

# Evaluación Formativa III

## I. Presentación

En este ejercicio, vamos a crear una aplicación React desde cero que permitirá a los usuarios ver una lista de restaurantes, agregar comentarios a los restaurantes y eliminar tanto restaurantes como comentarios. Utilizaremos react-router-dom para la navegación y axios para interactuar con una API backend.

### Unidad de Aprendizaje III:

Framework JavaScript

### Aprendizaje esperado

2.1.- Codifica Script para procesar formularios HTML y modificar el DOM, acorde a problemática presentada. (Integra Competencia Genérica Resolución de Problemas Nivel 1).

### • Criterios de evaluación

Esta actividad, considera los siguientes criterios de evaluación:

- 3.1.1 Codifica componentes en JavaScript, basado en un Framework.
- 3.1.2 Codifica CRUD usando Local Storage.
- 1.3 Codifica script que permita cargar datos desde una API.
- 3.1.4 Establece propuestas de solución pertinentes Crea documentos anidados según requerimiento.

### Actividades

1. SPA

## II. Instrucciones

1. Puede utilizar frameworks para el diseño si así lo desea.
2. Desarrollar formulario web según requerimientos.
3. Una vez finalizado el proyecto comprimir todo en un archivo con el siguiente nombre "evaluacionII\_NombreAlumno.rar".
4. Se prohíbe el uso de cualquier tipo de IA.
5. Subir el archivo en ambiente de aprendizaje -> Programación Front End -> Evaluación III.

## III. Actividades

### 1. Configuración del Proyecto

- a) **Crear la aplicación React:** Crea una nueva aplicación React utilizando Create React App.
- b) **Instalar dependencias necesarias:** Instala las dependencias axios, react-router-dom y bootstrap.

### 2. Crear un Directorio de Servicios y Configurar Axios

- a) **Crear el directorio service:** Crea un directorio src/service y un archivo api.js dentro de él.
- b) **Configurar Axios en api.js:** Configura Axios con la URL base de tu API backend.

### 3. Configurar Rutas y Crear Componentes

- a) **Configurar rutas en AppRouter.js:** Configura las rutas para los diferentes componentes usando react-router-dom.
- b) **Crear el componente para listar restaurantes:** Crea un archivo src/components/RestaurantList.js que muestre una lista de restaurantes obtenidos desde la API y un botón para eliminar cada restaurante.
- c) **Crear el componente para mostrar detalles de un restaurante:** Crea un archivo src/components/RestaurantDetail.js que muestre los detalles de un restaurante seleccionado, incluyendo los comentarios. Añade funcionalidad para eliminar comentarios y agregar nuevos comentarios.
- d) **Crear el componente para agregar un nuevo restaurante:** Crea un archivo src/components/RestaurantCreate.js que contenga un formulario para crear un nuevo restaurante. Valida los campos del formulario y envía los datos a la API.
- e) **Crear el componente para editar un restaurante existente:** Crea un archivo src/components/RestaurantEdit.js que contenga un formulario para editar los detalles de un restaurante existente. Rellena los campos con los datos actuales del restaurante y permite actualizarlos.

### 4. Crear y Configurar el Formulario de Comentarios

**Crear el componente CommentForm:** Crea un archivo src/components/CommentForm.js que contenga un formulario para agregar comentarios a un restaurante. Valida los campos del formulario y envía los datos a la API. Muestra un mensaje de éxito y actualiza la lista de comentarios sin recargar la página.

### 5. Agregar Funcionalidad de Eliminar

- a) **Agregar botones de eliminar:** Añade botones para eliminar restaurantes y comentarios. Implementa las funciones para manejar la eliminación de estos elementos mediante llamadas a la API.