CI/CD

Continuous Integration, Continuous Delivery and Continuous Deployment

Agenda

INTRODUCTION

CI/CD OVERVIEW

BENEFITS OF CI/CD

CONCLUSION

Introduction

- CI/CD stands for Continuous Integration, Continuous Delivery and Continuous Deployment, and it refers to the software development practice of frequently integrating code changes into a central repository, automatically building and testing the code, and then automatically deploying it to production. The goal of CI/CD is to improve the quality of software, speed up the development process, and reduce the risk of introducing bugs into production

CI/CD Overview

- Continuous Integration is a software development practice where members of a team integrate their work frequently. Every commits/changes made to a shared repository must be run through set of test cases to make sure new changes are accepted. The result of that run should be informed to responsible persons.
- Continuous Delivery is the ability to continuously deliver integrated code, be it bug fixes or new features, to production. Software can be deployed at any given time.
- Continuous Deployment: Every change that passes all stages of your production pipeline is released to your customers. There's no human intervention, and only a failed test will prevent a new change to be deployed to production.

BENEFITS OF CI/CD

The Implementation of CI/CD has a lot of benefits to organizations and businesses, which includes;

- **Reduce Costs and Protect Revenue:** CI/CD standardizes deployment processes across all project. It enables teams to systematically test every change made to the source code. As a result, this process stands to dramatically reduce the likelihood that any bugs or errors slip through the cracks and cause problems down the line. This practice can lower development costs by eliminating many of the costs incurred while building and testing code changes because teams would now spend less time on testing and bug fixes, which means that the organization would spend less money on tasks that don’t provide any value to the business or its customers.
- **Cost-effective:** Automation in the CI/CD pipeline helps to minimize the amount of defects or bugs that occur in CI and CD steps. This helps to reduce the developer’s time and effort and also minimizes the cost.

BENEFITS OF CI/CD

- **Reduced time-to-market**: The ultimate goal of the CI/CD is to build and deliver software to the end users very quickly. Moreover, software development has gone beyond introducing new features, writing robust code, and understanding users’ needs. With CI/CD, we can ship changes and launch new features not just weekly, but daily, and even hourly and these helps improve customer’s confidence in the business.
- **Creates Revenue**: CI/CD reduces human intervention across the DevOps lifecycle by automating the handoffs, version controlling, source code management, deployment processes, and testing, among others. This significantly saves the time and cost that’s required to develop and deliver high-quality software. However, with a successful CI/CD pipeline in play, the development teams aren't plagued with endless 'code fix' requests, so they can actually focus on the next projects, thereby maximizing the overall ROI for the company.

CONCLUSION

- Continuous integration, delivery and deployment (CI/CD) have enabled many organizations to release more frequently without compromising on quality. With CI/CD, code changes has been shepherded through an automated pipeline that handles the repetitive build, test and deployment tasks and alerts you about any issues. Many Organizations has leveraged on CI/CD to take their production activities to the next level thereby reducing costs, protecting revenue, control cost as well as creating more revenue.

Thank you

Ojelade Oluwatobi