Jira Software
Project and issue tracking

Jira by Nataraz Chowdary & Tausief Shaikh

Overview

Jira is a project & issue tracking software, which tracks end-to-end life cycle of software development. It provides a clear picture of what is happening, at any given time, in the project. It is one of the top software development tools used by agile teams.

Key aspects

I. Plan

Create user stories and issues, plan sprints, and distribute tasks across your software team.

II. Track

Prioritize and discuss your team's work in full context with complete visibility.

III. Release

Ship with confidence and sanity knowing the information you have is always up-to-date.

IV.   Report

Improve team performance based on real-time, visual data that your team can put to use.

Setting up software

1. Visit the link below, for small teams like less than 10 members the software is free. And can be used with a wide variety of features for day-to-day project tracking.

2. https://www.atlassian.com/software/jira.

3. During setup you can choose whether to go ahead with Scrum or Kanban approach of software development.

Project creation in Jira

Your lead will be creating a dedicated project in Jira, which would be used to track your project related activities. On joining any project, your email id will be granted access to work on the Jira project.
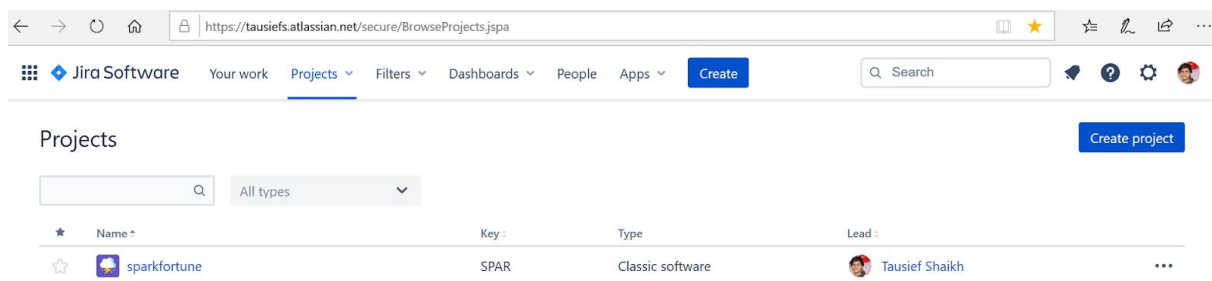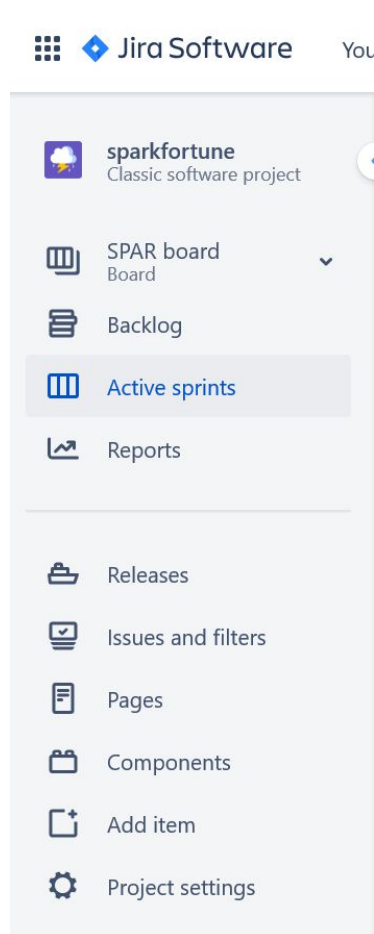

Fig. 1

Project selection view - Left panel overview

Post selecting a particular project, you would be seeing below left hand panel

SparkFortune

Name of the project.

SPAR board

There could be multiple boards created for a project. You can choose which one to select by this Board dropdown.

Backlog

All the project requirements which need to be picked up for development are added to the backlog.

Active Sprints

While working on Agile projects, this represents the current on going/active sprints. As to what work is happening and shows the project of the sprint.

Reports

At any given point of time, you can view a detailed report of what happened as part of any sprint.

Releases

All the production releases which have happened for a project will be covered as part of the Releases tab. Generally a release naming convention(release-v1.0.1) will be followed and you will have a track like what requirements will added as part of which release.

Issues and filters

This provides a searching view for all the work items which are there in Jira.

Pages & Components

On a need basis this will be applied for the project.

Agile - Scrum framework overview

The entire scrum approach of software development can be effectively performed using Jira. Below picture depicts the end-to-end software development modal using Scrum approach.
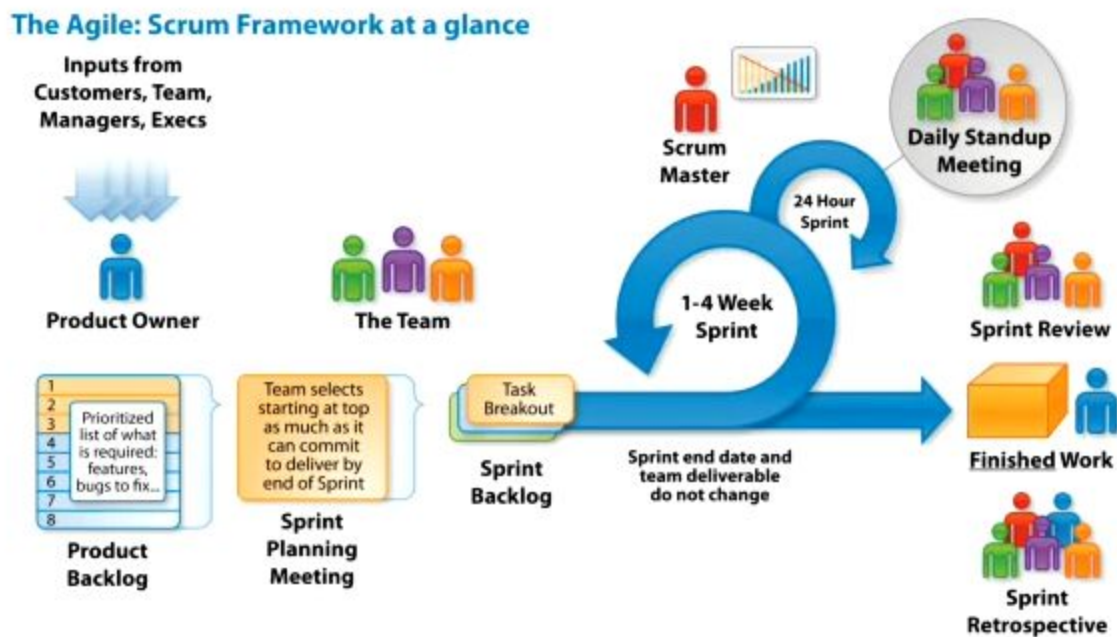

Fig. 3

Backlog

'Product Owner' interacts with clients(customers) / teams / managers / executives, to understand the project requirement and then writes the requirements in the form of user-stories so the tech team can pick it up for development.

User stories are requirements specified in a simple, clear-cut manner, generally of the nature, Given-When-Then modal, so it's easy for the tech team to start working on the requirements.

The product Owner arranges the items based on priority, so the tech-team can start picking up items one-by-one for development.
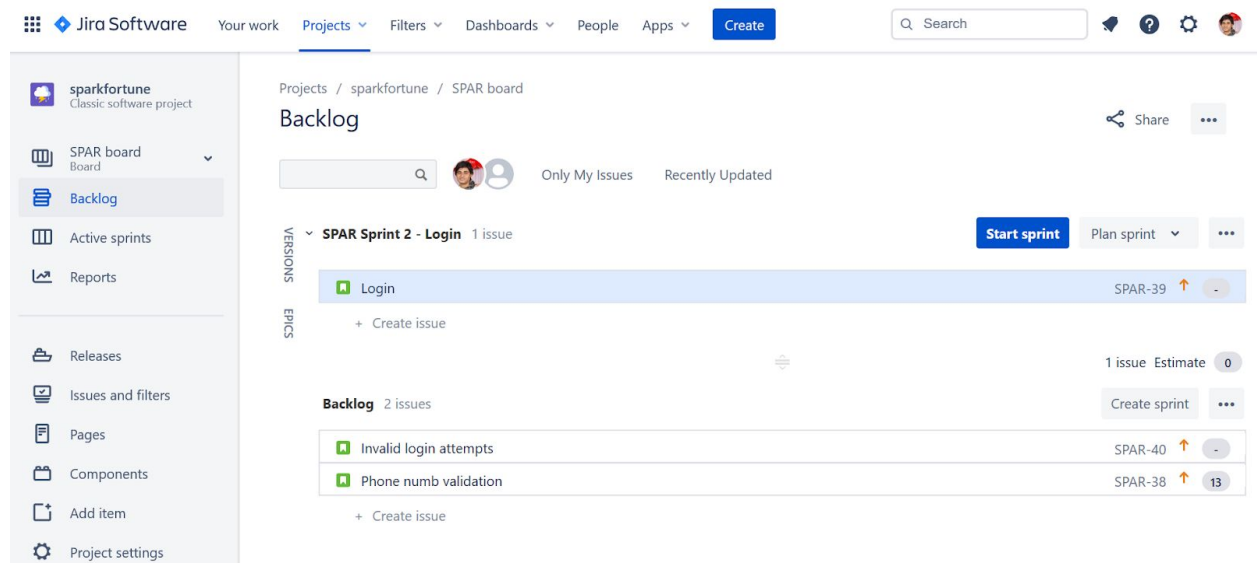
Fig. 4

Sprint Planning Meeting - Sprint creation

Under the Backlog page, the Technical manager (which is part of the agile team) will use the "Create Sprint" button to create a new sprint.

The entire team will come together as part of "Sprint Planning meeting", to decide how much work they can accomplish by the end of that sprint. The time frame of sprint will be defined (1 - 4 weeks)

Based on the amount of work planned for the sprint, user-stories will be added to this newly created sprint, by drag/drop, for using the "+ Create issue" option under the sprint.

During time roughly estimates will also be prepared.

Note: Refer to figure 4.

Sprint Backlog - Task breakdown

The amount of work / tasks / users-stories / tickets / issues, added as part of the sprint, compromises of Sprint backlog.

Detailed brain-storming will happen within the team, and sub-tasks will be created for every user-story (requirement) specified by Product owner.

Also work will be estimated by each of the 'Developer / QA'.

Generally this also happens as part of "Sprint Planning Meeting".

Start Sprint

Once the tasks are planned, the Manager/Lead of the team will start the Sprint by clicking on the "Start Sprint" button (Fig 4).

You can choose the duration of sprint, 1 - 4 weeks, even custom options there.

Start sprint

1 issue will be included in this sprint.

Sprint name:*

SPAR Sprint 2 - Login

Duration:*

2 weeks

Start date:*

28/Jun/20 7:23 PM

End date:*

12/Jul/20 07:23 PM

Sprint goal:

Fig. 5

Active Sprints

Once the sprint starts, the regular progress of the sprint, regarding how the work is progressing and are we on track to meet the sprint timeline, all such things would be reflected on Sprint Board.

Jira offers customized column view for tracking progress like, TODO, INPROGRESS, QA, DONE. You can add/remove based on your project needs.

During daily standup (every 24 hours, Fig. 3), this board will be project and progress will be tracked.

All team members are expected to keep updating their items on a regular basis to progress to reflect the current state of the system.
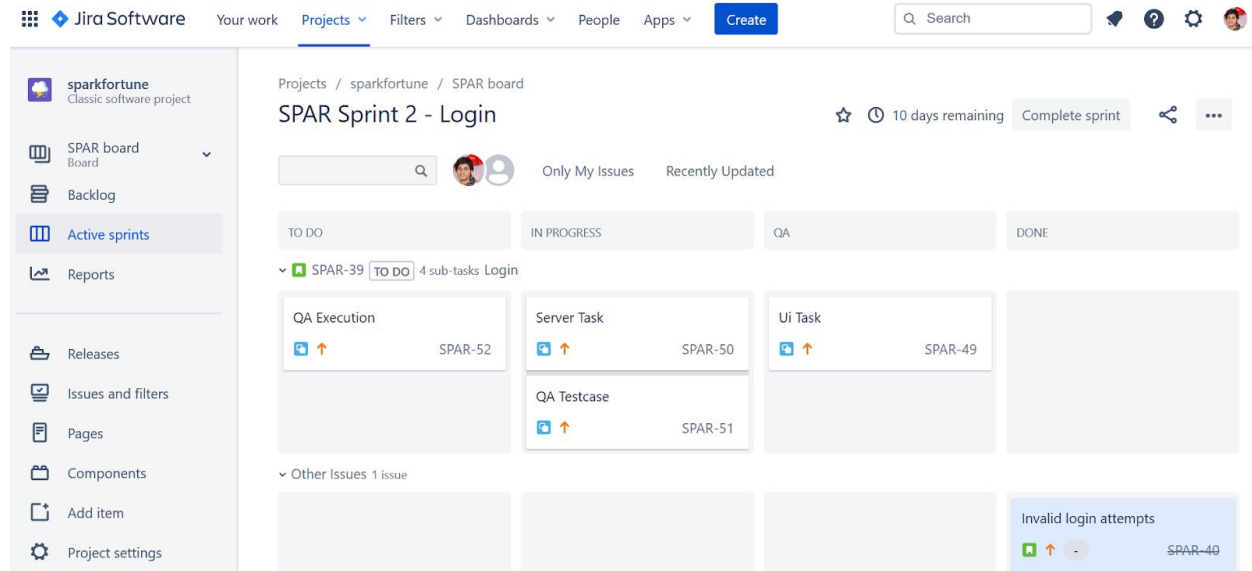
Fig. 6

Complete Sprint

The end expectations is, all the items move from TODO state to DONE state. If anything is not moved, then that item should either be moved to the next Sprint or should be moved to the backlog.

The manager / lead, will use the "Complete Sprint" button to close the sprint.

Fig. 7

Sprint Report

At any given point of time, you can visit report a sprint, but clicking "Reports => Sprint reports option". It gives you an overview of what happened for the project.

Sprint Demo

At the end of every sprint, the work will be demoed to Product Owner, Scrum master, Team. Also clients in some cases. If all approves the code will be released to LIVE users.

Managing releases

Every piece of code and feature being delivered in LIVE, should be tracked as to when delivered. The 'Releases' section in Left panel will be used for this tracking purpose. For whichever user-story, you want to assign to a release. Edit the "Fix version" field as part of the ticket and the appropriate mapping will happen.

Fig. 8

Quick vocabulary

I.   User Story

User stories are requirements specified in a simple, clear-cut manner, generally of the nature, Given-When-Then modal, so it's easy for the tech team to start working on the requirements.

II.   Task

These are generally created by the Tech team, to include more additional work which might not be expressed in the form of functional requirements.

III.   Bug

These are generally created by QA / testers, whenever the function is not working as expected.

IV.   Epic

The group of user-stories together is an EPIC. It is only to provide some structure and easy grouping of stories together for better reference

V.   Story point

For every story, a numeric value is specified which denotes the complexity & the amount of time taken for story completion. Generally a fibonacci sequence will be assigned.

Name: *Nataraz Chowdary*

Email: *nataraz@gmail.com*

Quick connect:

https://www.facebook.com/nataraz.raja

Name: *Tausief Shaikh*

Email: *tausiefshaikh17@gmail.com*

Quick connect:

https://www.facebook.com/tausiefs

https://www.linkedin.com/in/tausiefs/

https://www.instagram.com/tausief.s/