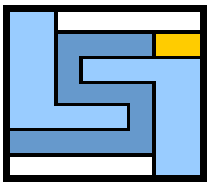
Diseño y Pruebas II



LINT REPORT

- **Cover**

Group: E7-06
Repository: https://github.com/plss12/Acme-Toolkits
Colleagues:

1. Cabezas Villalba, Juan Pablo (juacabvil@alum.us.es)
2. Martínez Jaén, Javier (javmarjae@alum.us.es)
3. Moreno Calderón, Álvaro (alvmorcal1@alum.us.es)
4. Navarro Rodríguez, Julio (julnavrod@alum.us.es)
5. Parejo Ramos, Salvador (salparram@alum.us.es)
6. Soto Santos, Pedro Luis (pedsotsan@alum.us.es)

Date: 23/04/2022

- **Revision table**

Version	Date	Description
1.0	23/04/2022	First Revision

- **Table of contents**

- Executive summary

El contenido de este informe describe los bugs y malos olores detectados por SonarLint en el código del *deliverable* 3.
Se comenta brevemente de qué tipo han sido esos bugs/malos olores y si se han conseguido o no solucionar, además de la manera en que se han solucionado.

- Introduction

En este documento mostraremos los resultados del análisis de la extensión de SonarLint, un plug-in de eclipse el cual ayuda a identificar fallos y malas prácticas en el código del proyecto. Mostraremos los resultados del análisis y explicaremos los que hemos solucionado gracias a este análisis y los que se van a dejar, y la razón por la que se van a mantener.

- Contents

Tras completar todas las task las cuales implementan código, utilizamos SonarLint para realizar el primer análisis de nuestro proyecto, los resultados fueron 26 malos olores y 11 bugs, la mayoría eran el mismo solo que aparecían en varios sitios.

- Bugs:
  Estos se encontraban principalmente en el archivo .jsp de administrator dashboard. Al tener que usar una serie de tablas para que visualmente fuera correcta la página, había algunas tablas que no contenían etiquetas <th> o <tr>. Esto podría resultar en un fallo grave más adelante, así que se corrigió.

  También encontramos bugs menores como el no usar scope dentro de un <th>, simplemente se le añadió scope=”row” a los <th> para solventar ese fallo.

Un último tipo de bug ocurrió en los *Map* que se usaban para el administrator dashboard, SonarLint nos recomendó que al contener esos *Map* un enumerado y un *Double* o *Long,* usáramos mejor un *EnumMap,* el cuál es perfecto cuando la key es un Enumerado, como era el caso.

- Malos Olores:
  El mal olor principal que encontramos fue en algunos archivos .jsp donde algunas de las tablas no tenían dentro la etiqueta *<caption>*, etiqueta la cuál añade una "descripción" a la tabla. Se añadieron para solventar este fallo.

Como se ha podido observar, todos los bugs y malos olores anteriores se decidieron arreglar cuanto antes para así evitar la deuda técnica, ya que nos suponía un esfuerzo leve el arreglarlos y nos podría ahorrar muchos problemas en el futuro desarrollo de la aplicación.

Tras el arreglo de las malas prácticas identificadas en el primer análisis, volvimos a lanzar un segundo análisis con SonarLint, obteniendo sus resultados y comprobando que el proyecto estaba libre de malos olores y bugs.

- Conclusions

SonaLint es una extensión IDE gratuita y open-source que identifica y te ayuda a solucionar los problemas de calidad y seguridad mientras escribes el código, esto la hace una herramienta muy útil y esencial para cualquier proyecto software.

Gracias a esta extensión se pueden identificar pequeños fallos y malas prácticas en el código fuente, las cuales muchas veces surgen de la inexperiencia o del desconocimiento sobre si realmente estamos creando un bad smell mientras programamos, y al ser prácticas que normalmente no afectan directamente al funcionamiento de la aplicación, se mantienen durante el completo desarrollo de la aplicación, acumulando así deuda técnica la cual se va haciendo una bola cada vez más grande con la que chocaremos en algún momento del desarrollo. Por esto, es necesario un continuo análisis del código fuente para ir limpiando el proyecto y eliminando malos olores haciendo más llevadero el mantenimiento del proyecto en futuros problemas.

- Bibliography

Intencionadamente en blanco