Introduction CI/CD

Course Agenda

✓ What is CI/CD

✓ What is the business benefits of CI/CD

✓ Conclusion;

What is CI/CD

- Continuous integration/continuous delivery (CI/CD) is a software development practice that combines development and operations teams and their day-to-day tasks. It applies automation to developing, testing, and delivering applications.
- By unifying these processes under one strategy, CI/CD supports a smoother deployment process, brings more structure to the entire code and the development process, and enables more frequent updates with fewer disruptions.
- While CI/CD is often discussed as a single strategy, CI and CD represent two distinct ends of the quality control spectrum

Continuous integration (CI)

- ❖ Continuous integration (CI) is a software development process where teams integrate code early and often into a central repository where they can run frequent tests and validate changes.

- ❖ The goal is to speed up the release process by enabling teams to identify and fix problems early in the development lifecycle.

- ❖ CI reduces the amount of time spent on bug fixes and regression testing and ensures that everyone has a deep understanding of what’s happening inside the codebase and what features they’re developing for end-users.

Continuous Delivery (CD)

- ❖ Continuous Delivery (CD) is the practice of getting all updates, fixes, features, and configuration changes either into production or into the hands of end-users as quickly (and safely) as possible.

- ❖ The goal is to streamline the delivery/deployment process so that predictable tasks can be performed on-demand.

- ❖ CD aims to ensure that the code is always in a deployable state, even with developers making continuous updates to the codebase by bringing integration and testing together.

- ❖ The result is, releases become a routine affair that can be performed anytime new code is ready. This allows teams to streamline the development process, reduce risk, and implement user feedback more quickly. If any errors are detected during production, they can be addressed immediately by simply rolling out the next update.

CI/CD Pipeline

- ❖ CI/CD pipelines are a series of steps that must be completed to deliver a new software release
- ❖ The CI/CD pipeline typically breaks down into the following stages:
  - ✓ Build
  - ✓ Test
  - ✓ Release
  - ✓ Deploy
  - ✓ Validate

Benefits of CI/CD

- Bring Products to Market Faster
- Allows Developers to Deliver Products Consumers Want Now.
- CI/CD enables organizations to respond to consumer needs as they evolve.
- CI/CD plays a crucial role in shortening time to value.
- CI/CD supports customer outcomes from a technical standpoint.
- Boosts DevOps efficiency
- CI/CD Improves App Quality
- Supports Cloud-Based App Development
- Reduce Costs and Boost Profits
- Gain Real-Time Visibility of the Development Process

Benefits of CI/CD

CI/CD language	Captured Value	Translation
Catch compile Errors after merge	Reduce cost	Less developer time on issue from new developer code
Catch unit test failures	Avoid cost	Less bug in production and less time in testing
Detect security vulnerabilities	Avoid cost	Prevent embarrassing or costly security holes
Automate infrastructure creation	Avoid cost	Less human error, faster deployments
Automate infrastructure cleanup	Reduce cost	Less infrastructure costs from unused resources

Benefits of CI/CD

CI/CD language	Captured Value	Translation
Faster and more frequent production deployments	Increase revenue	New value-generating features released more quickly
Deploye to production without manual checks	Increase revenue	Less time to market
Automated smoke tests	Protect revenue	Reduced downtime from a deploy-related crash or major bug
Automated rollback triggered by job failure	Protect revenue	Quick undo to return production to working state

Conclusion

- ❖ The benefits of CI/CD impact all ends of the development lifecycle, the customer experience, and the big-picture business strategy.
- ❖ It plays a critical role in software development and delivery and helps smaller teams move faster, respond to constant changes, and incorporate real time feedback—all of which contribute to cost savings, profitability, and a higher-quality end-product.

Thank you