## **Comparing CI/CD Tools: AWS CodePipeline vs Jenkins vs GitHub Actions**

## **What is This Project About?**

This project examines the way it may be possible to speed up and simplify software building and delivery. We are going to compare three known tools that assist developers in testing their code automatically and sending it: AWS CodePipeline, Jenkins, and GitHub Actions. Docker will also be used in making everything run smoothly.

## **Why Did We Choose This Topic?**

In the modern world, organizations have to deliver software updates very fast without error. Traditional techniques of software development and release are slow, and cause issues.

The fastness is important since those companies that send features faster acquire more customers. Software containing bugs is capable of breaking websites or losing data, which is both expensive and frustrating.

By limiting the amount of time that the developers spend on work, they are able to develop the good new features instead. The problem is common among most companies that produce software.

**What Problem Are We Solving?**

Lots of businesses are suffering from lagging software releases. New features take people many weeks or months to get to customers, as everything is done manually.Without decent automatic testing, there are too many bugs delivered to the customer. Developers spend time repeating the same actions over and over such as by copying files and executing tests rather than creating something new.There are also cases that the companies experience different outcomes such as code running on one computer but it breaks on another. Once something wrong happens, it is extremely difficult to identify the problem and rectify it within a short time.

## **Solution for this problem**

We are going to test three different tools in order to deduce which of them can mend these issues in a better way. AWS CodePipeline is an amazon cloud website, which does it all automatically online.Jenkins is a well known free service which is customized by many companies to fit their needs. The newest tool of Microsoft is called gitHub actions and it cooperates right with places of code storing. Also in our research we will integrate Docker. Docker places software in containers that works everywhere in the same way. This eliminates the issue of: it works on my computer and not on yours

## **What We Want to Do**

Our primary objectives are to locate the most suitable tool depending on the kind of project and the size of the company. We would like to demonstrate how software delivery can be reduced to hours and minutes in some cases when compared to weeks and would like to prevent errors by showing how automatic testing can detect bugs before they have the opportunity to reach the customer. To save money, we intend to show how much time companies can save through this tool.

## **How We Will Compare**

We are going to test each of the tools with the same app and to check the important things. We will test the time required to set it up, the speed at which they operate, their ease of use, the cost, also their reliability and how flexible they are.

## **What We Expect to Learn**

This research will enable us as to the most appropriate manner of establishing automatic software delivery. We shall find how far the companies can save money and how the speed of delivery can be given a boost to software deployment. We will be looking at how automation generates fewer bugs. Above all, we will make proper guidelines through which organizations can choose the appropriate tool.

## **Conclusion**

A comparison of AWS CodePipeline, Jenkins and GitHub Actions to Docker will give a comprehensive guide in achieving better software delivery. The given study examines actual issues of companies on a daily basis and provides practical answers to these questions. The consequences of this study can save hours of work and create software that is much improved and faster, as well as can be used by developers and companies when they are to choose the appropriate tools.