Head of Operations

Udapeople Int.

59, Adetola Street, Aguda Surulere,

Lagos, Nigeria.

PROPOSAL FOR THE IMPLEMENTATION OF CONTINUOUS INTEGRATION AND CONTINUOUS (CI/CD) FOR UDAPEOPLE INT

INTRODUCTION

In recent days Code updates that are merged into Project repository are made reliable by automated build-and-test procedures that are sparked by CI/CD. Thus, making the overall deployment process of the organization's application easier.

WHAT CONTINUOUS INTEGRATION AND CONTINUOUS DEPLOYMENT (CI/CD) ENTAILS

Continuous integration (CI) is a modern software development technique where incremental code changes are reliably and regularly made. It is the routine of often merging all developers' working copies into a shared repository. This method essentially cleans up the code that developers have created and makes sure it complies with all of the functional and non-functional requirements of the organization's product. Then, as part of the continuous delivery CD process, the code is swiftly and easily supplied, as It is a process of software engineering in which the product) is commonly supplied by automated deployment. Everything pertaining to the deployment of artifacts is handled in this stage.

The CI/CD pipeline, as used in this context, is the automation that enables developers to swiftly and reliably transfer incremental code changes from their desktops to production, and the creation of excellent, deployable artifacts is its ultimate objective.

The following are examples of CI/CD processes.

CI Processes:

- Compile
- Unit test
- Static analysis
- Dependency vulnerability testing
- Store artifact

CD Processes:

- Creating infrastructure
- Provisioning servers
- Copying files
- Promoting to production
- Smoke test
- Roll backs

A pipeline that delivers code in seconds at the push of a button is made possible by the combination of CI and CD procedures. While removing bugs, security vulnerabilities, human error, and our lead time to market, among other advantages.

Business Benefits of CI/CD

The benefits of CI/CD includes:

1. Boost sales by providing new features that add value as soon as possible.
2. Boost revenue since product can reach the market more quickly than our existing manual procedure can.
3. Cost reduction as developers spend less time on issues from new developers' code,

4. Cost avoidance as less bugs are present and developers' focuses on testing and debugging time is spent
5. We can immediately undo changes made by a failed deployment and return to the prior operating state to protect income by cutting down on downtime from a failed development.
6. Avoidance of cost as it helps prevent embarrassing or expensive security gaps.
7. Reduce costs by reducing human error and facilitating quicker deployments.
8. Lowers expenses by lowering infrastructure costs due to the ability to automatically

Thank you for your time and keen attention to this project as we gracefully anticipate your reliable response.

Blessing Akpan

DevOps Engineer, Udapeople Int.