What is CI/CD?

Continuous Integration is a development practice that requires developers to integrate code into a shared repository several times a day. Everything is then checked by an automated build, allowing teams to detect problems early.

Continuous Delivery, on the other hand, is the ability to get changes of all types into production safely and quickly in a sustainable way. This is achieved by ensuring the code is in a deployable state, even when developers are making changes every day.

Why business should adopt CI/CD?

Organisations that haven’t yet adopted Agile, CI/CD, or DevOps, they are at risk to be short-lived, slow to adapt, and fail to keep up with organisations of size moving at the pace of a startup.

Therefore, CI/CD plays a vital part in software building and deployment and are more necessary than ever if organisations want to move forward. They are able to provide various benefits to businesses as well as to stakeholders including product owners, development teams, and end-users, among others.

In the long term, having a CI/CD process will lead to big advantages, such as reducing costs and increasing return on investment. Moreover, it will also allow businesses to invest more time in building better mobile apps with faster time to market.

Benefits of CI/CD for Businesses

1. Reliable, Frequent Releases

The continuous updates and test automation in CI helps ensure reliable, high-quality releases that have fewer errors and bugs that could reduce their effectiveness. For end users, that translates to enhancements that they can put to use more quickly to drive business growth.

2. Reduced Costs

CI reduces manual tasks and errors, which also reduces the risk of outages or downtime after release. For end users, that provides a higher level of service and increases productivity, especially when CI is extended to CD, where deployments occur automatically across test environments and into production. More importantly, however, it can drastically reduce costs as businesses won't have to spend time and resources resolving outages.

3. Faster Software Delivery

Faster build times, more reliable code and tests, and reduced outages all translate to faster delivery of new enhancements to market for companies and their users – often as part of a larger continuous delivery process. Companies can use their enhanced functionality to improve products and services for their customers and gain a competitive advantage in the marketplace.

4. Better Manage Backlog

By implementing CI/CD into your development process, you have the chance to decrease the number of non-critical defects in your team's backlog. Such defects are often fixed before they become a critical issue. Any such defect is highlighted and is fixed before it makes it to production or impacts end-users.

5. Cost Deduction

Using a CI/CD pipeline limits the potential impact and loss that a deployment problem can cause by allowing it to be deployed in non-critical business hours. Also, repeated automated deployments during the development phase help developers catch the errors early before causing any significant damage. Such a pipeline implementation increases code quality, thus increasing overall ROI for the organization.

6. Ease Updates & Maintenance

Regular maintenance and updates are the backbones of making a great product, and this is one of the great benefits of CI/CD. It ensures that release cycles are shorter and targeted, which blocks fewer features that aren't ready for release. In a CI/CD pipeline, maintenance is usually done during non-business hours, saving precious time for the entire team.

Furthermore, features like toggles and blue-green deploys enable the seamless and targeted introduction of new product features by upgrading smaller units of change, which are less disruptive.

7. Performance Metrics

Applications, in all cases, need to be monitored carefully after releases. This can be considered amongst one of the top benefits of the CI/CD pipeline as it allows you to monitor the health, performance, and reliability of an app. Monitoring these metrics help in creating actionable insight and taking necessary actions to improve the product.

These metrics are also important as the software is nowadays developed and deployed quickly, thanks to CI/CD pipelines, and hence their behaviour becomes somewhat unpredictable at times. CI/CD provides metrics data in operational, time-based, and quality metrics, which is very useful in making things better.