Un conjunto de letras blancas en un fondo blanco

Descripción generada automáticamente con confianza media

**Escuela superior politécnica de Chimborazo**

**Facultad de informática y electrónica**

**Carrera software**

Texto, Carta

Descripción generada automáticamente

**Escuela**

**Superior Politécnica**

**De Chimborazo**

**Docente:** Julio Santillán

**Asignatura:** Aplicaciones Informáticas II

|  |  |
| --- | --- |
| **Estudiante** | **Código** |
| * Nina Sinaluisa | 6727 |

**Modalidad:** Asincrónica (Individual)

**Tema: I**nforme de problemas (PR).

**Fecha de entrega:** 08 de abril de 2025

**RIOBAMBA - 2025**

**INFORME DE PROBLEMAS (PR)**

# RESUMEN

Coz Flores, M. (2023) menciona que al evaluar un impacto implica desarrollar una técnica de análisis de riesgo de impacto completo acerca de las funciones que se afectarán. El mantenedor debe tener un conocimiento amplio de la estructura de códigos y contenido del programa a manera evaluar los recursos necesarios, para implementar los cambios y ser consciente de los riesgos involucrados en la implementación de una modificación específica Un requerimiento de modificación se denomina solicitud de cambio (MR), y un informe de incidente se denomina (PR) estos términos van de la mano ya que cumplen funciones de la misma índole. Esto debe interpretarse a términos de lenguaje de programación.

Se describe las actividades que debe realizarse para el análisis de impacto:

* Dimensionar los cambios a planificar e implementar.
* Estimar, precisar los recursos imprescindibles para culminar la actividad.
* Evaluar de costo/beneficio de la modificación propuesta.
* Expertos, dialoguen acerca de la dificultad de la modificación.

También se menciona que en casos de que se detecten problemas durante la ejecución del desarrollo, operaciones y procesos de mantenimiento de la norma ISO/IEC 12207, estos se registran y son supervisados por el proceso de resolución de problemas en la norma ISO/IEC 12207. Estas solicitudes de modificación son conocidas como MRS o informes de problemas RP. Dentro de los Sistemas de Ingeniería ISO/IEC 15288, gracias al mantenimiento se asegura que el producto software continúa para satisfacer las necesidades de los usuarios.

# ANÁLISIS

El informe de problema es un término usado para identificar y describir problemas detectado en un software, el informe de problemas consiste en realizar un documento donde se detalle qué está fallando, dónde ocurre el fallo, cómo se puede reproducir, qué impacto tiene, cuándo sucedió en un contexto de problemas, errores o incidencias en un sistema software, para realizar el respectivo mantenimiento y de ese modo asegurarse de que no vuelve a producirse ese conflicto, Además es muy importante ya que se lleva un registro y es una solicitud para que todo el personal involucrado este al tanto o este en conocimiento del problema que esta sucediendo y pueda buscar soluciones desde su área ya sea software, hardware u operaciones.

# CASO DE ESTUDIO

Entre junio de 1985 y enero de 1987 operaba una máquina de radioterapia llamada Therac-25 que se utilizaba para tratar a pacientes con cáncer, pero debido a un mal funcionamiento de software terminó matando a 3 personas debido a cantidades extremadamente altas de radiación. La máquina utilizaba algunos de los mismos programas que su predecesor Therac-6, pero debido a que el productor quería reducir los costes que no incluía los mecanismos de seguridad a nivel de hardware que se habían utilizado anteriormente.

El control de la máquina fue realizado mediante software con el aporte de los usuarios y por las fallas en el software esto conllevó en pacientes recibiendo una dosis más alta o varias dosis porque mostraba al usuario y operador que no se había dado ninguna dosis.

El software de Therac-25 conllevaba una mayor responsabilidad en el mantenimiento para la seguridad que el software de máquinas anteriores. Si los mecanismos de seguridad a nivel de hardware hubiesen estado presentes se habría impedido que tres personas perdieran la vida.

**Informe**

**Descripción del problema**

El sistema de control de dosis del Therac-25 presentó múltiples fallas críticas de software que provocaron la administración de **dosis letales de radiación a pacientes**, debido a errores no detectados ni prevenidos por el sistema. El software indicaba que no se había administrado ninguna dosis cuando en realidad se habían aplicado dosis 100 veces superiores a lo permitido.

**Pasos para reproducir el problema**

1. El operador ingresaba los datos del tratamiento.
2. Por la rapidez al modificar valores, el sistema entraba en un estado de error debido a condiciones de carrera.
3. El software fallaba al detectar el cambio, pero igualmente ejecutaba la orden con un valor incorrecto.
4. La interfaz mostraba que no se aplicó radiación.
5. Se administraba una nueva dosis, acumulando una radiación excesiva sin advertencia.

**Resultado esperado**

Que el sistema no permita la aplicación de una nueva dosis sin verificar la administración real anterior, y que exista una verificación cruzada entre el estado del software y el hardware.

**Resultado observado**

El software permitía múltiples aplicaciones de dosis sin verificar efectivamente si ya se había aplicado una, debido a fallos en la detección del estado interno. No existía ningún mecanismo de protección físico o lógico adicional.

**Impacto**

* Muerte de al menos **3 personas**.
* Múltiples lesiones graves permanentes.
* Juicios legales y pérdida de reputación del fabricante.
* Fallos éticos y técnicos en el desarrollo y mantenimiento del sistema.

**Causa raíz**

* Falta de mecanismos de seguridad a nivel de hardware.
* Dependencia total del software para funciones críticas.
* Condiciones de carrera no gestionadas.
* Reutilización inapropiada de código antiguo sin rediseño.
* Ausencia de pruebas adecuadas para casos extremos.

**Acciones correctivas sugeridas**

* Incorporar doble verificación físico-lógica (hardware + software).
* Rediseñar arquitectura del sistema con redundancia y validaciones cruzadas.
* Realizar pruebas exhaustivas de estrés y seguridad.
* Aplicar normas de desarrollo de software para sistemas críticos (por ejemplo, ISO 26262, IEC 62304, DO-178C, etc.).
* Capacitación ética y técnica continua a ingenieros de software.

**Diseño de software más correcciones.**

**Operador**

* Monitoreo de cambios de entrada y edición de un operador
* Actualización de la pantalla para mostrar el estado actual de la máquina
* Impresión en respuesta a comandos del operador

**Máquina**

* Monitoreo del estado de la máquina
* colocación del plato giratorio
* resistencia y forma de la viga
* funcionamiento de los imanes de flexión y escaneo
* Configuración de la máquina para el tratamiento especificado
* Encendiendo la luz
* Apagar el haz (después del tratamiento, por orden del operador o si se detecta un mal funcionamiento)

El software Therac-25 está diseñado como un sistema en tiempo real e implementado en lenguaje máquina (un lenguaje de bajo nivel y difícil de leer). El software dividió las tareas mencionadas en críticas (p. ej., configuración y operación del haz) y no críticas (p. ej., supervisión del teclado). Un programador gestionó la asignación de tiempo de computadora a todos los procesos, excepto a aquellos gestionados por interrupción (p. ej., el reloj de la computadora y la gestión de errores generados por el hardware).

1. **BILIOGRAFÍA**

* Coz Flores, M. (2023). MANTENIMIENTO DE SISTEMAS PLANIFICACIÓN DE PROYECTOS DE SOFTWARE. PLANIFICACIÓN Y GESTIÓN DE MANTENIMIENTO DE SOFTWARE. APLICACIONES ESTÁNDARES. DOCUMENTACIÓN DEL CENTRO DE PROCESAMIENTO DEL MEDIO FISICO. MANTENIMIENTO DEL MEDIO LÓGICO. NORMAS ESTÁNDARES. APLICACIONES.
* Herrera Caldas, V. A. (2015). *Desarrollo de un plan de gestión de mantenimiento de software para el Departamento de Sistemas de la Universidad Politécnica Salesiana basado en la Norma ISO/IEC 14764: 2006* (Bachelor's thesis).}
* *Therac-25 case narrative*. (s/f). Onlineethics.org. Recuperado el 9 de abril de 2025, de <https://onlineethics.org/cases/therac-25/therac-25-case-narrative>