

BAB I

PENDAHULUAN

1.1. Latar Belakang

Universitas Negeri Jakarta adalah salah satu perguruan tinggi negeri di Indonesia yang menyelenggarakan program pendidikan mulai dari jenjang diploma, sarjana, magister hingga doktoral. Universitas Negeri Jakarta memiliki beberapa tujuan, salah satunya adalah memberikan pelayanan teknologi, manajemen dan sistem informasi bagi civitas akademika Universitas Negeri Jakarta dan masyarakat luas (Luddin et al, 2015:16). Untuk mencapai tujuan tersebut, dalam melaksanakan fungsinya sebagai lembaga pendidikan tinggi, Universitas Negeri Jakarta didukung dengan lembaga, biro dan unit pelaksana teknis (UPT). Masing-masing lembaga, biro dan unit pelaksana teknis tersebut melaksanakan berbagai layanan-layanan baik untuk mahasiswa, dosen, karyawan, maupun pihak luar universitas.

Dalam melaksanakan pelayanannya, masing-masing lembaga, biro dan unit pelaksana teknis di Universitas Negeri Jakarta didukung dengan sejumlah sistem informasi. Contohnya Sistem Informasi Akademik (SIKAD), Sistem Informasi Uang Kuliah Tunggal (SIUKAT), Sistem Persuratan Mahasiswa (SIPERMAWA) dan aplikasi sistem informasi lainnya. Kondisi saat ini masing-masing layanan dilaksanakan oleh masing-masing lembaga, unit atau biro itu sendiri dengan aplikasi sistem informasinya masing-masing. Seperti layanan peminjaman ruang dan kendaraan di Universitas Negeri Jakarta dilayani oleh Biro Umum dan Kepegawaian (BUK), pengajuan surat-surat akademik mahasiswa yang dilayani oleh Biro Akademik, Kemahasiswaan dan Hubungan Masyarakat (BAKH), dan layanan permintaan jasa TIK oleh Unit Pelaksana Teknis Teknologi Informasi dan Komunikasi (UPT TIK).

Pelayanan terpisah yang dilaksanakan oleh masing-masing lembaga, unit atau biro dengan masing-masing aplikasi sistem informasinya sendiri tersebut memiliki beberapa kekurangan. Kekurangan pertama akibat dilaksanakan oleh masing-masing unit adalah pengguna harus mengakses aplikasi masing-masing layanan yang berbeda-beda dan harus datang ke masing-masing lokasi pelayanan apabila diperlukan. Kedua, tidak efisiennya proses pelayanan karena pelayanannya yang

terpisah-pisah. Layanan yang melibatkan beberapa unit tidak terintegrasi dan tidak dilaksanakan dalam satu waktu. Ketiga, beberapa lembaga, unit atau biro tidak melakukan sosialisasi yang baik dalam menginformasikan layanannya.

Untuk mengatasi permasalahan tersebut dapat diatasi dengan mengembangkan suatu sistem pelayanan terpadu satu pintu yang dilaksanakan dalam satu lokasi dan satu sistem yang terintegrasi dengan memanfaatkan aplikasi sistem informasi untuk melakukan pelayanannya. Tercantum pada Peraturan Presiden Republik Indonesia nomor 97 tahun 2014 tentang penyelenggaraan pelayanan terpadu satu pintu, Pelayanan Terpadu Satu Pintu (PTSP) adalah pelayanan secara terintegrasi dalam satu kesatuan proses dimulai dari tahap permohonan sampai dengan tahap penyelesaian produk pelayanan melalui satu pintu. Unit Layanan Terpadu (ULT) merupakan contoh unit yang melaksanakan pelayanan terpadu satu pintu. Namun saat ini Universitas Negeri Jakarta belum memiliki unit layanan terpadu. Apabila Universitas Negeri Jakarta memiliki unit layanan terpadu, maka akan lebih baik jika pelayanannya dibantu dengan perangkat lunak sistem informasi yang akan menyatukan semua layanan-layanan yang ada pada masing-masing lembaga, biro, atau unit. Perangkat lunak tersebut dapat berupa aplikasi berbasis web atau *website* yang dapat diakses oleh pengguna aplikasi tanpa harus datang ke lokasi unit layanan terpadu.

Pengembangan aplikasi berbasis web saat ini umumnya dibagi menjadi dua bagian, yaitu bagian *front-end* dan bagian *back-end*. Pengembangan *front-end* adalah praktik untuk menghasilkan HTML, CSS dan Javascript untuk sebuah *website* atau aplikasi berbasis web sehingga pengguna dapat melihat dan berinteraksi secara langsung (Lindley, 2019:8). Contohnya adalah halaman-halaman pada *website* yang terdapat fitur-fitur yang dapat digunakan oleh pengguna *website* tersebut. Pengembangan bagian *front-end* sebuah *website* menggunakan bahasa Javascript dengan dukungan banyak *framework* yang tersedia untuk mempermudah pengembangannya. Sedangkan pengembangan *back-end* adalah pengembangan bagian *website* yang tidak dapat dilihat oleh pengguna atau bagian akses data (Christensson, 2020). Bagian *back-end* merupakan rangkaian aturan untuk masuk keluarnya data yang akan disimpan di dalam *database*. Salah satu contoh aturan tersebut adalah bagian *backend* mengatur siapa saja *user* yang dapat

mengakses maupun merubah data yang tersimpan di dalam *database* untuk menjaga keamanan data. Bahasa pemrograman yang digunakan untuk pengembangan bagian *backend* ini bermacam-macam, contohnya bahasa pemrograman PHP, Java, Python dan Javascript.

Dalam pengembangan perangkat lunak terdapat arsitektur perangkat lunak yang dapat digunakan untuk membangun struktur sistem perangkat lunaknya yang terdiri dari komponen-komponen perangkat lunak yang saling berhubungan. Dalam pengembangan perangkat lunak berbasis web terdapat beberapa arsitektur yang dapat diimplementasikan, antara lain: arsitektur *monolithic*, arsitektur *service-oriented* (SOA), dan arsitektur *microservices*.

Menurut Richardson dan Smith (Richardson & Smith, 2016: 3), arsitektur *monolithic* adalah arsitektur yang mana sebuah aplikasi dipaketkan dan didistribusikan dalam sebagai kesatuan atau *monolith* meskipun memiliki logika yang modular. Arsitektur menggambarkan sebuah aplikasi yang menjalankan semua logika dalam satu server aplikasi. Dengan kata lain, arsitektur *monolithic* menggabungkan beberapa komponen perangkat lunak dalam satu macam *platform*. Biasanya, *website* yang dibangun menggunakan arsitektur *monolithic* terdapat sebuah *database*, *user interface* untuk sisi *client* dan sebuah aplikasi *server-side*. Semuanya tergabung dalam satu aplikasi dan fungsinya dikelola dalam satu tempat yang tidak dapat dipisahkan.

Menurut Ang dan Luo (Ang & Luo, 2004:24.), arsitektur *service-oriented* adalah arsitektur yang melakukan pendekatan untuk membangun sistem terdistribusi yang menyalurkan fungsionalitas aplikasi sebagai *service* kepada *end-user* maupun *service* lainnya. Arsitektur ini memiliki basis *service-oriented* di mana permasalahan diselesaikan dengan memberikan penekanan pada *service*. *Service* sendiri adalah merupakan kumpulan *logic* yang mengarah pada suatu perumusan dan penyelesaian suatu masalah pada suatu permasalahan. Arsitektur ini menyediakan *service-service* kepada komponen aplikasi lainnya melalui protokol komunikasi dalam jaringan. Masing-masing *service* akan menangani permintaan komponen aplikasi lain tersebut untuk menyelesaikan masalah yang dapat digunakan ulang.

Menurut Christudas (2019: 34), arsitektur *microservices* adalah varian dari arsitektur *service-oriented* yang membagi sebuah aplikasi menjadi koleksi *service* yang renggang namun saling berhubungan. Arsitektur ini fokus dalam membangun rangkaian *service* kecil yang otonom yang akan membangun sebuah aplikasi. Berlawanan dengan arsitektur *monolithic* yang merupakan unit aplikasi yang komponen-komponennya tidak dapat dipisahkan, arsitektur *microservice* terdiri dari banyak *service* terpisah yang independen yang saling berinteraksi untuk mendukung satu aplikasi yang sama. Pada arsitektur ini semua *service* dirancang untuk tidak bergantung pada *service* lainnya sehingga apabila ada masalah atau perubahan pada suatu *service*, *service* lainnya tidak akan terkena dampaknya.

Layanan pada aplikasi sistem informasi unit layanan terpadu Universitas Negeri Jakarta kemungkinan akan terus bertambah atau ada yang dapat diubah sesuai dengan standar operasional prosedur masing-masing lembaga, biro atau unit yang berlaku. Maka dari itu, aplikasi ini cocok dikembangkan dengan menggunakan arsitektur *microservice* karena aplikasi akan berkembang menjadi lebih besar seiring bertambahnya layanan. Dengan arsitektur *microservice*, penambahan layanan baru akan lebih mudah dengan cara pembuatan *service* baru yang otonom yang tidak mengganggu jalannya *service* yang lain. Dengan kata lain, apabila terdapat layanan baru yang ingin ditambahkan di aplikasi tersebut, maka layanan yang lainnya tidak akan terganggu. Berdasarkan pemaparan tersebut, maka dalam penelitian ini mengangkat judul “Implementasi Arsitektur *Microservice* pada Rancangan *Back-end* Aplikasi Unit Layanan Terpadu Universitas Negeri Jakarta”. Harapannya penelitian ini dapat berguna sebagai referensi pengembangan *back-end* aplikasi unit layanan terpadu dengan arsitektur *microservice*.

1.2. Identifikasi Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, maka dapat diidentifikasi masalah antara lain:

1. Pengguna layanan masing-masing lembaga, biro atau unit harus datang ke beberapa lokasi pelayanan

2. Tidak efisiennya proses pelayanan karena pelayanannya yang terpisah-pisah. Layanan yang melibatkan beberapa unit yang tidak terintegrasi dan tidak dilaksanakan dalam satu waktu
3. Beberapa unit tidak melakukan sosialisasi yang baik dalam menginformasikan layanannya
4. Belum adanya rancangan *back-end* aplikasi unit layanan terpadu Universitas Negeri Jakarta dengan menggunakan arsitektur *microservice*

1.3. Batasan Masalah

Agar memperoleh hasil penelitian yang terfokus, berdasarkan masalah-masalah yang telah teridentifikasi, penelitian dibatasi pada:

1. Penelitian ini hanya akan merancang rancangan konseptual aplikasi unit layanan terpadu Universitas Negeri Jakarta ini yang dibatasi pada bagian pengembangan *back-end* saja, tidak dengan bagian *front-end*.
2. Layanan yang akan dimasukkan ke rancangan aplikasi unit layanan terpadu Universitas Negeri Jakarta ini hanya layanan peminjaman ruang pada Biro Umum dan Kepegawaian (BUK), permohonan pencetakan pra-transkrip nilai mahasiswa pada Biro Akademik Kemahasiswaan dan Hubungan Masyarakat (BAKH), dan dukungan IT pada suatu acara pada Unit Pelaksana Teknis Teknologi Informasi dan Komunikasi (UPT TIK) Universitas Negeri Jakarta.
3. Penelitian ini menekankan pada pengimplementasian arsitektur *microservice* yang hanya menghasilkan sebuah *prototype*, bukan produk siap atau final.
4. Metode yang digunakan pada penelitian ini adalah metode pengembangan perangkat lunak Waterfall yang dimodifikasi.

1.4. Rumusan Masalah

Berdasarkan latar belakang, identifikasi masalah dan batasan masalah maka dirumuskan masalah yaitu: “Bagaimana implementasi arsitektur *microservices* pada rancangan *backend* aplikasi unit layanan terpadu Universitas Negeri Jakarta?”

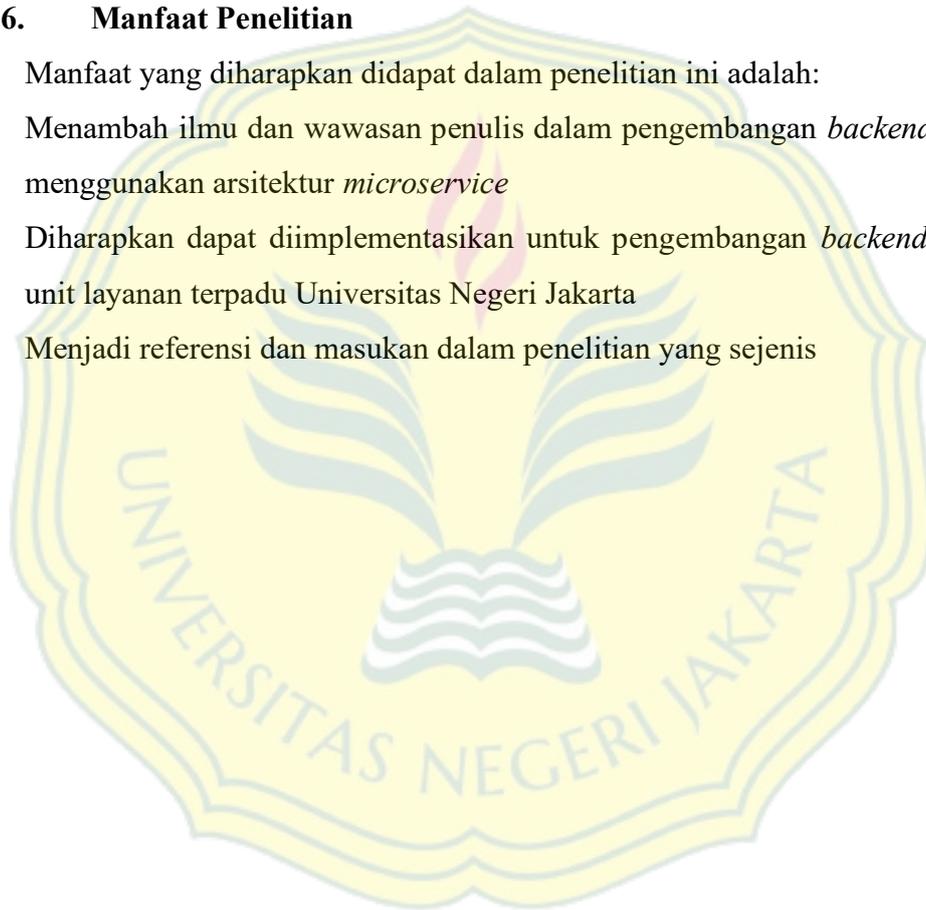
1.5. Tujuan Penelitian

Tujuan penelitian yang diharapkan pada penelitian ini adalah untuk menghasilkan sebuah rancangan perangkat lunak *back-end* dengan menggunakan arsitektur *microservice* yang digunakan dapat digunakan sebagai acuan apabila Universitas Negeri Jakarta memiliki unit layanan terpadu.

1.6. Manfaat Penelitian

Manfaat yang diharapkan didapat dalam penelitian ini adalah:

1. Menambah ilmu dan wawasan penulis dalam pengembangan *backend website* menggunakan arsitektur *microservice*
2. Diharapkan dapat diimplementasikan untuk pengembangan *backend* aplikasi unit layanan terpadu Universitas Negeri Jakarta
3. Menjadi referensi dan masukan dalam penelitian yang sejenis



*Mencerdaskan dan
Memartabatkan Bangsa*