**Guía para el desarrollo de pantallas html5 en FW6**

Para desarrollar una pantalla html5 utilizando las funciones definidas en la plataforma FW6 es necesario tener en cuenta una serie de aspectos básicos, sin los cuales la pantalla no se integra correctamente.

La pantalla se divide en tres archivos fundamentales, sin tener en cuenta los datos en formato JSON:

* Vista parcial HTML5 (ver ejemplo: fw6/views/p0.html)
* Hoja de estilos (CSS) (ver ejemplo: fw6/css/p0.css)
* Archivo Java Script (ver ejemplo: fw6/js/p0.js)

Para crear estos tres archivos se debe seguir la nomenclatura de acuerdo al identificador del tipo de la pantalla. En este caso como se trata de la pantalla tipo **0**, los archivos se nombran **p0.html**, **p0.css** y **p0.js**, siempre con la letra **p** seguida del identificador de tipo. *(Por ejemplo: en caso que se trate de la pantalla de tipo 30, sería p30.html, p30.css y p30.js)*

La ubicación de los archivos debe ser la siguiente:

* **views/p0.html**
* **css/p0.css**
* **js/p0.js**

Para estandarizar el desarrollo de pantallas, es necesario definir varias pautas, que se deben tener en cuenta a la hora de escribir el contenido de estos archivos. A continuación se describen los aspectos principales para cada uno de ellos.

**Vista parcial HTML5**

Dentro de **p0.html** se coloca el siguiente código, que no puede faltar:

**<script src="js/p0.js"></script>   
<div class="screen-container">**

**<!--Aquí va el código de los elementos de la pantalla (ver p0.html) -->**

**</div>**

Se incluye el archivo **p0.js** correspondiente a la pantalla y luego un **div** con **class="screen-container”** para encapsular los elementos de la pantalla. Si se desea incluir algún script debe incluirse antes de **p0.js**

**Hoja de estilos (CSS)**

Todos los estilos definidos en la pantalla deben estar enmarcados en el contenedor **.screen-container** *(definido previamente en p0.html)*, excepto los que pertenezcan a algún componente y necesiten ser redefinidos, como es el caso de los componentes de JQuery UI. Por ejemplo en el caso de **p0.css** los estilos definidos son los siguientes:

.screen-container {float:left; width: 100%;margin-top:30px;}  
.screen-container #title {float: left;width:100%;font-weight: bold;display: none;margin: 5px 0;font-size: 18px;}  
.screen-container #a1 {float:left; width: 100%;display: none;}  
.screen-container #a2 {float:left; width: 100%;display: none;}  
.screen-container a.message {float:left; display: none;}

Nótese que todos comienzan por **.screen-container**, para que no interfieran con ningún otro estilo de elementos que puedan estar definidos previamente por la plataforma y cuyos identificadores o clases sean iguales a los definidos en la pantalla.

**Archivo Java Script**

El archivo Java Script que encapsula el comportamiento de la pantalla, es la parte más importante, debido a que contiene varios elementos que, en caso que sean empleados incorrectamente o no se utilicen, pueden ocasionar un comportamiento inesperado en el flujo de ejecución de la pantalla. Para que todo funcione según lo previsto en la plataforma, es necesario tener en cuenta los siguientes aspectos:

1. La pantalla debe implementarse usando la guía descrita en **p0.js**
2. No puede faltar la definición de la función **shift.screen.play** al inicio del archivo, de la siguiente forma:  
   shift.screen.play = function() {  
    $.shiftScreen.initialize($("#content"));   
   };
3. La definición del objeto que encapsula el comportamiento de la pantalla debe ser una extensión de JQuery, tal y como se muestra en el archivo de ejemplo **p0.js**. Es necesario que se mantenga la estructura predefinida en **p0.js**, para poder lograr que cada programador entienda el código escrito por los demás y resulte fácil hacer modificaciones en las pantallas. Este objeto siempre debe llamarse **shiftScreen** tal y como está definido en **p0.js** y tiene que tener definidas las siguientes variables y funciones:  
   var \_self = this; //se utiliza para exportar o hacer públicas algunas funciones del objeto **shiftScreen**.var \_container = null; //se utiliza para contextualizar los elementos de la pantalla en un contenedor único.  
     
   var \_screenPath = $.extend(true, {}, shift.screen.path); //clona la ubicación de la pantalla actual para almacenar el índice de unidad, tópico y pantalla.

\_self.initialize = function (container) { //inicializa la pantalla llamando por orden a otras funciones   
 \_container = container;  
 \_loadScreenData();  
 \_startAnimation();  
 \_attachBehavior();  
 };  
var \_loadScreenData = function () { }; //carga los datos de la pantalla asignando textos y otros contenido iniciales para el funcionamientos de la pantalla.  
var \_attachBehavior= function () { }; //se encarga de asignar comportamientos a los eventos de los elementos de la pantalla que así lo requieran, por ejemplo click de botones, keypress, resize, etc.  
var \_resizeControls= function () { }; //debe incluir todos los ajustes de elementos que se deben ejecutar en el evento **resize** del objeto **window**.   
var \_startAnimation= function () { }; //debe incluir todo el comportamiento de la animación inicial de los elementos de la pantalla, puede desencadenar reproducción de audio y animaciones Edge.

1. No puede faltar la llamada a la función **shift.screen.loaded(\_screenPath)**, que debe realizarse siempreal terminar la ejecución de las animaciones iniciales de la pantalla. Esta función, es la encargada de habilitar los botones de la barra de navegación y si no se emplea correctamente pueden habilitarse antes de tiempo o no habilitarse en ningún momento e impedir la navegación hacia las demás pantallas. El parámetro **\_screenPath** es requerido y siempre está definido de acuerdo al lo descrito anteriormente en el punto número 3.
2. No puede faltar la llamada a la función **shift.screen.end(\_screenPath)** al terminar la ejecución de la pantalla. Esta función, es la encargada terminar la pantalla y poner a parpadear el botón de la barra de navegación, para avanzar a la pantalla siguiente. El parámetro **\_screenPath** es requerido y siempre está definido de acuerdo al lo descrito anteriormente en el punto número 3.
3. No puede faltar la llamada a la función **shift.screen.setScore(\_screenPath, score)** en las pantallas de evaluación o test, después de haber finalizado el examen. Esta función, habilita el botón de la barra de navegación, para avanzar a la pantalla siguiente. El parámetro **\_screenPath** es requerido y siempre está definido de acuerdo al lo descrito anteriormente en el punto número 3 y el parámetro **score** es igualmente requerido y debe ser un número entre 0 y 100.
4. No puede faltar la llamada a la función que inicia la reproducción de la pantalla, siempre al final del archivo después de la definición del objeto **shiftScreen**  
   $(function(){   
    shift.screen.play();   
   });
5. Todas búsquedas de audios, videos, imágenes, animaciones, textos y parámetros deben hacerse utilizando las funciones definidas previamente en la plataforma.  
   shift.screen.getAudio(id)  
   shift.screen.getVideo(id)  
   shift.screen.getText(id)  
   shift.screen.getImage(id)  
   shift.screen.getAnimation(id)  
   shift.screen.getParam(id)  
   shift.screen.getParamBool(id)  
   shift.screen.getParamString(id)  
   shift.screen.getParamNumber(id)  
   Se debe tener en cuenta que los identificadores de los elementos que vienen en el JSON son siempre conocidos. Por lo que no es necesario iterar por la colección de elementos en busca de alguno, pues ya las funciones anteriores hacen ese trabajo.  
   Para el caso de las animaciones de Adobe Edge, existe una guía detallada con los pasos para utilizarlas correctamente.
6. Para reproducir audios es importante hacerlo siempre utilizando la función **shift.screen.playAudio(id, callback)** y no utilizar en ningún caso la función **shift.screen.toggleAudioButton**, ya que las llamadas a esta función son totalmente responsabilidad de la plataforma y no del programador.
7. Para mostrar mensajes de alerta o textos se deben usar siempre los métodos del objeto **MessageBox** definido en **jquery.utilities.js.** Si se necesita que la ventana de diálogo tenga un estilo diferente al de JQuery UI, se pueden sobrescribir los estilos por defecto en cada pantalla en particular. Usando Firebug en Firefox resulta sencillo hacer este tipo de cambios.

**Referencias de código y de otras pantallas**

Todo el código de este ejemplo está disponible en el repositorio **FW6** en los archivos mencionados anteriormente (**p0.js**, **p0.html** y **p0.css).**

Para pantallas de múltiples layouts, se puede tomar como guía la p83 que ya está implementada usando toda la estructura descrita anteriormente y para pantallas de enlazar columnas se puede tomar como guía la p16.