DP2 2023-2024

Lint Report

Acme Software Factory

Repository: https://github.com/rafcasceb/Acme-SF-D03

Student #2:

- Flores de Francisco, Daniel danflode@alum.us.es

Other members:

- Castillo Cebolla, Rafael rafcasceb@alum.us.es
- Vento Conesa, Adriana adrvencon@alum.us.es
- Heras Pérez, Raúl rauherper@alum.us.es
- Mellado Díaz, Luis luimeldia@alum.us.es

GROUP C1.049

Version 1.1

23-04-24

Content Table

Abstract

This report offers a detailed examination of the lint analysis conducted for project deliverable D03. It presents a thorough evaluation of "bad smells" reported by Sonar Lint, justifying their innocuous nature if appropriate.

Revision Table

Date	Version	Description of the changes	Deliverable
23/04/2024	V1	• Abstract. • Introduction. • Contents: task detailing. • Conclusions.	3
24/04/2024	V1.1	• Added missing analysis.	3

Introduction

During this stage of delivery, attention was aimed at putting various features into action, resulting in the emergence of several unfavorable indicators along the way. The forthcoming examination in this report exclusively concerns the lint assessment of student five's portion, whereas the lint assessment for the collaborative group effort will cover evaluations of other sections.

The document layout is delineated as such: an initial segment contextualizing the lint analysis, succeeded by a thorough listing of unfavorable indicators identified by Sonar Lint within student five's section. Lastly, a brief conclusion will summarize the report, highlighting key discoveries.

Contents

Bad Smells

The subsequent undesirable characteristics manifested repeatedly in the individual files scrutinized for this report. To simplify matters, they have been amalgamated into a single listing.

1. Use concise character class syntax '\\d' instead of '[0-9]'.
   This bad smell is innocuous, as it performs the same functionalities as the alternative option proposed and does not hinder the maintainability of the code. The change is only more concise.

2. Rename this package name to match the regular expression '^[a-z_]+(\.[a-z_][a-z0-9_]*)*$'.
   As we follow the structure recommendations given by the Acme Jobs example project, this bad smell can be considered innocuous.

3. Replace this assert with a proper check.
   As we follow the Acme Framework recommendations given in the theory lectures, this bad smell can be considered innocuous.

4. Add explicit curly braces to avoid dangling else.
   This bad smell cannot be corrected, as the workspace automatically removes redundant curly braces in if-else structures. Nevertheless, it can be considered innocuous.

5. Override the "equals" method in this class.
   As we follow the Acme Framework recommendations given in the theory lectures, this bad smell can be considered innocuous.

6. Define a constant instead of duplicating this literal “X” times.
   This bad smell is innocuous by nature, as the duplication of literals does not directly affect the implementation of the features. It could potentially pose a refactoring problem, but as the recommendations and examples given by the subject also duplicate literals, we can safely ignore this.

Conclusions

In summary, this lint report document enables us to scrutinize the challenges encountered in analyzing and incorporating various requirements, along with a reassessment of our code to mitigate future issues with maintainability.

Bibliography

Intentionally blank.