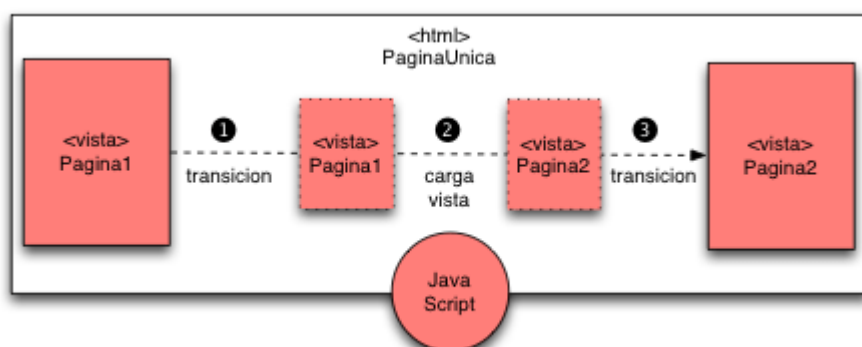
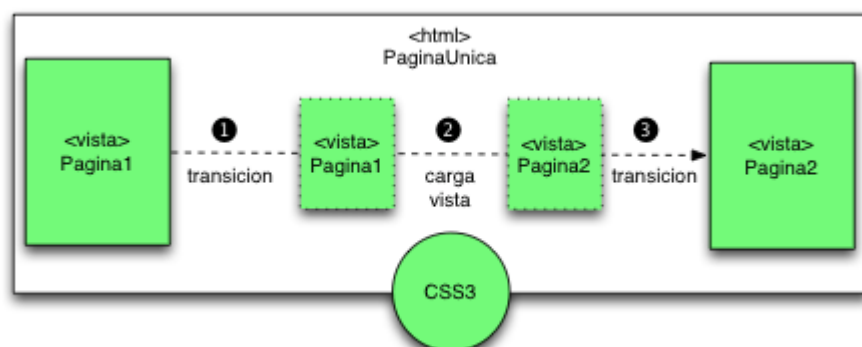


En muchas ocasiones los desarrolladores estamos acostumbrados a usar JQuery para realizar transiciones . Sin embargo es buen momento para recordar que el rendimiento en las aplicaciones móviles es muy importante y mucho mas el rendimiento de las transiciones



CSS3

Por ello en vez de apoyarnos en el motor de Javascript para realizar las transiciones casi siempre será mas útil apoyarnos en una primera instancia en las nuevas capacidades que CSS3 soporta para realizar este tipo de operativa.



Animaciones CSS3

Vamos a ver a continuación el código de la aplicación SPA apoyandose en animaciones de

CSS3. En un primer lugar definiremos una animación de CSS3 utilizando keyFrames . Es algo muy sencillo ya que en este caso simplemente disponemos de dos estados opacity:0 y opacity:1 con los cuales generamos la animación que denominamos “fadeIn” . En nuestro caso durará un segundo como se muestra en el siguiente bloque de código.

```
<style type="text/css">
@-webkit-keyframes fadeIn {
  from { opacity: 0; }
  to { opacity: 1; }
}

@keyframes fadeIn {
  from { opacity: 0; }
  to { opacity: 1; }
}

.ocultar {
display:none;

}

.mostrar {
display:block;
-webkit-animation:fadeIn 1s;

}

</style>
```

Una vez hecho esto modificamos las funciones de JavaScript que teníamos construidas para que realicen la animación. En un primer lugar añadiremos al document.ready el siguiente código que se encarga como siempre de solo mostrarnos la primera vista.

```
$("#div").addClass("ocultar");  
$("#div:first").addClass("mostrar");
```

Hecho esto nos valdrá con modificar la función que cambiaba de vista para que se apoye en CSS3.

```
function cambiarPagina(paginaOrigen,paginaDestino) {  
    $(paginaOrigen).removeClass("mostrar");  
    $(paginaDestino).addClass("mostrar");  
  
}
```

Una vez hecho esto hemos conseguido que nuestra aplicación SPA se apoye en animaciones CSS3 y el rendimiento mejore.