

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada masa modern ini penggunaan teknologi informasi banyak membantu kegiatan manusia terlebih memudahkan kegiatan dalam bisnis. Terlebih lagi dengan adanya fenomena pandemi dunia COVID-19 mendorong para pelaku usaha menggunakan teknologi informasi untuk menopang bisnis mereka. Hasil survey dari Badan Pusat Statistik Indonesia (BPS) terhadap lebih dari 34.000 pelaku bisnis di Indonesia terdapat sekitar 47% pelaku bisnis yang menggunakan internet dan teknologi informasi lainnya untuk menopang bisnis mereka dikala serangan pandemi dunia COVID-19 [1]. Hasil survey ini memperlihatkan bahwa pengimplementasian teknologi informasi dalam bisnis memberikan dampak positif.

Salah satu bentuk implementasi teknologi informasi adalah *database*. Sistem *database* dapat membantu mempermudah proses pengerjaan tugas dan jalannya bisnis, contohnya dalam bidang pendidikan *database* digunakan untuk menghimpun data yang dimana nantinya data inilah yang akan diolah dengan sistem manajemen informasi dan menghasilkan pengetahuan dalam membantu proses pengambilan keputusan. Untuk mencapai tujuan tersebut diperlukan juga sistem, perangkat lunak, perangkat keras dan juga orang yang dapat melakukan manajemen *database* untuk mendukung sistem *database* dalam bisnis [2].

Salah satu provider *database* adalah PostgreSQL. PostgreSQL merupakan salah satu layanan *database open-source* yang terdepan [3]. Perusahaan X yang merupakan salah satu perusahaan finansial yang beroperasi di Indonesia telah mengimplementasikan *database* PostgreSQL tetapi masih mengalami beberapa masalah dalam meningkatkan efisiensi dari *database* PostgreSQL mereka, dimana Perusahaan X mengalami kesulitan dalam penampungan koneksi sebelum masuk ke dalam *database*, manajemen dan mekanisme pemutusan koneksi *idle* pada *database*, efisiensi penggunaan

database tereplikasi dan pencegahan penggunaan *resource* berlebih pada satu *server database* saja.

Sejalan dengan kebutuhan perusahaan X yang merupakan salah satu perusahaan finansial yang beroperasi di Indonesia, implementasi *connection pooling* dan *load balancer* pada *database* ini menjadi solusi dari masalah yang dialami. Implementasi *connection pooling* dan *load balancer* menjadi menjawab masalah-masalah yang dihadapi Perusahaan X. *Connection pooling* diharapkan dapat menjawab masalah mengenai, penampungan koneksi kedalam *database*, mekanisme pemutusan koneksi *idle* yang terjadi dalam *database* dan juga manajemen koneksi dari *database*. Sementara *Load Balancer* dapat menjawab masalah mengenai, efisiensi penggunaan *database* tereplikasi dan pencegahan penggunaan *resource* berlebih pada satu *server database* saja.

Connection pooling pada PostgreSQL dapat dilakukan dengan menggunakan perangkat lunak *pgbouncer* dan *pgpool-II*, sedangkan *Load balancer* dapat digunakan dengan adanya perangkat lunak *pgpool-II*. Berdasarkan penelitian berjudul “*Measuring and Improving the Performance of GNU Taler on Grid’5000*”, dituliskan bahwa *pgbouncer* dapat membantu *database* PostgreSQL untuk memungkinkan adanya *connection pooling*. *Connection pooling* sendiri adalah metode yang menjaga koneksi yang aktif dalam *database* tetap terbuka sehingga tidak diperlukan penutupan dan juga pembukaan koneksi pada *database* secara berulang, *connection pooling* dalam *pgbouncer* juga memungkinkan terjadinya antrian koneksi dalam *database* sehingga *database* dapat menerima koneksi lebih dari maksimal koneksi dari *databasenya* [4].

Load balancer dalam *database* PostgreSQL merupakan proses pendistribusian lalu lintas transaksi secara efisien kedalam kelompok *database*, dimana transaksi yang dibagikan hanyalah transaksi *select query* saja. Berdasarkan penelitian dari Masaryk University berjudul “*High-availability for PostgreSQL in OpenShift*”, penggunaan *pgpool-II* sebagai

connection pooling dan *load balancing* memberikan pengaruh yang besar bagi sistem *database* PostgreSQL. *Load balancing* dari *pgpool-II* mencegah adanya kelebihan (*overloading*) transaksi pada 1 *database* saja [5]. Berdasarkan penelitian-penelitian tersebut ingin dibuktikan bagaimana performa dari *connection pooling* dan juga *load balancer* dengan mengkombinasikan *connection pooling* dari *pgbouncer* dan *load balancer* dari *pgpool-II* untuk memberikan penggunaan *best practice* untuk perusahaan X dengan menggunakan *tools benchmarking* dari PostgreSQL yaitu *pgbench*. *Pgbench* sendiri merupakan *tools* dari PostgreSQL yang digunakan untuk melakukan *bechmarking* ke *database* PostgreSQL dengan menjalankan SQL *command* secara terus menerus [6]. Hasil dari *benchmarking* akan diambil dalam bentuk *Transaction Per Second* (TPS) dan *connection latency* berdasarkan beberapa skenario dengan jumlah transaksi dan juga koneksi kedalam *database*, yang akan dibandingkan performa antara penggunaan *connection pooling* dan *load balancer* dari *pgpool-II* dengan penggunaan *connection pooling* dan *load balancer* dengan mengkombinasikan *pgpool-II* dan *pgbouncer*.

1.2 Rumusan Masalah

Adapun rumusan masalah yang akan diteliti dan dianalisa pada penelitian ini adalah:

1. Bagaimana merancang sebuah environment *database* PostgreSQL dengan fitur *connection pooling* dan *load balancer*?
2. Bagaimana mengevaluasi performa dari *connection pooling* dan *load balancer* pada *database* PostgreSQL?
3. Bagaimana *best practice* dari penggunaan *connection pooling* dan *load balancer* pada *database* PostgreSQL?

1.3 Batasan Masalah

Adapun batasan masalah yang akan diteliti dan dianalisa pada penelitian ini adalah:

1. Merancang *environment database* dengan menggunakan *database PostgreSQL* yang tereplikasi dengan menggunakan skema *database* yang sudah ada berdasarkan kebutuhan *user*.
2. Merancang *environment database* dengan fitur *connection pooling* dan *load balancer*.
3. Evaluasi performa *connection pooling* dan *load balancer* dengan metode *load test* menggunakan *tools* PGBENCH sehingga nilai yang digunakan untuk evaluasi adalah hasil benchmarking dari PGBENCH saja.
4. TPS yang dievaluasi hanya dilakukan pada *Database PostgreSQL* versi 12.
5. TPS yang didapatkan dapat dipengaruhi oleh koneksi internet digunakan.

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Merancang *environment database* berbasis PostgreSQL dengan fitur *connection pooling* dan *load balancer* sesuai kebutuhan perusahaan sesuai *best practice*.
2. Melakukan evaluasi implementasi kombinasi *tool* PGBOUNCER dan PGPOOL-II sebagai *connection pooling* dan *load balancer* pada Database PostgreSQL tereplikasi menggunakan *tool* pgbench.
3. Mendapatkan skenario terbaik dari 3 skenario kombinasi antara *tools* PGBOUNCER dan PGPOOL-II berdasarkan transaksi per detik dan *latency* dengan bantuan *tool benchmarking* pg-bench.

1.4.2 Manfaat Penelitian

Penelitian ini dilakukan untuk:

1. Dapat memberikan gambaran mengenai penggunaan *database* dalam bisnis.

2. Dapat menggambarkan peran *connection pooling* dan *load balancer* dalam *database*.
3. Dapat memberikan *best practice* dari penggunaan *connection pooling* dan *load balancer* pada *database PostgreSQL*.

1.5 Sistematika Penulisan

Dengan tujuan untuk mempermudah memahami alur hingga kesimpulan dari penelitian ini maka ditentukan sistematika penulisan penelitian ini sebagai berikut:

1. BAB I PENDAHULUAN

Pada bab ini berisikan informasi awal mengenai penelitian ini, dimana terdapat latar belakang dari penelitian ini, yang disambung dengan rumusan masalah, tujuan penelitian dan juga manfaat dari penelitian ini. Serta untuk mempermudah pemahaman pembaca terhadap penelitian ini terdapat juga sistematika penulisan.

2. BAB II LANDASAN TEORI

Pada bab ini berisikan penjelasan mengenai beberapa kata kunci yang menjadi landasan teori pada obyek penelitian kali ini yang disambung dengan *framework* penelitian sebagai pedoman dan juga *tools* yang digunakan guna membantu penelitian ini.

3. BAB III METODE PENELITIAN

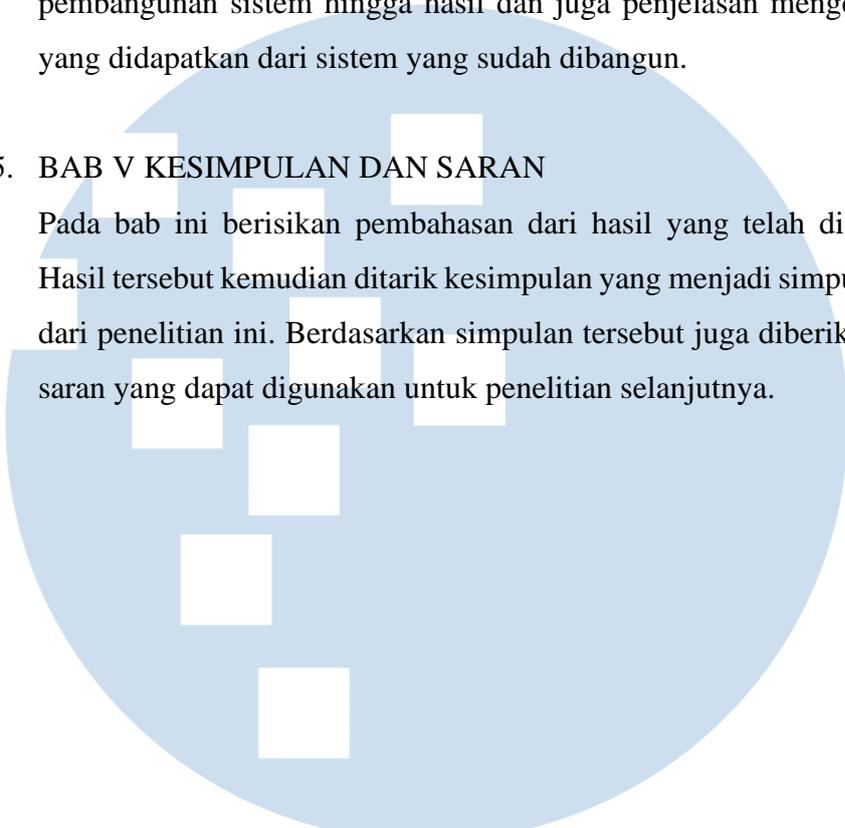
Pada bab ini berisikan informasi mengenai obyek dari penelitian. Disambung juga dengan metode-metode penelitian yang menggambarkan alur kerja dalam penelitian ini sehingga penelitian ini dapat mencapai hasil yang sesuai dan memiliki rangka pengerjaan yang sistematis.

4. BAB IV HASIL DAN PEMBAHASAN

Pada bab ini berisikan hasil dari penelitian yang berbentuk proses pembangunan sistem hingga hasil dan juga penjelasan mengenai hasil yang didapatkan dari sistem yang sudah dibangun.

5. BAB V KESIMPULAN DAN SARAN

Pada bab ini berisikan pembahasan dari hasil yang telah didapatkan. Hasil tersebut kemudian ditarik kesimpulan yang menjadi simpulan akhir dari penelitian ini. Berdasarkan simpulan tersebut juga diberikan saran-saran yang dapat digunakan untuk penelitian selanjutnya.



UMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA