# Layouts Responsivos: Trabajando con layouts mobile

## Conclusión 1: Herramientas, unidades e imágenes

* Analizar y separar el layout recibido;
* Utilizar el EMMET para crear bloques de código HTML;
* Utilizar las unidades relativas REM y EM;
* Tipos de imágenes y SVGs.
* <https://developer.mozilla.org/es/docs/Learn/HTML/Multimedia_and_embedding/Adding_vector_graphics_to_the_Web>

## Conclusión 2: Layout mobile y variables CSS

* Qué es y para qué sirve mobile-first;
* Cómo funciona el initial-scale dentro del contenido Viewport;
* Cómo declarar y llamar a las variables CSS.

## Conclusión 3: Particularidades de las pantallas

* Utilizar las unidades de viewport;
* Identificar la diferencia entre las unidades absolutas y relativas;
* Analizar las limitaciones físicas de los dispositivos.

## Conclusión 4: Semántica HTML

* Cómo trabajar con las imágenes del layout;
* Tags y los atributos que tienen más funciones de semántica de qué estilo;
* La importancia de escribir un código semántico.

## Conclusión 5: Evolución: Del mobile al desktop

* Crear estilos diferentes para diversos dispositivos con media queries;
* Adaptar la estructura y el contenido para diseños/layouts diferentes;
* Progredir de un diseño mobile hasta un diseño para desktop.

## Anexos

* <https://www.figma.com/file/Cv9OIfwW20qbM2ywcSXOnK/Apeperia-Mobile-First-(inicial)?node-id=0%3A1>
* Iphone 6.7/8 es el tamaño de un celular estándar
* ¡Opción correcta! 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). EM proviene de “ephemeral” que significa “variable” y es una variable de la propiedad font-size de la etiqueta madre. REM es una unidad relativa a la propiedad font-size de la etiqueta, por lo tanto si la etiqueta tiene 16px de font-size 1 REM equivale a 16px. 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.
* Opción correcta porque vw está específicamente relacionada a la largura de la pantalla (en pixel)
* Es importantes pensar en los márgenes y el tamaño de los botones en tamaño celular, porque las dimensiones pueden afectar a la imprecisión de touch pad y también la conexión a internet.
* <https://pagespeed.web.dev/?utm_source=psi&utm_medium=redirect>
* Inserta un enlace que la herramienta analizan las versiones mobile y desktop, con cálculos de velocidad y performance, sugestiones de lo que pueden acelerar la carga.
* <https://www.aluracursos.com/blog/guia-de-unidades-en-css>
* <https://css-tricks.com/snippets/css/media-queries-for-standard-devices/>

## Código

* box-sizing: border-box;
* :root
* --variable: valor;
* font-weight:
* text-transform: uppercase;
* flex-direction: column;
* align-items: center;
* figure
* figC
* line-height: normal;
* border-color: var(--boton-rojo);
* max-width: 600px;
* box-sizing: border-box;
* justify-content: space-evenly;
* line-height: normal;
* background-image: url(../img/icono-reloj.png);
* background-repeat: no-repeat;
* background-position: top 1.25rem left 2.5rem;
* <address
* <article
* <video
* Controls
* print and
* background-size: cover;
* flex-direction: row;
* justify-content: space-between;
* align-items: flex-end;
* justify-content: space-between;
* text-align: left;
* align-self: flex-end;
* flex-direction: row;
* flex-wrap: wrap;
* justify-content: space-between;

## Vocabulario

* Movile-first: concentra lo esencial, info. De fácil acceso, máximo rendimiento, reduce código fuente y usa HTML5.