



User Story: PlantCML Speeds Up Code Review Process with Code Collaborator

By Hussein Said, Senior Software Architect, PlantCML

Code Collaborator eliminated the manual aspects of the code review process for us. Now we don't have to fill out review forms, track comments and defects manually, and waste a lot of time in unproductive meetings. Instead, we spend our time doing better reviews.

PlantCML®, an EADS North America company, is the industry's leading provider of crisis communications and response technologies. For over 40 years, PlantCML has led the industry in providing mission-critical 9-1-1 emergency response and systems management solutions. Today, PlantCML is pioneering the next generation of E9-1-1 and secure network communications through Voice over IP-enabled applications, interoperable P25 radio systems and advanced emergency notification technologies. PlantCML's solutions are resident in 14 of the top 20 most populous U.S. cities and largest first responder operations.

As Senior Software Architect, I'm responsible for providing software and systems architecture for some of the PlantCML 9-1-1 call-taking and MIS platforms for the Public Safety Answering Points (PSAPs). We are developing a decision support system (DSS) for PSAPs: a .NET application with enterprise capabilities whose main purpose is data mining and reporting. Our development team spans two time zones. We are applying agile methodologies on this project, so we have scrum masters and a strong team of developers.

Code reviews are a company-wide policy for us, because we know how much more efficient it is to find bugs while the product is still in development and because software quality is so critical in our industry.

Before trying Code Collaborator, we used to review in excess of 500 lines of code in a room with a projector, where everyone was expected to bring their comments on the "diffs", which we generated using CSDiff and sent out prior to the meeting. During the meeting, we walked through the code together and everyone participated to ensure that the code was bug free. But this formal process takes too long and was not productive; we had no mechanism to provide comments directly on the lines of codes prior to or during the meeting. After the meeting, the author of the code would fill out a form with the results of the code review, but he or she usually found it difficult to capture all comments and issues on this form. In addition, subsequent meetings triggered either a form update or a new form to be completed, a tedious

Platform: Windows, Linux

Source Control Management Software: Perforce®

Development Environment: .NET, C#, C/C++, Embedded C/C++

Industry: 9-1-1 Emergency Response and Systems Management Solutions

The Challenge: Eliminating manual activities and inefficient formal meetings so developers can focus on the code reviews themselves.

"We like how easy Code Collaborator is to use. It lets us focus on the code inspection tasks."

manual process that makes it difficult to ensure that all issues have been addressed.

Since our formal process was not efficient, we looked for a better way to do code reviews using a tool. Our goals were to increase the productivity of the teams, allow online collaboration, and provide a single repository for reviews that can track issues in our defect tracking tool. We evaluated several code review tools and came to the conclusion that Smart Bear has the advantage over the other tools we evaluated. We decided to try Code Collaborator out on a fresh new project and started using it in trials at the end of 2008.

In modern software development, finding bugs is critical before software ships, but ensuring that code is well-formatted and documented is equally important. Writing clear software helps everyone. With Code Collaborator, we can easily exchange code via the Upload and Download features, and developers suggest improvements for the code, formatting, and documentation right on the code itself through online live chat and news-group-style comments. Defects are automatically tracked, and diffs on re-worked code show up automatically too. You cannot do that on paper!

During implementation, the only challenges we had were *how* (peer vs. formal) and *when* (before code check-in or after code check-in) to do reviews, but we found that each team conducts code reviews to suit their needs. Code Collaborator is flexible enough that it easily supports all the different workflows that we use.

We're very pleased with the results. Code Collaborator is a complete package, and now that it's in place, the manual code review process has been eliminated. All reviews are conducted online, where we find the inline comments and live chat correlated to each line of code particularly helpful. We like how easy Code Collaborator is to use – it lets us focus on the code inspection tasks.

The tool's integration with Perforce and configurable workflow also save us significant time and help us follow processes that work best for us. Code Collaborator gives us the option to enforce code reviews before code is committed to source control, thus raising the level and importance of code reviews to "first class citizens." Of course, this enforcement option can be turned on or off at various stages of development, giving us the flexibility to schedule code reviews as needed.

I wanted our team to spend more time reviewing code and less time filling in paper work. The time we saved by eliminating the manual process is time we now spend improving the quality of the code. We estimate that Code Collaborator will pay for itself in a matter of months based on the time savings it provides.

For more information on Code Collaborator, contact Smart Bear Software today!

Phone: +1 877.501.5651

Email: info@smartbear.com

www.CodeCollaborator.com

"We estimate the tool will pay for itself in a matter of months based on the time savings it provides."

"I wanted our team to spend more time reviewing code and less time filling in paper work. The time we saved by eliminating the manual process is time we now spend improving the quality of the code."

"Smart Bear has the advantage over the other tools we evaluated."

"We can easily exchange code via the Upload and Download features, and developers suggest improvements for the code, formatting, and documentation right on the code itself... You cannot do that on paper!"