

DOCUMENTACIÓN

DWEC

1	Introducción.....	2
2	Librerías usadas	2
3	Archivos .js	2
3.1	eliminarUsuario.js	2
3.2	reservas.js	2
3.3	crearUsuarios.js	3
3.4	calendario.js.....	4
3.5	administracion.js.....	4

1 Introducción

En este documento se hablará a cerca de las librerías que han sido utilizadas y los scripts que se han generado. Cabe destacar que todas las redirecciones desde estos scripts se han realizado por medio del objeto Windows.

2 Librerías usadas

Se han usado las siguientes librerías:

- Bootstrap para generar el slideShow.
- JQuery.
- JQuery UI para mostrar y obtener datos del datepicker de las paginas horario y citas.

3 Archivos .js

3.1 eliminarUsuario.js

En este archivo se ejecuta una conexión AJAX con un fichero PHP el cual yo he generado. Además de esta conexión, usando JQuery se obtiene el nick del usuario que hay logueado en ese momento para un posterior mensaje de alerta. Si se confirma la eliminación de dicho usuario se ejecuta AJAX y si no se confirma la eliminación no se ejecutará dicha eliminación.

3.2 reservas.js

En este archivo se tiene en cuenta la paginación de los registros de la base de datos usando sessionStorage que recogemos nada más ejecutar el archivo y después modificamos al hacer click sobre los botones pertinentes de la paginación. En ese momento se lanza una conexión AJAX para comunicar la página actual con PHP donde se genera otra sesión.

Por otra parte, al hacer click fuera de la caja que engloba los registros de reservas (#reservas), se elimina el sessionStorage de la página actual y se hace otra conexión AJAX para eliminar la sesión de PHP.

Por último, al lado de cada registro podemos ver 2 iconos, uno de un PDF, del cual no vamos a hablar, y otro de una papelera. En este último, al clickar sobre una papelera se coge el id que hay en esa misma fila/registro, se lanza mensaje de comprobación para saber si el usuario está seguro de eliminar esa reserva y si acepta la eliminación, se realiza una conexión AJAX para enviar el id de la reserva a PHP y que se encargue de eliminar dicho registro.

3.3 crearUsuarios.js

Este archivo está relacionado con la opción de poder crear un usuario, pero no es el encargado de crearlo por ninguna conexión AJAX. En él se hace lo siguiente:

Inicialmente el botón que hace submit se deshabilita usando JQuery, para el resto de input se han realizado eventos focus, blur y change.

En cuanto al evento focus, ejecuta una función llamada foco la cual se encarga de a ese elemento aplicarle unos estilos css, esto mismo ocurre para el evento blur que ejecuta la función noFoco.

En cambio, en el evento change se tienen en cuenta varias cosas:

Se mira el atributo name para saber que función de validación ejecutar y además si todos esos campos no están vacíos y el input de submit sigue estando desactivado le quita esa propiedad para que quede activo.

Las funciones que se ejecutan según el valor del name de cada input son:

- validarNombreOApellidos → Esta función se ejecuta tanto si el name es nombre como si es apellidos. Lo que esta hace es ver si se ha introducido un espacio, lo que significaría que se ha introducido más de un nombre y aplicarle a cada nombre una expresión regular, para que la pase tendría que escribir algo así: Antonio Jose.
- validarEdad → Esta función se ejecuta cuando el name es edad y comprueba si la edad introducida esta entre 7 y 100.
- validarCorreo → Esta función se ejecuta cuando el name del input es email y hace su validación por medio de una expresión regular.
- validarTelefono → Esta función se ejecuta cuando el name es telefono y solamente comprueba que el teléfono introducido tiene una longitud igual o superior a 9.
- validarNick → Esta función se ejecuta con el campo con el name nick, se encarga de comprobar que no haya espacios en el nick y que si longitud sea por lo menos 1.
- validarContraseña → Esta función se ejecuta tanto para el campo de la contraseña como para el de la contraseña de verificación, en esta solo se mira que el campo no este vacío.

A todas las funciones anteriores se le pasa como parámetro el input que se va a validar.

Cuando los datos no son correctos en las validaciones anteriores se ejecuta una función llamada err, esta limpiar el value del campo erróneo y le pone fondo rojo.

3.4 calendario.js

Este script se encarga de realizar la configuración del datepicker para que este en castellano.

Este datepicker es de una librería llamada JQuery UI.

Con este objeto, inicialmente oculto hasta que se ejecuta este script, uso el objeto Date para obtener los valores del mismo, además, si aún no he clickado en ningún día también uso el objeto Date para coger la fecha actual. Dichas fechas se guardan en un sessionStorage el cual se elimina al clickar fuera de la caja section y ahí mismo se hace una conexión AJAX para eliminar las sesiones generadas a partir de este archivo en PHP.

Para tener dicha fecha siempre en PHP se ha realizado una conexión AJAX para pasar por parámetro get el día de la semana en el que se encuentra ahora mismo y también para enviarle a PHP el objeto Date con la fecha.

En este archivo también se genera otro sessionStorage con las mismas características que el anterior, pero este guarda los elementos que han sido checked para que al cambiar de día sigan estando marcadas las opciones de motivo de la cita.

Por último, en el archivo actual se usa JQuery para modificar el elemento #fecha y concatenarle la fecha. También se usa DOM para modificar todos los value del select de las fechas a seleccionar para reservar la cita y se le incrusta dicha fecha antes de la hora que pone PHP.

3.5 administracion.js

En este archivo también existe la funcionalidad explicada anteriormente en el archivo reservas.js.

Esta página tiene un efecto que se ejecuta cuando se pulsa cualquier botón para que te aparezcan los registros asociados a cada botón, lo muestra a modo de slideDown. Antes de esto, nada más cargar la página se ha tenido que ocultar dicha tabla usando un slideUp.

También tiene la funcionalidad de si clickas fuera de #modificaciones se eliminan las sesiones tanto de JS como de PHP (conexión AJAX).

Por otra parte, se hacen otras 2 conexiones AJAX para las papeleras de los registros de servicios y usuarios, en las cuales se le envían a PHP los respectivos id's del registro en el que se ha hecho click para que sea eliminado. Tras ejecutar esto se hace una recarga de la página y usando JQuery se fuerza el click en el botón pertinente para volver a ver la tabla.

En este archivo existe una última funcionalidad, esta se encuentra en la tabla de horario, en dicha tabla al ir seleccionando horas de apertura y cierre, de izquierda a derecha, los campos se van actualizando y se van eliminando las opciones que no van a poder ser seleccionadas usando JQuery.