

Laboratorio 6. HTML, CSS y JavaScript

Objetivos:

- Seguir consolidando las capacidades para entender y diseñar pequeñas aplicaciones en HTML, CSS y JavaScript, que utilicen los elementos más básicos del lenguaje como variables, asignaciones, DOM, eventos, funciones, objetos o arrays.
- Entender mejor la arquitectura de aplicaciones Web estructuradas con el patrón MVC.

Recursos:

- El ordenador del laboratorio u otro equipo con Linux y un navegador instalados
- Documentación en el Moodle de la asignatura

Actividades previas:

Para asegurar que puede completar este ejercicio durante la sesión de laboratorio Lab6, debe realizarlo antes de dicha sesión de laboratorio. Antes de empezar la sesión Lab6 asegúrese de haber leído este documento, así como de haber completado el fichero ftel-t3-lab6-2019.html, cuyo funcionamiento se describe en el Tema 3.1, siguiendo las instrucciones dadas aquí. No se olvide de transferir a su cuenta en el laboratorio los ficheros con el trabajo realizado previamente, bien llevándolos en una memoria USB, transfiriéndolos a través de la nube o por otro método equivalente.

Resultados:

Como resultado de esta práctica debe subirse a Moodle el fichero generado (Grupo-Apellido1-Apellido2-L6.html) con el resultado de la tarea que debe contener tanto la parte realizada en casa, como la específica de su grupo realizada en el laboratorio.

Programa objeto de la práctica

Empiece desde el entorno gráfico, abra un terminal de texto, cree un nuevo directorio **lab6**, entre en este directorio (**lab6**).

Ahora debe descargar de Moodle, en el directorio **lab6**, el fichero ftel-t3-lab6-2019.html con el esqueleto de la práctica. Dicho fichero contiene el esqueleto de una aplicación Web estructurada con MVC, que incluye una lista de películas.

Al ejecutar dicho programa con el navegador (hacer doble clic), se muestra la lista de películas junto con una serie de botones (mostrado en la captura 1 de la figura siguiente).

La aplicación tiene botones añadir nuevas películas, editar o borrar los existentes. Los títulos también son clicables y deben mostrar el director de la película al hacer clic en ellos. Pero la aplicación esta incompleta y solo funciona el botón de editar una película o el de borrarla.

Al clicar el botón de *Editar* de una película, aparece el formulario de edición de películas (vista editView) mostrado en la captura 6 de la figura. Los cajetines del formulario se inicializan con la pregunta y la respuesta guardadas en el modelo.

Una vez editado, se pulsa el botón *Actualizar* para sustituirlo en el modelo y volver a la lista de películas (vista indexView), tal y como muestra la captura 3 de la figura.

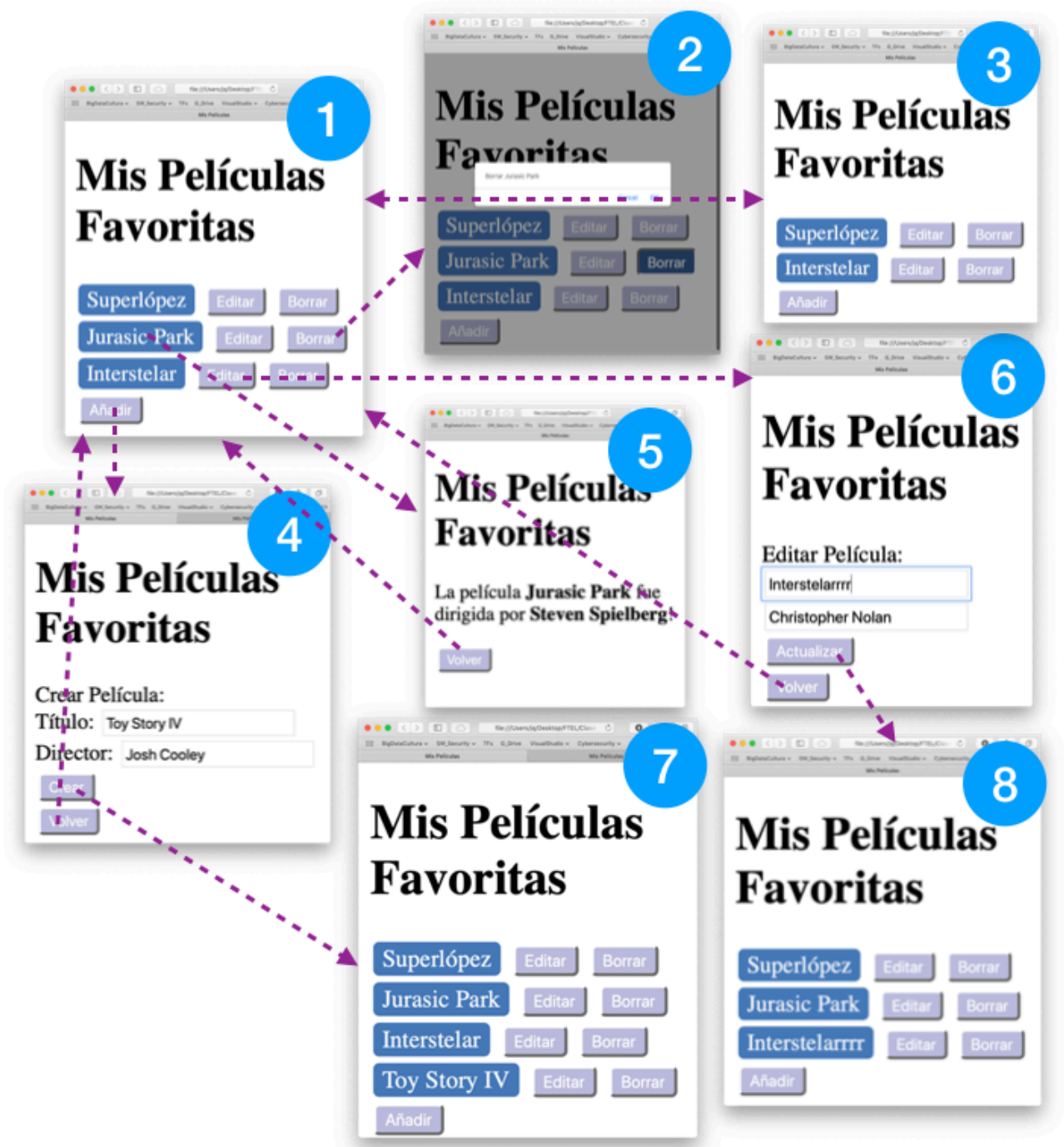
Al clicar el botón de *Borrar* de una película, aparece un desplegable de confirmación, mostrado en la captura 2 de la figura. Si pulsamos Ok, se borra la película del modelo y se vuelve a la lista de películas (vista indexView), tal y como muestra la captura 3 de la figura. Si se pulsa Cancel, simplemente desaparece el desplegable.

Cada película tiene un botón de *Editar* y de *Borrar* diferente para indicar al controlador con *i* la película a editar.

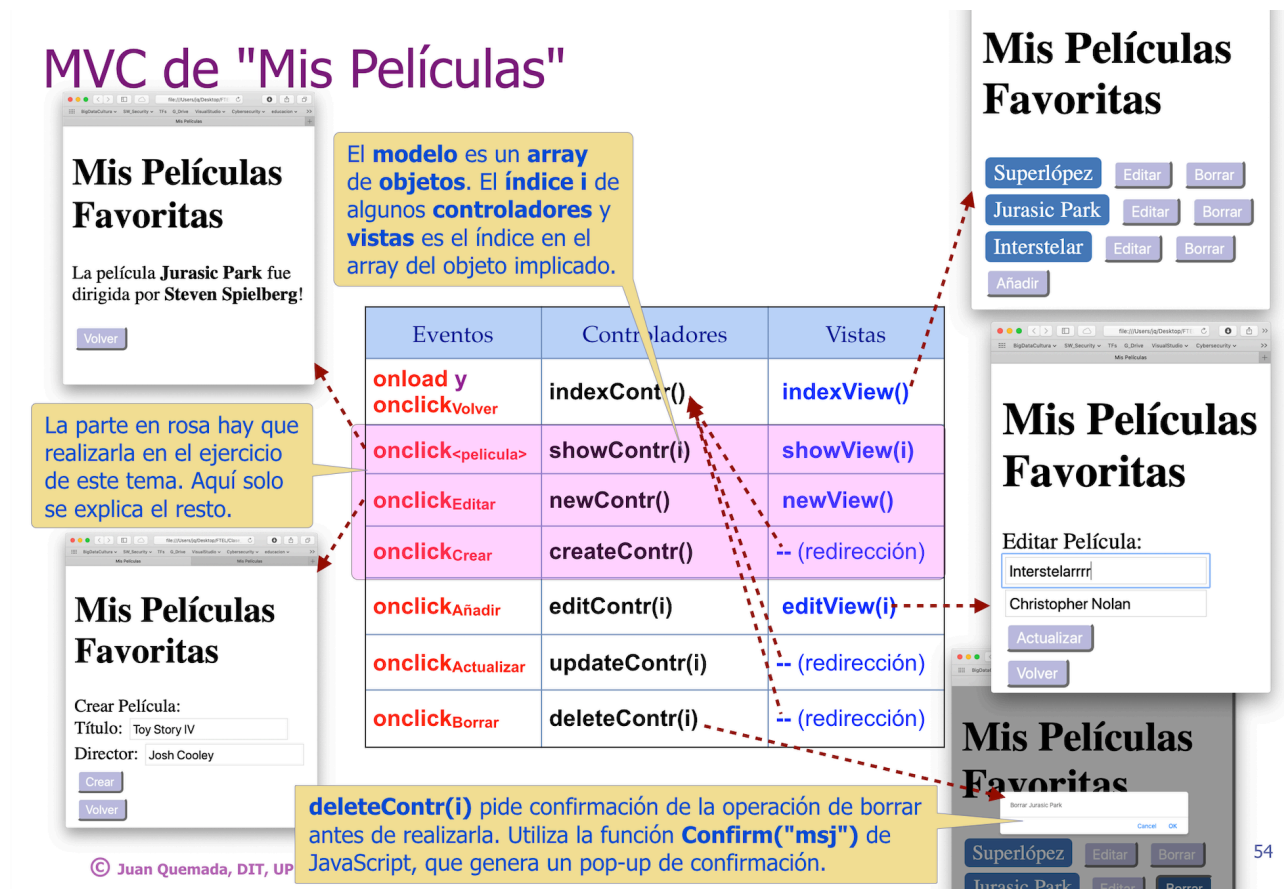
El esqueleto no incluye el código que implementa la creación de una nueva película (capturas 4 y 7) o el mostrar el director al clicar en el título (captura 5). El código asociado a estas funcionalidades debe crearse en esta práctica.

El código que implementa la creación de una película debe mostrar el formulario de creación de películas (vista *newView* mostrada en la captura 4 de la figura), al clicar el botón *Añadir*. Una vez tecleada la nueva película en los cajetines, se debe pulsar el botón *Crear* para añadirlo al modelo y volver a la lista de quizzes (vista *index*, la captura 7 de la figura) con la nueva película incluida.

También hay que añadir el código que muestra el director de la película al clicar en el título (captura 5). Al hacer clic en el título el controlador correspondiente debe recibir un índice *i* que indica que película es la clicada. El botón de *Volver* de esta vista y de potras donde aparece debe volver a la lista de películas (captura 1).



Esta aplicación se implementa con una estructura MVC con los eventos, controladores y vistas que muestra la siguiente tabla:



El esqueleto de código descargado que implementa la funcionalidad descrita es el siguiente:

```
<!doctype html><html><head><meta charset="utf-8">
<title>Mis Películas</title>
<style>
  button, .q {
    display: inline-block; text-decoration: none;
    padding: 2px 6px; margin: 2px;
    color: #FFF;
    border-radius: 4px; }
  .q { background: #4479BA; }
  button { background: #bbbbdd; }
  button:hover, .q:hover { background: #356094; }
</style>

<script type="text/javascript">

  // MODELO DE DATOS
  var mis_peliculas = [
    {titulo: "Superlópez", director: "Javier Ruiz Caldera"},
    {titulo: "Jurassic Park", director: "Steven Spielberg"},
    {titulo: "Interstellar", director: "Christopher Nolan"}
  ];
```

```

// VISTAS
function indexView () {
    var i=0;
    var view = "";

    while(i < mis_peliculas.length) {
        view = view + `
            <div>
                <span class="q" onclick="showContr(${i})">
                    ${mis_peliculas[i].titulo}</span>
                <button onclick="editContr(${i})">Editar</button>
                <button onclick="deleteContr(${i})">Borrar</button>
            </div>\n`;
        i = i + 1;
    };

    view = view + `
        <button onclick="newContr()"> Añadir</button>`

    return view;
};

function editView (i) {
    return `Editar Película: <br>
        <input type="text" id="titulo"
            value="${mis_peliculas[i].titulo}"><br>
        <input type="text" id="director"
            value="${mis_peliculas[i].director}"><br>
        <button onclick="updateContr(${i})">
            Actualizar</button><br>
        <button onclick="indexContr()">Volver</button>`;
}

function showView (i) {
    // Completar: genera HTML con información de la película
    // ...

}

function newView () {
    // Completar: genera formulario para crear nuevo quiz
    // ...

}

```

```

// CONTROLADORES

function indexContr() {
    document.getElementById('main').innerHTML = indexView();
};

function showContr(i) {
    // Completar: controlador que muestra la vista showView(i)
    // ...

};

function editContr(i) {
    document.getElementById('main').innerHTML = editView(i);
};

function updateContr(i) {
    mis_peliculas[i].titulo    = document.getElementById('titulo').value;
    mis_peliculas[i].director = document.getElementById('director').value;

    indexContr();
};

function newContr() {
    // Completar: controlador que muestra la vista newView()
    // ...

};

function createContr() {
    // Completar: controlador que actualiza el modelo con la nueva película
    // ...

};

function deleteContr(i) {

    // Genera desplegable: botón ok devuelve true, Cancel false
    if (confirm(`Borrar ${mis_peliculas[i].titulo}`)) {

        mis_peliculas.splice(i, 1); // elimina elemento i del array
        indexContr();
    };
};

// Inicialización
function init() {
    indexContr();
};
</script>
</head>

```

```

<body onload="init()">

    <h1>Mis Películas Favoritas</h1>
    <div id="main"> <!-- La vista se incluye aquí --> </div>

</body>
</html>

```

Realización de la práctica

Una vez creado el directorio lab6, descargado el fichero ftel-t3-lab6-2019.html y entendido su funcionamiento, debe realizar esta practica siguiendo estos pasos:

1. En primer lugar, debe cambiar el nombre del fichero ftel-t3-lab6-2019.html por otro que incluya su nombre con el siguiente formato Grupo-Apellido1-Apellido2-L6.html (p.e. G14-perez-sancho-L6.html).
2. A continuación debe cambiar el título de la aplicación por "Las Películas Favoritas de <Nombre y los 2 Apellidos>" en el código entre las marcas del elemento HTML <h1>.
3. También se debe añadir justo encima de la lista de películas un texto que diga: "Haga clic sobre una película para ver su director".
4. Debe completar el código que falta de la vista y del controlador que generan el formulario de creación de nuevas películas, para que funcione como muestra el diagrama de estados y vistas de la aplicación ya mostrado. En concreto, debe rellenar el código de:
 - a) La función *newView()* que implementa la vista del formulario de creación de nuevas películas y genera el HTML asociado.
 - b) La función *newContr()* que implementa el controlador que genera el HTML del formulario utilizando la vista (*newView()*) y muestra el formulario de creación de una nueva película en el navegador al insertarlo en el árbol DOM.

Para probar que el código que va introduciendo funciona bien debe guardar el contenido del fichero editado y hacer doble clic sobre el para abrirlo en el navegador y poder jugar con las películas.

5. Debe completar el código que falta en el controlador *createContr()*, que añade la nueva película tecleada como el último elemento del array con las películas del modelo. El método *mis_peliculas.push({..nueva-película...})* lo añade en esa posición (mas información en <https://javascript.info/array#methods-pop-push-shift-unshift>). El título y el director de la nueva película deben extraerse de los cajetines de la vista y deben utilizarse como valores asignados al nuevo objeto creado con el literal de objetos.

Para probar que el código que va introduciendo funciona bien debe guardar el contenido del fichero editado y hacer doble clic sobre el para abrirlo en el navegador y poder jugar con las películas.

6. Debe completar el código que falta en el controlador *showContr(i)*, que muestra el director de la película al clicar en el título, indicando con el índice *i* en el modelo (array de la variable *mis_peliculas*) de la película clicada.

Para probar que el código que va introduciendo funciona bien debe guardar el contenido del fichero editado y hacer doble clic sobre el para abrirlo en el navegador y poder jugar con las

películas.

7. Debe añadir al final de la vista *indexView()* un nuevo botón "Sorpresa" que inserte en un bloque `<div>` que se añada debajo del botón la vista *showView()* de una película del modelo elegida al azar. El botón debe invocar un nuevo controlador, p.e. *randomContr()*, que seleccione la película utilizando las funciones *Math.random()* y *Math.floor(..)* para generar un índice aleatorio entre *0* y *mis_peliculas.length-1*.
8. Al llegar a la sesión Lab6 deberá realizar las modificaciones específicas de su grupo y subir el fichero con la práctica: `Grupo-Apellido1-Apellido2-L6.html`.

Una vez realizados los cambios debe entregar este fichero.