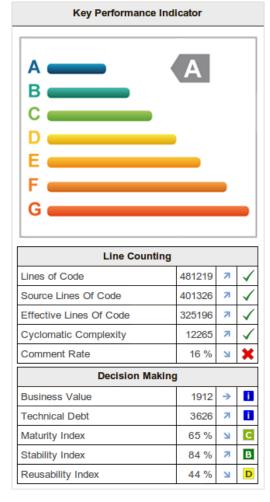


VALUING AND IMPROVING

SOFTWAR CAPITA





# Squore 17.0.8

**Eclipse Plugin Guide** 

Reference : SUM\_Squore\_ECL\_PLG Version : 17.0.8 Date : 30/11/2017

Squoring Technologies 76, allées Jean Jaurès / 31000 Toulouse - FRANCE / Tel. +33 581 346 397 http://www.squoring.com/ APE 5829 C - SIREN 521171629 – RCS Toulouse

Copyright © 2017 Squoring Technologies

#### Abstract

This edition of the Eclipse Plugin Guide applies to Squore 17.0.8 and to all subsequent releases and modifications until otherwise indicated in new editions.

#### Licence

No part of this publication may be reproduced, transmitted, stored in a retrieval system, nor translated into any human or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of the copyright owner, Squoring Technologies.

Squoring Technologies reserves the right to revise this publication and to make changes from time to time without obligation to notify authorised users of such changes. Consult Squoring Technologies to determine whether any such changes have been made.

The terms and conditions governing the licensing of Squoring Technologies software consist solely of those set forth in the written contracts between Squoring Technologies and its customers.

All third-party products are trademarks or registered trademarks of their respective companies.

#### Warranty

Squoring Technologies makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Squoring Technologies shall not be liable for errors contained herein nor for incidental or consequential damages in connection with the furnishing, performance or use of this material.



# **Table of Contents**

Typographical Conventionsiv	
Acronyms and Abbreviations	/
1. Introduction 1	L
1.1. Foreword 1	
1.2. About This Document 1	L
1.3. Contacting Squoring Technologies Product Support 1	L
1.4. Responsabilities 2	
1.5. Getting the Latest Version of this Manual 2	2
2. Introducing the Squore Eclipse Plugin 3	3
2.1. Squore Eclipse Plugin in the Squore ecosystem 3	3
2.2. Who is the Squore Eclipse Plugin destined to? 4	ł
3. Installing and setting up the Squore Eclipse Plugin 5	5
3.1. Installation Prerequisites 5	5
3.1.1. Supported Operating Systems 5	5
3.1.2. For All Systems	5
3.1.3. Packages for Windows	5
3.1.4. Packages for Linux 6	5
3.1.5. Packages for CentOS and Red Hat Enterprise Linux 6	
3.1.6. Packages for Ubuntu	7
3.2. Squore Eclipse Plugin installation	
3.3. Squore Eclipse Plugin setup 8	
4. Getting Started With the Squore Eclipse Plugin	)
4.1. Launching Your First Analysis	
4.2. Visualising Analysis Results 11	L
Index	ŝ

iii

#### http://www.squoring.com/



# **Typographical Conventions**

The following conventions are used in this manual.

Typeface or Symbol	Meaning
Bold	Book titles, important items, or items that can be selected including buttons and menu choices. For example: Click the <b>Next</b> button to continue
Italic	A name of a user defined textual element. For example: <b>Username</b> : <i>admin</i>
Courier New	Files and directories; file extensions, computer output. For example: Edit the config.xml file
Courier Bold	Commands, screen messages requiring user action. For example: <b>Username</b> : <i>admin</i>
>	Menu choices. For example: Select File > Open . This means select the File menu, then select the Open command from it.
<>	Generic terms. For example: <squore_home> refers to the Squore installation directory.</squore_home>

<u>Notes</u>

Screenshots displayed in this manual may differ slightly from the ones in the actual product.

iv

#### http://www.squoring.com/



# **Acronyms and Abbreviations**

The following acronyms and abbreviations are used in this manual.

CI	Continuous Integration
CLI	Command Line Interface
DP	Data Provider, a Squore module capable of handling input from various other systems and import information into Squore
RC	Repository Connector, a Squore module capable of extracting source code from source code management systems.

v

#### http://www.squoring.com/



# **1. Introduction**

### 1.1. Foreword

This document was released by Squoring Technologies.

It is part of the user documentation of the Squore software product edited and distributed by Squoring Technologies.

## **1.2. About This Document**

This document is the Eclipse Plugin Guide for Squore.

Its objective is to guide you through various aspects of the Squore Eclipse Plugin:

- Chapter 2, Introducing the Squore Eclipse Plugin positions the Squore Eclipse Plugin inside the Squore ecosystem, explaining its main features.
- Chapter 3, Installing and setting up the Squore Eclipse Plugin describes the steps to follow in order to install the Squore Eclipse Plugin.
- Chapter 4, Getting Started With the Squore Eclipse Plugin walks you through the features provided by the Squore Eclipse Plugin.

If you are already familiar with Squore, you can navigate this manual by looking for what has changed since the previous version. New functionality is tagged with **(new in 17.0)** throughout this manual. A summary of the new features described in this manual is available in the entry **\* What's New in Squore 17.0?** of this manual's Index.

For information on how to use and configure Squore, the full suite of manuals includes:

- → Squore Installation Checklist
- → Squore Installation and Administration Guide
- → Squore Getting Started Guide
- → Squore Command Line Interface
- → Squore Configuration Guide
- → Squore Eclipse Plugin Guide
- → Squore Reference Manual

## **1.3. Contacting Squoring Technologies Product Support**

If the information provided in this manual is erroneous or inaccurate, or if you encounter problems during your installation, contact Squoring Technologies Product Support: http://support.squoring.com/

You will need a valid Squore customer account to submit a support request. You can create an account on the support website if you do not have one already.

For any communication:

#### support@squoring.com

Squoring Technologies Product Support

76, allées Jean Jaurès / 31000 Toulouse - FRANCE

Ref : SUM\_Squore\_ECL\_PLG

Version : 17.0.8 Date : 30/11/2017 1



## **1.4. Responsabilities**

Approval of this version of the document and any further updates are the responsibility of Squoring Technologies.

## **1.5. Getting the Latest Version of this Manual**

The version of this manual included in your Squore installation may have been updated. If you would like to check for updated user guides, consult the Squoring Technologies documentation site to consult or download the latest Squore manuals at http://support.squoring.com/documentation/17.0.8. Manuals are constantly updated and published as soon as they are available.

2

#### http://www.squoring.com/

# 2. Introducing the Squore Eclipse Plugin

## 2.1. Squore Eclipse Plugin in the Squore ecosystem

Squore analyzes projects (in a broad sense) to provide high level indicators and produce personalised dashboards and tailored action lists for each kind of connected user. The whole process can be automated, and integrated in a continuous integration framework.

Information generated in this fashion can be refreshed with any desired rate, and the results will be available each time the Squore process concludes a new build. However, this scenario is best suited to a daily cycle at most, where all input files are the result of a daily commit, and results are generated along with the nightly build.

The Squore Eclipse Plugin is destined to operate at a faster rate, the objective being to deliver results during the development process, before files are committed and fed to the continuous integration process. Tightening the processing loop has several consequences for the Squore Eclipse Plugin behavior:

It doesn't need to be connected to a central Squore server to rate a project, unless for project activation / licence validation purposes.

By limiting communication with the server to a minimum, we make sure projects can evolve outside the global rating cycle, but still offer the rating service.

Should the server be unreachable, rating service is still available for a whole week, before being deactivated. Once a connection is available again, and the credentials are validated, the rating capabilities are restored.

It must be able to process and rate a project locally
 The Squore Eclipse Plugin runs the Squan Sources static code analyser on the sources in your Eclipse project.

Note that no other data provider is run by the Squore Eclipse Plugin, so the overall rating of your project may vary from the one obtained when analysing the same for the same code base on Squore Server. Metrics and violations returned by Squan Sources will nevertheless be the same.

It must use a configuration (in terms of Analysis Model and Data Providers) synchronised with the one installed on Squore Server

Since the Squore Eclipse Plugin uses a Squore rating module, it needs a rating environment, which must be synchronised with the server's, to ensure results equivalence.

The Squore Eclipse Plugin will perform such synchronisation during the product activation, which is a feature available in the Squore Eclipse Plugin preferences page.

It must be able to manage projects using other languages than Java (e.g. C, C++, C# Ada)
 The Squore Eclipse Plugin benefits from the Squore source code analyzer, which provides support for several languages (see Squore documentation for more details).

Analysis results will be available at folder and file level.

- It must be integrated into the Eclipse IDE, taking advantage of its extendability
   The Eclipse plugin development framework offers a number of mechanisms to enhance various parts of the IDE. The added features have been designed to be both unobtrusive and informative.
- It must enhance a project's available information by displaying ratings, Action Items, Findings and highlights for each compatible object
   The added information is assigned to each artefact, which means at project, file, module, class and function level. Each of these elements will have a rating, as well as a list of Action Items and Findings, depending on

the model. A search view offering project artefacts sorted by rating is also available.

It must be able to manage local versions, which are dissociated from the source code control versioning mechanism

Ref : SUM\_Squore\_ECL\_PLG

3



One of the differences between the Squore Eclipse Plugin and Squore is that Squore Server uses a database to store the project builds across multiple versions, whereas the Squore Eclipse Plugin uses a local file storage mechanism, to maintain a local history.

This allows functions such as DELTA\_VALUE or PREVIOUS\_VALUE to work as expected, and produce trend analysis indicators.

Trends are based on the evolution of indicators between a baseline analysis and the last-rated analysis.

## 2.2. Who is the Squore Eclipse Plugin destined to?

As hinted by its name, the Squore Eclipse Plugin is a developer tool. Its objective is to provide users with enough information to help them:

- → Process the generated list of Action Items
- → Check the evolution of the source code rating as it is being produced
- → Find what artefacts are responsible for a poor rating, and understand why
- Anticipate and address problems that will appear in the project's daily Squore server build before the code is referenced in a SCM system

As the Squore Eclipse Plugin seamlessly integrates Squan Sources results in the IDE, and uses the same model as the one on Squore Server, developers will gain visibility over the evaluation process, and improve their development practices to match the standard defined by the model.

The integrated rating feedback also gives the developer the opportunity to actively participate to the global effort on quality improvement.

4

#### http://www.squoring.com/



# 3. Installing and setting up the Squore Eclipse Plugin

## **3.1. Installation Prerequisites**

### 3.1.1. Supported Operating Systems

The following is a list of the officially supported and tested operating systems:

- → CentOS 6
- → CentOS 7
- → Fedora 19
- → Ubuntu Server 14.04
- Windows 7
- → Windows 8
- → Windows 10

#### Note

On Linux, a 64-bit version of the OS is required

On Windows, a 64-bit version of the OS is required if you want Squore to run as a Windows service, but the installation is also supported on a 32-bit system (and is started and stopped using .bat files instead of a Windows Service)

The following is a list of the operating systems that are not regularly tested but are known to be working:

- → RedHat EL 6
- → RedHat EL 7
- → SuSe Linux 11.1
- → Ubuntu Server 10.04
- → Ubuntu Server 16.04
- → Windows Server 2008 R2
- → Windows Server 2012 R2

### 3.1.2. For All Systems

For a successful installation of Squore, you will need:

- The latest version of the Squore Eclipse Plugin, which can be downloaded from http://localhost:8180/ SQuORE\_Server/eclipse
- The Oracle Java Runtime Environment version 8 (other versions are not supported)
- → A supported Eclipse version (Luna SR2 and up)
- → Valid credentials to log into a Squore Server
- $\rightarrow$  At least 2 GB of space available on the disk for a full installation
- → At least 8 GB of RAM on the server machine
- → At least 4 GB of RAM on the client machine

### 3.1.3. Packages for Windows

A compatible version of Eclipse must be downloaded from http://www.eclipse.org/downloads/.

Ref : SUM\_Squore\_ECL\_PLG



### 3.1.4. Packages for Linux

On Linux platforms, the following must be installed before installing Squore:

- → **Perl** version 5.10.1 or greater including the following extra-modules:
  - → Mandatory packages:
    - Algorithm::Diff [module details] [http://search.cpan.org/~nedkonz/Algorithm-Diff/lib/Algorithm/ Diff.pm]
    - Archive::Zip [module details] [http://search.cpan.org/~phred/Archive-Zip/lib/Archive/Zip.pm]
    - Date::Calc [module details] [http://search.cpan.org/~stbey/Date-Calc/lib/Date/Calc.pod]
    - Digest::SHA [module details] [http://search.cpan.org/~mshelor/Digest-SHA/lib/Digest/SHA.pm]
    - → HTTP::Request [module details] [http://search.cpan.org/~gaas/HTTP-Message/lib/HTTP/ Request.pm]
    - JSON [module details] [http://search.cpan.org/~makamaka/JSON/lib/JSON.pm]
    - JWP [module details] [http://search.cpan.org/~ether/libwww-perl/lib/LWP.pm]
    - LWP::UserAgent [module details] [http://search.cpan.org/~gaas/libwww-perl/lib/LWP/ UserAgent.pm]
    - Time::HiRes [module details] [http://search.cpan.org/~zefram/Time-HiRes/HiRes.pm]
    - > XML::Parser [module details] [http://search.cpan.org/~toddr/XML-Parser/Parser.pm]

#### Tip

If some of these modules are not available as packages on your operating system, use your perl installation's cpan to install the modules. Using the OS packages is recommended, as it avoids having to reinstall via cpan after upgrading your version of perl.

- → Tcl version 8.5 or greater,
- → A supported Eclipse version (Luna SR2 and up)

### 3.1.5. Packages for CentOS and Red Hat Enterprise Linux

On Red Hat Enterprise Linux and CentOS (6.5 and 7.1), the dependencies are satisfied by the following packages:

#### Mandatory packages:

- → java-1.8.0-openjdk
- ⇒ perl
- perl-Algorithm-Diff
- → perl-Archive-Zip
- → perl-Date-Calc
- → perl-Digest-SHA
- → perl-JSON
- → perl-libwww-perl
- → perl-Time-HiRes
- → perl-XML-Parser
- → tcl
- → A compatible installation of Eclipse downloaded from http://eclipse.org/downloads .

Ref : SUM\_Squore\_ECL\_PLG

6



For more information about how to install the Extra Packages for Enterprise Linux (EPEL) repository, consult https://fedoraproject.org/wiki/EPEL.

### 3.1.6. Packages for Ubuntu

On Ubuntu 14.04.3 LTS, the dependencies are satisfied by the following packages:

Mandatory packages:

- ibalgorithm-diff-perl
- ibarchive-zip-perl
- → libdate-calc-perl
- ibhttp-message-perl
- ibjson-perl
- → libwww-perl
- ibxml-parser-perl
- → openjdk-8-jre
- → perl
- → tcl
- → A compatible installation of Eclipse downloaded from http://eclipse.org/downloads .

## 3.2. Squore Eclipse Plugin installation

The Squore Eclipse Plugin complies with the Eclipse requirements, and is an OS-independent package to be deployed in the Eclipse plugins directory. Installation is done through an update site.

#### Tip

In order to launch analyses, the plugin needs to use an installation of perl and tclsh. If you are using a Windows operating system, the easiest way to install both dependencies is to install the Squore CLI package on your machine following the procedure outlined in the Command Line Interface manual.

- > Create a new update site within Eclipse
  - Select Help > Install new software;.
  - → Click on Available Software Sites;.
  - → Add a new site;
  - → Give it a name (for example: "Squore Eclipse Plugin");
  - → Set the location to http://localhost:8180/SQuORE\_Server/eclipse
- → Go back to the "Install" window with the "OK" buttons;
- → Select the newly created update site;
- $\rightarrow$  Proceed with the plugin installation as with any Eclipse plugin.

#### Note

- To speed-up installation procedure, it is possible to uncheck "Contact all update sites during install to find required software", as no external software is needed by the Squore Eclipse Plugin.
- → Should the plugin fail to deploy, Eclipse will notify the user of the problem cause, and will start its operations as usual. If necessary, copy and send the generated Eclipse log to support@squoring.com.

Ref : SUM\_Squore\_ECL\_PLG



## 3.3. Squore Eclipse Plugin setup

To use the Squore Eclipse Plugin, it must be activated, the server must be reachable with proper credentials, and rating preferences must be provided.

All these operations are accessible through the **Window > Preferences > Squore** page. Note that all provided information uses the Eclipse secure storage.

Preferences					$\times$
type filter text		Squore	¢	• -	- <b>-</b>
<ul> <li>General</li> <li>Ant</li> <li>Cloud Foundry</li> <li>Code Recommenders</li> <li>Data Management</li> <li>Help</li> </ul>	^	General settings for Squore Eclipse plugin         Environment configuration         Perl binary path:       C:\cli\tools\perl\perl\bin\perl.exe         Tcl binary path:       C:\cli\tools\tclsh\bin\tclsh85.exe			
<ul> <li>&gt; Install/Update</li> <li>&gt; Java</li> <li>&gt; Java EE</li> <li>&gt; Java Persistence</li> <li>&gt; JavaScript</li> </ul>		Server connection         Server url:       http://localhost:8180/SQuORE_Server         Check Server       Server is unreacheable.			
<ul> <li>&gt; JSON</li> <li>&gt; Maven</li> <li>&gt; Mylyn</li> <li>&gt; Oomph</li> <li>&gt; Plug-in Development</li> <li>&gt; Remote Systems</li> </ul>		User connection (optional) User login: demo User password: ******** Check User Server is unreacheable.			
<ul> <li>Run/Debug</li> <li>Server</li> <li>Squore</li> <li>Team</li> <li>Terminal</li> <li>Validation</li> </ul>		Import Configuration from Server (Unsynchronised) Use to retrieve enviro			
> Web	*	Restore <u>D</u> ef	aults	<u>A</u> ppl Cance	

The Squore Eclipse Plugin preferences page

Fill in the paths to the perl and tclsh executables and the server details and click **Check Server** to ensure that your configuration is correct.

If you can successfully connect to the server, click **Import Configuration from Server** to synchronise your environment with the one from Squore Server.

When the synchronisation finishes, click **OK** to close the preferences window.

8



# 4. Getting Started With the Squore Eclipse Plugin

## **4.1. Launching Your First Analysis**

Once the Squore Eclipse Plugin is installed in Eclipse, obtaining a rating for your project is rather straightforward.

First locate the Squore contextual menu entry for the project you want to rate.

dille.	Squore	>	Build
S	Refresh	F5	Build and Baseline
	Close Project		Clean
	Close Unrelated Projects		Build Automatically
	Assign Working Sets		Build Configuration

Contextual menu on an Eclipse project

Rating (or "building") a project with Squore Eclipse Plugin can either be performed as a baseline build or a draft build.

- When choosing baseline build, the project rating will produce a new reference version, with which all subsequent ratings will be compared to. This will affect rating computations based on metrics previous values, as well as all trends displayed in the Squore Eclipse Plugin views. The baseline mode is useful when a stable state of the project has been reached, and a development phase
- is about to begin.
- When choosing draft build, the project is rated as usual, but the reference version is not modified. Each new draft build overwrites the previous analysis and displays trends and deltas relative to the last defined baseline.

The draft mode is useful during a development phase, when the project can undergo several back and forth changes, and the user wants to keep comparing the current version to the last stable one.

Before you can build a project, you first need to define its build configuration:

9

#### http://www.squoring.com/



Squore Build Configuration (kaptcha)	×
Load project parameters from:	
Wizard	~
O Server Project	~
Кеу	Value
	OK Cancel

The Build Configuration screen

On the Build configuration screen, you can choose an analysis model for your project, or you can select the project on the server that corresponds to the code you are working on locally. By selecting an existing project, you can retrieve the project attributes from the server, and therefore run your local analysis in the same conditions as it is run on the server.

10

#### http://www.squoring.com/



Squore Build Co	nfiguration (kaptcha)			×
Load project param	eters from:			
○ Wizard	ANALYTICS			$\sim$
Server Project	kaptcha (software_anal	ytics)		~
Key		Value		^
APPLY_TO_BE_TEST	ED	1.0		
COV_BRANCH_A		0.0		
COV_BRANCH_B		100.0		
COV_BRANCH_C		100.0		
COV_BRANCH_D		100.0		
COV_MCDC_A		0.0		
COV_MCDC_B		0.0		
COV_MCDC_C		0.0		
COV_MCDC_D		100.0		
COV_STAT_A		100.0		
COV_STAT_B		100.0		
COV_STAT_C		100.0		
COV_STAT_D		100.0		
LEVL_THRESHOLD		4.0		
PATH_THRESHOLD		5.0		~
			ОК	Cancel

Project options for the kaptcha project on the server

Click **OK** to close the configuration dialogue. You are now ready to start building your project using the draft or baseline submenu in the Squore Eclipse Plugin context menu.

## 4.2. Visualising Analysis Results

After your analysis completes, switch to the Squore perspective to view the results.

#### Тір

Click **Window > Perspective > Open Perspective > Other** and search for **Squore** in the list of available perspectives if you are using it for the first time.

The Squore perspective displays various views with information relative to the file or folder you click in the Project Explorer.

11



orer ☆ ⓒ ☆ ⓒ ♡ □ □ System Library [JavaSE-1.8] ava > com > ⓒ com > ⓒ code > ⓒ kaptcha > ⓒ core DefaultBackground DefaultBackground DefaultNoisejava @ FishEyeGimpy java ads fish eye effect with vertical and horizontal i * ⓒ code > ⓒ kaptcha > ⓒ core @ DefaultBackground @ DefaultNoisejava @ FishEyeGimpy java ads fish eye effect with vertical and horizontal i * () ⓒ kaptcha > ⓒ core @ DefaultBackground @ DefaultNoisejava @ FishEyeGimpy java @ FishEyeGimpy java @ DefaultNoisejava @ ShadowGimpy java @ WaterRipplejava > ⓒ servlet > ⓒ text > ⓒ te				
Image: Solution of the solution			a - Eclipse	-
<pre>bystem Library [JavaSE-1.8] ava     com     figure Care</pre>			• * 🗘 • 🖒 •	Quick Access
<pre>System Library [JavaSE-1.8] ava com com com com code code code code code code code code</pre>	orer 🛛 🕞 🔄 🕼 🔻 🗖 🗖	DefaultNoise.java FishEyeGim	py.java 🖾	
FishEyeGimpy.java     FishEyeGimpy.java     bugs.xml     b.xml     Measures/Indicators     Value     Line Counting     Line Count     126     Cource Lines Of Code     50     Effective Lines Of Code     51     i     Complexity Distribution     Ratio of complex modules     8.3%     Class Complexity Ratio     0.0%     Number of Classes     1   i   Coding Rules Compliance   99.68%	a System Library [JavaSE-1.8] java com google Solution Google Solution Google Solution Google Solution Google Solution Google Solution Solut	<pre>1 package com.google.code.ka 2 3 import java.awt.Color;[] 8 9 /** 10 * {@link FishEyeGimpy} ac 11 */ 12 public class FishEyeGimpy 13 { 14 /** 15 * Applies distortion 16 * lines. 17 * 18 * @param baseImage th 19 * @return the distort 20 */ 1210 public BufferedImage ge 22 { 23 24 Graphics2D graph = 25 *</pre>	dds fish eye eff implements Gimp by adding fish be base image ed image getDistortedImag	<pre>yEngine eye effect and horizontal vertica e(BufferedImage baseImage) aseImage.getGraphics(); icht();</pre>
bugs.xml   d.xml     Measures/Indicators   Value   Line Counting   Line Count   126   Source Lines Of Code   73   Effective Lines Of Code   51   Executable Statements   42   Cyclomatic Complexity   12   Complexity Distribution   Ratio of complex modules   8.3%   Class Complexity Ratio   0.0%   Number of Classes   1   Coding Rules Compliance   99.68%	resources	FishEyeGimpy.java		
Imeasures/indicators       Value         Line Counting       Image: Counting         Line Count       126         Source Lines Of Code       73         Effective Lines Of Code       51         Effective Lines Of Code       51         Executable Statements       42         Cyclomatic Complexity       12         Complexity Distribution       Image: Complexity Distribution         Ratio of complex modules       8.3%         Number of Modules       3         Class Complexity Ratio       0.0%         Number of Classes       1         Rule Compliance       99.68%	ckstyle.xml lbugs.xml	D		
Line Count       126       i         Source Lines Of Code       73       i         Effective Lines Of Code       51       i         Executable Statements       42       i         Cyclomatic Complexity       12       i         Complexity Distribution       126       i         Ratio of complex modules       8.3%       C         Number of Modules       3       i         Class Complexity Ratio       0.0%       A         Number of Classes       1       i         Rule Compliance       99.68%       A	d.xml		Value	
Source Lines Of Code       73       i         Effective Lines Of Code       51       i         Executable Statements       42       i         Cyclomatic Complexity       12       i         Complexity Distribution           Ratio of complex modules       8.3%       C         Number of Modules       3       i         Class Complexity Ratio       0.0%       A         Number of Classes       1       i         Coding Rules Compliance       99.68%       A		S	126	7
Executable Statements 42   Cyclomatic Complexity 12   Complexity Distribution 1   Ratio of complex modules 8.3%   Number of Modules 3   Class Complexity Ratio 0.0%   Number of Classes 1   Number of Classes 1   Rule Compliance 99.68%				1
Cyclomatic Complexity       12       i         Complexity Distribution       12       i         Ratio of complex modules       8.3%       C         Number of Modules       3       i         Class Complexity Ratio       0.0%       A         Number of Classes       1       i         Coding Rules Compliance       99.68%       A		Effective Lines Of Code	51	i
Complexity Distribution         Ratio of complex modules       8.3%         Number of Modules       3         Class Complexity Ratio       0.0%         Number of Classes       1         Coding Rules Compliance       6         Rule Compliance       99.68%		Executable Statements	42	i
Ratio of complex modules       8.3%       C         Number of Modules       3       i         Class Complexity Ratio       0.0%       A         Number of Classes       1       i         Coding Rules Compliance       99.68%       A		Cyclomatic Complexity	12	<i>i</i>
Number of Modules       3       i         Class Complexity Ratio       0.0%       A         Number of Classes       1       i         Coding Rules Compliance           Rule Compliance       99.68%       A		Complexity Distribution		
Class Complexity Ratio       0.0%       A         Number of Classes       1       i         Coding Rules Compliance       99.68%       A			8.3%	С
Number of Classes     1     i       Coding Rules Compliance     99.68%     A				
Coding Rules Compliance       Rule Compliance       99.68%			0.0%	A
> Rule Compliance 99.68%			1	14
				-
	>	Rule Compliance	99.68%	A
				4

The Squore Perspective with the Squore Card, Action items, Findings and Highlights views in the bottom panel

Ref : SUM\_Squore\_ECL\_PLG

12

#### http://www.squoring.com/



Click a file or folder in the Project Explorer and use one of the following views in the bottom panel to understand your rating:

#### → The Squore Card view

displays the overall rating for the current artefact, and its trend. It also lists the tables of interests for this artefact, as defined in the Analysis Model used to rate the project.

📣 Squore Card 🔀 🏄 Action Items	🏭 Findings 🏄 Hig	hlights 📃 Console
FishEyeGimpy.java		
D		
Measures/Indicators	Value	
Line Counting		
Line Count	126	<i>i</i>
Source Lines Of Code	73	<i>i</i>
Effective Lines Of Code	51	<i>i</i>
Executable Statements	42	<i>i</i>
Cyclomatic Complexity	12	<i>i</i>
Complexity Distribution		
Ratio of complex modules	8.3%	С
Number of Modules	3	i
Class Complexity Ratio	0.0%	Α
Number of Classes	1	<i>i</i>
Coding Rules Compliance		
Rule Compliance	99.68%	А
Non Conformity Count	6 violations	1
Non Compliant Standards	2 rules	1
Coding Standards	618 rules	1
Issue Distribution		
Blocker Issues	0	1
Critical Issues	0	1
Major Issues	0	1
Minor Issues	6	1
The S	quore card	

→ The Findings view

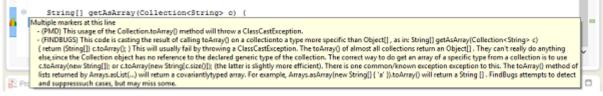
lists the Findings for the current artefact as well as underlying artefacts, with their location and definition details. This list can be sorted, and allows navigation to artefacts associated to each finding by double-clicking on their name.

#### http://www.squoring.com/



<pre>int i = 0; i = 1 / 0; if (i == 0) return }</pre>			(SQUORE) A function shall have a single point of exit at the end (see [MISRA-	-C:2004		
if (1 0)			(SQUORE) A function shall have a single point of exit at the end (see [MISRA-	-C:2004		
			(SQUORE) A function shall have a single point of exit at the end (see [MISRA-	-C:2004		
return }	17		(SQUORE) A function shall have a single point of exit at the end (see [MISRA-	C:2004		_
3					4): RULE 14.7)	Ŀ
						1
return i;						
1						
,						
String[] getAsJ	Array(Collecti	ion <string> c) {</string>				
return (Str	ring[]) c.toAr	erman() -				
		rray(),				
2		1103(),				
<					>	4
<					>	*
<					>	~ 1
< roblems @ Javadoc 🔯 D			🔥 Action Items 🛛 Task List		>	>
			🔥 Action items 🛛 Task List		÷	>
			🔥 Action items 🛛 Task List		÷	~
st.java (Total: 3; New: 3)	Declaration 🔤 Prog	ogress 👗 Findings 🕄				>
				New	DP	>
st.java (Total: 3; New: 3) rtefact	Declaration =;;; Prog	ogress 👗 Findings 🛙 Pract	ice	New true		>
st.java (Total: 3; New: 3) rtefact etAsArray(Collection <string>)</string>	Line Rule	rgress 🔺 Findings 🛙 Pract	ice usage of the Collection.toArray() method will throw a ClassCastException.		DP	
st.java (Total: 3; New: 3)	Line Rule	egress 🏄 Findings 🛙 Pract t Exception With This ssible downcast This	ice usage of the Collection.toArray() method will throw a ClassCastException. code is casting the result of calling toArray() on a collectionto a type more specifi	true	DP PMD	>

#### A finding and its location in the source code



A finding full description

→ The Action Items view

works much like the Findings one, except it lists the Action Items computed by the Decision Model, and the reason why they have been generated.

est.java (Total: 3)							
Artefact	Id	Name	Scope	Status	Reasons	Priority	
getAsArray(Collection <s< td=""><td>-351.0</td><td>Number of Blocker and critical rule increased</td><td>[Function]</td><td>OPEN</td><td>New artefact should be reviewed due to</td><td>High</td><td></td></s<>	-351.0	Number of Blocker and critical rule increased	[Function]	OPEN	New artefact should be reviewed due to	High	
getAsArray(Collection <s< td=""><td>-351.2</td><td>No 'Blocker' rules</td><td>[Function]</td><td>OPEN</td><td>'Blocker' rules (=1) detected in function.</td><td>High</td><td></td></s<>	-351.2	No 'Blocker' rules	[Function]	OPEN	'Blocker' rules (=1) detected in function.	High	
getAsArray(Collection <s< td=""><td>-351.1</td><td>Number of high and major rule increased</td><td>[Function]</td><td>OPEN</td><td>New Artefact should be reviewed due to</td><td>Medi</td><td></td></s<>	-351.1	Number of high and major rule increased	[Function]	OPEN	New Artefact should be reviewed due to	Medi	

→ The Highlights view

works much like the Findings one, except it provides a list of the current project's Files, Classes or Functions, sorted by decreasing rating. Selecting the type of object to sort (File, Class or Function) is performed through the top-left menu icons.

14



Problems 🛛 @ Javadoc 🔅 Decl				-
st worst methods (Total: 171)				
Artefact		Line		
doXMLFinding	С	310		
endElement	С	570		
getLocStr	С	58		
startElement	С	549		
CoreArtefact	в	76		
addData	в	179		
addDefectReports	в	115		
addDefectReports	в	219		
addDrLinksTestsDependencies	в	551		
addInformation	в	489		
addLocation	в	56		
addLogicalFatherStr	в	482		
addSon	в	428		

Highlights

#### Тір

Only the highlights created for the role **PLUGIN** are displayed in Eclipse. For more information about configuring highlights in your model, refer to the Configuration Guide.

15

#### http://www.squoring.com/



# Index

### A

Action Items view, 14

## B

Baseline build, 9

## D

Disk Space, 5 Draft build, 9

## F

Findings view, 13

### Η

Highlights view, 14

## J

Java, 5

### Μ

Memory, 5, 5

## Р

Prerequisites, 5

## S

Squore Card view, 13

#### http://www.squoring.com/