Writing Agile User Story and acceptance test requirements

Lab Guide

Table of Contents

Lab 1: Jira Overview (Server)

Estimated time: 45 minutes

In this lab, you will:

I. Create a kanban project.
II. Create issues.

Login

First, Log in to Jira:

- Username: admin
- Password: Charlie!

> *Congratulations, you have logged in to Jira*

I: Create a kanban project.

1. Using the information provided to you at the beginning of the course, navigate to your Jira Software Server environment and log in as admin with a password of Charlie!

   > *Note: The admin user is a Jira administrator. Jira administrators can create projects.*

2. Select the Projects dropdown, then View All Projects.

3. Click the Create project button on the upper right.

4. On the Create project screen, select the Kanban software development template. Click Next.

   > *A template is the initial configuration of your project. Each template provides different default behavior and tools to the project. We are selecting the kanban template. Among other things, this means that a project board will be included with the project. We will discuss project boards more later.*

5. You should see a screen showing the issue types and workflow used for the project. We will discuss these later in the course. Click Select.

6. Under Project Lead, enter Alana Grant.

   > *A project lead can be the default assignee for issues when they are created. The project lead can also be specified in permissions and notifications.*

7. Name the project projectA. A project key (such as PROJ) will automatically be created for you.

   > *A project key is a unique identifier for the project. Each issue in the project will have a unique issue key that begins with the project key. An issue key uniquely identifies an issue.*

8. Click Submit.

9. Verify that your project was created and you are viewing the kanban board.

10. Add Alana Grant to the Administrators role for the project:

    - While viewing your kanban board, select the gear icon (project settings) in the lower left. You can click on the » icon in the lower left to expand the sidebar.
    - Select Users and roles.
    - In the upper right, select Add users to a role.
    - Under Users or groups, enter "Alana Grant".
    - Under Role, select Administrators.
    - Click Add. You should now see Alana Grant listed under the Administrators role for the project.

11. Verify that Alana Grant is the Project Lead for this project. Under Users and roles, the Project Lead should say Alana Grant. If not, click Edit defaults in the upper right and for Project Lead, select Alana Grant.

12. Verify that Alana Grant is a board administrator for the Kanban board:

> *When you created the project, you should have selected Alana Grant as the project lead. This also makes Alana Grant a board administrator for the kanban board. If you did not originally set Alana Grant as the project lead, you need to explicitly add her as a board administrator.*

- Click on Kanban board in the sidebar.
- Click on the Board dropdown in the top right and select Configure.
- Under General > Administrators, you should see Alana Grant. If not, select the existing user, click the pencil icon and add Alana Grant.

> *Congratulations, you have created a kanban project. You have logged in as the "admin" user (a Jira administrator), created a project and assigned Alana Grant to the Administrators role. Now you will be able to log in as Alana Grant and configure the project.*

II: Create issues.

> *The planned work of a project is broken down into issues. Issues are also known as work items, stories and more. We will start with three simple issues, named "add item 1", "add item 2" and "add item 3".*

1. Click the user avatar in the upper right and select Log Out.

2. Click Log in again and log in as agrant with a password of Charlie! You should see Alana's picture in the upper right.

3. Navigate to your projectA project by selecting Projects > View All Projects.

4. View your projectA kanban board. If your sidebar on the left is collapsed, you can click the >> in the lower left to expand the sidebar. Click the Kanban board tab to view the board.

5. Create an issue named "add item 1":

   - Click the Create button to create an issue. The Create Issue screen appears.
   - Under Issue Type, select Task.
   - Under Summary, enter add item 1.
   - Leave the other values as their default.
   - Since you are creating more issues after this one, you have the option to click the Create another checkbox to expedite issue creation.
   - Click Create.

6. Create an issue named add item 2.

7. Create an issue named add item 3.

8. You should see your three issues in the BACKLOG column of your kanban board.

> *Congratulations, you have created three issues and completed this lab.*

Lab 2: Scrum Projects (Server)

Estimated time: 30 minutes

In this lab, you will:

I. Create a scrum project.
II. Create issues in the product backlog.
III. Create and plan a sprint.
IV. Execute a sprint.
V. Complete a sprint.

> *This lab creates a scrum project using the Server version of Jira Software. These instructions DO NOT APPLY to Cloud projects.*

I: Create a scrum project.

1. Log in to Jira as a Jira Administrator (admin/Charlie!). A Jira Administrator user is needed to create projects.

2. Select Projects > Create project.

3. Under Software, select Scrum software development and click Next.

4. In the Scrum software development dialog, click Select.

5. For the Project Lead, enter Alana Grant.

6. For the Name, enter projectB.

7. Make sure that the project Key is unique (for example "PROJB").

8. Click Submit. You should be brought to the Backlog tab of your new project.

9. Click Project settings > Users and roles. Click Add users to a role and add Alana Grant to the Administrators role. This means that Alana Grant is a Jira project administrator for the project.

   > *Congratulations, you have created a scrum project.*

II: Create issues in the product backlog.

1. Log in to Jira as Alana Grant (agrant/Charlie!).

2. Click Projects > View All Projects and select the projectB project.

3. Click the Backlog tab to view the backlog. It should be empty.

4. Create three issues of type Story (the green icon) in the backlog with summaries of add item 1, add item 2 and add item 3.

   > *Congratulations, you have created a backlog with three issues.*

III: Create and plan a sprint.

> *A sprint is a period of time where you complete a certain number of issues.*

1. Click Create Sprint to the right of the Backlog section. You should see an empty "Sprint 1" along with the Backlog section.

> *The start of the sprint includes a sprint planning meeting. In this meeting, the sprint team usually decides on the sprint goal, estimates the amount of work of issues and decides which issues to complete during the sprint. The development team decides how to accomplish the work of the sprint. All projects and sprint planning meetings are unique.*

2. Add estimates as story points to the issues. We will arbitrarily say that add item 1 is 1 point, add item 2 is 2 points and add item 3 is 4 points.

   - Click on each issue in the backlog and add its estimate next to the Estimate field. After entering an estimate, you should see the estimate in gray next to each issue in the backlog.

> *The development team usually is responsible for estimating story points. Story points are relative units, usually indicating the effort involved in completing the issue.*

3. Prioritize the backlog. We will arbitrarily give the 2 point story (add item 2) the highest priority and the 4 point story (add item 3) the lowest priority.

   - Drag and drop the stories in to their correct order in the backlog. (With add item 2 at the top.)

> *The product owner is usually responsible for prioritizing stories in the backlog.*

4. Add stories to the sprint backlog. We will arbitrarily assume that the team can execute up to four story points per sprint. This is known as the team's *velocity*.

> *The product backlog contains all of the backlog items for the project. A sprint backlog is a subset of the product backlog that contains the backlog items for a single sprint.*

   - Drag the add item 2 and add item 1 stories to the sprint backlog.

5. Notice that the team has estimated that its velocity for this sprint will be 3 story points.

> *The development team is usually responsible for deciding how many of the top issues to move to the sprint backlog.*

6. Add subtasks to the add item 1 story in the sprint backlog. Do this by opening the story, scrolling down and selecting the Create Sub-Task button, and adding subtasks named add item 1a and add item 1b.

> *During the sprint planning meeting, the team often breaks the work of a story down into subtasks. Each of the subtasks might contain a different type of work, such as user experience design or data storage work.*

7. Click the More icon (...) associated with Sprint 1 and select Add sprint goal. Add a sprint goal of Create the first product increment.. You should see the sprint goal in the heading of Sprint 1.

> *The scrum team agrees to the sprint goal during the sprint planning meeting.*

> *Congratulations, you have created and planned a sprint.*

IV: Execute a sprint.

1. Click the Start Sprint button associated with the sprint backlog for Sprint 1. Change the duration of the sprint to 1 week. Leave the other values as their defaults. Click Start. The Active sprints tab is displayed, showing your sprint board. Notice the sprint goal under the sprint name. Notice that you have two stories and two subtasks in the TO DO column. Notice that the other columns are IN PROGRESS and DONE.

> *A sprint board is a project board that only shows the issues of the sprint.*

2. Open an issue on the board and click View Workflow. You are brought to the Issues tab. Next to Status, click View Workflow. Notice that there are three statuses in the workflow, TO DO, IN PROGRESS and DONE. These are the default statuses in the workflow when you choose the "scrum software development" template while creating a project. Notice that there is no BACKLOG status. Click Close.

3. Select the Backlog tab. View the status of the add item 3 issue that is in the Backlog section. Notice that its status is TO DO, the same status as the issues in the first column of the sprint board. The items in the backlog section are there because they have not been added to any sprints. The status of each issue is independent of whether it is on the sprint board or in the backlog section.

4. Click the Reports tab. View the burndown chart for this sprint. Jira has added guidelines for story point completion during the sprint. The starting value is the total number of story points that you added to the sprint backlog. The ending value is zero. For the duration of the sprint, a linear decrease in the number remaining story points is assumed, excluding days off. Under the chart, you can see the issues of the sprint.

   > *Reports like this are a great way to quickly view the current status of the sprint.*

5. Navigate back to your sprint board (under Active sprints). Let's assume that Alana Grant a member of the development team and that she will work on the add item 2 issue. If Alana Grant is not the assignee of all issues on the sprint board, assign her to the issues.

6. Drag the add item 2 issue to the IN PROGRESS column.

7. Let's assume that Alana has finished the add item 2 issue. Drag it to the DONE column.

8. Repeat the process above and complete the two tasks of the add item 1 issue. You should be prompted to update the parent issue. Click Update.

   > *Congratulations, you have executed a sprint.*

V: Complete a sprint.

1. Now that the issues of the sprint are complete, you can end the sprint. In the upper right above the sprint board, click Complete Sprint and then Complete.

   > *You usually only complete a sprint at the end of the planned sprint duration. We are ending it early just for learning purposes.*

2. You should be brought to the sprint report. This includes the burndown chart. You estimated and completed three story points in this sprint, so your velocity for sprint 1 was three story points.

   > *A team's early estimations tend to be quite unreliable. As more sprints are executed, the team should become better at estimating velocity based on the historic performance of the team.*

3. At this point, you would usually have a sprint review meeting to show the product increment to the scrum team and optionally to its stakeholders.

4. After the sprint review meeting is a meeting called the sprint retrospective. This is a meeting for the scrum team to discuss how the team can execute better next time.

   > *Congratulations, you have completed a sprint and completed this lab.*

Lab 3: Writing Good User Stories and Acceptance Criteria in JIRA

Writing a useful user story can make a significant difference in your delivery timeline in the long run.

What is a user story?

In an agile delivery framework, a user story is the smallest unit of work to be delivered. The end deliverable should beable to demonstrate value to the user in the form of addressing the needs of the user specified in the user story; it's essentially a user requirement that's focused on the user's experience.

What makes a good user story? The way you write the user story would determine the outcome of the deliverable. Hence, it's important to include the essential components in the form of specifications to the delivery team that will be building the solution based on the story.

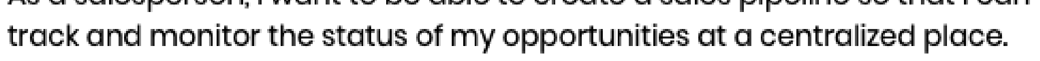
Often, the user story is communicated to the agile team members via a software delivery tool e.g. JIRA. In myexperience, the clearer the communication of the story, the less back and forth is required among the variousstakeholders — which leads to efficient delivery.

A good user story would generally contain the following

- five components: User-centricity
- Background context
- Screen flows
- Technical
- specifications Clear
  acceptance criteria

#1: User-centricity

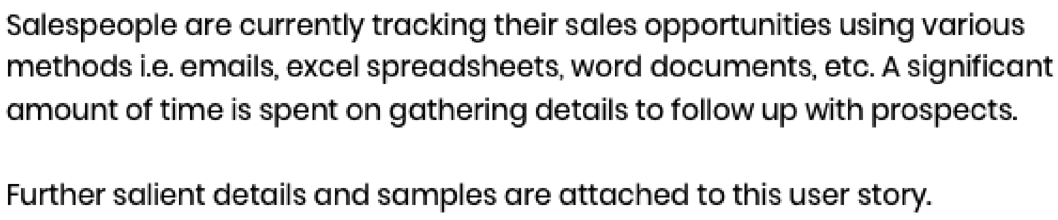
The user story must provide some form of value to a specific user/group. By writing the story in the following format: "As a user, I want to be able to… so that I can…", would help to ensure that the deliverables are tailored to the userand enable them to achieve their goals.



As a salesperson, I want to be able to create a sales pipeline so that I can track and monitor the status of my opportunities at a centralized place.

#2: Background and context

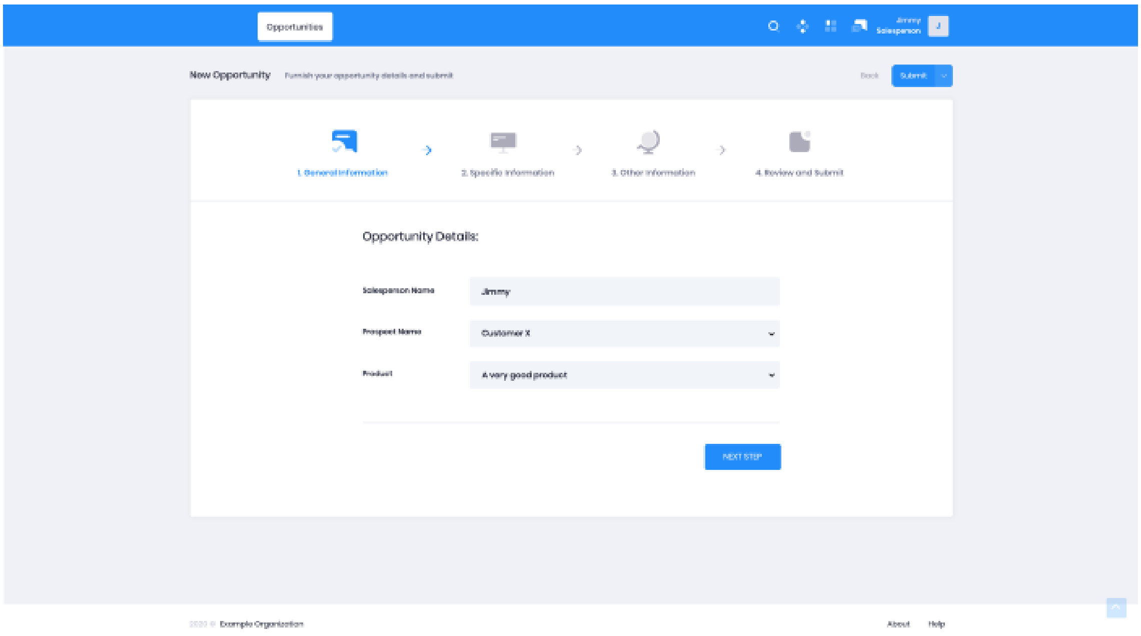
A brief background and context can help team members to understand the rationale of the requirement, which allows them to exercise discretionary decision making in times of uncertainty --- reducing unnecessary clarificationswith the stakeholders.



> Salespeople are currently tracking their sales opportunities using various methods i.e. emails, excel spreadsheets, word documents, etc. A significant amount of time is spent on gathering details to follow up with prospects.
>
> Further salient details and samples are attached to this user story.

#3: Screen flows

I have found the sketches and mock screens the best way to align requirements among the various stakeholdersfrom business users, designers, architects, and engineers --- to ensure that everyone is on the same page and looking at the same picture.



New Opportunity Furnish your opportunity details and submit

Opportunity Details:

Salesperson Name: Jimmy

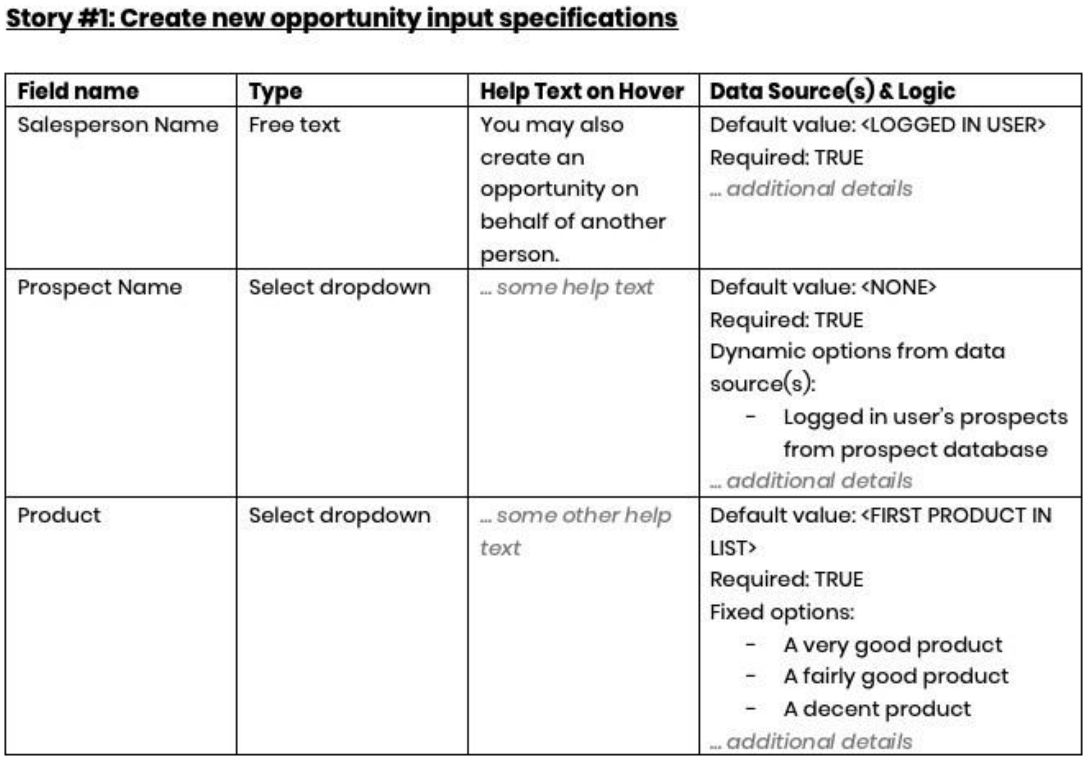
Prospect Name: Customer X

Product: A very good product

The sketch need not be fully complete but should include the essential details i.e. input fields and layouts to facilitatetechnical solution design and documentation.

#4: Technical specifications

Similar to screen flows, technical diagrams and specifications provide team members with clarity on the deliverables.



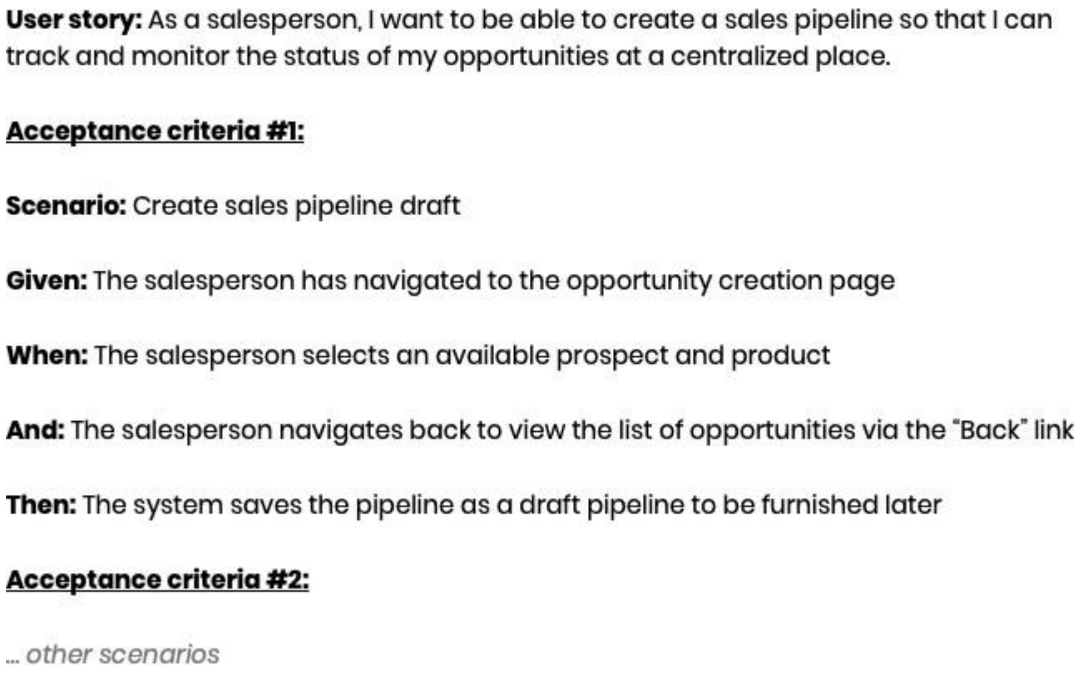
Story #1: Create new opportunity input specifications

Field name	Type	Help Text on Hover	Data Source(s) & Logic
Salesperson Name	Free text	You may also create an opportunity on behalf of another person.	Default value: <LOGGED IN USER> Required: TRUE *… additional details*
Prospect Name	Select dropdown	*… some help text*	Default value: <NONE> Required: TRUE Dynamic options from data source(s): - Logged in user's prospects from prospect database *… additional details*
Product	Select dropdown	*… some other help text*	Default value: <FIRST PRODUCT IN LIST> Required: TRUE Fixed options: - A very good product - A fairly good product - A decent product *… additional details*

There are many variations of technical specifications and documentation depending on the nature of your role andproject.

#5: Clear acceptance criteria

The acceptance criteria set the expectations of the deliverables. At the end of the sprint, the solution will be evaluated against these criteria to ensure that the stakeholders are satisfied. You may also refer to the severalvariations and types for acceptance criteria for inspiration.



User story: As a salesperson, I want to be able to create a sales pipeline so that I can track and monitor the status of my opportunities at a centralized place.

<u>Acceptance criteria #1:</u>

Scenario: Create sales pipeline draft

Given: The salesperson has navigated to the opportunity creation page

When: The salesperson selects an available prospect and product

And: The salesperson navigates back to view the list of opportunities via the “Back” link

Then: The system saves the pipeline as a draft pipeline to be furnished later

<u>Acceptance criteria #2:</u>

... other scenarios

How can story fail?

Vague user stories whereby the contents of the user story only contain the user-centric story itself e.g. "As a user, Iwant to be able to…"

This will result in some of the following:

- Developers making their own interpretations on the deliverables — which don’t usually go too well with thebusiness users
- Design inconsistencies e.g. different colour variations of "success" and "cancel" buttons
- Testers being unclear on the requirements and success criteria of the user story — resulted in an increase inbugs and issues subsequently

The above has led to delays in the delivery timeline and down prioritising of important backlogs, which ultimatelycan lead to project failure.

Takeaways

The key takeaway is to provide as much clarity as possible in the user story for team members to act and deliver thework item.