CONTENTS

INTRODUCTION/ FUNDAMENTALS OF CI/CD

The goal of our business is to get maximum results from Udapeople by increasing revenue and reducing cost. CI/CD helps to achieve this by ensuring that proven processes are in place to achieve set goals.

What is CI/CD

CI/CD is the acronym for the combined practices of Continuous Integration (CI) and Continuous Delivery (CD). They are widely accepted choice to lessen software development and delivery cycle time.

CI/CD methodology allows us to quickly respond to changing market and customer demands.

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

COMPONENT OF CI/CD PIPELINE

A CI pipeline usually involves the following tasks:

- Build
- Execute all unit tests
- Execute all integration tests
- Generate deployable artifacts

A CD pipeline usually involves the following tasks:

- Reviewing of Artifact
- Delivering build to staging
- Promoting to production

BENEFITS OF CI/CD

This can be divided into four major categories

i) Create revenue. ii) Protect revenue. iii) Control/Avoid cost. iv) Reduce cost.

Create revenue.

- Customer satisfaction is the goal of our business. Utilizing a CI/CD approach also keeps our product up-to-date with the latest technology and allows you to gain new customers who will select us over the competition through word-of-mouth and positive reviews.

- CI/CD makes maintenance and Updates easier as they are done during non critical hours, ensuring that our application is always up for new and existing users to access and in turn creating revenue.
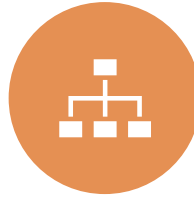
Protect revenue.

- It helps with Faster Mean Time To Resolution (MTTR) , we track the amount of time spent to recover from a failure. The helps protect revenue as the longer Udapeople is down, the more revenue is being lost.

- This helps with deploying our new features quicker and hereby ensuring that our customers are retained and also gaining new ones. Failures are detected faster and as such, can be repaired faster, leading to increasing release rates.

BENEFITS CONTINUED

Control cost/avoid cost

Few code changes and integrations makes the team have fewer issues that may need to be repaired at a later date which leads to more collaboration between teams and control of operational cost

Continuous reliability ensures that quality product is released every time an update or feature is added to our product, hereby ensuring that we don't deploy unfit products to the market

Reduce cost

This helps reduce cost by making sure people are accountable for their actions and feedbacks are received when there are build failures, merging problems, architectural setbacks, etc. It also helps get customer's feedbacks when products are released.

CI/CD comes with a lot of Automation which helps reduce the number of errors that can take place when using manual processes. This helps to keep failures to a minimum and quickly recover from any failures that do happen.

CONCLUSION

CI/CD are two DevOps best practices as they tackle the misalignment between developers and the operational team. With the presence of automation, our developers can release changes and new features more frequently creating more revenue and reducing cost for operation of UdaPeople.