What Is Java Sonar?

Java Sonar is becoming an increasingly popular tool for managing code and software quality. Make sure that you understand the basics of what it is, and how it works with today's blog post.

What is Java Sonar?

Java Sonar is an open source project that allows you to manage code quality with little effort. With this web-based program's analysis, **it is easy to find projects that are in technical debt and find solutions.**

Using statistical code analysis tools, Java Sonar combines metrics together and reports on these standards of code quality:

- Duplicated code
- Coding Standards
- Comments
- Unit Tests
- Code Coverage
- Design and architecture
- Complex code
- Potential bugs

While Java is the primary supported language of Sonar (hence Java Sonar), C, C#, Flex, PHP, PL/SQL, Cobol, Visual Basic 6, and Python are also supported with plugins.

How Sonar Works

Despite the complexity of the data collected, Sonar has a fairly simple architecture, which is comprised of three parts.

1. A Set of Code Analyzers

These code analyzers are grouped in a Maven plugin, and run on demand.

Through they work through a Maven plugin, these code analyzers are able to run Maven and non-Maven projects.

Popular Code Analyzers are:

- PMD, CheckStyle for locating violations of coding rules
- Findbugs for finding potential bugs
- Cobertura, Clover for measuring unit test coverage
- Squid for code analysis

2. Database

The results of the analysis come back to a database. They are stored here, along with historical analyses and project configurations. Common database engines for this are Oracle, MySQL, PostgreSQL, MS SQLServer.

3. Web Reporting Tool

There is a web reporting tool that allows you to see the quality of your projects. In this tool, there are many applications, especially with the use of plugins, but three of the most important are:

1. Dashboards

These dashboards give software quality information on all of your projects combined, as well as on each individual one. The information for the later can be obtained on the packages and modules level.

2. Defect Hunting Tools

This tool allows you to get a better look at what is reported as an issue. For example, you can have a multi-entry source viewer verify that the issues reported by the defect hunting tool are truly defects.

3. TimeMachine

TimeMachine records analytics of the project over time, so that you can watch how a project has evolved. This allows you to understand whether you had, for example, 30% or 50% of code coverage this time last month, year, or 5 years ago. Looking at these analytics allows you to understand how different approaches affect your project.

While Java Sonar gives you all of the information you could possibly need about the quality of your code, it boasts a fairly simplistic architecture. **A set of code analyzers, a database, and a web reporting tool give you all the information you need to know.**

Have you used Java Sonar in the past? Would you recommend it to others? Let us know in the comments section below, or join the conversation on Facebook (http://www.facebook.com/Base36), Twitter (https://twitter.com/Base36Inc), and LinkedIn (http://www.linkedin.com/company/base36-inc).

Thanks to mikecogh (http://www.flickr.com/photos/mikecogh/3086805262/sizes/s/in/photostream/) for the use of their respective photographs.

← 10 Tips for Telecommuting Successfully (http://www.base36.com/2012/11/10-tips-for-telecommuting-successfully/)

7 Reasons IT Projects Fail → (http://www.base36.com/2012/11/7-reasons-it-projects-fail/)

CONNECT WITH BASE36

(http://twitter.com/Base36Inc) (http://www.facebook.com/Base36) (http://www.linkedin.com/company/1441376?goback=%2Efcs_GLHD_base36_false_*2_*2_*2_*2_*2_*2_*2_*2_*2_*2_*2_*2&trk=ncsrch_hits) (https://plus.google.com/106783988067872038764) (http://base36.com/feed)

110 Main Street
Suite 1304
Saco, ME 04072
info@base36.com (mailto:info@base36.com)
PH: 207-221-3147
F: 207-221-1158