

Actividad– Técnicas de Caja Blanca

N° Proyecto 12345

Requerimiento /
Reportado Por

Teresa Tapia Soto

Nombre
Proyecto Tipos de Pruebas de Software

Fecha Requerimiento

Actividad N° 10: Revisión y análisis de código estático en un ejemplo práctico

Objetivo:

- Realizar prácticas de pruebas de software utilizando técnicas de caja blanca

Aprendizaje Esperado:

2.5.- Describe los conceptos asociados a la fase de pruebas de Software en el ciclo de vida de software.

Criterios de Evaluación

- 2.5.1.- Identifica los tipos de prueba en la etapa de testing y las condiciones para su implementación.
- 2.5.2.- Distingue los aspectos claves para la elaboración de un plan de pruebas de acuerdo a técnicas y normas.
- 2.5.3.- Describe los tipos de prueba a realizar en un plan de pruebas.
- 2.5.4.- Establece criterios de aceptación y protocolo en el desarrollo de las pruebas.

Actividades:

1. Aplicar revisión de código estático a un programa de software conocido como lo es el juego buscaminas realizado en Java o en algún programa de ejemplo para realizar la práctica. Descargar el programa buscaminas del material de la asignatura.
2. Realizar la revisión de código de dos formas:
 - a. Utilizar el plugin del PMD incorporado en el IDE NetBeans
 - b. Utilizar el Jar del PMD utilizando consola DOS. Descargar el PMD desde la página de PMD <https://pmd.github.io/>
3. Obtener los resultados de la revisión de código en un archivo de salida en formato txt.
4. Importar los resultados de la revisión en un archivo Excel para poder realizar entendimiento del análisis realizado.
5. Mostrar al docente la ejecución de la revisión de código en las maneras solicitadas.
6. Enviar al correo del docente (teresa.tapia02@inacapmail.cl):
 - a. Línea de instrucción de ejecución del PMD utilizada para realizar la revisión de código.
 - b. Archivo de resultado de la revisión de código en formato txt y excel.
 - c. Programa comentado
 - d. Documento con el paso a paso de la actividad.
7. Revisar junto al docente la actividad realizada.



Conceptos Básicos

PMD es una herramienta de auditoría y verificación de código desarrollada por SourceForge.

PMD escanea el código Java y busca posibles problemas en potencia como pueden ser:


- Posibles bugs (sentencias *try/catch/finally/switch* vacías)
- Código muerto (variable locales, parámetros y métodos privados que no se usan)
- Código no óptimo (derroches en el uso de String/StringBuffer)
- Expresiones excesivamente complicadas (sentencias *if* innecesarias, implementaciones con bucles *while*)
- Código duplicado (código copiar-pegar que significa errores copiados-pegados)

PMD proporciona un amplio conjunto de reglas de verificación de código en base a una serie de pautas para la aplicación de buenas técnicas de implementación. Normalmente será suficiente con los conjuntos de reglas aportados por la propia herramienta pero existe la posibilidad de creación de reglas de validación personalizadas.

PMD puede instalarse en servidor para ser ejecutado desde tareas ant, o como un plug-in de Eclipse en PC local u otras herramientas como NETbeans.

Para obtener información adicional sobre el producto acceder a su página Web:

<http://pmd.sourceforge.net/>

UNIVERSIDAD TECNOLÓGICA DE CHILE INSTITUTO PROFESIONAL CENTRO DE FORMACIÓN TÉCNICA 		
		Version N°:1.0
		Página 2

<https://pmd.github.io/>

Nota: Las versiones de PMD han ido modificando las instrucciones de ejecución para realizar las revisiones de código. Un claro ejemplo que ejecución de la versión de PMD 4.3 era:

```
c:\pmd-4.2.3\bin>pmd C:\Fuentes\programaEjemplo text basic,unusedcode,design,logging- java,finalizers,strings,j2ee,imports -reportfile
c:\fuentes\resultado.txt -target jdk 1.4
```

Investiga ¿cuál es la instrucción para realizar la revisión de código con la versión más actualizada de PMD?