Scrum live project

Document 1: Definition of Done:

A Definition of Done is a clear and concise list of requirements that software must adhere to for the team to call it complete. While the DoD usually applies to all items in the backlog, acceptance criteria are applicable to a specific user story. In order to complete the story, both the DoD and acceptance criteria must be met.

The DoD starts with organizational standards that all teams must follow as a minimum. That infers that teams can add to the organizational DoD but cannot omit items from it. If multiple teams work on one product, they comply with the same DoD

Steps to a definition of done

Create a DoD in Jira

The best way to have a DoD inside Jira is to use a Custom Field. We can use the standard text field or checkboxes, but both have drawbacks – for example, text fields do not show which items are complete and checkboxes are only visible in edit mode.

 Break it down

Definitions of done correspond to different parts of the development process: technical tasks, user stories, and bugs. For example, a project could have this DoD:

 Code builds without warnings (technical task)

 Code unit tested (technical task)

 Documentation updated (user story)

 Build pushed to demo server (user story)

Make it Global

By using custom field options, you can create DoD Items that are applicable to any issue: old or new. Moreover, modifying, adding, or removing an option is immediately reflected in all your Jira issues.

Manage it over time-A Definition of Done is a live document that should be reviewed regularly. As your dev team strives to improve, you can make your practices more stringent over time. Rather than deleting or modifying options, simply disable them. Disabling an option will keep the option in Jira but prevents it from appearing on issues. This allows you to keep a record of your DoD over time. And if you really want to challenge the team, you can add more DoD items but make some of them mandatory and others optional.

Make the product owner responsible and the team accountable-

Set up the DoD custom field such that only the product owner can add/modify/remove items. This will make him or her responsible for stating what they want. Then, let the team be accountable for delivering what the PO wants by crossing off every item in the DoD.

 Enforce it:The best way to have an agile team follow the DoD is to embed it into their scrum workflow. Use a Workflow Validator on the technical task or a user story workflow transition to prevent the resolution of the issue until all DoD items are done. This demands accountability and reinforces what “Done” really means.

Create an acceptance criteria list in Jira: At the end of the day, the acceptance criteria list is nothing more than a DoD that is specific to every user story. To implement an acceptance criteria list in Jira, either create a new custom field or piggyback on the global DoD. With Checklist, you can add items directly at the issue level. So, you could have a single DoD custom field with the mandatory global items as options and each individual acceptance criterion as an item at the issue level.

**Checklist for DOD:**

* Produced code for presumed functionalities
* Assumptions of User Story met
* Project builds without errors
* Unit tests written and passing
* Project deployed on the test environment identical to production platform
* Tests on devices/browsers listed in the project assumptions passed
* Feature approved by UX designer
* QA performed & issues resolved
* Feature is tested against acceptance criteria
* Feature approved by Product Owner
* Refactoring completed
* Any configuration or build changes documented
* Documentation updated
* Peer Code Review performed