Understanding CI-CD

Fundamentals – CI/CD

What is Continuous Integration?

The practice of merging all developers' working copies to a shared mainline several times a day.

What is Continuous Deployment ?

A software engineering approach in which the value is delivered frequently through automated deployments.

What is Continuous Delivery ?

An engineering practice in which teams produce and release value in short cycles.

In Summary :

Continuous Integration + Continuous Deployment = Continuous Delivery

Continuous Delivery Over Traditional Delivery

Continuous Delivery is not about what we deliver but it is all about how we deliver. It is a change in mindset so as to deliver continuously in the form of small functional features aligned with time to market to gain customer’s trust and confidence on quick and quality deliveries.

Whereas Traditional delivery is focused on delivering the final Product in one go which takes very long to get into Production , thus lags behind the market pace and thus loses the confidence of the customer.

Benefits - CI

- Continuous Integration –

Multiple teams can work in parallel on the various features of the target Product without waiting for anyone to upload their latest code. This feature reduces the dependency .

The issues in the code are resolved at an early state while integrating the code for the target model features.

The time to resolve the issues are less due to small pieces of code wrt the features being worked on.

Cost and Business Benefits :

1. Parallel Work saves a lot of time and enables faster delivery which leads to quick turned around in high ROI.
2. Resolving issues at an early stage leads to productionizing the quality product quick which results in faster time to market.
3. Focused delivery due to small chunk of code leads to developer satisfaction and eventually quality delivery.

Benefits - CD

- Smaller code changes are simpler (more atomic) and have fewer unintended consequences.
- Fault isolation is simpler and quicker.
- Mean time to resolution (MTTR) is shorter because of the smaller code changes and quicker fault isolation.

Cost and Business Benefits :

1. Smaller code have lesser issues and are quick on getting resolved thus faster deployment and faster delivery. Code in Production makes more money than code waiting for getting productionized in the traditional way.
2. Quality delivery is the key to Customer confidence and long-term relations with the customer . This is achieved by isolating the faults quicker.
3. Time to market is an important factor for business and with CICD, MTTR is shorter leading to adding an edge to business competitors who still follows traditional ways of delivery.

Business Benefits CI-CD

1. Increased speed of innovation and ability to compete in the marketplace – Speed will let the clients give the Project to the company fast in their deliveries and inline with Time to market.

2. Code in production is making money instead of sitting in a queue waiting to be deployed - Organizations that have implemented CI/CD are making revenue on the features they deploy, not waiting for a manual check to see if the code is up to par. They already know the code is good because they have tests that are automated, and continuous delivery means that code is deployed automatically if it meets certain standards.

3. CI/CD supports customer outcomes from a technical standpoint. one of the biggest benefits of CI/CD is that it allows businesses to ensure customers receive a seamless experience and uninterrupted service. When a product goes to market, organizations typically monitor and record user activity and collect feedback from consumers. This allows them to make sure that software bugs and usability issues don’t go undetected and cause lasting damage to the product’s (or the company’s) reputation.

Frequent updates, new feature releases, rapid response to feedback, and quick bug fixes all play a major role in customer satisfaction and long-term loyalty. By using CI/CD, organizations can continuously build on applications and enhance the experience without the risk of downtime or interruptions.