

# BAB I

## PENDAHULUAN

### A. Latar Belakang

Dalam siklus pengembangan perangkat lunak, tahap terakhir yang dilakukan agar perangkat lunak dapat di gunakan oleh *user* adalah *software deployment*. [1] Menurut riset yang dilakukan oleh *DevOps Research and Assesment* (DORA) pada tahun 2018 dan 2019, *software deployment* merupakan salah satu tolak ukur pengukuran *software delivery and operational* (SDO) *performance*. SDO *performance* adalah alat ukur yang di teliti oleh DORA untuk memberikan tingkat ukur terhadap efektifitas praktek *deployment* dan *delivery*. Dalam riset tersebut, *continuous integration* (CI), *continuous deployment* (CD) dan *continuous delivery* merupakan salah satu relasi prediktif untuk hasil SDO *performance*. [2] Dalam pengembangan perangkat lunak, proses *continuous integration* (CI) merupakan proses pengabungan atau penambahan fitur atau kode kedalam *repository* utama. Sedangkan proses *continuous deployment* (CD) adalah proses yang dilakukan secara otomatis dijalankan setelah proses *build* dan *test* berhasil dalam CI. Proses tersebut akan melakukan perilisan otomatis hasil *build* kedalam lingkungan *staging* atau produksi.

Keuntungan yang di dapat dalam mengadaptasi sebuah *CI/CD pipeline* adalah proses perilisan perangkat lunak menjadi lebih efisien dan konsisten. Selain itu keuntungan lain adalah dari segi waktu pengembang dapat menjadi lebih efisien tanpa harus melakukan proses perilisan secara manual. Dengan menghemat waktu tersebut maka sumber daya manusia yang sebelumnya terpakai untuk proses manual tersebut dapat di alihkan kepada kebutuhan yang lebih penting. Kelemahan dari sistem ini mengharuskan perusahaan mengeluarkan waktu dan biaya ekstra di awal untuk melakukan integrasi otomasi tersebut.

Menurut riset yang sama pada tahun 2021, organisasi yang melakukan transformasi *DevOps* dengan mengadaptasi *continuous delivery* cenderung memiliki proses kerja yang lebih berkualitas, memiliki resiko rendah dan hemat waktu dan biaya. [3] Penghematan yang di dapat dari segi waktu adalah, waktu *developer* dapat di alihkan ke tugas yang lebih penting. Penghematan biaya didapatkan dari peningkatan efektifitas waktu *developer*. Selain dari hasil riset yang dilakukan DORA, kondisi lapangan saat ini masih banyak pihak yang melakukan proses *deployment* dengan cara manual. Cara manual tersebut antara lain: *pull repository* maupun unggah berkas *build* dari *source code*. Berdasarkan hasil riset dan pengamatan tersebut, diperlukan sebuah solusi berupa implementasi *continuous integration, deployment* dan *delivery* untuk sebuah proyek perangkat lunak. Untuk memenuhi tolak ukur penilaian SDO *performance* maka akan dibuat sebuah implementasi CI/CD dengan objek penelitian sebuah *microservices*. *Microservices* tersebut merupakan sebuah sistem yang bertujuan untuk melakukan kompres gambar melalui REST API. Dengan menerapkan konsep CI/CD tersebut, maka akan dibuat sistem integrasi otomatis (CI) dan *deployment* otomatis (CD) untuk menghemat waktu, biaya dan resiko dalam proses pengembangan selanjutnya.

## **B. Rumusan Masalah**

Rumusan masalah menjadi pokok permasalahan yang akan dijawab dalam penelitian ini. Rumusan masalah ini diturunkan dari latar belakang yang telah di tulis.

1. Bagaimana implementasi proses pengembangan perangkat lunak yang terdiri dari proses integrasi *source code, testing, dan deployment* perangkat lunak secara manual menjadi otomatis?
2. Bagaimana membangun sebuah solusi CI/CD *pipeline* menggunakan perangkat sumber terbuka?
3. Bagaimana dampak otomasi yang diperoleh yang ditinjau dari segi waktu, biaya dan resiko?

### **C. Batasan Masalah**

Agar fokus penelitian tidak menyimpang dari rumusan yang telah di buat, maka batasan masalah pada penelitian ini adalah sebagai berikut:

1. Penelitian tidak membahas proses pengembangan perangkat lunak melainkan hanya proses pengembangan dan implementasi CI/CD.
2. Penelitian hanya membahas secara garis besar mengenai fungsi dari perangkat lunak yang digunakan sebagai objek penelitian sebagai informasi konteks bagi pembaca.
3. Layanan penyedia komputasi awan pihak ketiga yang digunakan adalah Microsoft Azure dengan memanfaatkan *education license* yang didapatkan bersama email students UAJY.
4. Penelitian yang dikembangkan terbatas terhadap bahasa pemrograman dan layanan pihak ketiga seperti: Bahasa Pemrograman Go, *Code Repository* GitHub, *Continuous Integration* Drone.io dan *Continuous Deployment* Agro CD.
5. Sistem orkestrator yang digunakan adalah Kubernetes.
6. Implementasi menggunakan bahasa pemrograman dan layanan pihak ketiga lain dapat dilakukan dengan modifikasi namun tidak menjamin keberhasilannya.

### **D. Tujuan Penelitian**

Penelitian ini dilakukan dengan tujuan sebagai berikut:

1. Memberikan rancangan perubahan proses pengembangan perangkat lunak yang terdiri dari integrasi *source code*, *testing* dan *deployment* menggunakan CI/CD sehingga proses yang sebelumnya dilakukan secara manual menjadi proses otomatis.
2. Memberikan hasil implementasi berupa kode tentang penggunaan perangkat lunak sumber terbuka untuk membangun sebuah *CI/CD pipelines*.

3. Memberikan analisis dampak otomasi yang diperoleh yang ditinjau dari segi waktu, biaya dan resiko.

## **E. Metode Penelitian**

### a. Studi Literatur

Studi literatur dilakukan untuk mendapatkan referensi yang berkaitan dengan proses penelitian dan penulisan tugas akhir. Materi referensi yang dikumpulkan adalah materi yang berkaitan dengan proses *CI/CD pipelines*. Sumber referensi yang digunakan dalam melakukan studi literatur adalah: artikel, jurnal, penelitian terdahulu dan buku.

### b. Analisis Kebutuhan Penelitian

Tahap ini merupakan tahapan yang dilakukan untuk menganalisis kebutuhan dasar yang harus disiapkan sebelum memulai penelitian. Analisis kebutuhan penelitian dilakukan dengan mencari kebutuhan seperti *tools* yang digunakan. Dengan adanya analisis kebutuhan proses penelitian akan memiliki kerangka sebagai acuan penelitian.

### c. Pengembangan Penelitian

Setelah mengumpulkan kebutuhan penelitian, akan dilanjutkan pembuatan implementasi pengembangan *CI/CD*. Tahapan ini dilakukan dengan mengikuti kerangka penelitian yang telah di rumuskan.

### d. Pengujian Penelitian

Tahap pengujian adalah tahapan untuk memastikan implementasi yang telah dilakukan di tahap pengembangan dapat dijalankan dengan baik. Tahapan ini akan menguji keberhasilan implemetasi dan pengujian efisiensi otomasi.

### e. Evaluasi Penelitian

Tahap evaluasi adalah tahapan yang berisi *feedback* dan hasil yang didapat dari tahap pengujian. Hasil yang di dapat dalam tahap

pengujian akan di analisis. Pada tahap ini juga akan dilakukan evaluasi terkait hasil analisis yang di dapat.

## **F. Sistematika Penulisan**

Penulisan tugas akhir ini dilakukan dengan menyusun bagian penulisan menjadi 6 (enam) bab.

### **BAB I PENDAHULUAN**

Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, metode penelitian dan sistematika penulisan.

### **BAB II TINJAUAN PUSTAKA**

Bab ini berisikan penelitian-penelitian terdahulu yang menyangkut dengan penelitian yang dilakukan. Dalam bab ini terdapat juga table perbandingan antara penelitian yang dilakukan dengan penelitian-penelitian terdahulu.

### **BAB III LANDASAN TEORI**

Bab ini berisikan penjelasan mengenai teori-teori yang menyangkut penelitian yang dilakukan.

### **BAB IV ANALISIS DAN PERANCANGAN SISTEM**

Bab ini berisi analisis dan perancangan sistem yang di kembangkan.

### **BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab ini berisi hasil implementasi dan pengujian dari system yang telah di rancang sebelumnya.

### **BAB VI PENUTUP**

Bab ini berisi kesimpulan yang diambil dari penelitian yang dilakukan serta saran untuk membangun pengembangan selanjutnya.