

Jama Integration With GitLab

The integration of Jama with GitLab ensures completely traceability of all work-items. With this integration, the product management team can easily track commit trends and volume.

Integration overview

In an Application Lifecycle Management (ALM) ecosystem, the choice of systems and the collaboration between the cross-functional teams play a great role. While the choice of systems impacts the productivity of a team, the cross-functional collaboration brings in collective wisdom to take better decisions, faster.

Best-of-breed systems such as GitLab and Jama bring rich functionalities to the ecosystem and make the work of the product and development team easier.



How OpsHub Integration Manager integrates Jama and GitLab

OpsHub Integration Manager integrates all commit related information from GitLab to Jama. It ensures that all data is available to each user, in that user's preferred system, with full context, in real-time. All the details related to a commit can be synced to Jama in various ways, following are a few examples:

- Synchronize every commit as a separate workitem linked to the workitem against which commit happens
- Synchronize commit details as comment to the workitem against which commit happens
- Change workitem field like 'close a workitem' when commit comments contain a pre-defined identifier

How Jama - GitLab integration is beneficial for an enterprise

- Shorten the delivery lifecycle, streamline manual processes and accelerate team velocity
- Track commit volume, track commit trends and edits/changes to commit files in real time
- Enforce authentic commits to make sure each commit is happening against a scheduled and open workitem

Commonly synchronized entities between Jama and GitLab



Benefits of integration for Jama and GitLab users

Jama users

GitLab users

Complete traceability from Jama to source code in GitLab	Each commit can be traced back to its respective workitem at any given point in time from GitLab itself
Visibility into the progress of development work, the volume and quality of commits made with full context, in real-time	Enforced checkpoints ensure that no mandatory steps/checks are missed while making a commit – this leads to high success rate for commits
Reduced dependency on manual communication to track the completion of a task	

Features of OpsHub Integration Manager



Bi-directional sync with conflict resolution



Support for the largest number of entities



Database-class reliability and recovery



Support from 50+ systems and growing



History preservation and Process customization



Pre-requisites to run OpsHub Integration Manager

Supported Operating Systems

Windows

- Windows Server 2016
- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2 (64 bit)

Linux

- RHEL 5.2 + (64 bit)
- RHEL includes Cent OS and Fedora

Tested on the following versions:

- CentOS release 5.5 (Final)
- CentOS release 5.6 (Final)
- CentOS Linux release 7.1.1503 (Core)
- Fedora 20

Database Prerequisites

The underlying database should be installed to install and run OpsHub Integration Manager. The database user created for OpsHub Integration Manager should have schema level and read write privileges.

- MySQL Server
- MS SQL
- Oracle
- HSQLDB