

BAB I PENDAHULUAN

I.1 Latar Belakang

Transformasi digital telah membawa banyak perubahan signifikan dalam berbagai industri, termasuk pendidikan. Pandemi COVID-19 mempercepat perkembangan transformasi digital pada industri pendidikan, memaksa institusi pendidikan untuk mengadopsi pembelajaran jarak jauh dan materi pembelajaran dalam format digital. Transformasi ini tidak hanya mengubah metode pengajaran dan pembelajaran, tetapi juga menuntut adanya peningkatan infrastruktur teknologi dan kemampuan digital di kalangan tenaga pendidik dan mahasiswa. Transformasi digital ini memaksa institusi pendidikan mulai mengubah layanan-layanan agar serba digital.

Universitas Telkom merupakan salah satu perguruan tinggi yang telah menyesuaikan diri dengan transformasi digital tersebut dan salah satu upaya Universitas Telkom dalam menjawab tantangan ini adalah digitalisasi pada layanan sidang tugas akhir. Bukti nyata dari digitalisasi ini adalah *website* Sidang *Online* Fakultas Rekayasa Industri (SOFI), sebuah platform berbasis web yang digunakan untuk memonitor Sidang Tugas Akhir Fakultas Rekayasa Industri di Universitas Telkom. Beberapa proses bisnis pada aplikasi SOFI adalah pendaftaran, penjadwalan, pelaksanaan, revisi, dan penilaian.

Aplikasi SOFI saat ini menggunakan arsitektur monolitik, yang cocok untuk skala pengguna kecil dan mudah untuk dikembangkan. Namun, penelitian terdahulu yang berjudul "A Comparative Review of Microservices and Monolithic Architectures" (Al-Debagy & Martinek, 2018) menunjukkan bahwa arsitektur monolitik tidak dapat memenuhi kebutuhan aplikasi SOFI yang memerlukan fleksibilitas dan skalabilitas untuk menangani pengguna dalam jumlah besar. Arsitektur monolitik ini sering menyebabkan keluhan dari pengguna dan sulit dipahami oleh pengembang tanpa pengembang tetap. Oleh karena itu, diperlukan perubahan ke arsitektur *microservices*, yang lebih efisien dan lebih mudah dipelihara, seperti yang dibuktikan oleh penelitian "The Comparison of Microservice and Monolithic Architecture" (Gos & Zabierowski, 2020). Dengan

arsitektur *microservices*, basis kode dipecah menjadi unit-unit kecil dan spesifik, sehingga setiap fungsi utama terpisah dan kesalahan pada satu *microservices* tidak mempengaruhi keseluruhan sistem.

Saat ini, aplikasi SOFI juga masih menggunakan konsep MPA. Dalam konsep MPA, setiap kali pengguna melakukan navigasi antar halaman, seluruh konten *website* harus dimuat ulang. Hal ini menyebabkan *website* menjadi lebih berat dan memperpanjang waktu muat. Kelemahan lain dari MPA adalah peningkatan penggunaan *bandwidth* karena setiap navigasi memuat ulang halaman sepenuhnya, yang bisa menjadi masalah bagi pengguna dengan koneksi internet lambat. Selain itu, transisi antar halaman yang tidak mulus dapat mengganggu kenyamanan pengguna dan membuat interaksi terasa lambat dan terputus-putus. Pembuktian permasalahan *frontend* pada aplikasi SOFI modul Pendaftaran ini ada pada sesi wawancara dengan pengembang aplikasi SOFI, yang menyebutkan bahwa setiap kali halaman dimuat ulang, ada potensi kehilangan data yang telah diinput oleh pengguna sebelum *error* terjadi, seperti dalam proses pendaftaran. *Error handling* yang tidak efisien ini sering kali mengakibatkan pengguna kehilangan data yang telah dimasukkan, memperburuk pengalaman pengguna secara keseluruhan. Selain itu, meningkatkan kinerja aplikasi juga diperlukan ketika memuat data pengguna dalam jumlah yang besar.

Pengguna SOFI terutama pada modul Pendaftaran memiliki rentang usia yang sangat luas, mencakup mahasiswa muda hingga dosen senior, karena mahasiswa perlu mengajukan sidang dan dosen perlu untuk melihat data pengajuan sidang. Karena perbedaan dalam tingkat keterampilan teknologi dan preferensi pengguna, penting untuk memperhatikan kebutuhan dan preferensi dari kedua kelompok ini. Intuitifitas, responsivitas, penanganan *error*, dan peningkatan kinerja dalam aplikasi SOFI menjadi semakin penting untuk memastikan bahwa semua pengguna dapat dengan mudah mengakses dan menggunakan platform tersebut sesuai dengan kebutuhan mereka.

Untuk menghadapi masalah tersebut, terdapat konsep SPA. Menurut Sangati (2022), Aplikasi SPA akan meningkatkan interaktivitas pengguna karena mereka akan merespon dalam waktu yang lebih singkat, meskipun waktu muat awal lebih

lama, untuk tindakan berikutnya waktu yang dibutuhkan lebih singkat dibandingkan dengan MPA. Dalam SPA, hanya bagian-bagian tertentu dari *website* yang diperbarui saat pengguna berinteraksi, sehingga tidak perlu memuat ulang seluruh halaman. Keuntungan SPA bagi pengguna adalah peningkatan performa dan penanganan *error* yang lebih efisien. Ketika terjadi *error*, data yang telah diinput tidak akan hilang karena halaman tidak perlu dimuat ulang sepenuhnya, yang mengurangi frustrasi dan ketidakpuasan pengguna. Masalah yang terjadi pada aplikasi SOFI dapat diatasi dengan menggunakan konsep SPA sebagai solusi untuk mengatasi permasalahan yang ada.

Penelitian ini berfokus pada migrasi MPA ke SPA pada aplikasi SOFI modul Pendaftaran, dengan metode yang digunakan adalah *Iterative Incremental Development*. Metode *Iterative Incremental Development* dipilih karena pendekatan yang iteratif dan fleksibel (Adel & Abdullah, 2015). Metode ini sangat sesuai untuk penelitian ini karena keunggulannya dalam memungkinkan penyesuaian berkelanjutan dan peningkatan terus-menerus selama siklus pengembangan. Pendekatan ini sangat diperlukan untuk menangani kompleksitas migrasi konsep dan memastikan performa serta pengalaman pengguna yang optimal pada website SOFI.

I.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan diatas, maka rumusan masalah dalam penelitian ini sebagai berikut:

- a. Bagaimana cara meningkatkan keintuitifan aplikasi SOFI modul Pendaftaran dalam berinteraksi dengan pengguna, khususnya dalam penanganan *error*, agar pengguna dapat lebih mudah memahami alur aplikasi?
- b. Bagaimana cara mengatasi masalah kinerja pada aplikasi SOFI modul Pendaftaran yang terjadi saat memuat data pengguna dalam jumlah besar, yang menyebabkan penurunan performa aplikasi serta tantangan dalam menjaga responsivitas dan efisiensi aplikasi?

I.3 Tujuan Penelitian

Penelitian ini bertujuan untuk:

- a. Mengembangkan *frontend* aplikasi SOFI modul Pendaftaran dengan mengimplementasikan konsep SPA untuk meningkatkan keintuitifan aplikasi SOFI dan meningkatkan kinerja dan efisien aplikasi, sehingga pengalaman pengguna menjadi lebih baik.
- b. Mengimplementasikan SPA pada aplikasi SOFI modul Pendaftaran agar meningkatkan performansi aplikasi SOFI saat memuat data pengguna dalam jumlah besar dan menjaga responsivitas dan efisiensi aplikasi.

I.4 Batasan Penelitian

Batasan yang ditetapkan pada penelitian ini sebagai berikut:

- a. Penelitian berfokus pada modul Pendaftaran pada aplikasi SOFI.
- b. Penelitian ini tidak mengubah tampilan atau mengubah alur fitur pada aplikasi SOFI
- c. Aplikasi *frontend* dibangun menggunakan bahasa *javascript* dengan *library* React.js.
- d. Pengujian aplikasi menggunakan UAT (*User Acceptance Testing*).

I.5 Manfaat Penelitian

Manfaat penelitian ini:

1. Bagi Universitas Telkom, penelitian ini bermanfaat dalam meningkatkan keintuitifan aplikasi SOFI sehingga mahasiswa dan dosen dapat lebih mudah menggunakannya untuk keperluan akademik dan administratif, serta meningkatkan efisiensi dalam mengakses informasi dan layanan yang disediakan oleh universitas.
2. Bagi peneliti, penelitian ini bermanfaat untuk memberi ilmu dalam pengembangan *website* dengan konsep SPA.
3. Bagi pendidik dan pengajar, penelitian ini bermanfaat untuk meningkatkan keintuitifan aplikasi SOFI, sehingga aplikasi memiliki fitur-fitur yang lebih mudah digunakan dan efektif untuk menunjang kebutuhan sidang.

4. Bagi pembaca dan peneliti lain, penelitian ini bermanfaat sebagai referensi untuk pengembangan *website* dengan konsep SPA.