

# Quality reports

Requirements document

Date: 2013.04.05

feedzai<sup>↑</sup>

## AUTHORS AND CONTRIBUTORS

Code	Name	Contact
AK	Andrey Klimachev	andrey.klimachev@feedzai.com

## DOCUMENT APPROVAL

Version	Date	Approved By	Role

## DISTRIBUTION LIST

SVP Product Development

## COPYRIGHT

All rights reserved. No part of this document may be reproduced or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of FeedZai S.A.

## TABLE OF CONTENTS

1	Purpose.....	6
2	Introduction .....	6
	2.1 Overview .....	6
	2.2 Requirement Structure .....	6
	2.3 Requirements.....	6

## GLOSSARY

Acronym	Description

## TABLE OF TABLES

Table 1 – Requirement QA-RP -01 .....	7
Table 2 – Requirement QA-RP-02 .....	7
Table 3 – Requirement QA-RP-03 .....	7
Table 4 – Requirement QA-RP-04 .....	7
Table 5 – Requirement QA-RP-05 .....	8
Table 6 – Requirement QA-RP-06 .....	8
Table 7 – Requirement QA-RP-07 .....	8
Table 8 – Requirement QA-RP-08 .....	9
Table 9 – Requirement QA-RP-09 .....	9
Table 10 – Requirement QA-RP-10 .....	9
Table 11 – Requirement QA-RP-08 .....	9
Table 12 – Requirement QA-RP-12 .....	10
Table 13 – Requirement QA-RP-13 .....	10
Table 14 – Requirement QA-RP-14 .....	10
Table 15 – Requirement QA-RP-15 .....	11
Table 16 – Requirement QA-RP-16 .....	11
Table 17 – Requirement QA-RP-17 .....	11
Table 18 – Requirement QA-RP-18 .....	11
Table 19 – Requirement QA-RP-19 .....	12
Table 20 – Requirement QA-RP-20 .....	12
Table 21 – Requirement QA-RP-21 .....	12
Table 22 – Requirement QA-RP-22 .....	13
Table 23 – Requirement QA-RP-23 .....	13
Table 24 – Requirement QA-RP-24 .....	13
Table 25 – Requirement QA-RP-25 .....	13
Table 26 – Requirement QA-RP-26 .....	14
Table 27 – Requirement QA-RP-27 .....	14
Table 28 – Requirement QA-RP-28 .....	15
Table 29 – Requirement QA-RP-29 .....	15
Table 30 – Requirement QA-RP-30 .....	15
Table 31 – Requirement QA-RP-31 .....	16
Table 32 – Requirement QA-RP-32 .....	16

Table 33 – Requirement QA-RP-33 .....	16
Table 34 – Requirement QA-RP-34 .....	17
Table 35 – Requirement QA-RP-35 .....	17
Table 36 – Requirement QA-RP-36 .....	17
Table 37 – Requirement QA-RP-37 .....	17
Table 38 – Requirement QA-RP-38 .....	18
Table 39 – Requirement QA-RP-39 .....	18
Table 40 – Requirement QA-RP-40 .....	18
Table 41 – Requirement QA-RP-41 .....	19
Table 42 – Requirement QA-RP-42 .....	19
Table 43 – Requirement QA-RP-43 .....	19
Table 44 – Requirement QA-RP-44 .....	19
Table 45 – Requirement QA-RP-45 .....	20
Table 46 – Requirement QA-RP-46 .....	20
Table 47 – Requirement QA-RP-47 .....	20
Table 48 – Requirement QA-RP-48 .....	21
Table 49 – Requirement QA-RP-49 .....	21
Table 50 – Requirement QA-RP-50 .....	21
Table 51 – Requirement QA-RP-51 .....	21
Table 52 – Requirement QA-RP-52 .....	22
Table 53 – Requirement QA-RP-53 .....	22
Table 54 – Requirement QA-RP-54 .....	23
Table 55 – Requirement QA-RP-55 .....	23
Table 56 – Requirement QA-RP-56 .....	23
Table 57 – Requirement QA-RP-57 .....	23
Table 58 – Requirement QA-RP-58 .....	24
Table 59 – Requirement QA-RP-59 .....	24
Table 60 – Requirement QA-RP-60 .....	24
Table 61 – Requirement QA-RP-61 .....	25
Table 62 – Requirement QA-RP-62 .....	25
Table 63 – Requirement QA-RP-63 .....	25
Table 64 – Requirement QA-RP-64 .....	25
Table 65 – Requirement QA-RP-65 .....	26

## TABLE OF FIGURES

No table of figures entries found.

# 1 Purpose

This document describes the requirements for several types of quality reports that will be sent periodically to show the quality status of the product.

The principal objective of these reports is to aggregate information of several resources and present it on quality report that will be sent periodically.

## 2 Introduction

### 2.1 Overview

Currently, FeedZai uses several platforms to collect information about the product, such as Sonar for continuous analysis and source code quality management, Jira to capture and organize issues and stay up-to-date with the team activity, GIT as a distributed version system and source code management tool, Jenkins to get information about builds.

There are two main groups of reports:

- Short quality status – the report with highlighted information about quality status of product that will be sent to Pulse internal development mailing list.
- Full quality status – the report with detailed information about quality status of product that will be sent to FeedZai management.

### 2.2 Requirement Structure

Each requirement has the following fields:

- **ID** – Unique identifier of the requirement
- **Name** – Name of the requirement
- **Priority** – Priority of the requirement
- **Report Type** – Short or Full report type
- **Section** – The section on the report
- **Description** – Brief description of the requirement

### 2.3 Requirements

Field	Description
ID	QA-RP-01
Name	Periodicity of send of the quality report.
Priority	Must
Report Type	N/D

Section	N/D
Description	The quality report should be sent with weekly periodicity to Pulse internal development mailing list and optionally to FeedZai management. In case, when there are no new data for several fields, they should be marked with specific flag.

Table 1 – Requirement QA-RP -01

Field	Description
ID	QA- RP-02
Name	Format of quality report document.
Priority	Must
Report Type	N/D
Section	N/D
Description	The quality report should be sent in pdf format.

Table 2 – Requirement QA-RP-02

Field	Description
ID	QA- RP-03
Name	Information representation
Priority	Must
Report Type	N/D
Section	N/D
Description	The information should be represented by modules. Modules are the separated parts of the Pulse project.

Table 3 – Requirement QA-RP-03

Field	Description
ID	QA- RP-04
Name	Evolution comparing with the last report
Priority	Must
Report Type	N/D
Section	N/D
Description	Each future report should represent evolution comparing the data with the last emitted report.

Table 4 – Requirement QA-RP-04

Field	Description
ID	QA- RP-05
Name	Format of quality report document.
Priority	Low
Report Type	N/D
Section	N/D
Description	The quality report should be available in html format

Table 5 – Requirement QA-RP-05

Field	Description
ID	QA-RP-06
Name	Information resources for quality report
Priority	Must
Report Type	N/D
Section	N/D
Description	The information for quality report should be extracted from Sonar, Jira, Git and Jenkins.

Table 6 – Requirement QA-RP-06

Field	Description
ID	QA-RP-07
Name	Number of classes
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of classes (including nested classes, interfaces, enums and annotations).

Table 7 – Requirement QA-RP-07

Field	Description
ID	QA-RP-08
Name	Number of files by file type
Priority	Must



Report Type	Full
Section	Sonar
Description	Number of files by file type.

**Table 8 – Requirement QA-RP-08**

Field	Description
ID	QA-RP-09
Name	Number of lines
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of physical lines.

**Table 9 – Requirement QA-RP-09**

Field	Description
ID	QA-RP-10
Name	Lines of code
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Number of physical lines that contain at least one character which is neither a whitespace or a tabulation or part of a comment by program language.

**Table 10 – Requirement QA-RP-10**

Field	Description
ID	QA-RP-11
Name	Number of methods
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of methods.

**Table 11 – Requirement QA-RP-08**

Field	Description
ID	QA-RP-12
Name	Public API
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of public classes, public methods, public properties.

Table 12 – Requirement QA-RP-12

Field	Description
ID	QA-RP-13
Name	Number of packages
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of packages for Java and Scala languages.

Table 13 – Requirement QA-RP-13

Field	Description
ID	QA-RP-14
Name	Statements
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of statements.

Table 14 – Requirement QA-RP-14

Field	Description
ID	QA-RP-15
Name	Number of blank comments
Priority	Must
Report Type	Full

Section	Sonar
Description	Number of non-significant comment lines (empty, special characters, etc.). <b>Table 15 – Requirement QA-RP-15</b>

Field	Description
ID	QA-RP-16
Name	Number of commented lines
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of lines containing a comment. <b>Table 16 – Requirement QA-RP-16</b>

Field	Description
ID	QA-RP-17
Name	Percentage of commented lines (%)
Priority	Must
Report Type	Full and short
Section	Sonar
Description	The percentage of comment lines is calculated as follows: $\% \text{ comment lines} = \frac{\text{Comment lines}}{(\text{Lines of code} + \text{Comment lines})} \times 100$ <b>Table 17 – Requirement QA-RP-17</b>

Field	Description
ID	QA-RP-18
Name	Public documented API
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Percentage of public documented API. <b>Table 18 – Requirement QA-RP-18</b>

Field	Description
ID	QA-RP-19
Name	Public undocumented API
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Public API without comments headers and with incomplete Javadoc. <b>Table 19 – Requirement QA-RP-19</b>

Field	Description
ID	QA-RP-20
Name	Complexity / class
Priority	Must
Report Type	Full
Section	Sonar
Description	Diplomatic complexity by class. <b>Table 20 – Requirement QA-RP-20</b>

Field	Description
ID	QA-RP-21
Name	Complexity / method
Priority	Must
Report Type	Full
Section	Sonar
Description	Complexity by method. <b>Table 21 – Requirement QA-RP-21</b>

Field	Description
ID	QA-RP-22
Name	Complexity / file
Priority	Must

Report Type	Full
Section	Sonar
Description	Complexity by file. <b>Table 22 – Requirement QA-RP-22</b>

Field	Description
ID	QA-RP-23
Name	Duplicated blocks
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of duplicated blocks of code. <b>Table 23 – Requirement QA-RP-23</b>

Field	Description
ID	QA-RP-24
Name	Duplicated files
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of duplicated files involved in a duplication. <b>Table 24 – Requirement QA-RP-24</b>

Field	Description
ID	QA-RP-25
Name	Percentage of duplications (%)
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Percentage of duplications. <b>Table 25 – Requirement QA-RP-25</b>

Field	Description
ID	QA-RP-26
Name	Number of violations
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Number of violations. Violations with high priority should be visible on all reports.

Table 26 – Requirement QA-RP-26

Field	Description
ID	QA-RP-27
Name	Rules compliance
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Rules compliance. Violations with high priority should be visible on the report.

Table 27 – Requirement QA-RP-27

Field	Description
ID	QA-RP-28
Name	Technical Debt
Priority	Must
Report Type	Full
Section	Sonar
Description	<p>Debt ratio, work to reimburse, cost to reimburse. The values is calculated as follows:</p> $\text{Debt(in \$)} = (\text{cost\_to\_fix\_duplications} + \text{cost\_to\_fix\_violations} + \text{cost\_to\_comment\_public\_API} + \text{cost\_to\_fix\_uncovered\_complexity} + \text{cost\_to\_bring\_complexity\_below\_threshold}) * \text{daile\_rate}$ <p>Where :</p> <p>Duplications = cost_to_fix_one_block * duplicated_blocks</p> <p>Violations = cost_to fix_one_violation * mandatory_violations</p>

Comments = cost\_to\_comment\_one\_API \* public\_undocumented\_api

Coverage = cost\_to\_cover\_one\_of\_complexity \*  
uncovered\_complexity\_by\_tests (80% of coverage is the objective)

Complexity = cost\_to\_split\_a\_method \*  
(function\_complexity\_distribution >= 8) + cost\_to\_split\_a\_class \*  
(class\_complexity\_distribution >= 60)

Daily rate = 500 (the default daily rate of a developer. in \$)

**Table 28 – Requirement QA-RP-28**

Field	Description
ID	QA-RP-29
Name	Branch coverage of unit tests
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Percentage of branch coverage of unit tests.

**Table 29 – Requirement QA-RP-29**

Field	Description
ID	QA-RP-30
Name	Line coverage of unit tests
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Percentage of line coverage of unit tests.

**Table 30 – Requirement QA-RP-30**

Field	Description
ID	QA-RP-31
Name	Total unit tests coverage
Priority	Must

Report Type	Full and short
Section	Sonar
Description	<p>Mix of line coverage and branch coverage. The mix is calculated as follows:  <math display="block">\text{Coverage} = (\text{CT} + \text{CF} + \text{LC}) / (2 * \text{B} + \text{EL})</math>           where            CT = branches that have been evaluated to 'true' at least once            CF = branches that have been evaluated to 'false' at least once            LC = covered lines            B = total number of branches            EL = total number of executable lines</p>

Table 31 – Requirement QA-RP-31

Field	Description
ID	QA-RP-32
Name	Skipped unit tests
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of skipped unit tests.

Table 32 – Requirement QA-RP-32

Field	Description
ID	QA-RP-33
Name	Number of unit tests
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Number of unit tests.

Table 33 – Requirement QA-RP-33

Field	Description
ID	QA-RP-34



Name	Unit tests duration
Priority	Must
Report Type	Full
Section	Sonar
Description	Time required to execute all the unit tests.

**Table 34 – Requirement QA-RP-34**

Field	Description
ID	QA-RP-35
Name	Unit tests errors
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Number of tests that have failed.

**Table 35 – Requirement QA-RP-35**

Field	Description
ID	QA-RP-36
Name	Unit tests failures
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Number of tests that have failed with an unexpected exception.

**Table 36 – Requirement QA-RP-36**

Field	Description
ID	QA-RP-37
Name	Unit tests success (%)
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Percentage of unit tests success

**Table 37 – Requirement QA-RP-37**

Field	Description
ID	QA-RP-38
Name	Branch coverage of integration tests
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Percentage of branch coverage of integration tests. <b>Table 38 – Requirement QA-RP-38</b>

Field	Description
ID	QA-RP-39
Name	Line coverage of integration tests
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Percentage of line coverage of integration tests. <b>Table 39 – Requirement QA-RP-39</b>

Field	Description
ID	QA-RP-40
Name	Total integration tests coverage
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Mix of line coverage and branch coverage of integration test. <b>Table 40 – Requirement QA-RP-40</b>

Field	Description
ID	QA-RP-41
Name	Branch coverage of unit tests and integration tests
Priority	Must

Report Type	Full and short
Section	Sonar
Description	Mix of branch coverage of unit tests and integration tests. <b>Table 41 – Requirement QA-RP-41</b>

Field	Description
ID	QA-RP-42
Name	Line coverage of unit tests and integration tests
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Mix of line coverage of unit tests and integration tests. <b>Table 42 – Requirement QA-RP-42</b>

Field	Description
ID	QA-RP-43
Name	Total coverage of integration tests and unit tests
Priority	Must
Report Type	Full and short
Section	Sonar
Description	Mix of coverage of unit tests and integration tests. <b>Table 43 – Requirement QA-RP-43</b>

Field	Description
ID	QA-RP-44
Name	Total opened bugs
Priority	Must
Report Type	Full and short
Section	Jira
Description	Number of opened bugs <b>Table 44 – Requirement QA-RP-44</b>

Field	Description
ID	QA-RP-45
Name	New bugs
Priority	Must
Report Type	Full
Section	Jira
Description	Number of bugs opened comparing with the last report

**Table 45 – Requirement QA-RP-45**

Field	Description
ID	QA-RP-46
Name	Blocker bugs
Priority	Must
Report Type	Full
Section	Jira
Description	Number of blocker bugs

**Table 46 – Requirement QA-RP-46**

Field	Description
ID	QA-RP-47
Name	Critical bugs
Priority	Must
Report Type	Full and short
Section	Jira
Description	Number of critical bugs

**Table 47 – Requirement QA-RP-47**

Field	Description
ID	QA-RP-48
Name	Major bugs
Priority	Must
Report Type	Full and short

Section	Jira
Description	Number of major bugs <b>Table 48 – Requirement QA-RP-48</b>

Field	Description
ID	QA-RP-49
Name	Minor bugs
Priority	Must
Report Type	Full and short
Section	Jira
Description	Number of minor bugs <b>Table 49 – Requirement QA-RP-49</b>

Field	Description
ID	QA-RP-50
Name	Trivial bugs
Priority	Must
Report Type	Full and short
Section	Jira
Description	Number of trivial bugs <b>Table 50 – Requirement QA-RP-50</b>

Field	Description
ID	QA-RP-51
Name	New commits
Priority	Must
Report Type	Full
Section	Git
Description	Number of new commit comparing with the last report <b>Table 51 – Requirement QA-RP-51</b>

Field	Description
ID	QA-RP-52
Name	Commit per author
Priority	Must
Report Type	Full
Section	Git
Description	Number of new commit per author comparing with the last report <b>Table 52 – Requirement QA-RP-52</b>

Field	Description
ID	QA-RP-53
Name	Commit per clock hours
Priority	Must
Report Type	Full
Section	Git
Description	Number of new commit per author comparing with the last report <b>Table 53 – Requirement QA-RP-53</b>

Field	Description
ID	QA-RP-54
Name	Performance tests
Priority	Must
Report Type	Full
Section	Sonar
Description	<p>The result for each performance configuration should be presented. The result should be separated by application (Banking, Paywatch, Servebase, SIBS). Each application's configuration should contain the follow parameters:</p> <ul style="list-style-type: none"><li>• Number of processed events</li><li>• Total execution time</li><li>• Number of threads</li><li>• Minimum Rate</li><li>• Average Rate</li><li>• Hot rate</li><li>• Maximum Rate</li><li>• Minimum CPU</li></ul>

- Average CPU
- Maximum CPU
- Minimum Memory
- Average Memory
- Maximum Memory

Table 54 – Requirement QA-RP-54

Field	Description
ID	QA-RP-55
Name	Number of killed mutants
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of detected mutants

Table 55 – Requirement QA-RP-55

Field	Description
ID	QA-RP-56
Name	Mutation coverage
Priority	Must
Report Type	Full
Section	Sonar
Description	Percentage of mutation tests

Table 56 – Requirement QA-RP-56

Field	Description
ID	QA-RP-57
Name	Number of non covered mutations
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of non covered mutations

Table 57 – Requirement QA-RP-57

Field	Description
ID	QA-RP-58
Name	Number of survived mutations
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of survived mutations

Table 58 – Requirement QA-RP-58

Field	Description
ID	QA-RP-59
Name	Total number of mutations
Priority	Must
Report Type	Full
Section	Sonar
Description	Represent total number of generated mutations

Table 59 – Requirement QA-RP-59

Field	Description
ID	QA-RP-60
Name	Detailed information about analysis with Checkstyle tool
Priority	Must
Report Type	Full
Section	Jenkins
Description	Detailed information about checking source code with Checkstyle tool. Information should be available for each project, module package and file.

Table 60 – Requirement QA-RP-60

Field	Description
ID	QA-RP-61
Name	Detailed information about analysis with Findbugs tool
Priority	Must
Report Type	Full
Section	Jenkins



Description	Detailed information about checking source code with Findbugs tool. Information should be available for each project, module package and file.
-------------	--

Table 61 – Requirement QA-RP-61

Field	Description
ID	QA-RP-62
Name	Detailed information about analysis with PMD tool
Priority	Must
Report Type	Full
Section	Jenkins
Description	Detailed information about checking source code with PMD tool. Information should be available for each project, module package and file.

Table 62 – Requirement QA-RP-62

Field	Description
ID	QA-RP-63
Name	Detailed information about analysis with Open Tasks tool
Priority	Must
Report Type	Full
Section	Jenkins
Description	Detailed information about checking source code with Open Tasks tool. Information should be available for each project, module package and file.

Table 63 – Requirement QA-RP-63

Field	Description
ID	QA-RP-64
Name	Detailed information about analysis with Static Analysis tool
Priority	Must
Report Type	Full
Section	Jenkins
Description	Detailed information about checking source code with Static Analysis tool. Information should be available for each project, module package and file.

Table 64 – Requirement QA-RP-64

Field	Description
ID	QA-RP-65
Name	Information about build stability
Priority	Must
Report Type	Full
Section	Jenkins
Description	<p>Information about build stability. This is Jenkins metric. The possible values are follow:</p> <p>Stable build - A build is stable if it was built successfully and no publisher reports it as unstable.</p> <p>Unstable build - A build is unstable if it was built successfully and one or more publishers report it unstable. For example if the JUnit publisher is configured and a test fails then the build will be marked unstable.</p> <p>(Un)Stable project - A project is (un)stable if its most recent (completed) build is (un)stable.</p> <p>Successful build - A build is successful when the compilation reported no errors.</p> <p>Broken build or Failed Build - A build is broken if it failed during building. That is, it is not successful.</p> <p>Broken project - A project is broken if its most recent (completed) build is broken.</p>

Table 65 – Requirement QA-RP-65

Field	Description
ID	QA-RP-66
Name	Duplicated lines
Priority	Must
Report Type	Full
Section	Sonar
Description	Number of duplicated lines involved in a duplication.

Table 66 – Requirement QA-RP-66