|  |  |
| --- | --- |
|  | *Code Inspection Report*  *Anti-Spam Configuration Software Development Project*  BSc/MSc in [LEI | LIGE | METI]  Academic Year 2017/2018 - 1º Semester  Software Engineering I  Group Id 52  64750, João Teixeira, METIA1  64822, Rodrigo Cortesão, METIA1  65217, João Martins, METIA1  65297, Mariana Barros, METIA1  ISCTE-IUL, Instituto Universitário de Lisboa  1649-026 Lisbon  Portugal  November 25th 2017 |
|  |  |

**Table of Contents**

[Introduction 3](#_Toc498465002)

[Code inspection – Name of the component being inspected 3](#_Toc498465003)

[Code inspection checklist 3](#_Toc498465004)

[Found defects 3](#_Toc498465005)

[Corrective measures 3](#_Toc498465006)

[Conclusions of the inspection process 3](#_Toc498465007)

# Introduction

O sistema produzido é um Filtro Anti-Spam. Este tem o objetivo de conseguir separar os e-mails de SPAM daqueles que não são. Para isso são criadas algumas regras que facilitam este processo, onde é dado um peso a cada uma dessas regras. Caso o peso conjunto das regras de um e-mail ultrapasse o valor 5, esse e-mail é considerado SPAM.

Para uma maior precisão, neste projeto são calculados os falsos positivos (FP) e os falsos negativos (FN), que ajudam a equipa a perceber a quantidade de e-mails que não está a ser bem avaliada, permitindo assim ajustes.

Existem duas formas de fazer a configuração do filtro Anti-Spam: a manual, onde o próprio utilizador introduz os pesos a cada regra e a automática, onde é utilizado um algoritmo NSGAII que permite um resultado melhor e de uma forma mais rápida do que a configuração manual.

É possível ao utilizador guardar qualquer uma das configurações.

# Code inspection – antiSpamFilter

Nesta secção é feita a inspeção do package antiSpamFilter, onde estão todas as classes desenvolvidas no âmbito deste projeto (com exceção aos testes jUnit).

|  |  |
| --- | --- |
| *Meeting date:*  *Meeting duration:*  *Moderator:*  *Producer:*  *Inspector:*  *Recorder:* | *13/12/2017*  *120 minutes*  *Mariana Barros*  *João Martins*  *Rodrigo Cortesão*  *João Teixeira* |
| *Component name (Package/Class/Method):* | *antiSpamFilter* |
| *Component was compiled:* | *Yes* |
| *Component was executed:* | *Yes* |
| *Component was tested without errors:* | *Yes* |
| *Testing coverage achieved:* | *99%* |

# Code inspection checklist

The checklist for Java code inspection used in this project is available at <http://www.cs.toronto.edu/~sme/CSC444F/handouts/java_checklist.pdf> and in blackboard ES1 page.

# Found defects

Identify and describe found defects, opinions and suggestions.

|  |  |  |  |
| --- | --- | --- | --- |
| **Found defect Id** | **Package, Class, Method, Line** | **Defect category** | **Description** |
| 1 | *antiSpamFilter, ReadRules,lerRules, 17* | FD | Nome do metodo em duas línguas diferentes |
| 2 | *antiSpamFilter, Table\_object, getValor, 17* | VC | Nome do método confuso |
| 3 | antiSpamFilter, Table\_Model, 14,20 | CM | Comentários não necessários |
| 4 | antiSpamFilter, Table\_object, getValue, 13 | FD | Nome do método em duas linguagens diferentes |
| 5 | antiSpamFilter, Interface, 36,37 | VC | Nomes similares e confusos |
| 6 | antiSpamFilter, Interface, 43,44,45,46 | FD | Começava com letra maiuscula |

# Corrective measures

Os erros encontrados na inspeção do código realizada pelos membros do grupo, foram corrigidos logo após a reunião pelo grupo.

# Conclusions of the inspection process

Esta inspeção revelou alguns erros no código, sendo estes, no entanto, erros que não impedem de forma alguma o funcionamento do software desenvolvido nem a experiência do cliente. Foram erros corrigidos para facilitar a leitura/análise do código.