
AUTOMATION AND DEPLOYMENT OF WEBSITE USING DEVOPS PIPELINE

Ujwal M ^{*1}, Mrs Shivaleela S^{*2}

^{*1}Student, Department of Master of Computer Applications, Dr. Ambedkar Institute of Technology, Bangalore, Karnataka, India.

^{*2}Assistant Professor, Department of Master of Computer Applications, Dr. Ambedkar Institute of Technology, Bangalore, Karnataka, India.

ABSTRACT

Websites play an important role in connecting people in the digital age. Hosting a website on a worldwide scale necessitates numerous processes and upgrades. Many manual tasks were required behind the scenes of the website, which might be automated using devops. The primary goal of integrating DevOps is to seamlessly combine the development and operations environments. During the experimentation phase, the development and deployment process may appear simple. However, if not carefully designed, deploying and using such models can result in complicated and time-consuming approaches that require significant and expensive maintenance, improvement, and monitoring efforts. These efforts use the principles, practises, and tools of continuous integration (CI) and continuous delivery (CD) to reduce waste, support rapid feedback, uncover hidden technical debt, and improve value delivery and maintenance. CI and CD are abbreviations for continuous integration and continuous delivery.

Keywords: Continuous Integration, Continuous Deployment, Jenkins, Docker, Devops, Github.

I. INTRODUCTION

Construction and Management Managing the numerous source code, fundamentally, requires each developer to work within the provided time, following a certain timetable, in order to design and develop the finest potential website. It allows you to maximise your creative potential. It takes a great deal of effort and time, but it also has a lot of potential. The 'Automation of website using devops' was created to overcome the issues that plagued the previous manual system. This software is designed to eliminate, and in some cases, decrease, the problems that the current system has. DevOps is a set of software and systems engineering processes and technologies. Software engineering is a field devoted to the development of tools and processes that enable the building and usage of complex software systems.

II. METHODOLOGY

The difficulties and obstacles of designing and distributing software applications from developers to clients are many in this project. To solve this problem, we must employ devops automation, which is a cost-effective, secure, and secure method of releasing software applications. We must lessen problems for both developers and clients in this project. The goal of the project is to remove manual intervention, making it easier to prevent mistakes during construction deployments.

III. ARCHITECTURE

This architecture show cases how automation using devops can help to build the product end to end from developer side to client side. In this architecture a group of developers develop the product application using shared repository tools like github and once the push the code in github the jenkins get notified and triggers the build steps continuous integration once the continuous integration completes it will be published to docker image and deploy in development side server for validation and then it is shipped to client side via docker hub and the client extracts the docker image from the docker hub and host in client server and the website will be running for end users.

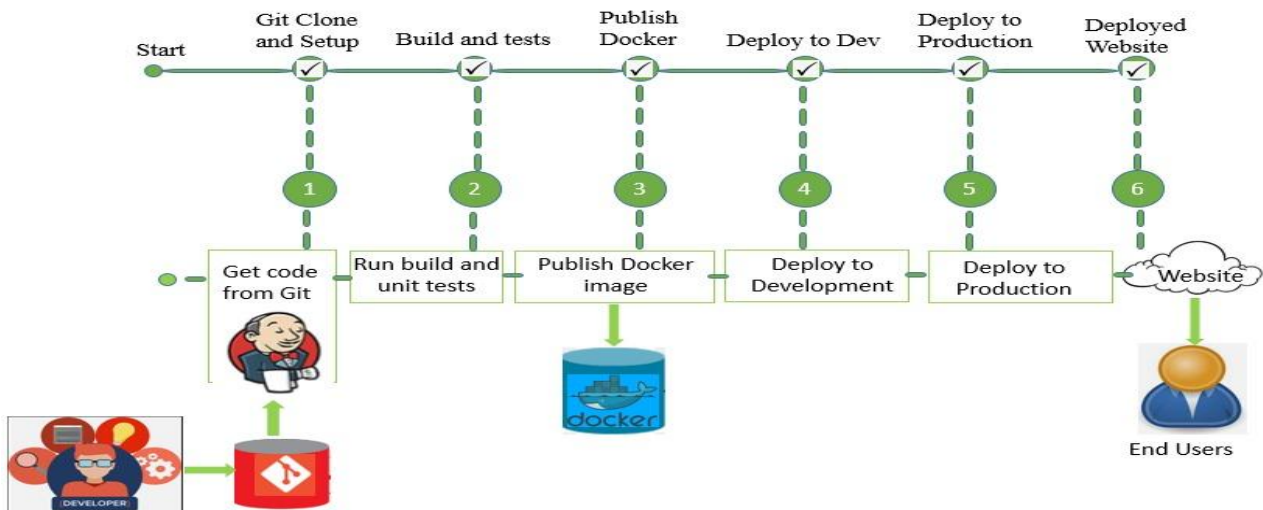


Figure 1: Architectural Design

Docker

A completely open platform for creating, deploying, and moving applications, Docker is based on boxes. By using containerization, you may build a box that contains your utility, any binaries or libraries it needs, and any configuration information.

Jenkins

Pipelines make it easier to combine various code branches into a single main branch by automatically testing and reporting in real time on changes to individual files inside a larger code base. They also develop the software, automate testing of their builds, prepare the code base for distribution (delivery), and finally distribute the code to containers, virtual machines, bare metal servers, and cloud servers. They also quickly find bugs in code bases. Jenkins is available in a number of commercial variations. Only the upstream open source project is covered by this definition.

Yum Configuration

The configuration file for yum and other programmes that are related can be found in the /etc/yum.conf directory. This file must contain at least one section that allows you to specify Yum parameters that have global effects. It may contain additional sections that allow you to specify repository-specific choices.

CI/CD Pipeline

The team can deliver codes rapidly and effectively thanks to the implementation of CI/CD. The procedure improves the team's flexibility, productivity, and self-assurance. The most popular DevOps tool for CI/CD pipelines is Jenkins.

IV. EXISTING SYSTEM

Clients and developers lose trust in the product and the software application developers as a result of improper deployments in their systems. It is also frequently challenging for them to obtain real-time software applications on time, which prevents them from deploying the application products on schedule in response to market demand. During the software development process, there will be several manual interventions and blunders.

V. PROPOSED SYSTEM

The development of software applications can use the new automation deployment utilizing devops technology, which has the advantages of decentralization and tamper ability, to make up for the functionality that the software application at this moment lacks. Continuous Integration and Continuous Deployments are the two key entities interacting with the system. The system is constructed by continuous integration. Its integration is based on developer's viewpoints. Its main goal is to find a solution to the issue of securely transferring software applications from developer's systems to client's systems.

VI. CONCLUSION

The goal of this project is to eliminate manual involvement and time consumption while shipping software from developers to clients, regardless of their geographic location. It makes no difference what kind of product is created; what counts is how quickly it reaches the customer. Devops technology enables fresh upgrades and functionality to be introduced on corporate websites every day in this quickly expanding technological world. Days of work burden are transformed into hours of work load using Docker Hub, Github, Jenkins continuous integration and continuous deployment pipeline. As a result, Agile principles/values, as well as DevOps practises and tools, are strongly recommended in order to provide continuous delivery and co-creation of value to customers, improve model quality, reduce waste, emphasise the importance of supporting a rapid feedback loop, accommodate early changes, and investigate the hidden technical debt that leads to a massive increase in real-world system operational costs. Future work could leverage our strategy to focus on specific parts of complicated systems and develop helpful tools to aid in model testing.

VII. REFERENCES

- [1] Vidroha Debroy, Seneca Miller, "Overcoming Challenges with Continuous Integration and Deployment Pipelines When Moving from Monolithic Apps to Microservices", IEEE Software, IEEE, ISSN: 2169-3536, Vol. 37, Issue: 3, Feb 2020, pp.21-29.
- [2] Keheliya Gallaba, "Improving the Robustness and Efficiency of Continuous Integration and Deployment", 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), IEEE, ISSN: 2576-3148, Vol. 10, Dec 2019, pp.619 - 623.
- [3] Alexander Poth, Mark Werner, and Xinyan Lei, "How to Deliver Faster with CI/CD Integrated Testing Services?", European Conference on Software Process Improvement, Springer, ISBN: 978-3-319-97924-0, Vol. 896, Aug 2018, pp.401 - 409.
- [4] Valentina Armenise, "Continuous Delivery with Jenkins: Jenkins Solutions to Implement Continuous Delivery", Release Engineering (RELENG), 2015 IEEE/ACM 3rd International Workshop, 19 May 2015, Date Added to IEEE Xplore: 30 July 2015.
- [5] M. Shahin, M. Ali Babar, and L. Zhu, "Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices", IEEE Access, 2017.