UdaPeople

07.07.2022

Ikechukwu Onyejekwe
Udacity
14 Murtala Mohammed Avenue
Port Harcourt, RV 500211

Overview

UdaPeople intends to be a groundbreaking entry in the world of human resources to help small businesses better manage their employees. However as team lead for the engineering team, I am bringing to your attention the need for the implementation of Continuous Integration, Delivery and Deployment (CI/CD) to our software development practice.

With the introduction of CI/CD, it is anticipated that we will be able to attain customer satisfaction while simultaneously increasing our speed and agility to be able to adapt to the ever changing needs of the market. Thus by keying into this idea there is huge potential to increase revenue while keeping costs down.

Background

Previously and up until now, our development process tends to be time consuming with a lot of friction between the various teams involved. And even when things manage to get done, it is often marred by errors which will require rollbacks and more time consuming iterations, thus incurring costs on man hours and resources.

This is where CI/CD comes in, at it's core CI/CD can be defined as follows

- Continuous Integration - Developers practicing CI merge their code changes back to the main branch as often as possible, after which a suite of automated tests are carried out.

- Continuous Deployment - In Continuous Delivery once code changes pass the suite of tests, the resulting artifact is released to customers automatically.

- Continuous Delivery - Continuous Delivery is an extension of Continuous Integration since it automatically deploys all code changes to a testing and/or production environment after the build stage.

One key takeaway from the above is the increased reliance on automation due to its repeatable and predictable nature.

Benefits

I. Respond to defects early

By Integrating changes frequently, we avoid the cost of integration by discovering conflicts at the boundaries between new and existing code.

II. Automated testing

Incorporating automated tests every step of the way protects revenue by asserting product quality, hence failures are detected and handled before proceeding down the pipeline.

III. Accelerate feedback loop with customers

More frequent releases allows for quick feedback directly from end-users, allowing for necessary changes and rollbacks to be made quickly and efficiently to protect revenue.

IV. Consistency throughout the deployment pipeline

Through the use of extensive automation, test and production environments are hence repeatable and reliable thus reducing the likelihood for the introduction of human error which helps avoid the cost involved with troubleshooting errors.

V. Detect security vulnerabilities

Use of automated test suites aids in detecting security vulnerabilities thus avoiding costs involved in the event of a security breach.