

BAB I. PENDAHULUAN

A. Latar Belakang

Dalam proses pengembangan sebuah aplikasi, pengembang harus melalui tahapan *framework activities*. Tahapan berupa *communication*, *planning*, *modelling*, *construction*, dan tahapan terakhir yang harus dilalui sebelum aplikasi sampai kepada *user* adalah *deployment*. [1] *Deployment* adalah proses yang penting dalam pengembangan aplikasi karena memastikan bahwa aplikasi dapat digunakan oleh *end user*.

Dalam pengembangan perangkat lunak, proses perilisasi aplikasi atau *deployment* sering kali menjadi hambatan. Pengembangan perangkat lunak secara *minor* dan frekuensinya sering akan sangat merepotkan pengembang dalam waktu perilisannya jika dilakukan secara manual. Pengembang harus melakukan langkah-langkah manual dan berulang, proses *build*, *test* dan *deploy* secara terus menerus, meskipun perubahannya adalah *minor*. Hal ini tidak hanya memakan waktu dan sumber daya, tetapi juga rentan terhadap kesalahan manusia. Maka dari itu, perlu mengubah proses dari yang awalnya manual menjadi otomatis, sehingga proses yang terjadi lebih efektif dan efisien. [2] Melalui penerapan CI/CD juga bisa meningkatkan produktivitas dari pembuatan aplikasi dengan metode *agile*. [3]

Continuous integration (CI) adalah praktik yang mengharuskan pengembang untuk menggabungkan kode secara berkala ke dalam *repository*. Melalui proses *continuous integration* (CI), setiap kali penggabungan kode dilakukan, pengecekan juga akan dilaksanakan untuk memastikan kualitas dan keamanan kode. Proses yang terjadi dalam CI adalah proses *integration*, *build* dan *testing* secara otomatis. Sedangkan, *continuous deployment* (CD) adalah praktik yang mengharuskan aplikasi yang telah dibangun di *continuous integration* (CI) untuk didistribusikan baik ke lingkungan *staging* atau *production* secara otomatis atau disebut otomatisasi. [4]

Pada *report* yang dibuat oleh *DevOps Research and Assesment*

(DORA) dan Google Cloud pada 2022, *software delivery performance* (SDP) akan menilai efektivitas proses *deployment*. Berdasarkan *report* tersebut, *continuous integration* (CI) dan *continuous deployment* (CD) adalah indikator yang sangat berpengaruh dalam *software delivery performance* (SDP). [5] *Report* tersebut merupakan program terbesar dan terpanjang yang berupaya untuk memahami kemampuan yang mendorong nilai dari *software delivery performance* yang dijalankan setiap tahunnya. Salah satu parameter yang ada dari *report* tersebut adalah *lead time for changes* yang mengukur waktu yang dibutuhkan untuk menerapkan perubahan kode dari saat perubahan itu dibuat atau *code commit*, hingga perubahan tersebut berhasil berjalan. Salah satu faktor yang sangat mempengaruhi *lead time for changes* adalah dari penerapan CI/CD.

Melalui permasalahan yang dihadapi dalam proses perilisan yang dilakukan secara manual atau tradisional dan penelitian yang telah dilakukan bahwa CI/CD mempengaruhi performa di dalam SDP. Maka akan dilakukan pengujian dari implementasi CI/CD mengenai waktu atau *lead time for changes*, biaya, sumber daya dan pemanfaatan paralel yang dibutuhkan dalam penerapannya yang kemudian akan dibandingkan dengan yang bukan CI/CD. Obyek yang akan diteliti adalah aplikasi web menggunakan React dan Express dalam pengembangannya. Hal ini dilakukan untuk membuktikan CI/CD memiliki efisiensi dan performa yang lebih baik daripada *deployment* secara manual atau tradisional.

B. Rumusan Masalah

Berdasarkan latar belakang masalah di atas dapat dirumuskan masalah sebagai berikut:

1. Bagaimana menilai performa efisiensi waktu dari *pipeline* CI/CD dibandingkan dengan proses manual dalam *deployment* aplikasi web?
2. Bagaimana menilai penggunaan sumber daya dari *pipeline* CI/CD dibandingkan dengan proses manual dalam *deployment* aplikasi web?
3. Bagaimana menilai biaya yang dibutuhkan dari *pipeline* CI/CD dibandingkan dengan proses manual dalam *deployment* aplikasi web?
4. Bagaimana pengaruh kemampuan paralel dari CI/CD dengan kerja sama tim dalam *deployment* aplikasi web?

C. Batasan Masalah

Penelitian memerlukan batasan agar tidak terlalu luas dalam arah penelitiannya. Terdapat batasan penelitian dalam melakukan penelitian, yaitu:

1. Penelitian ini hanya membahas tentang pengembangan dan implementasi CI/CD, bukan tentang proses pengembangan perangkat lunak secara keseluruhan.
2. Penjelasan aplikasi yang digunakan hanya mengenai kompleksitas dari aplikasi tersebut.
3. Penelitian menggunakan Google Cloud Platform sebagai penyedia komputasi awan, dengan memanfaatkan dana dari Bangkit Academy dan *free trial* yang tersedia.
4. Penelitian ini terbatas pada *framework* Express dan *library* React, serta *database* MySQL. Penggunaan *services* pihak ketiga tertentu, seperti: Cloud Source Repositories, Cloud Build, Compute Engine, Artifact Registry dan Cloud Monitoring.

D. Tujuan Penelitian

Berdasarkan rumusan masalah yang ada, maka penelitian ini bertujuan untuk:

1. Meneliti performa waktu *pipeline* CI/CD aplikasi web dibandingkan dengan proses manual.
2. Meneliti sumber daya yang dibutuhkan untuk proses CI/CD aplikasi web dibandingkan proses manual.
3. Meneliti biaya yang dibutuhkan dalam proses CI/CD aplikasi web dibandingkan dengan proses manual.
4. Meneliti pengaruh kemampuan paralel dari CI/CD dengan kerja sama tim dalam *deployment* aplikasi

E. Metode Penelitian

1. Studi Literatur

Pada tahap ini, peneliti akan mencari dan mengevaluasi berbagai sumber informasi, termasuk artikel, jurnal, penelitian sebelumnya, dan buku yang terkait dengan implementasi CI/CD. Tujuan studi literatur adalah untuk memperoleh pemahaman mendalam tentang proses-proses terkait dan kemajuan terkini dalam lingkup penelitian ini.

2. Analisis Kebutuhan

Pada tahap ini, peneliti akan mencari kebutuhan yang perlu dipersiapkan sebelum penelitian dimulai, seperti *tools* yang digunakan dan kebutuhan lainnya. Setelah menganalisis kebutuhan ini, penelitian akan mendapatkan kerangka utama yang akan digunakan sebagai basis dari penelitian.

3. Implementasi Penelitian

Setelah merinci kebutuhan penelitian, tahapan penerapan CI/CD akan dilaksanakan. Proses ini akan mengikuti kerangka yang telah disusun sebelumnya. Penerapan ini akan berfokus pada CI/CD dan bukan pada pengembangan aplikasi keseluruhan.

4. Pengujian Penelitian

Tahapan ini akan memastikan bahwa implementasi yang dilakukan di tahap implementasi berjalan dengan baik. Tahapan ini akan dilakukan untuk menjawab persoalan dari rumusan masalah penelitian ini. Pada tahap ini akan dilakukan pengujian keberhasilan implementasi, performa dan efisiensi CI/CD.

5. Evaluasi Kinerja

Pada tahap ini, *feedback* dan hasil dari tahap sebelumnya, yaitu tahap pengujian dibahas dan dianalisis. Selain itu, evaluasi terkait hasil pengujian juga dilakukan. Tahapan ini bertujuan untuk mengidentifikasi potensi perbaikan dan peningkatan yang dapat diterapkan pada *output* yang sedang dievaluasi.

F. Sistematika Penulisan

Sistematika penulisan tugas akhir ini disusun menjadi 6 bab.

BAB I PENDAHULUAN

Latar belakang, rumusan masalah, tujuan, keuntungan, dan kekurangan penelitian dibahas dalam bab ini. Metode penelitian juga dibahas. Selain itu, sistematika penulisan juga dijabarkan

BAB II TINJAUAN PUSTAKA

Penelitian yang akan dilakukan dan penelitian sebelumnya dibahas dalam bab ini. Tabel perbandingan digunakan untuk membandingkan penelitian yang akan dilakukan dan penelitian sebelumnya.

BAB III LANDASAN TEORI

Teori-teori yang mendukung penelitian dibahas dalam bab ini. Pembahasan akan dibahas secara umum dan keterkaitan dengan penelitian.

BAB IV ANALISIS DAN PERANCANGAN SISTEM

Sistem yang akan dibuat dianalisis dan dirancang dalam bab ini. Bab ini juga akan memberikan penjelasan alur dari sistem.

BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM

Hasil implementasi dan pengujian sistem yang telah dibuat dibahas dalam bab ini. Bab ini akan menampilkan *pipeline*, layanan serta *tools* yang digunakan.

BAB VI PENUTUP

Bab ini berisi kesimpulan dari penelitian yang dilakukan dan saran untuk penelitian di masa yang akan datang.