

Materia: Computación Tolerante a Fallas.

NRC: 179961

Maestra: Lopez Franco, Michael Emanuel

Aula: X-02

Sección: D06

Alumno: Zashuvath López Moreno, Ethan Israel

Código: 216493953



Índice

Introducción	3
Objetivo:	3
Contenido.....	3
Herramientas:.....	3
Técnicas.....	4
Ejemplo	4
Conclusión	5
Bibliografía.....	5

Introducción

Se hará una investigación sobre métodos y herramientas para poder solucionar problemas en el área de la programación para con alguna de estas elaborar en la segunda parte un código de nuestra elección.

Objetivo:

Encontrar herramientas y métodos para la solución de problemas de programación.

Contenido

Herramientas:

Depuradores o Debuggers: Estas herramientas nos permiten ejecutar un código paso a paso, inspeccionar variables y valores en tiempo real, así como detectar errores mediante la identificación de problemas en la ejecución.

Ejemplos:

- GNU Debugger.
- Visual DuxDebugger.
- Winpdb.
- Xcode Debugger.

Linters: Son herramientas que analizan el código en búsqueda de errores, convecciones de estilo y posibles errores con la condición de mantener el código coherente y limpio.

Ejemplos:

- ESLint.
- Pylint.

Monitores de rendimiento: Es un tipo de herramienta que analiza el rendimiento de la aplicación en tiempo real para identificar áreas que necesitan optimización o bien cuellos de botella.

- New Relic.
- AppDynamics.

Sistemas de seguimiento de problemas: Existen plataformas que permiten a los desarrolladores hacer un seguimiento de errores reportados, asignar grupos de trabajos a los mismos y estar pendientes hasta su correcta resolución.

- GitHub Issues
- GitLab Issues
- JIRA

Técnicas

Try-Catch: Estos son utilizados en leguajes como Java, C#, Python, JavaScript entre otros, este cuenta con el propósito de detectar y controlar una excepción generada por código en funcionamiento.

Validación de entrada: Incorporada en el código a través de comprobaciones de entrada y saneamiento, el objetivo principal de la validación de entrada es prevenir errores, vulnerabilidades y comportamientos inesperados que podrían surgir debido a datos incorrectos o maliciosos.

Logging: Bibliotecas como Log4j para Java o Winston para JavaScript ofrecen capacidad de registro avanzado, la función principal es registrar eventos, información de alta relevancia y errores durante la ejecución del código.

Control de versiones: Git es el sistema de control de versiones más utilizado en todo el mundo, la función de esta técnica es rastrar y administrar los cambios en el código fuente a lo largo del tiempo.

Ejemplo

Programa de divisiones de números enteros aleatorios que utiliza en “Logging” para crear un archivo “.txt” en el que muestra los datos recopilados durante la ejecución del programa y marcar las fallas en la misma o en los datos.

```
D: > DATOS > DOCUMENTOS > Pruebas > import random.py
1  import random
2  import logging
3
4  logging.basicConfig(filename='divisiones.log', level=logging.INFO, format='%(asctime)s - %(levelname)s - %(message)s')
5
6  def dividir(a, b):
7      try:
8          resultado = a / b
9          return resultado
10     except ZeroDivisionError as e:
11         logging.error("División entre cero: %s", e)
12         return None
13
14  valores = []
15
16  for _ in range(50):
17      a = random.randint(1, 100)
18      b = random.randint(0, 10)
19      valores.append((a, b))
20
21  for a, b in valores:
22      resultado = dividir(a, b)
23      if resultado is not None:
24          logging.info("División de %d entre %d es igual a %.2f", a, b, resultado)
```

Conclusión

Las herramientas y técnicas en la programación son fundamentales para crear un software confiable ya que ayudan a prevenir errores, optimizar el rendimiento y mejorar la seguridad del programa, por esta razón la combinación de estas garantiza una menor posibilidad de error.

Bibliografía

- Aston, B. (2022). Las mejores herramientas de control errores para identificar, rastrear y solucionar problemas más rápido. The Digital Project Manager.
<https://thedigitalprojectmanager.com/es/tools/mejores-herramientas-control-errores/>
- BillWagner. (2023, 7 abril). Procedimiento para controlar una excepción mediante try y catch. Microsoft Learn. <https://learn.microsoft.com/es-es/dotnet/csharp/fundamentals/exceptions/how-to-handle-an-exception-using-try-catch>
- colaboradores de Wikipedia. (2022). Depurador. Wikipedia, la enciclopedia libre.
<https://es.wikipedia.org/wiki/Depurador>
- Daityari, S. (2023). Las 12 mejores herramientas de revisión de código para desarrolladores (edición 2023). Kinsta®.
<https://kinsta.com/es/blog/herramientas-de-revision-de-codigo/>
- Diaz, G. A. (2021, 14 diciembre). Código de alta calidad en Python: Linters | Medium. Medium. <https://medium.com/@gonzaloandres.diaz/escribiendo-codigo-de-alta-calidad-en-python-parte-2-linters-64ffd8d2df91>

González, J. D. M. (2021). Errores al programar. www.programarya.com.

<https://www.programarya.com/Cursos/Fundamentacion/Errores>

Harsh, K. (2023). Guía definitiva para el manejo de errores en JavaScript. Kinsta®.

<https://kinsta.com/es/blog/errores-en-javascript/>

Hoyos, S. (2018, 26 marzo). Manten limpio tu código JavaScript usando Linters,

EditorConfig y Prettier. Medium. [https://medium.com/@simonhoyos/manten-](https://medium.com/@simonhoyos/manten-limpio-tu-c%C3%B3digo-javascript-usando-linters-editorconfig-y-prettier-25dad638b99)

[limpio-tu-c%C3%B3digo-javascript-usando-linters-editorconfig-y-prettier-](https://medium.com/@simonhoyos/manten-limpio-tu-c%C3%B3digo-javascript-usando-linters-editorconfig-y-prettier-25dad638b99)

[25dad638b99](https://medium.com/@simonhoyos/manten-limpio-tu-c%C3%B3digo-javascript-usando-linters-editorconfig-y-prettier-25dad638b99)

ManageEngine. (s. f.). Monitoreo de rendimiento de red | Software de monitoreo de rendimiento en redes - ManageEngine OPManager.

[https://www.manageengine.com/latam/network-monitoring/software-](https://www.manageengine.com/latam/network-monitoring/software-monitoreo-rendimiento-de-red.html)

[monitoreo-rendimiento-de-red.html](https://www.manageengine.com/latam/network-monitoring/software-monitoreo-rendimiento-de-red.html)

Mikejo. (2023, 13 junio). Herramientas y técnicas de depuración - Visual Studio

(Windows). Microsoft Learn. [https://learn.microsoft.com/es-](https://learn.microsoft.com/es-es/visualstudio/debugger/write-better-code-with-visual-studio?view=vs-2022)

[es/visualstudio/debugger/write-better-code-with-visual-studio?view=vs-2022](https://learn.microsoft.com/es-es/visualstudio/debugger/write-better-code-with-visual-studio?view=vs-2022)

Monitor de confiabilidad y rendimiento de un sistema Windows Server. (s. f.).

[https:// analisisyprogramacionoop.blogspot.com/2018/05/monitor-](https:// analisisyprogramacionoop.blogspot.com/2018/05/monitor-confiabilidad-rendimiento.html#:~:text=El%20Monitor%20de%20rendimiento%20permite, conjunto%20de%20recopiladores%20de%20datos.)

[confiabilidad-](https:// analisisyprogramacionoop.blogspot.com/2018/05/monitor-confiabilidad-rendimiento.html#:~:text=El%20Monitor%20de%20rendimiento%20permite, conjunto%20de%20recopiladores%20de%20datos.)

[rendimiento.html#:~:text=El%20Monitor%20de%20rendimiento%20permite,](https:// analisisyprogramacionoop.blogspot.com/2018/05/monitor-confiabilidad-rendimiento.html#:~:text=El%20Monitor%20de%20rendimiento%20permite, conjunto%20de%20recopiladores%20de%20datos.)

[conjunto%20de%20recopiladores%20de%20datos.](https:// analisisyprogramacionoop.blogspot.com/2018/05/monitor-confiabilidad-rendimiento.html#:~:text=El%20Monitor%20de%20rendimiento%20permite, conjunto%20de%20recopiladores%20de%20datos.)

Navis Code. (2020, 2 noviembre). Herramientas para buscar errores de programacion [Vídeo]. YouTube.

<https://www.youtube.com/watch?v=eQvbbANz-5c>

Ordenador, R. T. (2023, 14 junio). El arte de manejar errores en la programación: consejos y técnicas para principiantes. Reparar tu Ordenador.

<https://reparartuordenador.com/aprendiendo-a-manejar-errores-en-la-programacion>

Team, K. (2023, 10 mayo). ¿Qué es un Linter en programación? | KeepCoding Bootcamps. KeepCoding Bootcamps. <https://keepcoding.io/blog/que-es-un-linter-en-programacion/>