

## BAB 1 PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi pengembangan *website* semakin pesat. Salah satu aspek yang sangat penting dalam pengembangan aplikasi *web* adalah pengembangan antarmuka pengguna (*user interface* atau UI) atau yang sering disebut *front-end*. *Front-end* merupakan tampilan *website* yang bisa dilihat oleh siapapun di internet. Biasanya, *front-end* berisi konten-konten dari *website* tersebut, seperti bagian awal sebuah *website* atau yang biasa disebut Beranda, lalu ada Tentang Kami, Hubungi Kami, dan sebagainya. Bagian *front-end* biasanya dibangun menggunakan *HTML*, *CSS*, dan *JavaScript*. Sebenarnya, dengan menggunakan *HTML* dan *CSS* saja, tampilan antar muka sudah dapat dibangun. Tetapi, yang membuat *website* menjadi menarik adalah *JavaScript* karena *JavaScript* membuat *website* menjadi interaktif. Fungsi utama dari *JavaScript* adalah untuk membuat interaksi pada sebuah *website*, karena *JavaScript* dapat membuat kotak dialog, animasi, dan masih banyak lagi yang lainnya yang bisa membuat sebuah *website* menjadi interaktif. *Javascript* merupakan bahasa yang penting untuk pemrograman *website*. Hubungan bahasa *Javascript* dengan pemrograman *website* membuat bahasa ini menjadi salah satu bahasa pemrograman yang paling populer (Crockford, 2008).

Dalam pengembangan sebuah aplikasi, banyak *developer* yang menggunakan *framework*. *Framework* merupakan alat yang sangat penting untuk membantu *developer* dalam membuat aplikasi dengan cepat dan efisien. Dalam pengembangan *website*, banyak *developer* yang menggunakan *framework JavaScript* agar mempercepat pembangunan sebuah *website*. Di dalam ekosistem *JavaScript*, terdapat banyak pilihan *framework* sehingga menyulitkan setiap orang dalam menentukan manakah *framework* yang paling cocok untuk dipelajari (Riza Fahmi, 2020). Di antara banyak *framework JavaScript* yang tersedia, *ReactJS* dan *Svelte* merupakan dua *framework* yang cukup populer saat ini menurut survei yang dilakukan *Stackoverflow*. *Stackoverflow* merupakan sebuah *platform* yang menyediakan forum tanya jawab antar *developer* untuk menyelesaikan sebuah masalah yang ditemukan dalam sebuah kode. *Stackoverflow* merupakan salah satu

situs web yang paling populer di dunia. Menurut survei yang diadakan *Stackoverflow* di tahun 2022 dan diikuti oleh 57.654 responden, *ReactJS* merupakan *web framework* yang paling banyak dicari oleh banyak orang. Sedangkan *Svelte* terdapat di peringkat ke 5 dalam *web framework* yang banyak dicari. Lalu untuk survei *web framework* yang paling banyak dicintai *developer*, *Svelte* lebih unggul jika dibandingkan dengan *ReactJS* sebanyak 7.09% (2022).

*ReactJS* merupakan *framework* yang diluncurkan pada tahun 2013 dan dikembangkan oleh *Facebook*. *Framework* ini sangat populer digunakan untuk pengembangan aplikasi web berskala besar sampai tahun 2023 ini. *ReactJS* menawarkan fitur seperti *reusable component*, *virtual DOM*, *reactive values*, dan *JSX* yang membuatnya sangat cocok digunakan untuk pengembangan aplikasi *web* yang memiliki kompleksitas tinggi.

Tetapi, *ReactJS* juga dikenal memiliki kinerja yang tidak begitu baik pada aplikasi *web* yang memiliki ukuran kecil. *ReactJS* menggunakan metode *lifecycle* yang lama-kelamaan akan menambah baris kode dan akhirnya sulit untuk di-*maintain*. Hal ini juga akan membuat pekerjaan *developer* jauh lebih merepotkan dan meningkatkan jumlah waktu yang diinvestasikan untuk membangun sebuah *web* (Vishal, 2020). Sedangkan *Svelte* merupakan *framework* yang muncul tiga tahun setelah *ReactJS* diluncurkan. *Svelte* diklaim menawarkan kinerja yang lebih baik jika dibandingkan dengan *ReactJS*. *Svelte* menggunakan konsep yang berbeda dari *ReactJS*, yaitu melakukan kompilasi kodenya menjadi *JavaScript Vanilla* (Riza Fahmi, 2020). *Svelte* juga memiliki fitur yang tidak jauh berbeda dengan *ReactJS* seperti *reusable component* dan *reactive values*.

Jika dibandingkan secara performa, *ReactJS* memiliki performa yang lebih lambat dibandingkan *Svelte* karena *ReactJS* menginterpretasikan kodenya selama *runtime*. *Svelte* menginterpretasikan kode aplikasinya selama waktu *build* (Chidume Nnamdi, 2022). Secara *bundle size*, *Svelte* memiliki *bundle size* yang lebih kecil ketika dikompres menggunakan *gzip* dibandingkan dengan *ReactJS*. *Bundle Size* yang dimiliki *Svelte* berukuran 1.6 *kilobytes*, sedangkan *bundle size* yang dimiliki oleh *ReactJS* adalah 42.2 *kilobytes* (Gawkoski, 2022).

Terkait hal di atas, penulis ingin melakukan studi perbandingan performa kedua *framework* dalam pengembangan aplikasi *front-end* dengan membuat sebuah *website* yang sama dengan menggunakan dua *framework JavaScript* yang berbeda, yaitu *ReactJS* dan *Svelte*. Penulis ingin mengetahui lebih dalam performa yang lebih baik di antara kedua *framework JavaScript* tersebut dengan mengacu pada parameter yang dibandingkan yaitu performa dari kedua *website*.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah diuraikan, penulis mendapatkan beberapa rumusan masalah dalam penelitian:

1. Bagaimana mendapatkan data performa dari masing-masing *website* yang dibangun menggunakan *ReactJS* dan *Svelte*.
2. Bagaimana performa dari *website* yang dibangun menggunakan *ReactJS* dan *Svelte*.
3. Bagaimana laporan hasil yang akan didapatkan dari perbandingan uji performa yang dibangun menggunakan *ReactJS* dan *Svelte*.

## 1.3 Tujuan Tugas Akhir

Berdasarkan rumuan masalah yang sudah diuraikan, tujuan dari penelitian ini:

1. Mendapatkan data performa *website* yang dibangun menggunakan *ReactJS* dan *Svelte*.
2. Mengetahui performa *website* yang dibangun menggunakan *ReactJS* dan *Svelte*.
3. Mendapatkan data perbandingan performa *website* yang dibangun menggunakan *ReactJS* dan *Svelte*.

## 1.4 Manfaat Tugas Akhir

Perbandingan yang dilakukan antara *framework ReactJS* dan *Svelte* ini diharapkan dapat memberikan manfaat untuk para *developer* yang baru belajar dalam membangun sebuah *website*. Selain itu, perbandingan ini juga diharapkan agar para *developer* dapat memutuskan penggunaan *framework JavaScript* mana yang paling cocok dan memiliki performa yang baik berdasarkan skenario yang ada.

### 1.5 Batasan Masalah

Berdasarkan latar belakang yang dikemukakan di atas, batasan masalah dari penelitian ini adalah sebagai berikut:

1. *Website* yang akan dibuat adalah *website landing page*.
2. *Framework JavaScript* yang digunakan hanya bekerja pada *client-side*.
3. *Website* yang dibandingkan adalah *website* yang dibangun menggunakan *ReactJS* dan *Svelte*.
4. Parameter yang dibandingkan adalah parameter dari performa kedua *website* yang dibangun menggunakan *ReactJS* dan *Svelte*. (mencakup *Bundle Size*, *Page Loading time*, dan *Response Time*).