

Author

CAST



**CISQ Compliance Report**

Application Name –

Version –

CAST AIP -

|  |
| --- |
|  |
|  |

Monday, xx July 2012

My Application Name

Version Number

My CAST Version

# Table of Content

Table of Content 2

1. Introduction 4

1.1. Application Characteristics 4

2. CISQ Violation Overview 5

2.1. CISQ violations Summary 5

2.1.1. CISQ Security 5

2.1.2. CISQ Reliability 6

2.1.3. CISQ Performance Efficiency 7

2.1.4. CISQ Maintainability 8

3. CISQ Security Violation Summary 9

3.1. ASCSM-CWE-22 - Improper path traversal 9

3.2. ASCSM-CWE-78 - OS command injection 9

3.3. ASCSM-CWE-79 – Cross-site scripting 10

3.4. ASCSM-CWE-89 – SQL injection 10

3.5. ASCSM-CWE-99 – Unsanitized user input used to access a named resource 10

3.6. ASCSM-CWE-120 – Buffer overflow 11

3.7. ASCSM-CWE-134 – Improper format string neutralization 11

3.8. ASCSM-CWE-327 – Unvetted cryptographic algorithms 11

3.9. ASCSM-CWE-396 – Catch of overly broad exception types 12

3.10. ASCSM-CWE-397 – Throw of overly broad exception types 12

3.11. ASCSM-CWE-434 – Unsanitized user input in file upload statement 12

3.12. ASCSM – CWE – 672 - Access to released or expired resources 13

3.13. ASCSM-CWE-681 – Incompatible numeric type conversion 13

3.14. ASCSM-CWE-772 – Unreleased resource 13

3.15. ASCSM-CWE-789 – Unchecked range of user input to a buffer 14

3.16. ASCSM-CWE-798 – Hard-coded credentials for remote resources 14

3.17. ASCSM-CWE-835 – Infinite recursion 14

4. CISQ Reliability Violation Summary 15

4.1. ASCRM-CWE-120 – Buffer overflow 15

4.2. ASCRM-CWE-252-data – Unchecked return parameter from data handling operations 15

4.3. ASCRM-CWE-396 – Catch of overly broad exception types 16

4.4. ASCRM-CWE-674 – Recursion 16

4.5. ASCRM-CWE-704 – Incompatible data type conversion 16

4.6. ASCRM-CWE-772 – Unreleased resource 17

4.7. ASCRM-RLB-1 – Empty exception block 17

4.8. ASCRM-RLB-2 – Missing serialization control element 17

4.9. ASCRM-RLB-3 – Serialized data element containing non-serialized items 18

4.10. ASCRM-RLB-4 – Persistent data without proper comparison controls 18

4.11. ASCRM-RLB-5 – Improper runtime resource management 18

4.12. ASCRM-RLB-12 – Improper locking of singleton classes 19

4.13. ASCRM-RLB-13 – Cyclic dependencies 19

4.14. ASCRM-RLB-14 – Parent class referencing child class 19

4.15. ASCRM-RLB-18 – Hard-coded network resource information 20

4.16. ASCRM-RLB-19 – Synchronous call missing timeout 20

5. CISQ Performance Efficiency Violation Summary 21

5.1. ASCPEM-PRF-2 – Immutable text data 21

5.2. ASCPEM-PRF-4 – Complex read/write access 21

5.3. ASCPEM-PRF-5 – Incorrect indices 22

5.4. ASCPEM-PRF-8 – Resource-consuming operation in loop 22

5.5. ASCPEM-PRF-9 – Excessive data queries in non-stored procedure 22

5.6. ASCPEM-PRF-10 – Excessive data queries in client-side code 23

5.7. ASCPEM-PRF-12 – Excessively large data element 23

5.8. ASCPEM-PRF-15 – Unreleased data 23

6. CISQ Maintainability Violation Summary 24

6.1. ASCMM-MNT-1 – Control transferred outside of switch statement 24

6.2. ASCMM-MNT-4 – Excessive coupling 24

6.3. ASCMM-MNT-6 – Excessive commented-out code 24

6.4. ASCMM-MNT-7 – Circular dependencies among modules 25

6.5. ASCMM-MNT-8 – Excessively large file 25

6.6. ASCMM-MNT-11 – Excessive cyclomatic complexity 25

6.7. ASCMM-MNT-13 – Excessive parameterization 26

6.8. ASCMM-MNT-15 – Public data element 26

6.9. ASCMM-MNT-17 – Excessive inheritance levels 26

6.10. ASCMM-MNT-18 – Excessive child classes 27

6.11. ASCMM-MNT-19 – Element redundancy 27

6.12. ASCMM-MNT-20 – Dead code 27

7. Appendix 28

7.1. About CAST Software Intelligence 28

7.2. About CISQ Automated Quality Characteristic Measures 28

# Introduction

This assessment is an effort to determine the overall quality of the said applications against CISQ rules and measure the overall health of the application. This assessment uses the CAST Application Intelligence Platform (AIP) to automatically scan the implementation of these applications to review the architecture, design, and code against current industry best practices and known design flaws that may impact performance.

CAST AIP applies over 1200 engineering checks based on standards and measurements developed by the Software Engineering Institute (SEI), International Standards Organization (ISO), Consortium for IT Software Quality (CISQ), the Institute of Electrical and Electronics Engineers (IEEE) and the technology provider industry. The resulting analysis identifies specific flaws in the software and aggregates this information into metrics to objectively quantify the structural quality of the application.

## Application Characteristics

This assessment is focused solely on the technical implementation of the said application (user interface to database), with no investigation of the functionality.

|  |  |
| --- | --- |
| Name | Value |
| kLoC | 504 |
| Files | 6,586 |
| Classes | 593 |
| SQL Art. | 0 |
| Tables | 119 |

*Fig 1: Application Technology characteristics Table 1: Application characteristics*

# CISQ Violation Overview

This section provide a summary of the CISQ specified vulnerability identified in the structural quality analysis and mesurement by CAST AIP against the CISQ standard. Details about CISQ Standard can be found [here](http://it-cisq.org/standards/automated-quality-characteristic-measures).

## CISQ violations Summary

#### 2.1.1. CISQ Security

List of CISQ rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 2: CISQ Security Rules*

### 

#### 2.1.2. CISQ Reliability

List of CISQ rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 3: CISQ Reliability Rules*

### 

#### 2.1.3. CISQ Performance Efficiency

List of CISQ rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 4: CISQ Performance Efficiency Rules*

#### 2.1.4. CISQ Maintainability

List of CISQ rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 5: CISQ Maintainability Rules*

# CISQ Security Violation Summary

Security assesses the degree to which an application protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization (ISO 25010). Security measures the risk of potential security breaches due to poor coding and architectural practices. Security problems have been studied extensively by the Software Assurance community and have been codified in the Common Weakness Enumeration (CWE) at [cwe.mitre.org](http://it-cisq.org/standards/automated-quality-characteristic-measures/security/cwe.mitre.org).

The CISQ Automated Source Code Security Measure draws from the [CWE/SANS Institute Top 25 Most Dangerous Software Errors](http://cwe.mitre.org/top25/#Listing) and identifies the most widespread and frequently exploited security weaknesses in software. Twenty-two of these weaknesses are detectable through analyzing the source code and form the basis of the CISQ measure. These 22 weaknesses constitute the most frequent ways unauthorized parties breach a system. Thus, the CISQ measure is a good predictor of how easily an application can suffer unauthorized penetration that results in stolen information, altered records, or other forms of malicious behavior.

#### 

## ASCSM-CWE-22 - Improper path traversal

List of ASCSM-CWE-22 violations that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 6: ASCSM-CWE-22* *violations*

## ASCSM-CWE-78 - OS command injection

List of ASCSM-CWE-78 violations that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 7: ASCSM-CWE-78* *violations*

## ASCSM-CWE-79 – Cross-site scripting

List of ASCSM-CWE-79 violations that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 8: ASCSM-CWE-79* *violations*

## ASCSM-CWE-89 – SQL injection

List of ASCSM-CWE-89 rules that had any findings in this application

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 9: ASCSM-CWE-89* *violations*

## ASCSM-CWE-99 – Unsanitized user input used to access a named resource

List of ASCSM-CWE-99 rules that had any findings in this application

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 10: ASCSM-CWE-99* *violations*

## ASCSM-CWE-120 – Buffer overflow

List of ASCSM-CWE-120 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 11: ASCSM-CWE-120 violations*

## ASCSM-CWE-134 – Improper format string neutralization

List of ASCSM-CWE-134 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 12: ASCSM-CWE-134 violations*

## ASCSM-CWE-327 – Unvetted cryptographic algorithms

List of ASCSM-CWE-327 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 13: ASCSM-CWE-327 violations*

## ASCSM-CWE-396 – Catch of overly broad exception types

List of ASCSM-CWE-396 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 14: ASCSM-CWE-396 violations*

## ASCSM-CWE-397 – Throw of overly broad exception types

List of ASCSM-CWE-397 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 15: ASCSM-CWE-397 violations*

## ASCSM-CWE-434 – Unsanitized user input in file upload statement

List of ASCSM-CWE-434 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 16: ASCSM-CWE-434 violations*

## ASCSM – CWE – 672 - Access to released or expired resources

List of ASCSM-CWE-672 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 17: ASCSM-CWE-672 violations*

## ASCSM-CWE-681 – Incompatible numeric type conversion

List of ASCSM-CWE-681 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 18: ASCSM-CWE-681 violations*

## ASCSM-CWE-772 – Unreleased resource

List of ASCSM-CWE-772 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 19: ASCSM-CWE-772 violations*

## ASCSM-CWE-789 – Unchecked range of user input to a buffer

List of ASCSM-CWE-789 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 20: ASCSM-CWE-789 violations*

## ASCSM-CWE-798 – Hard-coded credentials for remote resources

List of ASCSM-CWE-798 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 21: ASCSM-CWE-798 violations*

## ASCSM-CWE-835 – Infinite recursion

List of ASCSM-CWE-853 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 22: ASCSM-CWE-853 violations*

# CISQ Reliability Violation Summary

Reliability measures the risk of potential application failures and the stability of an application when confronted with unexpected conditions. According to ISO/IEC/IEEE 24765, Reliability is the degree to which a system, product, or component performs specified functions under specified conditions for a specified period of time. The reason for checking and monitoring Reliability is to prevent or at least reduce application downtime, outages, data corruption, and errors that directly affect users.

The CISQ Automated Source Code Reliability Measure is composed from 29 critical violations of architectural and coding practice that affect the availability, fault tolerance, recoverability, and data integrity of an application. The CISQ Reliability measure produces a quality score based on the count of violations discovered in the software and can be turned into a density measure when divided by the size of the software.

## ASCRM-CWE-120 – Buffer overflow

List of ASCRM-CWE-120 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 23:* ASCRM-CWE-120 *violations*

## ASCRM-CWE-252-data – Unchecked return parameter from data handling operations

List of ASCRM-CWE-252-data rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 24:* ASCRM-CWE-252-data *violations*

## ASCRM-CWE-396 – Catch of overly broad exception types

List of ASCRM-CWE-396 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 25: ASCSM-CWE-396 violations*

## ASCRM-CWE-674 – Recursion

List of ASCRM-CWE-674 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 26: ASCSM-CWE-674 violations*

## ASCRM-CWE-704 – Incompatible data type conversion

List of ASCRM-CWE-704 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 27: ASCSM-CWE-704 violations*

## ASCRM-CWE-772 – Unreleased resource

List of ASCRM-CWE-772 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 28: ASCSM-CWE-772 violations*

## ASCRM-RLB-1 – Empty exception block

List of ASCRM-RLB-1 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 29: ASCRM-RLB-1* *violations*

## ASCRM-RLB-2 – Missing serialization control element

List of ASCRM-RLB-2 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 30: ASCRM-RLB-2* *violations*

## ASCRM-RLB-3 – Serialized data element containing non-serialized items

List of ASCRM-RLB-3 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 31: ASCRM-RLB-3* *violations*

## ASCRM-RLB-4 – Persistent data without proper comparison controls

List of ASCRM-RLB-4 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 32: ASCRM-RLB-4* *violations*

## ASCRM-RLB-5 – Improper runtime resource management

List of ASCRM-RLB-5 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 33: ASCRM-RLB-5* *violations*

## ASCRM-RLB-12 – Improper locking of singleton classes

List of ASCRM-RLB-12 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 34: ASCRM-RLB-12* *violations*

## ASCRM-RLB-13 – Cyclic dependencies

List of ASCRM-RLB-13 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 35: ASCRM-RLB-13* *violations*

## ASCRM-RLB-14 – Parent class referencing child class

List of ASCRM-RLB-14 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 36: ASCRM-RLB-14* *violations*

## ASCRM-RLB-18 – Hard-coded network resource information

List of ASCRM-RLB-18 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 37: ASCRM-RLB-18* *violations*

## ASCRM-RLB-19 – Synchronous call missing timeout

List of ASCRM-RLB-19 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 38: ASCRM-RLB-19* *violations*

# CISQ Performance Efficiency Violation Summary

Performance Efficiency assesses characteristics that affect an application’s response behavior and use of resources under stated conditions (ISO/IEC 25010). Performance Efficiency affects customer satisfaction, workforce productivity, application scalability, response-time degradation, and inefficient use of processing or storage resources. The Performance Efficiency of an application lies in each individual component ‘s performance, as well as in the effect of each component on the behavior of the chain of components comprising a transaction in which it participates.

The CISQ Automated Source Code Performance Efficiency Measure is composed from 15 critical violations of response time behavior, processor use, and memory use of an application. A quality score is produced based on the count of violations discovered in the source code and can be used as a density metric when divided by software size.

## ASCPEM-PRF-2 – Immutable text data

List of ASCPEM-PRF-2 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 39:* ASCPEM-PRF-2 *violations*

## ASCPEM-PRF-4 – Complex read/write access

List of ASCPEM-PRF-4 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 40:* ASCPEM-PRF-4 *violations*

## ASCPEM-PRF-5 – Incorrect indices

List of ASCPEM-PRF-5 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 41:* ASCPEM-PRF-5 *violations*

## ASCPEM-PRF-8 – Resource-consuming operation in loop

List of ASCPEM-PRF-8 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 42:* ASCPEM-PRF-8 *violations*

## ASCPEM-PRF-9 – Excessive data queries in non-stored procedure

List of ASCPEM-PRF-9 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 43:* ASCPEM-PRF-9 *violations*

## ASCPEM-PRF-10 – Excessive data queries in client-side code

List of ASCPEM-PRF-10 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 44:* ASCPEM-PRF-10 *violations*

## ASCPEM-PRF-12 – Excessively large data element

List of ASCPEM-PRF-12rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 45:* ASCPEM-PRF-12 *violations*

## ASCPEM-PRF-15 – Unreleased data

List of ASCPEM-PRF-15 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 46:* ASCPEM-PRF-15 *violations*

# CISQ Maintainability Violation Summary

## ASCMM-MNT-1 – Control transferred outside of switch statement

List of ASCMM-MNT-1 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 47:* ASCMM-MNT-1 *violations*

## ASCMM-MNT-4 – Excessive coupling

List of ASCMM-MNT-4 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 48:* ASCMM-MNT-4 *violations*

## ASCMM-MNT-6 – Excessive commented-out code

List of ASCMM-MNT-6 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 49:* ASCMM-MNT-6 *violations*

## ASCMM-MNT-7 – Circular dependencies among modules

List of ASCMM-MNT-7 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 50:* ASCMM-MNT-7 *violations*

## ASCMM-MNT-8 – Excessively large file

List of ASCMM-MNT-8 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 51:* ASCMM-MNT-8 *violations*

## ASCMM-MNT-11 – Excessive cyclomatic complexity

List of ASCMM-MNT-11 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 52:* ASCMM-MNT-11 *violations*

## ASCMM-MNT-13 – Excessive parameterization

List of ASCMM-MNT-13 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 53:* ASCMM-MNT-13 *violations*

## ASCMM-MNT-15 – Public data element

List of ASCMM-MNT-15 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 54:* ASCMM-MNT-15*violations*

## ASCMM-MNT-17 – Excessive inheritance levels

List of ASCMM-MNT-17 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 55:* ASCMM-MNT-17 *violations*

## ASCMM-MNT-18 – Excessive child classes

List of ASCMM-MNT-18 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 56:* ASCMM-MNT-18 *violations*

## ASCMM-MNT-19 – Element redundancy

List of ASCMM-MNT-19 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 57:* ASCMM-MNT-19 *violations*

## ASCMM-MNT-20 – Dead code

List of ASCMM-MNT-20 rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Violations | Added Violations | Removed Violations |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 58:* ASCMM-MNT-20 *violations*

# Appendix

## About CAST Software Intelligence

Software Intelligence creates understanding into software architecture, end to end transaction flows, data access patterns and more, helping teams work confidently and faster. Hundreds of companies rely on CAST Software Intelligence to improve end-user satisfaction and time-to-market, prevent business disruption and reduce cost, enabling them to move past today’s obstacles and to tackle the next wave of innovation.

[Click here](https://www.castsoftware.com/software-intelligence) for more information about CAST Software Intelligence.

## About CISQ Automated Quality Characteristic Measures

CISQ has developed Automated Quality Characteristic Measures to measure and manage the structural quality of IT application software. The automated measures for Security, Reliability, Performance Efficiency, and Maintainability are now OMG® approved standards making them global standards for use by IT organizations.

These measures were developed from coding rules covering some of the most serious violations of good architectural and coding practices that should be avoided and can be detected through static code analysis. Each measure counts the number of violations of the architectural and coding rules related to that quality characteristic, and then can be used in creating metrics for defect density, etc.

|  |  |
| --- | --- |
| [Security](http://it-cisq.org/standards/automated-quality-characteristic-measures/security/) | Critical security violations in the source code drawn from the Top 25 security weaknesses in the Common Weakness Enumeration (CWE) repository |
| [Reliability](http://it-cisq.org/standards/automated-quality-characteristic-measures/reliability/) | Critical violations of availability, fault tolerance, and recoverability of software |
| [Performance Efficiency](http://it-cisq.org/standards/automated-quality-characteristic-measures/performance-efficiency/) | Critical violations of response time, as well as processor, memory, and utilization of other resources by the software |
| [Maintainability](http://it-cisq.org/standards/automated-quality-characteristic-measures/maintainability/) | Critical violations of modularity, architectural compliance, reusability, analyzability, and changeability in software |