

Manejo de objetos y eventos con librería JavaScript (jQueryUI)

Módulo: **Desarrollo Web en entorno cliente**

CFGs Desarrollo de Aplicaciones Web

¿Qué es jQueryUI?

- Basado en el framework jQuery añadiendo funcionalidades que mejoran la experiencia del usuario.
- jQueryUI es la distribución oficial de componentes para la creación de interfaces de usuario avanzadas.
- <http://jqueryui.com/download> Página desde la que podemos descargar jQueryUI en función de los componentes que queremos usar en nuestra web, los temas que vayamos a usar y la versión de jQueryUI.
- <http://jqueryui.com/demos/> Página donde existen ejemplos de efectos, widgets, etc.

¿Cómo usar jQueryUI?

- Una vez descargados los ficheros debemos incluirlos en nuestra página:
<link type="text/css" href="css/le-frog/jquery-ui-1.8.1.custom.css" rel="Stylesheet" />
<script type="text/javascript" src="js/jquery-1.4.2.min.js"> </script>
<script type="text/javascript" src="js/jquery-ui-1.8.1.custom.min.js"> </script>
- Incluir el JS que necesitamos para usar los componentes en la página, así como configurarlos. Por ejemplo:

```
<html lang="es">
<head>
<title>Seleccionar fecha con jQuery UI</title>
  <link type="text/css" href="css/le-frog/jquery-ui-1.8.1.custom.css" rel="Stylesheet" />
  <script type="text/javascript" src="js/jquery-1.4.2.min.js"> </script>
  <script type="text/javascript" src="js/jquery-ui-1.8.1.custom.min.js"> </script>
</script>
$(document).ready(function(){
  $("#campofecha").datepicker();
})
</script>
</head>
<body>
<form>
  Fecha: <input type="text" name="fecha" id="campofecha">
</form>
</body> </html>
```

Interacción con jQueryUI – draggable

jQuery dispone de un método llamado `draggable()` que permite aplicar a elementos de la página la funcionalidad de arrastrar. Con HTML5 ya existe esta posibilidad.

```
$("#h1").draggable(); //Todos los encabezados H1 se pueden arrastrar
```

```
$(document).ready(function(){
```

```
$("#micapa").draggable({
```

```
    cursor: 'move',      //cambiamos la punta de flecha del ratón por una cruz
```

```
    delay: 500,          //retrasamos el arrastre
```

```
    distance: 50,        //indica el espacio en píxeles a partir del cual el elemento se mueve
```

```
    grid: [200,50],      //se mueve en una rejilla de 200px de ancho por 50px de alto
```

```
    containment: 'parent' //limitamos el arrastre dentro del padre. Puede ser: document,  
                        window o coordenadas [x1,y1,x2,y2]
```

```
}); })
```

Eventos:

- **create:** Se produce cuando se aplica la funcionalidad de arrastre a un elemento.
- **start:** Que ocurre cuando comienza un arrastre.
- **drag:** Se da cuando el ratón se está moviendo durante el proceso de arrastrar.
- **stop:** Se produce cuando se suelta el elemento.

Interacción con jQueryUI – draggable

```
$("#micapa").draggable({  
  start: function(){  
    $("#log").html("Evento start");  
  }  
});  
  
//Usando el método bind  
$("#micapa").bind("drag", function(evento, ui){  
  $("#log").html("Evento drag");  
});  
$("#micapa").bind("dragstop", function(evento, ui){  
  $("#log").html("Evento stop");  
});  
$("#micapa").draggable({  
  drag: function(event, ui){  
    ui.helper.html("Estoy siendo arrastrado");  
  }  
});
```

El método stop se ha de invocar como dragstop cuando lo usamos con bind.

Cada función recibe dos parámetros: evento y elemento draggable (ui)

- ui.helper: objeto jQuery que se ha transformado en elemento draggable
- ui.position: objeto que da la posición del elemento draggable con respecto a su contenedor.
- ui.offset: objeto que da la posición del elemento draggable con respecto al comienzo de la página.

Interacción con jQueryUI – draggable

Métodos:

- destroy: elimina la funcionalidad de arrastre de un elemento
- disable/enable: aplica el estilo desactivado/activado al elemento y no se permite mover/se permite mover.
- option: permite asignar valores a las opciones del elemento draggable.
- widget: devuelve el elemento draggable.

```
$("#micapa").draggable("disable");  
$("#micapa").draggable("option", "revert", true);  
elementoDraggable = $("#micapa").draggable("widget");  
$("#micapa").draggable("destroy");
```

CSS:

```
.ui-draggable {background: red}  
.ui-draggable-dragging { background: blue}
```

Interacción con jQueryUI – droppable

jQuery dispone de un método llamado `droppable()` que permite aplicar a elementos de la página la funcionalidad de soltar. Con HTML5 ya existe esta posibilidad.

```
$("#capasoltar").droppable({  
    accept: ".tarea",  
    tolerance: "fit"  
});  
$("#capasoltar").droppable({  
    drop: function( evento, ui ) { //mismos parámetros que en el plugin draggable  
        $(this).html("Droppable!!!");  
        ui.draggable.css("background-color","yellow");  
    }  
});
```

Propiedades:

- `tolerance`: nos indica la tolerancia con la que un elemento se acepta como soltado. Por defecto el valor que tiene es "intersect" (mínimo 50% del elemento dentro del elemento que lo recibe), "fit" (100% del elemento dentro del elemento que lo recibe).
- `accept` o `scope`: se indica el rango de elementos que se aceptarían en el elemento destino por medio de un selector.

Interacción con jQueryUI – droppable

Eventos:

- create: cuando se crea el elemento droppable
- activate/desactivate: cuando un elemento draggable que se podría aceptar por el droppable empieza a arrastrarse/deja de arrastrarse
- over/out: cuando un elemento draggable que se podría aceptar por el droppable entra/sale en un área de tolerancia donde se podría soltar sobre el droppable.
- drop: cuando se suelta un elemento draggable sobre un droppable, siempre y cuando, el draggable sea aceptado por el droppable y se suelte dentro del área de tolerancia del droppable.

Métodos:

- disable/enable: deshabilita/habilita un elemento droppable
- option: permite definir opciones de configuración después de crear el droppable

```
$("#capasoltar").droppable("option", "activeClass", "clasesuelta");
```

Cuando un elemento draggable se puede aceptar por el droppable se le cambia la clase para que el usuario lo distinga claramente.

<http://jqueryui.com/demos/>

Interacción con jQuery

Otras funcionalidades relacionadas con la interacción son:

- [Resizable](#)
- [Selectable](#)
- [Sortable](#)

Siguiendo los enlaces anteriores se puede llegar a los ejemplos de cada apartado y a la API con la documentación existente (opciones, métodos y eventos).

Widget jQueryUI – Datepicker

Este widget permite al usuario seleccionar una fecha de manera visual. En HTML5 ya se ha incorporado pero no es soportado por todos los navegadores ni tiene las mismas funcionalidades.

```
$("#campofecha").datepicker({  
  firstDay: 1, //la semana comienza en lunes  
  showOn: 'both',  
  buttonImage: 'calendar.png',  
  buttonImageOnly: true,  
  changeYear: true,  
  numberOfMonths: 2,  
  onSelect: function(textoFecha, objDatepicker){  
    $("#mensaje").html("<p>Has seleccionado: " + textoFecha + "</p>");  
  });  
$("#campofecha").datepicker( "option", "changeMonth", true );
```

showOn: el calendario se muestra al hacer focus en el campo de texto o al pulsar el botón

buttonImage: URL de la imagen que usamos como botón para abrir el calendario

buttonImageOnly: Sólo aparecea la imagen como botón

changeYear: Se muestra un campo select con el que cambiar el año actual del calendario

numberOfMonths: se muestran varios meses a la vez en varios calendarios

onSelect: evento que se da cuando el usuario selecciona una fecha. Hay muchos eventos: onChangeMonthYear, beforeShow, onClose, etc.

Widget jQueryUI – Datepicker

```
/* Inicialización en español para la extensión 'UI date picker' para jQuery. */
/* Traducido por Vester (xvester [en] gmail [punto] com). */
jQuery(function($){
    $.datepicker.regional['es'] = {
        closeText: 'Cerrar',
        prevText: '<Ant',
        nextText: 'Sig>',
        currentText: 'Hoy',
        monthNames: ['Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio', 'Julio', 'Agosto', 'Septiembre', 'Octubre', 'Noviembre', 'Diciembre'],
        monthNamesShort: ['Ene','Feb','Mar','Abr', 'May','Jun','Jul','Ago','Sep', 'Oct','Nov','Dic'],
        dayNames: ['Domingo', 'Lunes', 'Martes', 'Miércoles', 'Jueves', 'Viernes', 'Sábado'],
        dayNamesShort: ['Dom','Lun','Mar','Mié','Juv','Vie','Sáb'],
        dayNamesMin: ['Do','Lu','Ma','Mi','Ju','Vi','Sá'],
        weekHeader: 'Sm',
        dateFormat: 'dd/mm/yy',
        firstDay: 1,
        isRTL: false,
        showMonthAfterYear: false,
        yearSuffix: '';
    };
    $.datepicker.setDefaults($.datepicker.regional['es']);
});
```

Widget jQueryUI – Datepicker

Ocultar/mostrar el calendario:

- `$("#campofecha").datepicker().show()`
- `$("#campofecha").datepicker().hide()`

Habilitar/deshabilitar el calendario:

- `$("#campofecha").datepicker("option", "disabled", true)`
- `$("#campofecha").datepicker("option", "disabled", false)`

Mostrar la semana del año en la que nos encontramos:

- `$("#campofecha").datepicker("option", "showWeek", true)`

Mostrar la barra de botones:

- `$("#campofecha").datepicker("option", "showButtonPanel", true)`

Fijar la fecha del calendario:

- Desplaza la fecha "+3m" de la fecha actual:
`$("#campofecha").datepicker("option", "setDate", "+3m")`
- Establece la fecha por defecto al abrir el calendario "+1m" de la fecha actual:
`$("#campofecha").datepicker("option", "defaultDate", "+1m")`

Widget jQueryUI – Datepicker

Establecer límites en las fechas que se pueden seleccionar:

- `$("#campofecha").datepicker("option", "minDate", "-20D")`
- `$("#campofecha").datepicker("option", "minDate", "01/01/2016")`
- `$("#campofecha").datepicker("option", "maxDate", "+2M, -10D")`
- `$("#campofecha").datepicker("option", "maxDate", "31/12/2016")`

Seleccionar un rango de fechas mediante dos Datepicker asociados:

```
$(function () {  
    $("#from").datepicker({  
        onClose: function (selectedDate) {  
            $("#to").datepicker("option", "minDate", selectedDate);  
        }  
    });  
    $("#to").datepicker({  
        onClose: function (selectedDate) {  
            $("#from").datepicker("option", "maxDate", selectedDate);  
        }  
    });  
});
```

Widget jQueryUI – Caja modal

jQueryUI nos ofrece la posibilidad de crear cajas de diálogos totalmente configurables.

1. Primera caja de diálogo: `$("#dialogo").dialog();`
2. Segunda caja de diálogo: Se usa jQuery para crear el contenido de la caja de diálogo sin mostrar la capa en la página, respondiendo a un evento click sobre un encabezado. También se ha añadido un cambio en el puntero del ratón cuando se hace hover sobre el texto del encabezado.

```
var caja2 = $('<div title="Segunda caja"><p>Segunda caja de diálogo.</p><p>En este caso, se crear el div y, sin mostrarlo en la página, lo creo en una caja de diálogo.</p></div>');
```

```
$("#dos").click(function(){
    caja2.dialog()
}).hover(function(){
    $("body").css('cursor','pointer');
},function(){
    $("body").css("cursor", "auto");
});
```

Widget jQueryUI – Caja modal

3. Tercera caja de diálogo: Se crea una ventana modal con las propiedades más comunes. Se aplica el mismo método que en el punto 2.

```
var caja3 = $('<div title="Tercera caja"><p>Tercera caja de diálogo (Modal).</p></div>');
```

```
$("#tres").click(function(){
```

```
    caja3.dialog({
```

```
        modal: true, // Es ventana modal
```

```
        title: "Caja con opciones", // Título de la ventana
```

```
        width: 550, // Ancho de la ventana
```

```
        minWidth: 400, // Ancho mínimo permitido
```

```
        maxWidth: 650, // Ancho máximo permitido
```

```
        show: "Fold", // La ventana no se muestra de golpe sino con efecto
```

```
        hide: "Scale", // La ventana no se oculta de golpe sino con efecto
```

```
        buttons: { // Botones
```

```
            "Quiero abrir la segunda ventana": function() {
```

```
                caja2.dialog();
```

```
                $( this ).dialog( "close" );
```

```
            }, Cancel: function() {
```

```
                $( this ).dialog( "close" );
```

```
            }
```

```
        } });
```

```
}).hover(function(){ $("body").css('cursor','pointer'); },function(){ $("body").css("cursor", "auto");
```

```
});
```

Efectos: Para poder usar los efectos hemos tenido que descargarlos (seleccionándolos) en la página de descarga de jQueryUI.

Widget jQueryUI

jQueryUI nos ofrece otros widget como son:

- Accordion
- Autocomplete
- Button
- Menu
- Progressbar
- Selectmenu
- Slider
- Spinner
- Tabs
- Tooltip

Efectos jQueryUI

jQueryUI nos permite añadir efectos a los elementos de nuestra página.

Para consultar los widget y efectos que soporta jQueryUI ir a: <http://jqueryui.com/demos/>