Scaling Clean Code Across the Enterprise

Applying the right tools to solve real-world problems

The practice of Clean as You Code helps ensure that your software continues to be an asset and is a key driver for your business success. Clean as You Code from Sonar is vital for achieving a Clean Code state -- when your codebase has reached a problem-free state and is fit for development and fit for production.

Sonar enables you to implement Clean as You Code standards that scale across your enterprise, ensuring that your code is always clean. Here is a closer look at the unique, enterprise-grade capabilities that Sonar offers and how they are applied to solve real-world problems with some of our own customers.

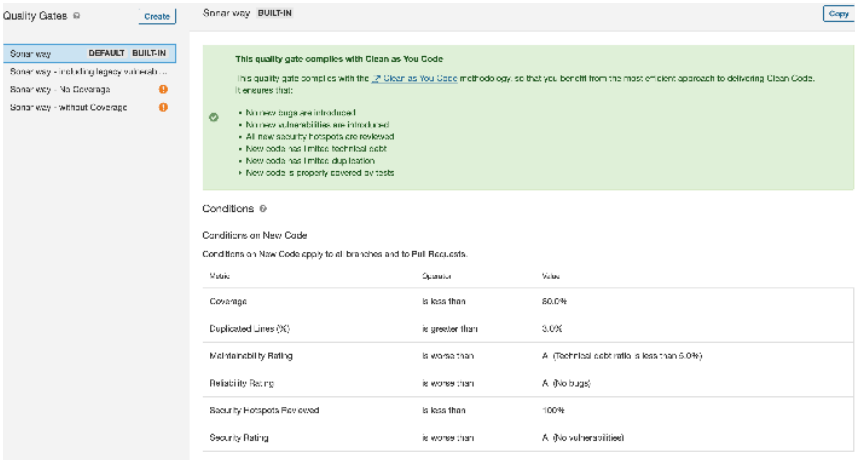
Quality gates

- **Problem**

  A logistics company has been using static analysis tools for several years. While they were getting value out of the tools, they found that many serious bugs and vulnerabilities were making their way into production, even though they were being detected. Developers were unsure of what issues needed to be fixed prior to release, and it was difficult to separate newly-created issues from the ones from previous development cycles. They needed a way to define and enforce a standard that worked with their process.

- **Solution**

  The company adopted SonarQube Server **Quality Gates** which focuses on newly developed and changed code. They set the Quality Gate conditions to enforce what they expected their developers to resolve immediately. They began enforcing this standard on each Pull Request submitted as well as on release branches. With this process in place, they became confident that newly released code was meeting their expected standards. Their codebases are improving over time as low quality code is replaced with Clean Code.



Quality profiles

- **Problem**

  A furniture retailer has been using a variety of static analysis tools to check the quality and security of their code. Teams were able to choose what tools they wanted to use and the ability to customize the level of testing that took place. Over time, each team adopted a large number of tools and the testing performed on their code during their development cycles showed varied results. It became extremely difficult for developers and teams to determine the quality and security of their code. Some teams adopted extensive testing which detected large amounts of issues, whereas others were using few or no tools and few problems were raised. When comparing different codebases, it became impossible to understand which represented high quality code. Developers that moved between teams required additional time to adopt the varying standards, resulting in a loss of productivity.

- **Solution**

  The organization standardized its testing using SonarQube Server and adopted the “Sonar Way” **Quality Profile** for each of the languages it was working in. Quality Profiles define the set of rules that are applied during code analysis. The company also established a process to gather and review developer feedback so that future changes would be uniformly adopted across all teams. These changes resulted in a clear picture of each team’s code quality across the organization.

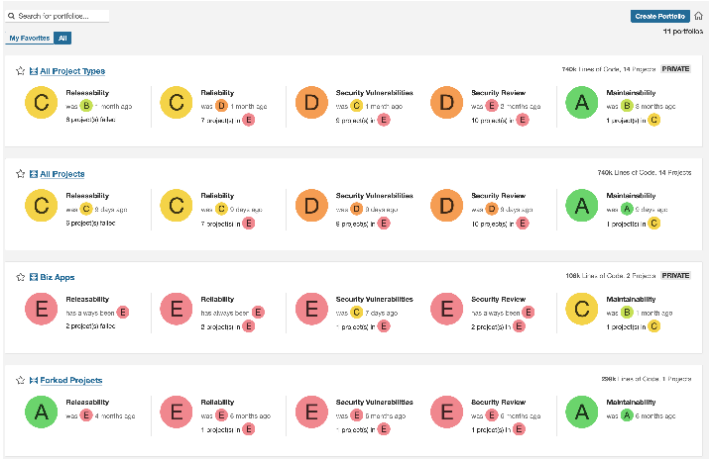
Enterprise reporting

- **Problem**

  A software consulting firm had a mandate from its largest client to demonstrate that it was delivering high quality and secure code. This included showing that the code was free of serious bugs and vulnerabilities and that an adequate level of automated testing was taking place. They were working on dozens of software applications for the client, and components were shared in some cases. To meet this requirement, an analyst spent several days each month gathering reports and tools from various teams and projects.

- **Solution**

  The firm adopted SonarQube Server’s **enterprise reporting** to fulfill this requirement automatically. They created software portfolios that lined up with each of the application’s repositories and automatically had up-to-date multi-view project reporting on the health of these codebases. In addition to saving time, they are able to show a consistent standard of quality across a large number of repositories.



Notifications

- **Problem**

  An international financial institution has been scanning its software with SonarQube Server for several years and was finding serious issues in its code. However, they found that many issues were not getting fixed. When discussing the issue with the development teams, they found that most developers in their organization had no way of knowing if or when their code was not meeting the expected standards. Due to the lack of awareness, issues continued to go unresolved and technical debt grew within their codebase.

- **Solution**

  This institution started taking advantage of several **notification mechanisms** within SonarQube Server to build awareness of the quality of their software. They added badges to project pages to let developers know the status of the code's Quality Gate. They encouraged developers to install and use SonarQube for IDE to highlight issues within their IDEs so developers could fix them before committing. They began scanning Pull Requests and posting the results to the comments in the source control. With these notification techniques in place, developers began to resolve issues earlier in the development process, resulting in higher quality code.

Conclusion

The Clean as You Code methodology from Sonar can be leveraged across your enterprise regardless of the size of your organization, software maturity, level of developer experience, or internal complexity. By implementing and communicating Clean as You Code standards using the capabilities described above, development and production teams can keep your new code clean and ensure that your software continues to be an asset and is key to your business success.