

CI/CD:

The problem: Integration hell.

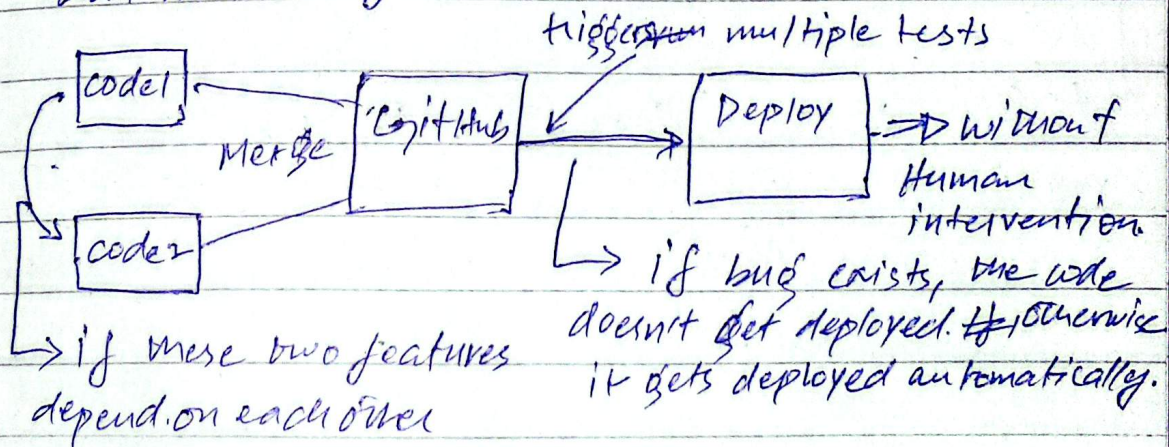
To optimise deployment, ~~push~~ test cases, & the complete development cycle.

CI (continuous Integration):

To merge every developer's code into a shared branch. (can be done multiple times a day).

Each merge triggers a build & a full test run.

And catch bugs instantly before deployment automatically.



CD (continuous Delivery):

The stage which requires human intervention before the code gets pushed to the main branch. We don't push code to the main branch via deployment (automatically).

Netflix has multiple pipelines & deployment happens almost automatically.

CD (Continuous Deployment):

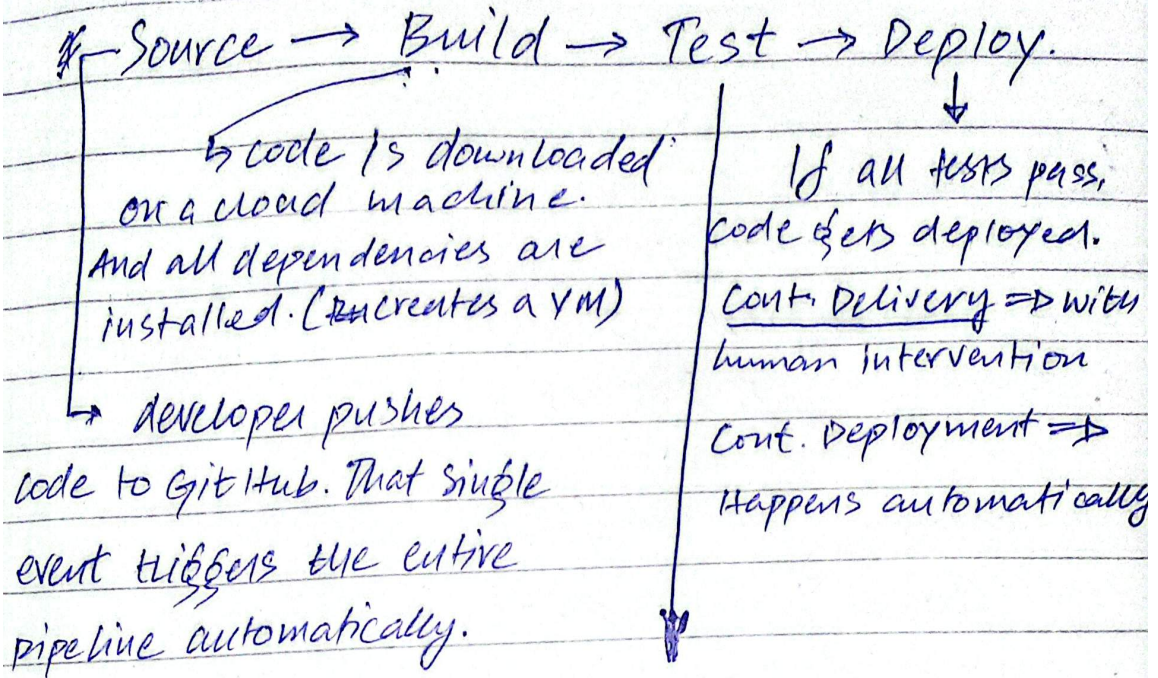
~~It doesn't~~ This stage doesn't need human intervention & if the code passes the automated tests, the code gets deployed to production.

Zero manual steps between a code commit & that being live ~~of~~ for users.

Up to the staging, the code works on continuous delivery, to ~~push~~ push the code to the production we ~~need~~ use continuous deployment.

⇒ This process depends upon the company requirement ~~which aspect~~ to where to use continuous delivery & continuous deployment.

The four stages of CI/CD pipeline:



Automated tests

run against the build code.

Unit tests, API tests, etc.

If any test fails, pipeline stops.