## Preguntas

¿Cómo se hace el manejo de un error en NodeJs?

Describa cómo es la creación de un error personalizado en NodeJs

¿Qué tipos o clases de errores existen en NodeJs?

## Ejercicio

En base a la investigación realizada sobre los errores en NodeJs y su manejo, implemente en su proyecto la captura y flujo de los errores que este pueda presentar en la ejecución, esto con el fin de minimizar fallas en el sistema y/o vulnerabilidades, **el entregable para este ejercicio son 2 proyectos, uno con el código fuente sin el manejo de los errores y otro con la implementación**.

**Desarrollo**

1. Para el manejo de errores, Node js nos ofrece una variedad de alternativas las cuales podremos utilizar de acuerdo al contexto en el que este se presente. La primera alternativa que tenemos es el uso del bloque try catch, esté normalmente lo utilizamos cuando queremos capturar errores de manera síncrona, por otro lado para funciones asíncronas generalmente vamos a utilizar funciones callback, cabe aclarar que aunque la mayoría de las funciones asíncronas en node js pueden recibir como parámetro una función callback para el manejo de errores no todas lo hacen. Otra alternativa que poseemos es el manejo de errores mediante “promesas”, la cual nos permite manejar errores para varias funciones a la vez y tiene la particularidad que nos da la posibilidad del manejo de errores de manera síncrona en funciones asíncronas. Otra forma que podemos implementar para el manejo de errores es la creación un evento que se dispare en el momento en el que un determinado error suceda, este método se utiliza en casos complejos en que las formas anteriores no sean del todo óptimos para el manejo de los errores; y por último podemos crear nuestros propios errores personalizados según nuestras necesidades.
2. Para la crear errores personalizados en node js debemos crear una clase que que herede de la clase error de node js:

Class miError extends Error{

constructor(mensaje){

super(message);

}

}

A esta clase podemos darle las características necesarias para el manejo de errores en nuestro código. Para lanzar este error debemos crear un objeto de esta clase escribiendo la siguiente instrucción:

throw new miError(‘Esto es un nuevo error’);

1. En node js encontramos 4 categorías de errores:

* Los errores estándares de javascript.
* Errores de sistema que puedan afectar el flujo del código, como podría ser la pérdida o inexistencia de un archivo necesario para el programa.
* Errores de los usuarios al utilizar la aplicación, como por ejemplo, ingresar un valor no válido.
* Assertion Errors, es una clase especial de error el cual es lanzado al detectar algo fuera de lo común en la lógica del código.