DevOps	QA	Devs

Work Item

Team decides on release for each work item in sprint. → during sprint planning?

Dev works on feature/A branch & performs continues unit testing until task is done → all features are done on feature/name-of-feature branch

code

Merge develop branch into feature/A branch, work out merge conflicts → You need to ensure, that your changes still work with the changes others might have pushed to develop branch since you started working on your feature

Test feature again, incl quick integration test

All works? — No

Input from QA

Discuss with Team, ensure release still the same as originally decided on

Work Item

Wait till develop branch is for the release your work item is for

Is develop branch ready to receive merges for this release? — No

That is, does the develop branch have the code for the release your features is scheduled for. if not, do not merge. keep your feature separate, merge only once develop branch is ready.
Ultimate Goal - do not marge in code that is not ready to be released with the code you are merging too

Yes

Marge feature/A into dev branch via PR → This means:
- re-merge base branch to feature branch, retest
- do Pull Request
- have the pull request be code reviewed asap
- complete PR / merge code

code

Test on DEV environment. Ensure all is well

Automatically deployed (full CI/CD)

go to "in QA process"

DEV

DevOps	QA	Devs

From "pre release to QA process"

Code is in develop branch

Refresh QA from prod

This has to be done at least once before serious testing on QA starts. Can be done as often as desired. Must be performed in case manual interventions were performed on QA, for what ever reason, during testing. See note linked to QA Server object

Devops Work Items

Gather list of changes in release

Asses the scope of testing required & come up with testing plan

Cut a Release/{version} branch

Release code to QA

Perform smoke / sanity check on QA

Make sure your release does not break the app, and that your changes at least load and appear OK - 2 min test. If not, wear the hat of shame!!

QA

we CANNOT manually touch the QA env - database, code etc.

Only releases via DevOps pipeline are allowed

Test on QA
(Execute test plan on QA)

Bugs found?

Yes → Fix bug

Regression needs to be performed on QA. Once testing on QA is completed, no new bugs should be discovered on staging

Asses the scope of retest

(optional) Refresh QA from prod

Release code to QA

Retest on QA

No
(aka QA sign-off)

Refresh staging from prod

Release code to Staging

Staging

Test on Staging

Bugs found?

Yes

No fixes are not permitted on staging. Staging can have only 1 release. The purpose of staging, is to prove that the release package we got, is flawless.

If issues are uncovered in staging, fix has to be replicated on QA, fixed and re-tested on QA. Test on staging needs to be repeated. We proceed to prod release only once the 1 release to staging is found to be flawless

No → Release code to Prod

Prod

Sanity check only

Release done

Go to "post prod release process

DevOps	QA	Devs
		From "In QA process"
		Release live in prod
		Check error log - are there any new errors ?
		Check perf logs. are there any spikes?
		Done