

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan penjelasan yang telah dipaparkan dari bab pertama hingga bab keempat pada penelitian ini, penulis dapat menarik kesimpulan sebagai berikut yaitu:

1. Dengan implementasi CI/CD 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.
2. Gangguan yang ada pada proses bisnis dapat diminimalisir karena pada saat masa pengembangan dibantu dengan proses CI/CD, hal ini membuat proses *testing* dan *deployment* dapat berjalan secara otomatis tanpa campur tangan tim pengembang maupun operasi. Dengan adanya CI/CD ini waktu pengembangan dapat berjalan dengan lebih cepat dan efektif, sehingga aplikasi bisa dirilis ke klien sesuai dengan target waktu yang sudah ditentukan.
3. Berdasarkan hasil dari 5 kali percobaan, didapatkan hasil bahwa proses *deployment* menggunakan CI/CD mampu mempersingkat proses *deployment* hingga 8 menit 43 detik. Selain itu CI/CD memberikan hasil yang lebih teliti dengan penemuan 3 *bug* hanya dengan 1 kali proses percobaan, sedangkan proses tradisional hanya menemukan 2 *bug* dengan 2 kali percobaan. Total *bug* yang terdapat di dalam kode adalah 3 namun dengan proses tradisional hanya mampu mendapatkan hasil 2 *bug*, karena proses testing dilakukan secara manual. Dari hasil tersebut juga didapatkan perbandingan keberhasilan 80% : 60%, dari perbandingan ini bisa disimpulkan juga bahwa jika jumlah percobaan ditambah tingkat keberhasilan pada proses CI/CD tetap lebih besar daripada proses tradisional.

5.2 Saran

Adapun saran dari penulis untuk penelitian selanjutnya, sebagai berikut :

1. Mengimplementasi *Logging & Monitoring* di dalam proses CI/CD dengan menggunakan *tools* seperti Grafana & Prometheus agar segala proses yang ada selama *pipeline* berjalan bisa dipantau dengan lebih baik dan tercatat dengan teliti.
2. Mengimplementasikan *unit testing* agar *code coverage* pada Sonarqube dapat mendapatkan hasil yang lebih baik dan dapat mendeteksi *bug* secara sepenuhnya
3. Menggunakan *tools* yang lebih *up to date* untuk proses CI seperti menggunakan *tools* Argo Workflow dan Flux, agar bisa mengikuti perkembangan fitur yang ada

