

{desafío}
latam_

Diseño responsivo _

Parte I



Conociendo el proyecto

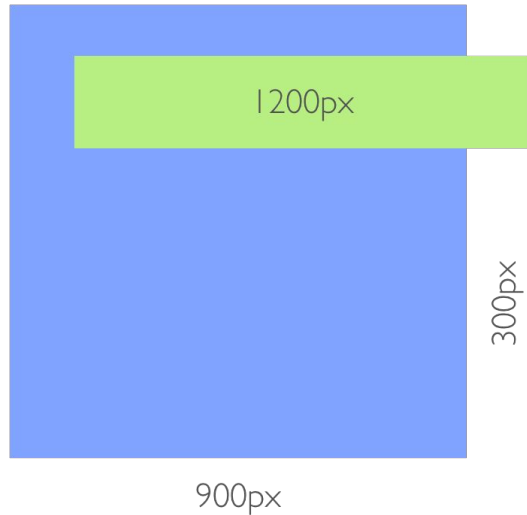
Transformar maqueta a diseño responsivo



- Luego de finalizar, nuestro product owner nos comenta que el cliente quiere que la maqueta sea responsiva.
- Para eso tenemos que crear el diseño para dispositivos móviles y luego presentar al cliente el resultado.
- Al aceptar debemos crear la interfaz responsiva.

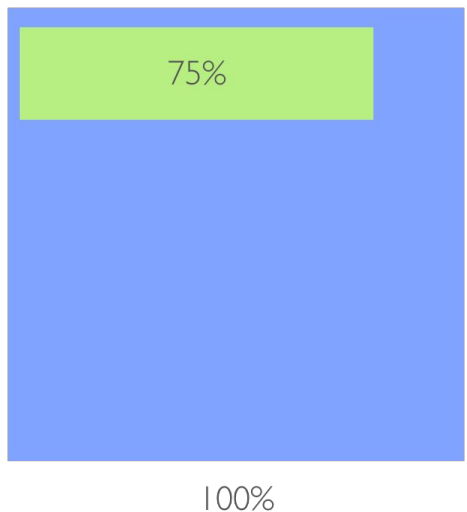
Conociendo los diseños responsivos

Tipos de diseño: Estático



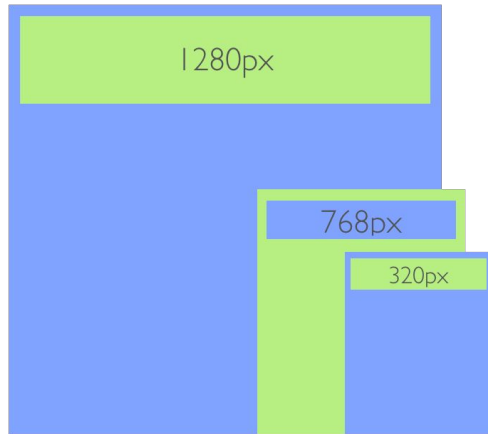
- Tienen un ancho especificado por unidades de medida absolutas.
- Este tipo de diseño por lo general se utiliza pensando en un sólo dispositivo.
- Si el elemento es más grande que el ancho de la pantalla aparecerá un scroll horizontal.

Tipos de diseño: Fluido



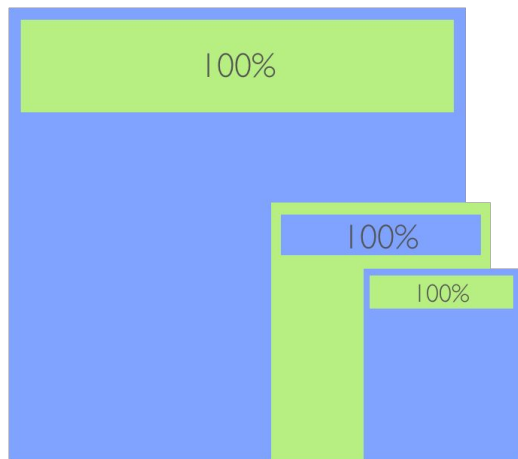
- Este tipo de diseño tiene un ancho relativo al *viewport* (**ventana del navegador**).
- Para crear este tipo de layout se requiere el uso de unidades relativas como %, rem, ems, vw, etc.

Tipos de diseño: Adaptativos



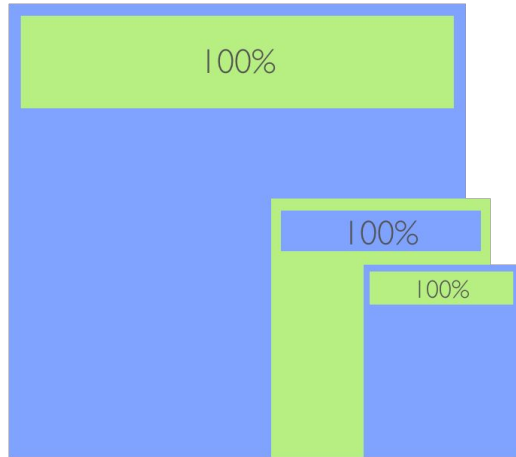
- Los diseños adaptativos se caracterizan por definir el diseño en diferentes tamaños.
- Asimismo los elementos contenidos en este tipo de diseño usan unidades de medida absolutas para mayor control en el flujo del contenido.

Tipos de diseño: Responsivos



- Estos tipos de diseño se definen por tres elementos principales constituidos por grillas, media queries e imágenes flexibles o fluidas.
- Las ventajas de los diseños responsivos se relacionan a su fácil implementación y popularidad.

Diseños responsivos v/s adaptativos



- Un diseño responsivo es más económico que uno adaptativo.
- Los diseños responsivos tienen una experiencia de usuario suficiente.
- Los diseños responsivos son populares.

Definir un diseño responsivo

Estrategias para diseñar página responsiva

Las estrategias que usaremos para resolver este problema serán tres:

- Definir el flujo del contenido usando patrones de diseño responsivo.
- Seleccionar el contenido en función a la importancia.
- Diagramar resultado en una representación visual de baja fidelidad.

Patrones de diseño responsivo

Patrones responsivos: *Mostly Fluid*



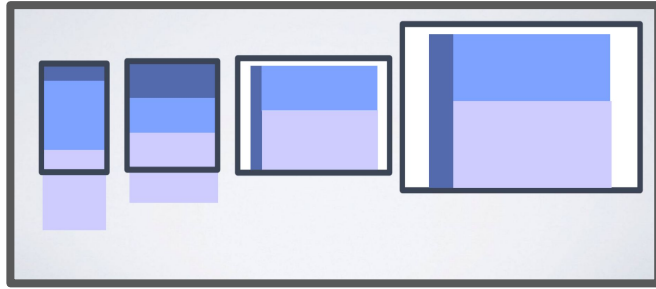
- El patrón que fluye la mayor parte del tiempo.
- En dispositivos pequeños los elementos fluyen poniéndose uno por sobre otro de manera vertical.
- En dispositivos medianos los elementos se encuentran contenidos en columnas.
- En dispositivos grandes se mantienen las columnas, pero se agrega un margen que fija los elementos.

Patrones responsivos: *Column drop*



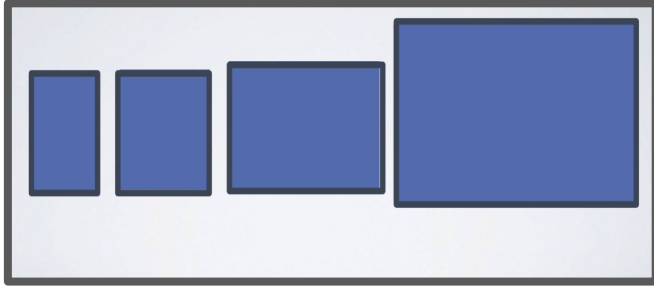
- Se conforman de varios contenedores que cubren el ancho total del viewport.
- A medida que van disminuyendo su tamaño las columnas comienzan a apilarse hasta que todos los contenedores se juntan de manera vertical.

Patrones responsivos: *Layout Shifter*



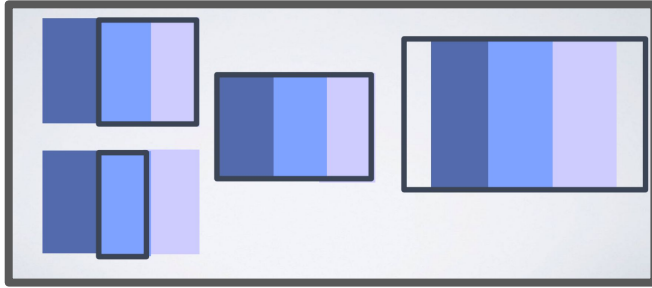
- Son aquellos que poseen varios puntos de quiebre en los cuales se define una posición específica del diseño.
- Estos patrones se parecen a los adaptativos obviando la unidad de medida.
- En tamaños grandes se fija un margen usado para no desbordar el diseño.

Patrones responsivos: *Tiny tweaks*



- Los diseños tiny tweaks son diseños que se destacan por su minimalismo.
- Contemplan cambios mínimos como los tamaños de fuentes y la distribución del relleno dentro del diseño.

Patrones responsivos: Off canvas



- Finalmente los patrones off canvas son diseños que se destacan por el uso de columnas verticales que se van escondiendo dependiendo de la importancia del contenido.

Mobile First

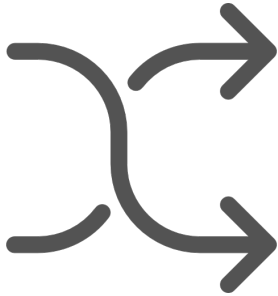
¿Qué es *Mobile First*?



La filosofía *Mobile First* nació como una alternativa para desarrollar sitios usando las capacidades que tienen los dispositivos móviles a modo de mejorar la experiencia de los usuarios en esos dispositivos.

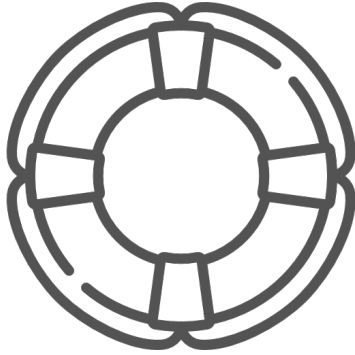
Recomendaciones para trabajar con *Mobile First*

Adecuarse a los cambios



Deberemos estar atentos a cambios en la tecnología móvil y siempre tomar la iniciativa para mejorar la experiencia de los usuarios utilizando estas nuevas funcionalidades.

Usar la tecnología a nuestro favor



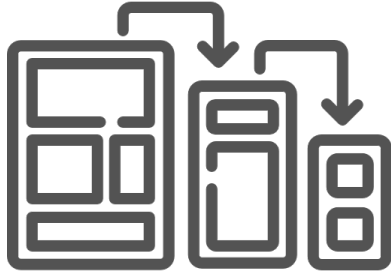
Esto significa que debemos aprovecharnos de todas las funcionalidades que nos ofrecen dispositivos y navegadores a modo de crear una mejor experiencia.

Uso de diseño responsivo



En esta filosofía el tipo de diseño que realicemos afectará directamente en la experiencia que tendrá posteriormente el usuario. En esencia este punto hace referencia a las decisiones de implementación de un tipo de diseño.

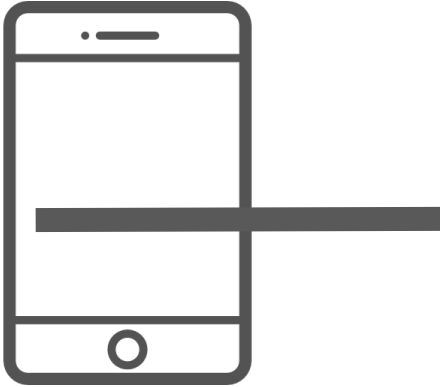
Homogeneizar la experiencia



Esto significa que debemos ser capaces de homogeneizar la experiencia en todos los dispositivos en los cuales deseemos mostrar contenido.

Componentes de un diseño responsivo: *Viewport*

Tipos de *viewport*



El *viewport visual* tiene como característica que su tamaño varía y el *viewport* de diseño no. Esto se debe a que el diseño del *viewport* es en esencia más amplio que el *viewport visual*.

¿Qué es el meta viewport?

```
<head>
  <meta name="viewport" content="directiva,directiva"
/>
</head>
```

- La etiqueta meta viewport controla y escala el diseño del viewport en diferentes dispositivos.
- Esta etiqueta, como toda etiqueta meta, se alojará dentro de **<head>**.

¿Qué directivas podremos usar?

Width

```
<head>  
  <meta name="viewport" content="width=device-width" />  
</head>
```

- Estas nos permiten especificar al navegador el ancho que deseamos usar en el sitio.
- Podremos usar un ancho específico usando unidades relativas, absolutas o definirlas de manera automática con ***width=device-width***.

Height

```
<head>  
  <meta name="viewport" content="height=device-height" />  
</head>
```

- Con height podremos definir el alto que tendrá nuestro sitio.
- Podremos definir un alto específico o definirlo de manera automática con ***height=device-height***.

User-scalable

```
<head>  
  <meta name="viewport" content="user-scalable=no" />  
</head>
```

- Nos permite decirle al navegador que dé o no la posibilidad al usuario de usar zoom en nuestra página.
- Como valores permitidos se encuentra el ***user-scalable=no*** y ***user-scalable=yes***.

Initial scale

```
<head>  
  <meta name="viewport" content="initial-scale=1"/>  
</head>
```

- Declara al navegador el zoom inicial que tendrá nuestra interfaz.
- Los valores que podremos usar comienzan con **0.1** (10%) y terminan con **10.0** (1000%).
- Si queremos que la escala de nuestro sitio inicie con el 100% de zoom, deberemos agregar como valor ***initial-scale=1***.

Maximum-scale

```
<head>  
  <meta name="viewport" content="initial-scale=1,  
maximum-scale=10">  
</head>
```

- Nos permite definir el máximo de *zoom* que los usuarios podrán hacer en nuestro sitio.
- Los valores al igual que *initial-scale* comienzan con **0.1** y terminan con **10.0**.

Minimum-scale

```
<head>  
  <meta name="viewport" content="initial-scale=1,  
minimum-scale=1">  
</head>
```

- Nos permite definir el máximo de *zoom* que los usuarios podrán hacer en nuestro sitio.
- Los valores al igual que *initial-scale* comienzan con **0.1** y terminan con **10.0**.

¿Cuáles son las directivas recomendadas?

```
<head>
  <meta name="viewport"
content="content=initial-scale=1, minimum-scale=1">
</head>
```

- La última directiva nos ayudará a definir el zoom mínimo que tendrá nuestro sitio.
- Los valores al igual que initial-scale comienzan con **0.1** y terminan con **10.0**.

Transformar unidades absolutas a relativas

Fórmula de conversión

`objetivo / contexto=resultado`

Para lograr esta conversión deberemos usar un poco de matemática básica, dividiendo el valor a transformar por el contexto de la unidad a transformar.

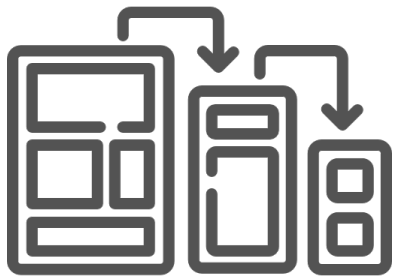
Fórmula de conversión

$$30 / 16 = 1.875$$

- Deseamos transformar el tamaño de un título de **30px** a una unidad relativa como rem.
- Para lograrlo deberíamos dividir los **30px** , por **16px**, de esa manera obtendremos un resultado de **1.875rem**.

Componentes de un diseño responsivo: Media queries

¿Qué son los *media queries*?



Los media queries son una directiva de CSS que nos permitirá agregar estilos en ciertos tamaños sólo si la condición es cierta.

Estructura de un media query

La estructura básica de un media query está constituida por:

```
@media (expresion)
{
    reglas CSS
}
```

- La directiva **@media**.
- El **bloque de estilos**.
- Pero además tendremos la opción de agregar algunos parámetros para hacer más especializado a nuestro *media query*.

Parámetros de *@media*

Tipos de medio

```
@media screen {  
  h1 {  
    font-size: 30px  
  }  
}  
  
@media print {  
  h1 {  
    font-size: 50px  
  }  
}
```

- Podremos especificar el tipo medio al cual afectarán nuestros estilos.
- Los tipos de medios más comunes que podremos encontrar son all, screen y print.

Expresiones

```
@media screen
(width: 350px) {
  h1 {
    font-size: 20px
  }
}
```

- las expresiones nos darán la posibilidad de usar alguna característica del dispositivo a modo de activar o desactivar media query.

Reglas lógicas

```
@media only screen
and (min-width:
300px) {
p {
    color:
blue;
}
```

- Las reglas lógicas nos permitirán agregar características a los media queries.
- Con **and** podremos agregar más de dos tipos de medio o expresiones y con **only**, podremos prevenir que los navegadores antiguos no soporten media queries.

{desafío}
latam_

*Academia de
talentos digitales*

www.desafiolatam.com