

Aprendizaje Basado en Proyectos: Uso de Hooks y Manejo de Errores en la Web del Hospital

Contexto:

En este proyecto, los estudiantes deberán implementar **Hooks** para la gestión de estado y efectos secundarios, además de manejar errores y excepciones dentro del sistema del hospital. A lo largo del proyecto, integrarán el **Hook de estado** y el **Hook de efecto** para interactuar con los datos del hospital. También se les pedirá que creen un **Hook personalizado** y apliquen estrategias para la detección y manejo de errores dentro de la aplicación React.

Duración: **2 horas**

Requisitos:

1. Uso de `useState` para la Gestión de Estado (1.5 puntos)

- Implementa el Hook **`useState`** para gestionar el estado en los componentes del sistema, como:
 - El manejo de formularios de pacientes o citas médicas.
 - Almacenar temporalmente la información ingresada por el usuario hasta que sea enviada o actualizada en la base de datos.

2. Uso de `useEffect` para la Gestión de Efectos Secundarios (1.5 puntos)

- Utiliza **`useEffect`** para manejar efectos secundarios como:
 - Realizar peticiones a la API para obtener los datos del sistema del hospital (doctores, pacientes).
 - Controlar cuándo y cómo se deben ejecutar ciertos procesos, como la actualización automática de datos.

3. Construcción de un Hook Personalizado (1.5 puntos)

- Crea un **Hook personalizado** para resolver una funcionalidad recurrente en la aplicación. Algunas ideas incluyen:

- Un Hook que maneje el formulario de registro de doctores y pacientes, validando los datos y devolviendo errores si es necesario.
- Un Hook para gestionar la autenticación del usuario, verificando si el usuario está autenticado o no.

4. Manejo de Errores en la Aplicación (1.5 puntos)

- Implementa un mecanismo para el manejo de errores y excepciones en la aplicación React, asegurando que:
 - Los errores durante las peticiones a la API se capturen y manejen adecuadamente, mostrando mensajes claros al usuario.
 - Se manejen errores comunes en la interfaz, como la validación de formularios o la ausencia de datos requeridos.

5. Aplicación Correcta de las Reglas de los Hooks (1 punto)

- Asegúrate de seguir correctamente las **reglas de los Hooks**, evitando:
 - Llamar Hooks de manera condicional o dentro de bucles.
 - Asegurarte de que los Hooks se llamen en el nivel superior del componente y respeten las mejores prácticas de ReactJS.

Herramientas a Utilizar:

- **ReactJS** para la implementación de Hooks.
 - **React Developer Tools** para depurar y verificar el comportamiento de los Hooks.
 - **API REST** para interactuar con los datos del sistema del hospital.
-

Entrega:

- **Formato de entrega:**
 - Opción 1: Enviar un **enlace al repositorio de GitHub** con el proyecto que incluya el uso de Hooks, el manejo de errores y un Hook personalizado.
 - Opción 2: Entregar un archivo **ZIP comprimido** con el proyecto ReactJS completo, asegurándose de que el manejo de errores, los Hooks y el Hook personalizado estén implementados correctamente.