CI/CD
Continuous Integration / Continuous Delivery

Muhammad Desouky

CD/CD

CI/CD is a two-step process that dramatically streamlines code development and delivery using the power of automation. CI makes developer tasks like source code integration and version control more efficient so software can get into production faster. CD automates software testing and deployment. Together, CI/CD is a powerful and unmatched engine of modern software development and it has untold benefits for businesses.

What are the CI/CD benefits for business

- **Ensure superior code quality**

  Because CI/CD pipelines offer test automation, developers can know about code problems nearly in real time. That concept of “failing fast” means teams aren’t wasting time or resources with buggy code, and devs aren’t plagued with endless “fix” requests when they’ve moved on to other projects. Time is saved, money is saved, and developers aren’t endlessly context switching… win, win, win.

- **Deliver faster with an accelerated release rate**

  Skeptics about the benefits of CI/CD need only hear about global financial firm Goldman Sach’s success story: It’s Technology Division went from one code build every two weeks to over 1,000 builds per day. A unified CI/CD pipeline is like a turbo engine when it comes to boosting the rate of software releases. The faster code is released, the more new code can be developed, and then released, ad infinitum. The business bottom line: Expensive developer resources aren’t sitting idle when a successful CI/CD pipeline is in play.

- **CI/CD pipelines: Automation reduces the cost**

  Anytime a human does not have to intervene in the software development process, time, and thus money, are saved. That’s why automation is the underpinning to successful DevOps practices. CI/CD automates the handoffs, the source code management, the version control system, the deployment mechanisms, and, of course, so much of the testing.

What are the CI/CD benefits for business

- ## Fault isolation

  Before DevOps and CI/CD gained traction in software development, development teams would know there was an issue with code, but would struggle to know exactly *where* the problem was happening. CI/CD and its automated testing has changed that. Developers can easily identify and then isolate code faults, dramatically improving productivity.

- ## Simplified rollback

  A CI/CD pipeline gives developers the power to fail fast and recover even faster. It's a simple process to push code into production and, if there are issues, simply roll it back. The ability to easily rollback code saves teams time, energy, and resources and leads to faster fixes of problem code.

- ## Continuous feedback

  A unified CI/CD process, operating as part of a DevOps platform, gives everyone on the team – including business stakeholders – a way to see what's happening, where it's happening, and what might be going wrong. This sounds like a simple thing, but in reality, a single window into software development is almost revolutionary.

- ## Optimum transparency and accountability

  Thanks to continuous feedback, a CI/CD pipeline makes the entire software development process completely transparent to the business side. Product managers can check project status in a glance and track accountability as needed.

What are the CI/CD benefits for business

- ### Improved mean time to resolution (MTTR)

  Thanks to the visibility provided by a CI/CD pipeline, DevOps teams see issues quickly and can fix them fast. The ability to rapidly resolve problems lies at the heart of a key development metric: mean time to resolution, or MTTR. The better the MTTR, the more efficiently the DevOps team is working and the more quickly software can be released; in other words, MTTR has a direct effect on a business's bottom line.

- ### Monitoring metrics data

  Teams and the business side need to know how code is functioning in the real world, but in traditional software development practices monitoring metrics are often absent. In an ideal world, teams would know there was a code problem and roll it back long before end users realized it. A CI/CD pipeline makes that "ideal world" a reality by delivering continuous feedback on a variety of metrics. Access to metrics data is more than just a time-saver, however, as no organization wants to be associated with bug-ridden code and applications that don't perform well.

- ### Reduction of non-critical defects in backlog

  By now it's clear CI/CD is a time and money saver, so much so that it gives developers time to work on things they wouldn't normally be able to, such as going back to fix older code and make it cleaner and more efficient. The idea that devs cannot only tackle the backlog (it's called a backlog for a reason after all – who has time for this?), but also work on non-critical defects, is a game-changer brought to teams by DevOps and CI/CD.

Thank You