

UNIVERSIDAD DE GUADALAJARA



CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERIAS

Computación tolerante a fallas

Reporte de actividad 2

Nombre del alumno: Sunem Sandoval Gil
Profesor: Michel Emanuel López Franco
Título de la actividad: (Par.1) Otras herramientas para manejar errores
Fecha: 29 Agosto 2023

Instrucciones

Genera un reporte con otras herramientas para el manejo de errores en programación.

Desarrollo

El manejo de errores es una parte fundamental de cualquier programa, ya que ayuda a identificar y gestionar problemas que puedan surgir durante la ejecución. Cada lenguaje de programación tiene su propia sintaxis y enfoque para el manejo de errores, pero los conceptos generales son aplicables en la mayoría de los casos. Un buen manejo de errores puede hacer que tus programas sean más robustos y fáciles de mantener.

La revisión del código es una parte del proceso de desarrollo de software que implica la comprobación del código fuente para identificar los errores en una etapa temprana. Un proceso de revisión de código se lleva a cabo típicamente antes de fusionarse con el código base.

Una revisión efectiva del código evita que los errores y fallos se introduzcan en el proyecto, mejorando la calidad del código en una fase temprana del proceso de desarrollo del software.

Las herramientas de revisión de código estático más populares son:

1. **Review Board:** es una herramienta de código abierto basada en la web para la revisión de códigos. Para probar esta herramienta de revisión de código, puedes explorar la demostración en tu sitio web o descargar y configurar el software en tu servidor.
2. **Crucible:** es una herramienta de revisión de código en colaboración de Atlassian. Es un conjunto de herramientas comerciales que permite revisar el código, discutir cambios en los planes e identificar errores en una serie de sistemas de control de versiones.
3. **GitHub:** para mantener tus repositorios de Git en la nube, puede que ya hayas usado bifurcaciones y solicitudes de revisión de código. En caso de que no tengas ni idea de lo que es GitHub, aquí tienes una guía para principiantes sobre GitHub y las diferencias entre Git y GitHub.
4. **Phabricator:** es una lista de herramientas de código abierto de Phacility que te ayudan a revisar el código. Mientras que puedes descargar e instalar el conjunto de herramientas de revisión de código en tu servidor, Phacility también proporciona una versión en la nube de Phabricator.
5. **Collaborator:** de SmartBear es una herramienta de revisión de códigos y documentos para equipos de desarrollo. Además de la revisión del código fuente, Collaborator permite a los equipos revisar también los documentos de diseño.

Ejemplo Try and catch

Es un simple código para hacer una suma, pero utilizando el **try** y el **catch** en dado caso de que el usuario quisiera agregar alguna letra en vez de un número, mediante un **if** para imprimir el mensaje de error si llegara a fallar.

```
1  function test() {  
2    try {  
3      let number1 = Number(prompt("Ingrese un numero: "));  
4      let number2 = Number(prompt("Ingrese otro numero: "));  
5  
6      if (isNaN(number1) || isNaN(number2)) {  
7        throw "No son numeros";  
8      }  
9  
10     alert(`El resultado de la suma es: ${number1 + number2}`);  
11   } catch (error) {  
12     alert(`Error: ${error}`);  
13   }  
14 }  
15
```

Bibliografía

Shaumik Daityari. (2020, April 10). *Las 12 mejores herramientas de revisión de código para desarrolladores* (edición 2023). Kinsta®. <https://kinsta.com/es/blog/herramientas-de-revision-de-codigo/>