Lecture Notes for Lecture 12 of CS 5500 (Foundations of Software Engineering) for the Spring 2021 session at the Northeastern University San Francisco Bay Area Campuses.

Managing and Tracking Development Process

Philip Gust,
Clinical Instructor
Department of Computer Science

Information and examples in this lecture are based on recommended readings.

http://www.ccis.northeastern.edu/home/pgust/classes/cs5500/2021/Spring/index.html

Managing and Tracking Development Process

Review of Lecture 11

- In lecture 11, we continued our discussion about managing and tracking the development process.
- Managing and tracking software development enables teams to assess progress against their own goals and estimates, and to make adjustments as soon as possible to stay on track.
- As we learned earlier, measurement is an important aspect of estimation, and managing and tracking the software development process provides information to improve future estimates.

Managing and Tracking Development Process

Review of Lecture 11

- We also learned from lecture 9, communication is key in a software project. For successful project execution, effective communication to all stakeholders is essential.
- In this lecture, we looked at methods and representative tools for managing various kinds of communications, and for monitoring communications and measuring its effectiveness.

Managing and Tracking Development Processes

Introduction

- In this lecture, we continue discussing managing and tracking in software development to enables teams to assess progress against their own goals and estimates, and to adjust as soon as possible.
- Measurement is an important aspect of estimation and managing and tracking software development provides information to improve future estimates.
- In this lecture, we will explore managing and tracking development processes. This is a complex topic because so many aspects of development are governed by processes that must be coordinated.
- We will also discuss tools for managing and tracking software development processes.

Managing and Tracking Development Processes

What is a process?

- In the most general terms, a process is a series of steps and decisions involved in the way work is completed.
- Processes are an aspect of almost everything we do to create a software product. It also includes measurement and evaluation, which we use to improve processes.
- We have already encountered examples of processes In software development include:
  - gathering requirements
  - specifying functionality
  - developing code
  - testing code
  - creating a release

Managing and Tracking Development Processes

Something is missing

- Some important aspects are missing from this basic definition of process, and it does not adequately cover many processes.
- An example is allocating resources to carry out a design created by analyzing requirements and gathering information to help improve the process in the future.
- Beneath the the basic definition are key pieces of information that highlight problem and any process improvement efforts.

Managing and Tracking Development Processes

Timing and Interdependency

- Timing and interdependency are key to understanding process behavior.
- In the example of allocating resources, the developers with the skillset for the project may already be working on another project and will not be available for a month or more, causing a delay.
- Furthermore, other projects being planned may also need that skillset, and it could take more time to train or higher other developers, resulting in another month or more of delay.

Managing and Tracking Development Processes

Timing and Interdependency

- Understanding the timing of processes and their interdependencies is critical to any improvement methodology, yet it remains unseen because the basic definition doesn't describe it.
- The basic definition of a process does not include the framework to describe when a process occurs.

Managing and Tracking Development Processes

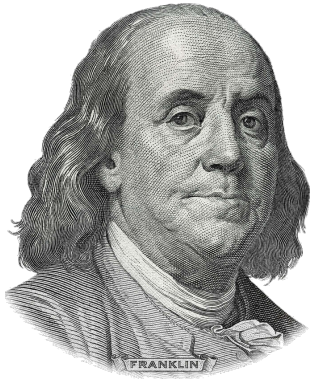
Variability

- Averages are used to describe processing time for activities under study. Averages work well to describe a process that contains no variability in time or direction, and without interruptions.
- In other words, averages hide critical business process behavior. Why do averages mask process behavior?
- It would be like trying to explain the dynamics of a team from a single photograph. You get some information from a picture, but you have no clue what happened before or after it was taken.

Managing and Tracking Development Processes

Variability

- Variable times cause loss of throughput because the system can not be balanced precisely. That means from the perspective of the entity, waiting will occur at various points in the process.
- Sometimes there is more waiting than processing. Benjamin Franklin once said, “Lost time is never found again.”



- If variability in “time and direction” are ignored, the real capability of a system will be hidden.
- Understanding the effect of the variability in the resource allocation process mentioned earlier can give rise to new solutions that mitigated the extreme variability.

Managing and Tracking Development Processes

Why are these factors ignored?

- If timing, interdependency, and variability are important, then why are they ignored in the definition of a process?
- Traditional tools don't have the framework for capturing or seeing the effect of these critical process factors.
- Without a framework for capturing and using this information, it will be ignored. The framework opens the eyes of software development teams so that they can make effective decisions.
- Basically, if there is no framework that encompass these additional aspects of process, then they cannot be seen, tracked, or managed.

Managing and Tracking Development Processes

An expanded definition of a process

- In an expanded definition, a process consists four major elements:
  - ***Steps and decisions***: a series of steps and decisions describing the way work is completed.
  - ***Variability of processing time and flow***: the pattern of processing times.
  - ***Timing and interdependence***: when the arrivals happen, when people work, etc.
  - ***Assignment of resources***: how many and where are they assigned.

Managing and Tracking Development Processes

An expanded definition of a process

- This expanded definition of a process describes actual behavior of the current process.
- Taking these additional elements into account, the characteristics of a process can be more accurately measured, changed, and key metrics tested before committing resources.
- Estimation and process analytic models can take these additional elements into account if they are based on large enough data sets that discriminate based on these elements.
- However empirical models using data from past performance within the same organization or group that include these elements can often yield the most accurate results.

Managing and Tracking Development Processes

Why is managing and tracking processes so complex?

- Managing and tracking processing is inherently complex. Having good tools that implement some processes and tracking tasks can assist in making it more uniform and certain.
- However, some aspects of process management and tracking still require human input, and in most cases the human has the primary responsibility.
- The complexity is the result of several aspects of the processes themselves. These include:
  - Uncertainty caused by process variability
  - Timing and interdependence within and between processes
  - Wide variety in the number and types of processes

Managing and Tracking Development Processes

Complexity caused by process variability

- *Processes variability* causes complexity because steps in the process are subjected to unknown delays. Variable times cause loss of throughput because the system can not be balanced precisely.
- Since these delays can occur at various points in the process, the task of managing the process becomes more complex, especially if the effect of delays can propagate during the process.
- One way to mitigate this kind of complexity is to carefully assess why delays occur, and redesign the process to minimize, or at least bound, the process variability and to eliminate extremes.

Managing and Tracking Development Processes

Complexity caused by timing and interdependence

- *Timing and interdependence* add complexity because the process is subject to influences that require accounting for interactions and dependencies within and among processes.
- As the number of relationships grow, the complexity of the process also grows, sometimes very rapidly, and it becomes increasingly difficult to analyze.
- Ways to mitigate this kind of complexity include use techniques like decomposition to analyze relationships, reduce their number where possible, and account for remaining ones during planning.

Managing and Tracking Development Processes

Complexity caused by number and variety of process types

- Software development includes many types of processes that must be managed during a project, with their own ways to manage and track them. Some of the types of processes include:
  - Projects
  - Requirements
  - Issues
  - Defects
  - Schedules
  - Tasks
  - Tests
  - Releases
  - Automations
  - Notifications

Managing and Tracking Development Processes

Complexity caused by number and variety of process types

- The many different process types introduces complexity because managing across different types of processes defines relationships among process that must be managed and tracked.
- The number of processes introduces complexity because as the number increases so does the number of relationships that must be managed and tracked
- One way to mitigate this kind of complexity is reducing the interactions among process types to limit how many need to be managed together and reducing numbers of processes and types.

Managing and Tracking Development Processes

Integrating process management and tracking

- The idea of integrating the management and tracking of processes is attractive, because managing the many types and the number of processes required during a software project is very challenging.
- Integrating management and tracking of processes during a project addresses the problems of timing and interdependence because relationships among different process can be made explicit.
- The problem of variability could also be analyzed and managed at a global level rather than for each individual process.
- Finally, common ways of interacting with different types of processes and the number of processes would make it easier to measure and track progress, and to gather statistics for future use.

Managing and Tracking Development Processes

Integrating process management and tracking

- A class of products known as *Integrated Management Systems* (*IMS*) seek to integrate process management and tracking across processes in a project or an enterprise.
- Because of the complexity of managing such a wide range process types, many of these systems specialize in domain-specific processes, rather than being a general-purpose system.
- For example, there are IMS to manage processes specific to the food industry, and ones designed for business processes.
- For the software development, the tools are either process-specific (e.g., defect tracking, issue management), or they are integration frameworks that tie together process-specific external tools.
- We will discuss the integration framework approach shortly.

Managing and Tracking Development Processes

Elements of an integrated process management system

- A 2005 paper by Ali Bahrami of Boeing Phantom Works proposed an integrated process management system. It is useful to look briefly at the architecture of the system (see Readings).
- According to the paper, an integrated process management system can be achieved through an integration of:
  - Process modeling and simulation
  - Process management and scheduling
  - Workflow management system (WFMS)
- The interface between scheduling and workflow management supports a rollup of workflow data to support workflow plans, schedules, status, and critical chain analysis.

Managing and Tracking Development Processes

Elements of an integrated process management system

- According to the author,
  - An integrated process management environment supports coordination of program tasks, reducing the time required to perform these tasks and the costs of coordinating them.
  - Process owners or their designees construct executable models that define the task elements, the roles responsible for each task element, and the rules that determine task element sequencing.
  - Tasks are initiated automatically in accordance with the program schedule by alerting the individuals responsible for performing the work.
  - As each person completes a task element, work is automatically routed in accordance with the established process to the next person, who may be selected from an available pool of people based on resource loadings.
  - The status of all current tasks is visible and updated automatically as task elements are completed.”

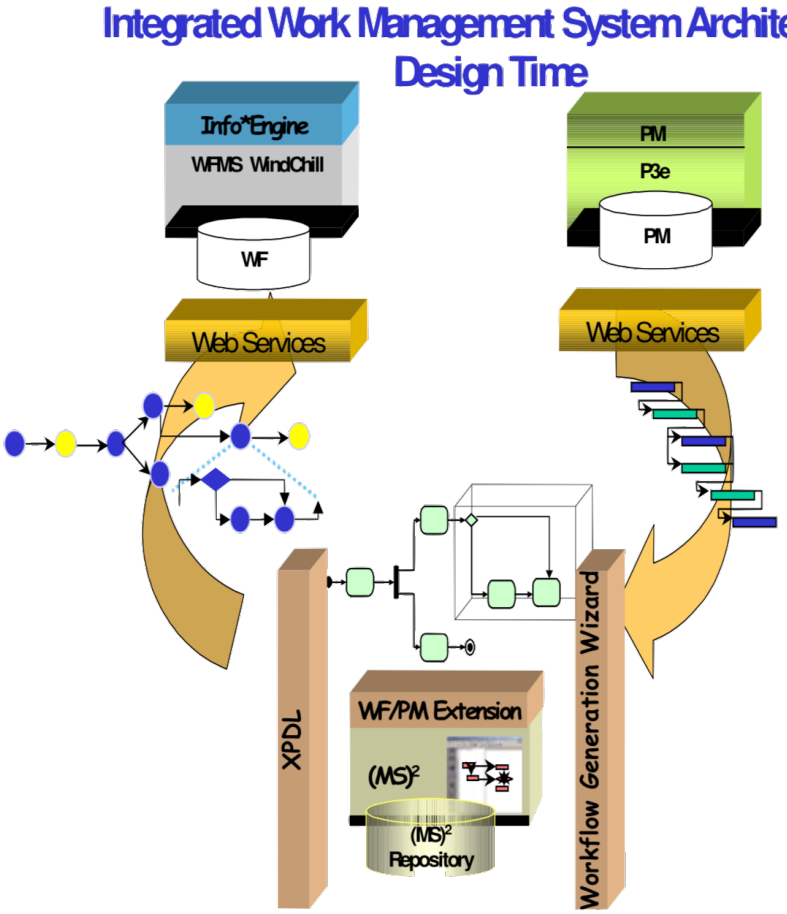
Managing and Tracking Development Processes

Elements of an integrated process management system

- The paper lists the benefits of such a system:
  - Tool/Vendor Neutral Workflow Process Definition
  - Reusing library of Process Components
  - Workflows executed according to schedule
  - Schedule changes reflected in the WFMS
  - Ad-hoc changes in WFMS reflected in schedule
  - Schedule shows always correct status and history
  - Improved efficiency
  - Better process control
  - Improved visibility and documentation
  - Flexibility to redesign with changing needs
  - Process improvement

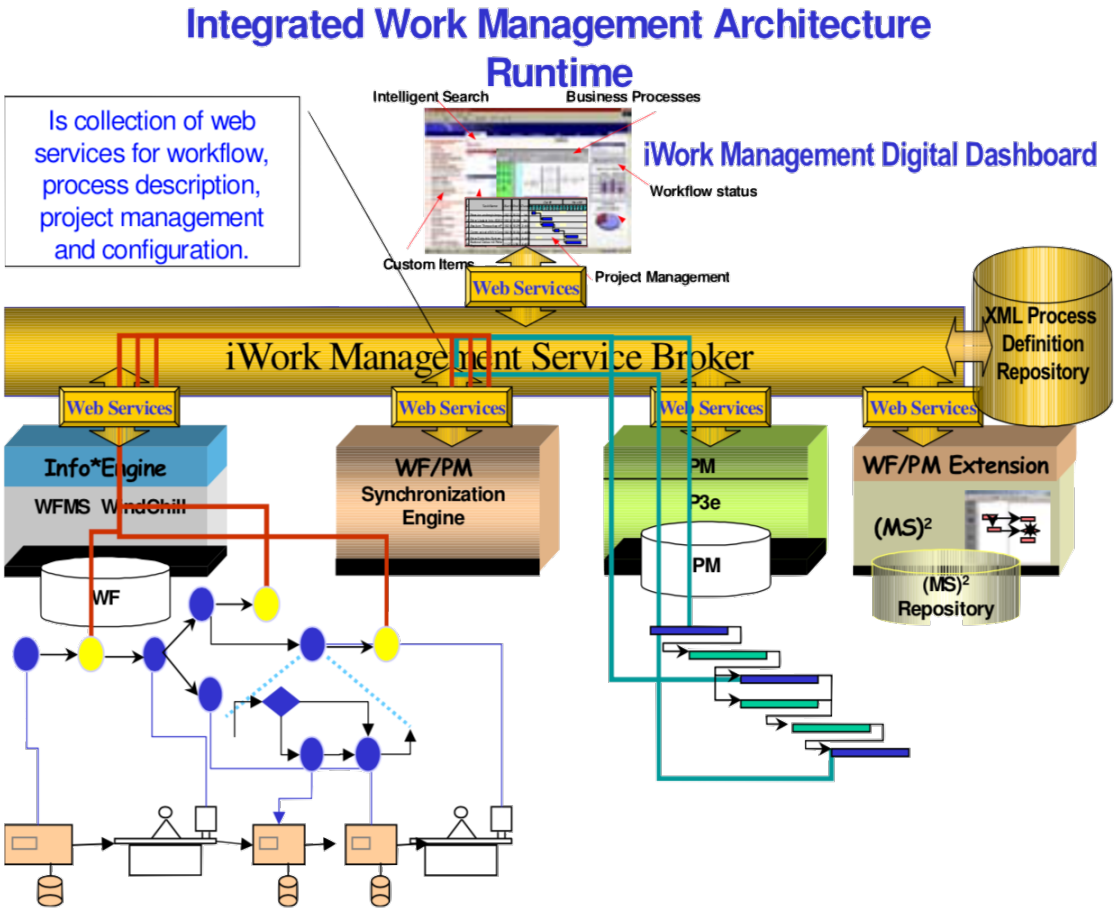
Managing and Tracking Development Processes

Elements of an integrated process management system



Managing and Tracking Development Processes

Elements of an integrated process management system



Managing and Tracking Development Processes

Elements of an integrated process management system

- A commercial product from ProcessModel seems to provide similar capabilities and to follow a similar architectural model. (https://www.processmodel.com/)

Managing and Tracking Development Processes

Process integration frameworks

- In cases where there are separate process management tools for different classes of processes, another approach is to use a process integration framework to tie these separate tools together.
- Process integration frameworks may be capable of modeling and executing processes, but they also provide mechanism for integrating external process tools:
  - Hooks to receive events or data from external products
  - Integration points for sending events or data to external products
  - Scripting or other automation mechanisms for adding new behaviors that receive or send data to external products
- In some cases, several process tools provide these integration features and are capable of 2- and even *n*-way integration.

Managing and Tracking Development Processes

Example: JIRA issue management tool from Atlassian.

- JIRA is a tool developed by Australian Company Atlassian. The name "JIRA" is inherited from the Japanese word "Gojira" which means "Godzilla".
- The basic use of this tool is to track issues bugs, and change requests related to your software.
- Jira can be used in help desk, support, and customer services to create tickets and track resolution and status of created tickets.
- It is also useful in project management, task tracking, requirements management, and for workflow and process management.
- Note: The instructor has no financial interest in this company. It is presented here to demonstrate IPM implementation strategies.

Managing and Tracking Development Processes

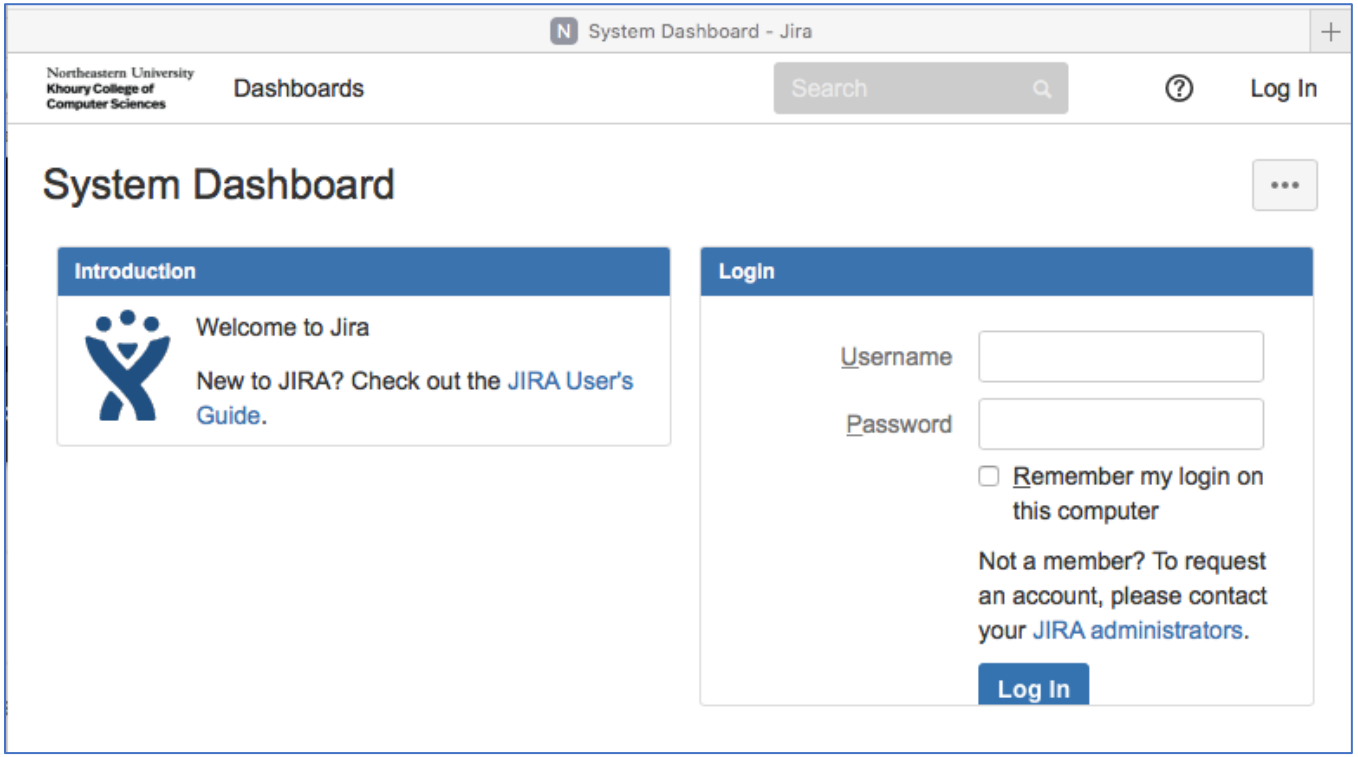
Example: JIRA issue management tool from Atlassian.

- JIRA is an example of a process management tool for managing specific kinds of process, including bugs, issues, and some project management features.
- JIRA also provides all three kinds of integration facilities:
  - Hooks to receive events or data from external products
  - Integration points for sending events or data to external products
  - Scripting or other automation mechanisms for adding new behaviors that receive or send data to external products
- As such, other tools like GitHub and SLACK can integrate into JIRA, but the other tools can also act as integration frameworks and integrate JIRA into themselves.

Managing and Tracking Development Processes

Example: JIRA issue management tool from Atlassian.

- Before looking at integrations, we will first look JIRA's features for managing the class of issue- and bug-based processes.



Managing and Tracking Development Processes

JIRA - core features

- **Boards**
  - JIRA supports Scrum and Kanban boards.
  - These boards provide an immediate snapshot of the project to the team.
  - Helps to quickly review the progress of the project and see the status of the individual tasks.
  - Board workflow can be customized to fulfil the way a team wants to proceed.

Managing and Tracking Development Processes

JIRA - core features

- **Business project template**
  - JIRA supports *several* business templates to manage simple tasks and complex tasks like workflow.
  - Template can be customized based on the team and their approach. Ex: Workflow can be customized based on each team's approach.
  - Every step is accounted, and team can move to achieve their goals.

Managing and Tracking Development Processes

JIRA - core features

- **Task details**
  - Tasks can be defined at the individual level to track the progress.
  - Status of every task, comment, attachment and due dates are stored in one place

Managing and Tracking Development Processes

JIRA - core features

- **Notifications**
  - An email can be sent for a particular task to the users.
  - Voting and watching features to keep an eye on the progress for the stakeholders.
  - Use @mention to get the attention of a specific team member at Comments/Description.
  - User will instantly notify if something is assigned or if any feedback is required.

Managing and Tracking Development Processes

JIRA - core features

- **Search**
  - JIRA supports a powerful search functionality with Basic, Quick and Advanced features.
  - Use the search tool to find answers like due date, when a task was last updated, what items a team member still needs to finish.
  - Project information at one place, search within a project.

Managing and Tracking Development Processes

JIRA - core features

- **Reports**
  - JIRA supports more than a dozen reports to track progress over a specific timeframe, deadlines, individual's contribution, etc.
  - Easy to understand and generate different reports that help to analyze how the team is going on.
  - Easy to configure these reports and display the matrices to the stakeholders.

Managing and Tracking Development Processes

JIRA - core features

- **Scales with team growth**
  - JIRA supports any business team and any project irrespective of size and complexity.

Managing and Tracking Development Processes

JIRA - core features

- **Add-ins**
  - JIRA supports more than 100 add-ins to connect with different software to make work easy.
  - Wide range of add-ins makes it as universal across the globe.
  - can be integrated with many other tools – Subversion, GIT, ClearCase, Team Foundation Software, Mercury, Concurrent Version System and many more.

Managing and Tracking Development Processes

JIRA - core features

- **Multilingual**
  - JIRA supports more than 10 languages that are as widely used as English (US, UK, India), French, German, Portuguese, Spanish, Chinese, Korean, Japanese and Russian.

Managing and Tracking Development Processes

JIRA - core features

- **Mobile App**
  - JIRA is available as a Mobile Application as well.
  - It is available on Google Play Store and App Store (iTunes) of Apple.
  - Easy to stay connected with the team while moving anywhere with notification, comments and project activity.

Managing and Tracking Development Processes

JIRA - core features

- **Database support**
  - JIRA supports MySQL, Oracle, PostgreSQL and SQL Server in the backend.

Managing and Tracking Development Processes

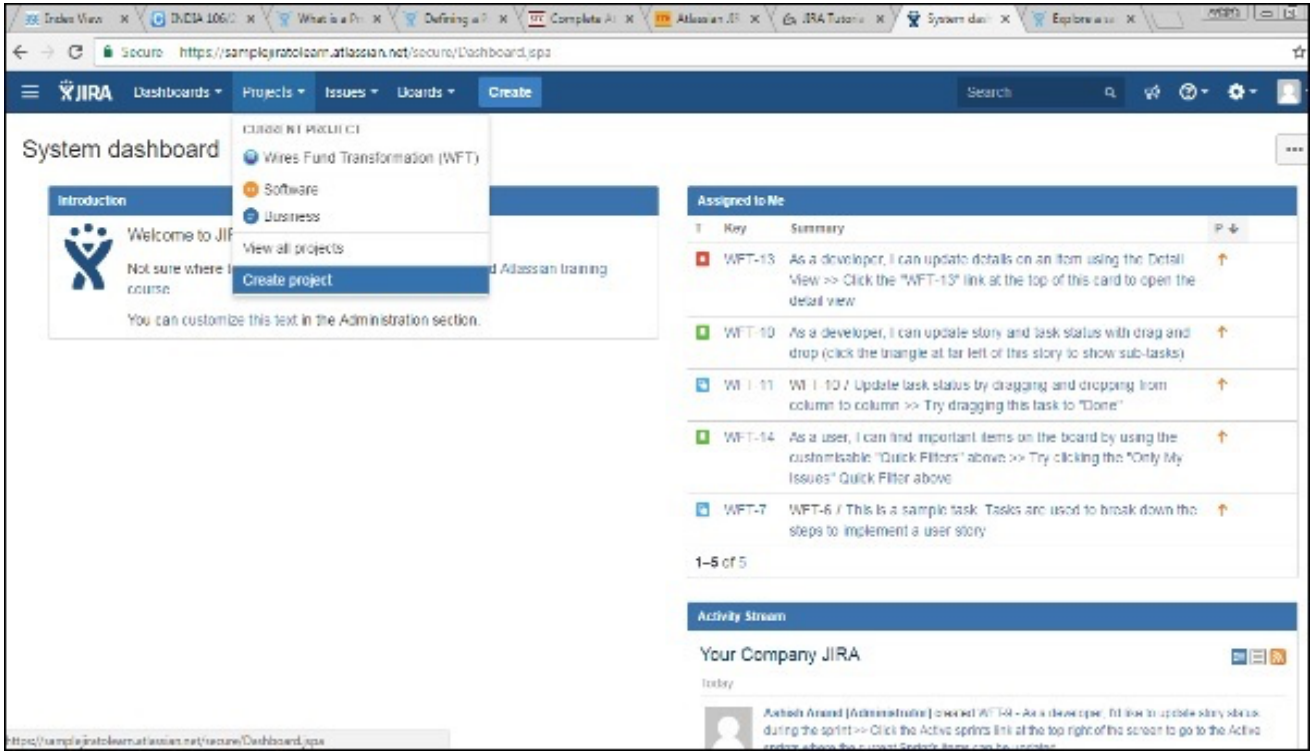
JIRA - project

- A JIRA project is a collection of issues of one of these types:
  - Software Development Project
  - Marketing Project
  - Migration to another platform project
  - Help Desk Tracking Project
  - Leave Request Management System
  - Employee Performance System
  - Website Enhancement

Managing and Tracking Development Processes

JIRA - project

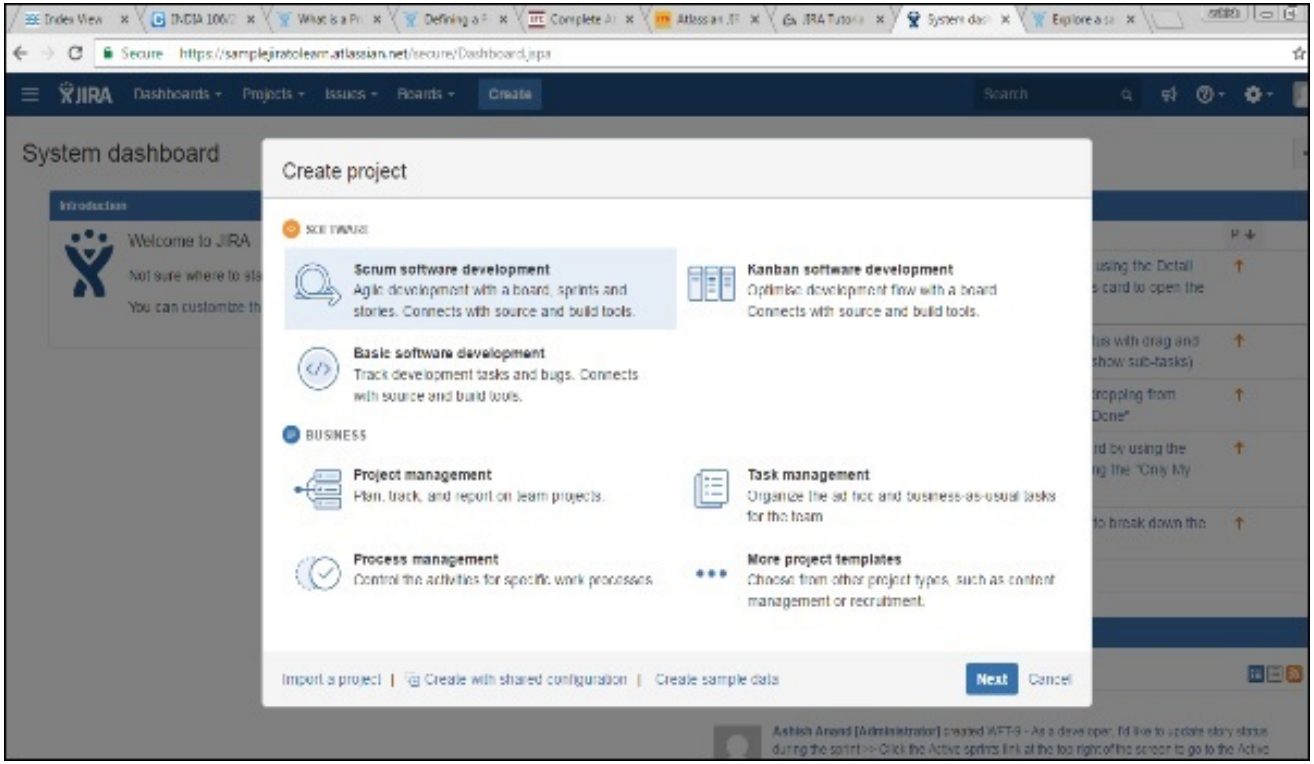
- Creating a project requires logging in as a JIRA Service Desk Admin. Then, select **Project** → **Create Project**.



Managing and Tracking Development Processes

JIRA - project

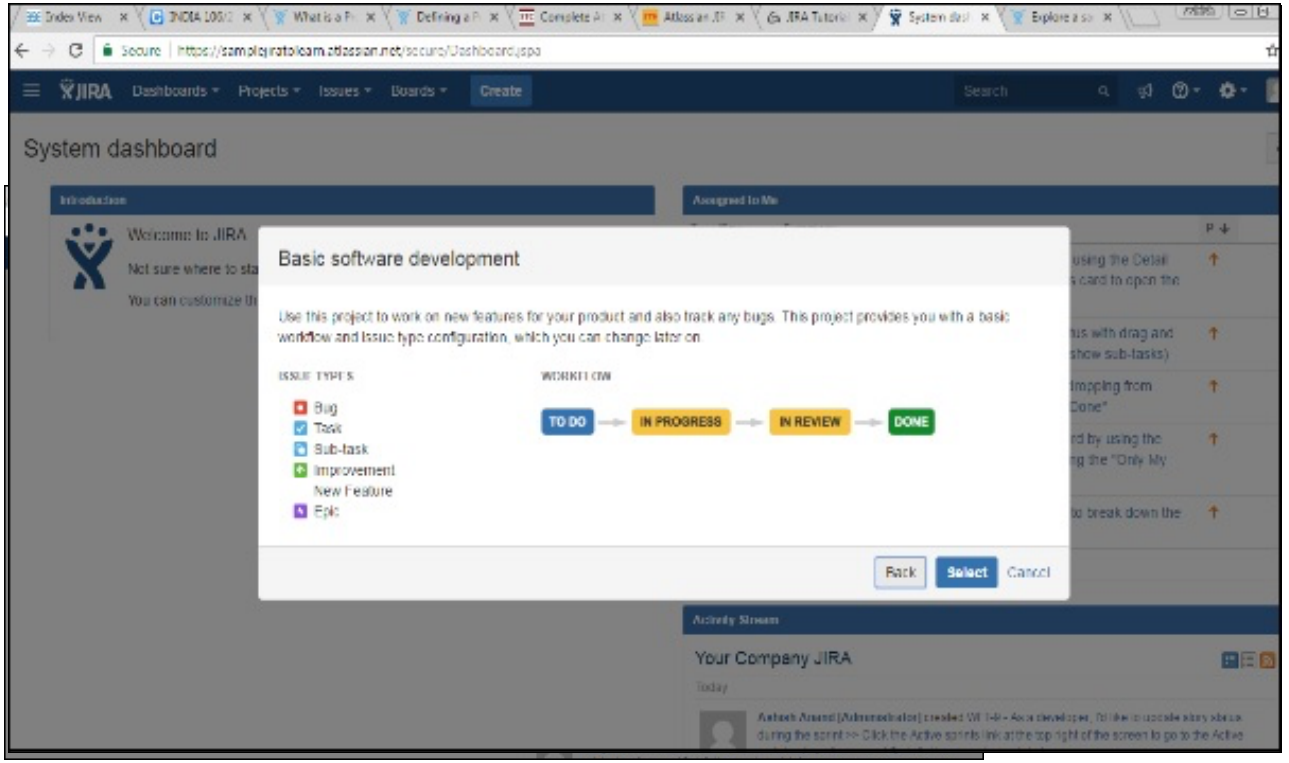
- Choose the type of project that suits your requirement. This screenshot shows the types of projects available in JIRA.



Managing and Tracking Development Processes

JIRA - project

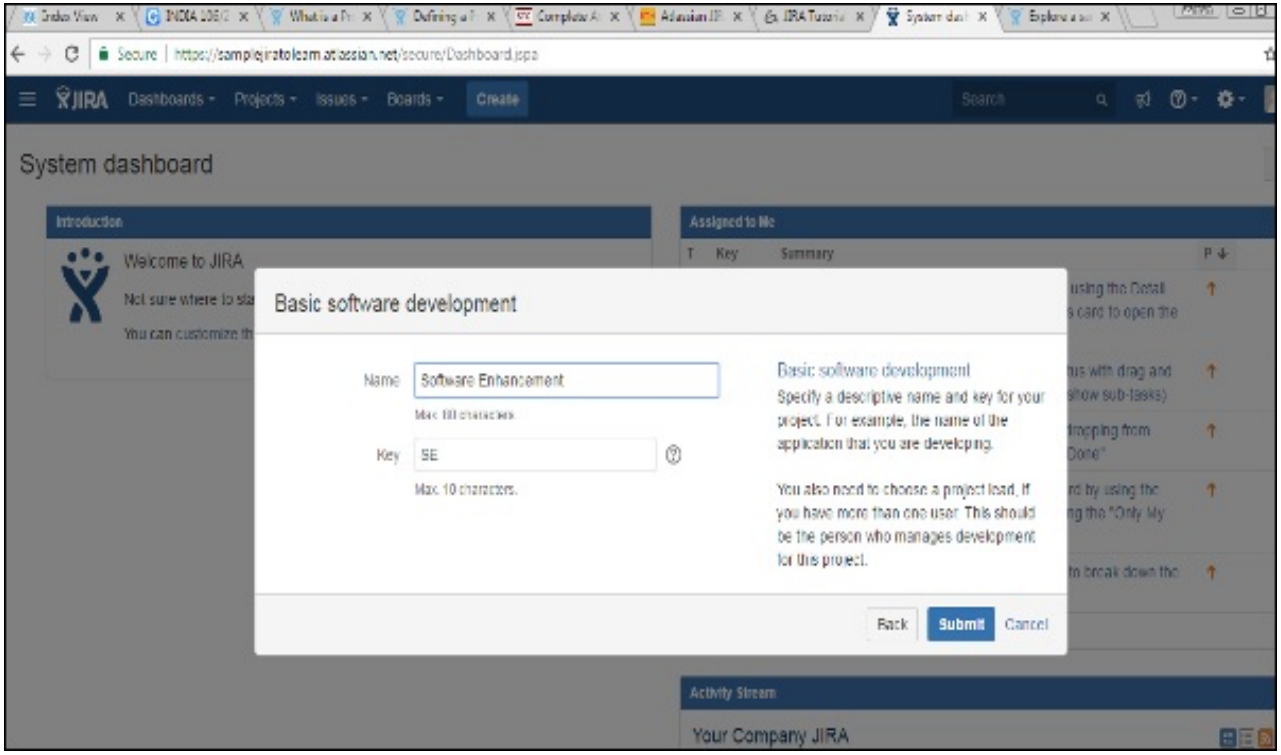
- Select Basic software development and click on Next. You will see the flow of the project you selected and the available issue types.



Managing and Tracking Development Processes

JIRA - project

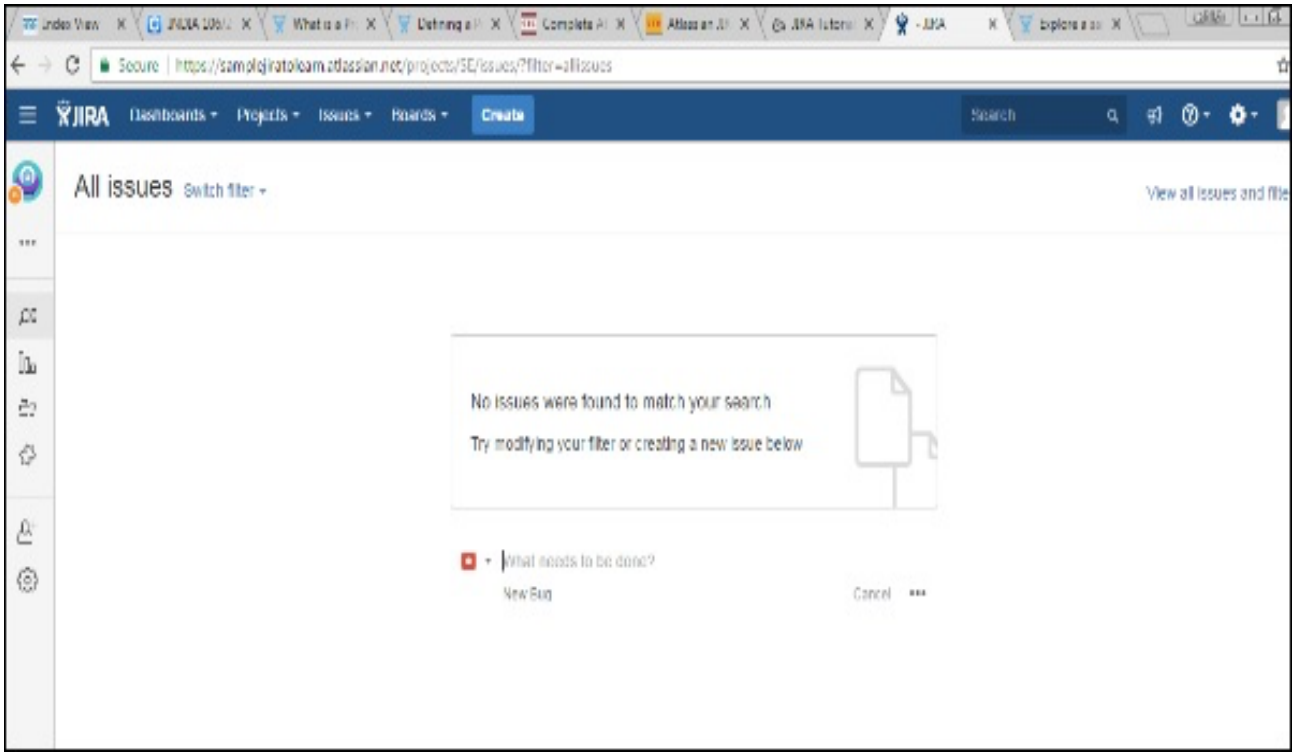
- Choose on **Select**, then enter project name, confirm the Key to display as a reference in all the issues, and select **Submit** button.



Managing and Tracking Development Processes

JIRA - project

- The page displays all project issues. This screenshot shows whether any issues are linked with a newly created project.



Managing and Tracking Development Processes

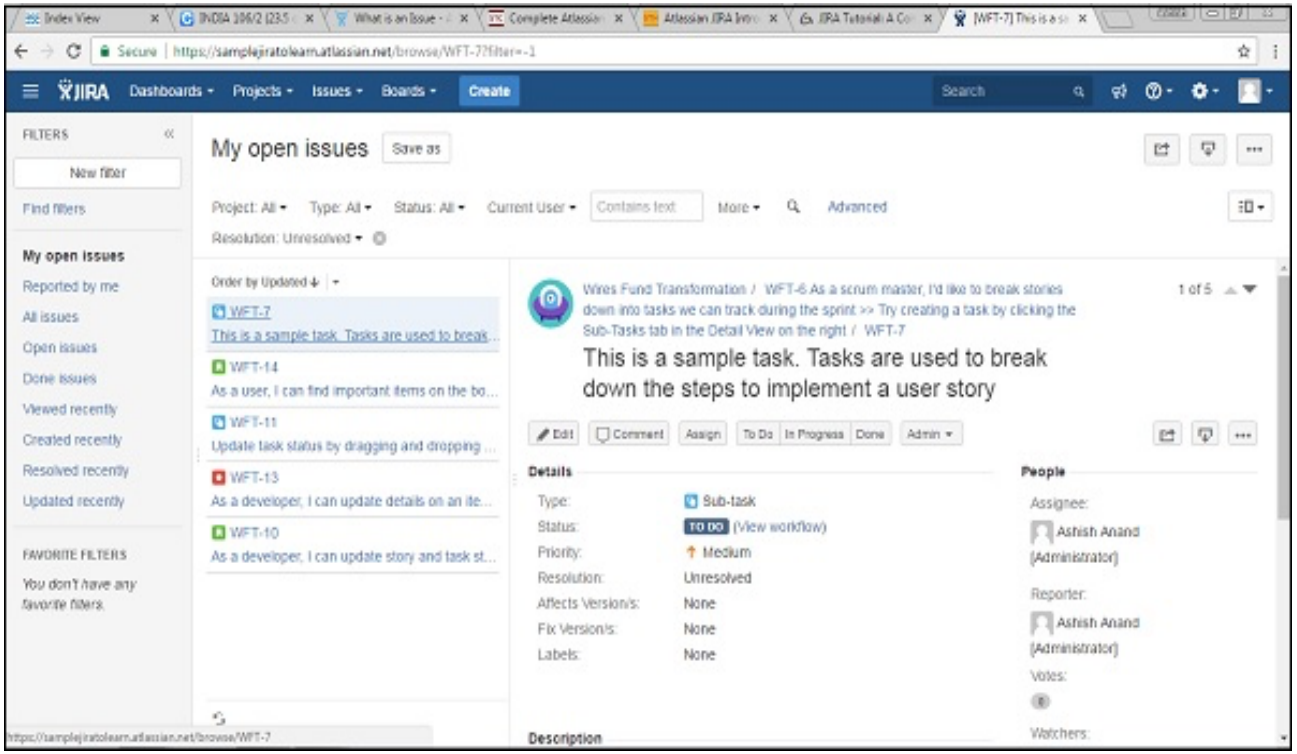
JIRA - issues

- JIRA is a project management tool and uses issues to track all the tasks. An issue helps to track all works that underlie in a project.
- In real time, every work or task either technical, non-technical, support or any other type of a project in JIRA are logged as an issue.
- An issue can be dependent on the organization and requirements:
  - Story of a project
  - Task of a story
  - Sub-task of a story
  - A defect or bug can be an issue
  - Helpdesk Ticket can be logged as issue
  - Leave Request

Managing and Tracking Development Processes

JIRA - issues

- This screen shot shows the page and information for open issues:



Managing and Tracking Development Processes

JIRA - workflow

- In JIRA, *workflow* is used to track the *lifecycle* of an Issue. Workflow is a record of statuses and transitions of an issue during its lifecycle.
- A *status* represents the stage of an issue at a particular point. An issue can be in only one status at a given point of time like *Opened*, *To Do*, *Done*, *Closed*, *Assigned*, etc.
- A *transition* is a link between two statuses when an issue moves from one status to another.
- For an issue to move between two statuses, a transition must exist. In a simple way, a transition is work done on the issue, while status is the impact of work on that issue.

Managing and Tracking Development Processes

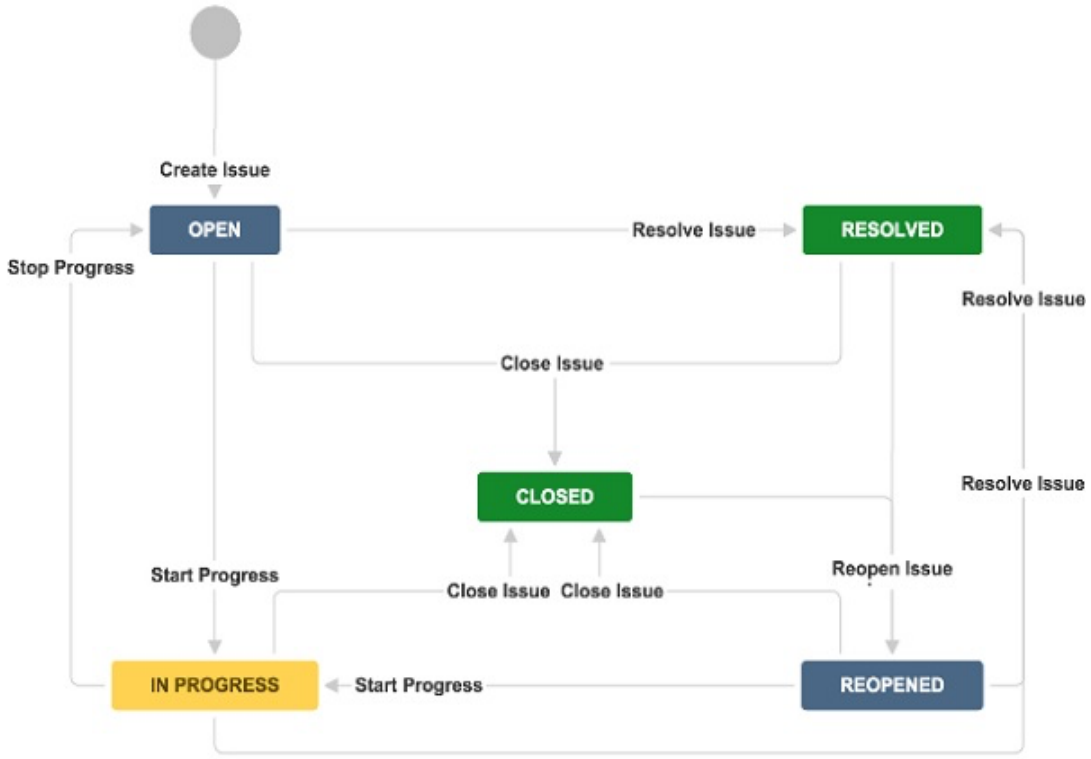
JIRA - workflow

- For example, an issue is created and opened. When the assignee starts working on it, the issue moves to In Progress. The transition is starting the work; the status of the issue is now progressive.
- JIRA workflow tracks these stages as soon as an issue is created:
  - **Open Issue** − After creation, the issue is open and can be assigned to the assignee to start working on it.
  - **In Progress Issue** − The assignee has actively started to work on the issue.
  - **Resolved Issue** − All sub-tasks and works of that Issue are completed. Now, the issue is waiting to be verified by the reporter. If verification is successful, it will be closed or re-opened, if any further changes are required.
  - **Reopened Issue** − This issue was resolved previously, but the resolution was either incorrect or missed a few things or some modifications are required. From Reopened stage, issues are marked either as assigned or resolved.
  - **Close Issue** − The issue is considered as finished; resolution is correct as of now. Closed issues can be re-opened later based on the requirement.

Managing and Tracking Development Processes

JIRA - workflow

- The following diagram shows a standard workflow.



Managing and Tracking Development Processes

JIRA - workflow

- A transition is a one-way link. If an issue moves back and forth between two statuses, two transitions should be created.
- For example, there are two-way transitions between closed and re-opened statuses. A closed issue can be reopened if modifications are required at any time until the project completes.
- A re-opened issue can be closed directly if additional work is taken care in another issue and no specific work has been done on the re-opened issue.

Managing and Tracking Development Processes

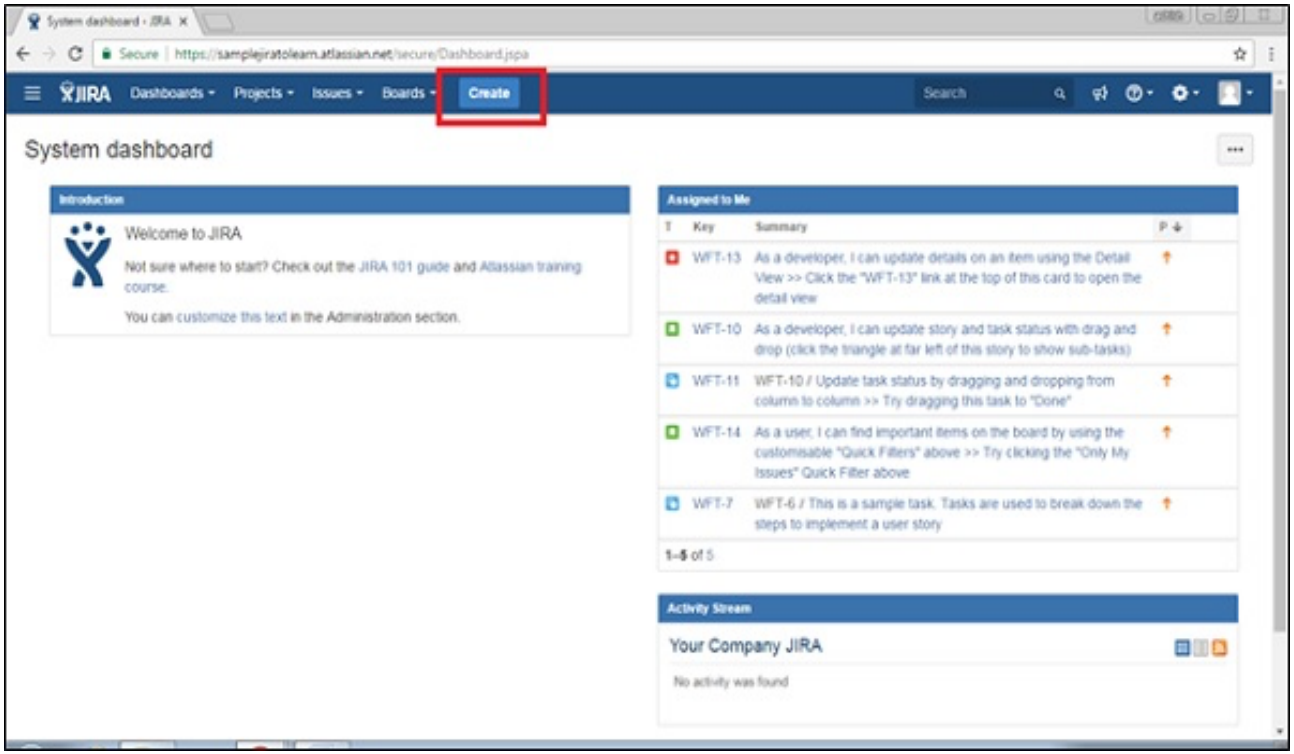
JIRA - issue types

- JIRA tracks tasks, sub-tasks, or work as an Issue. There are several types of Issues to identify work and categorize similar issues:
  - **Sub-Task** − This is the sub-task of an issue. In a logged issue, there can be different tasks to resolve it, which are called as sub-tasks.
  - **Bug** − A problem that impairs or prevents the functions of the product.
  - **Epic** − A big user story that needs to be broken down. Created by JIRA Software - do not edit or delete.
  - **Improvement** − An improvement or enhancement to an existing feature or task.
  - **New Feature** − A new feature of the product, which is yet to be developed.
  - **Story** − A user story. Created by JIRA Software - do not edit or delete.
  - **Task** − A task that needs to be done to achieve team's goal.

Managing and Tracking Development Processes

JIRA - issue creation

- Select the **Create** button in the navigation bar to open the create issue dialogue box.



Managing and Tracking Development Processes

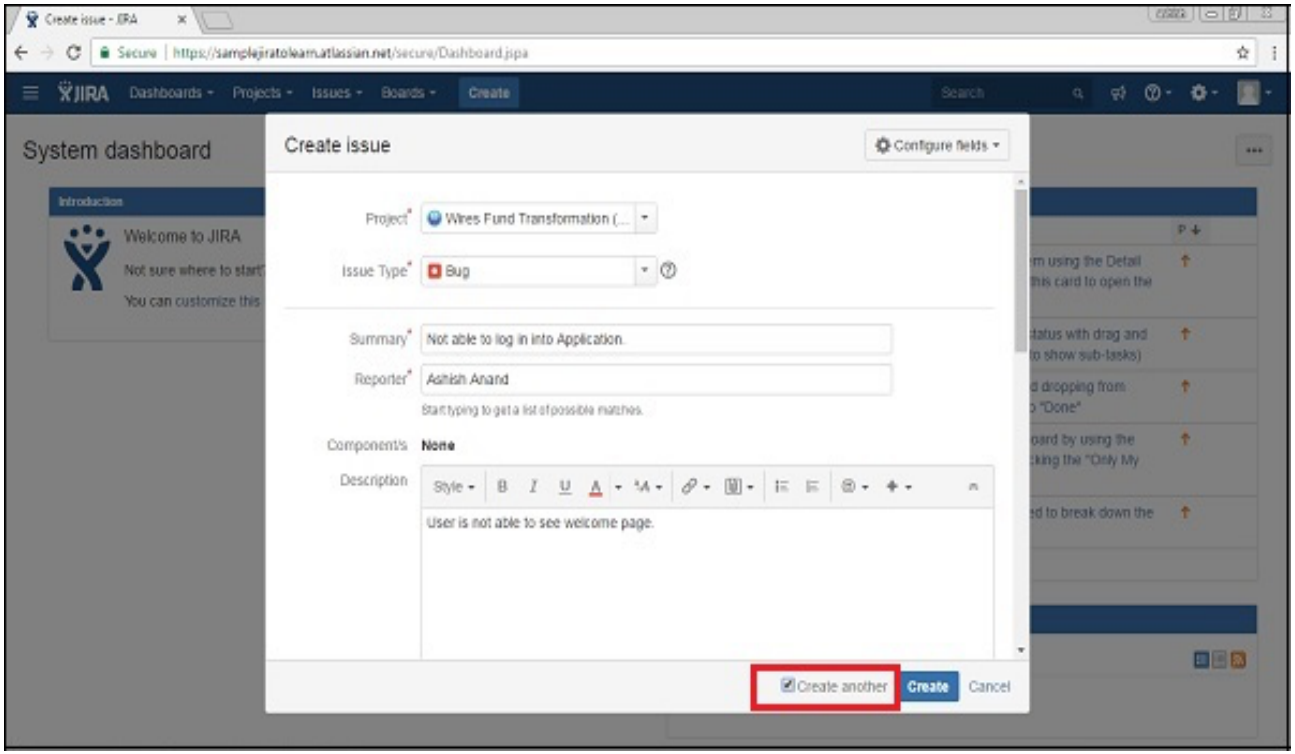
JIRA - issue creation

- To complete the process of creating an issue, follow these steps:.
  - Select the Project where the issue is.
  - Select the type of issue, whether it is a bug/new feature/story, etc.
  - Write a one-line summary to provide the overall idea about the issue.
  - Write the details of the issue in the Description field. Explain the issue, so that stakeholders can understand every detail of the issue.
  - To create a similar type of issue in the same project and issue type, check the checkbox of “Create another” otherwise keep unchecked.
  - After entering all the details, click on the Create button.

Managing and Tracking Development Processes

JIRA - issue creation

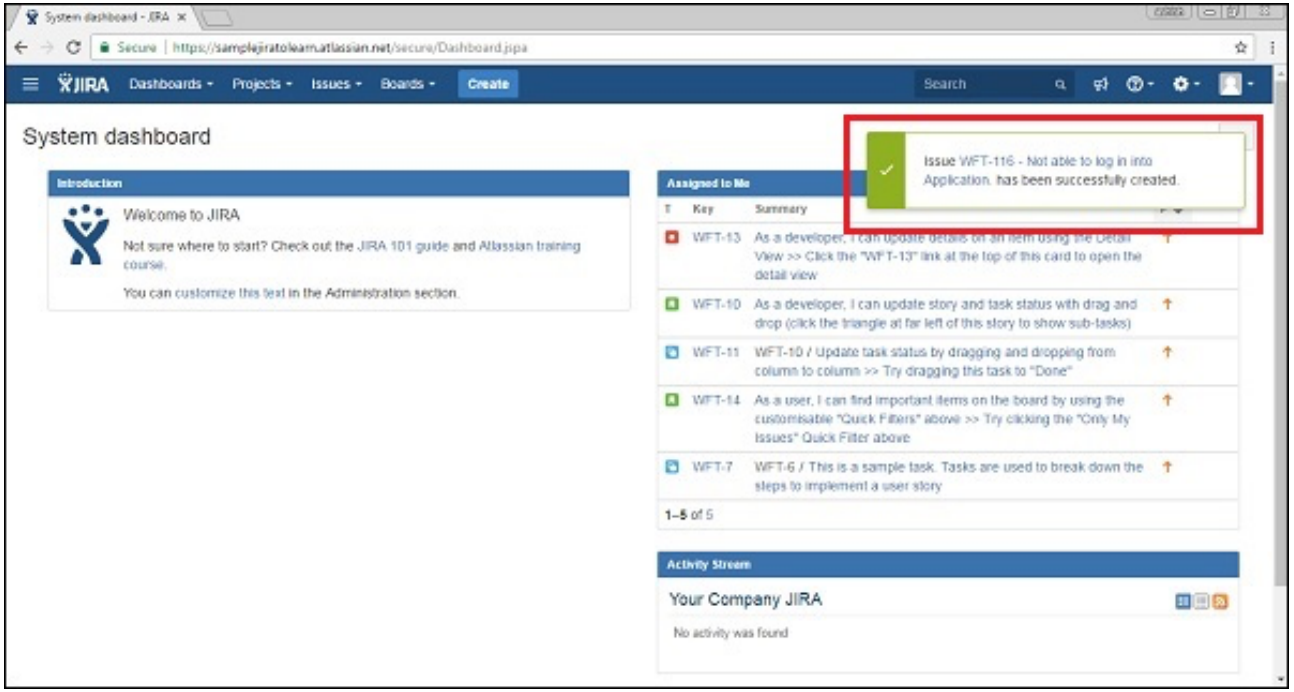
- This screenshot shows how to create an issue by providing the required and optional details.



Managing and Tracking Development Processes

JIRA - issue creation

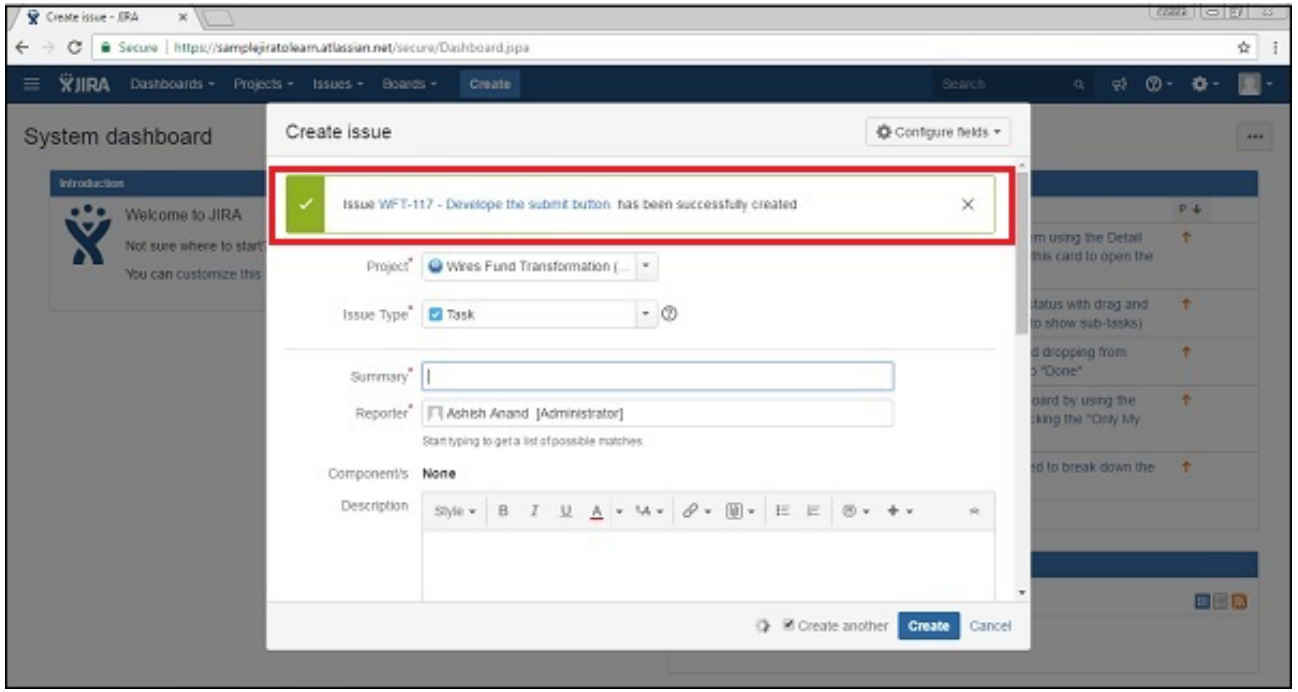
- This screen shot shows that the issue was created. If the “Create another” checkbox was not checked, you will have to navigate to the Dashboard to see the issue id and summary.



Managing and Tracking Development Processes

JIRA - issue creation

- If the “Create another” checkbox is checked you, will get a new create issue page along with the issue id.



Managing and Tracking Development Processes

JIRA - issue creation

- The fields in the Create Issue form are explained below.
  - **Priority** − Issue creator can set the priority to resolve the issue as High, Medium, Low, and Lowest.
  - **Labels** − It is similar to Tag; it helps in filtering out specific types of issues.
  - **Linked Issue** − It links other issues that are either dependent on this issue or this issue is dependent on them. Options in dropdowns are – block, is blocked by, duplicate, clone, etc.
  - **Issue** − User can link the issue by the Typing ID or summary of those that are related to the linked issue field.
  - **Assignee** − The person who is responsible to fix this issue. Assignee name can be entered by the issue creator.
  - **Epic Link** − An Issue creator can provide an epic link, if the issue belongs to any of those.
  - **Sprint** − The user can define in which sprint this issue belongs to, and when this issue should be addressed.

Managing and Tracking Development Processes

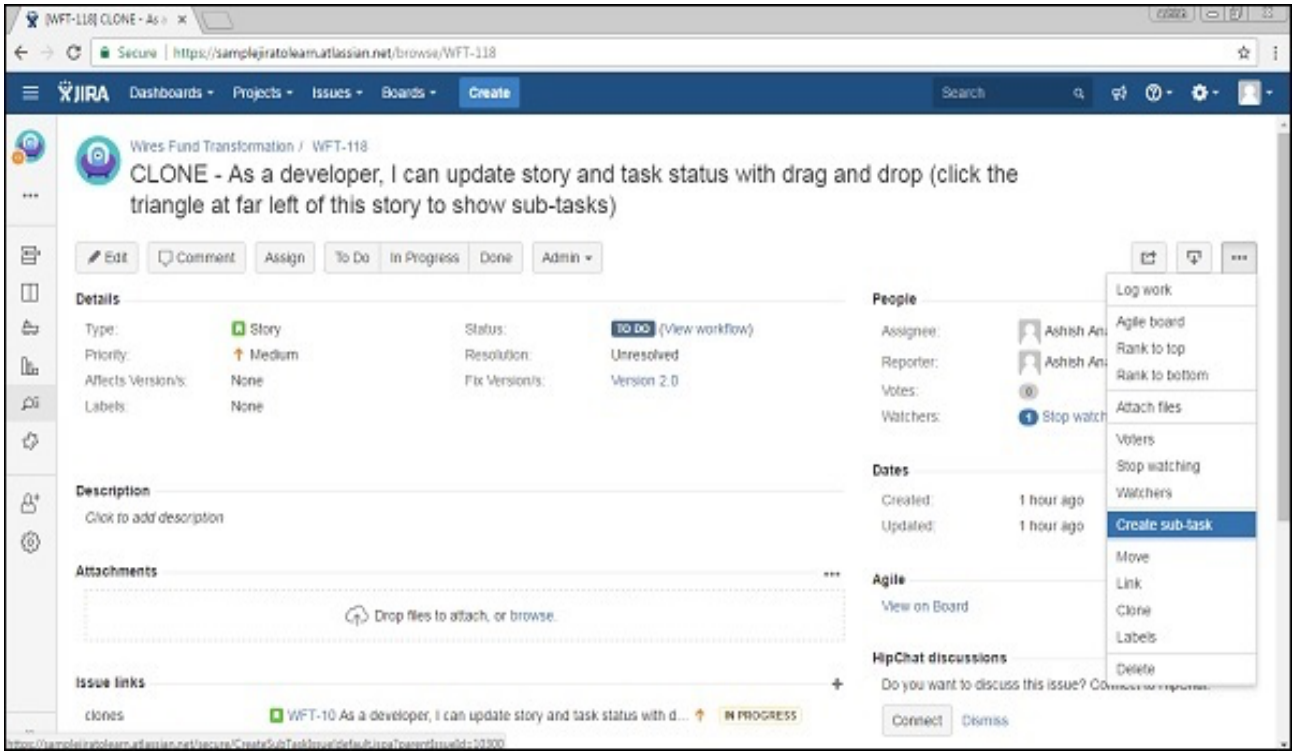
JIRA - subtask creation

- An issue can be completed by performing many tasks with different persons like Dev, QA, UAT, Business, Support, etc.
- To track the progress in each department, sub-tasks are created in an issue and assigned to the concern person. Once all the sub-tasks are resolved, the issue can be marked as completed.
- The following points are to be considered when creating a subtask.
  - All subtasks are an essential part of their parent issue.
  - All subtasks are visible on the main screen of the parent issue.
  - Subtasks always belong to the same project as their parent issue.
  - Subtask has all fields that are present in the standard issue.
  - Subtasks cannot have a subtask of their own.

Managing and Tracking Development Processes

JIRA - subtask creation

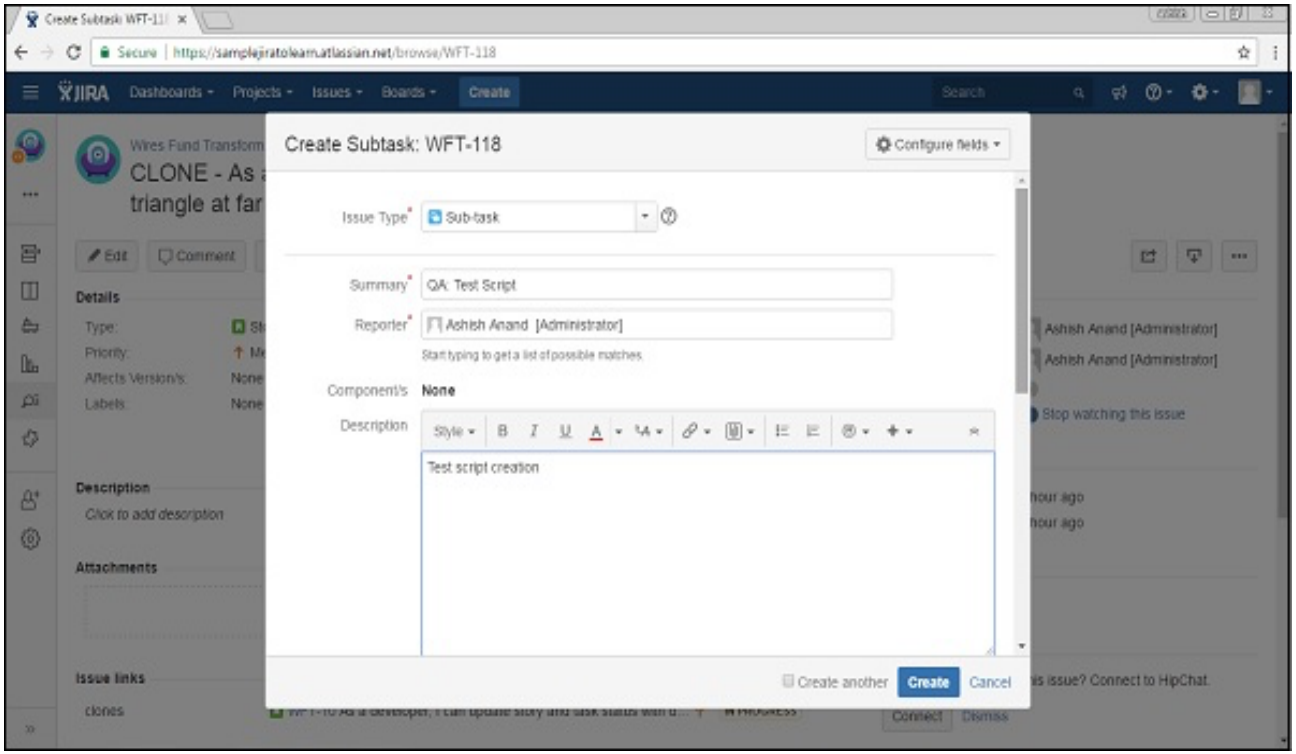
- To create a subtask, open an issue where subtasks should be created, and select More → Create Subtask. A Create subtask page will display.



Managing and Tracking Development Processes

JIRA - subtask creation

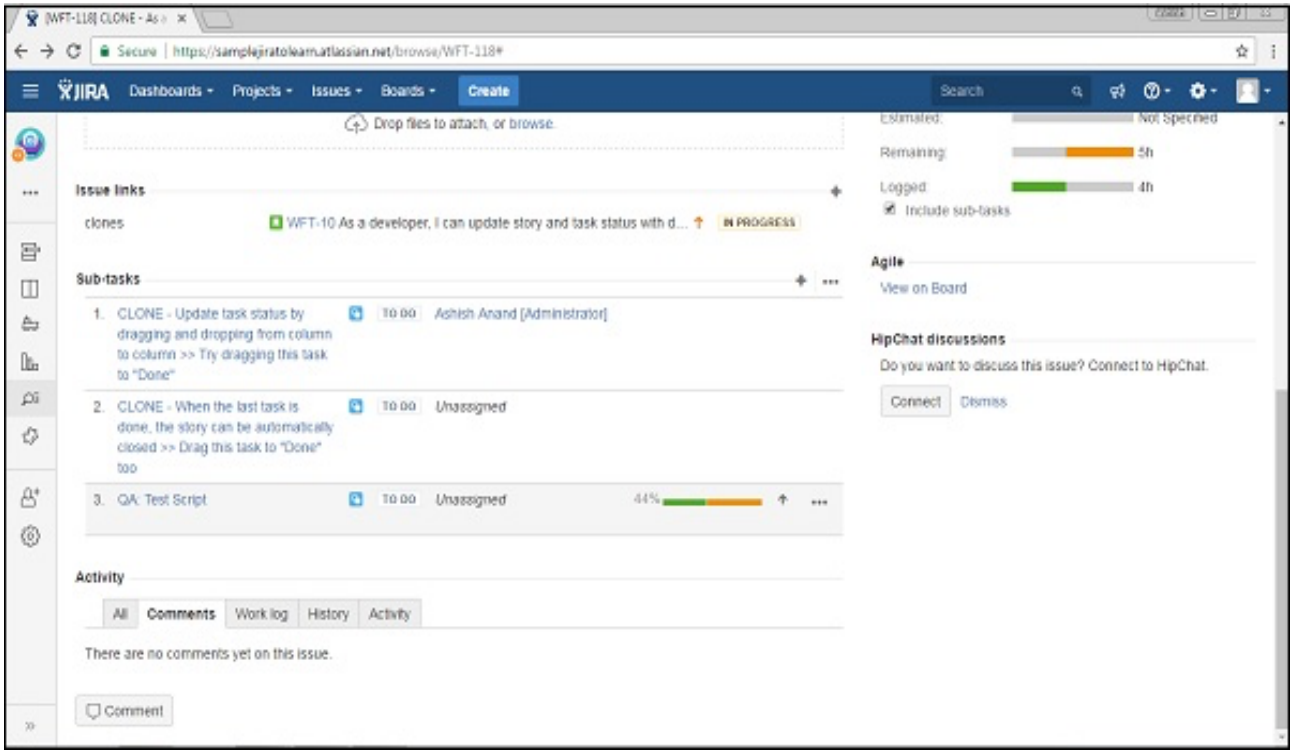
- Enter the required fields and then click on Create. This screenshot shows the Create Subtask Form with required and optional fields.



Managing and Tracking Development Processes

JIRA - subtask creation

- Once created, a subtask is on the **Main Issue page** → **Subtask** section. This screenshot shows how to view a subtask in an issue.



Managing and Tracking Development Processes

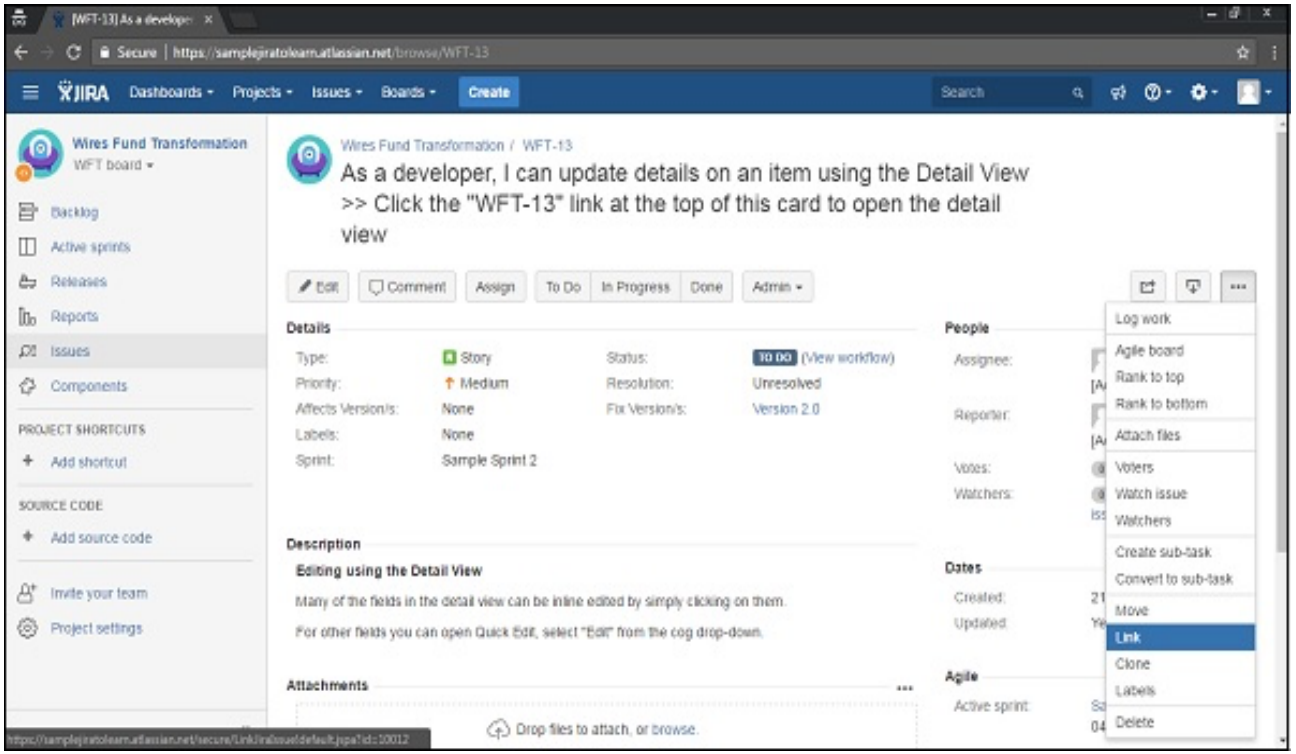
JIRA - issue link creation

- A link is useful when there is a dependency between two issues, such as if the two issues are duplicates, there is a up/downstream dependency, or one issue is blocking another issue.
- Links can have types, like “blocks” that record the purpose of the link.

Managing and Tracking Development Processes

JIRA - issue link creation

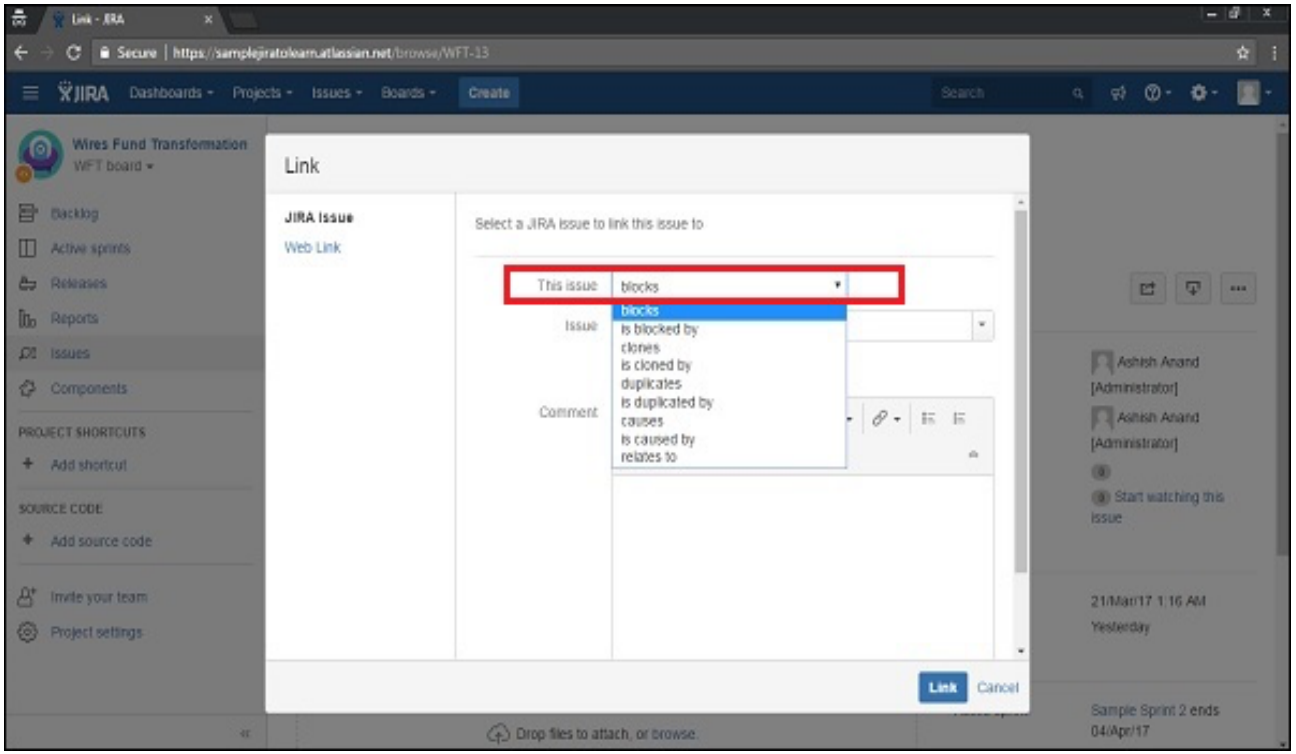
- Navigate to an issue, select **View Issue Page**, then select **More** → **Link** to display link dialogue page.



Managing and Tracking Development Processes

JIRA - issue link creation

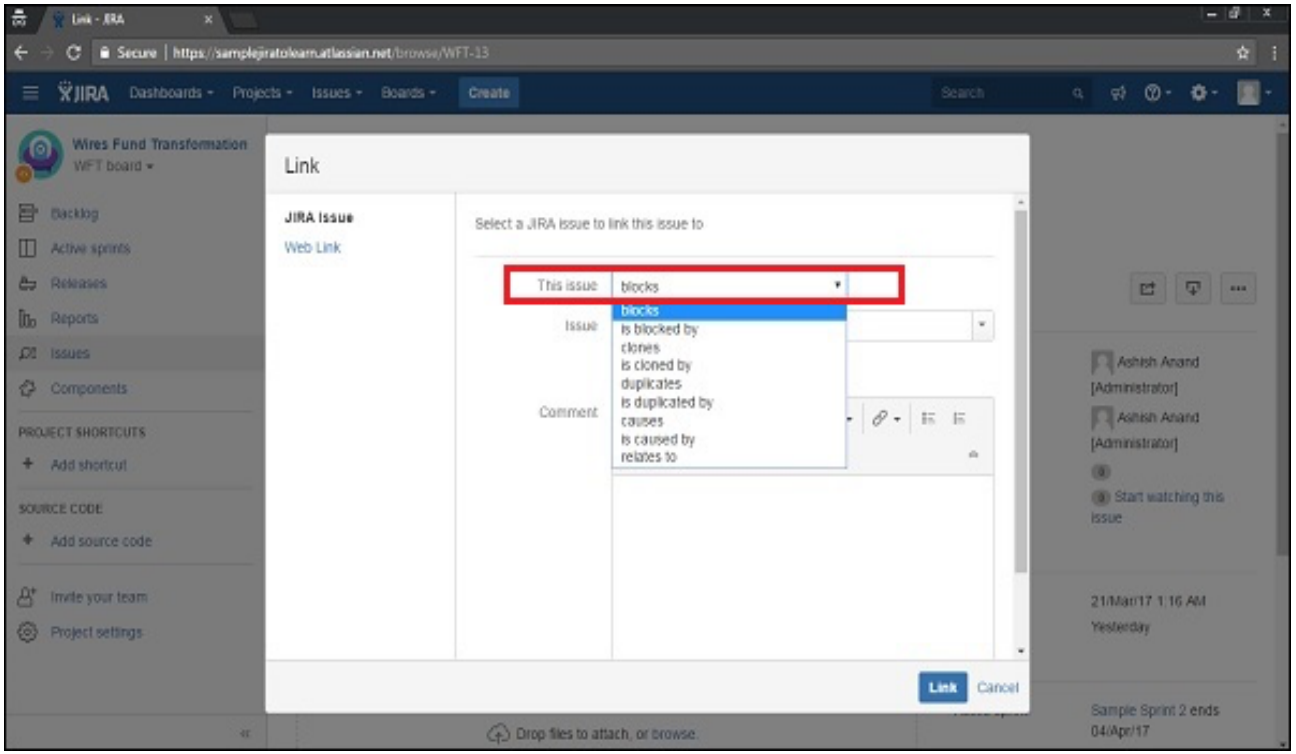
- Select **JIRA Issue** item in the left side of the dialogue and select “This issue” field from the dropdown list.



Managing and Tracking Development Processes

JIRA - issue link creation

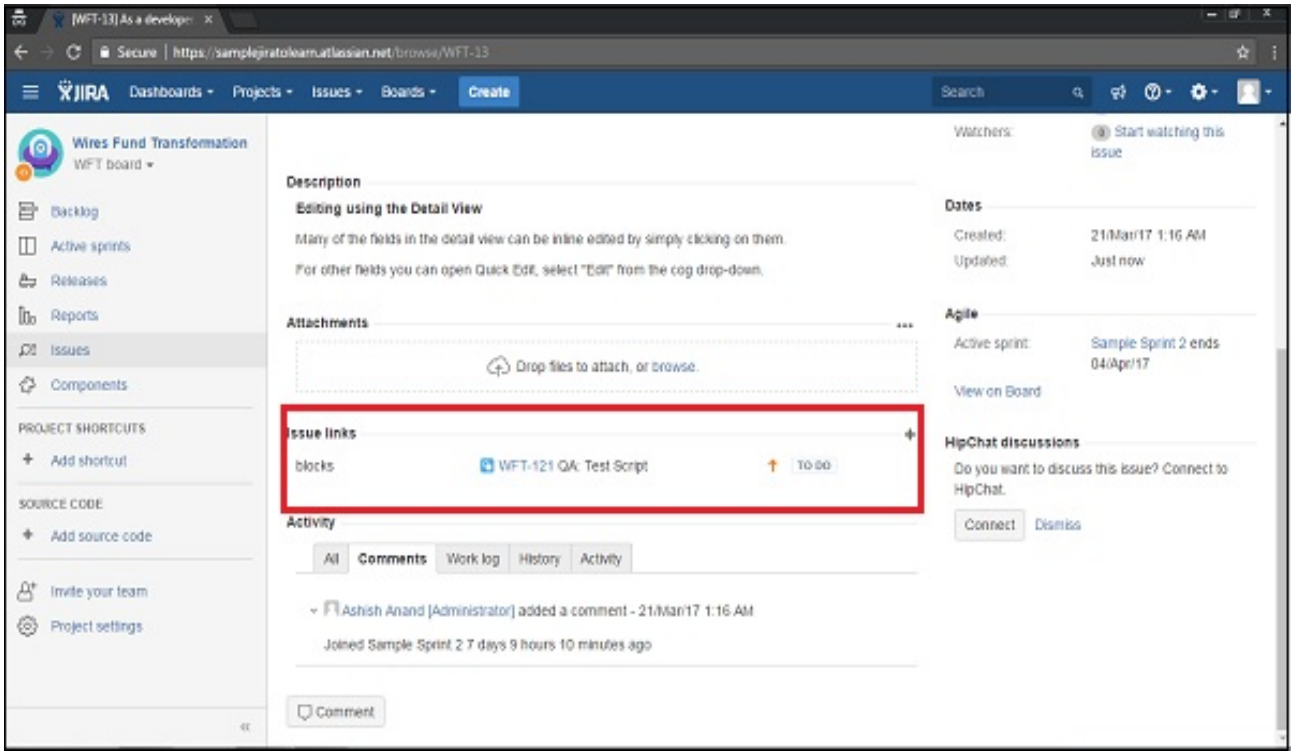
- The next step is to select/search issue. This screenshot shows how to add an issue as a link in another issue.



Managing and Tracking Development Processes

JIRA - issue link creation

- Adding a comment is optional. Then Click on Link. This screenshot shows how to view linked issue in parent issue:



Managing and Tracking Development Processes

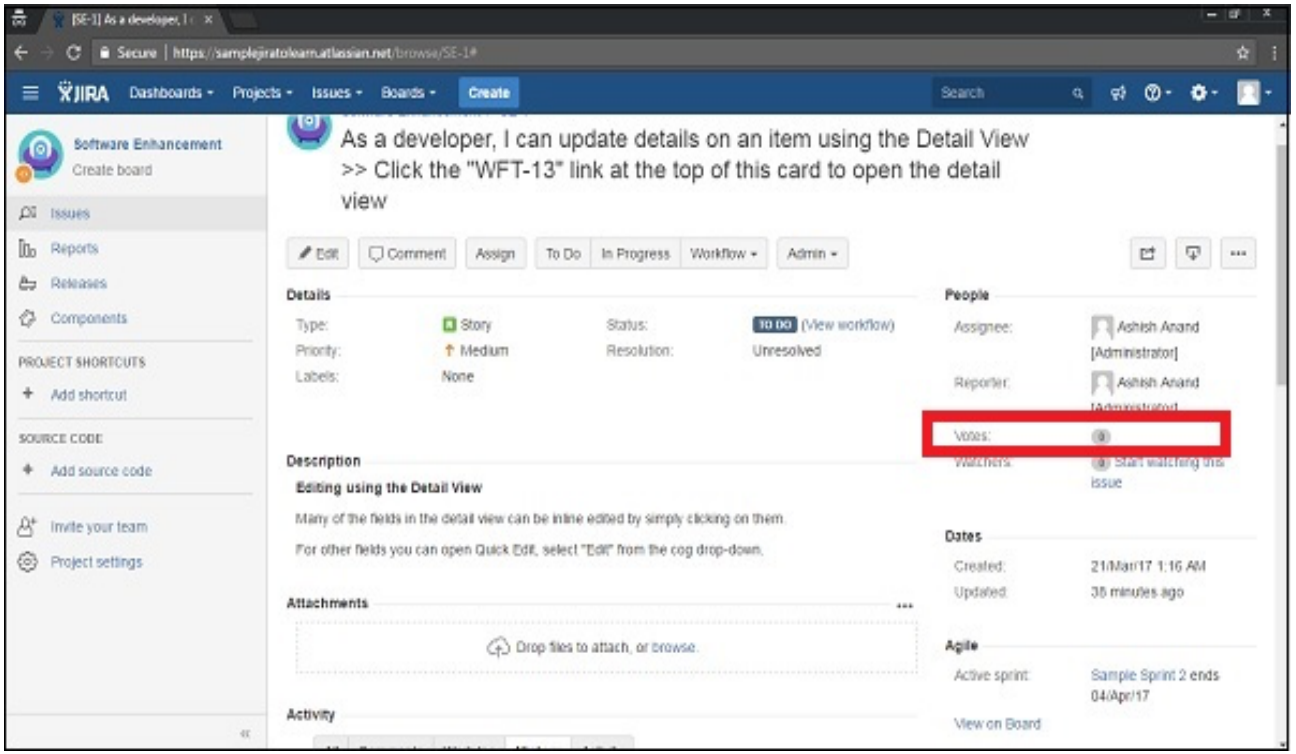
JIRA - issue voting

- JIRA has a feature called Vote to indicate a preference for the issue, whether resolved or completed.
  - Admin needs to add permission for users to use this feature.
  - Go to the View Issue page where voting is required.
  - Click Vote for this issue to instantly vote for that issue.

Managing and Tracking Development Processes

JIRA - issue voting

- The “Votes field” is on issue detail page. The user does not have permission to vote here, so cannot view the link beside Votes field.



Managing and Tracking Development Processes

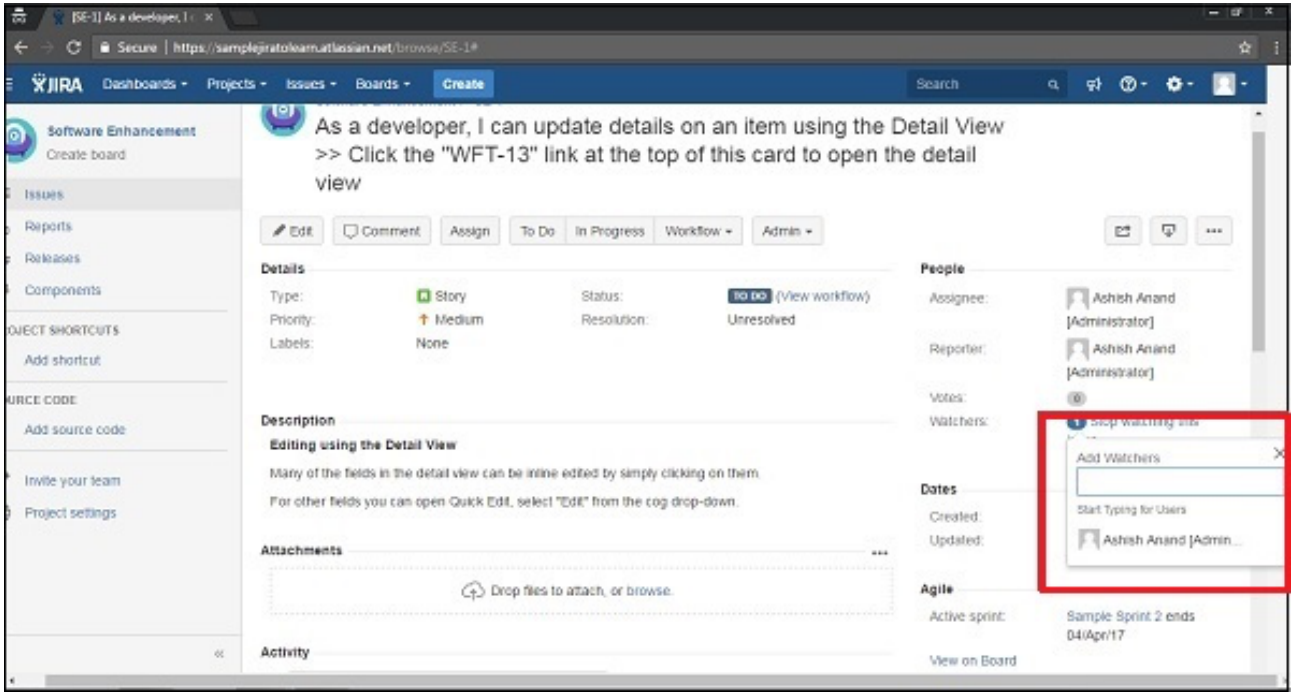
JIRA - issue watching

- JIRA has a feature that allows users/stakeholders to keep an eye on an issue, whenever there is any update or changes in that issue.
- If there is any for the issue, notifications/emails are sent to those who are listed as watchers for that specific issue.
  - Admin needs to add permission for users to use this feature.
  - Go to the view issue page that the user wants to watch.
  - Select the “Start Watching this issue” link to be listed as watchers.

Managing and Tracking Development Processes

JIRA - issue watching

- The user can click on 1, to allow adding other users as watchers. Start typing the user name in the “Add Watchers” box and select from the autocomplete box. The user will be listed as a watcher.



Managing and Tracking Development Processes

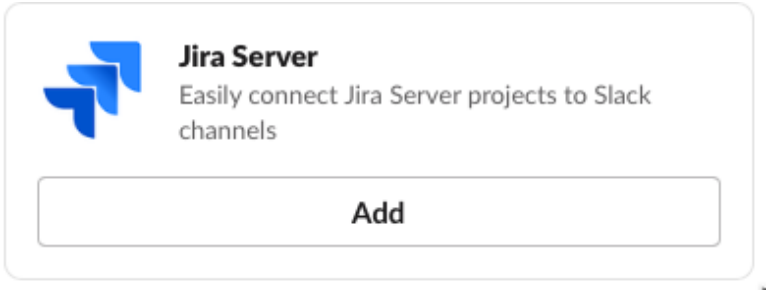
JIRA - integration with SLACK

- The JIRA-SLACK integration is a SLACK app. According to the Atlassian marketplace, the integration includes these features.
  - Customize Slack notification messages and group notifications with Slack threads
  - Instantly create new Slack Channels linked to your Jira issues without ever leaving Jira
  - All comments and events associated with your issue will be automatically shared in your Slack channel
  - Trigger notifications with more than 20 events and fine tune with JQL
  - Monitor an entire project or an entire instance
  - Send to Slack channels, private channels, and direct messages
  - Filter notifications by priority, component, resolution, labels, status, type, assignee, reporter, sprint
  - All notifications provide detailed info, including user, event, field names, updated data, and more
  - Enables new Slack slash commands to get, create, and transition Jira issues
  - Works with Jira Software, Jira Service Desk, and Jira Core

Managing and Tracking Development Processes

JIRA - integration with SLACK

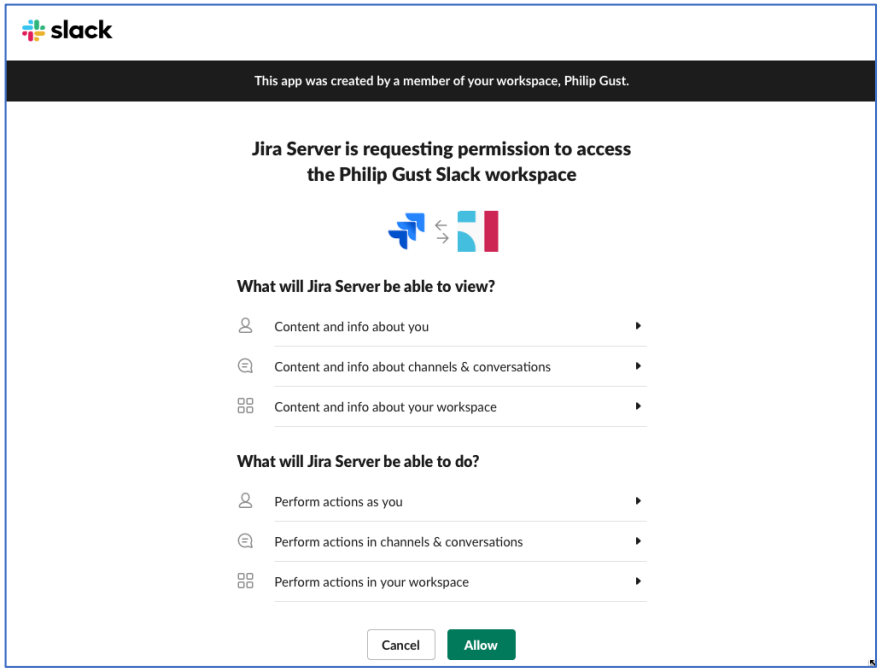
- To install the app, select the **Add Applications** button on the workspace screen. In the search box, enter “JIRA” and select “JIRA Server” from the results list.



Managing and Tracking Development Processes

JIRA - integration with SLACK

- Next, the app will ask for permission to access the SLACK workspace. This is because Jira will be sending information to SLACK and perform actions as the SLAC user. Click Allow.



Managing and Tracking Development Processes

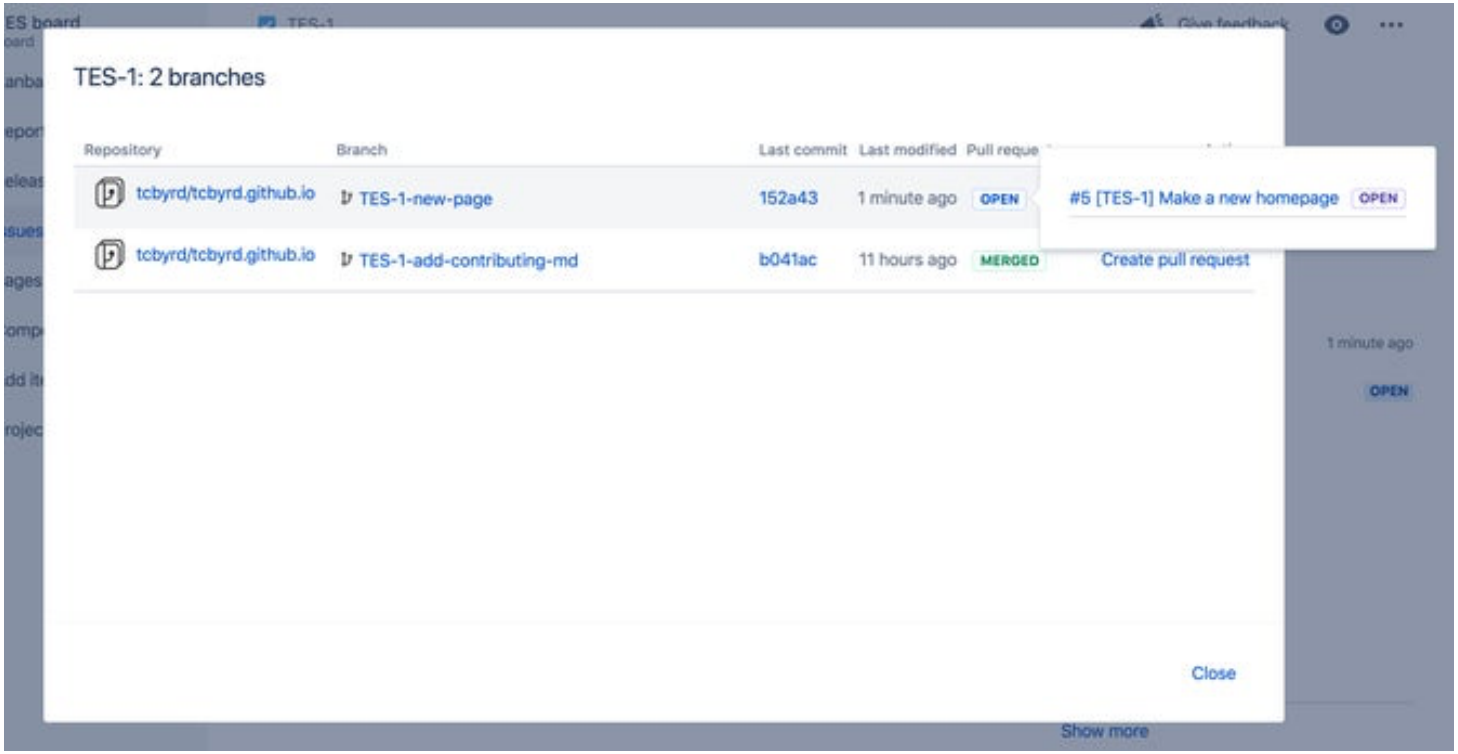
JIRA - integration with GitHub

- The GitHub JIRA+GitHub app includes these features.
  - Connect one or more GitHub organizations to a JIRA site and select specific repositories.
  - Use Smart Commits syntax to connect GitHub and JIRA together
  - Get updates about repository operations in JIRA issues for pull requests, commits, and branches
  - Use commands in GitHub commit messages to perform actions within JIRA like closing an issue, adding a comment, updating time tracking info, transition workflow states
  - Show JIRA information in GitHub by getting references to JIRA in GitHub issues and pull requests

Managing and Tracking Development Processes

JIRA - integration with GitHub

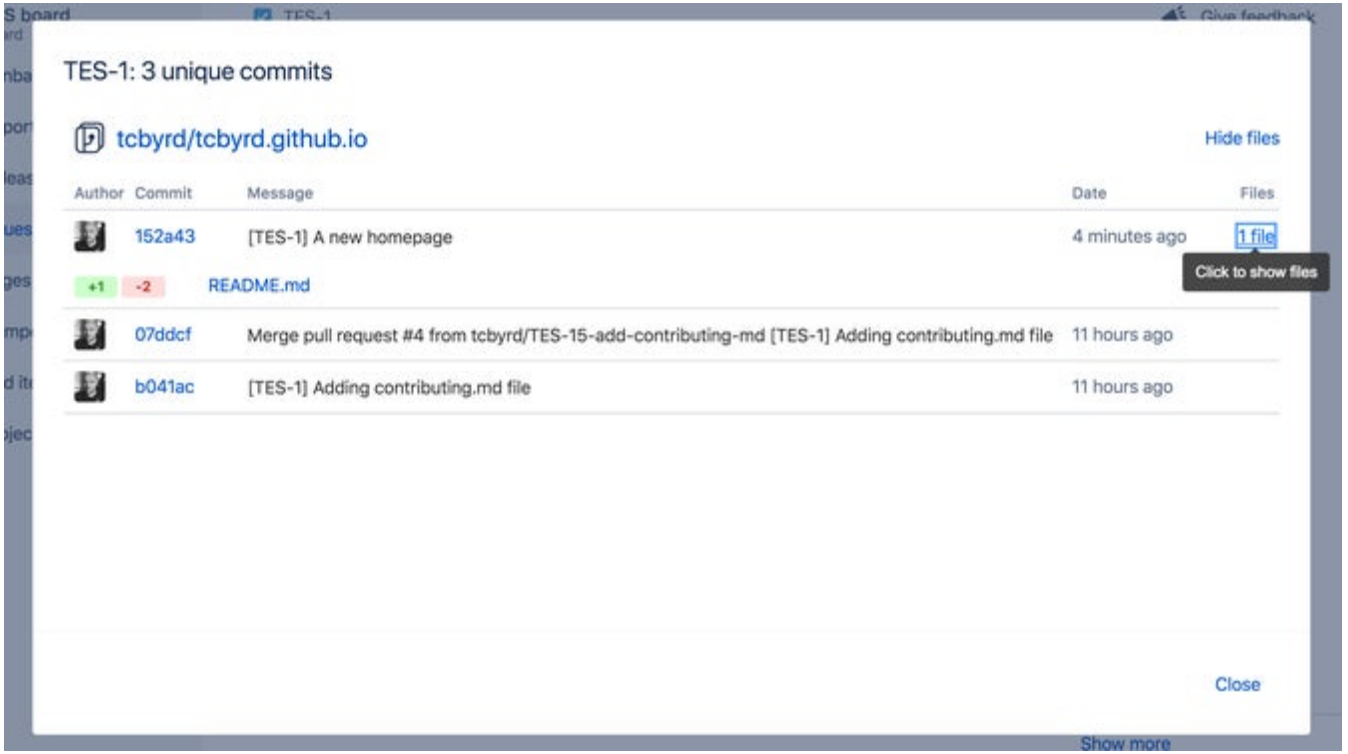
- Example JIRA screens showing GitHub actions



Managing and Tracking Development Processes

JIRA - integration with GitHub

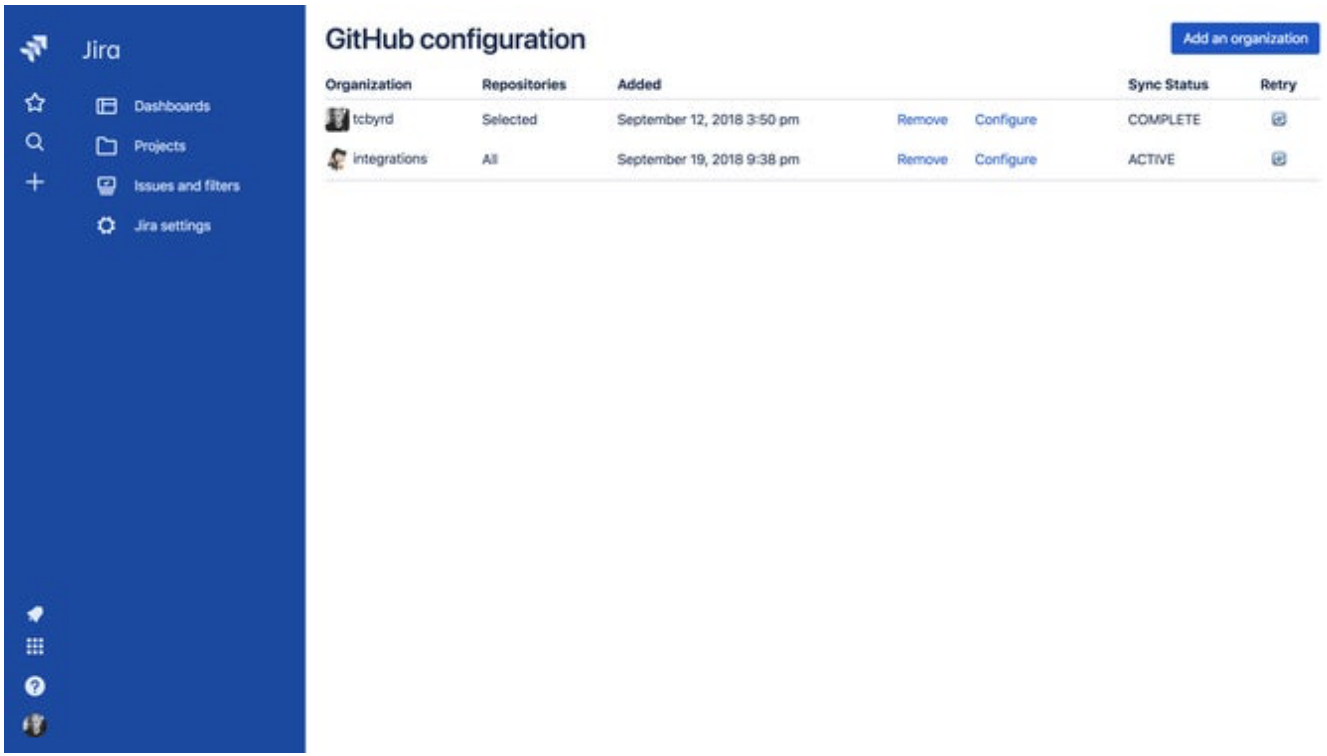
- Example JIRA screens showing GitHub actions



Managing and Tracking Development Processes

JIRA - integration with GitHub

- Example JIRA screens showing GitHub actions



Managing and Tracking Development Processes

JIRA – Conclusion

- JIRA offers advanced features for process integration by enabling users to manage both the project and the development process itself.
- Integration with SLACK, GitHub, and other tools also supports integrating communications and critical content used by the project.
- Atlassian provides a free JIRA account for those who would like to explore its features in more depth. For details, visit

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