Módulo 5
Desarrollo de aplicaciones Front-End con React

Hooks y Manejo de Errores

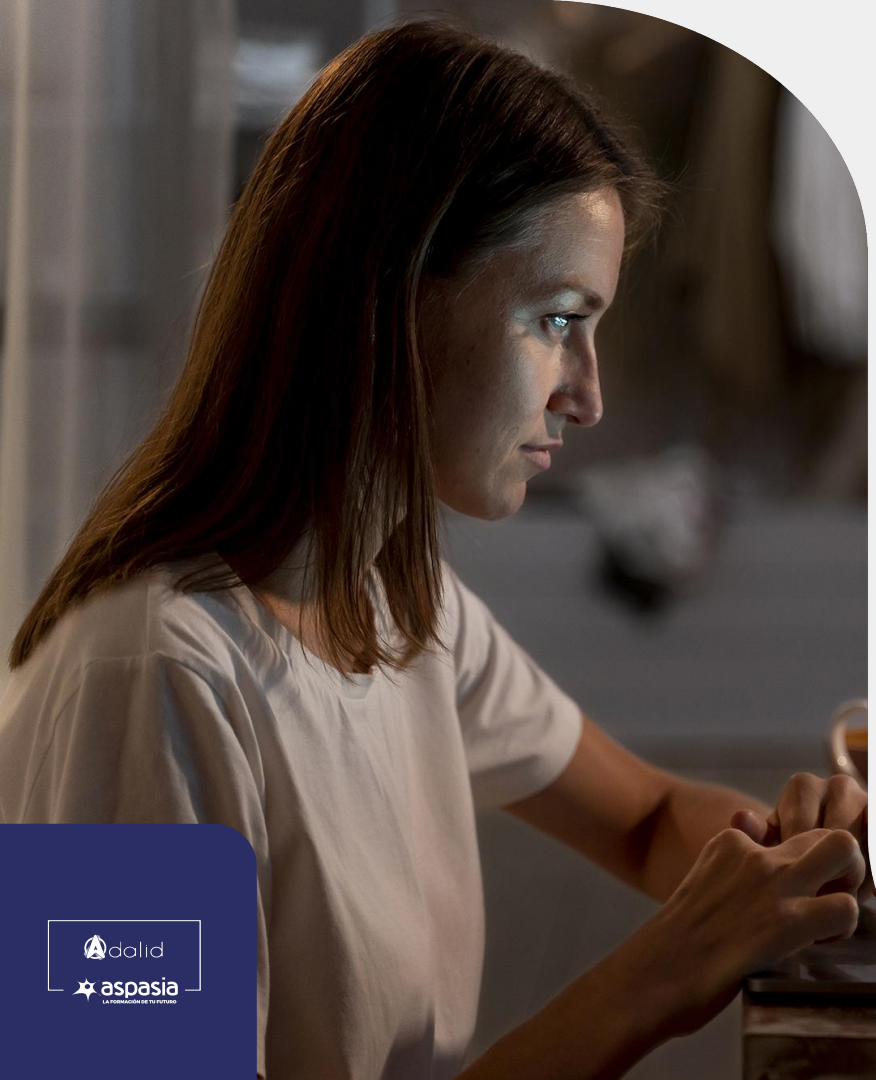
Módulo 5

AE 4.2

OBJETIVOS

Aprender a consumir APIs, manejar estado y efectos en React, optimizar rendimiento, manejar errores, y garantizar seguridad en aplicaciones Cliente/Servidor con buenas prácticas y herramientas modernas.

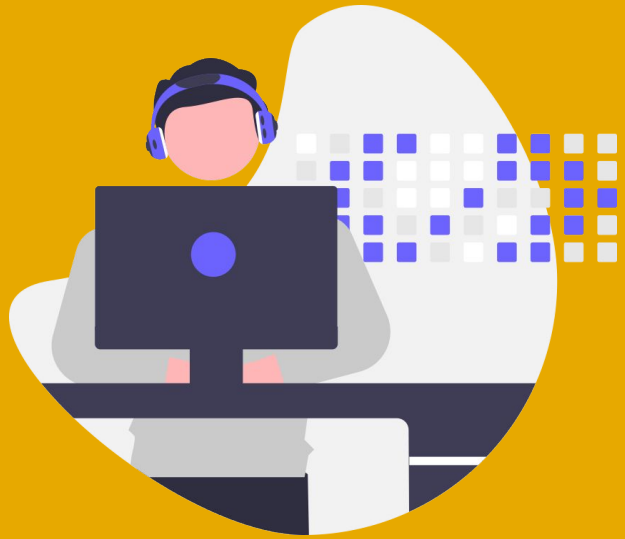
Chile
Adalid
aspasia
LA FORMACIÓN DE TU FUTURO



¿QUÉ VAMOS A VER?

- Qué son los Hooks.
- Un vistazo en detalle a los Hooks.
- Usando el Hook de estado.
- Usando el Hook de efecto.
- Reglas de los Hooks.
- Construyendo Hooks personalizados.
- Reconoce los mecanismos para el manejo de errores en un aplicativo React.
- Utiliza hooks en un aplicativo React para resolver un problema.
- Utiliza mecanismos disponibles en el entorno React para el manejo y control de los errores y excepciones.

Pongamos a prueba lo aprendido 😊!!!



Adalid aspasia

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

Tenemos problemas 🤨.

Actualmente tenemos una aplicación del clima que está conectada a una API y trae cierta información, pero presentó algunas fallas y tareas pendientes, afortunadamente un desarrollador nos ayudó identificando los errores y documentarlos en el proyecto.

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

Lo primero es descargar el proyecto y leer su documentación https://github.com/adalid-cl/ESPECIALIZACION_FRONTEND_M5_AE4, usaras una API en asi que te recomendamos leer parte de su documentación.

Como empezar el desarrollo y sus correcciones depende de ti, sin embargo te damos un resumen de lo que pasa en cada archivo.

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

WeatherForm.jsx

- **Error de Validación:**
  - No se verifica si el campo query está vacío antes de llamar a onSearch.
  - Esto podría causar una petición a la API con un valor vacío, lo que generaría un error.
- **Tarea Pendiente:**
  - Indicar que se debe agregar una validación para mostrar un mensaje o deshabilitar el botón si el input está vacío.

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

WeatherInfo.jsx

- **Validación faltante de datos:**
  - Si data es null o no tiene las propiedades esperadas (data.name, data.main, data.weather), el componente fallará.
  - Se introdujeron errores donde los datos podrían ser undefined.
- **Tareas pendientes:**
  - Validar la estructura de data antes de intentar renderizar las propiedades.
  - Agregar más información del clima, como humedad, velocidad del viento o presión atmosférica.
  - Mostrar una imagen genérica si el ícono del clima no está disponible.

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

ErrorAlert.jsx

- **Error de Validación:**
  - No verifica si message tiene un valor válido antes de renderizarlo.
- **Tareas Pendientes:**
  - Agregar un botón de cerrar para que el usuario pueda descartar la alerta.

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

App.jsx

- **Error de Validación:**
  - No se valida si el nombre de la ciudad es razonable antes de buscar (por ejemplo, longitud mínima).
  - Si el input está vacío, solo se muestra un mensaje en consola en lugar de informarle al usuario.
- **Tareas Pendientes:**
  - Agregar un estado de carga (loading) para indicar que la petición está en progreso.
  - Mostrar un mensaje o placeholder si el usuario aún no ha realizado ninguna búsqueda.
  - Mejorar el manejo de errores largos que podrían romper el diseño.

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

main.jsx

- **Errores en las importaciones:**
  - La importación de react-dom/client está escrita incorrectamente como react-dom.
  - La ruta del archivo principal App se cambió a ./application.
  - La ruta de los estilos de Bootstrap se modificó a una incorrecta: bootstrap/dist/bootstrap.css.
  - El archivo de estilos personalizado styles.css tiene una ruta mal escrita: ./style.css.
- **Tareas pendientes:**
  - Agregar un manejo de errores global para capturar cualquier problema que ocurra al cargar la aplicación.
  - Implementar un mensaje o spinner de carga inicial que se muestre mientras se renderiza el componente App.

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

useWeatherApi.js

- **Errores en las variables de entorno:**
  - El nombre de la variable API_KEY.
- **Estado de carga faltante:**
  - No se maneja un estado de carga (loading), lo que puede confundir al usuario si tarda la respuesta de la API.
- **Errores en la validación de la respuesta:**
  - No se valida si los datos recibidos de la API contienen los campos esperados (data.name, data.main, etc.).

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

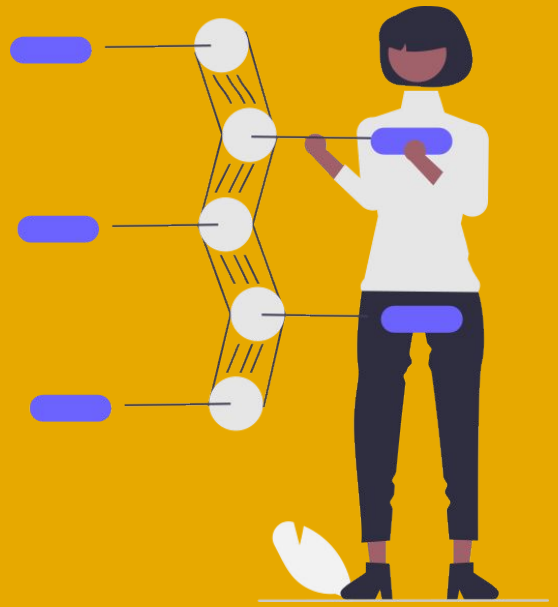
useWeatherApi.js

- **Error de manejo de excepciones:**
  - Se lanza un mensaje genérico para todos los errores (City not found), en lugar de diferenciar entre un error de red, una ciudad no encontrada o una clave de API inválida.
- **Tareas pendientes:**
  - Agregar un estado loading para mostrar un spinner mientras la API está en proceso.
  - Validar los datos recibidos de la API antes de usarlos.
  - Manejar errores específicos según el código de respuesta HTTP (401, 404, 500, etc.).

Ejercicio Propuesto: Solucion a problema de Aplicación de Clima

El proyecto tiene una versión que funcionaba, pero sin ninguna de las mejoras planteadas, revisalo y comparalo con lo que tú desarrollaste
https://github.com/adalid-cl/ESPECIALIZACION_FRONTEND_M5_AE4-2

Resumen de lo aprendido 🧑‍🎓



Adalid aspasia

Resumen de lo aprendido

- Aprendiste qué son los Hooks en React, cómo gestionar el estado con useState y manejar efectos secundarios con useEffect.
- Implementaste Hooks personalizados para reutilizar lógica común y optimizar el desarrollo en aplicaciones React.
- Descubriste cómo manejar y controlar errores en React, garantizando una experiencia de usuario fluida y segura.
- Aplicaste las reglas y mejores prácticas de los Hooks para mantener un código limpio, eficiente y sin errores comunes.

GRACIAS POR TU ATENCIÓN

Nos vemos en la próxima clase

Chile
Adalid
aspasia
LA FORMACIÓN DE TU FUTURO