ADOPTION OF CICD

By Emmanuel Ibikunle

WHAT IS CONTINUOUS DELIVERY (CD)?

There are a lot of definitions of Continuous Delivery out there. I believe it is more of a mindset - an engineering paradigm that influences and enhances the practices of **Continuous Integration** (how our developers combine their codes) and **Continuous Deployment** (how our operations team deliver products/features).

CONTINUOUS DELIVERY =
CONTINUOUS INTEGRATION (CI) + CONTINUOUS DEPLOYMENT (CD)

NB: *CICD can be done without tools (like we currently do). It is more about the approach to the work than the work itself.*


Sec
Dev
Ops
Plan
Create
Verify
Preprod
Monitoring and Analytics
Adapt
Release
Prevent
Detect
Respond
Predict
Monitoring and Analytics
Continuous Delivery

HOW CI/CD WILL BENEFIT US

- **Complete Automation**
  With CI/CD, we will be able to automate all of the time consuming tasks our development and operations team do almost all of the time. ***This means less man hours and we are able to reduce cost.***

- **Faster Time to Market**
  With CI/CD, we can ship new features quickly as soon as they are built, we don't have to wait for the entire application to be completed. This keeps us ahead in the market, our customers know they can depend on us to deliver quickly, ***hence we can be sure our revenue is protected***.
  Also our ability to reach the market quickly improves our chance of onboarding more customers, ***hence increasing our revenue.***

HOW CI/CD WILL BENEFIT US (CONT'D)

- **Improved Quality of Code**

  With CI/CD, stages like code review, code compilation, testing and infrastructure deployment are automated. This means that we can quickly catch hidden software bugs that would have made their way to production server. Catching this bug (which in some cases may be exploited by an outsider if unnoticed) before it goes live to production ***can help us avoid unnecessary cost.***

NB: *It is more about the approach to the work than the work itself.*

THANK YOU