

PARCIAL 2

Desarrollar una aplicación web que cumpla con los siguientes requisitos:

- Debe estar desarrollada con React utilizando el CLI de [create-react-app](#).
- Contar con una lista de recursos siguiendo la lógica de un [ABM](#).
- Debe contar con una librería para el manejo de formularios. (sugerencia [final-form](#))
- Debe contar con un [layout](#), el cual debe contener:
 - Header
 - NavBar (barra de navegación)
 - Body o contenido
 - Footer
- Debe contar con más de 1 ruta navegable.
- La app debe tener AL MENOS 2 pantallas:
 - Home: Esta debe contar con los nombres de los integrantes y el nombre de la aplicación. Es básicamente para poder probar el ruteo de la aplicación.
 - Recurso: en esta pantalla deberán mostrar la lista de recursos con la lógica del AMB (si trabajan solos, solo deberán tener home y la del recurso trabajado) para poder evidenciar la navegación dentro de la app.
- Debe contar con componentes tanto para el Header, Body y Footer, como también un componente para la tabla. A su vez, estos componentes deben utilizar componentes compartidos, como ser: botones, inputs, modal etc. Estos componentes deben contar con sus archivos .css, preferentemente utilizando CSS modules.
- Debe contar con formularios a la hora de realizar un POST, un DELETE o un UPDATE de un nuevo recurso a la lista.
- Debe contar con modals cuando se solicite agregar, eliminar o modificar un recurso a la lista. Dentro de los modals se deberá mostrar el formulario y/o el mensaje de confirmación de eliminación o actualización de un recurso.
- Cada formulario debe contar con las validaciones correspondientes para evitar agregar nuevos recursos con datos erróneos o sin datos.
- Debe ser posible realizar las diferentes request (GET, POST, PUT, DELETE) desde el ABM de cada recurso.
- Debe contar con un archivo donde se haga la configuración inicial del store de Redux.
- Debe contar con un reducer por cada recurso y un rootReducer.
- Debe contar con un archivo de actions por cada recurso.
- Debe contar con un archivo de types (constantes) para las actions.
- Debe contar con el uso de action creators utilizando la librería Redux Thunk, los cuales son necesarios para realizar la conexión del FE con el BE. Para esto, deberán utilizar la API ya realizada para el primer parcial.

- El proyecto del servidor debe estar subido a un repositorio de Github a nombre del alumno (o uno de los alumnos del grupo)
- Debe contar con un readme con los pasos a seguir para poder ejecutar correctamente la aplicación, URL del repositorio y nombre del alumno.
- El repositorio debe contener código prolijo, segmentado en commits.
- En caso de trabajar en grupo, todos los integrantes deberán contar con commits a su nombre, caso contrario no se podrá considerar que ese alumno haya trabajado en el proyecto.

El proceso de evaluación será:

1. Abrir la URL que se provee a la hora de levantar el proyecto.
2. Mostrar los diferentes métodos solicitados. Esto quiere decir, obtener, agregar, modificar y eliminar un recurso.
3. Mostrar los formularios para agregar y modificar los recursos, con sus respectivas validaciones.
4. Mostrar los modals creados para mostrar tanto los formularios previamente mencionados, como también los modals de confirmación a la hora de agregar o eliminar un nuevo recurso.
5. Utilizando la herramienta **Redux devtools**, deberán mostrar las acciones que se despachan cuando realizan las diferentes acciones, por ejemplo, la de GET.
6. Revisar la calidad del código del frontend en Github y que **el alumno tenga commits a su nombre**.
7. Revisar el correcto entendimiento de las funcionalidades desarrolladas, revisando el código y charlando sobre el flujo de datos en la aplicación.