

# [Resumen] Desarrollo de la interfaz de PULS3

## ¿Qué es una llamada al servidor?

Siempre que hablamos de solicitudes a servers y la demora que tienen estas añadiendo el tiempo que puede costar la carga de la pagina tienden a tener confusiones con el tema y mucho mas con el peso de los archivos así que hoy vengo a aclarar sobre que es una solicitud a un servidor y por que empresas como amazon dicen que cuando su pagina demora 1s en cargar demás pueden tener perdidas de dinero.

Teniendo en cuenta esto podemos afirmar que cuando creamos una pagina web solemos llamar un archivo CSS, muchas imágenes jQuery y un archivo JavaScript, como mínimo, y cada una de esas llamadas es una solicitud a un servidor que se demorara lo que demora el servidor en responder.

Pero ¿cómo se yo cuanto demora un servidor en responder y como saber si mi página demora mucho en cargar por esta misma cuestión? recordemos que una llamada a un servidor es totalmente independiente del archivo a descargar.

Primero debes abrir una consola o sistema de comandos en donde digitas el siguiente comando:

```
C:\Users\USUARIO>ping google.com

Haciendo ping a google.com [74.125.229.229] con 32 bytes de datos:
Respuesta desde 74.125.229.229: bytes=32 tiempo=81ms TTL=56
Respuesta desde 74.125.229.229: bytes=32 tiempo=98ms TTL=56
Respuesta desde 74.125.229.229: bytes=32 tiempo=81ms TTL=56
Respuesta desde 74.125.229.229: bytes=32 tiempo=81ms TTL=56

Estadísticas de ping para 74.125.229.229:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
        (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
```

```
Mínimo = 81ms, Máximo = 98ms, Media = 85ms
```

```
C:\Users\USUARIO>
```

Si vemos nos da 4 respuesta a las cuales no le pondremos atención ya que la que nos importa es el tiempo medio de respuesta que en este caso google tiene un tiempo medio de 85ms, es decir si hago una llamada a google.com se me demorara 85ms en responder, ¿te imaginas 10 llamadas a google? 850ms de espera para que cargue es demasiado.

### ¿Qué soluciones tenemos?

La mejor solución es usar opciones como llamar solo un archivo CSS, solo una imagen en cuanto a diseño a parte de las imágenes de contenido y llamar a lo menos 2 JavaScript.

¿Pero como podemos hacer esto?

Minificando archivos, es decir quitando espacios y enter a los archivos y dejando todo el código en uno solo.

Con JavaScript podemos tener un solo archivo con todas las dependencias y otro con el código propio de JS

Con tus imágenes, realizas una técnica llamada Sprites en donde pones todas las imágenes del diseño de una pagina en una sola y a partir de CSS las acomodas para la cuestión que sea necesaria.

¿Y con CSS?

Es muy feo tener un solo CSS si usamos normalize, fuentes, responsive y estilos, pues es aquí en donde viene la magia de Stylus.

## La Magia de Stylus

Con Stylus podemos pasar archivos CSS a styl con un simple comando:

```
> stylus -css ubicacion/estilos.css
```

De esta forma todos nuestros archivos que estén en .css los pasaremos a .styl, luego los importamos en nuestro styl madre que sera el que estemos procesando a css y en donde quedaran todos los css comprimidos y minificados.

Para importar archivos styl a stylus usamos:

```
@import archivo.styl
```

Ahora comprimimos y la magia esta hecha:

```
> stylus -c -w ubicacion/style.styl
```

Y nos devolverá un style.css con el contenido de style y el contenido de estilos en un mismo archivo que sera el que utilizamos al momento de hacer la petición del server.

Otra mágica función de Stylus es la de crear variables, operaciones matemáticas, mixins y funciones de la siguiente forma:

```
/* Esto es una variable */
pepito = 5px

/* Esto es un Mixin */
border-radius(n)
  -webkit-border-radius n
  -moz-border-radius n
  border-radius n
#cajita
  background black
  border-radius (pepito)
  /* & significa si mismo */
  &:hover
    background white
```

Entre otras funcionalidades que puedes encontrar [aquí](#).

## Responsive Design

Acá a partir de nuestro proyecto colocamos media queries las cuales evalúan el tamaño del navegador y a partir de este se aplican estilos para cada pantalla personalizando mas la característica de diseño fluido que ya habíamos aplicado.

En este caso Freddy coloca 2 medidas, la primera es un mínimo de 768px en donde se cumple cierto código y la segunda un mínimo de 1280px en donde se cumplirá el código de 768px y el de 1280px.

```
@media screen and (min-width:768px)
  #contenido
    #nota
      float right
      font-size 1em
      height auto
      width 20%
    #nota:hover
      height auto
    .item
      display inline-block
      vertical-align top
      width 30%
      .titulo_item a
        font-size 0.9em
        font-weight normal
@media screen and (min-width:1280px)
  #contenido
    .item
      width 20%
```

Dándonos cuenta que el Responsive es algo muy sencillo pero que llego para darnos una vida mucho mas agradable, recuerda visitar las lecturas en donde evaluamos mas a profundidad la filosofia responsive y el diseño retina para ser experto en el tema, ponte tareas y lleva a PULS3 a un nivel mucho mas pro como ya lo han hecho unos de tus compañeros.