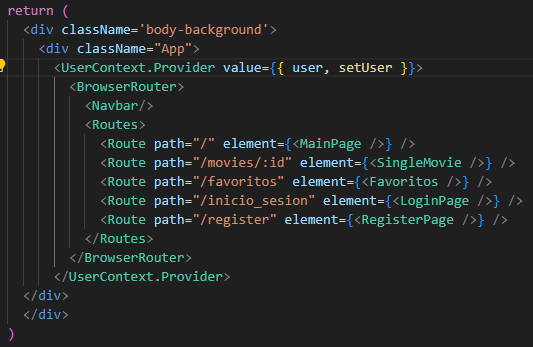
**I. COMPONENTES PRINCIPALES**

* Componente principal de la aplicación: **App.jsx** – Establece el orden en el que se mostrarán componentes principales y establece el estado inicial de los usuarios.

Árbol de componentes:

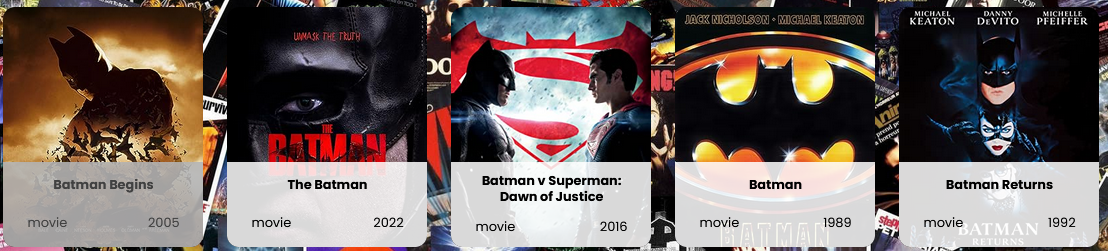
* **UserContext.jsx** – Permite que se puedan pasar todos los datos a través de todos los componentes sin tener que hacerlo en forma manual, se fija el estado inicial del usuario.
* **BrowserRouter** – Habilita el enrutamiento entre los componentes del navegador
* **Navbar.jsx** – Barra de navegación superior, al estar fuera de “Route” le permite permanecer en pantalla en todo momento.
* **Routes** – Establece todas las rutas contenidas en la app y los componentes que se mostrarán al ingresar a sus rutas.



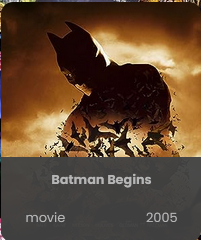
* **HOOKS - UseFetch.js** – En este archivo se realiza la conexión de la aplicación con la API de omdb y maneja todos los estados de las solicitudes, es decir, si la conexión a la API es exitosa o si existen errores. La constante useFetch se usa en otros componentes para extaer los datos desde la API.
* **CONTEXT – DataContext.jsx** – Crea la función que permite compartir los datos de la API entre los componentes. Se usa en App.jx en el componente <UserContext.**Provider**>. También acá está la función que permite realizar búsquedas en la API.
* **COMPONENTS – UserContext.jsx** – Crea el contexto para los usuarios, permite que se puedan pasar los datos dentro de todos los componentes hijos en App.jsx.

**II. PAGINA DE INICIO**

* **MainPage.jsx** – Página de inicio de la aplicación. Muestra los componentes **FormSearch.jsx** y **Movies.jsx**. Con “FormSearsch” se realiza la búsqueda y si existen resultados, se muestra “Movies”. Contiene función que cambia los datos de la página principal cada vez que se realiza una búsqueda (o sea muestra una nueva lista de películas)
* **FormSearch.jsx** – Contiene el logo de la página principal y la barra de búsqueda. Contiene función que realiza búsqueda al presionar el botón “Buscar”.
* **Movies.jsx** – Componente que muestra la lista de todas las películas. Como return contiene al componente “ItemMovies.jsx”. Contiene función que devuelve array con películas equivalentes a la búsqueda del usuario con los datos provistos por “ItemMovies.jsx”.



* **ItemMovies.jsx** – Componente que contiene la información de las películas que se muestra en los recuadros de información que se devuelven como resultados de búsqueda. Muestra carátula, nombre, tipo y año. Contiene función que devuelve datos de películas para renderizar.



* **NavBar.jsx** – Barra superior, muestra logo que redirige a página de inicio, barra de búsqueda, link de inicio de sesión, y una vez que usuario ha iniciado sesión, muestra mensaje de bienvenida, link a favoritos y link para cerrar sesión. Contiene funciones de cierre de sesión y búsqueda.

**III. RESULTADOS DE BUSQUEDA**

* **SingleMovie.jsx –** Muestra información de la película seleccionada desde los resultados de búsqueda. Se muestran en pantalla datos de películas obtenidos desde la API a través de variables que provienen de useFetch.jsx. Contiene funciones para añadir video desde YouTube y añadir a favoritos. Esta última función contiene variables que verifican si el estado está autenticado antes de habilitar la opción de añadir a lista (withCredentials: true). Función de añadir a favoritos contiene control de errores que avisan a usuario su estado y muestran pop-ups con información usando el componente “Modal” que se encuentra en el return.

En el return también se encuentran los componentes que permiten compartir info con redes sociales y mostrar el recuadro de comentarios de Disqus. Para usar Disqus es necesario registrarse en esa plataforma o usar alguna de las opciones provistas (Facebook, Gmail, etc). Estas dos últimas opciones no dependen de la autenticación del usuario.

**IV. AUTENTICACIÓN DE USUARIOS**

* **LoginPage.jsx**: Página de inicio de sesión. Contiene función que verifica nombre y contraseña de usuarios, y en caso de éxito lleva realiza solicitud post a /inicio\_sesion en backend, que realizar todas las consultas a la base de datos. Si usuario no existe, aparecerá un mensaje de error en pantalla. Contiene función que autentica al usuario al hacer clic en el botón “Iniciar sesión”
* **RegisterPage.jsx:** Página de creación de nuevos usuarios. Hace petición post a /register en el backend para insertar datos en base de datos. Una vez registrado, el usuario es redirigido a la página de inicio de sesión. Contiene función que crea cuenta de usuario al hacer clic en botón “Crear Cuenta”.

**V. FAVORITOS**

* **Favoritos.jsx**: Página que muestra la lista de favoritos. Contiene función que muestra lista de favoritos para usuarios autenticados. Primero verifica estado del usuario para luego hacer llamado al backend en la ruta /favoritos y realizar las consultas SQL de inserción y visualización de datos. También contiene función para eliminar película de la lista, al hacer llamado al backend y realizar la consulta pertinente.