

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Teknologi informasi yang telah berkembang sangat pesat telah menjadikan internet tersedia secara luas untuk digunakan. Hal ini pun berdampak pada mudahnya masyarakat untuk mengakses sebuah website. Dalam pembangunan sebuah website pun, antarmuka pengguna menjadi suatu hal yang sangat penting dan perlu diperhatikan oleh sang pengembang website tersebut, hal ini karena antarmuka pengguna adalah hal yang berhubungan secara langsung dengan pengguna dari situs tersebut. Antarmuka yang menarik namun mudah digunakan menjadi salah satu hal yang dapat menentukan apakah situs tersebut akan mendapatkan banyak pengunjung atau tidak.

Salah satu cara untuk membangun antarmuka yang menarik adalah dapat dengan menggunakan bahasa JavaScript, sebuah bahasa pemrograman *script* yang didukung oleh browser-browser yang pada umumnya digunakan seperti Firefox, Safari, Opera, maupun Google Chrome. Belakangan ini banyak bermunculan JavaScript *library* yang dapat digunakan untuk membantu pengembang web dalam mengembangkan antarmuka yang akan digunakan. Salah satu JavaScript *library* yang populer dan banyak digunakan saat ini adalah ReactJS yang merupakan JavaScript *library* yang berfokus pada pembangunan antarmuka yang cepat, kompleks, namun tetap handal untuk digunakan (M & Sonpatki, 2016).

Karena ReactJS merupakan JavaScript *library* maka dirinya bersifat *client-side* yang artinya berjalan pada komputer yang digunakan untuk mengakses situs yang dikembangkan dengan ReactJS. Hal ini menjadikan seluruh antarmuka yang dibangun menggunakan ReactJS akan di-*render* pada komputer *client*, yang mana dalam hal kecepatan adalah sangat cepat begitu ReactJS telah selesai diunduh dari *server*. Jika dibandingkan dengan situs-situs yang masih mengandalkan *rendering* antarmuka secara *server-side*, dengan tidak perlunya sebuah antarmuka untuk di-*render* secara *server-side* maka hal ini tentu saja akan mengurangi beban kerja dari web *server* yang digunakan untuk menjalankan sebuah web, menjadikan *server*

dapat melayani request *client* yang lebih banyak dikarenakan *server* dalam hal ini hanya menyediakan data yang akan diolah pada *client-side*. Kelebihan lain dari memakai *client-side rendering* sangat terlihat ketika pengguna melakukan *refresh* pada halaman web. Dengan menggunakan *client-side rendering*, maka hanya perlu data baru saja yang diambil dari *server* dan kemudian ditampilkan kepada pengguna, tanpa perlu *me-load* keseluruhan halaman web lagi dari awal yang tentu saja sangat mengurangi beban *server* serta waktu untuk menampilkan halaman web (Kumar & Singh, 2016). Bahkan dengan memberikan animasi bahwa halaman web tersebut sedang di-*refresh* maka pengguna dapat mengetahui apa yang sedang terjadi di halaman web tersebut.

Berdasarkan keunggulan yang telah dipaparkan di atas, peneliti membangun sistem layanan kolaborasi identifikasi taksonomi dan sentimen berita berbahasa Indonesia dengan memanfaatkan ReactJS sebagai alat untuk *rendering* antarmuka pengguna yang akan digunakan oleh pengguna untuk memberikan kontribusinya pada situs tersebut. Sistem layanan tersebut termasuk situs *crowdsourcing* yang artinya mengandalkan pengguna dalam penyediaan atau pengumpulan ataupun pengolahan data yang didapatkan. Sebuah situs *crowdsourcing* cocok untuk menggunakan ReactJS ini sebagai alat untuk *rendering* antarmukanya karena menghasilkan waktu *loading* halaman situs yang lebih cepat, serta *rendering* pun tidak memberatkan kerja *server* karena berjalan secara *client-side* sehingga jumlah situs yang dapat dibuka secara bersamaan oleh pengguna pun menjadi lebih banyak, yang pada akhirnya dapat membantu situs *crowdsourcing* ini, karena pengguna situs dapat berkonsentrasi lebih untuk memberikan kontribusinya tanpa perlu menunggu waktu *loading* situs yang lama atau situs perlu dimuat berulang-ulang saat selesai mengerjakan suatu hal.

## **1.2. Rumusan Masalah**

Berdasarkan latar belakang di atas, maka dapat dirumuskan masalah sebagai berikut :

1. Bagaimana memanfaatkan ReactJS untuk *rendering* antarmuka pengguna dalam layanan kolaborasi identifikasi taksonomi, struktur kalimat, dan sentimen berita berbahasa Indonesia?

### 1.3. Batasan Masalah

Agar penelitian tepat sasaran, batasan-batasan permasalahan terhadap topik penelitian ditentukan sebagai berikut:

1. Basis data berita memanfaatkan basis data hasil penelitian sebelumnya.
2. *Embed* dibatasi hanya pada web berbasis Wordpress, Blogger, dan situs web pada umumnya.
3. *Framework* yang dibahas hanya React.JS.
4. Struktur kalimat berbentuk SPOK.
5. Kosakata taksonomi berita hanya 5 kategori, yaitu: kategori orang, organisasi, geografis, perusahaan, dan topik saja.
6. Sentimen kalimat berita hanya 3 kategori, yaitu: negatif, netral, dan positif.

### 1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Memanfaatkan React.JS untuk *rendering* antarmuka pengguna dalam layanan kolaborasi identifikasi taksonomi, struktur kalimat, dan sentimen berita berbahasa Indonesia.

### 1.5. Metodologi Penelitian

Tahap metodologi pada pembangunan aplikasi dibagi menjadi beberapa tahap:

1. Alat

Aplikasi berbasis web ini dibangun, diuji, dan dioperasikan pada beberapa alat:

#### 1.1. CodeIgniter 3.1.3

Sebagai *framework* untuk membangun aplikasi web layanan kolaborasi identifikasi taksonomi, struktur kalimat, dan sentimen berita berbahasa Indonesia.

#### 1.2. Atom 1.20.1

Sebagai perangkat lunak untuk memprogram *framework* yang digunakan.

### 1.3. Apache 2.4.25

Sebagai perangkat lunak *web server* untuk menjalankan *framework* yang digunakan. Apache versi ini menjalankan PHP versi 7.1.4.

### 1.4. MySQL 10.1.22-MariaDB

Sebagai perangkat lunak *server* basis data untuk menyimpan data-data dari aplikasi web.

## 2. Langkah-langkah Penelitian

### 2.1. Studi Literatur

Studi literatur digunakan untuk mengetahui penelitian-penelitian yang pernah dilakukan oleh pihak lain sehubungan dengan topik peneliti. Studi literatur ini meliputi jurnal, laporan, artikel di internet, maupun buku.

### 2.2. Analisis Perangkat Lunak

Pada bagian ini dilakukan analisis kebutuhan serta analisis sistem daripada aplikasi yang akan dibangun menggunakan data-data yang telah diperoleh sebelumnya. Hasil dari bagian ini meliputi ERD, PDM, arsitektur sistem, serta *usecase*.

### 2.3. Perancangan Perangkat Lunak

Pada tahap perancangan perangkat lunak, dihasilkan prototipe daripada aplikasi yang akan dibangun, berupa *mockup* antarmuka, dan basis data dari aplikasi.

### 2.4. Pengkodean Perangkat Lunak

Pada tahap pengkodean perangkat lunak dilakukan proses pembangunan aplikasi dengan merealisasikan keseluruhan analisis dan perancangan perangkat lunak dengan pembangunan *backend* dan *frontend* dari aplikasi, serta pembangunan antarmuka aplikasi.

### 2.5. Pengujian Perangkat Lunak

Pada tahap pengujian perangkat lunak, dilakukan pengujian terhadap perangkat lunak yang sudah dibangun, untuk menguji setiap fungsionalitas yang ada.

## **1.6. Sistematika Penulisan Tugas Akhir**

Sistematika penulisan tugas akhir dapat dijabarkan sebagai berikut:

### **1. BAB I: PENDAHULUAN**

Bab I membahas landasan, masalah, dan tujuan pembangunan aplikasi, batasan-batasan, metodologi yang digunakan, serta sistematika penulisan tugas akhir.

### **2. BAB II: TINJAUAN PUSTAKA**

Bab II berisi tentang uraian singkat hasil penelitian terdahulu yang ada hubungannya dengan permasalahan yang ditinjau di dalam Tugas Akhir.

### **3. BAB III: LANDASAN TEORI**

Bab III berisi tentang uraian dasar teori yang digunakan dalam melakukan perancangan dan pembuatan aplikasi yang dapat dipergunakan sebagai pembanding atau acuan di dalam pembahasan masalah.

### **4. BAB IV: ANALISIS DAN PERANCANGAN SISTEM**

Bab IV berisi penjelasan mengenai analisis permasalahan yang akan diatasi serta membahas mengenai perancangan perangkat lunak yang dibuat.

### **5. BAB V: IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab V berisi penjelasan mengenai implementasi perangkat lunak yang dibuat dan gambaran umum sistem.

### **6. BAB VI: KESIMPULAN DAN SARAN**

Bab VI merupakan bab penutup, berisi kesimpulan dan saran yang didapatkan selama pembuatan tugas akhir.