

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan tentang perancangan dan pembangunan aplikasi *Billing System* menggunakan metode pengembangan *Devops Life Cycle* diperoleh hasil antara lain :

1. Aplikasi dibangun dengan mengkombinasikan 2 arsitektur aplikasi yaitu *microservice* dan *monolithic*. Berdasarkan hasil *monitoring server* pada CPU dan RAM aplikasi dapat berjalan dengan stabil. Penggunaan CPU tertinggi diangka 81,9%, terendah diangka 8,5% kemudian mendapatkan rata-rata diangka 29,2%. Disisi lain penggunaan RAM pada saat pemantauan hanya diangka 28%. Hal ini menyebabkan masih banyaknya sumber daya pada *server* yang belum digunakan dan aplikasi memiliki potensi *scaling* yang lebih besar lagi.
2. Implementasi aplikasi *Billing System* menggunakan metode *Sprint*. Dari 52 jumlah *backlog* dan dibagi menjadi 4 tahap *Sprint*. Didapatkan hasil *Cumulative Flow Diagram* yang cukup baik 2 diantaranya berjalan secara konsisten dan stabil. Disisi lain pada data *Burndown Chart* 3 dari 4 *sprint* yang berjalan berhasil diselesaikan sesuai target waktu yang ditentukan. Rata-rata waktu yang dibutuhkan setiap *backlog* dapat diselesaikan dalam waktu 1 hari.
3. Pengujian aplikasi *Billing System* menggunakan metode *BlackBox Testing* dengan teknik *Boundary Value Analysis* yang dilakukan oleh *tester* dari PT. Cahz Teknologi Inovasi menghasilkan 1264 *test case*. Pengujian menggunakan metode *BlackBox Testing* dengan teknik BVA membuktikan dapat menangkap masukan dari yang terkecil hingga terbesar, senyatanya *use case diagram* yang dibuat hanya memiliki 25 *use case* dengan 2 aktor berbeda. Dari pengujian ini diperoleh nilai presentase kelayakan aplikasi yang dikonversi menjadi nilai kualitatif dengan menggunakan skala likert. Nilai kelayakan dari aplikasi *Billing System*

adalah 98,97% . Sesuai dengan interpretasi skala likert, hasil pengujian aplikasi *Billing System* mendapatkan predikat “Sangat Baik”.

4. Deployment menggunakan teknik CI/CD (*Continuonus Integration/ Continuonus Deployment*) dapat diimplementasikan dengan baik, dimana proses *build-and-test* dapat mendeteksi kesalahan dengan baik dan berhasil melanjutkan proses *deployment* ke *server staging*. *Lead time test* menunjukkan waktu terpanjang terjadi pada proses *build-and-test* dengan dengan rerata 140,75 detik. Sedangkan rerata waktu yang dibutuhkan untuk proses *deployment* ke *server staging* sebesar 2 detik. Hal ini menunjukkan bahwa proses *deployment* ke *server* cukup singkat. Disisi lain proses *deployment* menggunakan teknik penjadwalan pada *server production* sangat efektif dikarenakan aplikasi dapat di *deploy* pada waktu tertentu disaat *load* aplikasi sedang rendah. *Deployment* dengan teknik ini juga sangat mudah karena konfigurasi *deployment* sudah didefinisikan diawal, proses ini memerlukan waktu rerata sebesar 35,25 detik. Hasil ini menunjukkan bahwa penerapan *DevOps* pada aplikasi *Billing System* yang memiliki kombinasi arsitektur *microservice* dan *monolithic* dapat dilakukan. Dengan menggunakan infrastruktur *DevOps* pengembangan dan perubahan aplikasi dapat dilakukan secara cepat dan ditanggapi oleh sistem.

5.2. Saran

Berdasarkan hasil evaluasi dan penelitian yang telah dilaksanakan, terdapat beberapa saran yang bertujuan untuk mempermudah para peneliti selanjutnya untuk mengembangkan, memperbaharui maupun menyempurnakan penelitian ini antara lain :

1. Dapat menggunakan metode pengembangan lain yang memiliki persyaratan yang lebih mudah dan sumber daya yang lebih murah untuk membandingkan dengan metodologi yang digunakan peneliti yaitu *DevOps Life Cycle*.

2. Pada penelitian selanjutnya dapat melakukan pembaruan dan penambahan fitur agar aplikasi dapat terus memenuhi kebutuhan pengguna.
3. Dapat mengkombinasikan metode pengujian *Blackbox Testing* yang dapat dilakukan secara otomatis, terdapat pengujian statistik, dapat melakukan pembobotan selain dari lokal dan uji kelayakannya dapat dilakukan secara langsung.