

RANCANG BANGUN SISTEM CONTINUOUS INTEGRATION (CI) DAN CONTINUOUS DELIVERY (CD) MENGGUNAKAN *JENKINS* PADA APLIKASI BERBASIS WEB

I Wayan Widiyantara¹, Gede Putu Megayasa², Luh Putu Risma Noviana³

¹Universitas PGRI Mahadewa Indonesia, Denpasar, Indonesia; xwidiyantara@gmail.com

²Universitas PGRI Mahadewa Indonesia, Denpasar, Indonesia; megayasa@mahadewa.ac.id

³Universitas PGRI Mahadewa Indonesia, Denpasar, Indonesia; risma@mahadewa.ac.id

ARTICLE INFO

Article history:

Received April 16, 2024

Revised April 23, 2024

Accepted April 25, 2024

Available online April 30, 2024

Keywords: Cloud Computing, Google Cloud, Serverless, Cloud Translation API

Copyright ©2023 by Author. Published by Lembaga Pengembangan Pembelajaran, Penelitian, dan Pengabdian Masyarakat Universitas PGRI Mahadewa Indonesia

Abstract. The evolution of technology resulting on application development that is also keeps improving where demand of the application software is increasing, a tech company always developing and updating their application that is already developed. Development process takes a longer time due to the complicated process especially in build and deployment process so the application software can be enjoyed by people around the internet. In order to these problems, this research is assisting to build and develop a system that could automate all of the build and deployment process so the process won't be done with human interference that can lead up into some human errors and also downtime so the application that being deployed can't be accessed due to these errors, RAD or Rapid Application Development method was used in order to develop this system with an eye to make the development process a lot easier and faster. DevOps Engineer assists in the Continuous Integration implementation process which is expected to shorten the time of the entire SDLC (Software Development Life Cycle) using an open source tool called Jenkins, so that the application development process can run quickly and on time at the allotted time for all users to enjoy Internet.

PENDAHULUAN

Sebelumnya, pada proses pengembangan perangkat lunak dan penyebaran perangkat lunak kepada pengguna masih dilakukan secara terpisah dan manual pada sebagian besar perusahaan yang masih berkembang. Praktik seperti ini sudah mulai ditinggalkan karena kedua proses ini membutuhkan waktu yang cukup lama, sedangkan seorang pengembang ingin perangkat lunaknya agar selalu dalam keadaan yang terbaru. Kini, proses pengembangan aplikasi menjadi semakin cepat. Beberapa tahun lalu, pengembang aplikasi merilis versi aplikasi baru dalam hitungan tahun atau bulan. Kini sebuah aplikasi bisa memiliki versi baru hanya dalam hitungan minggu atau hari

Pada proses pengembangan perangkat lunak, penerapan praktik CI/CD telah meningkatkan efisiensi proyek. Dengan menerapkan praktik CI/CD, fitur baru aplikasi disebarkan kepada pengguna di setiap pengiriman sprint yang akan selalu melewati tahap uji coba sebelum dikirim ke pengguna. Otomatisasi telah memungkinkan untuk melakukan uji coba yang lebih efisien, sehingga mempercepat proses pengembangan serta menghasilkan aplikasi yang lebih handal (Arachchi & Perera, 2018). Saat ini ada praktik yang sering digunakan oleh perusahaan yang bernama CI / CD

Menurut (Shahin et al., 2017) CI/CD adalah industri pengembangan perangkat lunak yang memungkinkan sebuah organisasi untuk lebih sering dan lebih baik dalam merilis sebuah fitur dan produk yang baru. Penerapan CI/CD juga menghasilkan beberapa dampak yang positif yaitu, mengurangi waktu

pembuatan dan pengujian dalam Continuous Integration (CI), meningkatkan visibilitas pada hasil pengujian, mendukung pengujian otomatis, mendeteksi kekurangan dan kesalahan, menangani masalah keamanan dan skalabilitas, serta meningkatkan proses penyebaran sebuah aplikasi

METODE

Pokok dari penelitian ini adalah membuat sistem CI/CD pada aplikasi website untuk mempermudah dalam pengembangan aplikasi website dan mempercepat proses update aplikasi website. Dalam penelitian ini menggunakan metode sebagai berikut:

1. Metode Literatur Metode ini merupakan metode pengumpulan data dan referensi baik dari media cetak maupun media elektronik yang menunjang dalam penyusunan dan pembuatan tugas akhir di Fakultas Teknik dan Informatika Universitas PGRI Mahadewa Indonesia
2. Metode Observasi Metode ini merupakan metode pengumpulan data dengan cara pengamatan terhadap alat-alat dan data yang akan digunakan untuk membuat Rancang Bangun Sistem Continuous Integration (CI) Dan Continuous Delivery (CD) Menggunakan Jenkins Pada Aplikasi Berbasis Web

Dalam Sistem Implementasi CI/CD Menggunakan Jenkins pada aplikasi berbasis web menggunakan beberapa komponen perangkat keras, diantaranya sebagai berikut :

- a. Satu unit laptop asus X441U
- b. Memory RAM 4GB
- c. Harddisk 500gb
- d. SSD 250gb
- e. Processor Intel core i3

Perangkat lunak (software) yang dibutuhkan dalam Merancang Bangun Sistem Continuous Integration (Ci) dan Continuous Delivery (CD) Menggunakan Jenkins pada aplikasi berbasis web ini adalah sebagai berikut :

- a. Jenkins adalah sebuah open source automation server untuk mengotomatisasi tugas-tugas didalam proses continuous integration dan continuous delivery pada perangkat lunak. Jenkins merupakan aplikasi berbasis Java yang dapat dipasang dari repository Ubuntu atau dengan mengunduh dan menjalankan file Web application ARchive (WAR), sebuah koleksi file yang sudah lengkap dan tinggal dijalankan di sebuah server
- b. GitHub adalah layanan host web yang digunakan untuk meletakkan kode aplikasi yg dibuat oleh programmer dan untuk mengatur versi kode aplikasi
- c. Visual Studio Code merupakan sebuah aplikasi editor code open source yang dikembangkan oleh Microsoft untuk sistem operasi Windows, Linux, dan MacOS. Visual Code memudahkan dalam penulisan code yang mendukung beberapa jenis pemrograman, seperti C++, C#, Java, Python, PHP, GO
- d. Kubernetes adalah platform open source untuk mengelola kumpulan kontainer dalam suatu cluster server
- e. Docker merupakan open-source platform yang digunakan untuk pengembangan, shipping, dan menjalankan aplikasi. Docker memungkinkan sebuah aplikasi dapat dipisahkan dari infrastruktur utama karena Docker dibangun berdasarkan teknologi container.

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah agile software development methods, dengan menggunakan Model Scrum. Adapun tahapan-tahapan scrum yang dilakukan, sebagai berikut :

- a. Backlog Menyusun rincian prioritas pada fitur-fitur yang akan dibangun pada sistem CI/CD
- b. Sprints Menyusun kegiatan yang akan dilakukan untuk memenuhi kebutuhan yang ditetapkan dalam backlog dengan durasi realisasi selama 30 hari kerja
- c. Scrum Meeting Menyelenggarakan rapat dengan tim yang telah ditunjuk untuk membahas kemajuan

kegiatan pengembangan sistem CI/CD

- d. Demos Menunjukkan fitur-fitur software yang telah dihasilkan untuk dievaluasi oleh pengguna sesuai dengan waktu yang telah ditentukan

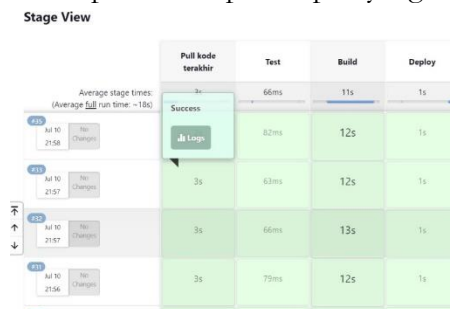
HASIL DAN PEMBAHASAN

Dalam implementasi Jenkins untuk otomatisasi langkah-langkah (pipeline) dari aplikasi. Pipeline ini memiliki beberapa stage (tahapan) yang dimulai dari mengirim notifikasi untuk build, dilanjutkan dengan checkout code dari Version Control System (Github), lalu code review menggunakan Docker, Apabila code review lolos dilanjutkan dengan build image menggunakan Docker dan dilanjutkan mengirim notifikasi build selesai dengan status selesai. Seluruh langkah diatas berjalan secara otomatis tanpa campur tangan tim pengembang maupun operasi. Dengan penggunaan jenkins dan juga docker sebagai platform otomatisasi dan virtualization, maka alur kerja dapat distandarisasi, baik standarisasi dari bentuk alur kerja (pipeline), hingga penyetaraan environment. Hal ini dapat menghilangkan adanya dependensi antara tim pengembang dan operasi

Setelah dilakukan penelitian, penulis berhasil membangun CI/CD pipeline dan berhasil mendeploy aplikasi website yang digunakan dalam penelitian ini secara semi otomatis ke Kubernetes. Untuk mengimplementasikan sistem CI/CD pada aplikasi website HMPS FTI melakukan beberapa langkah sebagai berikut :

1. Menyiapkan docker image pada docker file
2. Membuat perintah untuk mendeploy kode aplikasi terbaru pada GKE (Google Kubernetes Engine)
3. Membuat CI/CD pipeline di aplikasi jenkins
4. Menjalankan CI/CD pipeline pada aplikasi website HMPS FTI.

Dengan implementasi CI/CD pipeline pada aplikasi HMPS FTI yang berbasis web proses deployment menjadi lebih cepat dan meningkatkan kinerja. Selain itu CI/CD memberikan hasil yang lebih teliti dengan penemuan *bug* pada saat pengujian. Dengan implementasi CI/CD alur kerja dapat distandarisasi, baik standarisasi dari bentuk alur kerja (pipeline) maupun penyetaraan environment. Sehingga hal ini dapat menghilangkan adanya dependensi antara tim pengembang dan operasi. Berikut penulis lampirkan tampilan CI/CD pipeline yang menampilkan tahap- tahapan yang terjadi di CI/CD pipeline



Gambar 1. Fungsional CI/CD Pipeline



Gambar 2. Hasil Website yang Berhasil Dideploy melalui CI/CD Pipeline.

Dari hasil percobaan penulis sebanyak 7 kali perubahan aplikasi yang dilakukan, semua perubahan berhasil di-eksekusi oleh CI/CD pipeline tanpa error. Adapun detail dari proses percobaan tersebut seperti gambar berikut :

Tabel 1. Hasil Pengujian

| Percobaan | Waktu yang Dibutuhkan oleh CI/CD |
|-----------------------------------|----------------------------------|
| Mengubah Menu Navbar | 23 detik |
| Menghapus Background | 19 detik |
| Mengubah Warna Tulisan | 23 detik |
| Menambahkah Berita | 19 detik |
| Mengubah Logo Pada Halaman Berita | 18 detik |
| Menambahkan berita yang kedua | 19 detik |
| Menghapus Colum Berita Kanan | 18 detik |

SIMPULAN

Dari hasil penelitian didapatkan kesimpulan bahwa proses deployment menggunakan CI/CD mampu mempersingkat proses dan meningkatkan kinerja. Selain itu CI/CD memberikan hasil yang lebih teliti dengan penemuan bug pada pengujian tersebut. Dengan implementasi CI/CD alur kerja dapat distandarisasi, baik standarisasi dari bentuk alur kerja (pipeline) maupun penyetaraan environment. Sehingga hal ini dapat menghilangkan adanya dependensi antara tim pengembang dan operasi

DAFTAR PUSTAKA

- Arachchi, S. A. I. B. S., & Perera, I. (2018). Continuous Integration And Continuous Delivery Pipeline Automation For Agile Software Project Management. Mercon 2018 - 4th International *Multidisciplinary Moratuwa Engineering Research Conference*.
<https://doi.org/10.1109/Mercon.2018.8421965>
- Shahin, M., Ali Babar, M., & Zhu, L. (2017). Continuous Integration, Delivery And Deployment: A Systematic Review On Approaches, Tools, Challenges And Practices. *IEEE Access*, 5(Ci), 3909–3943.
<https://doi.org/10.1109/ACCESS.2017.2685>