

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dewasa ini, internet sudah menjadi kebutuhan primer bagi banyak orang. Jumlah pengguna internet pada bulan Januari tahun 2024 telah mencapai 5,35 miliar pengguna (Petrosyan, 2024). Pada umumnya internet digunakan untuk mengakses *website* untuk mendapatkan informasi karena *website* merupakan media yang efektif untuk menyampaikan informasi seperti yang dijelaskan oleh Surentu dkk. (2020). *Website* tidak hanya digunakan untuk menyampaikan informasi, namun juga untuk banyak hal seperti untuk bisnis. Maharani dkk. (2021) menjelaskan bahwa *website* memberikan keuntungan dan dampak yang besar bagi para pelaku bisnis yang menerapkannya pada bisnis mereka. *Website* digunakan oleh bisnis untuk mempromosikan bisnis mereka kepada orang banyak.

Salah satu faktor yang membuat orang-orang tetap berada pada suatu *website* adalah performa dari *website* tersebut (Karlsonn, 2021). *Website* yang lamban dan memiliki performa yang buruk cenderung membuat orang-orang meninggalkan *website* tersebut. Penelitian yang dilakukan oleh Xilogianni dkk. (2022) mengungkapkan bahwa kecepatan dari *website* sangat berpengaruh kepada pengalaman pengguna. Enright (dalam Xilogianni dkk., 2022) mengemukakan bahwa *website* AutoAnything.com mengalami peningkatan penjualan *online* sebesar 12-13% karena *website* mereka yang cepat dimuat. Hal ini menunjukkan bahwa performa dari sebuah *website* merupakan hal yang penting. Oleh karena itu, *website* perlu memperhatikan berbagai faktor dengan baik agar *website* memiliki performa yang baik. Salah satu faktor yang dapat diperhatikan adalah teknologi yang digunakan untuk mengembangkan aplikasi berbasis *website*.

Dutonde dkk. (2022) menjelaskan bahwa sebuah *website* secara garis besar terdiri dari dua bagian, yaitu *front-end* dan *back-end*. Bagian *front-end* dari *website* merupakan bagian yang berinteraksi langsung dengan pengguna. Sedangkan bagian *back-end* merupakan sisi yang tidak dilihat oleh pengguna dan berguna untuk menyediakan data untuk ditampilkan pada *website*. Dikarenakan *back-end* merupakan bagian yang menyediakan data, maka performa *back-end* yang baik

akan meningkatkan performa *website*. Salah satu cara yang populer untuk menyediakan data kepada *website* adalah menggunakan REST API. REST atau *Representational State Transfer* adalah arsitektur yang digunakan untuk merancang layanan yang digunakan di berbagai platform dan lingkungan untuk mendukung interoperabilitas dan WWW (*World Wide Web*) seperti yang dijelaskan oleh Li dkk. (dalam Ehsan dkk. 2022). Sedangkan REST API (*Application Programming Interface*) merupakan jenis API yang disusun sesuai dengan batasan arsitektur REST (Glantz dan Hurtig, 2022).

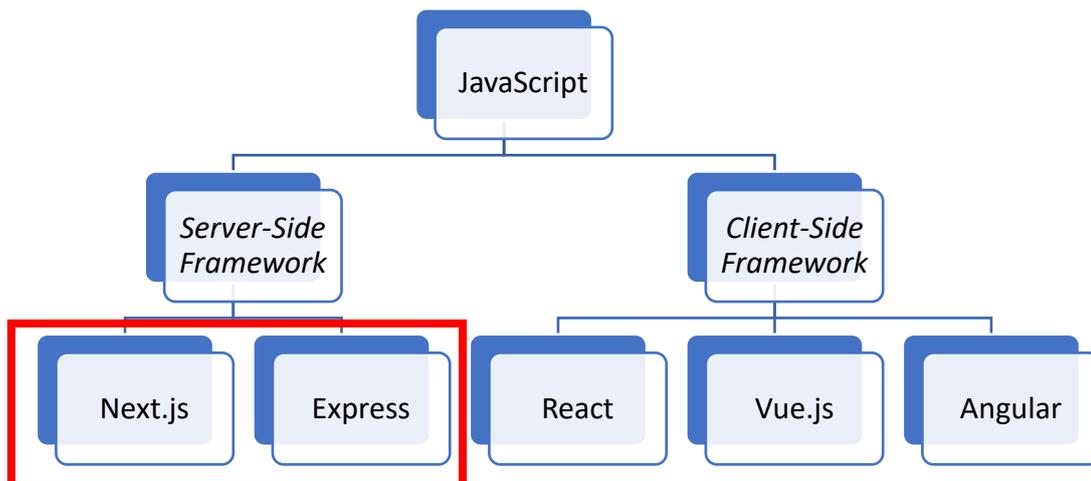
Salah satu cara untuk membangun REST API adalah dengan menggunakan *framework*. *Framework* atau kerangka kerja adalah kumpulan *packages* dan *libraries* yang membantu pengembang dalam mengembangkan perangkat lunak (Qvarnström dan Jonsson, 2022). Terdapat banyak *framework* yang digunakan untuk mengembangkan REST API. Beberapa *framework* yang ada diantaranya adalah Express, ASP.NET, Spring, Symfony, dan Flask. Masing-masing *framework* memiliki kelebihan dan kekurangannya tersendiri.

Untuk mengembangkan REST API yang memiliki performa yang cepat, pengembang perlu memikirkan kebutuhan mereka secara teliti. Salah satunya adalah dengan memilih *framework* yang memiliki performa yang baik. Berdasarkan penelitian-penelitian yang telah ada, faktor yang mempengaruhi performa dari *framework* diantaranya adalah *response time* (Karlsonn, 2021), *throughput* (Qvarnström dan Jonsson, 2022), *CPU usage* (Hadinata dan Stianingsih, 2024), dan *memory usage* (Greiff dan Johansson, 2019). Pemilihan *framework* dengan performa yang baik untuk implementasi REST API merupakan hal yang penting untuk dilakukan (Karlsonn, 2021; Hadinata dan Stianingsih, 2024; Siahaan dan Wijaya, 2024). Memilih *framework* yang baik merupakan hal yang penting untuk mencegah perlunya perubahan *framework* di masa depan (Qvarnström dan Jonsson, 2022). Terdapat beberapa penelitian terdahulu yang telah dilakukan untuk menganalisa performa antara *framework* yang digunakan untuk mengembangkan REST API.

Salah satu penelitian yang telah dilakukan adalah penelitian yang dilakukan oleh Glantz dan Hurtig (2022) yang melakukan analisa perbandingan performa antara Express dan Ktor. Glantz dan Hurtig (2022) dalam penelitiannya menyatakan

bahwa Express memiliki waktu respon yang lebih cepat dari Ktor, terutama ketika digunakan oleh 150 dan 200 pengguna secara bersamaan. Penelitian lainnya juga dilakukan oleh Qvarnström dan Jonsson (2022) untuk menganalisa performa dari *framework* ASP.NET core, Express, dan Flask. Penelitian tersebut menyatakan bahwa Express memiliki waktu respon yang lebih cepat dari Flask. Penelitian-penelitian dilakukan untuk memberikan pemahaman kepada pengembang dalam memilih *framework* untuk mengembangkan *website*. Namun, penelitian-penelitian terdahulu memiliki kelemahan, yaitu *framework* yang digunakan adalah *framework* populer pada masanya. Selain itu, *framework* yang digunakan pada penelitian terdahulu juga menggunakan bahasa pemrograman yang berbeda, hal ini mempengaruhi performa dari *framework* (Qvarnström dan Jonsson, 2022).

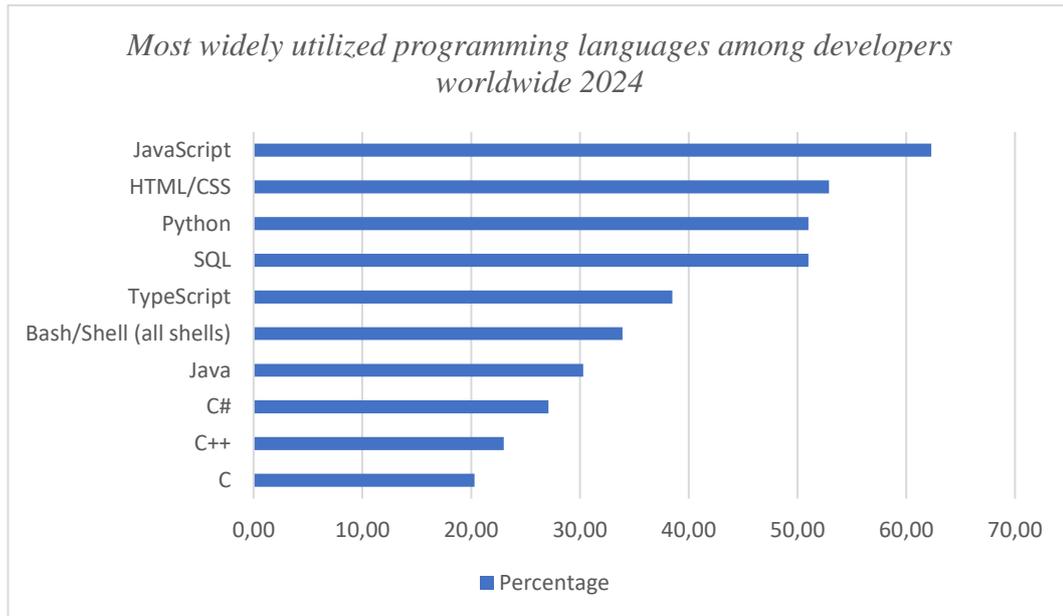
Berdasarkan pemaparan diatas, terdapat *gap* antara penelitian terdahulu dengan teknologi yang digunakan saat ini. Hal ini membuka peluang untuk dilakukannya penelitian yang melakukan penelitian untuk menganalisa *performance* antara *framework* yang menggunakan bahasa pemrograman yang sama, seperti pada Next.js dan Express. Pada saat ini, Next.js dan Express merupakan *server-side framework* yang populer dan keduanya menggunakan bahasa pemrograman yang serupa, yaitu JavaScript. Adapun hirarki dari *framework* yang menggunakan bahasa pemrograman JavaScript dan adalah sebagai berikut:



Gambar 1.1 Hirarki *framework* yang menggunakan bahasa JavaScript

Stack Overflow (2023) mengungkapkan bahwa Next.js dan Express.js berada pada peringkat 6 dan 4 pada survey *web framework* yang paling populer yang mereka lakukan pada tahun 2023. Oleh karena itu, untuk menjembatani *gap* tersebut, dilakukanlah analisa performance pada Next.js dan Express pada sisi implementasi REST API pada kedua *framework* tersebut. Alasan mengapa dipilihnya *framework* Next.js dan Express walaupun mereka tidak berada pada peringkat 1 dan 2 pada survey Stack Overflow (2023) adalah karena teknologi pada peringkat 1, 2, 3, dan 5 bukan merupakan *server-side framework*, melainkan *client-side framework*. Alasan lain mengapa Next.js dan Express dipilih selain karena popularitas adalah oleh karena Next.js dan Express merupakan *server-side framework* yang paling banyak digunakan di kalangan *developer* pada tahun 2024 (Vailshery, 2024b). Selain itu, minimnya penelitian performa REST API pada *framework* Next.js juga menjadi alasan mengapa Next.js dipilih pada penelitian ini.

Berdasarkan pemaparan-pemaparan diatas, diputuskan untuk melakukan penelitian untuk menganalisa performa implementasi REST API pada Next.js API Routes dan Express. Next.js dan Express oleh karena keduanya merupakan *framework sever-side* yang populer (Stack Overflow, 2023) dan yang paling banyak digunakan oleh *developer* pada tahun 2024 (Vailshery, 2024b). *Framework server-side* dipilih karena *framework* yang termasuk dalam kategori *server-side* inilah yang dapat digunakan untuk membangun REST API. Selain itu, keduanya juga menggunakan dasar bahasa pemrograman yang sama, yaitu JavaScript. Alasan dipilihnya bahasa pemrograman JavaScript pada penelitian ini dikarenakan JavaScript merupakan bahasa pemrograman paling populer (Stack Overflow, 2023) dan bahasa pemrograman yang paling banyak digunakan di kalangan *developer* pada tahun 2024 (Vailshery, 2024a).



Gambar 1.2 Bahasa pemrograman yang paling banyak digunakan *developer* tahun 2024 (Vailshery, 2024a)

Pada penelitian ini dilakukan pengujian performa REST API pada *framework* Next.js dan Express. Pengujian yang akan dilakukan adalah *load*, *stress*, dan *spike testing*. Parameter yang digunakan dalam penelitian ini adalah *response time*, *throughput*, *CPU usage*, dan *memory usage*. Parameter ini dipilih oleh karena hal-hal ini dapat menjadi ukuran untuk performa dari REST API yang diimplementasikan pada Next.js dan Express. Parameter-parameter ini juga telah digunakan dalam penelitian terdahulu, seperti penelitian yang dilakukan oleh Qvarnström dan Jonsson (2022), Glantz dan Hurtig (2022), dan penelitian lainnya. Pengujian akan dilakukan menggunakan aplikasi Apache JMeter karena JMeter dapat digunakan untuk melakukan *performance testing* serta telah digunakan oleh penelitian terdahulu. Penelitian ini diharapkan dapat memberikan pemahaman dan gambaran mengenai performa REST API pada Next.js dan Express. Pemahaman yang didapatkan dari penelitian ini juga diharapkan dapat dijadikan pertimbangan oleh pengembang yang akan mengembangkan aplikasi berbasis *website*.

## 1.2 Rumusan Masalah

Berdasarkan penjelasan pada latar belakang, maka dirumuskan masalah sebagai berikut:

- 1) Bagaimana performa REST API pada Next.js API Routes dan Express dalam pengujian *load*, *stress*, dan *spike testing*?
- 2) Bagaimana analisa performa REST API pada Next.js API Routes dan Express berdasarkan *response time*, *throughput*, *CPU usage*, dan *memory usage*?

### 1.3 Tujuan Penelitian

Berdasarkan rumusan masalah dalam penelitian ini, maka ditentukan tujuan dari dilakukannya penelitian ini sebagai berikut:

- 1) Melakukan pengujian performa REST API pada Next.js API Routes dan Express menggunakan *load*, *stress*, dan *spike testing*.
- 2) Menganalisis performa REST API pada Next.js API Routes dan Express berdasarkan *response time*, *throughput*, *CPU usage*, dan *memory usage*.

### 1.4 Manfaat Penelitian

Manfaat yang diharapkan penulis dapat dirasakan dari penelitian ini adalah sebagai berikut:

- 1) Dapat menjadi acuan untuk penelitian selanjutnya.
- 2) Memberikan pemahaman mengenai performa REST API pada Next.js API Routes dan Express.
- 3) Memberikan kontribusi kepada pengembang agar dapat memilih *framework* sesuai kebutuhan dengan performa yang baik.
- 4) Memberikan pemahaman yang mendalam bagi penulis tentang performa REST API pada Next.js API Routes dan Express.

### 1.5 Batasan Masalah

Berikut ini merupakan batasan masalah pada penelitian ini:

1. *Framework* yang diteliti terbatas pada *framework server-side*.
2. Kriteria yang digunakan untuk menguji performa REST API berfokus pada *response time*, *throughput*, *CPU usage*, dan *memory usage*.
3. Pengujian dilakukan secara lokal menggunakan perangkat penulis.

4. *Framework* yang dianalisis terbatas pada *framework* yang menggunakan bahasa pemrograman JavaScript.

## 1.6 Sistematika Penelitian

Sistematika penulisan dalam penelitian ini mengikuti sistematika penulisan yang diatur dalam Pedoman Penulisan Karya Ilmiah UPI Tahun 2019. Sistematika penulisan terbagi dalam lima bab dan terdapat subbab dalam setiap bab. Berikut sistematika penulisan yang digunakan dalam penelitian ini:

- a) Bab I: Pendahuluan

Bab I merupakan bab perkenalan dalam penelitian. Bab I berisikan latar belakang penelitian, rumusan masalah penelitian, tujuan penelitian, manfaat penelitian, serta sistematika penulisan yang digunakan dalam penelitian.

- b) Bab II: Kajian Pustaka

Bab II adalah bab yang memberikan konteks yang jelas terhadap permasalahan atau topik yang diteliti. Bab II berisikan konsep-konsep, teori-teori, dalil-dalil, model-model, dan hukum-hukum yang berkaitan dengan penelitian.

- c) Bab III: Metode Penelitian

Bab III berisikan hal terkait bagaimana peneliti merancang alur penelitiannya dari mulai pendekatan penelitian yang diterapkan, instrumen yang digunakan, tahapan pengumpulan data yang dilakukan, hingga langkah-langkah analisis data yang dijalankan.

- d) Bab IV: Temuan dan Pembahasan

Bab IV berisikan dua hal penting, yaitu temuan penelitian yang didasarkan pada hasil pengolahan serta analisis data dan pembahasan temuan penelitian yang didapatkan untuk menjawab pertanyaan penelitian.

- e) Bab V: Simpulan, Implikasi, dan Rekomendasi

Bab V berisikan simpulan, implikasi, dan rekomendasi yang menyajikan pemaknaan peneliti terhadap hasil analisis temuan penelitian sekaligus mengajukan hal-hal penting yang dapat dimanfaatkan dari hasil penelitian tersebut.