Reglas de estilo de codificación

1. **Convenciones de Nomenclatura**:
    - Utilizar camelCase para nombres de variables y funciones, comenzando con minúscula.
    - Comenzar los nombres de los componentes con una letra mayúscula y usar PascalCase.
        - Ejemplo: `InputField`, `SelectField`

2. **Indentación**:
    - Mantener una indentación consistente usando espacios.

3. **Ubicación de Llaves**:
    - Colocar las llaves de apertura en la misma línea que la declaración de la función o estructura de control.
        - Ejemplo: `const InputField = ({ label, name, type, min, value, onChange, error }) => {`

4. **Longitud de Líneas**:
    - Mantener las líneas dentro de una longitud razonable, típicamente alrededor de 80-120 caracteres.

5. **Patrones de Nomenclatura**:
    - Usar nombres descriptivos para variables y funciones que comuniquen su propósito.
        - Ejemplo: `handleImageUpload`, `handleFileChange`

6. **Manejo de Errores**:
   Implementar manejo de errores para conexiones a la base de datos y solicitudes a la API.

7. **Mensajes de Respuesta**:
    - Utilizar códigos de estado HTTP apropiados y mensajes de respuesta claros en las rutas.
        - Ejemplo: `res.status(500).json({ error: 'Error al obtener los tipos de cargas' })`

8. **Otras Consideraciones**:
    - Utilizar hooks de React para el manejo del estado en componentes funcionales.

- Implementar renderizado condicional para mostrar u ocultar elementos según el estado o las props.
- Agregar comentarios para explicar la lógica compleja o las secciones de código no obvias mejora la capacidad de mantenimiento.