

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Teknologi informasi dan komunikasi saat ini telah berkembang begitu cepat dan dapat memudahkan kegiatan manusia dalam berbagai bidang, salah satunya dalam bidang perdagangan. Dalam aktivitas perdagangan akan selalu berhadapan dengan kegiatan transaksional, monitoring dan rekapitulasi secara berkala. Untuk dapat melakukan beberapa aktivitas perdagangan tentu diperlukan sebuah sistem yang dapat memudahkan aktivitas tersebut. Sistem yang digunakan adalah sistem informasi penjualan. Sistem ini dapat melakukan berbagai aktivitas perdagangan seperti mengelola data konsumen, mengelola data *inventory* hingga dapat digunakan untuk mengelola data penjualan.

Sistem informasi penjualan umumnya memiliki berbagai fitur yang lengkap untuk melakukan segala aktivitas perdagangan. Banyaknya fitur yang disediakan di sistem maka akan berdampak pada besarnya ukuran *file* sistem tersebut. Sebagai sistem besar yang berperan penting dalam kegiatan perdagangan, tentunya sistem harus memiliki kinerja yang baik dan akses yang cepat sehingga diharapkan dapat menghadapi keadaan dimana banyaknya akses dan proses yang dilakukan oleh kasir. Supaya dapat menggunakan sistem dengan kinerja yang baik dan dapat diandalkan tentunya perlu dibangun sebuah sistem dengan arsitektur yang tepat dan menggunakan teknologi komunikasi pertukaran data yang tepat.

*Microservices* merupakan sebuah konsep arsitektur perangkat lunak yang dapat diimplementasikan untuk sistem berskala besar seperti sistem informasi penjualan. *Microservice* memungkinkan aplikasi besar dapat dipecah menjadi bagian-bagian kecil yang memiliki peran dan fungsi masing-masing yang tidak saling bergantung pada program lain, dengan API (*Application Programming Interface*) sebagai antarmuka (Newman, 2015). Selain itu, dengan membagi sistem menjadi layanan-layanan kecil dapat memberikan kemudahan dalam melakukan *maintenance*, membuat layanan lebih cepat dalam testing, dan dapat mengimplementasikan teknologi yang berbeda sesuai kebutuhan layanan masing-masing (Richardson, 2018).

Seperti halnya arsitektur lain, *Microservice* mempunyai banyak pola untuk mendesain arsitekturnya. Taibi dkk., dalam penelitiannya yang berjudul “*Architectural Patterns for Microservices: a Systematic Mapping Study*”, mengumpulkan berbagai macam pola arsitektur *microservice* dari berbagai penelitian dan makalah, untuk membandingkan pola-pola tersebut berdasarkan kelebihan dan kekurangannya. Pada hasil penelitiannya, Taibi dkk. Merekomendasikan pola *API Gateway* sebagai pola terbaik untuk diimplementasikan ke dalam arsitektur *microservices* (Taibi, Lenarduzzi, & Pahl, 2018).

*GraphQL* sebagai teknologi baru dalam komunikasi data memberikan kemudahan dan fleksibilitas dalam menyajikan data dari server ke *client*, dikarenakan *GraphQL* hanya mengambil data yang diperlukan oleh *client*, sedangkan *backend* hanya perlu membuat sebuah *endpoint* utama untuk

berkomunikasi dengan server, sehingga masalah umum dalam pengambilan data seperti *overfetching* dan *underfetching* dapat terhindar (Barus, 2019). Selain itu, berdasarkan hasil eksperimen tentang analisis perbandingan kinerja antara *RESTful API* dan *GraphQL*, (Helgason, 2018) menyimpulkan *GraphQL* memberi kemudahan dalam memilih data secara spesifik, sehingga total ukuran paket yang dikirim antara klien dan server dapat dikurangi, memungkinkan waktu respon yang lebih cepat, dan juga dapat mengurangi kerja klien.

Penggunaan *GraphQL* sebagai penghubung komunikasi data antara client-server dapat diimplementasikan sebagai *API Gateway* pada arsitektur *microservices*, yang bertugas sebagai manajemen API dari berbagai *services*, karena *GraphQL* memungkinkan *client* untuk menentukan kueri khusus mereka sendiri, dan *GraphQL single endpoint* dapat digunakan oleh banyak klien, sehingga *GraphQL* sangat pas untuk digunakan pada *microservices* sebagai *API Gateway* (Touronen, 2019).

Berdasarkan seluruh pemaparan di atas, maka penulis tertarik untuk mengangkat penelitian dengan judul “Implementasi Arsitektur *Microservices* dengan *GraphQL* Sebagai *API Gateway* pada Sistem Informasi Penjualan”.

## 1.2 Rumusan Masalah

Berdasarkan uraian dari latar belakang diatas maka didapatkan rumusan masalah sebagai berikut :

Bagaimana implementasi arsitektur *microservice* dengan *GraphQL* sebagai *API Gateway* pada sistem informasi penjualan?

### 1.3 Batasan Masalah

Berdasarkan latar belakang permasalahan yang telah dipaparkan di atas, perlu adanya batasan masalah agar penelitian yang dilakukan tetap fokus dan dapat menyelesaikan permasalahan yang ada. Maka batasan masalah dalam penelitian ini yaitu :

1. Sistem yang akan dibuat adalah sistem informasi penjualan.
2. Desain arsitektur sistem menggunakan arsitektur *microservice*.
3. Arsitektur *microservice* yang akan diterapkan adalah dengan membuat *multi database*.
4. Sistem yang akan dibuat menggunakan 4 *database*.
5. *Services* yang akan dibuat adalah *service* data konsumen, *service* data *inventory* dan *service* data penjualan.
6. *API Gateway* yang akan digunakan adalah *GraphQL*.
7. *Schema GraphQL* yang akan diterapkan adalah *Query*.
8. Penerapan *GraphQL* sebagai *API Gateway* hanya digunakan untuk mengambil data.
9. Bahasa pemrograman yang digunakan adalah *PHP* versi 7.4.26, *Javascript* dan *CSS*.
10. *DBMS* yang digunakan adalah *MySQL* versi 5.5.62.
11. Sistem dibuat dengan *framework CodeIgniter* versi 3.
12. Penelitian ini tidak membahas keamanan.

#### **1.4 Tujuan Penelitian**

Tujuan dari diadakannya penelitian ini adalah untuk membuat sebuah sistem informasi penjualan yang dapat dikembangkan tanpa mempengaruhi fitur dan modul yang ada dan supaya komunikasi data antara *server* dengan *client* dapat lebih efektif, sehingga perlu melakukan implementasi arsitektur *microservices* dengan *GraphQL* sebagai *API Gateway* pada sistem informasi penjualan.

#### **1.5 Sistematika Penulisan**

##### **BAB I PENDAHULUAN**

Bab ini berisi latar belakang masalah, rumusan masalah, ruang lingkup, tujuan, manfaat penelitian dan sistematika penulisan.

##### **BAB II TINJAUAN PUSTAKA DAN DASAR TEORI**

Bab ini berisi tinjauan pustaka dan dasar teori. Tinjauan pustaka membahas tentang kasan dari berbagai pustaka yang berhubungan dengan topik pada penelitian yang sedang dilakukan dalam penyusunan skripsi. Sedangkan pada dasar teori membahas tentang teori-teori yang digunakan dalam penelitian ini.

##### **BAB III ANALISIS DAN PERANCANGAN SISTEM**

Bab ini berisi analisa sistem berupa analisa kebutuhan sistem, analisa perangkat keras dan perangkat lunak yang dibutuhkan. Perancangan membahas tentang rancangan pemodelan, rancangan basis data, diagram alir, serta rancangan input dan output sistem.

##### **BAB IV IMPLEMENTASI DAN PEMBAHASAN SISTEM**

Bab ini berisi implementasi dari perancangan sistem yang dibahas pada bab sebelumnya sesuai dengan pemrograman dan teknologi yang digunakan. Pada bagian ini menguraikan implementasi sistem yang dianggap penting atau inti dari penelitian yang dilakukan. Pembahasan sistem menjelaskan tentang hasil setelah penelitian ini diimplementasikan.

## BAB V PENUTUP

Pada bab ini berisi kesimpulan dari penelitian dan saran untuk penelitian lain yang ingin dikembangkan dari penelitian ini.