

# Release Milestones

## Overview

There is a well laid out process for release branches to move from a development phase into a final product launch:

- New features are developed for performance testing.
- A new branch is created and opened for devs to sync to and test.
- Once development is determined to be complete, the branch enters hard lock or code freeze.
- Temporary access is granted to a few allow-listers for anything considered to be a shipblocker (blocking the game from being able to be distributed to one of our client platforms).
- At Pencils Down all temporary access is removed, and only a handful of individuals have access to the final release candidate builds.
- Once certified, an approved build is released into the live environment for end user consumption.

The rest of this wiki page is a deeper dive into the milestones of this process.

① While the Release Team has Milestone documentation, the process is shared across many Fortnite teams. Knowledge capture from other teams is represented in this space.

### On this page:

- [Overview](#)
- [Release Team Training Deck: Release Milestones](#)
- [Release Team Session Recording: Release Milestones](#)
- [Core Milestones](#)
- [Feature Complete](#)
- [Branch Milestones](#)
- [Development Complete](#)
- [Hardlock](#)
- [Pencils Down](#)
- [Cert Submissions](#)
- [Release](#)

## Release Team Training Deck: Release Milestones



Sign in to your Google Account

You must sign in to access this content

[Sign in](#)

## Release Team Session Recording: Release Milestones

Link to session recording [here](#).

## Core Milestones

- Feature Complete
- Branch Operation / Creation
- Branch Opening / Work Cutover
- Development Complete
- Hard Lock
- Pencils Down
- Cert Submission
- Release

There are also Seasonal BR Team specific milestones outside of the release specific ones. Focusing on Release specific for this training, but they are listed to dig in more on the definition in confluence: <https://confluence-epicgames.atlassian.net/wiki/display/ERM/Milestone+Definitions>

UE has some additional milestones and expectations but is very similar to the FN milestones. We will spend more time on UE specifics in a future training session, but the UE definitions and expectations can be found here: <https://confluence-epicgames.atlassian.net/wiki/display/UEWiki/UE+Release+Milestone+Definitions>

## Feature Complete

### Definition/Expectations (Summary)

- Features are functional and playable.
- Design and World Content is in the build (Perf Testing can begin).
- First pass Art is in and represents complexity expected in the release for proper perf testing.
- Bugs/Perf Issues are common at this phase.
- Team leads and disciplines align on criteria for each feature and begin testing to vet completeness.
- Remaining work should be bug fixes, performance tuning, and polish.

### RM's Role

- Currently this is a very early milestone with little to no RM direct engagement.
- Over time, the goal is that Embedded RM's are more involved with initial planning within their products. Ideally, this will lead to a deeper understanding of:
  - tracked work going into each release
  - checklist items
  - timelines & general organization
  - earlier identification of key beats and potential risks to be ironed out earlier in the pipeline

## Branch Milestones


### Definition/Expectations

- Branch Operation / Creation
  - This is when the branch is created and syncs can begin. The branch will be locked until the Branch Opening / Work Cutover milestone. These release branches primarily branch off of Main but on certain occasions branch off of the previous release branch for special initiatives.
- Branch Opening / Work Cutover
  - This is when all continued work for a given release should move to that specific branch.

### RM's Role

- Release's Operations Pillar coordinates with Build Farm engineers to ensure stream setup dates are aligned, and coordinates initial stream creation and opening:
  - Update UGS banner
  - Enable scheduled Horde jobs
  - Etc..
- Release's Operations Pillar send outs communication on branch status, and an email once the branch is "Open for Business": when the branch is open and teams can begin submitting changes.
- These branches become actively monitored as part of build health processes (covered in prior trainings).
- Ensure we have proper dashboards, Slack channels, and other coordination items covered in the Release Checklist.

[Branch News] 34.40 Opens for Business Tuesday 3/11 9am ET inbox x

**Stan Hormell**  
to Release-Team, FN-Branch-News ▾

**Friendly reminder that 34.40 will branch from Dev-FN-34 and open for business Tuesday 03/11 at 9am ET.**

After that, all 34.40 work must be completed in the `//Fortnite/Release-34.40` stream.

**Key Dates:**

- **Weekend, March 7-9th, 2025 (Evening) : Branch Operation/Creation**
  - This is when syncs can potentially begin. We will send a reply to this email thread when we feel comfortable opening it
- **Monday, March 10th, 2025: Sync Release-34.40**
- **Tuesday, March 11th, 2025 9am ET : Branch Opening/Work Cutover**
  - From this point forward all 34.40 work will be required to be in the `//Fortnite/Release-34.40` stream.
  - All 35+ work should be done in the `//Fortnite/Main` stream.

🔴🔴 Dev-FN-34 will be archived 3/14 next week. Please sync Release-34.40 on Monday and move over to it as soon as possible. 🔴

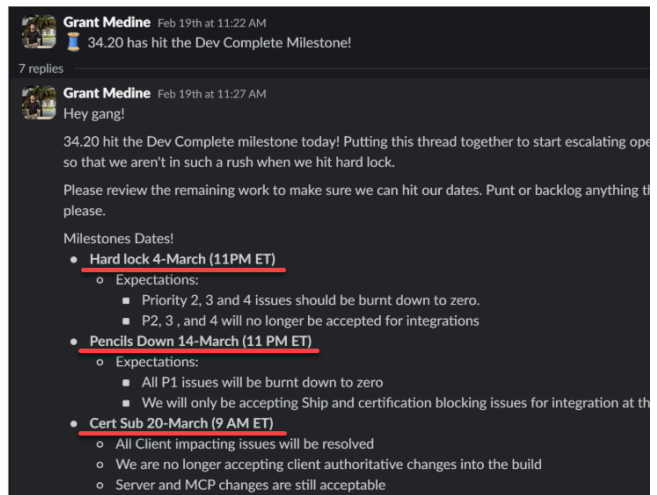
## Development Complete

### Definition/Expectations

- All features, art, and associated development efforts are completed, and developers are only performing bug fixing and final polish activities.
- String Lock occurs for localization efforts. Only bug fixes should change strings beyond this point.
- The team shifts at this point to focus all efforts on eliminating bugs before Hardlock.

### RM's Role

- Remind teams that we are now dev complete and only bug fixes and polish tasks should remain.
- Callout high volume of tickets in a given area, high buckets of triage, a lot of preSH, etc.
- Push for healthier buckets going towards Hardlock.



## Hardlock

### Definition/Expectations

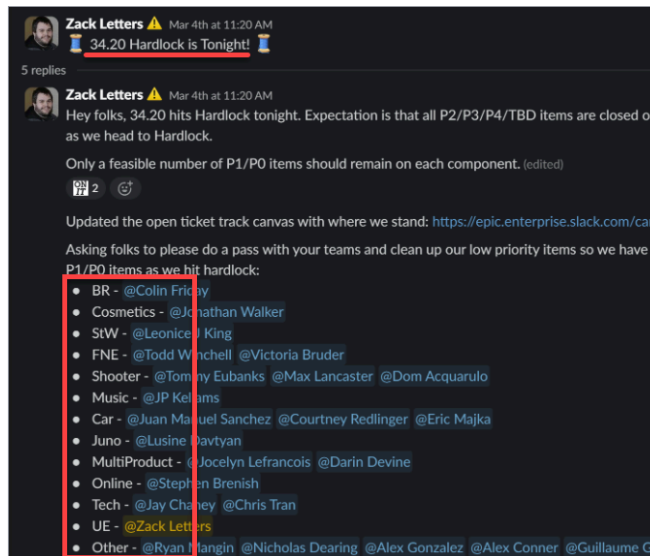
- All known tasks and bugs we've planned for this release are resolved.
- Final functional and performance testing may surface new bugs. We should be testing close to what our final release client will look like...
- Shipblocker bugfixes (P0/P1) only beyond this point.\*

① \*Through the early part of HL, we make exceptions for trailing loc, audio, and perf polish following FC polish coming in at the very end of the previous phase.

- Mechanical lock/code freeze is put in place. Submissions from this point forward need approval by BOTH a release manager and a designated senior/lead/director dev approver. A small amount of approved devs may be temporarily allow-listed during this time to avoid friction on any remaining critical iteration.

### RM's Role

- Escalate threads for any low priority TBD items (we should only have P0/P1 remaining).
- Begin integration process on the hardlocked stream (covered in more detail in Integrations).
- As we head to Pencils Down escalate PreSH, large triage buckets, and other critical spaces that are tracked on release dashboards and in checklist items.



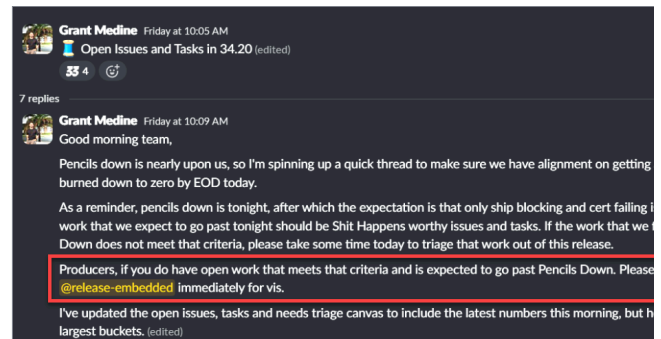
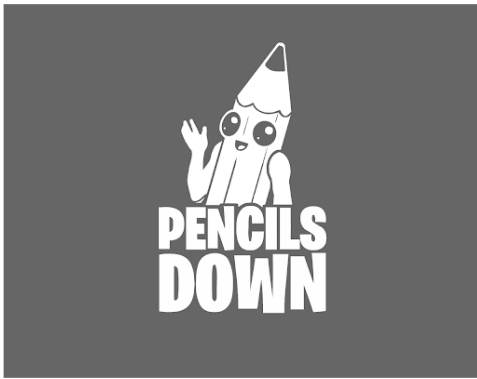
## Pencils Down

### Definition/Expectations

- All temporarily allow-listed individuals mentioned above are removed. Ideal delta between this build and final Cert build is only a handful of extremely critical submissions.
- We have a build in-hand that is a healthy release candidate.

### RM's Role

- Integrations go to a RM-only approval and submission, and require much more scrutiny on severity to take any changes in (only ship blocking items). Requires embedded testing prior to commit and hotfix cvars when possible, to give us a lever to revert if needed.
- RM will track all remaining items targeting the release, and will push in a more individual-focused way: following up with a specific engineer, production, component owner, etc. to finalize all work for the release with additional scrutiny.
- Push for SH's for late critical-work to align on severity / potential impacts to taking late changes.
- End goal is to finalize a build for us to submit for certification.



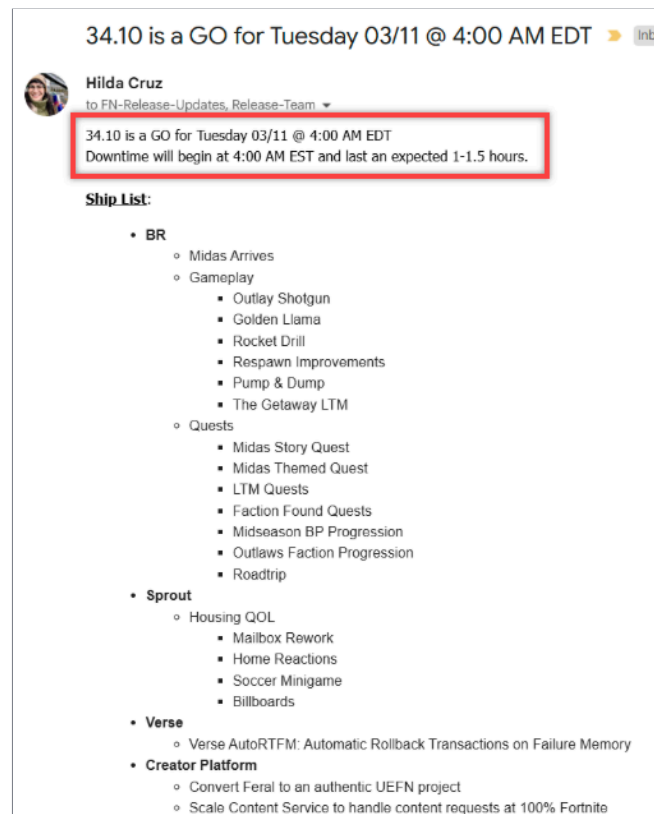
## Cert Submissions

### Definition/Expectations

- Final release candidate builds submitted for certification. Target builds start at 10:30pm ET the night prior.

### RM's Role

- Submissions Pillar coordinates and uploads our builds to certification partners and monitors/updates based on the results from submissions.
- Once we have sign off from all partners the Submissions team will send "Release is a Go" communications.



## Release

### Definition/Expectations

- Cert approved builds are released to the live environment for end-user consumption.

### RM's Role

- Operations team executes the run of show, and all standard deployment processes and the final builds are released live!!!

