SEBASTIAN PATRICIO SOTO OYARCE

TNS. Informática mención ciberseguridad

calidad De software

Profesora Paulina González

**Informe de Calidad de Software**

**Introducción**

En el presente documento, se analizará el proyecto de la aplicación web Pokedex desarrollada con React, evaluando diversos aspectos según los requerimientos.

1. **Implementación de Linter (Standard)**

**Instalación de Standard**

Para implementar un linter en el proyecto, utilicé Standard, una herramienta que ayuda a mantener un estilo de código consistente y sin configuración. Los pasos de instalación fueron:

1. Instalación global mediante npm install standard -D

|  |
| --- |
|  |

1. Configuración de Script en package.json

|  |
| --- |
|  |

**Características de Standard**

No requiere configuración adicional, Aplica reglas de estilo de manera automática, Detecta y corrige problemas comunes de estilo de código React

**2.- Explicar 1 requerimientos funcionales, fundamente explicando su código**

**Descripción del Requerimiento**

El sistema debe permitir filtrar Pokémones por tipo.

En el componente MiApi.jsx, se implementa la funcionalidad de filtrado:

|  |
| --- |
|  |

La función handleTypeChange recibe un tipo de Pokémon si no se selecciona ningún tipo, muestra todos los Pokémones y si se selecciona un tipo, filtra los Pokémones que contengan ese tipo seleccionado ejemplo todos los Pokemones tipo fuego.

**3.- Explicar 1 requerimientos no funcionales, fundamente explicando su código.**

**Descripción del Requerimiento**

**Requerimiento**: La aplicación debe cargar los datos de los Pokémones de manera eficiente.

En el componente MiApi.jsx, se implementa una estrategia de carga eficiente de datos:

|  |
| --- |
|  |

Se utiliza Promise.all() para cargar múltiples datos en paralelo, mejora de la eficiencia al cargar datos de manera concurrente y también permite monitorear y optimizar el rendimiento al cargar datos, garantizando una carga de datos rapída y cumpliendo con el requerimiento no funcional.

**4.- Evidencie que sabe utilizar GIT**

**(subiré el informe a git)**

**5.- Analizar 1 aspecto de su proyecto en Owasp, ISO 27001, ITIL o CobitAnálisis de Seguridad**

El proyecto utiliza OWASP para asegurar la aplicación y prevenir vulnerabilidades, implementando las siguientes medidas:

**Análisis de Seguridad y Evaluación de Vulnerabilidades:**

**Inyección de Código:** Valida entradas en componentes para prevenir inyecciones maliciosas.

**Exposición de Datos:** Verifica el uso seguro de la API (PokeAPI) con conexiones HTTPS y asegura que no se exponga información sensible.

**Gestión de Errores:** Controla excepciones con registro seguro de errores sin exponer detalles críticos al usuario.

**Principios OWASP Aplicados:**

Control de acceso, prevención de inyecciones de código y manejo seguro de datos y gestión segura de errores.