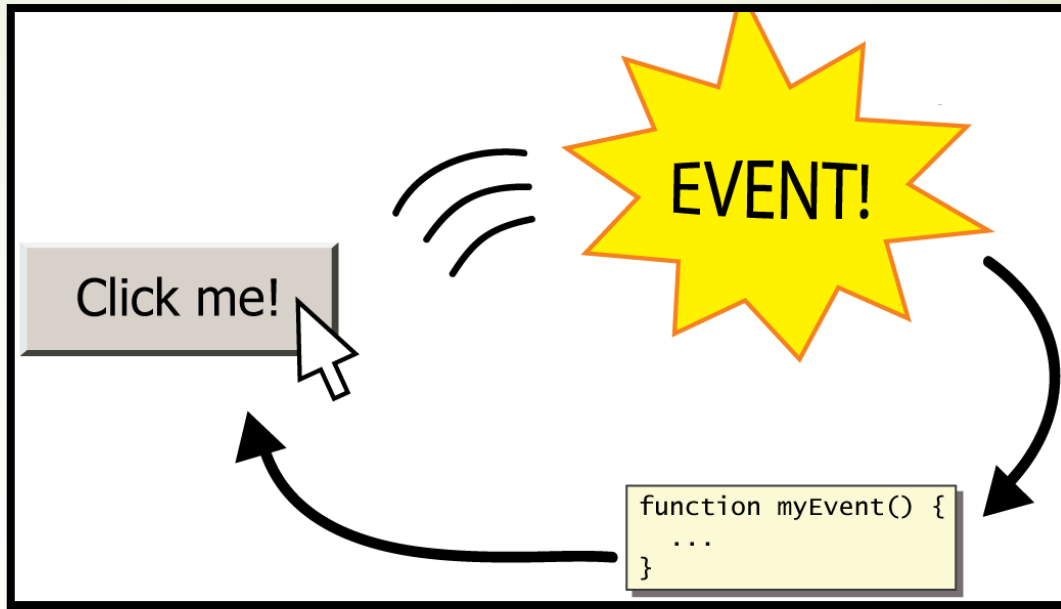


Desarrollo de Aplicaciones Web

DOCENTE: Daniel López Lozano





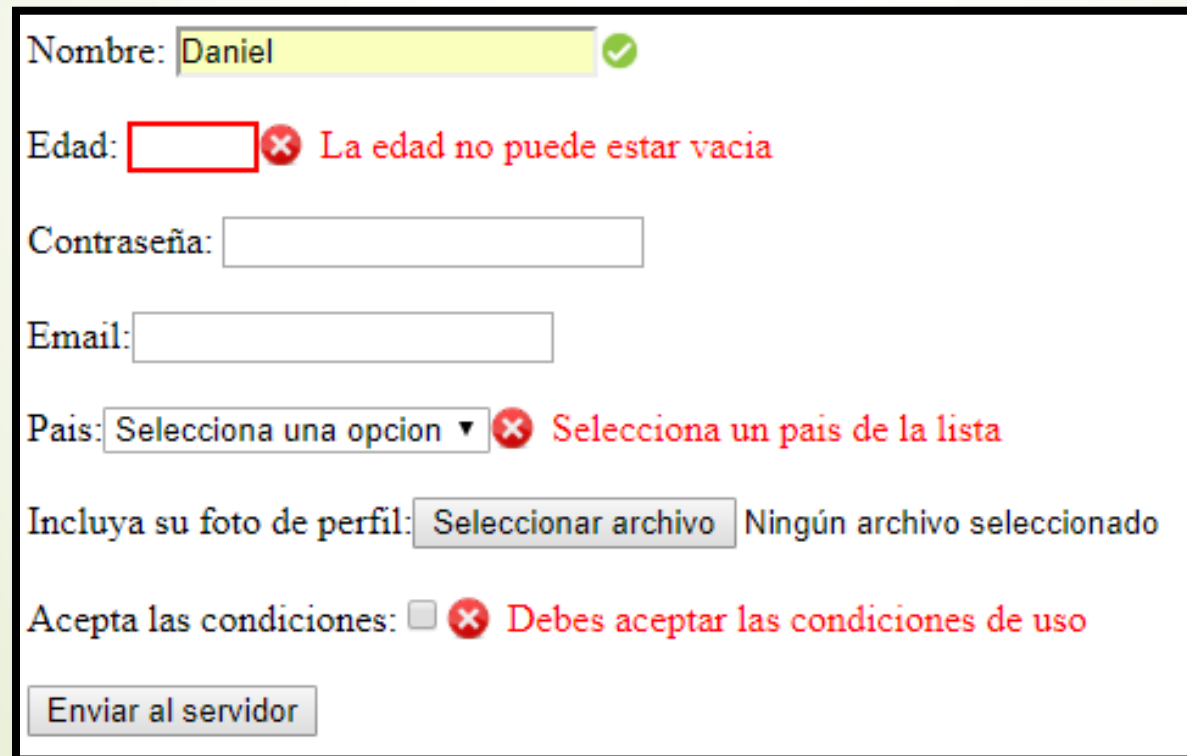
Tema 4.

Validación de formularios

Índice de contenidos

- ❑ Validación de componentes especiales.
- ❑ Validación con JQuery Validation

- ❑ Vamos a realizar de manera sencilla una validación completa de formularios.
- ❑ Como vimos la forma fundamental es usar `preventDefault` e ir comprobando lo que se cumple o no mediante condiciones para cada caso



Nombre: ✓

Edad: ✗ La edad no puede estar vacia

Contraseña:

Email:

Pais: ✗ Selecciona un pais de la lista

Incluya su foto de perfil: Ningún archivo seleccionado

Acepta las condiciones: ☐ ✗ Debes aceptar las condiciones de uso

El código JavaScript y CSS necesario

```
function Validar(evento)
{
    let input_nombre=$("#nombre");
    if(input_nombre.val()=="")
    {
        input_nombre.next().html("Nombre vacio");
        input_nombre.next().removeClass("ok").addClass("error");
        evento.preventDefault();
        ...
    }else{
        input_nombre.next().removeClass("error").addClass("ok");
    }
}
```

```
label.ok
{
    background-size: 18px;
    background-repeat: no-repeat;
    background-image: url("error.png");
    padding-left: 25px;
    color:red;
}
```

```
label.error
{
    background-size: 18px;
    background-repeat: no-repeat;
    padding-left: 25px;
    background-image: url("ok.jpg");
}
```

- ❑ Todas estas validaciones podríamos comprobarlas en el momento de pulsar el botón submit o conforme se va escribiendo.
- ❑ Para ello programaríamos alguno de los siguientes eventos

```
$("#form").submit(Validar);  
$("#nombre").keypress(ValidaNombre);
```

- ❑ Pero en lugar de hacerlo desde 0 vamos a usar un plugin basado en JQuery llamado **JQuery Validation**

<https://jqueryvalidation.org/>

- ❑ La documentación que dispone JQuery Validation es muy completa por lo que vamos a desarrollar un ejemplo y el resto lo veremos en la propia documentación.



- ❑ Una lección básica en programación es no reinventar la rueda.
- ❑ Para ello necesario incluir los ficheros jquery.validate.min.js y additional-methods.min.js que podemos encontrar en su web.

- ❑ El formulario sobre el que vamos a trabajar como ejemplo.

Validando un formulario completo

Nombre

Apellidos

Email

Password

Confirma password

Estoy de acuerdo con la política

☐

Selecciona

▼

Seleccionar archivo

Ningún archivo seleccionado

Enviar

- ❑ En primer lugar hay que indicar de que formulario/s se va a ocupar JQuery Validation aplicando la función validate
- ❑ Necesita únicamente un objeto JavaScript que le proporcione las siguientes claves.

```
$("#registro").validate({  
    submitHandler: EsValido, //Funcion que se ejecuta antes de enviar todo  
    success: "ok", //Clase que se aplica a elementos validos  
    errorClass: "error", //Clase que se aplica a elementos invalidos  
    ...  
});
```

```
function EsValido(form)  
{  
    alert("Es valido siga adelante");  
    form.submit();  
}
```

- ❑ La parte más importante y funcional son las claves **rules** y **message**.
- ❑ Las reglas indican que comprobaciones se va hacer a cada elemento.
- ❑ Los mensajes indican que mensaje se va a escribir en caso de incumplir una regla concreta.
- ❑ Para hacer referencia a un objeto es necesario usar su atributo name para hacer la correspondencia entre regla y mensaje.

```
rules:{
  nombre:{
    required:true,
    minlength:5,
    maxlength:20
  },
  apellidos:{
    required:true,
    range:[10,50]
  },
  correo:{
    email:true
  },
  password:{
    required:true,
    minlength:5
  },

```

```
  re_password:{
    required:true,
    minlength:5,
    equalTo:"#password"
  },
  tema:{
    required:true,
    minlength:2
  },
  deacuerdo:{
    required:true
  }
},//Cierra rules
```

Mensajes

```
messages:
{
  nombre:{
    required:"El nombre es obligatorio",
  },
  apellidos:{
    required:"Los apellidos son obligatorios",
  },
  correo:{
    email:"_El correo no es valido_"
  },
  password:{
    required:"El password es obligatorio",
    minlength:"Tiene que ser minimo 5 caracteres"
  },
}
```

Mensajes

```
re_password:{
  required:"El password es obligatorio",
  minlength:"Tiene que ser minimo 5 caracteres",
  equalTo:"Tienen que coincidir los password"
},
tema:{
  required:"Tienes que elegir algun tema",
  minlength:"Debes elegir al menos 2 temas"
},
deacuerdo:{
  "Tienes que estar de acuerdo con las condiciones"
}
} //Cierra messages
```

Regla	Descripción
required	Hace al elemento obligatorio
minlength	Establece el numero mínimo de caracteres
maxlength	Establece el numero máximo de caracteres
rangelength	Es una mezcla de los dos anteriores
min	Establece el valor numérico mínimo
max	Establece el valor numérico máximo
range	Es una mezcla de los dos anteriores
email	Obliga a que sea un email básico
url	Comprueba que empieza por https:// y una url básica
date	Comprueba una fecha estándar AAAA-MM-DD
number	Obliga a que sea un numero con valor decimal usando . decimal
digits	Solo dígitos sin . decimal
accept	Comprueba que el tipo de un archivo a subir

Comprobación de tipos de fichero

```
rules:{  
  fichero:{  
    accept:"image/jpeg,image/png"  
  }  
  ó  
  fichero:{  
    accept:"image/*"  
  }  
}
```

https://developer.mozilla.org/es/docs/Web/HTTP/Basics_of_HTTP/MIME_types/List_a_completa_de_tipos_MIME

- ❑ Este plugin contiene una base muy completa, pero puede no tener todo lo que se necesite.
- ❑ Permite añadir reglas nuevas que se hacen creando métodos de validación.
- ❑ La forma de expresar formatos de datos más complejos de los de por defecto de JQuery Validation es mediante expresiones regulares.
- ❑ Las expresiones regulares no es algo fácil por lo que no vamos a estudiarlas demasiado.

- ❑ Podemos ver ejemplos aquí de expresiones regulares.
 - ❑ <http://jquery-manual.blogspot.com/2012/05/expresiones-regulares-con-jquery-match.html>
- ❑ En este caso vamos a hacer que el nombre debe empezar por letra.

```
//Añadir la regla
$.validator.addMethod("letra", EmpiezaPorLetra, "Error de empezar por numero");

//La function debe devolver true si es valido
//Si no ponemos this.optional no tendria en cuenta si fuera opcional
function EmpiezaPorLetra(valor, elemento)
{
    var empiezaletra = /^[a-zA-Z][a-z0-9_-]*$/;
    return this.optional(elemento) || valor.match(empiezaletra);
}
```

- ❑ Una vez hecho esto se puede utilizar como una regla más dentro del objeto JavaScript de validación.

```
rules:{  
  nombre:  
  {  
    required:true,  
    letra:true  
  },  
}
```

- ❑ A continuación vamos a mostrar como hacer comprobaciones a mano de elementos especiales como son los select, checkbox y tipo de los ficheros.
- ❑ El objetivo es no depender directamente de JQuery Validation y poder aplicarlos por separado.
- ❑ Se supone que comprobaciones sobre un input de tipo text se determina usando distintas condiciones como vimos al principio del tema.

- ❑ Para saber si un checkbox esta seleccionado usamos el método prop.

```
let input_acepta=$("#acepta");  
if(input_acepta.prop("checked"))  
{  
    alert("Muy bien hecho");  
}else{  
    alert("Debes aceptar las condiciones de uso");  
    evento.preventDefault();  
}
```

- ❑ De un select podemos la propiedad selectedIndex que nos da el numero de opción que se ha seleccionado.
- ❑ La función val nos devuelve el valor que ve el usuario.

```
let select_pais=$("#pais");  
if(select_pais.prop("selectedIndex")==0)  
{  
    alert("Selecciona un pais de la lista");  
    evento.preventDefault();  
}else{  
    alert("ok");  
}
```

Comprobación del tipo de fichero a mano

```
let input_foto=$("#foto").files[0];  
if(input_foto.type.indexOf("image/")===-1)  
{  
    alert("Debe ser un formato de imagen MIME");  
    evento.preventDefault();  
}else{  
    alert("ok");  
}
```

Bibliografía

- ❑ Gauchat, Juan Diego: “El gran libro de HTML5, CSS3 y Javascript”. Editorial Marcombo. 2012
- ❑ Vara, J.M. y otros: “Desarrollo Web en Entorno Cliente. CFGS”. Editorial Ra-Ma. 2012
- ❑ “Programación en Javascript”. Colección de artículos disponibles en la url.
<http://www.desarrolloweb.com/manuales/> Última visita: Septiembre 2017.
- ❑ Tutoriales
<http://www.w3schools.com/> Última visita Septiembre 2017
- ❑ Comesaña, J.L. : Técnico en Desarrollo de Aplicaciones Web.
<http://www.sitiolibre.com/daw.php>
Última visita Septiembre 2017
- ❑ Píldoras Informáticas. Curso de JavaScript.
<https://www.pildorasinformaticas.es/course/javascript-desde-0/>
Ultima visita Septiembre 2018.