

Jira for CRM team



Why Jira?

Change is always difficult at the beginning. Time permits us to better accept changes. Migrating to JIRA is our first chance to be opened to the world of agility.

Shorter release cycle. Simplification on ticketing tools. Reknown and standards of the market. Standardisation of major platforms like Core CRM and iCARE. Continuous integration tools among developers. Structured way of doing tests. Better visibility of tests performed and the results.

This move is one step ahead for transformation. To better prepare us to take new challenges!

YES WE CAN! 😊

Table of contents

- Why Jira?
- 1.Lost in JIRA as I used to work thay way...tell me more!
 - 2.1 Requirements /Story workflow
 - 2.2 Defects /Bug workflow
 - 2.3 Detailed activites in the issue lifecycle
- 3.0 Step-by-step guidelines

Useful links:

- [Link to ppt](#)

1.Lost in JIRA as I used to work thay way...tell me more!

Used to be in PMO...	...Going to be in Jira
Requirements	Story (issue type)
Defects	Bug (issue type)
Test scripts	Test (issue type)
SFDC process	Epic (issue type). Remember to assign issue to an Epic!
Release	Fix version (field). In order to assign issue to a release, first needs to create a release record by Change Manager
Urgent change - quick and small hot fixes that are pushed to production quickly	Create as either Story or Bug. Each move-to-production has to be a release. Urgent change is tied to a minor release

A Backlog is a list of issues in the pipeline, waiting to be qualified and assigned to a sprint.

Backlogs

Backlog

CEP board

Quick filters ▾ Assignee ▾

TEST View ▾ ...

3d 3h 2w 2d 3h 30m 0 Plan sprint ▾ ...

Sprint 0 - PROTOTYPE 19 issues
Sprint 0 preparation for a PROTOTYPE release
07/Mar/18 5:43 PM - 30/Mar/18 6:43 PM

Mouse roll over to see number of issues owned by each member

Quick indicator to show those still in TODO (grey), In progress (blue), done (green)

Workload estimation

User stories linked to EPIC (process)

Issue	Category	Assignee	Estimate
As a distributor (GPM: Global Product Manager), I want to view business opportunities created by Solvay distribution manager in the CRI	Opportunity Manageme...	CEP-9	2d 4h
Distributor has an access to the list of opportunities (open or closed)	Opportunity Manageme...	CEP-13	5h
Distributor views the detail of an opportunity	Opportunity Manageme...	CEP-14	30m
As a distributor, I report the advancement on the shared opportunities with Solvay through the chatter	Opportunity Manageme...	CEP-10	30m
As a distributor, I am notified when there is a new message in the chatter	Opportunity Manageme...	CEP-11	1h 30m
As a distributor, I am notified when the opportunity is edited	Opportunity Manageme...	CEP-12	1h
Direct hyperlink to solvay-eCommerce for reordering	Security	CEP-67	1h
Analyse as-is customer notion	Security	CEP-68	1d
Functional design on security model	Security	CEP-82	1d
Functional design on data sharing model	Security	CEP-56	1h
As a distributor, I want to be able to log in so that I can access secured areas	Security	CEP-58	1d
As an administrator, I need to grant user permission/configuration to each registered user	Sample Request Mana...	CEP-59	1d
Analyse as-is sample request process	Sample Request Mana...	CEP-60	1d 6h
Functional design on sample request management			

A Sprint can be planned ahead and issues can be assigned to planned sprints.

At any time, you cannot have 2 sprints running at the same time.

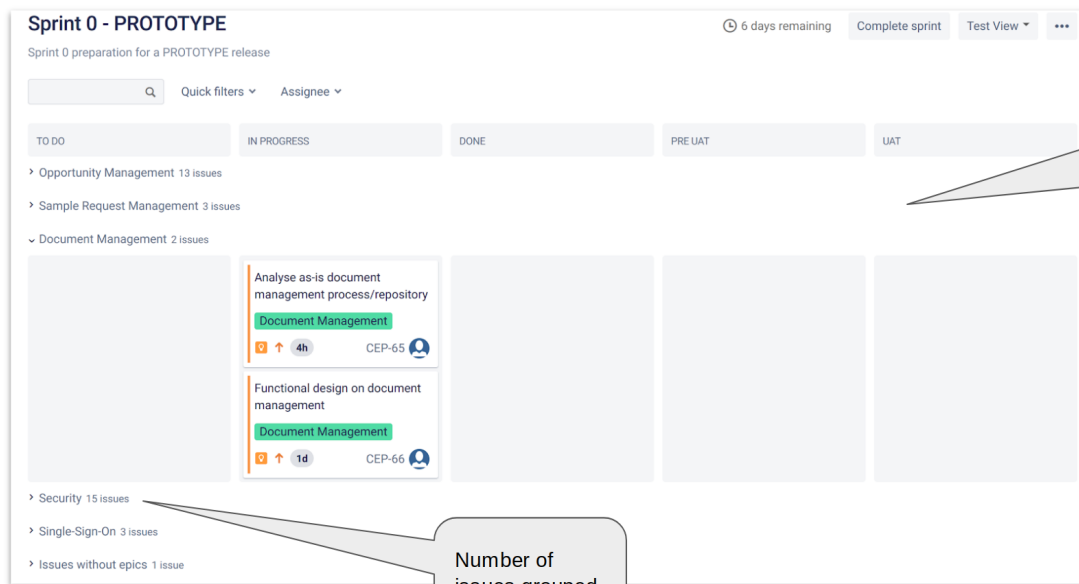
You need to complete a sprint before to start another one.

Backlog 22 Issues Create sprint

Issue	Category	Assignee	Estimate
Aroma: New rules for Customer request	P9 - Customer Request ...	CCCME-977	-
Community: Doc mgmt - PDF rendition	P24 - Community	CCCME-949	-
Community: Doc mgmt - TDS	P24 - Community	CCCME-913	-
Add the Lead creation from the SFDC panel in gmail		CCCME-972	-
Automatize Qlikview Access	PC - Security Model	CCCME-971	-
Soda Ash: Business rules & queue members review	FALL '18 RELEASE - R... P - Cross Processes	CCCME-934	-
Creation of new roles for Peroxide GBU	PC - Security Model	CCCME-952	-
Add ability to edit closed complaints to Quality Manager	P6 - Complaint Manage...	CCCME-953	3h
Transactionan pricing approval	P14 - Transactional Pri...	CCCME-954	-
Improve send e-mail internally for visit report	P18 - Visit Report Mana...	CCCME-955	-
Novecare SFDC CS Roles structure amendment	PC - Security Model	CCCME-956	-
Home Page - LEX Rework	P - Cross Processes	CCCME-964	-
Contact Language picklist	P1 - Account and Conta...	CCCME-965	1m
Add "Add default team" button on Account	P1 - Account and Conta...	CCCME-966	4h

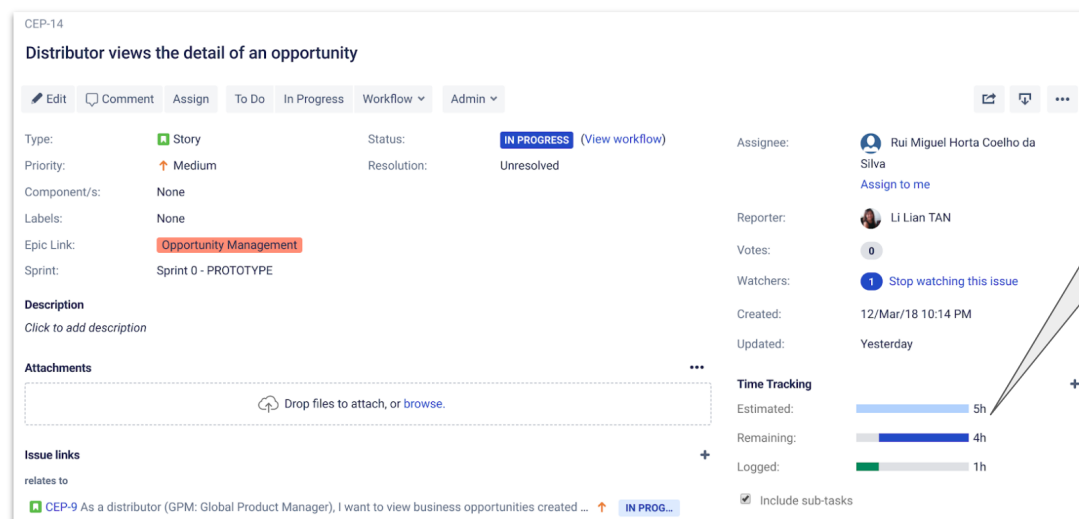
An active sprint is the current running sprint.

Active sprint - board review (daily scrum)



For an active sprint (or release), daily scrum to see each tasks move from one status to another

Workload estimation and time logged



A Test is an issue type. When you create a Test, it has other properties of a Test Case.

It lists down the Test Step, Test Data required to execute that Test Step, Expected result of the Test Step.

You can only include an attachment to a step should it require graphical verification.

Test (issue type)

The screenshot shows a 'User story test case' interface. It includes a header with tabs like 'Edit', 'Comment', 'Assign', 'To Do', 'In Progress', 'Workflow', 'Admin', 'Execute', and 'Add to Test Cycle(s)'. The main content area displays details for a test case, including its type ('Test'), priority ('Medium'), status ('TO DO'), resolution ('Unresolved'), and assignee ('Li Lian TAN'). A 'Description' section is present with a 'Click to add description' link. Below this is a 'Test Details' table with columns for 'Test Step', 'Test Data', 'Expected Result', 'Attachments', and 'Action'. The table lists two test steps: '1. Login as a partner community' and '2. Enter username and password. Click on login'. A callout points to the 'Assign to versions' link. Another callout points to the 'List of test steps defined for a test case' table. A third callout points to the 'Add' button at the bottom right of the table.

Assign to versions

List of test steps defined for a test case

Easily add new test steps whenever requirement/user story evolve

Test cycle contains a list of all the Test issues that need to be executed.

Test cycle can be easily created and then clone, for example, the same test cycle cloned for Devpro and UAT.

For a given Test cycle, we have a good visibility of all tests executed and it's percentage of passes/fails.

Test cycle view

The screenshot shows a 'Test cycle view' interface. It includes a sidebar with a 'test project' dropdown and a 'Reports' section. The main content area displays a 'Cycle Summary' for 'UAT User testing'. It includes a search bar, a tree view of versions, and a table of test cases. A callout points to the 'Test cycle = test session created' text. Another callout points to the 'Test cycle contains list of test cases. A view of all test cases ran, showing defect, and status of tests.' text. A third callout points to the 'Tester click on "E" for test case execution' text.

Test cycle = test session created

Test cycle contains list of test cases. A view of all test cases ran, showing defect, and status of tests.

Tester click on "E" for test case execution

Tester: Execution view

test project \ Fall '18 release \ UAT User testing \ TP-12

Return to Test
Return to Test Cycle

User story test case

Description

Test Execution

Execution Status : UNEXECUTED

Assigned To : Add assignee

Defects : Enter defects

Comment : Enter comment

Testers execute each test step and provide status.
Enter defects, comments or attachments

Attachments (Execution)

Test Details

Test Step :	Test Data :	Test Result	Attachments	Comment :	Attachments (Execution)	Defects	
1	1. Login as a partner community user	Use "Test Partner" user	Login page is displayed	UNEXECUTED	Enter comment	+	Enter defects
2	2. Enter username and password. Click on login	When login, arrives in homepage	UNEXECUTED	Enter comment	+	Enter defects	

Transition

Allows you to move from one status to another

CCCME-898

Community: Customer Request

Edit Comment Assign Estimation completed Close issue Admin

Type: Story

Affects Version/s: None

Component/s: None

Labels: cep

Epic Link: P9 - Customer Request Management

Sprint: Fall'18 release

Status: UNDER REVIEW (View workflow)

Resolution: Unresolved

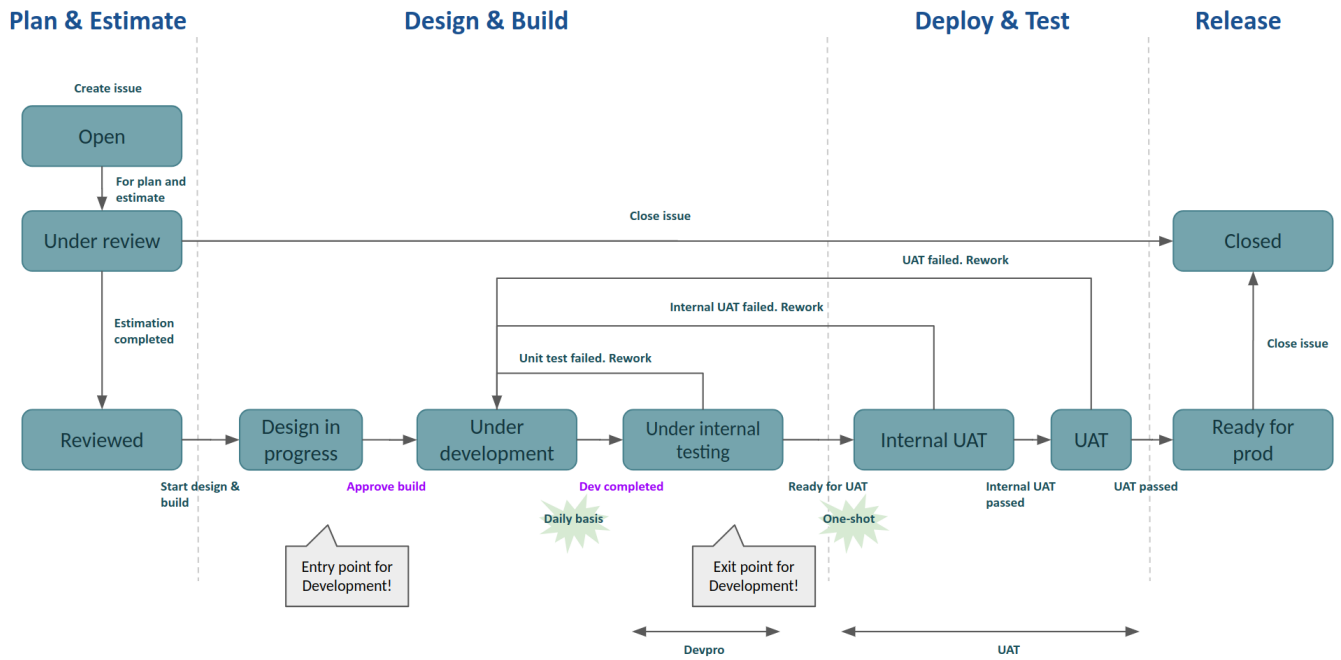
Fix Version/s: Fall '18 Release

Workflow Status

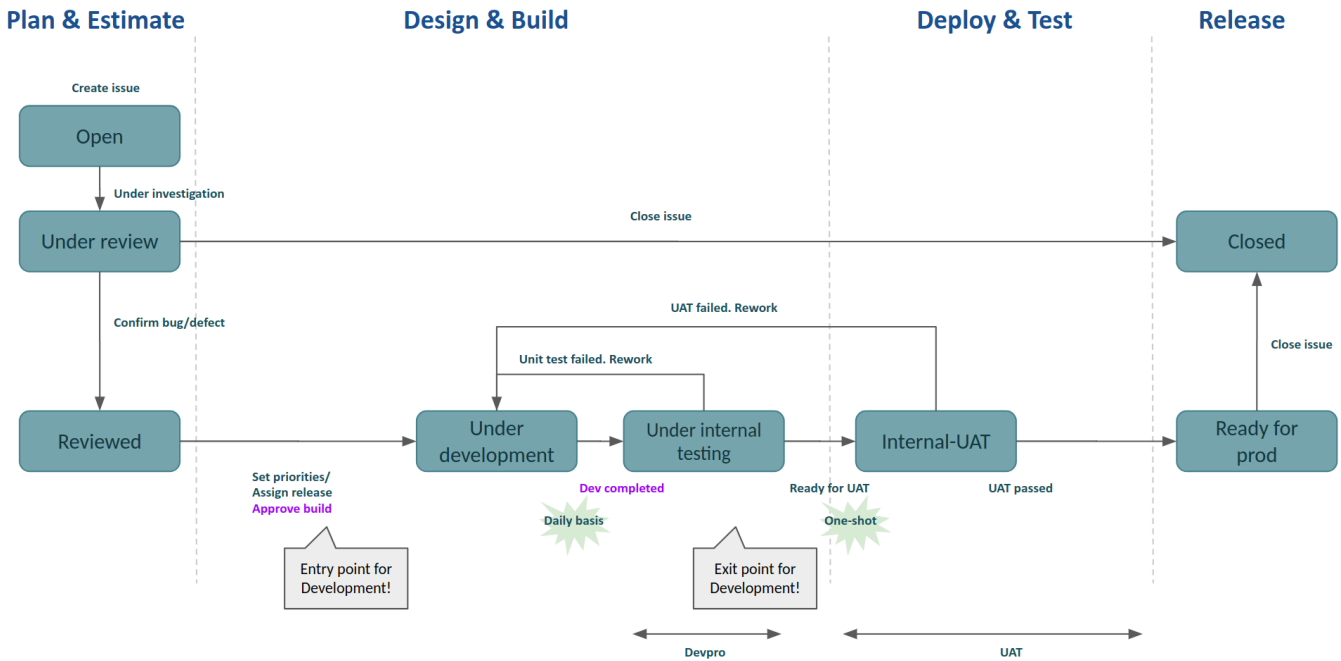
At which status of the release cycle does the issue belongs to

2.0 Workflow

2.1 Requirements/Story workflow



2.2 Defects/Bug workflow



2.3 Detailed activities in the issue lifecycle

Plan & Estimate

Anybody can quick-create issue.
Update and better describe issue by Func Lead.
High-level estimation by Developer.
Agreed by Func Lead.
Once OK, Func Lead set to "Reviewed" for release planning.

Design & Build

Change manager assigns for a release.
Detailed description by Func Lead if required.
Create Test issue/associate Test to Story.
Detailed solution by Developer. Submit for review.
Tech Lead approve build.
Developer develops.
Once development completed, notify Tech Lead.
Code review in Bitbucket by Tech Lead.
Security review in Bitbucket by Security.
Once OK, Tech lead set to "Under internal testing".
Assign to Deployment manager.
Deploy in DEVPRO.
Func Lead and Developer perform early testing in Devpro.
Rework issues if any.
Change manager prepares Test Cycle (test scope for the release) and associate Tests to the Test Cycle.

Deploy & Test

Based on Test Cycle, IS testers from Adoption execute Tests.
Logs pass/fail in Tests.
Once all completed, IS testers notify Change manager.
Test review by Change Manager, Func Lead and Developers.
Rework what is major. Create new bug/defect to be treated if minor.
Adoption Lead schedules meeting with Key Users for tests. Explains upcoming changes.
Based on same planned Test Cycle, key users perform test in UAT.
Logs pass/fail.
Those pass, set status to Ready for prod.

Release

On release day, Deployment manager deploys to PROD.
Change manager/Adoption Lead closes issues.
Communicate release note to Key users/others



3.0 Step-by-step guidelines

Plan & Estimate



- **Create** Story issue

- Functional description of the issue
- Link to Process (epic)
- Add labels, components
- **Assign to developer** for high level estimation

- Check Original estimate done by developer
- If OK, **change status to "Reviewed"**

Design & Build



- Once assigned by Change Mgr and a release, complete more **detailed description** of the issue if required
- Notify developer for detailed solution
- **Create or associate** a Test issue to the Story

Deploy & Test



- **Participate** in Test Review session to acknowledge rework

Release



Plan & Estimate



- Understand functional description by Func Lead
- Perform high level estimation
- **Original estimate**
- **Notify Func Lead for review**

Design & Build



- Once assigned by Change Mgr and a release, complete more **detailed solution** of the issue if required
- **Assign to Tech Lead** for build approval

- **Develops**
- **Bitbucket** activities
- Code review in Bitbucket by Tech Lead

- Once development completed, **perform early testing** in Devpro
- Review and rework with Func Lead

Deploy & Test

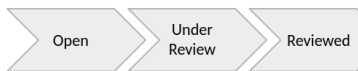


- **Participate** in Test Review session to acknowledge rework

Release



Plan & Estimate



Design & Build



- **Security review** in Bitbucket

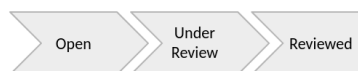
Deploy & Test



Release



Plan & Estimate



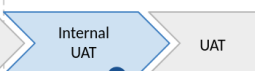
Design & Build



- **Participate** in release planning with Change Mgr and Adoption Lead
- Quick review of **detailed solution** proposed by Developer
- **Approve build**

- **Bitbucket** activities
- **Code review** in Bitbucket
- **Change to "Under internal testing"**

Deploy & Test



- **Participate** in Test Review session to acknowledge rework

Release

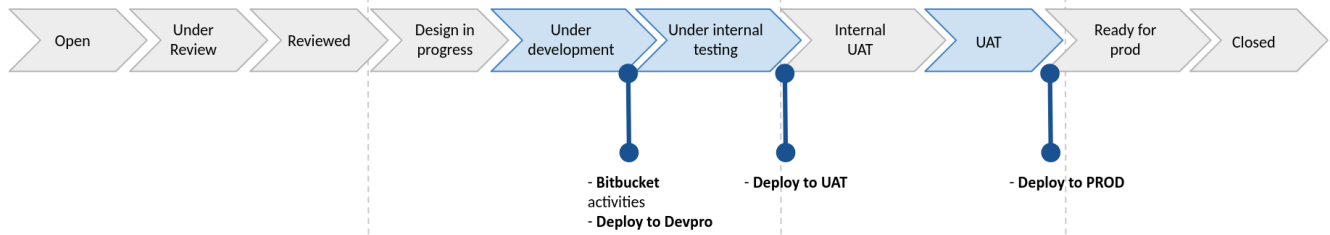


Plan & Estimate

Design & Build

Deploy & Test

Release

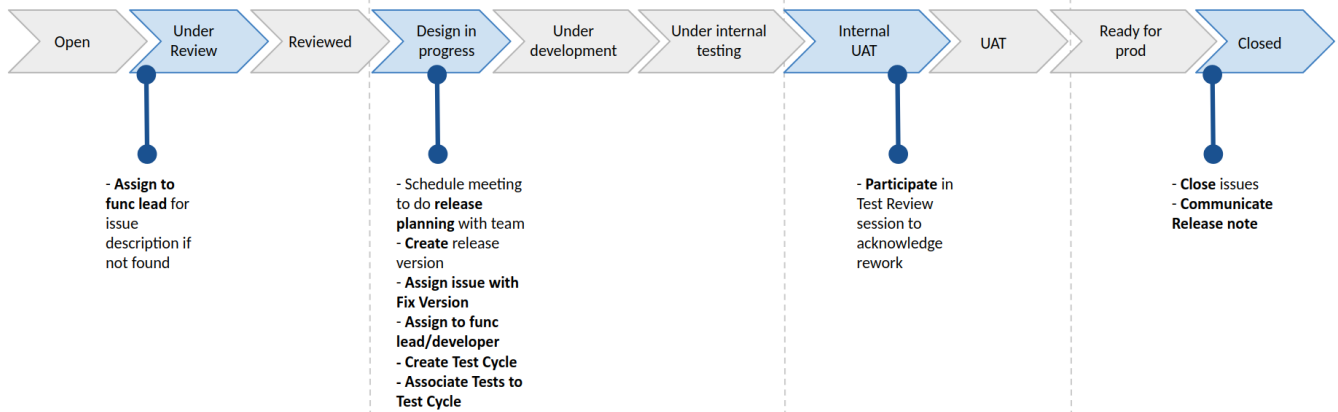


Plan & Estimate

Design & Build

Deploy & Test

Release



Plan & Estimate

Design & Build

Deploy & Test

Release

