|  |
| --- |
| Metadata\_Framework  Version 1.0  Code analysis |

|  |
| --- |
| **By: default**  **2023-08-21** |

# Content

[Content 1](#_Toc97156258)

[Introduction 2](#_Toc97156259)

[Configuration 2](#_Toc97156260)

[Synthesis 3](#_Toc97156261)

[Analysis Status 3](#_Toc97156262)

[Quality gate status 3](#_Toc97156263)

[Metrics 3](#_Toc97156264)

[Tests 3](#_Toc97156265)

[Detailed technical debt 3](#_Toc97156266)

[Metrics Range 5](#_Toc97156267)

[Volume 5](#_Toc97156268)

[Issues 6](#_Toc97156269)

[Charts 6](#_Toc97156270)

[Issues count by severity and type 8](#_Toc97156271)

[Issues List 8](#_Toc97156272)

[Security Hotspots 9](#_Toc97156273)

[Security hotspots count by category and priority 9](#_Toc97156274)

[Security hotspots List 9](#_Toc97156275)

# Introduction

This document contains results of the code analysis of Metadata\_Framework.

# Configuration

* Quality Profiles
  + Names: Sonar way [Python];
  + Files: AYeVFt0viG7mriH640Ur.json;
* Quality Gate
  + Name: Sonar way
  + File: Sonar way.xml

# Synthesis

## Analysis Status

|  |  |  |  |
| --- | --- | --- | --- |
| Reliability | Security | Security Review | Maintainability |
| A.png | **A.png** | **E.png** | **A.png** |

## Quality gate status

|  |  |
| --- | --- |
| Quality Gate Status | **ERROR.png** |

|  |  |
| --- | --- |
| Metric | Value |
| Reliability Rating on New Code | OK |
| Security Rating on New Code | OK |
| Maintainability Rating on New Code | OK |
| Coverage on New Code | ERROR (8.4% is less than 80%) |
| Duplicated Lines (%) on New Code | ERROR (25.1% is greater than 3%) |
| Security Hotspots Reviewed on New Code | ERROR (0.0% is less than 100%) |

## Metrics

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Coverage | Duplication | Comment  density | Median number of lines of code per file | Adherence to coding standard |
| 11.5 % | **22.7 %** | **24.6 %** | **773.5** | **99.8 %** |

## Tests

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Total | Success Rate | Skipped | Errors | Failures |
| 0 | **0 %** | **0** | **0** | **0** |

## Detailed technical debt

|  |  |  |  |
| --- | --- | --- | --- |
| Reliability | Security | Maintainability | Total |
| - | - | 0d 3h 47min | 0d 3h 47min |

## Metrics Range

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Cyclomatic  Complexity | Cognitive  Complexity | Lines of code per file | Comment  density (%) | Coverage | Duplication (%) |
| Min | 1.0 | 0.0 | 70.0 | 7.1 | 0.0 | 0.0 |
| Max | 203.0 | 679.0 | 8289.0 | 54.9 | 42.9 | 73.0 |

## Volume

|  |  |
| --- | --- |
| Language | Number |
| Python | 8289 |
| Total | 8289 |

# Issues

## Charts

## Issues count by severity and type

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type / Severity | INFO | MINOR | MAJOR | CRITICAL | BLOCKER |
| BUG | 0 | 0 | 0 | 0 | 0 |
| VULNERABILITY | 0 | 0 | 0 | 0 | 0 |
| CODE\_SMELL | 0 | 0 | 11 | 1 | 0 |

## Issues List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Description | Type | Severity | Number |
| Cognitive Complexity of functions should not be too high | Cognitive Complexity is a measure of how hard the control flow of a function is to understand. Functions with high Cognitive Complexity will be difficult to maintain. See Cognitive Complexity | CODE\_SMELL | CRITICAL | 1 |
| Assertions should not fail or succeed unconditionally | Assertions are meant to detect when code behaves as expected. An assertion which fails or succeeds all the time should be fixed. This rule raises an issue when an assertion method is given parameters which will make it succeed or fail all the time. It covers three cases: an assert statement or a unittest’s assertTrue or assertFalse method is called with a value which will be always True or always False. a unittest’s assertIsNotNone or assertIsNone method is called with a value which will be always None or never None. a unittest’s assertIsNot or assertIs method is called with a literal expression creating a new object every time (ex: [1, 2, 3]). Noncompliant Code Example import unittest class MyTestCase(unittest.TestCase): def expect\_fail1(self): assert False def expect\_fail2(self): self.assertTrue(False) # Noncompliant. This assertion always fails. def expect\_not\_none(self): self.assertIsNotNone(round(1.5)) # Noncompliant. This assertion always succeeds because "round" returns a number, not None. def helper\_compare(param): self.assertIs(param, [1, 2, 3]) # Noncompliant. This assertion always fails because [1, 2, 3] creates a new object. Compliant Solution import unittest class MyTestCase(unittest.TestCase): def expect\_fail(self): self.fail("This is expected") def expect\_not\_none(self): self.assertNotEqual(round(1.5), 0) def helper\_compare(param): self.assertEqual(param, [1, 2, 3]) See Python documentation - the unittest module Python documentation - the assert statement | CODE\_SMELL | MAJOR | 11 |

# Security Hotspots

## Security hotspots count by category and priority

|  |  |  |  |
| --- | --- | --- | --- |
| Category / Priority | LOW | MEDIUM | HIGH |
| LDAP Injection | 0 | 0 | 0 |
| Object Injection | 0 | 0 | 0 |
| Server-Side Request Forgery (SSRF) | 0 | 0 | 0 |
| XML External Entity (XXE) | 0 | 0 | 0 |
| Insecure Configuration | 0 | 0 | 0 |
| XPath Injection | 0 | 0 | 0 |
| Authentication | 0 | 0 | 0 |
| Weak Cryptography | 0 | 0 | 0 |
| Denial of Service (DoS) | 0 | 0 | 0 |
| Log Injection | 10 | 0 | 0 |
| Cross-Site Request Forgery (CSRF) | 0 | 0 | 0 |
| Open Redirect | 0 | 0 | 0 |
| Permission | 0 | 0 | 0 |
| SQL Injection | 0 | 0 | 0 |
| Encryption of Sensitive Data | 0 | 0 | 0 |
| Traceability | 0 | 0 | 0 |
| Buffer Overflow | 0 | 0 | 0 |
| File Manipulation | 0 | 0 | 0 |
| Code Injection (RCE) | 0 | 0 | 0 |
| Cross-Site Scripting (XSS) | 0 | 0 | 0 |
| Command Injection | 0 | 0 | 0 |
| Path Traversal Injection | 0 | 0 | 0 |
| HTTP Response Splitting | 0 | 0 | 0 |
| Others | 0 | 0 | 0 |

## Security hotspots List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Category | Name | Priority | Severity | Count |
| Log Injection | Configuring loggers is security-sensitive | LOW | CRITICAL | 10 |