

OpsHub Integration Manager for Atlassian JIRA and Rally



System Type: Issue & Project Tracking Product

System Description: Atlassian JIRA is the issue tracker, bug tracker and project tracker for teams planning, building, and launching great products. JIRA offers both on-premises and on-demand deployment offering customer full flexibility on deployment options. Thousands of teams choose JIRA to capture and organize issues, assign work, and follow team activity. At their desks, or on the go with the new mobile interface, JIRA helps teams get the job done.

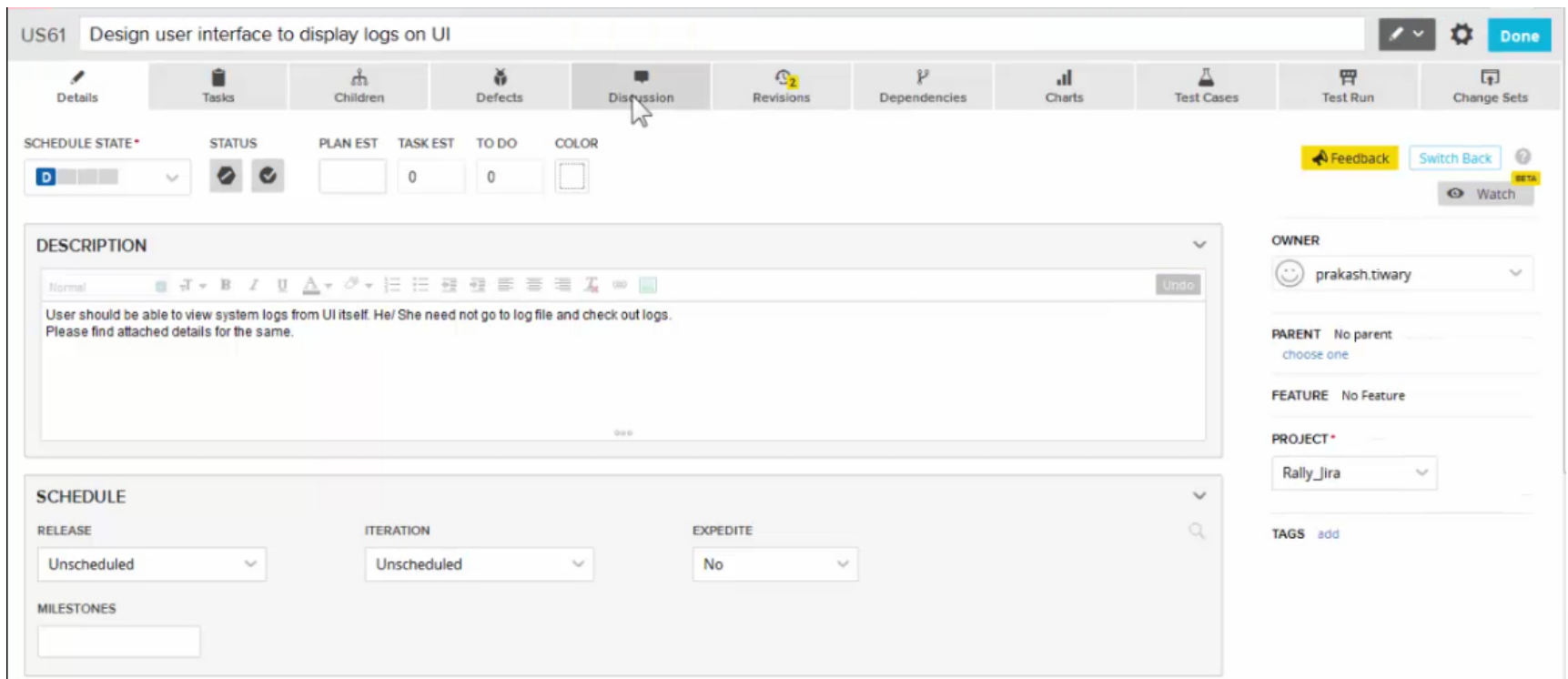
Integration Description: OpsHub Integration Manager (OIM) allows the cross-functional teams using disparate tools like Atlassian JIRA and Rally in application development organizations to effectively communicate and collaborate with each other, thus increasing overall team agility, productivity and efficiency.

OIM supports the synchronization of Issue, Bug, Requirement, Improvement, Sub Task and all other issue types available in JIRA to all the possible entities existing in Rally viz., Defects, User Story, Test cases, Test sets, Test Case Result (Only write support), Task.

- Supports Unidirectional and Bidirectional synchronization between JIRA and Rally for all supported entities.
- The synchronization includes the field data, attachments, comments and issue-relationship traceability.
- Provides deployment options to fit the needs of business whether On-Premise or on Customer Cloud.
- Fully manageable and functional via the web based administration tool including defining entities for synchronization between JIRA and Rally.
- Maintains history of both systems. If any of JIRA or Rally becomes unavailable, it will reflect all transactions that occurred during that time once the system is operational again.
- Provides with templates and ability to define custom mapping between ALM systems to support mapping of fields and linkages.
- Supports number of replication models to support development organization such as master/slave, partitioned ownership, dynamic ownership, peer-to-peer and custom.
- Proactive conflict management with support for automated conflict resolution.

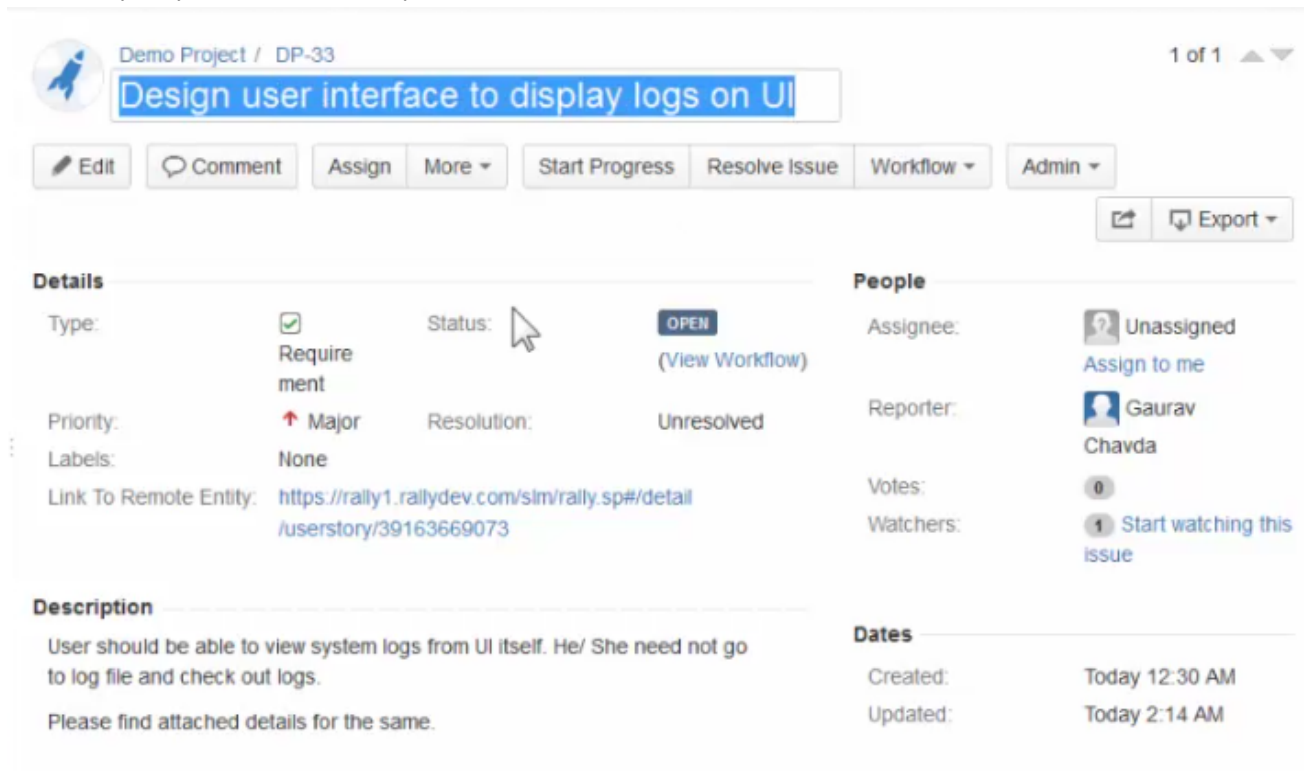
Integration Screenshot:

User creates a User Story in Rally



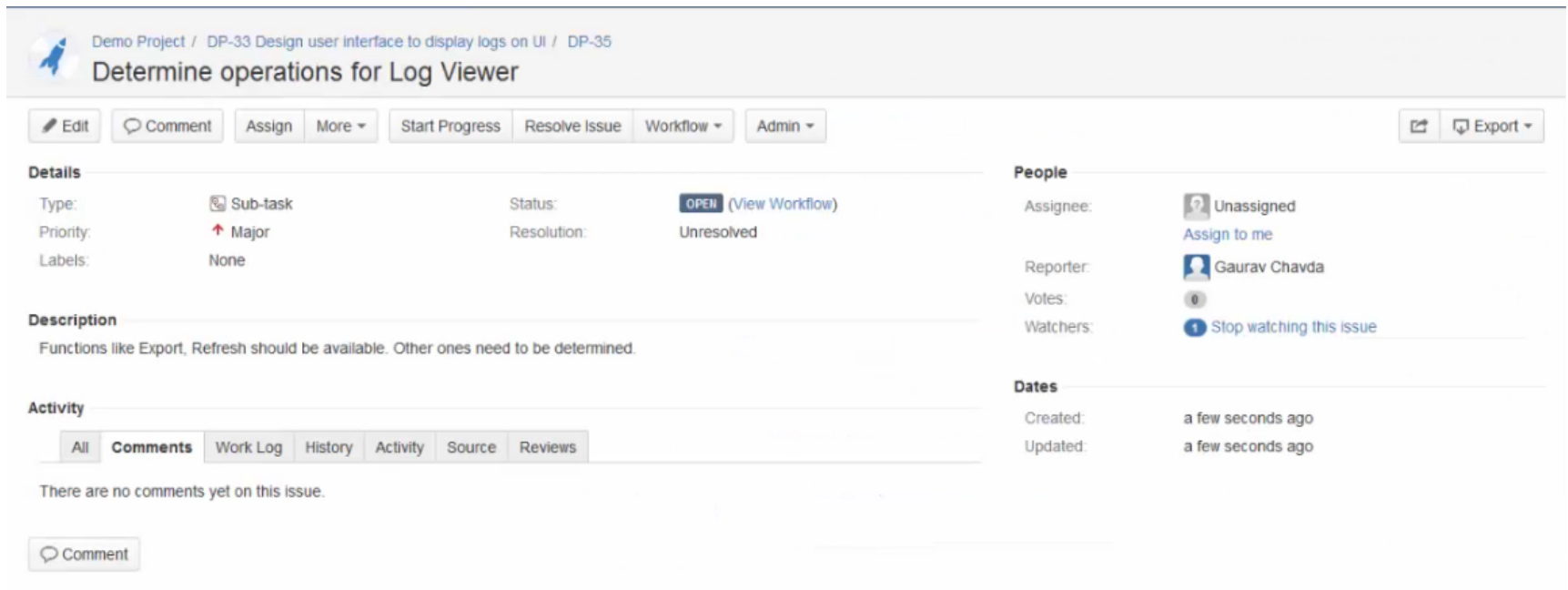
The screenshot shows the Rally user story creation interface. The title is "US61 Design user interface to display logs on UI". The interface includes a navigation bar with tabs for Details, Tasks, Children, Defects, Discussion, Revisions, Dependencies, Charts, Test Cases, Test Run, and Change Sets. Below the navigation bar, there are fields for SCHEDULE STATE, STATUS, PLAN EST, TASK EST, TO DO, and COLOR. The DESCRIPTION field contains the text: "User should be able to view system logs from UI itself. He/ She need not go to log file and check out logs. Please find attached details for the same." The SCHEDULE section includes fields for RELEASE (Unscheduled), ITERATION (Unscheduled), and EXPEDITE (No). The right sidebar shows the OWNER (prakash.tiwary), PARENT (No parent), FEATURE (No Feature), PROJECT (Rally_Jira), and TAGS (add).

User Story is synced to Jira as Requirement



The screenshot shows the Jira requirement interface. The title is "Design user interface to display logs on UI". The interface includes a navigation bar with buttons for Edit, Comment, Assign, More, Start Progress, Resolve Issue, Workflow, and Admin. Below the navigation bar, there are sections for Details, People, Description, and Dates. The Details section shows Type: Requirement, Status: OPEN, Priority: Major, Resolution: Unresolved, and Labels: None. The People section shows Assignee: Unassigned, Reporter: Gaurav Chavda, Votes: 0, and Watchers: 1. The Description section contains the text: "User should be able to view system logs from UI itself. He/ She need not go to log file and check out logs. Please find attached details for the same." The Dates section shows Created: Today 12:30 AM and Updated: Today 2:14 AM.

Developer divides Requirement in Jira into Sub Tasks



Demo Project / DP-33 Design user interface to display logs on UI / DP-35

Determine operations for Log Viewer

Edit Comment Assign More Start Progress Resolve Issue Workflow Admin Export

Details

Type: Sub-task Status: **OPEN** (View Workflow)
 Priority: Major Resolution: Unresolved
 Labels: None

Description

Functions like Export, Refresh should be available. Other ones need to be determined.

Activity

All **Comments** Work Log History Activity Source Reviews

There are no comments yet on this issue.

Comment

People

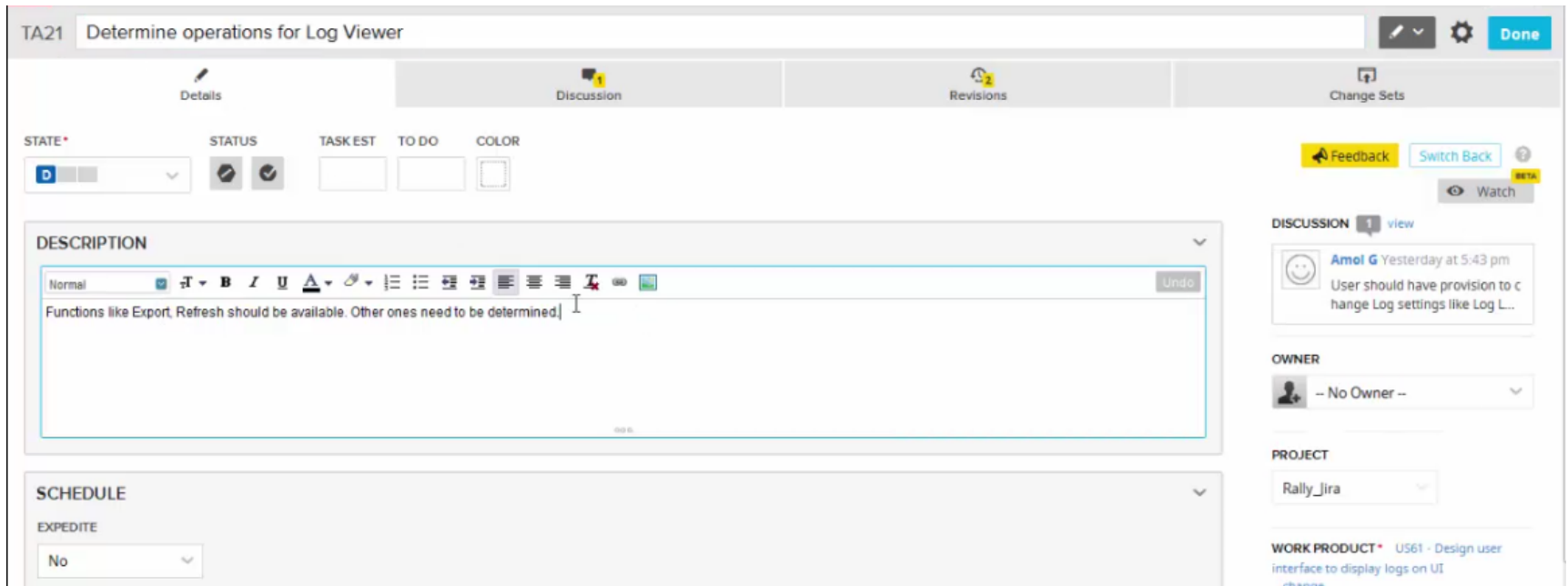
Assignee: Unassigned
Assign to me

Reporter: Gaurav Chavda
 Votes: 0
 Watchers: 1 Stop watching this issue

Dates

Created: a few seconds ago
 Updated: a few seconds ago

Sub Task synced to Rally as Task through OpsHub



TA21 Determine operations for Log Viewer

Details Discussion Revisions Change Sets

STATE: D STATUS: TASK EST: TO DO: COLOR:

DESCRIPTION

Normal Undo

Functions like Export, Refresh should be available. Other ones need to be determined.]

SCHEDULE

EXPEDITE: No

DISCUSSION 1 view

Feedback Switch Back Watch

Amol G Yesterday at 5:43 pm
 User should have provision to change Log settings like Log L...

OWNER

-- No Owner --

PROJECT

Rally_Jira

WORK PRODUCT U561 - Design user interface to display logs on UI change

Integration Demo:

The video show cases the integration of two systems and how easily a user can configure integration and maintain it. The integration flow shown in video highlights the integration of JIRA Requirement, Bug and Sub Task and Rally User Story, Defect and Sub Task. The video also shows how any comment or attachment added in either system is synced to the other system, how the developer working in Rally link User Stories and Defects and the same relationship is reflected in JIRA. And once the manager acknowledges the requirement and accept the requirement in Rally, the same is synced to JIRA and hence the backend team and manager gets instant visibility of the updates.

Video Link: http://www.opshub.com/ohftp/Resources/RallyPartnershipVideos/Rally_Jira.mp4