Top 10 Benefits of Continuous Integration & Continuous Delivery

Continuous Integration (CI) allows you to continuously integrate code into a single shared and easy to access repository. Continuous Delivery (CD) allows you to take the code stored in the repository and continuously deliver it to production. CI/CD creates a fast and effective process of getting your product to market before your competition as well as releasing new features and bug fixes to keep your current customers happy.

1. Smaller Code Changes

One technical advantage of continuous integration and continuous delivery is that it allows you to integrate small pieces of code at one time. These code changes are simpler and easier to handle than huge chunks of code and as such, have fewer issues that may need to be repaired at a later date.

2. Fault Isolations

Fault isolation refers to the practice of designing systems such that when an error occurs, the negative outcomes are limited in scope. Limiting the scope of problems reduces the potential for damage and makes systems easier to maintain.

3. Faster Mean Time To Resolution (MTTR)

MTTR measures the maintainability of repairable features and sets the average time to repair a broken feature. Basically, it helps you track the amount of time spent to recover from a failure.

4. More Test Reliability

Using CI/CD, test reliability improves due to the bite-size and specific changes introduced to the system, allowing for more accurate positive and negative tests to be conducted.

5. Faster Release Rate

Failures are detected faster and as such, can be repaired faster, leading to increasing release rates. However, frequent releases are possible only if the code is developed in a continuously moving system.

6. Smaller Backlog

Incorporating CI/CD into your organization’s development process reduces the number of non-critical defects in your backlog. These small defects are detected prior to production and fixed before being released to end-users.

7. Customer Satisfaction

The advantages of CI/CD do not only fall into the technical aspect but also in an organization scope. The first few moments of a new customer trying out your product is a make-or-break-it moment.

8. Increase Team Transparency and Accountability

CI/CD is a great way to get continuous feedback not only from your customers but also from your own team. This increases the transparency of any problems in the team and encourages responsible accountability.

9. Reduce Costs

Automation in the CI/CD pipeline reduces the number of errors that can take place in the many repetitive steps of CI and CD. Doing so also frees up developer time that could be spent on product development as there aren’t as many code changes to fix down the road if the error is caught quickly.

10. Easy Maintenance and Updates

Maintenance and updates are a crucial part of making a great product. However, it’s important to note within a CI/CD process to perform maintenance during downtime periods, also known as the non-critical hour. Don’t take the system down during peak traffic times to update code changes.

Conclusion

There are many tools that can help enable a smoother transition to a CI/CD process. Testing is a large part of that process because even if you are able to make your integrations and delivery faster, it would mean nothing if was done so without quality in mind. Also, the more steps of the CI/CD pipeline that can be automated, the faster quality releases can be accomplished.