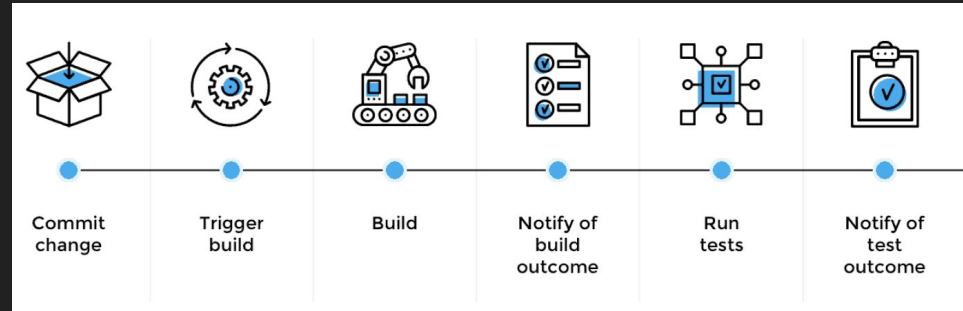# UDA People CI/CD

Fastest way of work integration and features delivery

What is more important than developing new features ?

- Developing new features is awesome, no one denies that, BUT, definitely it's not a single developer who made that cool feature alone!!.
- Having subsets of that feature working properly on each developer machine will not help to increase our company revenue.
- A true success is to organize all those cool feature subsets without any conflicts or fails, build and ship this to the market ASAP with less human interactions required.
- Here comes the CI/CD(Continuous Integration/Continuous Deployment) process into play.
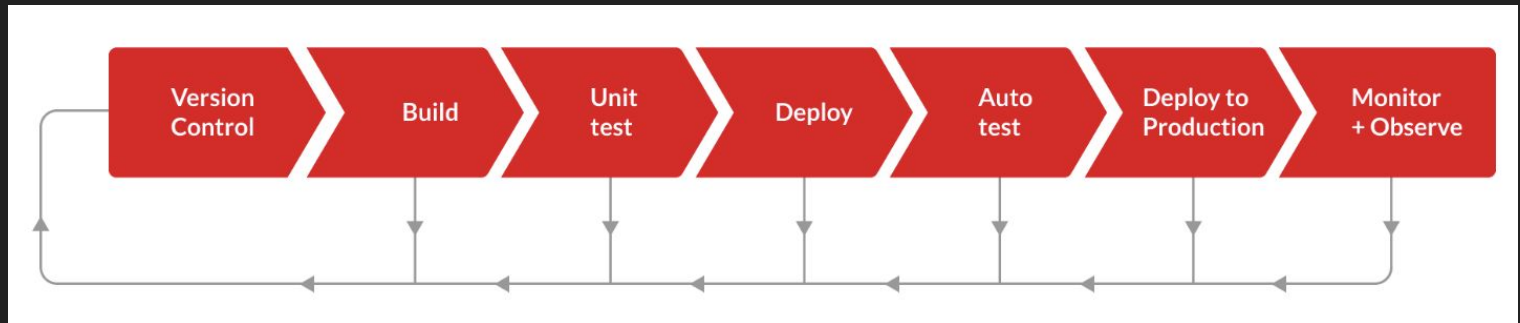
CI Continuous Integration

- Continuous Integration is the process of merging all developers work to a shared line (many times a day if needed), and testing the merged copy to make sure nothing breaks after any merge.
- Mainly it's related to the code related tasks.
- The outcomes of this phase are a high quality deployable artifacts.
- This process includes several sub-tasks/stages such as:
  Compile / Unit Test / Static Analysis / Storing artifact

| Commit change | Trigger build | Build | Notify of build outcome | Run tests | Notify of test outcome |

CD Continuous Deployment

- This process deals with the outcomes of the previous step (The deployable artifacts).
- Mainly it's related to setup of the infrastructure needed + deploying the artifacts that we have to the end users without human interactions and without having painful downtime to drive revenues up.
- This process includes several stages such as: Provisioning / Configuring infrastructure / Promoting the new developed version to production.

Benefits of CI/CD

What is the business impact of CI/CD ?

- Avoid Cost: catching test failures before shipping features to production means less bugs exposed to end users and hence less time in testing. Besides automation of deployment process means less human errors and faster deployments.

- Reduce Cost: early detecting of compile errors and conflicts means less time spent on issues debugging. Also automating infrastructure clean up after shipping new feature will reduce the cost of unused resources.

- Increase Revenue: faster and repeatable deployments means more and quicker features releases that gain more income to the company (Less Time to Market).

- Protect Revenue: automated smoke tests will help to reduce downtime related to new release related crashes/bugs.