What is Agile Application Lifecycle Management?

OVERVIEW

Agile Application Lifecycle Management (Agile ALM) is a central platform that allows teams using Agile methods, alone or in combination with other development methodologies (e.g., RUP, waterfall), to manage all phases of the software development lifecycle from requirements through release. By uniting business users and developers and providing cross-team visibility, Agile ALM enables organizations to achieve a faster time to market and higher-quality software releases while reducing infrastructure costs.

ALM BACKGROUND

Application Lifecycle Management (ALM) is the management of the software application lifecycle from initial development to final release. ALM encompasses all of the practices, processes, and tools that aid in managing an application’s lifecycle from both a business and development perspective. Key capabilities of an ALM platform include the ability to handle change management, workflow, source code management, task management, testing and bug tracking, lab management, reporting and analytics. To facilitate seamless distributed development, an ALM platform should also include a central repository “in the cloud” for managing all of the various types of content created (i.e. code, tasks, roles, requirements, and other artifacts) as well as a system for establishing traceability and accountability across the ALM platform’s many processes, locations, and tool types.

Given the complexity and the magnitude of time and resources a company invests in software development, managing risk becomes extremely important. All software development contains inherent risk, since software development is not grounded in mathematical or physical certainty, but rather based on innovation, discovery, and artistry. Unlike a manufacturing system where processes can be automated, the software development process cannot be translated into a one-size-fits-all, cookie-cutter solution. At the outset, it is impossible to predict any and all of the variables that could possibly impact a software project.

In the past, development teams have relied mainly on the "waterfall" approach to manage the ALM. However, much like the manufacturing assembly line, the waterfall method is a "linear" approach where the product requirements are gathered upfront and the development process follows a predefined sequence of events from coding to testing to QA and then release. The risk in following this type of approach is that it increases the risk of project failure. By the time the final product is released, it may no longer be relevant to the market for which it was designed.
WHY AGILE? WHY SCRUM?

In order to reduce the risk of project failure, software development teams are constantly looking for ways to improve both the process and technology of software development. No software system is so simple that the entire development can be entirely scripted from beginning to end in a linear fashion, which is why we are seeing a shift in the market to "Agile ALM". Simply stated, both the ALM and Agile communities are focused on improving the current state of software development. While ALM approaches the challenge from a technology point of view, Agile is focused on improving the "process". Because of the common goals and synergies of ALM and Agile, CollabNet and Danube joined forces to integrate the CollabNet TeamForge™ platform and ScrumWorks®.

Unlike the "linear" waterfall method, Agile teams use an "iterative" or "inspect and adapt" approach to ALM to address the changing requirements, complexities and risk factors that will arise over the course of the software project. Scrum, the most popular implementation of the Agile movement, has seen exponential growth in the past few years for both small and large scale development projects, and is now rapidly expanding its footprint in the enterprise. Scrum's success, for the most part, is due to its focus on enabling "high feature value" creation, as well as, efficient collaboration among self-organizing teams. Teams using Scrum see Scrum as a benefit because they realize business ROI sooner, minimize project rework, and accelerate product innovation.

PROJECT FLOW OF SCRUM

ABOUT COLLABNET

CollabNet leads the industry in Agile application lifecycle management (Agile ALM) in the Cloud. The CollabNet TeamForge™ ALM platform, CollabNet Subversion software configuration management (SCM) solution, and ScrumWorks® project and program management software enable teams using any environment, methodology, and technology to increase productivity by up to 50% and reduce the cost of software development by up to 80%. The company also offers training, including Certified ScrumMaster training, software development process improvement services, and an innovative community management approach to driving enterprise development success. As the founder of the open source Subversion project, CollabNet has collaborative development for distributed teams in its DNA. Millions of users at more than 2,500 organizations, including Applied Biosystems, Capgemini, Deutsche Bank, Reuters, and the U.S. Department of Defense, have transformed the way they develop software with CollabNet. For more information, visit www.collab.net.