# BAB V

# SIMPULAN DAN SARAN

## 5.1 Simpulan

Jika melihat dari metrik evaluasi, yaitu dari *average response time*, *throughput*, maupun *error rate* sebelum *query* SQL dioptimalkan, Microsoft SQL Server menjadi *database* yang memiliki performa terbaik. Setelah *query* dioptimalkan, PostgreSQL memiliki performa yang lebih baik dengan *thread* yang hanya berkisar di 10 sampai dengan 50 saja. Ketika sudah masuk ke *thread* lebih dari 50, Microsoft SQL Server menjadi *database* yang memiliki performa terbaik dibandingkan dengan dua *database* lainnya. Melihat hasil performa *database* yang sudah dioptimalkan, PostgreSQL menjadi *database* yang paling direkomendasikan untuk digunakan di dalam perusahaan karena jumlah penggunanya saat ini tidak terlalu banyak, namun dengan catatan perlu dilakukan optimasi *query* SQL untuk mendapatkan performa yang optimal. Walaupun begitu, Microsoft SQL Server juga tetap dapat dipertimbangkan penggunaannya karena performa *database*-nya yang bisa lebih stabil tanpa perlu melakukan optimasi yang terlalu banyak seperti pada *database* PostgreSQL untuk mendapatkan hasil yang optimal.

Berdasarkan kesepuluh *query* yang sudah diujikan, beberapa optimasi yang dilakukan adalah melakukan *join reorder* pada tabel dengan jumlah baris data yang lebih sedikit, lalu menggunakan *index database* untuk kolom tabel yang sering digunakan untuk *filter* data pada klausa WHERE, menggunakan *partition tables* untuk tabel "selling_transactions" karena memiliki kolom "tgl_transaksi", sehingga bisa dilakukan *horizontal partition tables*, dan melakukan seleksi data dengan *subquery* atau *common table expressions* (CTE) di awal sebelum masuk ke klausa SELECT. Penggunaan optimasi ini perlu disesuaikan juga dengan *query* SQL yang diujikan agar dapat mendapatkan hasil yang lebih optimal dibandingkan sebelumnya. Jika dilihat dari kesepuluh *query*, penggunaan seleksi tabel dari awal menjadi optimasi yang paling banyak digunakan karena melakukan *filter* baris dan kolom data yang diperlukan saja dapat membantu mengurangi cukup banyak jumlah data yang perlu dibaca oleh *database*, melihat tabel yang digunakan dalam *query* hampir

semuanya berkaitan pada tabel “selling_transactions” dan “detail_transactions” dengan jumlah baris data yang cukup banyak.

## 5.2 Saran

Dalam proses penelitian, ada beberapa hal yang dapat ditingkatkan untuk penelitian serupa selanjutnya. Beberapa saran untuk penelitian berikutnya yang dapat dilakukan untuk penelitian selanjutnya terkait dengan pengujian *query database*, yaitu:

1. Menambahkan tampilan *user interface* (UI) pada *website* untuk pengoptimalan *query* SQL agar *user* dapat langsung memasukkan *query* SQL tersebut untuk dioptimalkan dengan lebih mudah.
2. Menambahkan penggunaan algoritma *machine learning* untuk membantu melakukan optimasi *query* SQL.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA