

CICD

What is CICD:

CICD stands for Continuous Integration & Continuous Delivery where

Continuous Integration (CI) is a set of practices to integrate developer's work into the main source code to be deployed later. It contains of an automated process to build, test and analyze the code before merging it into the source code.

CI: Code Build -> Unit Case execution -> Static Code analysis -> Vulnerability Test

Continuous Deployment(CD) is a set of practices to automate the deployment process. It handles the infrastructure creation, configuration, and even quick smoke tests before promoting a new code into production.

CD: Infra Provision -> Infra setup & configure -> Database migration -> Deploying application -> PROD release -> Smoke Testing Finally, Celebrations

Benefits of CICD

There are majorly for business benefits:

1. Reduce Cost
2. Avoid Cost

3. Increase Revenue

4. Protecting the money to not waste

Enterprise challenges	CI/CD	CI/CD over containers
Bug fixes/new features New features and fixes have to wait for the release cycle which may take 6 months to 1 year.	Fully overcome	Fully overcome
Dependency Gap between development and operations creates a lot of unnecessary dependencies, which add additional time and cost	Partially overcome	Fully overcome
Efforts in release From development to deployment, multiple manual processes are involved leading to high effort, time, and scope for errors	Partially overcome	Fully overcome
Tools restriction Enterprises have a restriction on using new and sophisticated tools and test suites, due to incompatibility with the existing systems.	Not overcome	Fully overcome
Production Downtime Inability to deploy new major releases on production environment without downtime.	Not overcome	Fully overcome
Infrastructure Capacity Management Capacity management in a dynamically growing and changing user base is a continuous challenge as it contributes to increased infrastructure costs.	Not overcome	Fully overcome

- Less infrastructure costs from unused resources
- Less bugs in production and less time in testing
- Less human error, Faster deployments
- Prevent embarrassing or costly security holes
- New value-generating features released more quickly
- Less time to market
- Quick undo to return production to working state
- Reduced downtime from a deploy-related crash or major bug

Product development journey using CICD:

- Backlog of non-critical defects is lower because defects are often fixed before other feature pressures arise. •
- More time for developers/QAs to focus on the big picture instead of spending their valuable time on fixes
- More time for QAs to find larger problems before being a product is released. • Product release velocity is high.
- Release cycles are shorter with targeted releases and this blocks fewer features that aren't ready for release.
- Customers are happy with fast turnaround & roll-out of new features and bug fixes.
- Product release cycles are shorter with targeted feature releases and this only blocks fewer features that aren't ready for release.
- Keeps Customer happy as they end up finding zero to none errors in your product.
- Add new requirements based on customer's needs/feedbacks on a daily basis during continuous development, which leads to usability improvements.
- Seamless & targeted introduction of new production features by toggles and blue-green deploys features