



CUCEI

Centro Universitario de Ciencias Exactas
e Ingenierías

TAREA 3 (P2)

Computación Tolerante a Fallas

Clave: I7036 | Sección: D06

Calendario: 2023B

Michel Emanuel López Franco

DIVTIC

— División de Tecnologías —
para la Integración Ciber-Humana

Cortés Pérez Leobardo Leonel

leobardo.cortes4751@alumnos.udg.mx

Departamento De Ciencias Computacionales

Principios de prevención de defectos

¿Qué es Orthogonal Defect Classification (ODC)?

ODC es un marco de trabajo para clasificar defectos de software en una forma que permita identificar las causas raíz de los defectos y tomar medidas para prevenirlos. ODC utiliza un conjunto de atributos ortogonales para clasificar los defectos, lo que significa que cada defecto puede clasificarse en varios atributos. Esto permite a los equipos de desarrollo de software obtener una visión más completa de los defectos en su software y tomar medidas más efectivas para prevenirlos.

Los atributos ortogonales utilizados en ODC son:

- **Fase de ciclo de vida:** La fase del ciclo de vida en la que se encontró el defecto.
- **Tipo de defecto:** El tipo de defecto, como un error de lógica, un error de interfaz o un error de rendimiento.
- **Gravedad:** La gravedad del defecto, como menor, moderada o mayor.
- **Prioridad:** La prioridad del defecto, como alta, media o baja.
- **Recursos necesarios:** Los recursos necesarios para corregir el defecto, como tiempo, esfuerzo o costo.
- **Orígenes:** El origen del defecto, como un error de desarrollador, un error de diseño o un error de usuario.

Cada atributo ortogonal proporciona información sobre un aspecto diferente del defecto. Por ejemplo, la fase de ciclo de vida proporciona información sobre cuándo se encontró el defecto. El tipo de defecto proporciona información sobre la naturaleza del defecto. La gravedad proporciona información sobre el impacto del defecto en el software. La prioridad proporciona información sobre la importancia de corregir el defecto. Los recursos necesarios proporcionan información sobre el costo de corregir el defecto. Los orígenes proporcionan información sobre la causa del defecto.

ODC se utiliza para clasificar los defectos de software de una variedad de fuentes, incluyendo informes de defectos, registros de pruebas y revisiones de código. Los defectos se clasifican en los atributos ortogonales utilizando un proceso manual o automatizado.

Una vez que los defectos se han clasificado en los atributos ortogonales, los equipos de desarrollo de software pueden utilizar esta información para identificar las causas raíz de los defectos. Por ejemplo, si los defectos se encuentran principalmente en la fase de diseño, los equipos de desarrollo pueden necesitar mejorar sus procesos de diseño. Si los defectos se encuentran principalmente en un tipo de defecto específico, los equipos de desarrollo pueden necesitar proporcionar más capacitación a los desarrolladores en ese tipo de defecto.

ODC es una herramienta valiosa para la prevención de defectos de software. Al proporcionar una visión más completa de los defectos, ODC ayuda a los equipos de desarrollo de software a identificar las causas raíz de los defectos y tomar medidas para prevenirlos.

Aquí hay algunos ejemplos específicos de cómo ODC puede ayudar a los equipos de desarrollo de software a prevenir defectos:

1. Si los defectos se encuentran principalmente en la fase de diseño, los equipos de desarrollo pueden necesitar mejorar sus procesos de diseño. Esto podría incluir el uso de herramientas de modelado de diseño o la realización de revisiones de diseño más frecuentes.
2. Si los defectos se encuentran principalmente en un tipo de defecto específico, los equipos de desarrollo pueden necesitar proporcionar más capacitación a los desarrolladores en ese tipo de defecto. Esto podría incluir la creación de cursos de capacitación o la realización de sesiones de entrenamiento.
3. Si los defectos se encuentran principalmente en un origen específico, los equipos de desarrollo pueden necesitar tomar medidas para abordar ese origen. Esto podría incluir la mejora de los procesos de desarrollo, la implementación de nuevas herramientas o la capacitación de los usuarios.
4. ODC es una herramienta poderosa que puede ayudar a los equipos de desarrollo de software a producir software de alta calidad. Sin embargo, es importante tener en cuenta que ODC no es una solución mágica. ODC es una herramienta que puede ayudar a los equipos de desarrollo de software a identificar las causas raíz de los defectos, pero es necesario tomar medidas para abordar esas causas raíz.

Un defecto pasa por DOS secciones con ODC

Sección de Apertura: Cuando encuentras un defecto, se pueden clasificar los siguientes atributos:

- Actividad: Esta es la actividad real que se realiza en el momento del descubrimiento del defecto. Por ejemplo, durante la fase de prueba de función, un ingeniero puede decidir realizar una inspección de código. La fase sería la prueba de función, pero la actividad es la inspección de código.
- Disparador: El entorno o condición que tuvo que existir para que el defecto apareciera. ¿Qué se necesita para reproducir el defecto? Durante las actividades de revisión e inspección, elija la selección que mejor describa lo que estaba pensando cuando descubrió el defecto. Para otros defectos, coincida la descripción con el entorno o condición que fue el catalizador de la falla.
- Impacto: Para los defectos en proceso, seleccione el impacto que cree que el defecto habría tenido en el cliente si hubiera escapado al campo. Para los defectos reportados en el campo, seleccione el impacto que tuvo la falla en el cliente.
- Sección de Cierre: cuando se sabe cómo se corrigió el defecto, se pueden clasificar los siguientes atributos:

Objetivo: Representa la identidad de alto nivel de la entidad que se corrigió.

- Tipo de defecto: Representa la naturaleza de la corrección real que se realizó.
- Cualificador (aplica al tipo de defecto): Captura el elemento de una implementación inexistente, incorrecta o irrelevante.
- Origen: Identifica el origen del Objetivo (es decir, Diseño/Código, ID, etc.) que tenía el defecto.
- Edad: Identifica la historia del Objetivo (es decir, Diseño/Código, ID, etc.) que tenía el defecto.