

# JIRA

## Lesson 04: Hands-on JIRA



# Lesson Objectives

In this lesson, you will learn about:

- Register and Login to JIRA
- Creating a Project
- Creating an Issue – story, task, sub-task, bug
- Managing Issues – editing, deleting
- Creating & Starting Sprints
- Workflow Progress
- JIRA Reports





- When you login , you get a snapshot about description of your existing project – dashboard page (or) you can even create a new project.
- The dashboard page displays the description of the project you belong to : issue summary and the activity stream (the issues that are assigned to you, the issues that you created etc).



- Project is a collection of issues
- Project has following attributes:
  1. Name: given by the administrator
  2. Key: It is an identifier that all the issue names under the project are going to start with. This value is set during the creation of a project and cannot be modified later even by an administrator
  3. Components: subgroups within a project based on common grounds
  4. Version: For a project different versions can be tracked



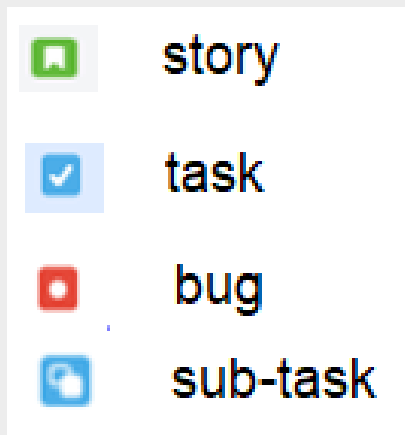
Demo 1: Register & Login to JIRA

Demo 2: Create your own site & Project in JIRA





- Edit: Used to modify issue details.
- Summary: The issue name and short description.
- Issue Type : displays all the types of issues that can be created and tracked via JIRA – Epic, story, task, sub-task, bug.





## Managing Issues (Continue...)

- Epic and story are more relevant to the agile projects.
- An Epic is a group of stories.
- A Story or user story is a software system requirement that is expressed in a few short sentences, ideally using non-technical language
  - Non-technical language, Story expresses user needs using the syntax below
  - User Story Syntax (not mandatory but preferred) :

<pre>As a &lt;user&gt; = who I want to &lt;be able to do ABC&gt; = what So that &lt;XYZ can be done&gt; = why</pre>
---

- A Task represents a technical activity
- A sub-task is sub-set of a parent issue.



# Examples of Epic & Stories

Project Name: Product Order System

Issue Type	Summary
<b>Epic 1</b>	As a user, I should be able to back up my entire hard drive.
<b>Story 1</b>	As a power user, I can specify files or folders to backup based on file size, date created and date modified.
<b>Story 2</b>	As a power user, I can indicate folders not to backup so that my backup drive isn't filled up with things I don't need saved.





## Examples of Epic & Stories (contd..)

Project Name: Product Order System

Issue Type	Summary
<b>Epic 2</b>	Develop this software any way you'd like as long as you achieve 100,000 concurrent users.
<b>Story 1</b>	As a customer, I want to access your application on all versions of Windows from Windows 95 on.
<b>Story 2</b>	As the CTO, I want the system to use our existing orders database rather than create a new one, so that we don't have one more database to maintain.
<b>Story 3</b>	As the Customer, I want the site to be available 99.999 percent of the time I try to access it, so that I don't get frustrated and find another site to use.



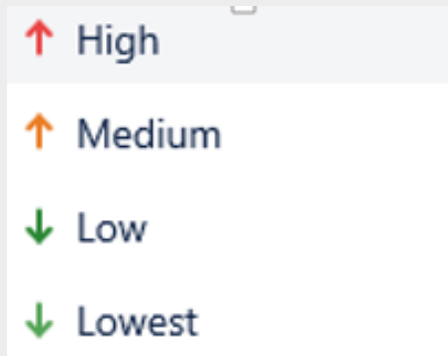
## Managing Issues (Continue...)

- Component : displays the components of the project. Choose appropriately
- Description: This is an optional text field that aids you to enter as much information as you would like about your issue. In case of a bug, it is typical to use this field to give in a detailed information about the steps to reproduce the defect. It is of utmost importance to give all the information.
- Attachment: Any supporting document can be uploaded with an issue.
- Assignee: You can type the name of the person to whom this issue should be handed over further. You can also assign an issue to yourself.



## Managing Issues (Continue...)

- Priority : The priorities and their symbolic representations in JIRA are as below :



- Linked Issues : allows you to logically link issues with one another and establish relationships/dependencies.
- Labels : You can categorize issues using labels. For example, all the issues raised to track peer review comments can be labelled "peer review" to view and track them easily.



- A sub-task is nothing but a division of a parent issue into chunks of work that can be assigned and tracked individually.
- You cannot create a sub-Task from the 'Create Issue' button
- Example : consider the task of Test documentation. Test documentation itself is an activity that might take a week to finish. Say, it involves : Test plan documentation which takes 2 days; Test case documentation – 2 days; Test plan review – ½ day and Test case review – 1 day. Also, assume that there are 2 resources.
- In this case, we can create a JIRA issue type task for “Test documentation” and have the following four subtasks under it:
  - Test plan documentation – assigned to resource 1
  - Test case documentation – assigned to resource 2
  - Test plan review- assigned to resource 2
  - Test case review- assigned to resource 1



# Examples of Tasks & Sub-tasks

Project Name: Product Order System

Issue Type	Summary
<b>Epic 1</b>	Browse Product Catalog
<b>Story 1</b>	As a customer, I want to filter the product catalog by product attributes such as price, range, brand, colour, size etc. so that I can make a correct purchase decision.
<b>Task 1</b>	Evaluate 3rd party library XXX
<b>Task 2</b>	Develop high level Interaction for browsing product catalog



## Examples of Tasks & Sub-tasks (contd..)

Project Name: Product Order System

Issue Type	Summary
<b>Epic 2</b>	As a customer, I should be able to maintain wishlist
<b>Task 1</b>	As a customer, I want to save a product in my wishlist so that I can view it again later
<b>Sub-task 1</b>	Put 'Add to wishlist' button on each product page.
<b>Sub-task 2</b>	create a new db to store wishlist items.

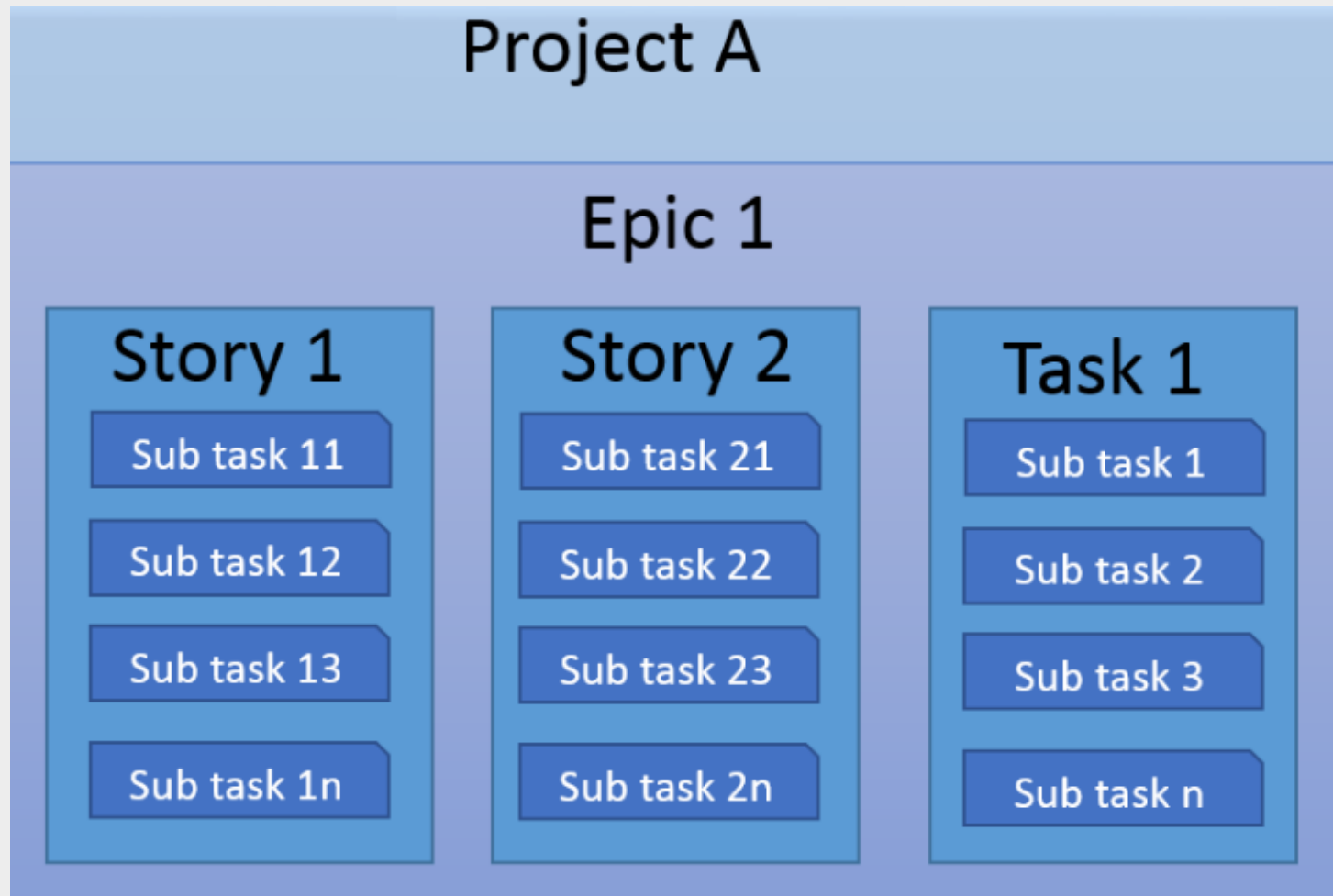


## Examples of Tasks & Sub-tasks (contd..)

Project Name: Product Order System

Issue Type	Summary
<b>Epic 2</b>	As a customer, I should be able to maintain wishlist
<b>Story 1</b>	As a customer, I want to view my wishlist so that I can buy items from it.
<b>Sub-task 1</b>	Create a page to display user's wishlist
<b>Sub-task 2</b>	Add 'view wishlist' link to home page

# Issue Hierarchy in JIRA







# Examples of Bugs

Project Name: Product Order System

Issue Type	Summary
Bug 1	It is found that the application is not accessible on firefox browser i.e. below version 40.
Bug 2	It is found that the product cannot be searched using the keyboard.
Bug 3	It is found that there is no 'view wishlist' link on the homepage of the application
Bug 4	It is found that very often the performance of the application slows down between 10 pm to 2 am
Bug 5	It is found that the button 'Add to wishlist' is mistyped as 'create wishlist'



## 1.4: Introduction to GO Demo

### Demo 3: Managing an Issue in JIRA

- 3.1 Create Issue
- 3.2 Delete an Issue
- 3.3 Create a sub-task





## JIRA Agile Project

### Agile Projects:

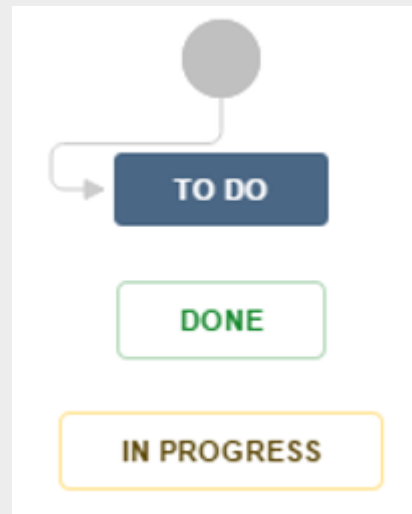
- Requirements are created and grouped together in Product Backlog
- Product Backlog is prioritized according to business importance
- Product Backlog items are assigned to **Sprints** which are 2-4 week duration
- JIRA supports all above activities followed by Agile Methodology



## 2.1: Workflow to JIRA

### Sprints in JIRA

- JIRA Issues can move through different states of workflow in a Sprint
- On creating issue, its status is always "To Do"
- It can go through "In Progress" and "Done"






## 2.1: Workflow to JIRA

# Sprint Attributes

### Start sprint

 Issue SAM-2 does not have a value for the 'Estimate' field. Values entered after the start of the sprint will be treated as scope change.

2 issues will be included in this sprint.

Sprint name: \*

SAM Sprint 2

Duration: \*

Custom



Start date: \*

10/Apr/18 12:58 PM



End date: \*

11/Apr/18 12:58 PM



Sprint goal:

login validated

Start

Cancel



# Demo

## Demo 4: Managing Sprints in JIRA

4.1 Create Sprint

4.2 Start Sprint





## Administrative tasks

- Administration -> Issues -> Add Issues
- Here we can create custom Issues
  
- Administration -> Issues -> Workflow
- Here we can create our own workflow
  
- Administration -> Projects -> Components
- Here we can create new project components



## 2.5: JIRA Reports

### Reports

- JIRA comes with a robust reporting feature. There are many kinds of reports that it facilitates :
  - Agile Reports - Burndown Chart, Sprint Report, Version Report, Epic report.
  - Issue Analysis Reports – Created vs. Resolved Issue Report, Recently Created Issue Report.
  - Forecast & Management – Time Tracking report, User workload report, Version workload report.
- Browse through a project whose details you would want to get a report of and go to the “Reports” tab. Here you will see a list of all the reports that are present. Choose the report and set the data criteria and you will have a custom made the report in hand within no time.





Lab

Issues Lab



# Summary



- Creating and Managing Issues
- Creating Sprints and tracking the Workflow progress – move issues through different status





## Review Question

Question 1 : Which of the following is not an administrative task?

- Option 1: Creating Version
- Option 2: Creating Component
- Option 3: Create Issue type



Question 2: Velocity chart shows the amount of work left to be done.

- True/False

Question 3: Fill in the Blanks:

The sequence of status of an issue workflow is \_\_\_\_\_,  
\_\_\_\_\_ and \_\_\_\_\_.