desventajas que deben ser evaluadas de acuerdo con las necesidades del proyecto.

Para demostrar, aquí están algunas motivaciones de utilizar el desarrollo mobile-first:

* Gran parte de los accesos y ventas son de dispositivos móviles;
* El diseño es minimalista y simplificado;
* Enfoque en el contenido.

Las motivaciones para utilizar el desarrollo desktop-first:

* La interfaz tiene features más ricas;
* Mayor capacidad de ejecución de las instrucciones;
* El producto es optimizado para desktop (ejemplo: Google Docs).

Así que al desarrollar es importante analizar todos los puntos que influencian en el proyecto. Clientes, features, costo y lenguaje visual son algunos de los puntos que favorecen un abordaje sobre el otro.

Meta Viewport: Habilita al navegador para que reconozca esa visualización de la pagina como si fuera un dispositivo móvil.

Initial Scale: Define cuanto zoom (o como se muestra el contenido) puede ser aplicado en la pagina.

MaxWidth:

BoxSizing:

Bloques: Usar propiedad margin.

Textos: Usar propiedad padding.

Variables CSS: Se pueden crear usando el selector :root {variables}. El root es una etiqueta html, pero su especificidad es mayor que la del selector html normal. La sintaxis de estas variables son las siguientes: --nombre-variable: valor;

Referenciar variables CSS: font-family: var(--open-sans);