

Release Process Proposals

- [The purpose of the release process](#)
- [Prerequisites](#)
- [Release Process](#)
 - [Milestones goal and acceptance criteria](#)
- [Release Cadence, Life Time & Support](#)

The purpose of the release process

The Release Process is dedicated to creating a plan and then following and monitoring that plan to deliver an increment to the product. It is extremely important, especially in a community where there are lots of developers from different companies, countries, to have a common understanding of what and how we would like to achieve. The release plan will help us to coordinate our efforts to achieve agreed goals and to deliver proper, coded, tested, and working as an expected increment of product. To achieve that, we need to act in accordance with agreed rules, time schedule, and milestones.

Prerequisites

All who would like to attend community effort and contribution must follow bellow prerequisites:

- Read the [Getting Started as a Contributor page](#) (including Registration and User Creation, environment installation)
- Check/assign to one of the [existing projects](#) or create a [proposal for a new project](#)

Release Process

The unit piece of functionality/work which can be included in the release is represented by the Epic ticket in Jira. The release process consists of several Milestones in the pipeline of epics with clearly defined acceptance criteria. One can treat these acceptance criteria as a checklist that has to be followed to say that particular epic achieved this Milestone. Overall flow is outlined as follows:

1. [Invitation of proposals for features, enhancement requests](#) - approved till M0
2. Blueprints/Design/Architectural approval cut-off date (4 months before release date)
3. Code delivery cut-off date (2 months before release date) - till M4
4. Branch cut-off date (1 month before release date)
5. Stabilize the branch and build daily builds for 1 month
6. Release the stable build as Community Release
7. Post release notes to the docs.tungsten.io
8. Build weekly/bi-weekly if any change committed to the branch

Milestones goal and acceptance criteria

Milestone	Goal of the Milestone	Description of the Milestone - acceptance criteria
M0	Declaration of participation in the release, initial scope of the release	• Blueprint proposal represented as a Jira EPIC ticket • Feature lead assigned to Jira Epic ticket - declaration of participation in the simultaneous release • The initial version of Increment Blueprint stored on Github/tf-spec/Release folder, • The blueprint presented, discussed, and approved by related committers • Release Scope Overview updated accordingly • The initial version of the Release Plan for the project presented • Dependencies between projects/components/areas identified
M1	Identification of dependencies between modules and new functionalities, release plan for new functionality	• All dependencies between existing functionality and requested change/feature discussed and agreed by PTL and related approvers (based on Modules/Committers) • Jira Epic ticket broke down into stories in related Jira project (backlog, plan) - presented and discussed on TWS meeting • The final version of Blueprints, Release Scope and Release Plans for participating projects discussed and approved by approvers

M2	Feature/Functionality technical design freeze	• Technical design fully provided, agreed (documentation on the Confluence/Github/tf-spec available) • Jira Epic ticket updated with a link to technical design • Documentation started • Feature tests started • Feature/Functionality Freeze - Technical Design approved by relevant approvers (Modules/Committers)
M3	API freeze (beta available)	• External API available for beta-tests • Jira Epic ticket updated with information about API availability (documentation link - tf repo) • Documentation in progress • Tests in progress • API Freeze confirmed by Feature Lead (comment on Jira Epic ticket with repo /commit ID link)
M4	Code freeze - only bug fixing allowed	• all functionality and APIs available for testing • Jira Epic ticket updated with information about final API documentation, executed UI tests documentation • Documentation provided - confirmed by Documentation project PTL • Code freeze confirmed by Feature Lead (comment on Jira Epic ticket) - only bug fixing allowed
RC0	Release candidates freeze - branch stabilization	• Release candidates agreed, approved, tested • System tests conducted, quality confirmed by CI/CD PTL • Final documentation provided, reviewed, approved by Documentation PTL • Marketing information provided to Marketing Advisory Council • release branch cut off - branch stabilization from now on • Release Candidates freeze, confirmed by PTL and TSC
Deployment	Release deployment	official deployment executed

Release Cadence, Life Time & Support

The initial release cadence is planned for every 6 months.

The Community intends to actively support any release for up to 9 months, where any bug fixes will be committed and made available.