



Courses > 310 Build Robust and Production Quality Applications > Week 5 > Continuous Delivery

# Continuous Delivery

**Continuous Delivery** (CD) or sometimes known as **Continuous Deployment** goes one step beyond CI to automatically deploy features when the new code passes the continuous integration phase. Continuous Delivery encourages small and incremental software updates over big and infrequent releases to shorten the feedback loop and fix bugs earlier.

Let's look at an example development workflow that has both CI and CD enabled, based on the Github Flow process:

- we pull the latest code from Github
- we create a new feature branch and develop a new feature
- after we finish the feature, we push it to a branch with the same name on Github
- we create a PR from this branch to the `staging` branch.
- we wait for the the CI server to ensure all tests pass.
- we allow the CI server to automatically deploy the code from the `staging` branch to our staging server
- we perform sanity tests on our staging server
- we create a PR from the `staging` branch to the `master` branch on Github
- this will trigger another round of integration and if it passes, the CI server will automatically deploy the code to the production server.

If adopted by everyone in the development team, this process will make sure our master branch is always in sync with the production server and new features are also always build on top of what's on the production server.

You marked this topic or exercise as completed.

[◀ Assignment: Set Up Continuous Integration with Circle CI](#)

[Assignment: Set up Continuous Delivery with Circle CI ▶](#)