## DP2 2021-2022

## Informe Lint

## Proyecto Acme Toolkits

https://github.com/Alex-GF/Acme-Toolkits.git

#### Miembros:

- VICENTE CAMBRÓN TOCADOS (viccamtoc@alum.us.es)
- FRANCISCO JAVIER CAVERO LÓPEZ (fracavlop@alum.us.es)
- IRENE XIANG DOMÍNGUEZ GAROZ (iredomgar4@alum.us.es)
- ALEJANDRO GARCÍA FERNÁNDEZ (alegarfer4@alum.us.es)
- JOSÉ LUIS GARCÍA MARÍN (josgarmar31@alum.us.es)
- MARTA REYES LÓPEZ (marreylop4@alum.us.es)

**GRUPO E1-03** 

Versión 1.0

24/04/22

# <u>Índice</u>

Índice	2
Resumen ejecutivo	3
Tabla de revisiones	3
Introducción	4
Contenido	4
LintReport	4
EnumMap	5
Acme.js	5
Areyousure.js	5
Bootstrap	5
Conclusión	5
Bibliografía	5

## Resumen ejecutivo

El objetivo de este documento es inspeccionar el código realizado y así encontrar errores y malos olores reportados por Lint con respecto al proyecto. Se solucionarán los malos olores reportados y en caso de ser un mal olor no real, se proporcionará una clara justificación de este.

## Tabla de revisiones

Fecha	Versión	Descripción	Entregable
24/04/2022	1.0	Versión inicial	3

## <u>Introducción</u>

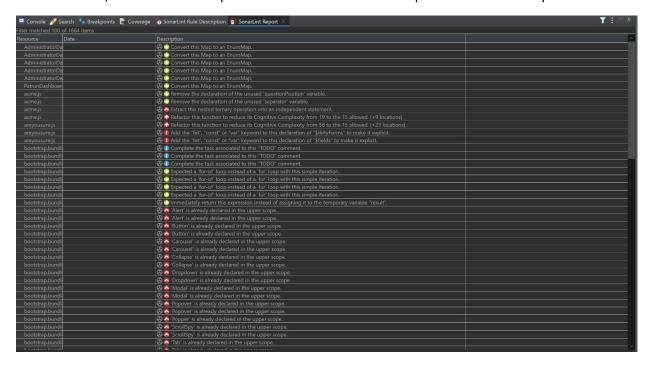
En este documento se enumeran y detallan todos los malos olores encontrados en el proyecto tras la realización de las distintas pruebas. Por tanto, una vez realizadas y pasado las pruebas se realizó con el SonarLint plug-in que nos proporciona Eclipse la detección de los malos olores.

Todos los malos olores reportados en el LintReport se solucionarán refactorizando hasta que deje de ser reportado por SonarLint. Si por cualquier caso nos reportase un error no real se proporcionará una justificación clara y definida del porqué de este.

### Contenido

### LintReport

A continuación, se muestra una captura con los distintos errores reportados en el LintReport:



#### EnumMap

Observamos que los primeros 6 malos olores reporta que debemos convertir los "Map" en "EnumMap" de la clase AdministratorDashboard. Sin embargo, por indicaciones del profesorado, dicho reporte no debemos solucionarlo. Por tanto, no debemos refactorizarlo ni arreglarlo.

#### Acme.js

Observamos que los siguientes 4 errores que continúan son de "acme.js" por lo que se debe al framework dado y no al código realizado. Por tanto, no debemos arreglarlo.

### Areyousure.js

Observamos que los siguientes 3 errores que continúan son de "areyousure.js" por lo que se debe a una librería de jQuery. Por tanto, no debemos arreglarlo.

### Bootstrap

Observamos que, a partir de aquí, todos los malos olores restantes se deben a la herramienta Bootstrap. Por tanto, no debemos refactorizarlo ni arreglarlo.

### Conclusión

Observamos que, al haber realizado un código muy estructurado y limpio, esta parte tan tediosa de refactorizar y arreglar errores y malos olores nos la hemos podido evitar, ahorrado así mucho tiempo y esfuerzo.

Es por ello por lo que, aunque inicialmente le dedicamos mucho tiempo al diagrama UML, teniéndolo que rehacer cantidad de veces, y la implementación de cada funcionalidad fue lenta y cuidadosa, hemos comprobado que ha resultado una decisión exitosa, pues al final hemos conseguido ahorrar tiempo y esfuerzo en tareas como estas.

### Bibliografía

No aplica.