Process

click "Create" → "New"

Issue is in a draft state while it's written and defines a new or updated function or bug fix. Business logic and designs are finalized.

Issue will need sufficient information to convey the work that needs to be done so that it can be evaluated and prioritized.

click "Select for Dev" → "TODO"

An issue is determined to be next is line to be coded and have sufficient information to be worked on. It awaits assignment.

Issue has been vetted and determined to be work that needs to be completed as soon as possible. It must be added to a sprint before work can begin and assigned to a developer.

click "Start Progress" → "In Progress"

An issue is being worked on actively. Any changes to the issue or new information should be brought to the coders attention.

When code is completed a pull request will be made. Reviewers are assigned to code review the issue.

click "Sent to Review" → "Code Review"

Completed code has been put into a PR and is being reviewed by at least one other coder.

After code is reviewed the code gets merged and auto-deployed to the Development environment. Make sure build comletes before proceeding.

click "Move to Dev QA" → "Ready for QA"

Completed code is now live in Development and can be viewed by QA for the first time.

The acceptance criteria of the issue determines what must pass for this issue to be verified. Otherwise it's confirmed to be doing no harm and working.

click "Verified in Dev" → "Verified in Dev"

Code that on Development has been greenlighted to be deployed to Staging. Note this should be done as a release not piecemeal.

Issues must be grouped into a release and all moved into the next state together.

Verified Items released to Stage

A release is created from tickets verifed on DEV. This is then pushed to the Staging Environement.

click "Deployed to Stage" → "Deployed to Stage"

Code is now on Staging as a mock release and ready for a regression test of the environment to prepare for a release to Production.

Issues must pass all acceptance criteria and a regression test of the site must find no issues that can not be hidden by feature flagging.

click "Ready for prod" → "Ready for production"

Issue has been marked and viable for release to Production and it can now be added to a release.

When all the issues that released to stage are now ready for production and release can happen and then all those issue move to the next status.

Verified Items released to Prod

A release is created from mock release on Stage. This is then pushed to Production

click "Deployed to prod" → "Deployed to Prod"

Issues are now live and the app is smoke tested.

When the team feels confident the release is good with no immediate issue that need to be addressed the issue can be closed.

click "close" → "closed"

The issue is now archived, no further work is expected.

JIRA STATUS

Dev Role | QA Role

Backlog

--- Sprint --->

--- QA State --->

Create

New

Backlog (todo)

TODO

Added to sprint

Reopened

reopen -->

Start Progress

In Progress

send to review

Code Review

Move to Dev QA

Ready for QA

Verified in Dev

Verified in Dev

Deployed to Stage

Deployed to Staging

Ready for Production

Ready for Production

Deployed to Prod

Deployed to Prod

QA test fail

Failed

Close

Closed

Code

Local

Pull Request

Dev Branch

merge and autodeploy

Dev Main

Deployed to stage

Staging

Deploy to Prod

Production