

**TALENTO  
DIGITAL**  
INTELIGENCIA  
HUMANA

# Talento Digital para Chile:

## Módulo 3 Fundamentos de Desarrollo Web

UN PROYECTO DE:

DESARROLLADO POR:



## MÓDULO 3 – FUNDAMENTOS DE DESARROLLO WEB

### 3.2.- Contenido 2: Construcción de páginas Web responsivas

---

#### Objetivo de la jornada

---

- Conocer y comprender el concepto de responsividad
  - Entender los conceptos Mobile First y Media Query
  - Validar responsividad en distintos dispositivos
- 

#### 3.2.1.- Responsividad

##### 3.2.1.1.- ¿Qué es la responsividad?

La llegada del iPhone en el año 2007 cambió la forma en que las personas se conectaban a Internet. Hasta ese momento, los dispositivos móviles tenían pequeñas pantallas con pocas opciones de interacción; en resumen, si se deseaba acceder a una página web, debía ser desde un computador personal. En ese entonces las resoluciones más populares consideraban al menos 1024 píxeles de ancho en una pantalla, suficiente para un computador de estas características. Lo anterior, por cierto, fue un problema con los primeros dispositivos inteligentes, ya que los sitios se tenían que desplegar en pequeñas ventanas sin posibilidad a adaptar su contenido.

Un diseño web adaptable o “responsivo” es usar HTML y CSS para redimensionar, ocultar, reducir o ampliar automáticamente un sitio web, a fin que se vea bien en todos los dispositivos (computadores de escritorio, tabletas y teléfono). No olvidar: una página web debe verse bien en cualquier dispositivo.

Para hacer una página de diseño adaptable, se recomienda considerar los siguientes puntos.

#### Configuración de la ventana gráfica

Todos los sitios web con diseño adaptable deben considerar la etiqueta **<meta>** en la cabecera respectiva. Esta etiqueta permitirá definir la ventana gráfica del sitio, dando instrucciones al navegador sobre cómo manipular las dimensiones manteniendo una relación de aspecto.

```
<meta name="viewport" content="width=device-width,
initial-scale=1.0">
```

*Ilustración 1: Etiqueta para la definición de ventana gráfica*

## Imágenes responsivas

Son imágenes que se escalan automáticamente para adaptarse a cualquier tamaño del navegador. Si la propiedad de ancho de CSS se establece en 100%, la imagen se adaptará y escalará en ambas direcciones. Considere que si la imagen es más pequeña que el máximo de la ventana, se agrandará y no se verá correctamente. Como respuesta, se puede usar la propiedad **max-width**; si se establece en 100%, la imagen se adaptará a la ventana, pero no más de su tamaño original. A continuación se indican ambas maneras de modificar una imagen.

```



```

*Ilustración 2: Definición de imágenes con tamaño adaptable*

## Mostrar diferentes imágenes según ancho de ventana

La etiqueta HTML **<picture>** permite definir distintas imágenes para diversos tamaños de ventana.

```
<picture>
  <source srcset="img_smallflower.jpg" media="(max-width: 600px)">
  <source srcset="img_flowers.jpg" media="(max-width: 1500px)">
  <source srcset="flowers.jpg">
  
</picture>
```

*Ilustración 3: Mostrar distintas imágenes según ancho de ventana*

## Texto con tamaño adaptable

El tamaño del texto puede ser establecido con una unidad “vw”, que significa “tamaño de la vista” (“**viewport width**”). En general, un 1 vw = 1% del ancho del **viewport**.

```
<h1 style="font-size:10vw">Hello World</h1>
```

*Ilustración 4: Texto con tamaño adaptable*

### 3.2.1.2.- Utilización de Media Query

Adicional a cambiar el tamaño de texto y de las imágenes, es común usar el concepto de “**media queries**”. Con este tipo de consultas se pueden definir estilos para distintos tamaños de navegador.

En el ejemplo que se indica a continuación, se crean tres contenedores de distintos colores. Si el ancho de la pantalla es menor a 800px, los contenedores tendrán un nuevo ancho de 100%, y se ubicarán uno sobre otro; en caso contrario se mostrarán de forma horizontal.

```
<style>
.left, .right {
  float: left;
  width: 20%; /* El ancho de 20%, por defecto */
}

.main {
  float: left;
  width: 60%; /* El ancho es 60%, por defecto */
}

/* Use un "media query" para agregar un punto de quiebre
a los 800px de ancho de la ventana: */
@media screen and (max-width: 800px) {
  .left, .main, .right {
    width: 100%; /* El ancho es 100%, cuando el
contenedor es de 800px o menor. */
  }
}
</style>
```

*Ilustración 5: Uso de media query en diseño adaptable*



En el ejemplo anterior se crean tres clases, una para cada celda que se desplegará. En un computador personal, se verían tres elementos situados uno al lado del otro, usando 20% del ancho de la ventana los pequeños, y 60% el más grande. En una Tablet y en otros dispositivos móviles y en caso que la ventana contenedora mida menos de 800px, las etiquetas **<div>** se posicionarán de forma vertical, una sobre otra, usando el ancho actual del contenedor.

### 3.2.1.3.- Concepto Mobile First

El concepto **“Mobile First”** significa hacer diseños de sitios web pensando en dispositivos móviles antes de crear diseños que apunten a equipos de escritorio o en cualquier otro; esto hará que la página se muestre más rápido en dispositivos más pequeños. Lo anterior, por cierto, implica hacer cambios a nivel de CSS.

Por ejemplo, en vez de cambiar estilos cuando la ventana sea menor que 800px, se debería cambiar el diseño para cuando el ancho sea mayor a 800px. Las instrucciones en CSS quedarían de la siguiente manera:

```
/* Para teléfonos móviles: */
[class*="col-"] {
    width: 100%;
}

@media only screen and (min-width: 800px) {
    /* Para escritorio: */
    .col-1 {width: 8.33%;}
    .col-2 {width: 16.66%;}
    .col-3 {width: 25%;}
    .col-4 {width: 33.33%;}
    .col-5 {width: 41.66%;}
    .col-6 {width: 50%;}
    .col-7 {width: 58.33%;}
    .col-8 {width: 66.66%;}
    .col-9 {width: 75%;}
    .col-10 {width: 83.33%;}
    .col-11 {width: 91.66%;}
    .col-12 {width: 100%;}
}
```

*Ilustración 6: Usando el concepto Mobile First*



*Ilustración 7: Esquema del sitio al que se aplica el estilo de la ilustración 6*

En la ilustración anterior se muestra un ejemplo de cómo se vería el sitio con las configuraciones del CSS indicadas en un computador personal. Si se accede al sitio desde un Tablet o desde un teléfono móvil, se notará que el diseño se adaptará al tamaño de la ventana original.

### 3.2.1.4.- Tipos de dispositivos y orientaciones

En CSS se pueden declarar puntos de interrupción para definir estilos. Estos son estructuras que establecerán comportamientos dado un ancho máximo de ventana. Una idea bastante común, es crear puntos de interrupción para diseño orientado a tablets y orientado a celulares.



*Ilustración 8: Diseño de sitio web enfocado en tablet y dispositivo móvil*

En el ejemplo a continuación, note que se indican dos clases que son casi idénticas (col- y col-s-).

```
/* Para teléfonos móviles: */
[class*="col-"] {
  width: 100%;
}

@media only screen and (min-width: 600px) {
  /* Para tabletas: */
  .col-s-1 {width: 8.33%;}
  .col-s-2 {width: 16.66%;}
  .col-s-3 {width: 25%;}
  .col-s-4 {width: 33.33%;}
  .col-s-5 {width: 41.66%;}
  .col-s-6 {width: 50%;}
  .col-s-7 {width: 58.33%;}
  .col-s-8 {width: 66.66%;}
  .col-s-9 {width: 75%;}
  .col-s-10 {width: 83.33%;}
  .col-s-11 {width: 91.66%;}
  .col-s-12 {width: 100%;}
}

@media only screen and (min-width: 768px) {
  /* Para equipos de escritorio: */
  .col-1 {width: 8.33%;}
  .col-2 {width: 16.66%;}
  .col-3 {width: 25%;}
  .col-4 {width: 33.33%;}
  .col-5 {width: 41.66%;}
  .col-6 {width: 50%;}
  .col-7 {width: 58.33%;}
  .col-8 {width: 66.66%;}
  .col-9 {width: 75%;}
  .col-10 {width: 83.33%;}
  .col-11 {width: 91.66%;}
  .col-12 {width: 100%;}
}
```

*Ilustración 9: Definiendo estilos según tipo de dispositivo*

Si se desea referenciar los estilos anteriores desde HTML, se debe indicar en el atributo “**class**” la clase correspondiente definida.

```
<div class="row">
  <div class="col-3 col-s-3">...</div>
  <div class="col-6 col-s-9">...</div>
  <div class="col-3 col-s-12">...</div>
</div>
```

*Ilustración 10: Etiquetas HTML haciendo referencia a estilos de ilustración 9*

En la ilustración anterior se indica un código HTML en el que se llama a estilos que definirán el comportamiento de acuerdo al tamaño de la ventana. Frente a esto se deben definir dos casos:

- Para escritorio: La primera y tercera sección tendrán tres columnas cada una, mientras que la central tendrá 6
- Para tabletas: La primera sección considerará 3 columnas, la segunda abarcará 9, y la tercera se mostrará bajo las dos primeras secciones y considerará 12 columnas.

### Puntos de interrupción más comunes

Existen diversidad de pantallas y dispositivos, de diferentes anchos y altos, lo cual hace que sea difícil definir puntos de interrupción para cada uno de ellos. A fin de simplificar escenarios, se ha dividido en cinco grupos.

```
/* Dispositivos pequeños (teléfonos, 600px y menos) */
@media only screen and (max-width: 600px) {...}

/* Dispositivos pequeños (tabletas verticales y teléfonos
largos, 600px y más) */
@media only screen and (min-width: 600px) {...}

/* Dispositivos medios (tablets horizontales, 768px y más) */
@media only screen and (min-width: 768px) {...}

/* Dispositivos largos (laptops/escritorio, 992px y más) */
@media only screen and (min-width: 992px) {...}

/* Dispositivos extra largos (grandes laptops and equipos
escritorio, 1200px y arriba) */
@media only screen and (min-width: 1200px) {...}
```

*Ilustración 11: Combinación atributo-valor en CSS*



### Orientación: vertical / horizontal

Las “media queries” pueden ser usadas también para cambiar el diseño de una página dependiendo de la orientación del navegador. Se pueden considerar un conjunto de propiedades CSS que solo se aplicarán cuando la ventana del navegador sea más ancha que su altura o, dicho de otro, en orientación horizontal.

En el ejemplo siguiente, la página web tendrá un fondo azul claro si la orientación está en modo horizontal.

```
@media only screen and (orientation: landscape) {  
  body {  
    background-color: lightblue;  
  }  
}
```

*Ilustración 12: Cambiar color de fondo en ubicación horizontal de pantalla*

#### 3.2.1.5.- Herramientas para testear diseños

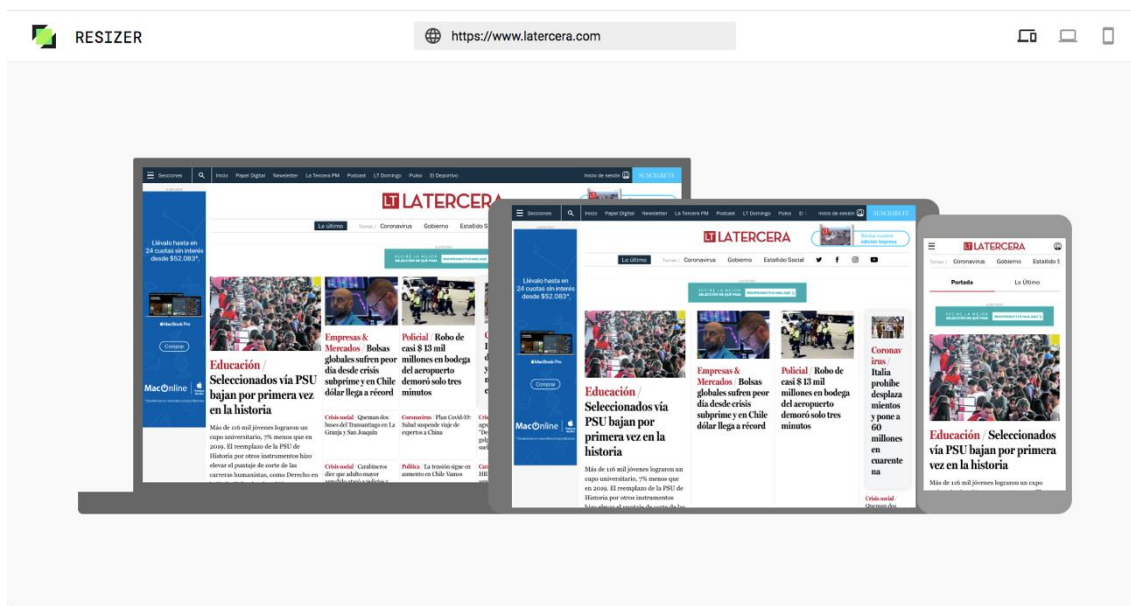
Hasta ahora se han analizado las bondades que provee el diseño adaptativo, y la forma en la que se debe implementar en un sitio. Una vez que se implementa un diseño de este tipo, se debe verificar que se adapte automáticamente a cualquier tamaño de dispositivo posible, en especial de aquellos que son más comunes en su uso.

En las herramientas que existen en la red, se destacan las siguientes:

##### Resizer

Este es un sitio web que permite validar el diseño de un sitio creado bajo la lógica responsiva. Para poder hacer uso de ella, el sitio a validar debe estar en la red, y se debe acceder a él ingresando su dirección en la barra de navegación, y presionar “Enter”. Con esto se mostrará el sitio en los tres recuadros que aparecen más abajo. Cada recuadro es un tipo de dispositivo genérico.

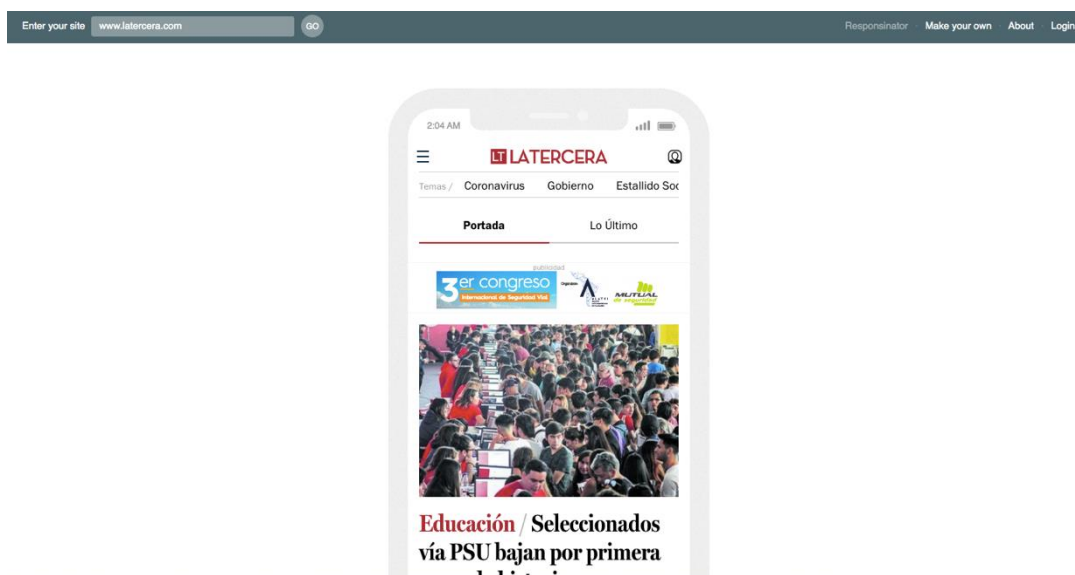
Dirección: <https://material.io/resources/resizer/>



## Responsinator

Simula la representación de la web en un total de seis dispositivos móviles diferentes, en todos los casos considerando opción vertical y horizontal. Sigue la misma idea de la página indicada en el párrafo anterior.

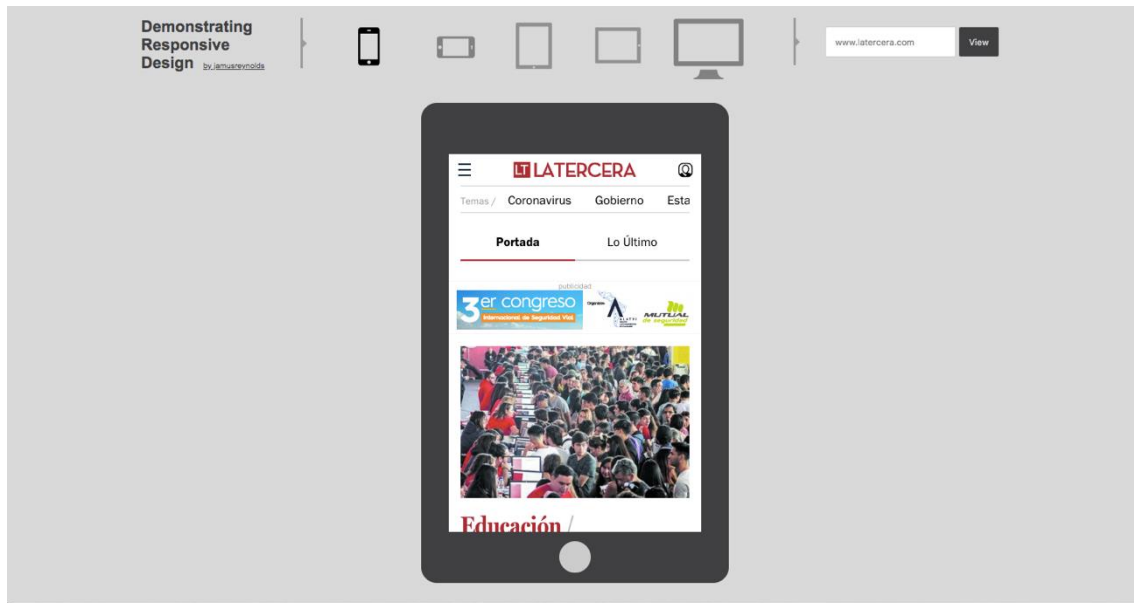
Dirección: <http://www.responsinator.com/>



## Demonstrating Responsive Design

En esta plataforma el usuario tiene solo a su disposición tres tamaños de pantalla. Independiente de esto, esta herramienta facilita el cambio rápido y claro entre diversos formatos.

Dirección: <http://www.jamus.co.uk/demos/rwd-demonstrations/>



## Anexo 1: Referencias

**[1] w3schools - Diseño Web HTML adaptable**

Referencia: [https://www.w3schools.com/html/html\\_responsive.asp](https://www.w3schools.com/html/html_responsive.asp)

**[2] J.D Gauchat “El Gran libro de HTML5, CCS3 y JavaScript”  
3ª edición**

**[3] Responsive Test**

Referencia: <https://www.ionos.es/digitalguide/paginas-web/disenio-web/herramientas-para-realizar-un-responsive-test/>