

# PRÁCTICA 3.06 Interfaces con rutas en React

## Normas de entrega

- En cuanto al **código**:
  - en la **presentación interna**, importan los **comentarios**, la claridad del código, la significación de los nombres elegidos; todo esto debe permitir considerar al programa como **autodocumentado**. No será necesario explicar que es un **if** un **for** pero sí su funcionalidad. Hay que comentar las cosas destacadas y, sobre todo, las **funciones** y **clases** empleadas. La ausencia de comentarios será penalizada,
  - en la **presentación externa**, importan las leyendas aclaratorias, información en pantalla y avisos de ejecución que favorezcan el uso de la aplicación,
  - si no se especifica lo contrario, la información resultante del programa debe aparecer en la consola del navegador **console.log(información)**,
  - los ejercicios deben realizarse usando **JavaScript ES6**. No se podrá utilizar ninguna biblioteca (si no se especifica lo contrario en el enunciado),
  - para el nombre de **variables**, **constantes** y **funciones** se utilizará *lowerCamelCase*,
  - el nombre de los componentes debe comenzar con letra **mayúscula**,
  - el estilo que afecte a un componente debe estar contenido en un fichero **CSS** junto a ese componente y con su mismo nombre.
- En cuanto a la **entrega** de los archivos que componen los ejercicios de **React**:
  - entrega **la práctica** en un sólo proyecto (el nombre a tu discreción),
  - los componentes creados deben estar separados en carpetas (los creados en el Ejercicio1 dentro de una carpeta denominada **Ejercicio1**),
  - el código contendrá ejemplos de ejecución, si procede,
  - comprime la carpeta **src** junto con los ficheros **package.json** y **package-lock.json** en un fichero **ZIP** , y
  - sube a **Aules** el fichero comprimido.

### Ejercicio 1 - Rutas en **React** con **react-router-dom**

Crea la interfaz para una aplicación de gestión de películas. **Tan sólo se deberá crear la interfaz** con contenido de relleno, pero deberá incluir todas estos apartados (te aconsejo que primero dibujes en papel la estructura de la aplicación y luego la dividas en componentes):

- debe tener una estructura fija con componentes que hagan las siguientes funciones: **cabecera**, **menú**, **contenido** y **pie** de la aplicación (pueden, a su vez, **dividirse en componentes más específicos**, si lo crees oportuno),
- en **<Contenido>** se verterá la información de la web que vaya cambiando, por lo que será la parte que más se modifique a lo largo de la ejecución de la aplicación,
- crea un menú con los siguientes apartados: **inicio** (que mostrará el contenido inicial), **películas** (que mostrará el listado de las películas), **intérpretes** (que mostrará información sobre los actores y actrices), **galería** (con imágenes de las cartelas de las películas) y **acerca de** (información de la aplicación como su versión, creador/a y fecha de modificación),
- dentro de **películas** crea un **submenú** que permitirá (no hay que implementar esta funcionalidad ahora, sólo crear el menú) filtrar las películas por título, intérprete y/o director,
- en **galería** existe otro **submenú** para filtrar las imágenes (no hay que implementar esta funcionalidad ahora) que permita filtrar las cartelas por título, intérprete y/o director,
- crea una **estructura de carpetas** adecuada a la aplicación, que ya empieza a crecer,
- en caso de error, debe configurarse **una ruta por defecto** que muestre un nuevo componente denominado **<Error>**.

La aplicación puede tener un aspecto parecido a este, aunque es mejor que apliques tu propio estilo:

