

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Perkembangan teknologi yang pesat telah membawa perubahan besar dalam dunia pendidikan, termasuk dalam pengelolaan data di perpustakaan. Perpustakaan memiliki peran penting sebagai sarana untuk memajukan pendidikan dan kualitas ilmu pengetahuan. Sebagai unit kerja, perpustakaan berfungsi sebagai tempat untuk mengumpulkan, menyimpan, dan menyediakan koleksi buku yang dikelola secara sistematis untuk memudahkan akses informasi. Dengan adanya sistem informasi perpustakaan dapat membantu pustakawan lebih mudah mengelola bahan pustaka dan memberikan pelayanan peminjaman yang efisien (Putri dkk., 2022). Salah satu aspek penting dalam hal ini adalah pengolahan dan integrasi data dan informasi yang ada di dalam sistem.

STMIK AMIKOM Surakarta atau Sekolah Tinggi Manajemen Informatika dan Komputer AMIKOM Surakarta adalah perguruan tinggi di Surakarta yang fokus pada bidang teknologi informasi dan komputer. Dikenal sebagai Kampus Multimedia, perguruan tinggi ini berlokasi di Singopuran, Kartasura, Sukoharjo, dengan fasilitas kampus yang terdiri dari gedung berlantai empat. STMIK AMIKOM Surakarta telah mengembangkan sistem perpustakaan menggunakan SLiMS versi 8 (Akasia), sebuah perangkat lunak *open-source* berbasis web yang mengelola sumber daya perpustakaan. Meskipun sistem ini sudah berfungsi dengan baik dan memenuhi kebutuhan perpustakaan saat ini, namun berdasarkan hasil

wawancara dengan pustakawan ditemukan kendala dalam proses peminjaman buku. Mahasiswa harus berinteraksi langsung dengan pustakawan untuk melakukan peminjaman, karena sistem belum mendukung fitur *booking* atau peminjaman mandiri yang bisa diakses langsung oleh mahasiswa. Selain itu, keterbatasan integrasi data yang hanya bisa diakses di dalam sistem itu sendiri mengakibatkan data bersifat terisolasi dan sulit dijangkau oleh *platform* lain, seperti aplikasi *mobile*. Kondisi ini membatasi fleksibilitas dan efisiensi layanan sistem perpustakaan. Oleh karena itu, sebagai pendukung perkembangan teknologi saat ini tidak menutup kemungkinan perlunya pengembangan lintas *platform* seperti aplikasi berbasis *mobile* sebagai inovasi layanan sistem Perpustakaan AMIKOM Surakarta saat ini. Dengan tujuan memperluas aksesibilitas, dan memberikan pengalaman pengguna yang lebih baik.

Tantangan dalam pengembangan sistem perpustakaan yang telah ada sebelumnya, yaitu bagaimana data dan informasi lama dapat diintegrasikan pada sistem lintas *platform*, bahasa pemrograman, maupun berbagai sistem operasi untuk memastikan keberlangsungan pengelolaan data di masa mendatang. Selain itu, penting untuk melakukan pengujian performa pada sistem yang dikembangkan untuk memastikan bahwa *API* dan aplikasi dapat berfungsi secara optimal, terutama dalam menghadapi berbagai kondisi beban yang tinggi. Oleh karena itu, perlu pengembangan lebih lanjut untuk memperluas aksesibilitas dan integrasi sistem di tengah perkembangan teknologi yang terus berkembang.

Sebagai solusi untuk tantangan yang dihadapi, pengembangan *REST API* (*Representational State Transfer Application Programming Interface*) menjadi

salah satu solusi efektif dalam mengatasi masalah integrasi dan pengelolaan data yang ada pada sistem Perpustakaan AMIKOM Surakarta. *REST* adalah arsitektur komunikasi sumber daya (*resources*) menggunakan protokol *HTTP* (*Hypertext Transfer Protocol*) untuk pertukaran data antar aplikasi. *API* merupakan sebuah *interface* yang mampu mengintegrasikan data serta menghubungkan aplikasi di berbagai *platform* sehingga dapat saling terhubung satu sama lain (Mulana dkk., 2022). Jadi *REST API* memungkinkan untuk menghubungkan sistem atau aplikasi yang berbeda *platform* secara *real-time* melalui pertukaran data dalam format seperti *JSON* (*JavaScript Object Notation*) atau *XML* (*Extensible Markup Language*) dengan memanfaatkan protokol *HTTP*. *REST API* menjadi jembatan penghubung berbagai sistem atau aplikasi untuk saling berkomunikasi secara terstruktur dan efisien, tanpa terikat pada teknologi atau bahasa pemrograman tertentu. Dengan menggunakan *REST API*, data yang tersimpan dalam sistem Perpustakaan AMIKOM Surakarta lama dapat diakses dan digunakan pada *platform* baru, baik itu berbasis web, *mobile*, ataupun sistem eksternal lainnya.

Pengujian performa *REST API* dapat dilakukan dengan memanfaatkan perangkat lunak Apache JMeter. Perangkat lunak ini merupakan alat pengujian kinerja berbasis *desktop* yang dikembangkan Apache yang digunakan untuk menguji beban halaman web, aplikasi web, dan sumber statis atau dinamis lainnya, termasuk *database*, file, server, dan lain-lain (Mangapul Siahaan & Wijaya, 2024). Apache JMeter bekerja dengan cara mensimulasikan sejumlah pengguna yang mengirimkan permintaan ke server target. Setelah itu, JMeter akan mengumpulkan data respons untuk kemudian menghitung metrik kinerja dan menampilkannya

dalam berbagai format. Hasil metrik kinerja seperti waktu respons (*response time*), tingkat kesalahan (*error rate*) pada setiap *endpoint*, dan *throughput* untuk mengetahui banyaknya pengguna yang dapat diproses sistem per detik. Pengujian performa yang dapat dilakukan meliputi pengujian beban (*load test*), pengujian stres (*stress test*), dan pengujian daya tahan (*soak test*).

Pengujian performa pada *REST API* penting dilakukan untuk memastikan *REST API* dapat berfungsi dengan baik di bawah berbagai kondisi dan beban yang mungkin terjadi. Hal ini dapat menjadi acuan bagi pengembang atau *developer* untuk mengidentifikasi *bottleneck* atau kelemahan dalam arsitektur *API* yang telah dikembangkan (Permatasari dkk., 2020). Selain itu, data dari *Think with Google* menyatakan pentingnya kecepatan atau performa situs *mobile*, dengan menunjukkan bahwa 53% kunjungan akan ditinggalkan jika sebuah situs *mobile* memerlukan waktu lebih dari 3 detik untuk dimuat (Lampiran 1). Hal ini semakin menegaskan pentingnya *response time*, yang dapat dipengaruhi oleh performa *REST API* yang dikembangkan. Oleh karena itu, pengujian performa tidak hanya bertujuan untuk memastikan bahwa *API* berfungsi sesuai harapan, tetapi juga untuk menjamin keandalan dan kesuksesan aplikasi dalam jangka panjang.

Dalam aplikasi *mobile* dengan integrasi *REST API*, keandalan kinerja menjadi faktor penentu pengalaman pengguna. Ketika jumlah peminjam atau pencari buku meningkat pada jam-jam sibuk, setiap permintaan (*request*) yang lambat akan menurunkan kepuasan dan pada kasus tertentu dapat menimbulkan kegagalan transaksi. Oleh karena itu, diperlukan pengujian performa yang

sistematis untuk memastikan *REST API* mampu merespons secara konsisten pada beban harian normal maupun saat lonjakan trafik.

Apache JMeter dipilih sebagai alat uji karena mampu mensimulasikan lalu lintas pengguna secara terukur melalui *thread group* (merepresentasikan jumlah pengguna bersamaan) dan *test plan* yang fleksibel untuk berbagai skenario. Dengan JMeter, metrik-metrik kunci seperti waktu respons rata-rata, *throughput* (permintaan per detik), dan error rate dapat diukur secara objektif. Pendekatan tersebut memungkinkan peneliti mengidentifikasi ambang aman kapasitas, memetakan potensi *bottleneck*, serta merumuskan penyesuaian konfigurasi server apabila dibutuhkan.

Pengujian performa menggunakan Apache JMeter dalam penelitian ini tidak hanya bertujuan untuk mengetahui capaian kinerja *REST API*, tetapi juga berfungsi sebagai bagian dari pengendalian mutu sistem. Hal ini penting agar layanan sistem mobile perpustakaan tetap responsif dan andal, meskipun dihadapkan pada kondisi operasional dengan keterbatasan sumber daya maupun pola trafik pengguna yang dapat berubah-ubah. Hasil pengujian performa menjadi dasar pertimbangan pada tahap implementasi dan pembahasan, khususnya dalam menetapkan target kinerja yang realistis, menilai kemampuan skalabilitas sistem, serta merumuskan strategi perbaikan berkelanjutan, baik pada sisi aplikasi maupun infrastruktur server yang digunakan.

Metode pengembangan *Rapid Application Development* (RAD) dipilih dalam pengembangan sistem Perpustakaan Amikom Surakarta karena mendukung pengembangan secara cepat dan iteratif. Selain itu, RAD memudahkan dalam

optimasi performa *REST API* melalui pengujian berulang, serta memungkinkan sistem diuji dan disempurnakan sebelum dirilis ke pengguna akhir. Penelitian ini bertujuan untuk menguji integrasi *REST API* yang dibangun menggunakan JavaScript dan *framework* Express.js pada aplikasi *booking book* berbasis *mobile* untuk sistem Perpustakaan AMIKOM Surakarta. Aplikasi *mobile* ini dikembangkan menggunakan React Native dan *framework* Expo, bertujuan untuk memperluas aksesibilitas data perpustakaan yang ada sehingga dapat terhubung dengan sistem lintas *platform*, serta memastikan konsistensi data yang diakses antara aplikasi *mobile* dan server perpustakaan. Pengujian performa dilakukan menggunakan Apache JMeter untuk mengukur kinerja *REST API* yang meliputi waktu respons dan error rate dalam menghadapi beban yang tinggi dan memastikan aplikasi berjalan dengan lancar, aman, dan efisien. Penelitian ini diharapkan dapat memberikan pemahaman terkait pengujian performa *REST API* dan bagaimana penerapannya untuk menghubungkan sistem antar *platform*.

1.2. Rumusan Masalah

Berdasarkan dari uraian latar belakang diatas, maka rumusan masalah dalam penelitian ini adalah:

- a. Bagaimana pengembangan *REST API* pada sistem Perpustakaan AMIKOM Surakarta dan integrasinya ke aplikasi *booking book* berbasis *mobile*?
- b. Bagaimana pengujian performa *REST API* menggunakan Apache JMeter berdasarkan parameter *response time*, *error rate*, dan *throughput*?

- c. Bagaimana hasil evaluasi tingkat kegunaan (usability) aplikasi *mobile booking book* berdasarkan metode *System Usability Scale* (SUS)?

1.3. Batasan Masalah

Batasan masalah yang akan diuraikan dalam penelitian ini memuat penjelasan sebagai berikut:

- a. Penelitian ini mengembangkan *REST API* sistem SLiMS Perpustakaan AMIKOM Surakarta ke aplikasi *booking book* berbasis *mobile*.
- b. Menganalisis pengujian performa *REST API* menggunakan Apache JMeter berdasarkan parameter *response time*, *error rate*, dan *throughput*.
- c. Pengujian kegunaan dari sisi pengguna terhadap aplikasi *mobile booking book* menggunakan metode *System Usability Scale* (SUS).
- d. Level pengguna dibatasi pada dua jenis, yaitu anggota perpustakaan sebagai pengguna aplikasi *mobile* dan pustakawan sebagai pengguna sistem SLiMS Perpustakaan AMIKOM Surakarta.
- e. Fitur utama yang menjadi fokus adalah *booking* buku pada aplikasi *mobile* yang diintegrasikan dengan sistem SLiMS pada fitur sirkulasi yang meliputi peminjaman, pengembalian, perpanjangan, dan validasi reservasi.

1.4. Tujuan Penelitian

Tujuan yang ingin dicapai dan diperoleh dalam penelitian ini, sebagai berikut:

- a. Membangun *REST API* untuk aplikasi *mobile booking book* sebagai pengembangan sistem perpustakaan yang sudah berjalan.
- b. Mengetahui hasil parameter pengujian performa *REST API* seperti *response time*, *error rate*, dan *throughput* dengan menggunakan Apache JMeter.
- c. Melakukan evaluasi tingkat kegunaan aplikasi *mobile booking book* menggunakan metode *System Usability Scale* (SUS).

1.5. Manfaat Penelitian

Penelitian ini diharapkan memberikan dampak positif dan manfaat sebagai berikut:

- a. Bagi *developer* Sistem Perpustakaan AMIKOM Surakarta, penelitian ini dapat memberikan gambaran pengembangan sistem dengan aksesibilitas dan integrasi data antar berbagai *platform* melalui penggunaan *REST API*.
- b. Bagi penulis, penelitian ini dapat memberikan wawasan mengenai pengembangan, pengujian, dan integrasi *REST API* dengan *platform mobile*, khususnya dalam konteks sistem perpustakaan.
- c. Bagi STMIK AMIKOM Surakarta, penelitian ini dapat mendorong kampus untuk beradaptasi dengan perkembangan teknologi terkini, terutama dalam hal integrasi data. Hal ini menjadi indikator bahwa kampus dapat menyediakan layanan yang lebih modern.