

## BAB I

### PENDAHULUAN

#### 1.1 Latar Belakang

Semenjak diperkenalkan oleh Roy Fielding sekitar tahun 2000 pada disertasi PhD miliknya, teknologi pengembangan perangkat lunak berbasis REST (*Representational State Transfer*) adalah yang saat ini paling banyak digunakan untuk mengembangkan aplikasi berukuran personal maupun *enterprise*. Teknologi tersebut memanfaatkan sebuah protokol yang sudah disediakan oleh HTTP untuk berkomunikasi pada *database*. Implementasi dari REST yang memiliki tingkat kehandalan tinggi membuat para pengembang aplikasi web mulai berpindah dan mengimplementasikan sistem menggunakan jenis arsitektur *n-tier application* yang semulanya bertipe *client-server*.

Arsitektur *n-tier application* merupakan perantara disebut juga sebagai *middleware* atau lebih populer dengan sebutan API (*Application Programming Interface*). Melalui *middleware*, aplikasi dapat dikembangkan lebih masif karena tidak perlu membuat secara berulang untuk menjadikan aplikasi yang berbeda. Pengembang aplikasi hanya perlu mendesain ulang tampilan sesuai kebutuhan dan berinteraksi dengan data sesuai aturan yang sudah ditetapkan pada API.

PHP merupakan salah satu bahasa pemrograman *server-side* yang dapat digunakan untuk mengembangkan aplikasi REST API. Kemudahan dalam mengembangkan aplikasi menggunakan PHP disertai dengan komunitasnya yang

sangat besar, menjadikan bahasa pemrograman ini pilihan bagi pengembang aplikasi pemula maupun yang sudah berpengalaman.

Berbagai *framework* ditawarkan PHP dengan kompatibilitas dan kemampuan beragam. Jenis *framework* yang dirancang khusus untuk mengembangkan *middleware* disebut juga *microframework*. *Microframework* merupakan sebuah versi "kecil" dari *full-stack framework*. Fitur yang ditawarkan *microframework* pada kondisi *default* adalah protokol HTTP. Secara spesifik, *microframework* digunakan untuk membangun sebuah API. Beberapa *microframework* PHP yang populer saat ini diantaranya adalah FatFree, Lumen, Phalcon, dan Slim. Setiap *microframework* tersebut memiliki fitur yang diunggulkan seperti penanganan koneksi *database*, URL *routing*, kemudahan pengembangan, dan yang paling banyak diunggulkan adalah tentang performa. Banyaknya *microframework* PHP dengan metode pengembangan yang berbeda tentunya memiliki perbedaan pada segi performa.

Hal itu menjadikan pemilihan *microframework* sebagai *core* pengembangan aplikasi berbasis API sangat penting. Terlebih jika sebuah API diproyeksikan akan menangani pertukaran data atau *request client* dalam jumlah besar. Sebagai dasar pemilihan, maka pengujian performa perlu dilakukan pada setiap *microframework* untuk menentukan kesesuaian pada pengembangan aplikasi berbasis API. Dari permasalahan tersebut, pada penelitian ini akan dilakukan analisis perbandingan performa *microframework* PHP pada REST API.

## 1.2 Rumusan Masalah

Berdasarkan penjabaran dari latar belakang, maka rumusan masalah penelitian adalah sebagai berikut:

1. Bagaimana melakukan analisis perbandingan performa terhadap *microframework* PHP?
2. Apa hasil yang didapat dalam pengujian performa *microframework* PHP?

## 1.3 Ruang Lingkup

Penelitian yang dilakukan memiliki batasan masalah dan ruang lingkup sebagai berikut:

1. Aplikasi yang akan diuji berfokus untuk analisis perbandingan dan dibuat menggunakan *microframework* PHP populer pada REST API (FatFree v3.7, Lumen v7.2.0, Phalcon-micro v4.0, Slim v4.5.0).
2. Aplikasi REST API menggunakan seminimal mungkin fitur bawaan dari *microframework* PHP. Pada kasus ini adalah fitur *routing* dan koneksi *database*.
3. Pengujian performa menggunakan metode *load testing*.
4. Pengujian performa dilakukan melalui berbagai jenis pengujian meliputi plaintext, JSON *serialization*, *single query*, dan *multiple query*.
5. Data *response* hasil uji *query* berasal dari *row data* tabel city pada *database* world\_x yang didapat melalui situs resmi MySQL.

6. Akses *database* dan eksekusi *query* menggunakan PDO (*PHP Data Object*) fitur bawaan dari tiap *microframework* PHP.
7. Hasil pengujian performa diambil dari pengukuran RPS (*Request Per Seconds*) dan *latency* menggunakan alat uji wrk2.
8. Menggunakan CLI (*Command Line Interface*) sebagai antarmuka untuk melakukan pengujian.
9. Menggunakan laptop sebagai *local server* dengan *web server built-in* PHP.

#### **1.4 Tujuan Penelitian**

Tujuan dari penelitian yaitu melakukan analisis performa *microframework* PHP pada REST API untuk mendapatkan hasil perbandingan performa dari jenis pengujian plaintext, JSON *serialization*, *single query*, dan *multiple query*.

#### **1.5 Manfaat Penelitian**

Hasil penelitian diharapkan dapat bermanfaat sebagai pertimbangan dan bahan evaluasi pengembang aplikasi untuk menentukan pilihan *microframework* PHP berdasarkan performa.

#### **1.6 Sistematika Penulisan**

### **BAB I PENDAHULUAN**

Menjelaskan latar belakang masalah serta tujuan secara umum.

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

Menjelaskan apa saja yang menjadi dasar pengetahuan pada penelitian.

### **BAB III METODE PENELITIAN**

Menjelaskan tentang gambaran mengenai perancangan serta analisis dalam penelitian.

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

Menjelaskan implementasi dari perancangan dan analisis dari temuan saat penelitian.

### **BAB V PENUTUP**

Menjelaskan kesimpulan dan saran dari hasil penelitian. Kesimpulan menjelaskan rangkuman dari penelitian yang dilakukan. Saran menjelaskan tentang masukan untuk pengembangan sistem selanjutnya.