


JIRA Strategy Admin Workbook

Templates for the application administrator to set up, clean up, and maintain JIRA.



Is your application an organized, tidy, and trimmed garden or a foggy, contaminated, overgrown swamp?

Rachel Wright
Industry Templates, LLC

Projects

Quick Explanation

Project – A collection of issues.

Initial Attributes:

- Name
- Key
- Project Lead

Additional Attributes:

- URL
- Project Type
- Project Category
- Avatar
- Description

Benefits:

- Projects can be set up per department, per team, or to track large initiatives.

When to Use:

- Create a new project when you need different Components, Versions, Permissions or Notification strategies.

Documentation link: jirastrategy.com/link/jira-project

Strategy for Creating New Projects

Creating a repeatable and standard procedure will save you time when collecting and fulfilling project creation requests.

Does this scenario seem familiar to you?

Mary: *"Hi John, I need a new JIRA project created for the ecological initiative we'll start working on soon."*

John: *"Sure Mary, what would you like this new project to be called? Who will be the project lead? Will you need a task-based workflow or a support-based workflow? Also, when do you need this created by?"*

Mary: *"John, let's call it 'Wetlands Reconstruction'. I need it by next week."*

John: *"Thanks Mary. I still need to know who the project lead will be and what type of workflow you desire."*

The conversation continues back and forth, likely over a few hours or even days until John finally has the basic information he needs to create a new project. This is a waste of time. Instead, create a standard template for each new project request.

Wording: Sample New Project Request Procedure

JIRA is a fantastic way to track your team's work or individual assignments. And it's not just for developers! Project Managers, individual teams, and really anyone can benefit from JIRA's project and task management features.

If your team doesn't already have a JIRA project, it's easy to get one created. To request a new JIRA project, create an issue in the JIRA Support project. (Select the "Add Project" issue type.)



TIP

Consider reusing existing workflows, statuses, fields, etc. for your new project.

Once the administration team has received your request, you may receive questions or recommendations based on your request. Please note that it may take up to a week to complete your project setup, depending on its complexity.

Your project will be in "test mode" until the configuration is complete. Therefore, your team members may not be able to access the project during this period.

Only create test issues until you receive notice that the project is ready to process real issues.

Download this wording at: jirastrategy.com/link/project-wording. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.

Worksheet: New Project Request

Ask questions that will help determine if and how to complete a request or propose an alternate solution.



RECOMMENDATION

Ask what *type* of project is needed and assign the default or standard schemes built to fit those needs. Don't ask questions that lead to customizations that you aren't willing to accommodate. For example, asking the project lead to choose specific workflow steps could lead to setup that is difficult to maintain and has too many custom elements. If any customizations are needed, they should be handled separately.

Questions

Ask your users to answer the following questions as part of their new project request.

1. How many issues do you expect will be created per month?
2. What is the desired project name?
 - This is the official name of the new JIRA project. It will be seen everywhere in JIRA like the "All Projects" listing, issue breadcrumb links, and the header for project summary and admin pages. Choose something that's descriptive, unique, short, and works for the long term. The maximum character limit is 80.

3. What is the desired project key?
 - The key is a few letters used as a project's "short" name and to identify issue IDs. For example, the issue ID "PROJ-1234", the key is "PROJ." The maximum character limit is 10. The key can't be changed once the project is created.
 - 📌 Recommendation: Encourage users to choose their project key wisely or help them choose. While you can technically change a project's unique key, it's not recommended.
4. Who is the desired project lead?
 - The name of the person in charge of the project from a JIRA perspective. This person is listed as the project lead on the "All Projects" page and often the default issue assignee. This person will be responsible for certain project settings, like managing the list of users who can perform certain tasks within the project. For more information, see the "Responsibilities" documentation.
5. What is the purpose of this project?
 - For example, do you need a setup for task management, software development, support, etc.?
6. Is there an existing project whose setup that you would like to use?
 - If yes, provide an instruction like: "Set up this project like the existing "PROJECTKEY" project."
7. How long will the project be used?
 - Does the project have a known end date or archive date?
8. Are there any access restrictions?
 - Specify who should have restricted view access. List any individual issue restrictions. Provide restrictions for any actions like edit, set fix version, etc. that need to be restricted.
9. Is there any other information you would like to provide?
10. Do you have any questions?


Download this worksheet at: jirastrategy.com/link/new-project-request. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.



Turn this worksheet into a "Create" screen in your JIRA Support project!

Considerations

When you are processing new project requests, there are a few other things to consider:

- Is this new project necessary to support the use case? Or should an existing project be used to accommodate it?
- Should an existing similar project be archived?
- Does the requested project lead understand their responsibilities?  See the "Responsibilities" section.
- Are any requested customizations or new schemes warranted? If so, what impacts would they have on other projects, user groups, of the application as a whole?
- Is a future archive action needed? If yes, create an issue now to remind the team to perform it at a future date.



RECOMMENDATION

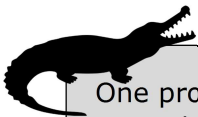
You should spend less than 10 minutes on each new project set up. Andre Lehmann, JIRA/Confluence Administrator and Leader of the Saxony, Germany Atlassian User Group, recommends using the JIRA Command Line Interface (CLI) add-on to set up new projects in under a minute.

Name Your Project

Your project name and project key are critical attributes to help you identify your project. A best practice is to make the project's title similar to the project key. Users should be able to communicate using either project name.

Your project name should be descriptive yet generic. Example: A project with the key "ACME" doesn't tell the user what kind of issues should be reported in that location.

The project key should be short in length.



EXAMPLE FROM THE SWAMP

One project had a 13 character key! While a long key is allowed in JIRA, it makes little sense to make end users type something long.



DON'T

- Use keys containing "reserved" words.
 - Example: A project with the key "JIRA" will result in strangely formatted queries.
 - Atlassian uses "JRA", "CONF", and CWD" for their JIRA, Confluence, and Crowd public bug reporting JIRA projects.
- Avoid keys containing numbers. The combination of a key with a number and sequential issue IDs are likely to be confusing.
 - Example: In the "YEAR2017" project, the first issue would be YEAR2017-1.

Project Categories

Project categories are simple groupings of similar projects. A project can only belong to one category and there's no way to create a project hierarchy. Categories are different than and in addition to a project's type.

Image: Example Categories

The screenshot shows the JIRA project management interface. On the left, under 'PROJECT TYPES', there are 'All project types' (Software and Business) and a 'CATEGORIES' section (highlighted with a red box) containing 'All categories', 'Industry Templates', 'Rachel Wright', 'x Archived', 'No category', and 'Recent projects'. The main area is titled 'All project types - All categories' and contains a search bar and a table of projects.

Project	Key	Project Type	Project Category
Assets	ASSET	Software	No category
Atlassian User Group	AUG	Business	No category
Box	BOX	Business	No category
Bucket List	BUCKET	Business	Rachel Wright
Industry Templates	INDT	Software	Industry Templates
JIRA Strategy	STRAT	Business	Industry Templates
Training	TRAIN	Software	Rachel Wright
x ARCHIVED: To Do	TODO	Business	x Archived



RECOMMENDATION

Always assign projects to a category. This will allow users to filter large groups of projects with JQL. JQL example: `category = "IT Support"`

Share Project Schemes and Assets

One of the goals of the Advisory Board and the Application Administrative team is to share schemes and assets between projects. All projects should be set up in the same way until there is a clear and justifiable need to differ from the standard. A small list of shared schemes:

- makes new project creation faster,
- makes project maintenance easier,
- makes admin page load faster,
- allows user page loads and queries to return faster, and
- requires less records in the database.

The following assets were meant to be shared by many projects:

- Issue Types and Issue Type Schemes
- Workflows and Workflow Schemes
- Screens, Screen Schemes, and Issue Type Screen Schemes
- Fields, Field Configurations, and Field Configuration Schemes
- Notification Schemes
- Permission Schemes
- Issue Security Schemes
- Roles

The following assets are project-specific:

- Users
- Components
- Versions



RECOMMENDATION

If you must create a set of schemes that are specific to a certain project (and not likely to be shared), it's helpful to use the project's unique key in the scheme's name. Example:
PROJECTKEY: Scheme Name.



DON'T

Don't create brand new schemes for every new project.

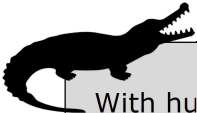
Establish Scheme Defaults

JIRA comes with pre-loaded schemes named with the keyword "default." If those schemes lose their integrity, they no longer represent good default settings, and new projects become harder to create.



RECOMMENDATION

See how far your application has strayed from the default by comparing it to the default set up reference at: jirastrategy.com/link/clean-instance. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.



EXAMPLE FROM THE SWAMP

With hundreds of schemes in every scheme category, one company had trouble selecting the correct ones for a simple new project set up. The customized defaults no longer represented what the ideal default settings should be.



RECOMMENDATION

There are a number of add-ons that can help you create a "project template." All you really need to do, however, is keep your list of scheme options to a minimum. If there are only a few choices, it won't be hard to pick the correct one for the project being built.



DON'T

Don't let the "default" schemes become no longer the ideal default.

Project Configuration Strategy

It's important to set expectations before the project is configured. You'll want to avoid users creating issues and trying to work in a project you are still configuring.



EXAMPLE FROM THE SWAMP

The second a new project appeared in the "All Project's" list, users began creating and working new issues. Since the project was still being configured and customized, the project likely didn't have the correct issue types, the desired workflow, and the requested custom fields yet. This led to premature trouble reports and questions about the project's configuration.



RECOMMENDATION

To combat users interacting with a partly configured project, hide the project until all set up items (including testing) are complete. Only let users testing the project have the "Browse" permission.

Here's a helpful checklist when creating a new JIRA project. Completing items in a specific order can make set up easier to complete. See "Project Configuration Recommendations" for additional details.



RECOMMENDATION

For large, custom projects, paste the checklist into the request ticket and "check off" items as you complete them. This will help requestors know how close they are to being able to test their new project.

Worksheet: New Project Configuration Checklist

The checklist below is designed to guide you through the project configuration process. Please do not start using your project (adding real issues) until all line items are completed. Look for a green check mark next to each line item. Also, please look for any notes or action items for you to complete. We look forward to completing the setup of your project.

Configuration Process

1. Create Project
 - Display Name: [long name]
 - Project Key: [key]
 - Project URL: [URL]
2. Set Users and roles
 - Set/Verify Project Lead
 - Set Default Assignee
3. Set Project Details
 - Link
 - Project Type
 - Project Category
 - Icon
 - Description
4. Create/Set/Verify Issue Types
5. Create/Set/Verify Permission Scheme
6. Create/Set/Verify Notification Scheme
7. Create/Set/Verify Workflow
8. Create Components
 - A temporary "Test" value is available. Project Lead: please set up your Components here: [insert link]
9. Create Versions
 - A temporary "Test" value is available. Project Lead: please set up your Versions here: [insert link]
10. Add Custom Fields

11. Verify Standard Capabilities
12. Create/Set/Verify Field Configuration
13. Create/Set/Verify Screens
14. Create Test Issues
15. Requestor Testing
16. Address Change Requests
17. Delete Test Issues
18. Unhide Project
19. Attach Configuration Screenshot
20. Remove Unused Schemes

Download this worksheet at: jirastrategy.com/link/project-checklist. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.

Project Configuration Recommendations

Consider the following recommendations as you complete each configuration step.

1. Create Project
 - Display Name
 - 📌 Recommendation: Echo the project's name back to the user. Note any modifications needed. (Example: removal of special characters, shortening the length, spelling or capitalization correction, etc.)
 - Project Key
 - 📌 Recommendation: Echo the project's unique key back to the user. Note any modifications from the original request.
 - Project URL
 - 📌 Recommendation: Paste the project's URL so the user can easily access and favorite or bookmark the new location.
2. Set Users and roles
 - 📌 Recommendation: Remove general users from the "Users" role during project build out. This will prevent users from creating issues before the project is ready to accept them. Add just a few tester names to the "Users" role so others will be able to test the project in the steps to follow. Add the "jira-administrators group" to the "Administrators" role (unless already given access through the Permission Scheme) so they can support project management efforts in the future.
 - Set/Verify Project Lead
 - 📌 Recommendation: Communicate the responsibilities of project ownership with this user. 📎 See the "Responsibilities" section for tips.
 - 📌 Recommendation: Resist the urge to list a distribution list or generic user as the lead. 📎 See the "distribution list warning" in the "Project Leads" section for more information.
 - Set Default Assignee
3. Set Project Details
 - Link
 - 📌 Recommendation: Link to a location where a user can obtain more information about the initiative or the team.
 - 📌 Recommendation: If you use Confluence, make the unique JIRA key and the Confluence key match, so both locations are easy to users to find.
 - Project Type
 - Project Category
 - Icon
 - Description

- 🛠️ Recommendation: Provide a single point of contact for the project for application administrators and end users. If you decide to set a distribution list or generic user as the Project Lead, add the name of the distribution list owner here.
- 🛠️ Recommendation: Include the ID of the issue used to create the project, so any set up or customization notes are easy to access.

Image: Project Description with Request ID

Activity [Switch view ▾](#)

This project is for: [insert purpose here]

The owner and single point of contact for this project is: [insert name and contact details]

This project was created with: SUPPORT-123

- Create/Set/Verify Issue Types
- Create/Set/Verify Permission Scheme
- Create/Set/Verify Notification Scheme
- Create/Set/Verify Workflow
- Create Components
 - 🛠️ Recommendation: If component values are not provided as part of new project creation requirements, add a temporary value of "Test" and ask the project lead to set them. This will avoid a UI error if components are required by the field configuration and visually show that components still need to be set.
- Create Versions
 - 🛠️ Recommendation: If version values are not provided as part of new project creation requirements, add a temporary value of "Test" and ask the project lead to set them. This will avoid a UI error if components are required by the field configuration and visually show that versions still need to be set.
- Add Custom Fields
- Verify Standard Capabilities (See the "Standard Capabilities" worksheet.)
- Create/Set/Verify Field Configuration
- Create/Set/Verify Screens
- Create Test Issues
 - 🛠️ Recommendation: Build administrative testing into your new project creation workflow before requestor testing begins. Create one test issue for each available workflow and/or screen/field configuration. Test both the expected workflow path and any alternate paths. Completely populate every field to show the next set of testers what data is expected and what it should look like.
 - 🛠️ Recommendation: Don't be afraid to have fun with your new project test issues! This is a good way to get your requestor interested in the project configuration process. Create very silly sample issues, which are more likely to elicit response over the bland summary title "Test Issue #1." 📎 See the "Character Users" section for ideas.
- Requestor Testing
 - 🛠️ Recommendation: Build user acceptance testing into your new project creation workflow. This testing is done by the requestor and any project or team leads before the project becomes available to all end users. You may need to provide instructions to help the testers understand what to test and what to look for. Have them create, edit, and transition test issues.
- Address Change Requests
- Delete Test Issues

- 🌱 Recommendation: Delete test issues in bulk and suppress email notifications during the deletion process. Alternately, close issues and make it clear they were created for test purposes.
18. Unhide Project
- 🌱 Recommendation: Make the project visible by adding general users to the Users role. (NOTE: This may only include the "internal" user's group. [🔗](#) See the "External Users" section for details.)
19. Attach Configuration Screenshot
- 🌱 Recommendation: Take a screenshot of the project's administrative "Summary" page so you'll have a record of the initial set up. Attach the screenshot to the new project request issue.
20. Remove Unused Schemes
- 🌱 Recommendation: JIRA may have auto created new schemes as part of new project creation. Remove them if the project will use shared schemes instead.

Wording: Sample New Project is Ready Message

@username - your new JIRA project has been configured, tested, and is ready for use! Please add the names of any users not already listed on your "Users and roles" page here: [URL].

For more information about your project's settings and the role of the project-level administrator, please see: [URL]. [🔗](#) See the "Responsibilities" section.

Finally, if all is well, please click the "Pass" transition button, to close this request.

Download this wording at: jirastrategy.com/link/project-wording. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.

Test New Project Configuration



RECOMMENDATION

Make sure the requestor has tested the project before making it available for general use. Have them create a new issue (or one issue per Issue Type) and transition it fully through the workflow. This may help uncover missed needs or requirements. It's much easier to make configuration changes *before* the project is filled with real issue records. You'll also avoid needing to retrain users or migrate entered data.



EXAMPLE FROM THE SWAMP

A new project to hold sensitive company data was requested. Everything was configured except for the permissions. The requested restriction requirements were never provided however. Regardless, people started using the project. The sensitive information was available for any JIRA user to view.



DO

As noted in the previous recommendation, hiding a project from JIRA users until all requirements are received and implemented is useful.

Configure Your Project

Periodically check in with your project leads to make sure the current configuration is meeting their needs. Also provide documentation and resources to support additional requests and customization needs.



DON'T

Don't set up a project and assume the lead knows how to properly maintain it.



RECOMMENDATION

Give the project lead simple instructions for configuring, maintaining, and requesting further customizations. (See the documentation templates below.)

Wording: Configure Your New Project

Your new project is ready! Now what? As the project lead you can add or modify the project to best suit your team's needs. This document covers how to configure and maintain your project in JIRA.

Administration Area

Each project has its own administrative area. A project-level administrator can manage all settings for your project. They cannot however manage settings for other projects or make application or system-level changes. As a project-level admin, you can't break anything, so feel free to experiment with the project settings so they fit your needs. The only thing we can't "undo" in JIRA is deletion of issues.

[Insert instructions for accessing the admin area in your version of JIRA.]



NOTE

If you do not see this link, it's possible that you were not added to the "Administrator" line in the "Users and roles area" for this project. Please contact the JIRA Support team for assistance.

Your first priority is setting up your Versions, Components, and Users and roles.

Versions

Versions are points-in-time for a project. They help you schedule and organize your releases. You can also use Versions as another distinguisher for something you want to track. Skip this if your team has no work to release.

If you are part of a development team, use the "Releases" page to add your "Versions" using

unique software version numbers. The "Version" column is what will be queried against, so make the naming format short and easy. (Example: A simple name like "1.0" is easier to query than "Version 1.0 Special Release").

Image: Sample Versions

Version	Status	Progress	Start date	Release date	Description
1.2	UNRELEASED	No issues	02/01/17	02/28/17	
1.15	UNRELEASED	No issues	01/19/17	01/20/17	Emergency Release
1.1	UNRELEASED	No issues	01/01/17	01/31/17	
backlog	UNRELEASED	No issues			



NOTE

Only projects of the type "Software" and users in the "jira-software-users" group will see the Versions feature.

Read more: jirastrategy.com/link/managing-versions

Components

Components represent a grouping of issues in a project. This feature allows you to query all items in your project associated with that component. You can use this in a number of ways. Here are some examples:

- **Example 1:** Your project is to build a car. You might want to group issues in your "car project" by parts of the car that need work.
 - Sample Components: *Tires, Engine, Windows, Headlights, etc.*
- **Example 2:** You have a cross-functional team where many different types of jobs are performed. You might want to group issues by type of job or department.
 - Sample Components: *Marketing, Design, Support, Legal, etc.*
- **Example 3:** Your project is for writing copy for the entire company. You might want to group issues by where the copy will be posted.
 - Sample Components: *Website, Social Media, Employee Handbook, Product Manual, etc.*

The Component is the most powerful feature in a project. Not only can the project-level administrator maintain the selection list, but this field has auto-assignment capabilities. Example: If Component "X" is selected, assign Issue to user "Y." Using the "build a car" example above, you could use this to automatically assign any "Tires" request issues straight to Mary, and any "Engine" related issues to John, etc.

Set up auto-assignment by entering a username in the "Component Lead" field. Then, set the "Default Assignee" field to the value "Component Lead." Alternatively, you can assign all issues to one specific person for triage or assign all issues no one (using the "Unassigned" option) for later assignment.



NOTE

The auto-assignment ability only happens when a Component is selected as part of an issue "create" action. Later, if a component is selected or changed, as part of an "edit" action, no auto-assignment change will occur.



DO

Make the component names short. Really long and multiple word values make issues harder to search for.



DON'T

The only thing you should NOT use Components for is to duplicate an existing JIRA field or function. (Example: Don't use this to group issues by a person's name. JIRA uses the "Assignee" field to specify who needs to complete an issue.)

If you're not sure how you might want to use the Components feature, skip it and come back to it later. After you've used the project for a while, issue patterns should become clear.

Read more: jirastrategy.com/link/managing-components

Users and Roles

The "Users and roles" area gives specific users, or established groups of users, access to perform certain actions within the project. The options to update the Project Lead and the Default Assignee for all (not otherwise assigned) issues are at the top of the page.

There are a number of pre-established roles. For now, let's focus on the "Administrators" and "Users."

Administrator

Any listed user or group with project-level admin rights can modify all settings for the project. They will see what you are seeing right now. In general, this list should have as few names in it as possible.

User

A user is someone involved with any aspect of a project. This is someone who creates issues, requests work, completes work, reviews work, etc. Add the names of your individual team members here.

Read more: jirastrategy.com/link/role-memberships

Conclusion

If you haven't already, create some sample issues so you can see how the project works. Also, try transitioning an issue all the way from creation to completion to see how it behaves in its life cycle. If the issue screens have the fields you need and the workflow functions as you need it to, then you are done with your configuration! If not, you'll need to request further customizations from the JIRA Support team.

Download this wording at: jirastrategy.com/link/project-wording. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.

Wording: Request Customization for Your New Project

Your JIRA project has been created and configured, but what if you need additional customization? The following areas of your project can be further modified:

Fields

Are there fields on your *create*, *edit*, or *view* screens that will never be used? If so, ask for these to be removed for your project.

Example: *"Please remove the [fieldname] field from [projectname]'s" create, edit, and view screens."*

Is an expected field missing? Does a new custom field need to be created?

For any new field requests, provide the following information:

1. What is the desired field's name? (label)
2. What will the field be used for?
3. What field description should be shown to the user?
4. What is the field's type?
 - What kind of field is needed? (What type of data will be collected?) Example field types: text, number, date, checkbox, select list, URL, etc.
5. What field properties are needed?
 - For a text field, will a single line or multiple lines of text be collected? For a checkbox or select list, what are the individual selection values? (Example: choice 1, choice 2, etc.)
6. What screens should the field be shown on? (Create, edit, view, or all?)
7. What are the validation rules? (if any)

Screens

Each issue type has three standard screens: create, edit, and view. In addition, you may have transition screens that display as part of the workflow.

On these screens, you can customize the following:


- The fields displayed.
- The order of the fields.
- The presence of tabs. (If you have many fields, use tabs to break them up into logical groups.)

Image: "Internal Details" Tab Example

Create issue

⚙️ Configure Fields

Project*

 Demo (DEMO)

▼

Issue Type*

☒ Task

▼

?


Some issue types are unavailable due to incompatible field configuration and/or workflow associations.

Request Details

Internal Details

Summary*

Priority

 Minor


▼

?

Component/s

None


Assignee

 Automatic

▼

[Assign to me](#)

Reporter*

 Rachel Wright

Start typing to get a list of possible matches.

Description

Style ▼

B


I


U


A


▼


↻A ▼

 ▼

 ▼





 ▼

+

⋮

☐ Create another

Create

Cancel



NOTE

It is possible to hide a field on your "create" screen, but show it on your "edit" and "view" screens. This is helpful for projects with complex data needs. It's best for your end users to only ask for the information you absolutely need up front and collect additional details later. This certainly means more screens to maintain, so only use this technique only when truly warranted.

Workflows

There are two parts of a workflow to know about: statuses and transitions. A status is a descriptor of an issue's current state. (Example: "In Progress" or "Closed.") A transition is forward or backward movement between statuses. Between each status are a number of transition buttons to facilitate movement. A transition button can also trigger a screen to collect additional information, update existing information, or provide user instructions.

Use the provided "Custom JIRA Workflow Template" to craft your custom workflow. Draw a workflow on paper first to ensure it makes logical sense and all forward and back transitions are accounted for. After drawing the workflow, write the workflow out in words. This can uncover

additional needs you may have neglected to draw or consider. Note any steps that require restrictions or special conditions.

Download this worksheet at: jirastrategy.com/link/workflow-documentation. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.

🔗 See also: "Responsibilities" in the "Project Leads" section.

Notifications

Your project can send email notifications for various standard and custom events. Notifications can be sent to individuals, project roles, or (externally managed) corporate distribution lists.

Example notification rules:

- When an issue is updated, email the Reporter and Assignee.
- When an issue is reassigned, email the original Assignee and the new Assignee.
- If transition button X is clicked, email group/user Y.
- If action X occurs, email all the 'Watchers'.

This feature should be used sparingly, and only for most important events. If you send the entire team an email for each and every content and status update, they are likely to filter their JIRA email and miss the important messages. Email notifications should be considered a passive and secondary monitoring method. (The primary method is for users to proactively monitor issues by regularly logging into JIRA and utilizing filters and dashboards.)

Permissions

Your project has its own set of permissions that can be further customized. This includes abilities to perform certain standard actions like: "Create Issue", "Assign Issue", "Manage Watchers", etc. Standard actions can be restricted to project roles, groups, and individuals. Restrictions can quickly become a maintenance nightmare however. It's best to only add restrictions when absolutely necessary.

Questions? Please contact the JIRA Support team.

Download this wording at: jirastrategy.com/link/project-wording. Use the code in the "Worksheets, Templates & Companion Materials" section to download it for free.

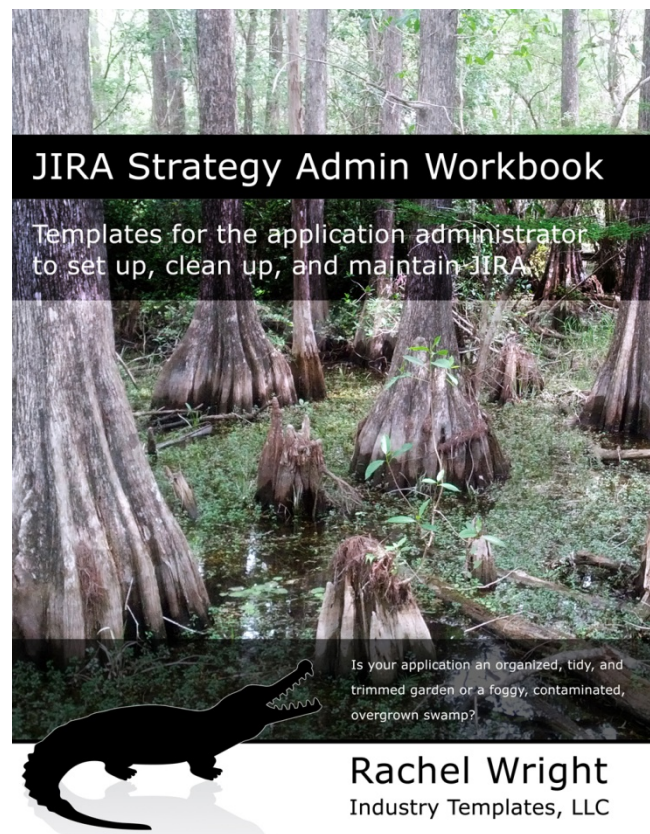
About the Workbook

Are you a **part-time Administrator** who helps out with JIRA in addition to your “official” role? A **full-time Administrator** for JIRA or the Atlassian product suite? A **Project Manager, Business Analyst**, or other team member, who relies on JIRA? A **Systems or Database Administrator** who supports company tools?

The **JIRA Strategy Admin Workbook** will save you time, money and frustration. This book is different – **it's not documentation**. It's recommendations from years of cleaning up horrible JIRA configurations!

This workbook shows you:

- actions for a well-planned implementation,
- simple ways to streamline administration,
- how to audit and clean up the application,
- ways to maintain and extend JIRA,
- how to create repeatable procedures, and
- how to stay out of the “**JIRA swamp**”.



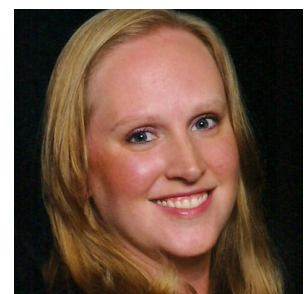
About the Author

Rachel Wright is an entrepreneur, process engineer, and Atlassian Certified JIRA Administrator.

She started using JIRA in 2011, became a JIRA administrator in 2013, and was certified in 2016.

She is the owner and founder of Industry Templates, LLC, which helps companies grow, get organized, and develop their processes.

Rachel also uses Atlassian tools in her personal life for accomplishing goals and tracking tasks. Her first book, the “JIRA Strategy Admin Workbook”, was written in Confluence and progress was tracked in JIRA!



Get your copy of the **JIRA Strategy Admin Workbook** at jirastrategy.com.