



UNIVERSIDAD DE  
GUADALAJARA  
Red Universitaria e Institución Benemérita de Jalisco

# Centro Universitario de Ciencias Exactas e Ingenierías

---

## Computación tolerante a fallas

Profesor: López Franco Michel

Emanuel

Sección: D06

**Herramientas para el manejo de errores**

**Alumno:**

**Botello Martínez Nadia  
Noemi**

28 Enero 2024

1. Introducción	3
2. Desarrollo	3
3. Conclusión	5
4. Bibliografía	6

## 1. Introducción

*El manejo de errores* es el proceso de responder y recuperarse de situaciones de error en tu programa. En otras palabras, es un proceso que consiste en expectativas. Detectar y resolver diversos tipos de errores en aplicaciones y/o programas.

Existen muchos desafíos de diseño al considerar las técnicas de manejo de errores.

El objetivo principal del proceso de revisión del código es evaluar cualquier nuevo código para detectar errores, fallos y normas de calidad establecidas por la organización

## 2. Contenido

Los lenguajes de programación modernos y no tan modernos se diseñan o mejoran en contraste con muchas utilidades pueden realmente automatizar procesos, haciéndolos aún más seguros o incluso procesos alternativos. Aquí también se han realizado muchas mejoras en el tratamiento.

Las herramientas o el software de gestión de errores son para ayudarnos a servir como desarrolladores, Identificar, rastrear y corregir el origen del error. Estas herramientas pueden ser muy proyectos grandes o complejos porque nos ayudamos a administrar y determinar la prioridad de error y seguimiento, su origen y progreso. Teniendo esto en cuenta, al menos deberían estar presentes herramientas de gestión de errores.

Una de las siguientes características: registrabilidad (capacidad de identificar y representar), trazabilidad (capacidad Seguimiento), priorización (severidad de la señal) y gestión (selección de tareas).

### *Ejemplos de herramientas para el manejo de errores:*

- Visual Studio Code:

Es un editor de código que cuenta con una extensa cantidad de plugins que hacen del proceso de desarrollo un entorno más cómodo, con este podemos tener herramientas de detección de errores pre-runtime, así como análisis léxico, sintáctico y semántica de la mayoría de los lenguajes en el mundo, por lo que podríamos considerarlo como una herramienta muy potente.

- LSP :

LSP (Language Server Protocol) es un servicio, creado por Microsoft, que agrega características como auto completado, ir a las definiciones, documentación, referencias y más, sobre el curso en el proceso de programación, funcionando como un protocolo de

comunicación entre el IDE o editor y un servidor de lenguaje, lo que nos proporciona retroalimentación en tiempo real de errores, definiciones incorrectas, y más.

- Jira:

Es un software para la gestión de proyectos, en esta se puede generar un estado de aplicación y realizar un seguimiento, asignación, priorización y gestión de los errores encontrados en nuestro software.

*Estos son los beneficios de las herramientas de control de errores:*

- a) Proporciona una interfaz simple y común para compartir archivos y comunicar errores.
- b) Proporciona notificaciones y registros para ayudar al equipo a seguir, controlar y estimar el trabajo relacionado con errores.
- c) Proporciona una base de datos de errores que el equipo de desarrollo pueda usar como referencia en el futuro.
- d) Automatiza las tareas manuales asociadas con la captura de errores y la actualización de problemas

### 3. Conclusión

En general, se puede concluir que el manejo de errores implica la identificación, captura y resolución de situaciones inesperadas que pueden ocurrir durante la ejecución del programa. Al implementar estrategias adecuadas, como el uso de excepciones, mensajes de error claros y registros detallados, los problemas se pueden detectar y corregir más fácilmente. Además, el manejo de errores no consiste solo en resolver los problemas cuando ocurren, sino también en prevenir que ocurran mediante una planificación cuidadosa, pruebas exhaustivas y una validación adecuada de la entrada de datos.

## 4. Bibliografía

1. Villa, J. P. (n.d.). Errors and exceptions in Haskell. Stackbuilders.com. Retrieved August 28, 2023, from <https://www.stackbuilders.com/blog/errors-and-exceptions-in-haskell/>
2. *IBM documentation*. (s. f.). <https://www.ibm.com/docs/es/integration-bus/10.0?topic=extension-errors-exception-handling>
3. Official page for Language Server Protocol. (n.d.). Github.Io. Retrieved August 27, 2023, from <https://microsoft.github.io/language-server-protocol/>
4. Gunnell, M., Haqshanas, R., & Cooling, S. (2017, May 1). Error Handling. Techopedia. <https://www.techopedia.com/definition/16626/error-handling>