



Getting More from Jira Software Server

Lab Workbook

Copyright © 2018 Atlassian Pty Ltd. All rights reserved.

Atlassian ®, Atlassian Bamboo ®, Atlassian Clover ®, Atlassian Confluence ®, Atlassian Crowd ®, Atlassian Crucible ®, Atlassian Marketplace ®, Bitbucket ®, Confluence ®, Confluence Questions ®, Fisheye ®, Hipchat ®, Jira ®, Jira Portfolio ®, Jira Service Desk ®, Sourcetree ®, Statuspage ®, Stride ™, Trello ®, ▲ **ATLASSIAN** ®, and ▲ ® are trademarks of Atlassian Pty Ltd. All other trademarks and trade names are the property of their respective owners. Information in this document is subject to change without notice. Websites, names, and data used as examples in this document are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording by any information or retrieval system, for any purpose, without prior written permission of Atlassian Pty Ltd. Address inquiries to Atlassian Pty Ltd, Level 6, 341 George Street, Sydney, NSW 2000 Australia.

Table of Contents

Introduction	i
Lab 2 – Exploring Jira Software Integration	2
Scenario	2
Exercise 1 – Exploring Development Tool Integration	2
Exercise 2 – Exploring Confluence Integration	6
Lab 3 – Using Scrum to Manage the Feedback Loop	12
Scenario	12
Exercise 1 –Triaging the Backlog	13
Exercise 2 – Organizing and Prioritizing Issues According to the Roadmap	15
Lab 4 – Setting up the Scrum Board	22
Scenario	22
Exercise 1 – Creating and Configuring the Scrum Board	22
Exercise 2 – Configuring Project Roles and Project Links	28
Lab 5 - Using Estimation and Tracking	37
Scenario	37
Exercise 1 – Configuring Estimation and Setting Original Estimates	37
Exercise 2 – Logging Time	40
Lab 6 - Sprints for Complex Projects	45
Scenario	45
Exercise 1 – Working with Sub-tasks	45
Exercise 2 – Creating a Scrum of Scrums Board	54
Lab 7 – Managing Releases	59
Scenario	59
Exercise 1 – Exploring the Release Hub	59
Lab 8 – Using Reports and Dashboards	66
Scenario	66
Exercise 1 – Selecting the Appropriate Report	67
Exercise 2 - Experimenting with Jira Dashboard Gadgets	71

Introduction

Let's take a minute to review how this Lab Workbook is organized. You'll find instructions for each lab in this course. Each lab has the following sections:

- **Scenario:** Describes what you will do in the lab. This section lays out the use case and other details you need to know.
- **High-level Instructions:** Challenging, less detailed instructions if you like figuring things out on your own.
- **Detailed Instructions:** If you prefer to follow step-by-step instructions, use the more detailed instructions.

There are also appendices, which you don't need during class. They are full of useful information like additional technical detail, best practices and additional reading – all conveniently grouped by lab. Dig into this section after class!

Lab access and login instructions follow:

- Access to Jira is explained in a handout called "Accessing Atlassian in CloudShare" PDF document – please find this document in your on-line registration and follow the instructions.
- Login credentials are found throughout the instructions.

Lab 2 – Exploring Jira Software Integration

Scenario

In this lab, you examine the development tool data that is provided by Jira Software in the development panel.

The instance of Jira Software you’re accessing in the lab environment is integrated with Bitbucket (a repository management tool), Bamboo (a Continuous Integration tool), and Confluence (a collaboration tool).

You will be logging in as Alana Grant. Alana is the scrum master and wants to check on the team’s development progress. She is interested in what work is underway (on a Git branch), what work has been committed to the repository, and what code reviews (pull requests) are in progress or approved. All of this data comes from Bitbucket. She’d also like to know how many successful builds have been performed. This data is provided by Bamboo.



More on the Git development lifecycle: In this project, changes to source code are stored in a Git repository managed by Bitbucket. When a developer begins work on a story, he or she typically creates a Git “branch” to store committed pieces of work (called “commits”). The developer pulls the branch to his or her local environment to work. After one or more commits on the branch, the developer creates a “pull request”, asking one or more reviewers to do a code review. When all reviewers have approved the work, the branch is pushed back and merged into the main code line of the repository. Builds, releases, and deployments are managed and often automated by Bamboo.

Exercise 1 – Exploring Development Tool Integration

High-level instructions

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on the next page.

1. Log in to Jira Software as Alana Grant (**agrant/Charlie!**).
2. Open the Teams in Space (TIS) project.
3. Briefly examine the backlog, and then open the active sprint board from the side panel.
4. Open the Development Panel for the issue, TIS-74.
5. Follow the steps below and answer the questions as you explore the development data for this issue.



Each of the following steps will result in a popup window launched by Jira. Each of these contains links that will take you into Bitbucket or Bamboo. Unless you already know how to navigate in Bitbucket and Bamboo, we recommend you do not click the links in the popup windows, except where you are instructed to do so.



- a. Click the **1 branch** link. What information is available to you here? What action can you take here?
- b. Click the **1 commit** link. How many files were involved in this commit? Click the **Show files** link to view them within this window.
- c. Click the **1 pull request** link. Who are the 2 designated reviewers for this pull request? Have they approved it?

Answers:

- a) We can see the name of the repository (Website), the branch name, and the fact that there's an open pull request for this branch. You can create a pull request from this page.
 - b) There are 2 files in the commit: full_saturn.jpg and saturn-page.html
 - c) The reviewers are Kevin Campbell and Mitch Davis. The pull request has not been approved because there are no green checkmarks on the reviewer icons.
6. Open the development panel for TIS-26 and answer the questions below as you explore the development data for this issue.



- a. What is the status of the pull request? What does it mean?
- b. The Development panel shows the most recent build for this issue. Can you tell how many builds have been executed for this issue? What was the result of the most recent build?

Answer:

- a) The pull request was approved and merged. This means that the branch under review was merged back into the main code line.
- b) We can assume there were 4 builds, since the build # is 4. The most recent build completed successfully in 12 seconds.

Detailed instructions

1. Log in to Jira Software as Alana Grant (**agrant/Charlie!**).
2. Open the Teams in Space project.
 - a. Go to the **Projects** menu.
 - b. Select **Teams in Space (TIS)**.
3. Examine the backlog and open the active sprint board.
 - a. The left navigation panel provides links to different project views (Backlog, Active Sprint, Releases, Reports, Issues, and Components). Click in the navigation panel to open the Backlog view. The sprint backlog is shown at the top, and the product backlog is beneath it. (The navigation panel can be expanded by clicking the >> at the bottom of the page.)

- b. Briefly scroll through the list of issues in the product backlog to familiarize yourself with them.
- c. Click the Active Sprint link in the side panel to open the sprint board.



You can open the **TIS Scrum Lab 2** board at any time directly from the Boards menu.

4. View the development panel for TIS-74.
 - a. Click the TIS-74 story to open the Issue Detail panel.
 - b. Scroll the panel quickly down to the development data by clicking the {} icon in the narrow panel to the left of the Issue Detail panel.

5. Follow the steps below and answer the questions as you explore the development data for TIS-74. This issue has 1 branch, 1 commit, and 1 pull request that reference it.



Each of the following steps will result in a popup window launched by Jira. The windows contain links that will take you into Bitbucket. Unless you already know how to navigate in Bitbucket, we recommend you do *not* click the links in the popup windows except where you are instructed to do so.



- a. Click the **1 branch** link. What information is available to you here? What action can you take here?
- b. Click the **1 commit** link. How many files were involved in this commit? Click the **Show files** link to view them within this window.
- c. Click the **1 pull request** link. Who are the 2 designated reviewers for this pull request? (Hint: Move your cursor over each reviewer icon.) Does it look like the pull request has been approved by the reviewers?

Answers:

- a) We can see the name of the repository (Website), the branch name, and the fact that there's an open pull request for this branch. An action you can perform on this page is to create a pull request.
 - b) There are 2 files in the commit: full_saturn.jpg and saturn-page.html
 - c) The reviewers are Kevin Campbell and Mitch Davis. The pull request has not been approved because there are no green checkmarks on the reviewer icons.
6. Open the development panel for TIS-26 and answer the questions below as you explore the development data for this issue.



- a. Click the **1 pull request** link. What is the status of the pull request? What does it mean?
Hint: Review the **More on the Git development lifecycle** paragraph on page 1.
- b. The Development panel shows the most recent build for this issue. Click the **1 build** link for more detail. Can you tell how many builds have been executed for this issue? What was the result of the most recent build?

Answer:

- a) The pull request was approved and merged. This means that the branch under review was merged back into the main code line.
- b) We can assume there were 4 builds, since the build # is 4. The most recent build completed successfully in 12 seconds.

Exercise 2 – Exploring Confluence Integration

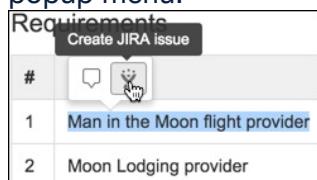
In this exercise, we'll examine the integration between Confluence and Jira Software. You will log in to Confluence as Ryan Lee, the product manager who wants to add some product requirements, and in doing so, generate the corresponding issues in the product backlog automatically.

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on page 7.

High-level instructions

1. Open Confluence in another tab of your browser, logging in as Ryan Lee ([rlee/Charlie!](#)).
2. Select Teams in Space from the Spaces menu. Notice the list of links (“Space Shortcuts”) in the left panel. This provides easy access to product roadmap management, meeting notes, team calendars, and any other information you’d like to see, related to the development effort.
3. Open the **Product Requirements for Teams in Space** page. The product manager wants to add two new requirements to the page. In adding the requirements, you can automatically generate the issues in Jira without leaving Confluence.
 - a. Click **Edit** at the top of the page.
 - b. Add two new requirements to the Requirements table. The Title and User Story are given in the table below. You can add some text to the Importance and the Notes column for each if you wish:

Title	User Story
Man in the Moon flight provider	Customer wants to be able to book a flight from Man in the Moon provider
Moon Lodging provider	Customer wants to be able to book a hotel with Moon Lodging provider

- c. Click **Update** when you've finished, exiting Edit mode.
4. Generate the issues in the Teams in Space project in Jira Software.
 - a. Select the title of the first requirement then select the Jira icon from the popup menu.

 - b. In the Create Issue window, enter the necessary information to create the issue in Jira (**Teams in Space** project, **Story** issue type. Enter the title of the requirement in the Summary field.)

- c. Instead of clicking Create, click the **Create multiple issues from table** link at the bottom of the window. This will generate Jira issues for both requirements at once.
- d. Click **Create** in the next window.
- e. The Jira issue links now appear in the requirements table.

#	Title
1	Man in the Moon Flight provider TIS-84 OPEN
2	Moon Lodging provider TIS-85 OPEN

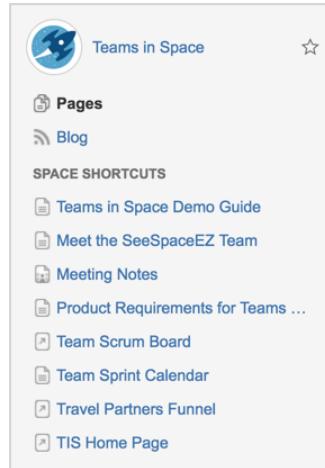
- 5. At the top of the Product Requirements page, find the **2 Jira links** link. Click it to see a list of Jira links on this page. This is just another place to get quick access into Jira to view the issues.

You can see how easy it is for the product owner to get new issues into the product backlog while working on the product requirements in Confluence!

- 6. Return to the browser tab page that has a Jira session open for Alana Grant.

Detailed instructions

1. Open Confluence in another tab of your browser.
 - a. Expand the dropdown menu to the left of Jira in the menu bar.
 
 - JIRA
 - Confluence
 - Bitbucket
 - Bamboo
 - b. Control-click on **Confluence** to open it in another tab.
 - c. Sign in as Ryan Lee, the product manager. (**rlee/Charlie!**)
2. Select **Teams in Space** from the Spaces menu. Notice the list of links (“Space Shortcuts”) in the left panel. This provides easy access to product roadmap management, meeting notes, team calendars, and any other information that you’d like to see, related to the development effort.



3. As product owner, you want to add some requirements to a product requirements page that you started earlier. Click **Product Requirements for Teams in Space** and add two requirements.
 - a. Click **Edit** at the top of the page.
 - b. Scroll down to the table listing the requirements. Click in the Title column of the first row of the table.
 - c. Enter “Man in the Moon flight provider” in the Title column and “Customer wants to be able to book a flight from Man in the Moon provider” as the User Story. Optionally, enter something in the Importance and Notes columns.
 - d. Tab to create a new row and enter “Moon Lodging provider” as Title. Fill in the rest of the row similarly to the following:

Requirements				
#	Title	User Story	Importance	Notes
1	Man-in-the-Moon flight provider	Customer wants to be able to book a flight from Man-in-the-Moon provider	Large customer demand	Must integrate their interface with our other providers.
2	Moon Lodging provider	Customer wants to be able to book a hotel with Moon Lodging provider	Large customer demand	Make sure they can indicate non-smoking

- e. Click **Update** to save the issue and exit Edit mode.
4. Generate the issues in Jira.
 - a. Highlight the title text for one of the requirements. Hover your cursor over the highlighted text until a small popup appears. Click the Jira icon in the popup.

Requirements	
#	Create JIRA issue
1	Man in the Moon flight provider
2	Moon Lodging provider

- b. The Create Jira Issue window opens. If necessary, correct the project and issue type to **Teams in Space** and **Story**. If necessary, enter the Title text of your requirement into the Summary field as shown below. Also enter the User Story into the Description field.

- c. Instead of clicking Create, find the **Create multiple issues from table** link. Jira recognizes that you have a table of requirements and offers to automatically generate all listed requirements as Jira issues. Click the link.

- d. In the Create Multiple Issues window, click **Create**.

- e. The Jira issue links now appear in the requirements table.

#	Title
1	Man in the Moon Flight provider TIS-84 OPEN
2	Moon Lodging provider TIS-85 OPEN

5. Look for a link at the top of the Product Requirements page that indicates there are **2 Jira links** in the page. Click it. You should see both of the new Jira issues you generated. This is just another place to get quick access into Jira to view the issues.

See how easy it is for the product owner to get new issues into the product backlog while working on the product requirements in Confluence?

6. Return to the browser tab page containing the Jira session for Alana Grant.

Congratulations on completing the lab!

Lab 3 – Using Scrum to Manage the Feedback Loop

Scenario

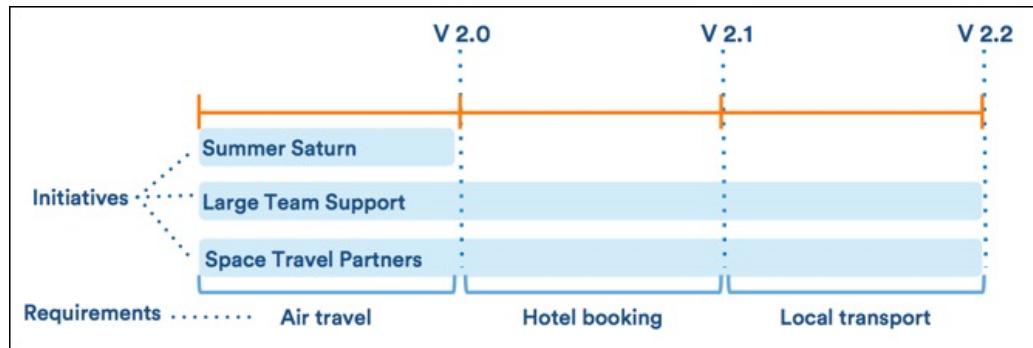
In this lab, you examine the existing Scrum board's backlog and close issues deemed irrelevant or unfeasible. You also review issues in several passes in order to organize the backlog to best support the following product roadmap.

Product Roadmap

For the next 3 releases, we will focus on the following initiatives:

- **Summer Saturn Sale** - The marketing team will work on this initiative.
- **Large Team Support** - This requirement provides support for new functionality that will allow customers to book trips for large groups. This includes the ability to book with external/partner providers.
- **Space Travel Providers** - This requirement provides the web interface to support the large team initiative and integrate with new providers. We've decided not to include Pluto travel at this time, as it is too remote to be feasible.

Initiatives 2 and 3 are the focus of the development team. The marketing team will work on the first initiative. The work will have to be coordinated between the two teams and therefore a single backlog might be beneficial. We foresee releasing this functionality over 3 releases. The air travel portion will be released first, followed by hotel booking, and finally, local transport. And, as always, we will take care of as many bugs as we can with each version. The following chart shows the alignment of initiatives to releases.



In the first exercise, you triage issues that won't be acted on. Then, in exercise 2, you organize and prioritize the backlog. Many of the issues are already assigned to a specific epic (initiative), but you will need to make assignments for other issues. You'll also assign version numbers where needed. When prioritizing issues, you can refer to the roadmap for guidance.

You will be logging in as Ryan Lee, the product manager. Ryan wants to organize the feedback into a tight, clean backlog so the development team can plan their sprints more easily.

Exercise 1 –Triaging the Backlog

High-level instructions

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on this page.

1. Log into Jira Software as Ryan Lee (**rlee/Charlie!**).
2. Open the Teams in Space (TIS) project. Open the Backlog view.
3. Identify issues that should be triaged from the backlog. Look for redundant issues first. There are 3 of these:
 - TIS-82 and TIS-84 are redundant.
 - TIS-83 and TIS-85 are redundant.
 - TIS-81 is very similar to TIS-48, which we will keep.
4. Close TIS-82 and TIS-83 and select **Duplicate** as the resolution.



To close issues that are not in a current sprint, open each issue for editing in Jira and close it from the Close Issue button in the issue detail window.

5. Close TIS-81 in the same way.
6. Now remove issues that, according to the roadmap, will not be acted upon because they are out of scope. Set the Resolution field to **Won't Do**.



The roadmap indicates a travel destination that is not feasible. Any issues related to this destination should be closed. There are two of these.

Detailed instructions

1. Log into Jira Software as Ryan Lee (**rlee/Charlie!**).
2. Open the Teams in Space project.
 - a. Go to the **Projects** menu.
 - b. Select **Teams in Space (TIS)** from the list of projects. (If Teams in Space does not appear in the menu, click **View all projects** and select it from the list of projects.)
 - c. Open the Backlog view.
3. Look through the backlog for redundant issues and close them. You will close 3 issues.
 - a. TIS-82 and TIS-83 are the same as the new issues, TIS-84 and TIS-85.
 - b. TIS-81 is very similar to TIS-48.
4. **Close TIS-82 and TIS-83** as follows:
 - a. Click TIS-82, opening the detail view panel of the board.
 - b. In the Detail View panel, click the issue link to open the issue in Jira.

- c. Click the **Close Issue** button and, in the Close Issue window, choose a resolution of **Duplicate**. Be sure to click **Close Issue** again in the popup window to save the change.

- d. Click the browser's back button to return to the backlog.
e. Follow the same steps to close TIS-83.
5. Close **TIS-81** in the same way because it is a duplicate of TIS-48.
6. Remove issues that, according to the roadmap, will not be acted upon because they are out of scope. Travel to Pluto will not be supported because it is not feasible.



Which 2 issues relate to Pluto travel?

Answer: TIS-28 and TIS-80

- a. Close these two issues. Select **Won't Do** as the resolution.
b. Return to the backlog.

Exercise 2 – Organizing and Prioritizing Issues According to the Roadmap

In this exercise, you organize your backlog using epics and components. You also decide that better use of components would make it easier to view related issues, so you add a new component and make some additional component assignments.

For this exercise, only concern yourself with the Large Team Support epic and the Space Travel Partners epic. Also, don't worry about the issues in the current sprint. You will address that in the next module.

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on Page 15.

High-level instructions

1. Review the issues that have not been assigned to an epic. Assign those that are obviously related to Space Travel Partners. (Hint: 5 issues should be assigned)
2. Add a new component to track issues related to the Air Travel requirement. According to the roadmap, it will be important to be able to track the progress of Air Travel, Hotel booking, and Local transport requirements. We can use components for this purpose. However, there is no component for air travel.
 - a. Log in as the Jira admin (admin/Charlie!) and create the **Air Transportation** component in the Teams in Space (TIS) project.
 - b. Set the Default assignee to Project Lead.
 - c. Click **Add**.
3. Log out and log in again as Ryan Lee (**rlee/Charlie!**)
4. Assign 3 issues from the Large Team Support epic to the UI component. (See Detailed Instructions section for a table showing which components are to be assigned to which issues.)
5. Assign 5 issues from the Space Travel Partners epic to the Air Transportation component. Use the Bulk Change feature to assign all of these at once.
6. Assign the Hotel Booking component to one of the issues from Space Travel Partners.



The remaining steps have no definitive correct answer. Use your best judgment.

7. Assign version 2.0 to any issues that relate to the space travel requirement as described in the roadmap. Focus on the issues in the Space Travel Partners epic and the Large Team Support epic. Also assign all of the bugs **and TIS-74** to version 2.0. (The latter is required for successful completion of lab 7.)



The easiest way to assign version numbers is to, first filter for one particular epic, and then drag issues from the backlog to the version of choice in the Version panel. Multi-selecting the issues for dragging is also possible.

8. Assign version 2.1 to issues that relate to the hotel booking requirement.
9. Assign version 2.2 to issues that relate to the local transport requirement.

10. Determine what the top 20 issues should be in the backlog and then rank them accordingly.
11. Flesh out the top 20 issues by confirming that their component assignments make sense. Add any additional information you think is needed for each issue.

Detailed instructions

1. Assign some of the epic-less issues to an epic.
 - a. Click the **Issues without epics** link in the Epics panel.



Ignoring the bugs for now, 5 of these issues relate to one of our epics of concern. Which epic is this?

Answer: Space Travel Partners

- b. Multi-select the issues using Control-click (or Command-click).
- c. Drag the block of stories to the Space Travel Partners epic as shown below. (TIS-77, TIS-78, TIS-79, TIS-84, TIS-85)

The screenshot shows the Jira interface with the Epics panel on the left and the Backlog on the right.

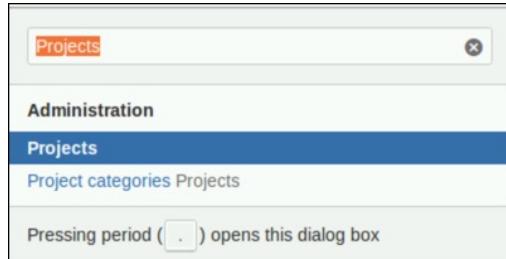
Epics Panel:

- All issues
- Space Travel Partners** (selected, highlighted in blue)
- Summer Saturn Sale
- Large Team Support
- Website Redesign
- Issues without epics

Backlog: 14 of 42 issues visible | Clear all filters

Issue Key	Description
TIS-9	After 100,000 requests the SeeSpaceEZ server dies
TIS-8	Requesting available flights is now taking > 5 seconds
TIS-7	500 Error when requesting a reservation
TIS-10	Bad JSON data coming back from hotel API
TIS-37	When requesting user details the service should return prior
TIS-38	Suggested Destinations
TIS-55	Links should not be in italics
TIS-57	Buttons need to be red
TIS-77	Add Star-Flight provider
TIS-78	Add Men-On-Mars provider
TIS-79	Add Airbnb Saturn provider
TIS-81	TIS-81 Cancel reservation
TIS-84	Man in the Moon Flight provider
TIS-85	Moon Lodging provider

2. Add a new component. According to the roadmap, it will be important to be able to track the progress of Air travel, Hotel bookings, and Local transport requirements. We can use components for this purpose; however, there is no Air travel component so you will act as Jira admin to create the component.
 - a. Log out as Ryan Lee and log in again as **admin/Charlie!**
 - b. Open the Projects administration page quickly by typing **.Projects**. When you type the '.', Jira pops up a window for you to enter the particular administration page name you want.



- c. In the Projects administration page, select the Teams in Space project.
- d. Scroll down the page to find the **Components** section. Click **Show All**.
- e. Enter “Air Transportation” as the Name and anything you want for the Description field. We will not assign a component lead for this component. Select **Project Lead** for the Default Assignee. Click **Add**.

Components			
Name	Description	Component Lead	Default Assignee
Air Transportation	Air travel to space destinations		Project Default (Add)
Hotel Booking	Issues concerning hotels, motels, and other accommodations in space	Project Default	Delete
Deals and Offers	All discounts related to parties less than 8	Project Default	Delete
Local Transportation	Issues concerning all hired local travel	Project Default	Delete

- f. Scroll through the rest of the components and notice the component leads that have been assigned to some components.
3. Log out as admin and log in again as Ryan Lee. Reopen the TIS project backlog.
 4. The table below shows all the issues and the corresponding components that you will assign to them. First, assign the UI component to TIS-43, TIS-39, and TIS-47.

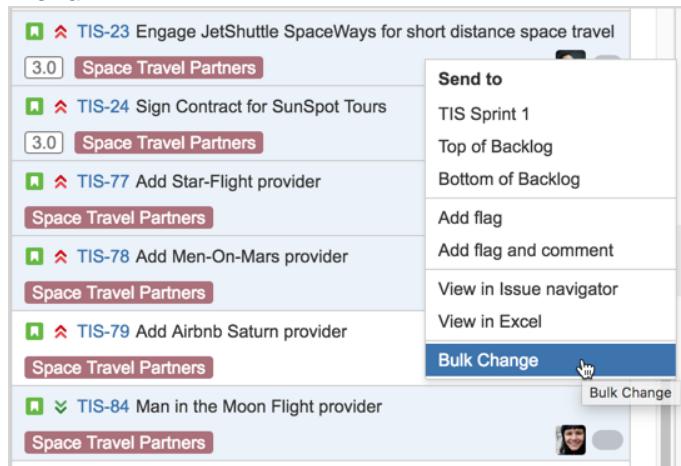
Epic	Issue #	Components
Large Team Support	TIS-43	UI
	TIS-39	UI
	TIS-47	UI
Space Travel Partners	TIS-25	Air Transportation
	TIS-23	Air Transportation
	TIS-24	Air Transportation
	TIS-77	Air Transportation
	TIS-78	Air Transportation
	TIS-84	Air Transportation
	TIS-85	Hotel Booking

- a. Filter the backlog to view only the Large Team Support epic issues.

- b. Select TIS-43 and then, in the Issue Detail panel, click **Edit** in the context menu.
- c. In the Edit Issue popup window, use the Component option list to select **UI**, adding it to the existing Space Travel component.

The screenshot shows the 'Edit Issue' dialog box. At the top right is a 'Configure Fields' button. The 'Summary' field contains the text 'Extend booking experience in UI to include multiple flights on one reservation'. The 'Priority' field is set to 'Critical'. The 'Component/s' field shows 'Space Travel' and 'UI' selected. Below it is a note: 'Start typing to get a list of possible matches or press down to select.' The 'Affects Version/s' field is set to '2.0'. Another note below it says: 'Start typing to get a list of possible matches or press down to select.' The 'Assignee' field is set to 'Kevin Campbell'. At the bottom left is a link 'Assign to me'.

- d. Click **Update**.
 - e. Repeat steps a – d to add the component to TIS-39 and TIS-47.
5. Use the Bulk Change feature to update all of the issues that will be assigned to the Air Transportation component.
- a. Multi-select all of the Air Transportation components using Control-click (or Command-click).
 - b. Right click in the selected area and click **Bulk change** in the context menu.



- c. In the Bulk Operation page, select **Edit Issues** and click **Next**.

Bulk Operation

<ul style="list-style-type: none"> ● Choose Issues Selected 5 issues from 1 project(s) ● Choose Operation <ul style="list-style-type: none"> <input type="radio"/> Operation Details <input type="radio"/> Confirmation 	<p>Step 2 of 4: Choose Operation</p> <p>Choose the operation you wish to perform on the selected 5 issue(s).</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;"><input checked="" type="radio"/> Edit Issues</td> <td>Edit field values of issues</td> </tr> <tr> <td><input type="radio"/> Move Issues</td> <td>Move issues to new projects and issue types</td> </tr> <tr> <td><input type="radio"/> Transition Issues</td> <td>Transition issues through workflow</td> </tr> <tr> <td>N/A <input type="radio"/> Delete Issues</td> <td>NOTE: You do not have permission to delete</td> </tr> <tr> <td><input type="radio"/> Watch Issues</td> <td>Watch all the selected issues. You will receive notifications.</td> </tr> <tr> <td><input type="radio"/> Stop Watching Issues</td> <td>Stop watching all the selected issues. You will no longer receive notifications.</td> </tr> </table> <p style="text-align: right;">Next Cancel</p>	<input checked="" type="radio"/> Edit Issues	Edit field values of issues	<input type="radio"/> Move Issues	Move issues to new projects and issue types	<input type="radio"/> Transition Issues	Transition issues through workflow	N/A <input type="radio"/> Delete Issues	NOTE: You do not have permission to delete	<input type="radio"/> Watch Issues	Watch all the selected issues. You will receive notifications.	<input type="radio"/> Stop Watching Issues	Stop watching all the selected issues. You will no longer receive notifications.
<input checked="" type="radio"/> Edit Issues	Edit field values of issues												
<input type="radio"/> Move Issues	Move issues to new projects and issue types												
<input type="radio"/> Transition Issues	Transition issues through workflow												
N/A <input type="radio"/> Delete Issues	NOTE: You do not have permission to delete												
<input type="radio"/> Watch Issues	Watch all the selected issues. You will receive notifications.												
<input type="radio"/> Stop Watching Issues	Stop watching all the selected issues. You will no longer receive notifications.												

- d. On the next page, select **Air transportation** from the Component dropdown list. Retain the default setting of **Add to existing**.

Step 3 of 4: Operation Details

Choose the bulk action(s) you wish to perform on the selected 5 issue(s).

<input type="checkbox"/> Change Issue Type	<input checked="" type="checkbox"/> Bug	<input type="checkbox"/> Minor	
<input type="checkbox"/> Change Priority	<input checked="" type="checkbox"/> Minor	<input type="checkbox"/> Major	
<input type="checkbox"/> Change Fix Version/s	Add to existing		
Start typing to get a list of possible matches or press down to select.			
<input type="checkbox"/> Change Affects Version/s	Add to existing		
Start typing to get a list of possible matches or press down to select.			
<input checked="" type="checkbox"/> Change Component/s	Add to existing		
Start typing to get a list of possible matches or press down to select.			
<input type="checkbox"/> Change Assignee	<input checked="" type="checkbox"/> Automatic		

- e. Scroll down to the bottom of the page and click **Next**.
f. Finally, confirm this action.
g. When the bulk operation progress has completed, click **Acknowledge**.
6. Assign the Hotel Booking component to TIS-85.



The remaining steps have no definitive correct answer. Think about the roadmap and decide how you will proceed. You are the product owner. You are empowered to make these decisions.

7. Assign issues to Version 2.0. Look for any issues that relate to air transportation from within the Space Travel Partners and Large Team Support epics. Since the product owner wants bugs to be fixed in version 2.0, you should also assign them to version 2.0. The product owner also considers **TIS-74** a high priority issue, so assign it to this version as well.
- a. Filter the backlog to show just the issues of the Space Travel Partners epic.

- b. Identify and multi-select issues that relate to space travel (air transportation, for instance). (Hint: It might be helpful to make a list.)
- c. Open the Versions panel.
- d. Click and drag the selected issues to the Version 2.0 in the Versions panel.
- e. Repeat this process for Large Team Support issues.
- f. Repeat this process to assign all bugs to version 2.0 and also the **TIS-74** issue.



Lab 7 depends upon TIS-74 being included in version 2.0)

8. Assign issues to Version 2.1 in the same way. Look for issues that relate to the hotel booking requirement.
9. Assign issues to Version 2.2 that relate to the local transport requirement.
10. Determine what the top 20 issues should be in the backlog and then rank them accordingly. Typically, the product owner would get input from the development team when ranking issues. Here are some things to consider:
 - a. Is this issue dependent upon another issue?
 - b. Which issues are of higher customer priority?
 - c. Which issues require feedback more urgently?
 - d. What is the relative implementation difficulty?



Rank issues by dragging them up or down, one-by-one. An easy, quick way to do this is Control-click (or Command-click) on an issue and use the menu to send it either to the top or the bottom of the backlog.

11. Flesh out the top 20 issues.
 - a. Confirm that the component assignment for each issue makes sense, given the version assignment you've made. Make corrections and/or assignments as needed.
 - b. Add any additional detail to issues that seem vague to you.

Congratulations! You have just built your product backlog and it's ready for the engineering team!

Lab 4 – Setting up the Scrum Board

Scenario

The development team has some special needs for its Scrum board, so the Scrum master will create a new board based on an existing filter. The workflow also needs some help. It will need an additional status for code reviews, so a different workflow schema, created by the Jira admin, will be applied to the project.

The project roles have been defined by the Jira admin, but they have the default groups assigned to each role. In Exercise 2, you fine-tune the role assignments to allow just the team members for this specific project to have the Developer role. The product owner would like to keep track of the progress of some of the Summer Saturn Sale issues that relate to the work being done in the sprint. You will setup a project link between the Jira Teams in Space project and the Teams in Space Confluence space. You create a Confluence page to display linked issues to assist the product owner to stay abreast of related issues.

You'll be logging in as Alana Grant, the scrum master, for most of this. However, you will need to log in as Jira admin to switch the workflow scheme for the project.

Exercise 1 – Creating and Configuring the Scrum Board

High-level instructions

This particular project team will be working strictly with stories from the Large Team Support epic and the Space Travel Partners epic. You've decided that it would be easier and less confusing for everyone to narrow the scope of the backlog issues. Therefore, you create and save a filter that gives you and the team exactly the issues you're interested in. You create a Scrum board based on this filter. You'll need to also enable ranking and share the filter.

You also realize you'll need a status (and corresponding column) to accommodate code reviews. After explaining this to the Jira admin, she created a new workflow schema containing the workflow you've requested. You log in as Jira admin and switch the schema for the project.

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on the next page.

1. Log in to Jira Software as Alana Grant (**agrant/Charlie!**).
2. Create a filter that returns issues from the TIS project that are assigned to either the Large Team Support epic or the Space Travel Partners epic. You also want any bugs included in the filter, regardless of epic. Save the filter to make it available for use as the board filter. Name the filter as “TIS filter for Lab 4”.



The filter will have to return all those issues in the Teams in Space project that are in Large Team Support epic (TIS-3) or Space Travel Partners epic (TIS-6), and also all of the bugs. In order to have the epics appear in the Epics panel (for easy filtering of the backlog), we'll also need the actual epics.

3. Create a new Scrum board based on the filter you just created. Be sure to select **Board based on an existing filter**. Name it "TIS Dev Team Scrum Board".
4. Configure the board. You want all of the Teams in Space project members to have access to the board, so you must share the filter. Also, the board should allow ranking of issues. Both of these changes can be made on the General tab of the Board Configuration page.
5. View the board. You should see the issues sorted by priority - the priorities you set in the previous lab.
6. Log in as the Jira administrator (**admin/Charlie!**) and configure the project in order to switch the workflow scheme to the (currently inactive) **Teams in Space Workflow with In Review**. This is a 3-step process:
 - a. Switch the workflow and click **Associate**.
 - b. You're prompted to associate the statuses from the previous workflow to the new one, however, this will be done automatically in our case. Click **Associate** again.
 - c. The next page tells you statuses have been automatically migrated to the new workflow. Acknowledge this and then return to the project.
7. Log in now as Alana Grant and configure the new Scrum board to add a new column for the In Review status. When you've completed this, confirm the board shows the new column.

Detailed instructions

1. Log into Jira Software as Alana Grant (**agrant/Charlie!**). Alana is the Scrum master and also lead developer.
2. Create a JQL issue filter that returns issues from the TIS project that are assigned to either the Large Team Support epic or the Space Travel Partners epic. The filter should also include all issues of type Bug.



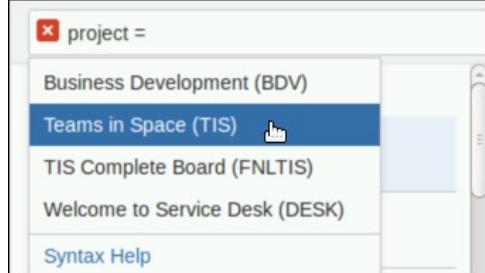
The filter will have to return all those issues in the Teams in Space project that are in Large Team Support epic (TIS-3) or Space Travel Partners epic (TIS-6), and also all of the bugs. In order to have the epics appear in the Epics panel (for easy filtering of the backlog), we'll also need the actual epics.

- a. Go to **Issues > Search for Issues**.
- b. Enter the following JQL query into the Query field:

```
Project="Teams in Space" AND ("Epic Link" in(TIS-3,TIS-6) OR issuekey in(TIS-3,TIS-6) OR issuetype = Bug)
```



Use the syntax help to avoid typos. It will pop up to offer suggestions when you type a few characters.



- c. Execute the query. It should return issues from the 2 desired epics, the epic issues (TIS-3 and TIS-6), in addition to all the bugs.

```
project = "Teams in Space" AND ("Epic Link" in (TIS-3, TIS-6) OR issuekey in (TIS-3, TIS-6) OR issuetype = Bug)
```

- d. Click **Save as** and name the filter “TIS filter for Lab 4”. Click **Submit** to complete the save.
3. Create a new Scrum board based on the filter you just created.
 - a. Return to the Teams in Space project page.
 - b. Click **Board > Create board**.



- c. In the Create Agile board wizard, click **Create a Scrum board**.
- d. In the next page, select **Board from an existing saved filter** and then click **Next**.

Create an Agile board

Board created with new Software project
A new board based on a new Software project

Board from an existing project
Boards can contain one or more projects.

Board from an existing Saved Filter
An advanced option using a JQL filter.

- e. Name the board “TIS Dev Team Scrum Board” and select your new filter from the options available.

Name this board

Board name * TIS Dev Team Scrum Board

Saved filter * TIS filter for Lab 4

Shares No shares

Owner Alana Grant

Back Create board Cancel

- f. Click **Create board**.
4. Configure the board so all team members will have access to it and so it will sort the issues according to the ranking you did in the previous lab.
- Go to **Board > Configure**. You will be in the General configuration page by default.
 - Share the filter with all users. In the **Filter** section, click **Edit Filter Shares**.

Filter

Saved Filter TIS filter for Lab 4

[Edit Filter Query](#)

Shares No shares

[Edit Filter Shares](#)

- c. In the **Add Shares** section of the Edit Current Filter window, select **Project, Teams in Space**, and **All**. Click the **Add** button.

Edit Current Filter

Name * TIS Scrum Board Filter for Lab 4

Description

Favorite ★

Shares Not shared

Add Shares Project > Teams in Space > All + Add

Shared with everyone with permission to browse the 'Teams in Space' project

Save Cancel

- d. The share you've specified now appears in the Shares field. Click **Save**.
- e. Open the TIS Dev Team Scrum Board from the Boards menu. Open the Board configuration page.
- f. In the Filter section, notice that ranking is disabled because the filter does not contain an Order By clause. Click **Add Rank**. This adds the necessary Order By Statement.

Filter Query: project = "Teams in Space" AND ("Epic Link" in (TIS-3, TIS-6) OR issuekey in (TIS-3, TIS-6) OR issuetype = Bug)
 ORDER BY Rank ASC

Ranking: Using Rank

Projects in board: Teams in Space
[View permission](#)

5. Click **Back to board**. You should now see the issues sorted according to the ranking you did in the last lab.
6. Your board needs a column and status for issues that are in code review. Since the board does not use the Simplified Workflow, you can't just add a new column to get the new status. Instead, you have requested the Jira administrator to create this workflow for you. It's available within a new workflow scheme called **Teams in Space Workflow with In Review**. Configure the project to switch to the new workflow scheme.
 - a. Log out and then log in again as the Jira administrator (**admin/Charlie!**).
 - b. Click the menu icon in the upper left corner of the browser page and select **Configure...**



- c. Select **Issues** and then click **Workflows** in the left panel.

Administration

Applications Projects Issues Add-ons User management System

- d. Expand the **Inactive** workflows section to find the workflow that was created for you called **Teams in Space with In Review**. Notice it has been assigned to the Teams in Space Workflow with In Review scheme.

Name	Last modified	Assigned Schemes	Steps	Operations
jira (Read-only System Workflow) <small>DEFAULT</small> The default JIRA workflow.			5	View · Copy
Teams in Space with In Review The classic JIRA default workflow	30/Aug/16 System Admin	• Teams in Space Workflow with In Review	6	Edit · Copy

- e. Now click **Projects** in the same menu and then select **Teams in Space**. This will take you to the main project administration page.
- f. In the **Workflows** section, click the link for the **Teams in Space** scheme.

The screenshot shows the 'Workflows' section of the Jira interface. It features a title 'Workflows' with a gear icon, a description about workflows mirroring team practices, and a 'Scheme' dropdown menu. The 'Teams in Space' scheme is highlighted with a blue border and a cursor icon pointing at it. Below the dropdown is a link to 'classic default workflow'.

- g. Click **Switch Scheme**. This is a 3-step process.
- h. In step 1, select the scheme, **Teams in Space Workflow with In Review** and then click **Associate**.

The screenshot shows the 'Associate Workflow Scheme to Project' dialog. It has a title 'Associate Workflow Scheme to Project' and a sub-section 'Step 1 of 3: Select the scheme you wish to associate'. A note below says 'Note: It is recommended to [backup JIRA data](#) before proceeding with the workflow association.' A dropdown menu labeled 'Scheme' contains 'Teams in Space Workflow with In Review'. At the bottom are 'Associate' and 'Cancel' buttons.

- i. Step 2 reminds you the statuses from the previous workflow need to be migrated to the new one. In our case, this will happen automatically. Click **Associate** again.
 - j. Step 3 lets you know the statuses have all been migrated. Click **Acknowledge**.
7. The new workflow is in effect; however, you need to add a new column to your board to accommodate the In Review status. The Project administrator can do this, so log in as Alana Grant.
- a. Open the **TIS Dev Team Scrum Board** from the Boards menu and then click **Board > Configure**.
 - b. In the Column Management page, click **Add Column**.
 - c. Make sure the new column is between In Progress and Done.
 - d. Name the column "In Review".
 - e. Drag the In Review status into the new column.

The screenshot shows the 'Column Management' page for a scrum board. It lists columns: 'To Do', 'In Progress', 'In Review', and 'Done'. Each column has a 'Drag to rearrange, or delete' handle. Under each column, there are boxes for 'OPEN' (26 issues), 'IN PROGRESS' (4 issues), 'IN REVIEW' (No issues), and 'RESOLVED' (No issues). A 'CLOSED' column is also present with 3 issues. An 'Add column' button is visible at the top right.

- f. Click **Back to board**.
- g. Confirm that the new column appears on the board by viewing the active sprint. Click the Active Sprint icon in the left panel.

Exercise 2 – Configuring Project Roles and Project Links

High-level instructions

The Jira admin has created the project roles and has assigned the roles to default groups. Remember groups are global, whereas roles are project-specific. The Developer role currently is assigned to the Jira-developers group. This group contains 11 users. Your development team for this project has only 5 members. You have decided it would best to narrow the number of users who have access to the issues in your project.

You also setup project links between the TIS project in Jira and the TIS space in Confluence and then modify the Product Requirements page to include links to issues from the Summer Saturn Sale epic. The product owner wants to see these as there is some dependency between these initiatives. The application links between Jira, Bitbucket, Confluence, and Bamboo have already been setup by the Jira administrator.

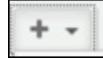


The Jira Software installation, as well as the Teams in Space project you are using for these lab exercises, is one that was upgraded from a 6.x version, therefore, some of the default settings in the project (default groups and default permissions) are different than you would see with a fresh install of 7.x. For instance, in 7.x, Jira Software has one default group (in addition to the Jira-administrators group), **Jira-software-users**. In our lab image, there are 3 default groups: **Jira-administrators**, **Jira-developers**, and **Jira-users**. These are all implemented in Teams in Space. In 7.x, the new Manage Sprints permission is assigned to any logged user (Jira-software-users), however, in 6.x, it is assigned to Jira-administrator. Consequently, you may see differences between this lab image and your own Jira Software installation.

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on the next page.

1. Logged in as the project administrator, configure the Teams in Space project.
2. In the Users and Roles page, add the following users to the Developer role:
 - Alana Grant
 - Kevin Campbell
 - Jennifer Evans
 - Max Taylor
 - Cassie Owens
3. Delete the Jira-developer group from the Developer role.
4. Test this change by logging in as Ryan Lee, the product owner and trying to edit an issue in the backlog. Can you do it? Now log in as Alana and try to edit the issue.
5. As the project administrator, create a project link between the Teams in Space Jira project and the Confluence space, Teams in Space. (Application links have already been configured by the Jira admin.)
 - a. Configure the project.

- b. In the Settings section of the General tab, click Configure Project Links. A project link already exists between Jira Software and a Bitbucket Server project.
 - c. Click **Add Link** and then select **Confluence**.
 - d. Click **Create** to link the Teams in Space Jira project to the Teams in Space Confluence space. The link now also exists in Confluence.
6. Use the link by inserting a list of Jira issues from the Summer Saturn Sale epic (TIS-5) into a Confluence page.
- a. Open the Confluence space called Teams in Space.
 - b. Open the page, **Product Requirements for Teams in Space Website Redesign**.
 - c. Click **Edit**.
 - d. Scroll down the page to the section entitled “Related Marketing Effort”. Put your cursor just beneath the paragraph.
 - e. From the + menu in the editing toolbar, select **Jira Issue/Filter**.



- f. In the popup window, enter the following Jira JQL filter query:
Project = "Teams in Space" and "Epic link" = TIS-5
- g. Execute the query to make sure you've typed it correctly. It should bring up a list of issues.
- h. Expand the **Display options** panel. Select a table of issues and limit the display columns to: Key, Summary, Updated, Assignee, Reporter, and Status.

Display options

Display as Single issue
Display the macro as a single issue.

Total issue count
Display total number of issues as a link. E.g. 10 issues

Table
Customise your columns below.

Maximum issues

Leave empty to get all issues.

Columns to display Key × Summary × Updated × Assignee × Reporter × Status × |

- i. Click **Insert**. The table will appear in iconized format.
- j. Click **Save** to see the effect in the entire page.

Key	Summary	Updated	Assignee	Reporter	Status
TIS-36	Line up panel of former travelers to Saturn for interview content	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-35	Draft out analytics plan for Saturn Summer Sizzle website	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-34	Create Launch Plan for the Saturn Summer Sizzle	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-33	Select key travel partners for the Saturn Summer Sizzle	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-32	Create Video Assets for Saturn Summer Sizzle website	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-31	Create Teams in Space website copy for the Saturn Summer Sizzle	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-30	Create Saturn Summer Sizzle Logo	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-29	Create Banner Ads to use for partner marketing	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-22	Create Blog Post for Saturn Summer Sale	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-21	Create Email Campaign for Saturn Summer Sale	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
10 issues 					

- k. Notice all the links into Jira. Click the 10 issues link at the bottom of the list. This takes you into the Jira Search Issues page to view the results of this query.
- 7. Return to Jira and add a link to this page in the Teams in Space backlog. At the top of the backlog, click the **Linked Pages** link. Start typing **Product Requirements** and your page will be found. Select it. This will be helpful for the product owner to have instant access to important information.

Detailed instructions

1. Configure the Teams in Space project.
 - a. If necessary, log in as Alana Grant (**agrant/Charlie!**), the project administrator.
 - b. Click **Configure** on the main Jira menu on the left side of the Jira title bar.
 - c. Click **Projects** from the Administration page, then select Teams in Space from the list of projects.
2. Add the following users to the Developer role:
 - Alana Grant
 - Kevin Campbell
 - Jennifer Evans
 - Max Taylor
 - Cassie Owens
 - a. From the Project administration page, scroll to the **Roles** section and click **View Project Roles**.
 - b. In the Users and Roles page, notice the Developer role has been granted to the Jira-developers group. Click **Add users to a role**.

The screenshot shows the 'Users and roles' page in Jira. At the top right, there is a red box around the 'Add users to a role' button. Below it, the 'Edit defaults' link is visible. The page is divided into two main sections: 'ADMINISTRATORS' (Showing 2 of 2) and 'DEVELOPERS' (Showing 1 of 1). Each section has a table with columns: Name, License, Username, Email address, and Last session.

Name	License	Username	Email address	Last session
jira-administrators				
Alana Grant	Service Desk, Software...	agrant	agrant@nowhere.com	Today 4:21 PM

Name	License	Username	Email address	Last session
jira-developers				

- Begin typing Alana Grant in the **Users or groups** field until her name appears in the context help. Select it, and then change the Role field to **Developers**. Do not click Add yet.
- Continue to add the other users from the list above to the Developers role in the same way.
- When all the users appear in the Users or Groups field, click **Add**.

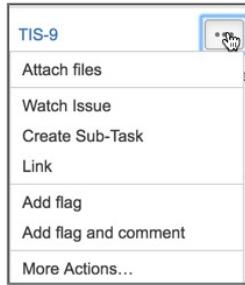
This is a modal dialog titled 'Add users to a role'. It contains a search bar labeled 'Search by name or email' and a list of users: Cassie Owens, Max Taylor, Jennifer Evans, Kevin Campbell, and Alana Grant. Below the list is a 'Role' dropdown menu set to 'Developers'. At the bottom is a blue 'Add' button.

- Delete the Jira-developers group from the Developers role by clicking the trashcan icon in the Jira-developers row.

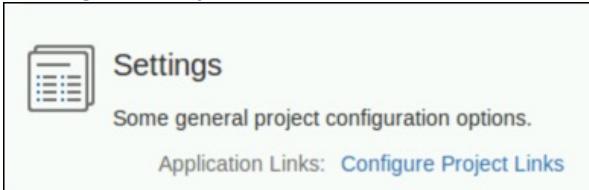
The screenshot shows the 'DEVELOPERS' user list. A red box highlights the trashcan icon next to the 'jira-developers' row. The table has columns: Name, License, Username, Email address, and Last session.

Name	License	Username	Email address	Last session
jira-developers				Remove group from role
Alana Grant	Service Desk, Software...	agrant	agrant@nowhere.com	Today 4:21 PM
Cassie Owens	Service Desk, Software...	cowens	cowens@nowhere.com	04/Aug/13 8:51 PM
Jennifer Evans	Service Desk, Software...	jevans	jevans@nowhere.com	04/Aug/13 7:36 PM
Kevin Campbell	Service Desk, Software...	kcampbell	kcampbell@nowhere.c...	21/Jun/16 11:56 AM

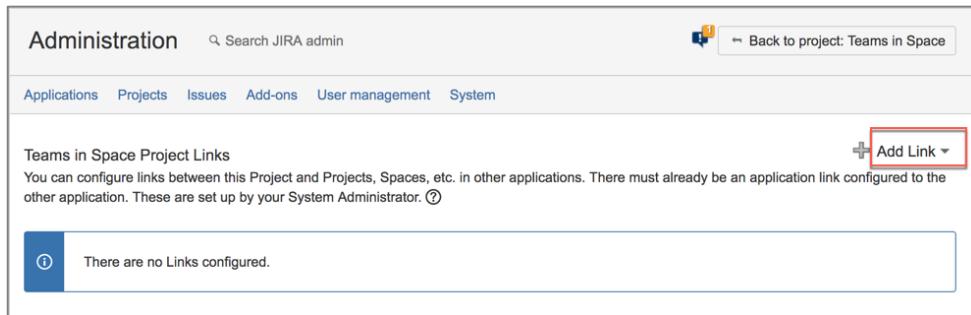
- Test this change by logging in as Ryan Lee (**rlee/Charlie!**), the product owner. View the backlog for the TIS Dev Team Scrum Board and select any issue.
 - In the detail view, click the menu and see if there's an Edit option. Ryan should not be able to edit issues.



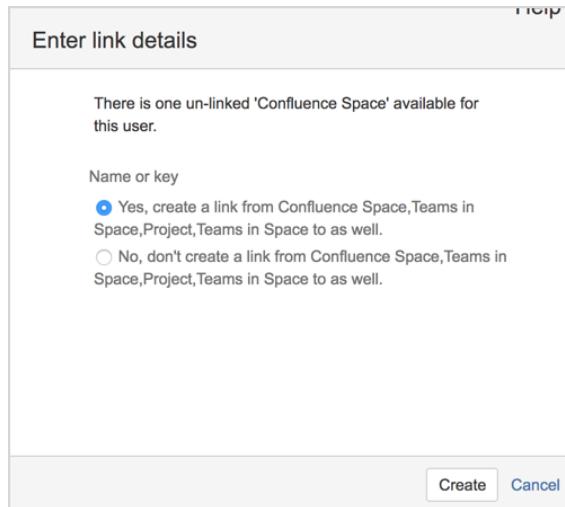
- b. Log in again as Alana. She should have the ability to edit issues. Make sure the Edit menu item appears in the issue detail panel when you select an issue in the backlog.
5. Create a project link between the Teams in Space Jira project and the Teams in Space Confluence space. The application links between Jira and Confluence have already been configured.
 - a. Click the menu to the left of the Jira logo in the title bar and then select **Configure...**. If necessary, confirm the administrative password for Alana Grant.
 - b. Click **Project** in the menu bar and then click the Teams in Space project.
 - c. On the General tab, scroll down to the Settings section and click **Configure Project Links**.



- d. Click **Add link**.



- e. Select Confluence from the dropdown list of options.
- f. In the Enter Link Details window you can accept the default selection.



- g. Click **Create** to link the Teams in Space Jira project to the Teams in Space Confluence space. This link now also exists in Confluence.

Application	Type	Name	Key	Action
Confluence (Confluence)	Confluence Space	Teams in Space	TIS	Delete Edit

6. Now you will use the link by inserting a list of Jira issues from the Summer Saturn Sale epic (TIS-5) into a Confluence page.
- Open Confluence from the menu to the left of the Jira icon in the title bar.
Log in as Alana Grant.
 - Click **Space > Teams in Space**. (Or access the Space Directory to select the space.)
 - In the left panel, click **Product Requirements for Teams in Space Website Redesign** to open the page.
 - Click **Edit**.
 - Scroll down the page to the section entitled “Related Marketing Effort”. Put your cursor just beneath the paragraph. You are going to insert a list of Jira issue links in a table at this location.
 - From the + menu in the editing toolbar, select **Jira Issue/Filter**.



- g. In the Insert Jira Issue/Filter window, enter the following JQL filter query and then click the magnifying glass to check the filter results.

Project = "Teams in Space" and "Epic link" = TIS-5

The screenshot shows a modal dialog titled 'Insert JIRA Issue/Filter'. On the left is a sidebar with 'Search' (containing 'Create New Issue' and 'Recently Viewed'), 'OTHER JIRA CONTENT' (containing 'JIRA Charts'), and a 'Display options' section. The main area has a search bar containing the JQL query 'Project = "Teams in Space" and "Epic link" = TIS-5'. Below the search bar is a 'Summary' section with a checkbox for 'Key'. The main list displays ten Jira issues, each with a checkbox, a key (e.g., TIS-36, TIS-35, TIS-34, etc.), and a brief description. A mouse cursor is hovering over the list. At the bottom of the list is a 'Display options' button.

Key	Summary
TIS-36	Line up panel of former travelers to Saturn for interview cont...
TIS-35	Draft out analytics plan for Saturn Summer Sizzle website
TIS-34	Create Launch Plan for the Saturn Summer Sizzle
TIS-33	Select key travel partners for the Saturn Summer Sizzle
TIS-32	Create Video Assets for Saturn Summer Sizzle website
TIS-31	Create Teams in Space website copy for the Saturn Summe...
TIS-30	Create Saturn Summer Sizzle Logo
TIS-29	Create Banner Ads to use for partner marketing
TIS-22	Create Blog Post for Saturn Summer Sale

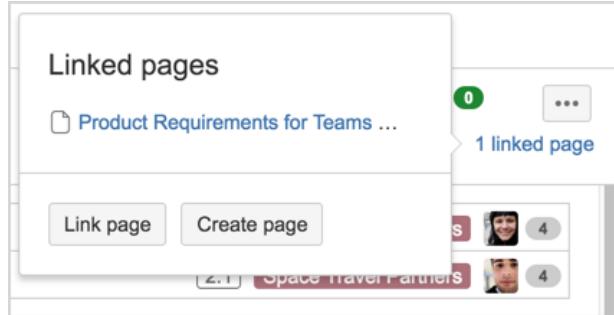
- h. Click **Display options** to configure how you want these issues displayed in the page.
 i. Select **Table** and then reduce the number of columns to: Key, Summary, Updated, Assignee, Reporter, and Status as shown below:

- j. Click **Insert**.
- k. Back in the Confluence page editor, click **Update**.
- l. The table of issues is now populated. Click the **10 issues** link to view them in the Jira issue filter page.

Key	Summary	Updated	Assignee	Reporter	Status
TIS-36	Line up panel of former travelers to Saturn for interview content	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-35	Draft out analytics plan for Saturn Summer Sizzle website	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-34	Create Launch Plan for the Saturn Summer Sizzle	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-33	Select key travel partners for the Saturn Summer Sizzle	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
TIS-32	Create Video Assets for Saturn Summer Sizzle website	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-31	Create Teams in Space website copy for the Saturn Summer Sizzle	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-30	Create Saturn Summer Sizzle Logo	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-29	Create Banner Ads to use for partner marketing	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-22	Create Blog Post for Saturn Summer Sale	Sep 07, 2016	Cassie Owens	Ryan Lee	OPEN
TIS-21	Create Email Campaign for Saturn Summer Sale	Sep 07, 2016	Ryan Lee	Ryan Lee	OPEN
10 issues					

7. In the TIS Dev Team Scrum Board backlog, add a link to the Product Requirements page.
 - a. At the top of the sprint backlog, click **Linked Pages**.
 - b. Click **Link Page** in the popup window.
 - c. Start typing **Product Requirements** in the search field. Your page will be found before you finish typing the name.

- d. Select it and notice the new link. This will be helpful for the product owner have instant access to important information.



Congratulations on completing the lab! Your team can now finish sprint planning and get to work!

Lab 5 - Using Estimation and Tracking

Scenario

The development team has decided tracking actual hours logged against stories would make monitoring progress more efficient. They still prefer story points for estimating, however.

The entire team plans the sprint by consulting together about how many story points to assign to each of the highest priority issues. They use the story point estimates to make a decision about what goes into the new sprint.

Once the sprint is populated, the development team enters original time estimates for each issue and then the sprint begins. As the sprint progresses, developers log time on their issues and monitor their progress using the Time Tracking graph.

In Exercise 1, you configure estimation and set original estimates in both the backlog and the sprint. In Exercise 2, you begin logging work and examine the tracking graph for different logging options.

You will log in as Alana Grant, the scrum master and lead developer. This assures you have the necessary privileges to perform all of the steps described above.



Leave one of your sprint issues unlogged as you work through this lab. This will ensure that you have a clean issue (no time logged against it) to experiment with in the next lab.

Exercise 1 – Configuring Estimation and Setting Original Estimates

High-level instructions

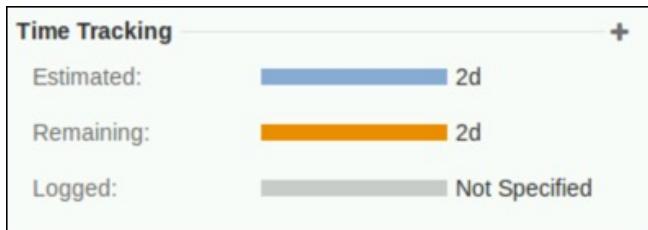
If you would prefer to follow step-by-step instructions, go to Detailed Instructions on the next page.

1. Log in to Jira Software as Alana Grant (**agrant/Charlie!**) and open the TIS Dev Team Scrum Board.
2. Complete the current sprint and then return to the Backlog view. If any of the incomplete issues are assigned to a version later than 2.0, move them lower in the list. We won't want them included in the new sprint.
3. Configure the board to set estimation to story points and enable time tracking by choosing **Remaining Estimate and Time Spent**. Return to the board when finished.
4. Make sure the first 8 issues in the backlog have story point estimates set. Enter an estimate where needed.



Notice that, after enable time tracking, there's a new field in the Issue Detail view: the **Remaining** field. This is where you'll enter your original time estimate later. Be sure that you enter the story points in the **Estimate** field.

- Create a new sprint. (You can just accept the default sprint name.)
- Velocity for this team is 16 story points per sprint. Select enough stories to equal roughly 16 story points and move them into the sprint.
- Select one of the issues in the sprint and set an original time estimate value in the **Remaining** field. This is your original estimate of time to complete the task. Set in time increments (w = weeks, d = days, h = hours).
- Open the issue in Jira and find the **Time Tracking** graph. Notice the value you entered into the Remaining field now appears in the Estimated bar. No work has yet been logged for this issue, therefore, the Estimated and Remaining values are identical.



- Enter original time estimates for the remaining issues in the sprint.
- Start the sprint.

Detailed instructions

- Log in to Jira Software as Alana Grant (**agrant/Charliel!**) and open the TIS Dev Team Scrum Board.
- From the Active Sprint view, click **Complete Sprint** to end the current sprint.
 - In the confirmation window, notice incomplete issues will be moved to the backlog, so you can include them in the next sprint if you wanted.

Complete Sprint: TIS Sprint 1

0 issues were done
5 issues were incomplete

Incomplete issues will be moved to the backlog

Sub-tasks are not included in the total(s) above, and are always included in the same sprint as their parent issue.

Complete [Cancel](#)
 - The sprint report is now displayed. Switch to the backlog view by clicking Backlog in the left navigation panel.
 - Issues that were not completed in the previous sprint will appear at the top of the backlog. If any of these are assigned to a later version than 2.0, move them lower in the list. We don't want them included in the new sprint.
- Configure the board to set estimation statistics and to enable time tracking.
 - Click **Board > Configure**. Select **Estimation** in the left navigation panel.

- b. Estimation is set to Story Points by default. This is what we want. Set Time Tracking to **Remaining Estimate and Time Spent**.

Estimation
Issues can be estimated when in the Backlog to get an idea of how much work is being committed to in a sprint. [Read tracking.](#)

Estimation Statistic: **Story Points**

Estimate issues in the Backlog by entering values for **Story Points**. Your velocity from sprint to sprint will be measured against these estimates.

Time Tracking: **None**: Issues will burn down their **Story Points** value upon completion.
 Remaining Estimate and Time Spent: Track time against issues using JIRA's **Remaining Estimate and Time Spent** fields.

c. Click **Back to board**.

4. Enter story point estimates for any of the first 8 issues in the backlog that aren't yet estimated. Notice the Issue Detail view now displays a new field, the **Remaining** field. This is where you will enter your original *time* estimates later. Make sure you enter your story point estimates in the **Estimate** field.

Teams in Space / TIS-82

Man-in-the-Moon flight provider

Estimate: **Unestimated** ←

Remaining: **Unestimated** X

5. Create a new sprint. Click **Create sprint** at the top of the backlog. Accept the default sprint name.
6. Since your sprint velocity is 16 story points, select enough of the first 8 issues to roughly equal 16 story points and move them into the sprint.
7. Set an original time estimate for the first issue in the sprint.
- Select the issue so it appears in the Issue Detail view.
 - The **Estimate** field in this view displays the story point estimate you set in the last exercise.
 - The **Remaining** field accepts *time* units. Enter your original estimate of time to complete the task. Use increments of weeks (w), days (d), or hours (h). Press Enter to accept the value.
8. Click the issue link at the top of the Issue Detail view to open it in the Jira issue window. Find the Time Tracking graph. The value you entered into the Remaining field appears in the Estimated field. (Remember, this is a time estimate. It is not the same as the Estimate shown in the backlog view.) No work has been logged for this issue yet so the Estimated and Remaining fields show the same value.

Time Tracking

Estimated: 2d

Remaining: 2d

Logged: Not Specified

9. Repeat step 7 for each of the remaining stories in the sprint. For each, enter an original time estimate in the Remaining field.
10. Click **Start sprint**. The ensuing popup window allows you to rename the sprint, change the start date or the duration. Just accept the defaults for now and click **Start**.

Exercise 2 – Logging Time

High-level instructions

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on the next page.

1. Log some time against one of the stories.
 - a. Log work that is less than the Remaining estimate value. For instance, if you've estimated 3 days, log some amount of time less than that. Use the **Adjust Automatically** setting.

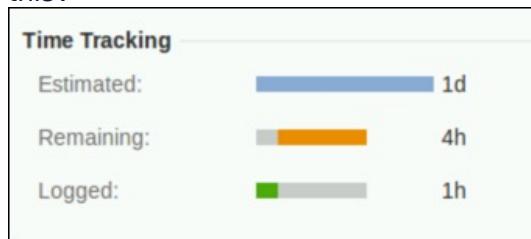


Notice the Remaining estimate in the Issue Detail View is now reduced by the amount of work you logged.

- b. Open the story in Jira to view the graphical display. It probably looks similar to this:



2. After working on another issue, you realize the original time estimate is not accurate. It's going to take you less time to finish the issue than you had expected. Log your work and, before clicking the Log button or pressing Enter, reduce the remaining estimate by an amount of time that leaves some value in the remaining estimate. How does the graph look now? Does it look similar to this?



i

The original time estimate is unchanged, but the Remaining bar has been reduced by 4 hours. Only 1 hour has been logged.

3. You now log some time on either a different issue, or one you've already logged time on. You discover this one will take longer than you had originally estimated. Log some time in the issue and also increase the remaining estimate by 1 day. *Hint:* Use the **Set** option.
4. What does the Time Tracking graph look like now? It might look similar to this example. Here, the original time estimate was 5 days, the Remaining estimate had been already reduced to 4 days after a previously logging. Now, the developer logs an additional day but sets the Remaining estimate to 5 days.

Answer the following questions about what you see.



?

What does the orange portion of the Remaining bar represent?

Answer: The amount of the remaining estimate - 1 week. A total of 2 days has been logged against the issue. Normally, the remaining estimate would have been reduced to 3 days, but it was reset to 5 days - a net gain of 2 days to the remaining estimate.

?

What does the gray portion of each of the bars seem to represent?

Answer: It shows the remaining estimate has been manually adjusted with the addition of 2 more days.

Detailed instructions

1. Log some time for one of the stories.
 - a. Select one of the stories. If the story is still in the To Do column, move it to **In Progress**. In the Issue Detail view, click **Log Work**.



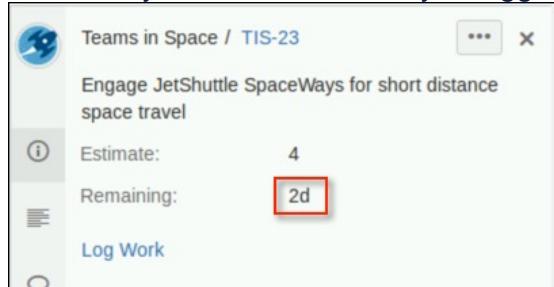
- b. In the Log Work window, enter an amount of time that is less than the Remaining estimate value. For example, if the Remaining estimate is 3d, log 1d. You can also enter time in hours (h) or weeks (w).
- c. Enter the date when this unit of work was begun or accept today's date if that seems appropriate. Accept the **Adjust automatically** setting for the Remaining Estimate.

Time Spent * eg. 3w 4d 12h) ?
An estimate of how much time you have spent working.

Date Started *

Remaining Estimate Adjust automatically
the estimate will be reduced by the amount of work done,
but never below 0.
 Use existing estimate of 3 days
 Set to (eg. 3w 4d 12h)
 Reduce by (eg. 3w 4d 12h)

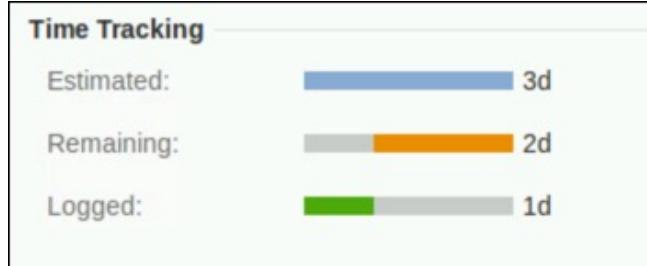
- d. Click **Log**.
- e. In the Issue Detail panel, notice that the Remaining estimate has been reduced by the amount of time you logged.



Don't confuse the Estimate field here with your original time estimate. This is the story point estimate. The only place the original *time* estimate is displayed (once some time has been logged) is in the Time Tracking graph.

- f. Click the issue link at the top of the Issue Detail panel to view the issue in Jira. Find the Time Tracking graph in this view. Your original time estimate is preserved in the Estimated bar. The Remaining bar shows the result of subtracting all logged work from the Estimated value. In

other words, in this example, the 1 day logged was subtracted from the original time estimate of 3 days, indicating there are 2 days of work remaining to complete this issue.



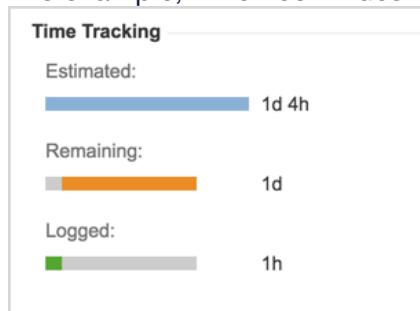
2. Log time on another issue. This time, you realize there isn't as much work here as you originally thought. You will reduce the remaining time estimate in addition to logging your actual work.
 - a. Click the browser's back button to return to the sprint board. Select another issue and click Log Work.
 - b. Enter some time in the Time Spent field. Do **not** press Enter as it will activate the Log button and the window will close.
 - c. In the Remaining Estimate section, select **Reduce by** and enter a time value.

A screenshot of a 'Log Work' dialog box. It includes fields for 'Time Spent' (1h), 'Date Started' (22/May/18 7:39 PM), and 'Remaining Estimate'. Under 'Remaining Estimate', the 'Reduce by' option is selected, and the value '4h' is entered in the input field. A note below explains that the estimate will be reduced by the amount of work done, but never below 0.



In this example, only 1 hour of work is being logged, but we're asking the remaining time estimate be reduced by 4 hours.

- d. Click **Log**.
- e. Check the Time Tracking graph for your issue. It might look similar to this example, which continues the previous example from step c.





The graph shows the original time estimate unchanged. The remaining estimate has been reduced by 4 hours; yet only 1 hour has been logged.

3. Now log some work on either a different issue or one that you've already logged time on (your choice). You've discovered this issue will take longer than you had estimated.
 - a. Select an issue and click Log Work.
 - b. Enter an increment of time in the Time Spent field. Do **not** press Enter.
 - c. Select the **Set** option and enter a value that increases the current Remaining value by 1 day.

Time Spent * (eg. 3w 4d 12h) [?](#)
An estimate of how much time you have spent working.

Date Started * [Edit](#)

Remaining Estimate

Adjust automatically
the estimate will be reduced by the amount of work done,
but never below 0.

Use existing estimate of 4 days

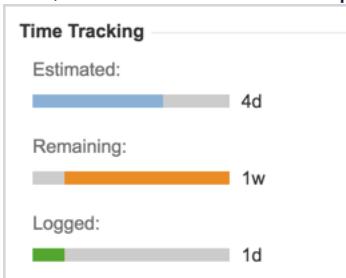
Set to (eg. 3w 4d 12h)

Reduce by (eg. 3w 4d 12h)



In this example, a day of work is being logged against an existing remaining estimate of 4 days. We're asking that the remaining time estimate be increased, however, to 5 days.

- d. Click **Log**.
4. Examine the Time Tracking graph for your issue. It might look similar to this one, which continues the previous example from step 3c.



Here, one day has been logged against that original time estimate of 4 days. We're asking Jira to change the remaining time estimate to 5 days (shown here as 1 week). The original estimate is retained so the burndown chart can accurately monitor the progress of the sprint.

Congratulations on completing the lab!

Lab 6 - Sprints for Complex Projects

Scenario

The sprint has begun and the developers are now subdividing some issues into sub-tasks and beginning to log work against these and complete their stories.

The Scrum master also creates a Scrum of Scrums board as an aid to sharing information across teams such as a view into the dependencies and their progress and also looking for any impediments.

You will log in as Alana Grant, the scrum master and lead developer. This assures you have the necessary privileges to perform all of the steps described above.

Exercise 1 – Working with Sub-tasks

High-level instructions

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on page 42.

1. Log in to Jira Software as Alana Grant (**agrant/Charlie!**)
2. Open the **TIS Scrum Lab 2** board. Configure the board to use Time Tracking.



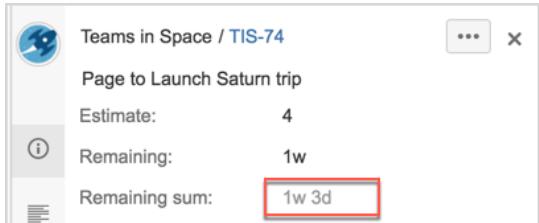
Notice the **Remaining** field for each issue in the sprint already has a value. This illustrates the fact there is only one set of estimation and time tracking data for an issue. If it is set in another board, it pertains to all boards that include the issue.

3. Create two sub-tasks for one of your larger stories. Choose a story for which you have not logged any time yet. *Hint:* Check the Time Tracking chart.
4. Enter an *original* time estimate for each sub-task in the **Remaining** field of the Issue Detail view panel.
5. Select the parent issue.



How are the sub-task original estimates reflected in the parent issue detail panel?

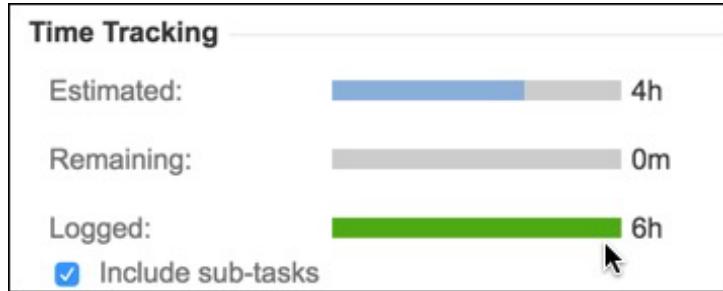
Answer: The parent issue original estimate remains the same, but there is a new field, **Remaining sum**, added to the display. The Remaining sum is calculated by adding the sum of the sub-tasks' estimates to the parent issue remaining estimate. In this example, the parent issue had an original estimate of 1w. The sum of the two sub-tasks estimates = 3d. Thus, the **Remaining sum** field shows **1w 3d**.



6. View the parent issue in Jira to look at the Time Tracking panel. Uncheck **Include sub-tasks**. What happens to the original and remaining time estimates? It should return to its earlier original estimate value.
7. Now turn on **Include sub-tasks** once more.
8. Open the **Log work** window for one of the sub-tasks and log the full amount of time you estimated. Use the default **Adjust Automatically** option.
9. Check the Time Tracking chart for the parent issue to see the effect of logging work against the sub-task.
10. In the same page, find the list of sub-tasks for this parent issue and notice the one for which you just logged time. It indicates it is 100% complete. Resolve this sub-task from its context menu. Use **Fixed** for the resolution.



11. Select the other sub-task from the list of sub-tasks (shown above) and log the total Remaining time against it.
12. Resolve the second sub-task from the Scrum board by dragging it to the Done column.
13. Jira recognizes that both sub-tasks for the parent issue are now resolved and it prompts you to resolve the parent issue, as well, if you wish. Go ahead and resolve the parent issue. When you click **Update**, the Resolve Issue window opens.
14. In the Resolve Issue window, log an amount of time that equals the remaining estimate (effectively zeroing out the remaining estimate). Let Jira automatically adjust the remaining estimate.
15. Select another story and create a sub-task for it.
 - a. Enter an estimate for the sub-task and then log 100% of the time estimate for the sub-task.
 - b. Resolve the parent issue, logging 100% of the remaining estimate. Note: You can't drag the parent issue to the Done column, so you'll have to resolve it from the Jira issue window.
 - c. Leave the sub-task open. Your Time Tracking chart should look similar to this:



An interpretation of this chart is provided in step 16 of the Detailed Instructions.

16. Try to complete the sprint.



What happens?

Answer: You are not allowed to complete a sprint if there are any open sub-tasks.



How could you resolve this and complete the sprint?

Answer: Either resolve the open sub-task or convert it to an issue (or story). In the latter case, the issue would simply be moved into the backlog when you complete the sprint.

17. Since the sub-task has logged 100% of its time estimate, we can assume that the task is complete. Go ahead and resolve it and then complete the sprint.

18. Open the other scrum board that we worked on in the previous lab, **TIS Dev Team Scrum Board**.

19. Create a sub-task for one of these issues.



Why doesn't it appear on the board?

Answer: The board's filter prevents it from showing. It filters issues by epic and sub-tasks can't be assigned to an epic.



How can we work on this sub-task?

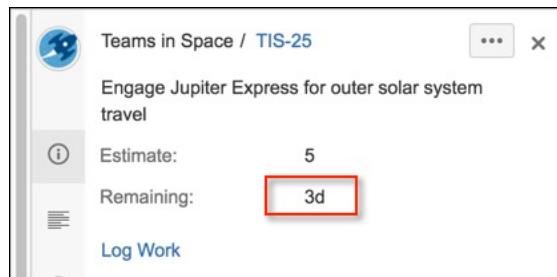
Answer: You can perform actions (such as changing status and logging time) from the Jira issue page. This is not a very satisfactory solution, however, because you can't set an original estimate for the sub-task in this window. That can only be done in the Issue Detail view of the board. Best to avoid using epics to filter issues for a board if you expect to use sub-tasks.

Detailed instructions

1. Log in to Jira Software as Alana Grant (**agrant/Charlie!**).
2. Open the **TIS Scrum Lab 2** board and configure it to use time tracking.
 - a. Open the Configure Board page and click **Estimation**.
 - b. Select **Remaining Estimate and Time Spent** in the Time Tracking section.
 - c. Click **Back to board**.

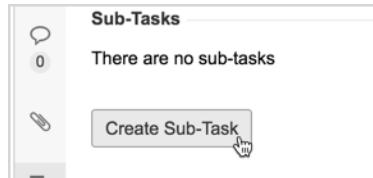


Notice that the **Remaining** field for each issue in the sprint already has a value (either the original estimate or the remaining estimate). This illustrates the fact that there is only one set of estimation and time tracking data for an issue. If it is set in another board, it pertains to all boards that show the issue.



3. Create two sub-tasks for one of your larger stories. Choose one for which you have not logged any work time yet. You can check this by clicking the issue link in the Issue Detail panel to open the issue in Jira.

- a. In the Active Sprint view, select the issue for which you will create a sub-task. In the Issue Detail view, click **Create Sub-Task**.



- b. In the Create Sub-task window, enter a **Summary** for the sub-task.



Notice that the only option you have for the issue type is **Technical task**. Sub-tasks are a special type of issue that has the parent-child relationship with the issue from which it was created.

- c. To quickly and easily create the second sub-task, check the **Create another** checkbox, then click **Create**.

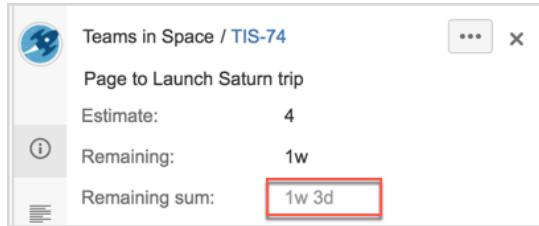
- d. Enter a Summary for the second sub-task, uncheck **Create another** and then click **Create**.
 - e. Back in the Active Sprint view, your sub-tasks now appear directly beneath their parent issue.
4. Enter an original time estimate for each of the sub-tasks.
- a. Select the sub-task in the board.
 - b. Enter the original time estimate in the **Remaining** field.

5. Select the parent issue.

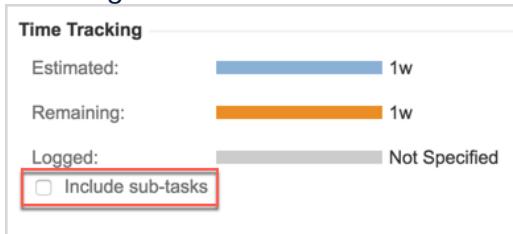


How are the sub-task original estimates reflected in the parent issue detail panel?

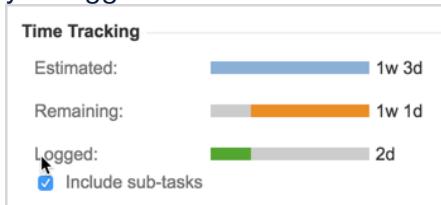
Answer: The parent issue original estimate remains the same, but there is a new field, **Remaining sum**, added to the display. The Remaining sum is calculated by adding the sum of the sub-tasks' estimates to the parent issue remaining estimate. In this example, the parent issue had an original estimate of 1w. The sum of the two sub-tasks estimates = 3d. Thus, the **Remaining sum** field shows **1w 3d**.



6. Open the parent issue in Jira by clicking the issue link in the Issue Detail panel. Notice that **Include sub-tasks** is enabled in the Time Tracking area. Uncheck it to see what happens. The screenshots below show the difference between including sub-tasks in the calculation and not including them.



7. Turn on **Include sub-tasks** once more for parent issue.
8. Log some time against one of the sub-tasks
 - Select a sub-task in the active Scrum board.
 - Click **Log time** in the Issue Detail view.
 - Enter the full amount of time you estimated for this sub-task. For example, if the original time estimate was 1d, log 1d.
 - Use the default setting of Adjust Automatically.
 - Click **Log**.
9. Open the parent issue in Jira and view the Time Tracking chart to see the time you logged reflected in the chart.



10. In the same page, find the list of sub-tasks for this parent issue and notice that the one for which you just logged time shows it is 100% done. Resolve this sub-task from its context menu, selecting **Resolve Issue**. Use **Fixed** for the resolution.

The screenshot shows a 'Sub-Tasks' section with two items:

- 1. Do some technical work: Status OPEN, Alana Grant, 100% complete, progress bar green.
- 2. Do some more technical work: Status OPEN, Alana Grant, 0% complete, progress bar orange.

11. Go back to the sprint board. Notice that the sub-task that you resolved has now been moved to the Done column.

The screenshot shows the 'Sprint Board' with three columns: To Do, In Progress, and Done. An issue TIS-74 is in the In Progress column. Its sub-task 'Do some more technical work' is also in the In Progress column. After resolution, it is moved to the Done column, highlighted with a red border.

12. Select the other sub-task from the list of sub-tasks (shown above) and log time against it. Log the total amount of time estimated for this sub-task.
 13. Back in the Scrum board, notice that the Remaining estimate for this sub-task is now 0. Resolve this sub-task by dragging it to the Done column. Use **Fixed** for the resolution.
 14. Jira recognizes that all sub-tasks for this issue are complete and asks if you also want to close the parent issue. Select **Resolve Issue** and click **Update**.

The dialog box is titled 'Update Parent Issue'. It contains the message: 'All sub-tasks for parent issue TIS-74 are now Done.' Below it, it says 'To update this parent issue to match, select a transition:' with two options: 'Resolve Issue' (selected) and 'Close Issue'. At the bottom are 'Update' and 'Cancel' buttons.

15. In the ensuing **Resolve Issue** window, select Fixed for the resolution and enter a value in the **Time spent** field that is the same as the remaining estimate.

The dialog box is titled 'Resolve Issue'. It includes a note: 'Resolving an issue indicates that the developers are satisfied the issue is finished.' The 'Resolution' dropdown is set to 'Fixed'. The 'Time Spent' field is highlighted with a red box and contains '1w'. Other fields include 'Fix Version/s' (2.0), 'Assignee' (Alana Grant), 'Date Started' (23/May/18 1:32 PM), and 'Remaining Estimate' options. The 'Time Spent' field is the primary focus.

16. Select another issue and create a sub-task for it and then complete the parent issue, leaving the sub-task open.

- Assign an original time estimate for the sub-task that is somewhat less than the original estimate for the parent task.
- Log 100% of the original estimate value for the sub-task but do not resolve it.

The screenshot shows a Jira interface for managing sub-tasks. A single sub-task titled "1. Do some technical work" is listed. To its right, the status is shown as "OPEN", assigned to "Alana Grant", and the progress bar is at 100%. There are also "..." and "+" buttons for more actions.

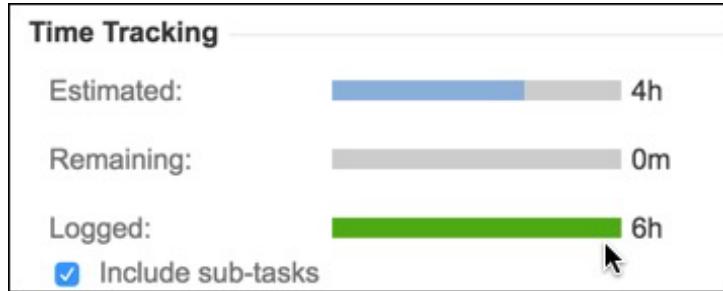
- In the Scrum board, select the parent issue. Notice that you can't drag it to the Done column when it has sub-tasks. Open it in Jira instead, and then resolve it from the **Resolve Issue** button.

The screenshot shows the Jira issue details page for TIS-55. It displays various issue fields: Type (Bug), Priority (Major), Affects Version/s (2.0), Component/s (Web Site), Labels (None), Epic Link (Large Team Support), Platform (PC/MAC), and Sprint (TIS Sprint 2). The "Resolve Issue" button is highlighted with a cursor. The status is OPEN and the resolution is Unresolved.

- In the Resolve Issue window, enter a time value that will use up the remaining time estimate.

The screenshot shows the "Resolve Issue" dialog box. It includes fields for Resolution (Fixed), Assignee (Jennifer Evans), Time Spent (4h), Date Started (26/Sep/16 4:52 PM), and Remaining Estimate options. The "Use existing estimate of 4 hours" option is selected. The "Comment" section contains rich text editing tools. At the bottom are "Resolve" and "Cancel" buttons.

- Your Time Tracking chart might look similar to this:



i In this example, the original time estimate for the parent issue was 4h. 2h was logged against the sub-task (100% of its original estimate). Then 4h were logged against the parent issue when it was resolved. The total amount of time logged against the parent and child was 6h.

17. From the Scrum board, click **Complete sprint**. Jira Software won't allow you to complete the sprint because one of your stories has a sub-task that is still open.

? What can you do in order to complete the sprint?

Answer: Since the sub-task has logged 100% of its time estimate, we can assume that the task is complete and go ahead and resolve it. Alternatively, you can convert the sub-task into a regular issue and the sprint can be completed. The sub-task, turned issue, will simply be put back into the backlog.

18. Go ahead and resolve the sub-task (as you did in step #10) and then complete the sprint.
19. Open the other scrum board that you used in the previous lab, TIS Dev Team Scrum Board.



If there is no active sprint in this board, just create a sprint and add a few issues to it and start the sprint. You may have inadvertently completed the sprint in an early exercise.

20. Create a sub-task for one of these issues.

? Why doesn't it appear on the board?

Answer: The board's filter prevents it from showing. It filters issues by epic and sub-tasks can't be assigned to an epic.

? What options do you have for working on this sub-task?

Answer: You can perform actions (such as changing status and logging time) from the Jira issue page. This is not a very satisfactory solution, however, because you can't set an original estimate for the sub-task in this window. That can only be done in the Issue Detail view of the board. Best

to avoid using epics to filter issues for a board if you expect to use sub-tasks.

Exercise 2 – Creating a Scrum of Scrums Board

At Teams in Space, the number of development teams has grown and the number of requirements increased. A decision was made to improve coordination of teams by holding a weekly Scrum of Scrums meeting.

We have to consider what aspects of the information available in Jira are relevant and which can be ignored. As a Scrum of Scrums is focused on high-level progress and impediments to progress, we can ignore any stories that are progressing without impediments.

The Scrum master has been tasked with creating a board to support the Scrum of Scrum meetings. We want to make sure that the team representatives at the meeting can focus on reporting on progress and working together to resolve any impediments stopping progress. To achieve this the board should show all epics and only stories flagged as an impediment.

You will log in as Alana Grant to perform this task.

High-level instructions

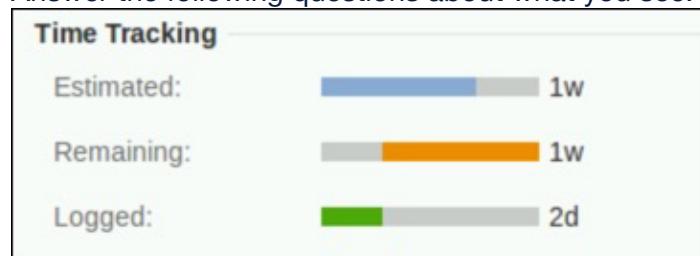
If you would prefer to follow step-by-step instructions, go to Detailed Instructions on the page 49.

1. Search for issues from all projects except “TIS Complete Board”. Exclude issues that are closed or resolved.
 One of the projects, “TIS Complete Board”, duplicates the “Teams in Space” project, so exclude any issues in this project.
2. Select 4 or 5 issues and flag them as impediments. Ideally, these should represent a variety of projects. You will have to first add the Flagged field to each of these issues.
3. Create a saved filter that returns all epics and flagged issues that are not in the “TIS Complete Board” project.
4. Create a Kanban board based on the saved filter you created in the last step. Name it “Kanban board for Scrum of Scrums”.
5. Configure the board in order to share the filter with everyone. Allow ranking??
6. Create a swimlane for issues flagged as impediments.
7. Test the board.



The original time estimate is unchanged, but the Remaining bar has been reduced by 4 hours. Only 1 hour has been logged.

- You now log some time on either a different issue, or one that you've already logged time on. You discover that this one will take longer than you had originally estimated. Log some time in the issue and also increase the remaining estimate by 1 day. *Hint:* Use the **Set** option.
- What does the Time Tracking graph look like now? It might look similar to this example. Here, the original time estimate was 5 days, the Remaining estimate had been already reduced to 4 days after a previously logging. Now, the developer logs an additional day but sets the remaining estimate to 5 days. Answer the following questions about what you see.



What does the orange portion of the Remaining bar represent?

Answer: The amount of the remaining estimate - 1 week. A total of 2 days has been logged against the issue. Normally, the remaining estimate would have been reduced to 3 days, but it was reset to 5 days - a net gain of 2 days to the remaining estimate.



What does the gray portion of each of the bars seem to represent?

Answer: It shows that the remaining estimate has been manually adjusted with the addition of 2 more days.

Detailed instructions

- Search for issues from all projects except “TIS Complete Board”. Exclude issues that are closed or resolved.



One of the projects, “TIS Complete Board”, duplicates the “Teams in Space” project, so exclude any issues in this project.

- Select **Issues > Search for issues** from the main menu.
- Enter the following JQL query into the search bar:

The search interface shows the following JQL query in the search bar:

```
project != "TIS Complete Board" and status not in (Closed,Resolved)
```

- c. Execute the search by pressing Enter.
2. Select 4 or 5 issues and flag them as impediments. Ideally, these should represent a variety of projects. You will have to first add the Flagged field to each of these issues.
- a. Select an issue.
 - b. In the Jira issue panel, select **Admin > Add field**.

The screenshot shows a Jira issue panel for BDV-17. The issue title is "Upload pictures to the website and email all customers". The "Details" section shows Type: Sub-task, Status: OPEN, and Priority: Major. A context menu is open over the "Admin" button, with "Add field" highlighted. The menu also includes "Where is my field?", "Permission helper", and "Notification helper".

- c. In the Add Field window, select **Flagged** from the options list. Click **Edit Field**.
- d. In the Edit Flagged window, select **Impediment** and click **Submit**.
- e. You should now see the Flagged field in the Jira issue panel.

The screenshot shows the same Jira issue panel as before, but now the "Details" section includes a "Flagged" field. The dropdown menu for "Flagged" shows "Impediment" selected. The "Labels" field also contains "None".

- f. Follow the same procedure to add the impedance flag to several more issues.
3. Create a saved filter that returns all epics and flagged issues that are not in the “TIS Complete Board” project.
- a. In the Search window, clear the filter field.
 - b. Enter the following new filter query:

```
Project != "TIS Complete Board" AND (issuetype= Epic OR
Flagged = Impediment)
```

- c. Execute the query. You should see your flagged issues, in addition to all epics (except any issues in TIS Complete Board project).
- d. Click **Save as** at the top of the page and name your filter “Filter for Scrum of Scrums”.

4. Create a Kanban board based on the saved filter you created in the last step.
 - a. Click **Boards > View all boards**.
 - b. Click **Create board**.
 - c. Select **Kanban** and then select **Board from an existing saved filter**.
Click **Next**.
 - d. Name the board “Kanban board for Scrum of Scrums”.

Name this board

Board name * Kanban board for Scrum of Scrums

Saved filter * Filter for Scrum of Scrums

Shares No shares

Owner Alana Grant

Saved Filters
Choose from a list of existing filters as a base for your new board. To create a new Saved Filter, save a search in the Issue Navigator.

Back Create board Cancel

- e. Click **Create board**. The board appears. Your impediment issues will be colored coded yellow.
5. Configure the board in order to share the filter with members of all the projects shown on the board. You will also enable ranking.

Filter

Saved Filter Filter for Scrum of Scrums

Shares No shares

Filter Query project != "TIS Complete Board" AND (issuetype = Epic OR Flagged = Impediment)

Ranking Ranking is disabled, as the Filter Query is not ordered by Rank

Add Rank

Projects in board

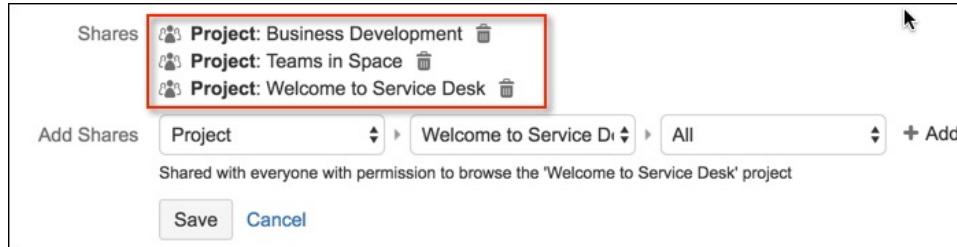
- Teams in Space
- Business Development
- Welcome to Service Desk

View permission

Kanban board sub-filter fixVersion in unreleasedVersions() OR fixVersion is EMPTY

Further filtering of issues for unreleased work.

- a. Open the Configure Board page. On the General tab, you can see the list of projects involved. You’ll have to add all members of each of these projects in the Filter share.
- b. First, enable ranking by clicking **Add Rank**.
- c. Click **Edit filter shares**.
- d. In the Edit Current Filter page, add the first share by selecting **Project**, then **Business Development**, then **All**.
- e. Click **Add**.
- f. Repeat steps d and 3 to add 2 more shares, changing the project name to each of the other two names of projects included in this filter.



- g. Click **Save**. You will be returned to the Issue search page.
 h. Reopen the Kanban board for Scrum of Scrums.
6. Create a swimlane for issues flagged as impediments.
- Open the Configure board page.
 - Select **Swimlanes** in the left panel.
 - Make sure the **Base swimlanes on** field is set to **Queries**.
 - Add a new swimlane. Name it “Impediments” and use the following JQL to define it: **Flagged = Impediment**

Name	JQL	Description
Impediments	Flagged = Impediment	

- e. Click **Add**.
7. Confirm the changes you've made in the Kanban board. Do you think this board would be helpful for the Scrum of Scrums team?

Congratulations on completing the lab!

Lab 7 – Managing Releases

Scenario

Alana Grant has been asked by the product owner to help him decide whether it would make sense to combine versions 2.0 and 2.1. You examine the Release Hub for summary information and then make your recommendation to merge the two versions.

The Release Hub provides warnings for issues that are out of sync with the development tools. Alana wants to gain a better understand of what triggers these warnings. She examines the Manage Warnings window and then looks for an issue that, if completed, would cause a warning, testing the scenario.

She also releases the current version, examining the release options.

Exercise 1 – Exploring the Release Hub

High-level instructions

If you would prefer to follow step-by-step instructions, go to Detailed Instructions on the next page.

1. Log in to Jira Software as Alana Grant (**agrant/Charlie!**).
2. Open the Release Hub for the Teams in Space project.



How many unreleased versions are listed?

Answer: 4 versions are unreleased: 2.0, 2.1, 2.2, and 3.0.

3. Examine the summary information for versions 2.1 and 2.0.
4. Manage the versions and then click **Merge** to merge 2.1 into 2.0. Return to the Release Hub main screen. (You may need to refresh the page to see version 2.1 removed.)
5. Click version 2.0 to examine the details.



How many issues appear in each view category? Do you see **development information** provided for any of the issues? Take note of these issues, as you will need this information later.

Answer: Here is one suggested set of numbers. Your image may differ somewhat, however.

- Warnings: 0 issues listed
- Issues in version: 23 issues, 3 with development information (TIS-26, TIS-74, and TIS-75)
- Issues done: 3 issues
- Issues in progress: 3 issues in progress, 2 with development information (TIS-26 and TIS-74).
- Issues to do: 17 issues, none with development information

- In the Warnings view, click **Manage Warnings** and review the descriptions of each of the warning conditions that are enabled. Consider the issues you found that displayed development data. Can you relate these issues to the warning conditions shown here?
- Find an issue that, if it were to be resolved, would be out of sync with the development tools, thereby triggering a warning.



Only 3 issues in the product backlog have development data available: TIS-26, TIS-74, and TIS-75. Only TIS-74 can serve this purpose without actually changing the data in the development tools.

- Open TIS-74 in the Jira issue page and resolve it.
- Return to the Version 2.0 summary in Release Hub to confirm the warning was triggered. You may need to reload the page to see it.



Which warning condition does it satisfy?

Answer: TIS-74 is Under Review. It triggers a warning because it shows up in Jira as being complete, yet it has an open pull request.

- Change the status of TIS-74 back to **In Progress** to clear the warning.
- Release version 2.0.
 - Click **Release** from the Version 2.0 summary view page.
 - In the Release 2.0 window, examine the Bamboo settings, but keep the default values here.
 - Change the **Remaining issues** setting to **Move issues to version 2.2**.
 - Change the release date to today's date
 - Click **Release**.
 - When build has finished, the status will be reported in the upper right corner of the Version 2.0 summary page. The version is now listed as **Released** at the top of the page.

Version 2.0 RELEASED <small>Start date not set Released: 01/Oct/16 Release Notes</small> <small>Version 2.0</small>	<small>Build #6 Build TIS-26-Red-tit...</small> <small>Completed 5 minutes ago</small>
---	---

Detailed instructions

- Log into Jira Software as Alana Grant (**agrant/Charlie!**).
- Open the Release Hub for the Teams in Space project.
 - Open the Teams in Space project from the Projects menu.
 - Click the Release Hub icon in the left panel of the main project page.





How many unreleased versions appear?

Answer: 4 versions are unreleased: 2.0, 2.1, 2.2, and 3.0.

- Examine the summary information for versions 2.1 and 2.0. Hover your cursor over the progress bar of each of the versions. A popup summary appears.

Version	Status	Progress	Start date	Release date	Description
3.0	UNRELEASED	<div style="width: 50%;"></div>		26/Sep/15	Version 3
2.2	UNRELEASED	<div style="width: 80%;"></div>			Version 2.2
2.1	UNRELEASED	<div style="width: 70%;"></div>			Version 2.1
2.0	UNRELEASED	<div style="width: 20%;"></div>			Version 2.0



How many total issues would there be in version 2.0 if we merged it with version 2.1?

Answer: You might see slightly different numbers, but it's probably close to 15 total issues between versions 2.0 and 2.1.



How does this merged total compare to the number of issues in each of the remaining two versions?

Answer: The total number of issues for the merged version would be slightly higher than version 2.2 (in this example). The mere number of issues, however, is not as significant as the number of sprints required to complete the version.

- Merge versions 2.0 and 2.1.
 - Click **Manage Versions**.
 - In the Manage Versions window, click **Merge**.
 - Select **2.1** in the **Merging From Versions** field and **2.0** in the **Merge To Version** field.

Merge Versions

⚠️ WARNING: You cannot un-merge these versions once they have merged.

Merging From Versions: 2.1

Merge To Version: 2.0

Merge Cancel

- d. Click **Merge**.
 - e. Return to the **Releases** page of the Release Hub by clicking the Back arrow of the browser. If you still see 2.1 listed, reload the page.
5. Click the link for version **2.0** to view the details. Your lists could differ from the screenshots shown here.
- a. Take note of the issues that provide **development information**. You will need this later.
 - b. View the full list of issues by clicking the **Issues in version** tab. Do you see some issues that show development information?

0 Warnings		23 Issues in version	3 Issues done	3 Issues in progress	17 Issues to do
1–23 of 23					
P	T	Key	Summary	Assignee	Status
✗	⚠	TIS-9	After 100,000 requests the SeeSpaceEZ server dies	Jennifer Evans	OPEN
↑	⚠	TIS-26	Engage the Red Titan Hotel as a preferred provider	Ryan Lee	IN PROGRESS MERGED
↑	⚠	TIS-38	Suggested Destinations	Ryan Lee	OPEN
↑	⚠	TIS-41	Update LodgingController to handle multiple travel providers in one reservation	Max Taylor	OPEN
✗	⚠	TIS-7	500 Error when requesting a reservation	Emmet Paris	OPEN
✗	⚠	TIS-8	Requesting available flights is now taking > 5 seconds	Max Taylor	OPEN

- c. View the list of issues that are done. Which issues are they?
- d. View the list of issues that are in progress. How many are there? Do you see some, showing development information?

0 Warnings		23 Issues in version	3 Issues done	3 Issues in progress	17 Issues to do
1–3 of 3					
P	T	Key	Summary	Assignee	Status
↑	⚠	TIS-26	Engage the Red Titan Hotel as a preferred provider	Ryan Lee	IN PROGRESS MERGED
✗	⚠	TIS-56	Add pointer to main.css file to instruct users to create child themes	Jennifer Evans	IN PROGRESS
✗	⚠	TIS-74	Page to Launch Saturn trip	Alana Grant	IN PROGRESS UNDER REVIEW

- e. Finally, view the Warnings view. It is likely no issues appear here.
6. Examine the types of warnings in effect for this project.
- a. In the Warnings view, click **Manage Warnings**.
 - b. Review the descriptions of each of the three warning conditions that are enabled.

Manage Warnings	
These warning settings affect all existing and future versions in this project.	
Open pull requests	<input checked="" type="checkbox"/> Enabled
These issues have been marked complete but have open pull requests.	
Unreviewed Code	<input checked="" type="checkbox"/> Enabled
These issues have been marked complete but the commits are not part of a pull request or review.	
Failing Builds	<input checked="" type="checkbox"/> Enabled
These issues have been marked complete but have failing builds.	

7. Think about and examine the issues that have development information displayed. Select one of these issues that, if completed, would trigger a warning from Release Hub?



Only 3 issues in the product backlog have development data available: TIS-26, TIS-74, and TIS-75. Only **TIS-74** can achieve this purpose without making changes to the data in the development tools.

8. TIS-74 is currently **Under Review**. If it were to be completed, it would trigger the **Open pull requests** warning. Change its status to Resolved to test the trigger.
 - a. Open the Release Hub view showing TIS-74.
 - b. Click the issue link to open it in Jira.

Warnings	Issues in version	Issues done	Issues in progress	Issues to do
0 Warnings	23 Issues in version	3 Issues done	3 Issues in progress	17 Issues to do

1–3 of 3

P	T	Key	Summary	Assignee	Status	Development
↑	!	TIS-26	Engage the Red Titan Hotel as a preferred provider	Ryan Lee	IN PROGRESS	MERGED
↑	!	TIS-56	Add pointer to main css file to instruct users to create child themes	Jennifer Evans	IN PROGRESS	
↑	!	TIS-74	Page to Launch Saturn trip	Alana Grant	IN PROGRESS	UNDER REVIEW

- c. Click **Resolve issue**. Select **Fixed** as the resolution and then click **Resolve**.
9. Reopen the Release Hub and select Version 2.0. Click the browser's Back button and then reload the page to see the warning.

Warnings	Issues in version	Issues done	Issues in progress	Issues to do
1 Warnings	23 Issues in version	4 Issues done	2 Issues in progress	17 Issues to do

30+ days overdue

Warnings indicate when the status of a JIRA issue doesn't reflect related development activity. For example: an issue marked complete that has an open pull request should be marked as still being in progress. [Manage Warnings](#)

Open pull requests

These issues have been marked complete but have open pull requests.

P	T	Key	Summary	Assignee	Status	Development
✗	!	TIS-74	Page to Launch Saturn trip	Alana Grant	RESOLVED	UNDER REVIEW

10. Change the status of TIS-74 back to **In Progress** to clear the warning.
 - a. From the Warning view, select the issue link to open it in Jira.
 - b. Click **Reopen** and then confirm the action in the next window.
 - c. Return to the Release Hub by clicking the browser's back arrow.
11. Release version 2.0.
 - a. Click **Release** from the Version 2.0 summary view page.
 - b. In the Release 2.0 window, examine the Bamboo settings, but keep the default values here.

- c. Change the **Remaining issues** setting to **Move issues to version 2.2**.
- d. Change the release date to today's date.

Release 2.0

Bamboo

Release with no build
 with new build
 with existing build

Plan: Build TIS-26-Red-titan-hotel-preferre

Stages: Default Stage

Remaining issues

19 unresolved issues Ignore and proceed with release
 Move issues to version: 2.2

Release date: 1/Oct/16

Release **Cancel**

- e. Click **Release**.
- f. When build has finished, the status is reported in the upper right corner of the Version 2.0 summary page. The version is now listed as **Released** at the top of the page.

Version 2.0 **RELEASED**

Start date not set Released: 01/Oct/16 Release Notes

Build #6 Build TIS-26-Red-tit...
Completed 5 minutes ago

Version 2.0

Congratulations on completing the lab!

Lab 8 – Using Reports and Dashboards

Scenario

In this lab, you will be analyzing specific reporting requirements in Exercise 1, and exploring several Jira dashboard gadget charts in Exercise 2.

Several team members have asked you to help the team with their reporting needs. They all need reports appropriate for their role and work responsibilities. You should advise each team member on the appropriate report(s) and how to interpret them.

In the High Level instructions section of Exercise 1, you are given three scenarios, each of which is accompanied by a requirements breakdown for that scenario. You should recommend the best report for each scenario.

The Detail instructions section of Exercise 1 briefly restates the scenario and then provides the answer and additional supporting information.

To help you in this task, this table of available Jira Software reports is provided.

Report type	Restrictions	Purpose
Burndown Chart	SCRUM ONLY	Shows the actual and estimated amount of work to be done in a sprint.
Control Chart	SCRUM/KANBAN	Shows the cycle time (or lead time) for your product, version or sprint.
Cumulative Flow Diagram	SCRUM/KANBAN	It is an area chart that shows the various statuses of work items for a particular time interval.
Epic Burndown	SCRUM ONLY	Shows you how your team is progressing against the work for an epic.
Epic Report	SCRUM ONLY	Shows a list of complete, incomplete and unestimated issues in an epic.
Release Burndown	SCRUM ONLY	Shows you how your team is progressing against the work for a release.
Sprint Report	SCRUM ONLY	Shows the list of issues in each sprint.
Velocity Chart	SCRUM ONLY	Shows the amount of value delivered in each sprint, enabling you to predict the amount of work the team can get done in future sprints.

Report type	Restrictions	Purpose
Version Report	SCRUM/KANBAN	Shows your team's progress towards completion of a version.

Exercise 1 – Selecting the Appropriate Report

High-level instructions

1. Mary is the Scrum Master for the Rocket Engineering team and she would like to review a report before each daily scrum meeting in order to check the progress on the sprint to date. She would particularly like to see if the team is on target to complete at the end of the sprint.

The requirements breakdown follows.

- Mary needs to review the progress of the current sprint. This can be achieved with a standard Agile report on the project. The key aspects that she needs to see are:
 - Total scope – how much work
 - Predicted rate of closure to achieve completion
 - How much has been done? What is left to do?
- From this information, she can judge whether there are any concerns for completion on time.

What report do you recommend?

2. Nigel, the product owner for Rocket Engineering, would like to know how many sprints it will take to complete the current estimated backlog. He wants to be able to sufficiently prioritize and estimate the backlog for at least three sprint planning meetings.

The requirements breakdown follows:

- By understanding the number of story points completed in previous sprints (velocity) by the team, Nigel and Mary can predict how many will be completed in three further sprints.
- For example:

Sprint	Committed	Completed
1	25	24
2	25	26
3	28	27
4	30	27

Sprint	Committed	Completed
Totals	108	104

- The average number of story points completed, over the four sprints is 26 ($104 / 4$).
 - The planning horizon, therefore, should be at least 78 story points (26×3)
- What report do you recommend?
3. Nigel, the product owner, would like to be reassured that a team is on course to deliver the next release. He does not want to continually ask Mary and the other scrum masters for a progress report that he would then have to collate.

The requirements breakdown follows:

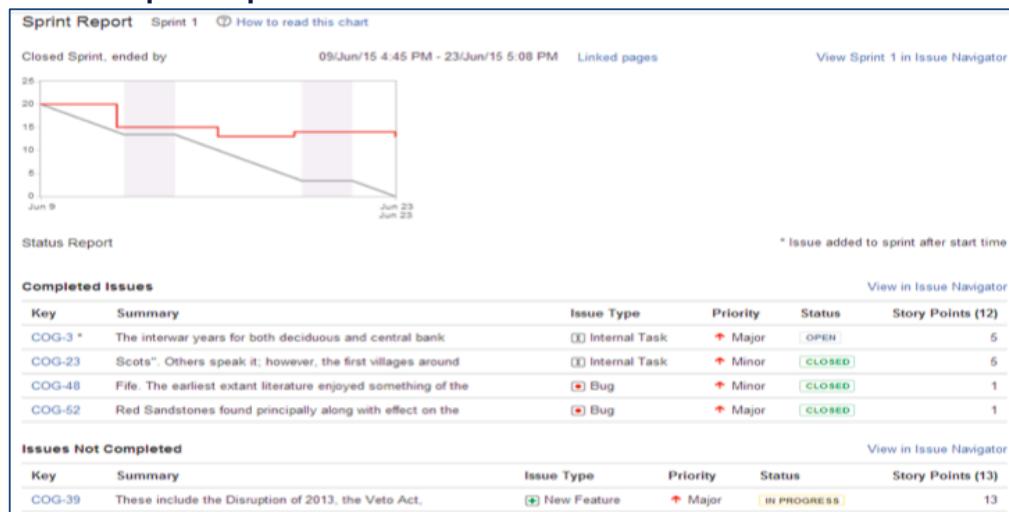
- Nigel does not want to disturb the teams to find out the information. He must be able to access this information by himself.
- The report should show a prediction of completion, as well as a listing of already completed work.

What report do you recommend?

Detailed instructions

1. Mary, the scrum master, needs to review a report before each daily scrum meeting to check progress on the sprint to date.

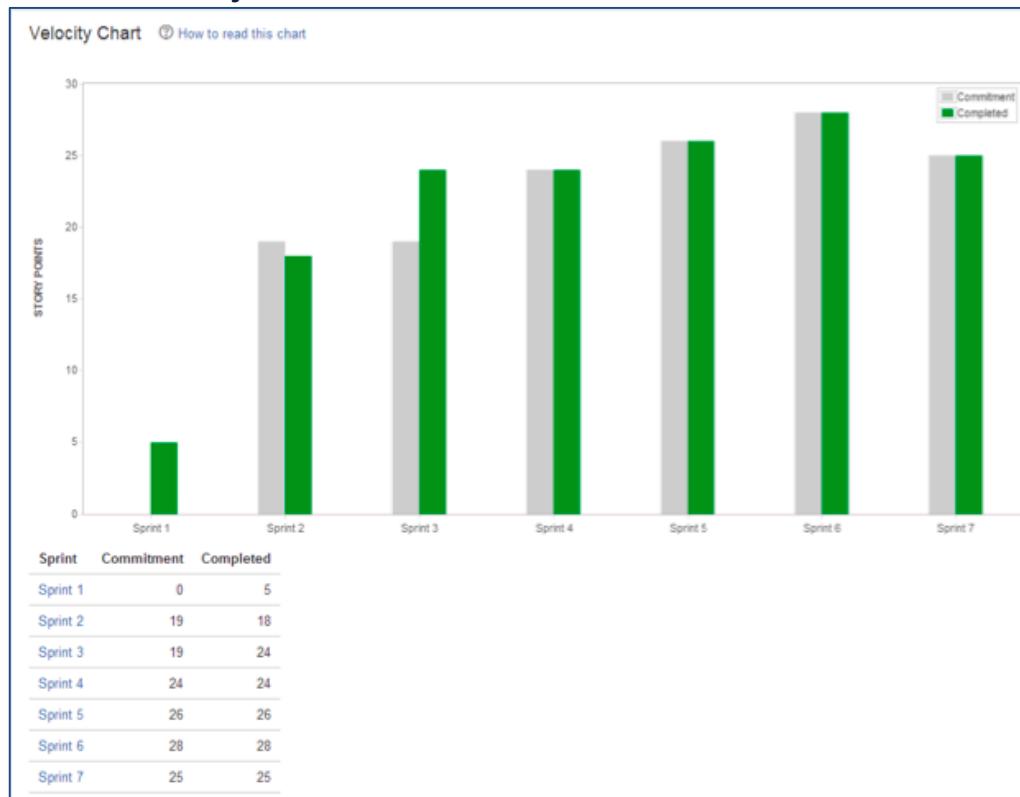
Answer: Sprint report



- By reviewing the sprint report in advance of the standup meeting, Mary will be aware of whether the team is on track. This allows her to

- concentrate on any blockers or changes, reported by the team, that need to be resolved.
- The sprint report consists of two sections - a burndown chart, and a status report on all issues within the sprint.
 - The burndown chart can help identify any challenges for the team, such as overruns. It can also indicate early completion.
 - The status report indicates what issues are done and that are still to be completed. As the issues are listed in order of priority, she can see if any high ranked items are still to be completed.
- Nigel, the product owner, wants to be able to figure how many sprints it will take to complete the current backlog.

Answer: Velocity chart

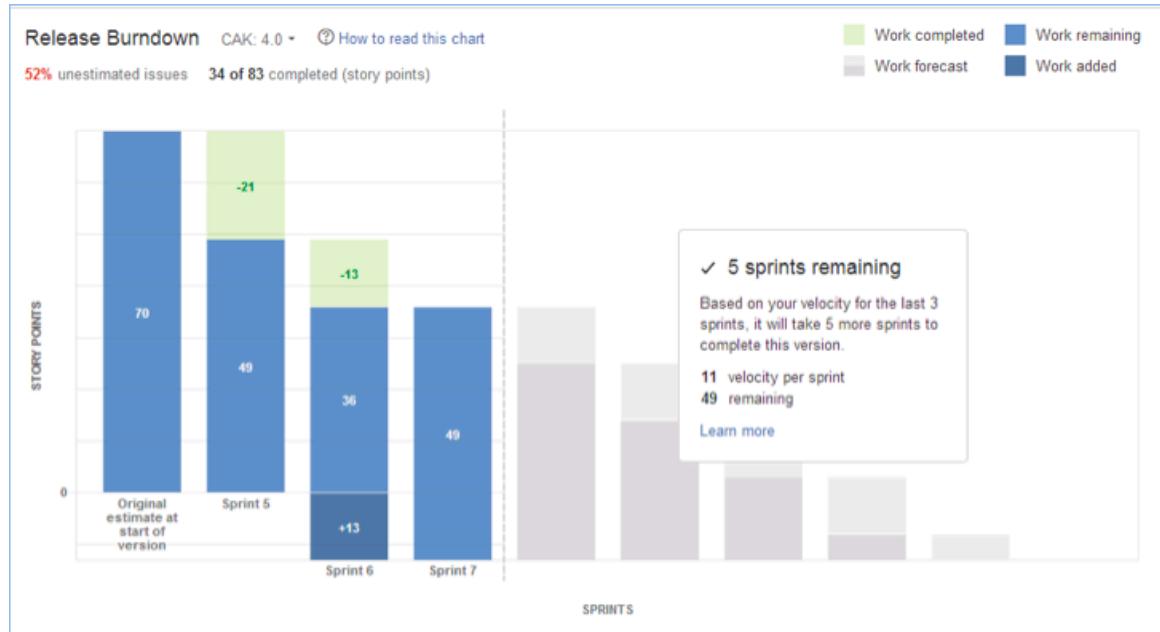


- By reviewing the Velocity of the team across a number of sprints (especially the most recent), Nigel and Mary can predict how many story points will be completed in a single sprint. Using this information, they can manage the estimation of the backlog to ensure that the sprint planning meeting can proceed with enough items present in the product backlog.

- It's not expected that each Sprint will have identical size, as many factors (holiday periods, team secondments, etc.) or have an impact. It should be possible to make reasonable predictions.
- The chart above shows that the team completed 150 story points across 7 sprints. But we can see that the first two sprints were much less. This is typical of a project when it first starts and so they would take the average across the more recent sprints (for example 127 points across 5 sprints giving an average of 25 points per sprint)
- Now Nigel and Mary know the number of story points per sprint that were achieved, they can multiply by 3 to get the number of story points that they would like to have estimated. In our example, this would be $25 \times 3 = 75$ story points.
- Another calculation that can be useful is to calculate the average story points per story completed and then multiply by the number of stories left to be completed to give an indication of total scope. This approach often underestimates, however, because complex stories would be broken up prior to completion. Conversely, some stories may never be developed. Obviously, this leads to overestimation.
- In summary treat any such number calculated as a very crude indicator only.

3. Nigel wants to be sure that the team is on track to deliver the next release.

Answer: Release Burndown



- By creating a scrum board across all the teams, Nigel is able to see progress towards the completion of the Version for each team without bothering the various scrum masters for information and then collating it.

- As Nigel needs to know the predicted completion date for each version, he can use the Release Burndown for each project release on a shared scrum board.
- By viewing the Release Burndown, he can see the historic progress as well as the future predictions based upon the team velocity.

Exercise 2 - Experimenting with Jira Dashboard Gadgets

High Level Instructions

Because of the atypical way that we work on issues during the course labs (all changes to issues happening on a single day), it will be hard to get interesting, meaningful results in the reports or dashboards. However, in this exercise, you will experiment with creating dashboard gadgets just to see how it's done and how different settings can change the graph.

1. Create a new dashboard from the Manage Dashboards page. Name it “Demo Dashboard” and share it with everyone.
2. Open the Demo Dashboard from the Dashboards menu.
3. Add and configure a Created vs Resolved Chart. You will have to expand the list of Jira gadgets by clicking the **Load all gadgets** link.



Try using the **TIS Filter for Lab 4** as the “project or filter” setting. Experiment with the other settings to see how they affect the display. The following graph was configured to look at **10 previous days** using **weekly periods**. It also uses a Collection Operation of **Cumulative**, rather than Count. Your graph, using the same settings, however, may differ.



4. Add and configure a Jira Roadmap gadget using the **TIS Complete** project.
5. Add and configure an Issue Statistics gadget that measures progress on the different issue types. **TIS filter for Lab 4** has good data for this chart. This gadget is helpful for checking to see if you're falling behind on bug fixes, in

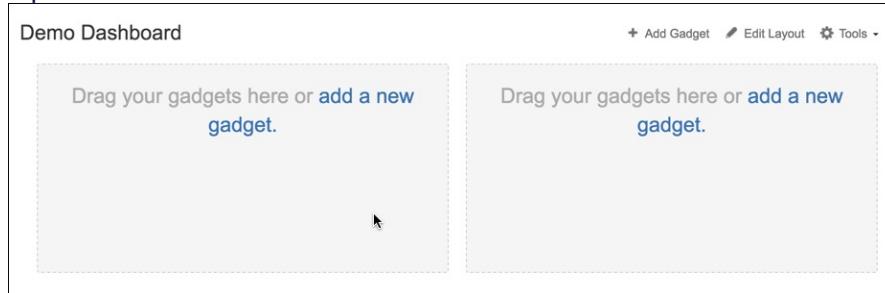
- which case you would either need to either reprioritize the bugs or allocate more resources.
6. Feel free to experiment with other gadgets if you have time.

Detailed Instructions

1. Create a new Jira dashboard.
 - a. Go to **Dashboards > Manage Dashboards**.
 - b. In the Manage Dashboards page, click **Create dashboard**.
 - c. Name it “Demo Dashboard”.
 - d. Select **Any logged-in user** and Click **+ Add** to share the dashboard with anyone who is logged in to Jira. You can accept all other default settings.

The screenshot shows a modal dialog titled 'Create New Dashboard'. It has fields for 'Name' (set to 'Demo Dashboard'), 'Description' (empty), and 'Start From' (set to 'Blank dashboard'). Below these, there's a note: 'Choose a dashboard whose gadgets will be copied to the new dashboard. Alternatively, choose 'Blank dashboard' to create a dashboard with no gadgets.' Under the 'Shares' section, there's a dropdown set to 'Shared with logged-in users' and a button '+ Add'. At the bottom are 'Add' and 'Cancel' buttons. A red box highlights the 'Shares' dropdown.

- e. Click **Add** to complete the dashboard and return to the Manage Dashboards page.
2. Open the Demo Dashboard from the Dashboards menu.



3. Add the Created vs. Resolved gadget to the dashboard.
 - a. Click **add a new gadget** in one of the gadget containers.
 - b. In the Gadgets window, select Jira in the left panel. Then click **Load all gadgets** in the right panel as shown below to expand the number of Jira gadgets.

- c. Scroll through the list to find the **Created vs. Resolved** gadget. Click **Add gadget**.

- d. The gadget is added to the dashboard container, but the window remains open. Close it from the X in the top right corner. You can now edit the gadget's settings.
e. Use the **TIS filter for Lab 4** filter instead of a project name. Begin typing the filter name until a dropdown list of options appears. Select the filter.

- f. Change settings as follows. This configuration may provide a more satisfying result in the graph.
- Period - Weekly
 - Days Previously - 10
 - Collection Operation - Cumulative
 - Display Versions - All versions

Created vs. Resolved Chart

Project or Saved* **TIS filter for Lab 4**

Filter: Search

Project or saved filter to use as the basis for the graph.
[Advanced Search](#)

Period*: **Weekly**

The length of periods represented on the graph.

Days Previously* **10**

Days in the past to collect data for the selected period.

Collection Operation* **Cumulative**

Progressively add totals (1.. 2.. 3), or show individual values (1.. 1.. 1).

Display the Trend of* **No**

Show the number of unresolved issues over time in a subplot.

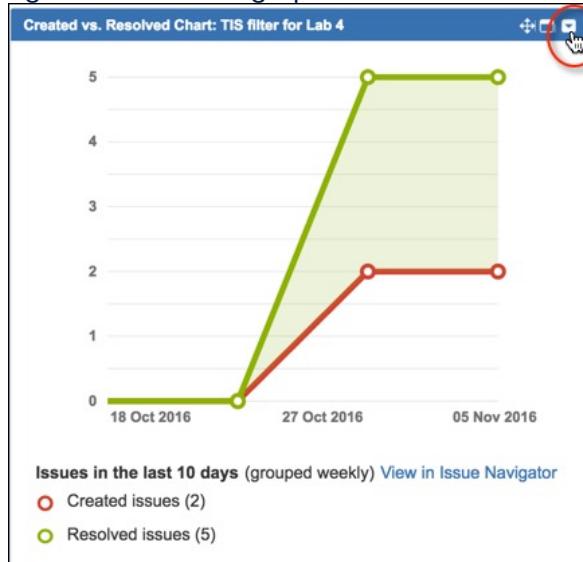
Display Versions* **All versions**

Show when versions were released on the chart.

Auto refresh Update every 15 minutes

Save **Cancel**

- g. Click **Save** to view the graph. Your graph may look different than this example. Feel free to play with the graph settings to see how they affect the result. You can reopen the Edit window from the icon in the upper right corner of the graph.



4. Add the Jira Roadmap gadget to the dashboard.
- Click **Add a new gadget** in the other gadget container.
 - Scroll through the list of Jira gadgets to find the Jira Roadmap gadget.
 - Click **Add gadget**.

- c. Close the Add Gadget window.
- d. Edit the gadget to use the **TIS Complete Board** project. Accept all other settings for now.

JIRA Road Map

Projects: All Projects
Business Development
Teams in Space
TIS Complete Board

The projects to display information for.

Days: 30

Period to cover (in days)

Number of Results: 10

Number of results to display (maximum of 50).

Refresh Interval: Never

How often you would like this gadget to update

Save

- e. Click **Save** to view the graph.



Since the dates for TIS Complete issues are very old, your graph will look pretty discouraging. Real world data would be much more satisfying.

- 5. Add the Issue Statistics gadget to the dashboard. We'll configure it to show the percentage of each issue type to be completed. This can help determine whether we're keeping up with our bug fixes.
- a. Click **Add Gadget** at the top of the dashboard page.
- b. Scroll through the list of Jira gadgets to find the Issue Statistics gadget. Add it to the dashboard and close the window.

- c. Again, use the TIS filter for Lab 4 as the "Project or filter" setting.
- d. Set the Statistic type to **Issue type**.
- e. Set the gadget to **show resolved issue statistics**.

Issue Statistics

Project or Saved* **TIS filter for Lab 4**

Filter:

Project or saved filter to use as the basis for the graph.
Advanced Search

Statistic Type: **Issue Type**

Select which type of statistic to display for this filter.

Sort By: **Natural**

Sort by row total or natural field order.

Sort Direction: **Ascending**

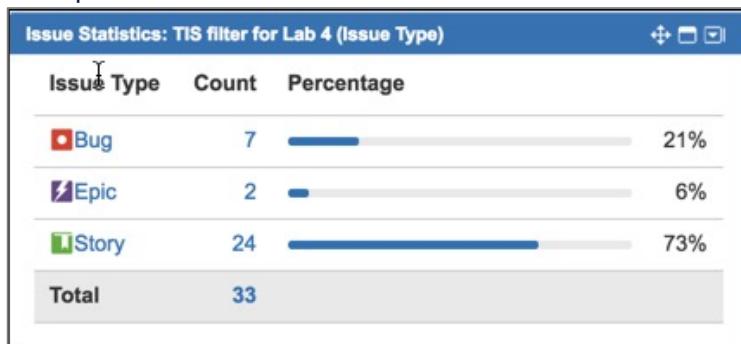
Show Resolved Issue Statistics: **Yes**

Include resolved issues in the set of issues from which statistics are calculated.

Auto refresh: Update every 15 minutes

Save

- f. Click **Save** to view the graph. Your graph may look different than this example.



6. If you have time, feel free to experiment with other types of gadgets. Think about which ones would be most helpful to your stakeholders.

Congratulations on completing the lab!

