

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam era informasi ini, integrasi merupakan salah satu kunci dalam menciptakan inovasi baru dalam berbagai aspek kehidupan. Beberapa bentuk integrasi telah dilakukan dengan menghasilkan inovasi baru yang dapat menjadi jawaban untuk beberapa permasalahan yang ada. Contoh dari integrasi tersebut adalah sebagai berikut.

- Integrasi yang dilakukan oleh Republik Kazakhstan pada ilmu pengetahuan dan pendidikan mampu memperkaya pengetahuan baru hingga membuat inovasi dalam dunia pendidikan (Myrkhalykov, Aidarova, Seidahmetov, Abishova, & Dosmuratova, 2014).
- Integrasi beberapa institusi kependidikan di Rusia menciptakan sebuah bentuk dari komunitas pendidikan berskala global yang diyakini mampu bersaing dengan institusi kependidikan lainnya (Platonova, Bogomolova, Musarskiy, & Igumnov, 2015).
- Integrasi lain yang dilakukan di Rusia juga meliputi integrasi antara ilmu pengetahuan, pendidikan, dan industri yang ditujukan untuk kolaborasi antara tiap komponen demi menciptakan masyarakat yang lebih baik (Kartashova, Shirko, Khomenko, & Naumova, 2015; Shegelman, Shchukin, & Vasilev, 2015).

Penelitian lain juga menunjukkan upaya dalam menjadikan integrasi sebagai inovasi baru terutama dalam dunia teknologi. Konsep *Intercultural Education* merupakan konsep baru yang mengedepankan integrasi antara budaya yang berbeda-beda, meski pada tahap implementasi dalam bentuk *Smart City* belum membuahkan hasil (Aguaded-Ramírez, 2017). Strategi integrasi dari penelitian akademik dengan pendidikan pada lapangan juga mampu memberikan inovasi dalam memberikan pengalaman berupa implementasi hasil didik di lapangan dengan menggunakan teknologi Sistem Informasi Geografis (Lisetskii, Terekhin,

Marinina, & Zemlyakova, 2015). Integrasi lain yang menjadi inovasi ialah integrasi antara ICT dengan pendidikan berjangka panjang yang bertujuan sebagai penunjang dalam memperluas pengetahuan, mengasah kemampuan, hingga memberikan pendidikan ekstrakurikuler (Hubackova & Klimova, 2014).

Dalam implementasi, teknologi integrasi berkembang dari waktu ke waktu. Pada mulanya, teknologi integrasi diimplementasikan dalam bentuk *point to point*. Tiap *node* saling terhubung dengan *node* lain secara langsung. Implementasi *point to point* mudah diimplementasikan dalam jaringan integrasi berskala kecil. Namun, jaringan integrasi yang dirancang menggunakan implementasi *point to point* sangat sulit untuk diperluas dan dipelihara. Kelemahan model implementasi *point to point* kemudian diatasi dengan penggunaan *hub* dan *spoke* pada jaringan integrasi. Dengan menggunakan *hub*, tiap *node* terhubung dengan *node* lain dijumpai dengan perangkat *hub* sehingga perluasan jaringan integrasi lebih mudah dan terpusat. Namun, teknologi *hub* memiliki kelemahan yaitu teknologi integrasi yang terkunci pada *vendor* dari *hub* yang digunakan dalam jaringan integrasi sehingga hanya beberapa tipe *node* yang dapat terintegrasi dalam jaringan. Teknologi implementasi berikutnya yang digunakan dalam integrasi adalah *Enterprise Message Bus*. Implementasi *Enterprise Message Bus* memanfaatkan perangkat berupa *Bus* yang digunakan dalam bertukar pesan. Kelemahan pada *Enterprise Message Bus* terletak pada integrasi yang mengharuskan tiap *node* berjalan pada *platform* yang sama. Hal ini berdampak pada kebutuhan tambahan berupa *adaptor* bagi *node* dengan *platform* berbeda sebagai penerjemah yang tidak disediakan oleh *Enterprise Message Bus*. Kelemahan pada *Enterprise Message Bus* kemudian diatasi pada teknologi *Enterprise Service Bus*. Teknologi *Enterprise Service Bus* berjalan dengan *Service Oriented Architecture* yang bersifat *loosely-coupling*. Teknologi ini juga berjalan dengan *built-in adaptor* sehingga tiap *node* tidak memerlukan *adaptor* tambahan untuk dapat terhubung dengan jaringan integrasi (Juric, 2007; Juric, 2010).

Di Indonesia sendiri, implementasi dari integrasi telah dilakukan. Integrasi tersebut digunakan dalam pemerintahan sebagai wadah integrasi serta sarana pertukaran data tanpa memedulikan perbedaan aplikasi, perbedaan sistem operasi, hingga perbedaan data basis yang digunakan. Integrasi dalam bidang pemerintahan

yang telah diimplementasikan adalah *Enterprise Service Bus* (ESB) berdasarkan arsitektur *Service Oriented Architecture* (SOA) namun khusus untuk digunakan dalam pemerintahan: *Government Service Bus* (GSB) yang bernama Manajemen Integrasi dan Pertukaran Data (MANTRA). MANTRA sendiri menggunakan konsep Web-API sebagai media pertukaran data antar instansi. Aplikasi MANTRA sendiri sudah berjalan dan telah digunakan dalam beberapa bidang pemerintahan seperti Ditjen Dukcapil, BNP2TKI, dan Pemprov Jawa Barat (Kementerian Komunikasi dan Informatika RI, 2013).

Perkembangan teknologi dalam implementasi integrasi tidak lepas dari perkembangan integrasi dan penyediaan layanan. Sistem besar memiliki kelemahan dalam mengatur fungsi yang ada di dalamnya. Hal ini dapat diraih dengan mengatur tiap komponen sebagai bagian kecil melalui orkestrasi dan mengatur tiap komponen kecil di luar sistem melalui integrasi (Kousalya, Balakrishnan, & Pethuru Raj, 2017). Salah satu teknologi penyediaan layanan yang mempengaruhi perkembangan integrasi adalah *web service*. *Web services* dapat diimplementasikan secara terdefinisi dalam bentuk fungsi tunggal. Hal ini berimbas pada kelebihan dari *web services* yang dapat dikomposisikan/diorkestrasikan untuk membuat proses bisnis (Bouguettaya, Sheng, & Daniel, 2013). Orkestrasi *web services* menjadi sebuah tantangan tersendiri dikarenakan luasnya implementasi dari kegiatan komputasi sehari-hari yang dijadikan layanan. Permasalahan ini kemudian dijadikan landasan dalam penelitian terkait untuk menciptakan landasan dalam pemilihan *web services* (Lemos, Daniel, & Benatallah, 2015). Berkaitan dengan implementasi, orkestrasi beberapa *web services* pernah dilakukan untuk menghasilkan peta tematik dalam penyediaan informasi spasial (Rautenbach, Coetzee, & Iwaniak, 2013).

Mengikuti perkembangan zaman, kebutuhan integrasi kian meningkat. Evolusi dalam teknologi informasi saat ini terbagi menjadi empat tahap sebagai berikut (Wähler, 2015).

1. *Data Integration*.

Evolusi ini berpusat pada perekaman data dan pekerjaan yang dilaksanakan dalam bentuk *batch jobs* sehingga bersifat *non-realtime*. Teknologi yang

digunakan dalam integrasi ini ialah *mainframe* dan *databases* serta membutuhkan teknologi *data processing* juga *Management Information System*.

2. *Enterprise Application Integration*.

Evolusi ini berpusat pada antarmuka dan *adapter* tiap layanan secara *realtime* sehingga perbedaan arsitektur pada tiap aplikasi tidak menjadi masalah. Teknologi yang digunakan dalam integrasi ini adalah *client-server* dan *messaging middleware* serta membutuhkan teknologi *Enterprise Routing Protocol* dan *Analytics* agar dapat bekerja.

3. *Service Oriented Architecture*.

Evolusi ini berpusat pada layanan dan *web services* secara *realtime* (Bouguettaya, Sheng, & Daniel, 2013). Penggunaan *web services* ini mengubah cara pandang organisasi dalam menghadapi perkembangan pasar terutama dalam kebutuhan proses bisnis yang semakin beragam (Delen & Demirkan, 2013). Teknologi yang digunakan ialah *extensible markup language*, *simple object access protocol*, *WS-**, dan *process modelling* serta memerlukan *business process modelling* agar dapat bekerja.

4. *Web Oriented Architecture*.

Evolusi ini diprediksikan akan berfokus pada *microservices* dan *web API* secara *realtime*. Teknologi yang dapat digunakan pada evolusi ini ialah *In-Memory Processing*, *Multiple Processing*, dan *Representational State Transfer*. Evolusi ini memerlukan teknologi *Mobile*, *Cloud*, *Fast Data*, dan *Internet of Things* dalam operasinya.

Saat ini, evolusi pada *Web Oriented Architecture* (WOA) mulai berkembang terutama dengan munculnya teknologi *microservices*. Teknologi ini berkembang dari kebutuhan SOA namun dalam skala yang lebih kecil (Haddad, 2014). Tidak seperti ESB yang berpusat pada *bus* utama sebagai media integrasi, *microservices* berfokus pada kecerdasan pada *service gateway* dalam menentukan layanan mana yang ingin digunakan (Clark, 2016). Terdapat permasalahan yang dihadapi dalam integrasi *microservices*, salah satunya ialah *microservices integration and discovery*. Permasalahan ini bersumber dari konsep *microservices* yang berpusat pada *service gateway* sehingga diperlukan sarana tambahan bagi tiap perancang

layanan dalam menemukan dan melihat definisi dari *microservices* (Zimmermann, 2017). Secara garis besar, SOA cocok digunakan untuk *business process* berskala besar dan kompleks sementara *microservices* lebih cocok digunakan untuk sistem kecil yang terbagi menjadi bagian-bagian kecil yang berdiri sendiri. Dalam tingkat orkestrasi, SOA memiliki kelebihan dikarenakan penggunaan ESB sementara *microservices* hanya mengandalkan dirinya sendiri untuk melakukan orkestrasi (Watts, 2017).

Beberapa penelitian berkaitan dengan implementasi *microservices* memiliki beragam hasil. Hasil dari penelitian berkaitan tersebut dijabarkan sebagai berikut.

- Dibandingkan dengan arsitektur *software* monolitik, arsitektur *microservices* mampu menekan jumlah penggunaan *resources* hingga sebagian besar namun dengan hasil serupa (Mazlami, Cito, & Leitner, 2017).
- Penelitian berkaitan dengan SOA menghasilkan arsitektur berbasis *microservices* yang diimplementasikan dalam prototipe MicroART (Francesco, 2017).
- Pemanfaatan arsitektur *microservices* juga mampu memberikan integrasi yang lebih banyak pada perangkat kerja *Internet of Things* (Sun, Li, & Memon, 2017).
- Penelitian berkaitan dengan keamanan menghasilkan pandangan mengenai perangkat kerja *microservices* berbasis *Internet of Things* dalam usaha menjawab tantangan pengembangan arsitektur *microservices* pada *Internet of Things* yang aman (Lu, Huang, Walenstein, & Medhi, 2017).
- Ada pendapat dari beberapa peneliti yang merasa bahwa perancang *microservices* memiliki tugas yang cukup berat dalam memelihara *microservices* sehingga diperlukan pendekatan lebih pada *microservices* dalam mengatasi permasalahan tersebut (Sampaio, et al., 2017).
- Penelitian berkenaan dengan *middleware* menghasilkan *middleware* berbasis *microservices* yang digunakan untuk *digital factory* (Ciavotta, Alge, Menato, Rovere, & Pedrazzoli, 2017).

- Penelitian berkenaan dengan orkestrasi *microservices* menghasilkan arsitektur yang dapat digunakan dalam orkestrasi layanan *Industrial Internet of Things* (Rufino, Alam, Ferreira, Rehman, & Tsang, 2017).
- Teknologi *realtime microservices* digunakan bersamaan dengan *environmental sensor* dalam menentukan rute pengantaran barang berbahaya di wilayah permukiman (Cherradi, Bouziri, Boulmakoul, & Zeitouni, 2017).

Teknologi integrasi saat ini telah berkembang menjadi lebih luas. Dari segi arsitektur, SOA dan WOA merupakan teknologi integrasi yang dikenal saat ini. Kedua teknologi integrasi ini memiliki kelebihan dan kekurangan masing-masing yang tidak dapat dipisahkan. SOA memiliki *dedicated system* berupa ESB yang digunakan dalam integrasi namun menjadikan aplikasi sangat berat karena memiliki beban tambahan baik pada proses maupun waktu eksekusi yaitu *message transformation* dari dan menuju *Simple Object Access Protocol*. Sebaliknya, WOA dengan memanfaatkan *microservices* memiliki kemudahan dalam penggunaan karena hanya menggunakan *Representational State Transfer* dalam berkomunikasi namun dapat menjadi tantangan tersendiri dalam orkestrasi dan integrasi karena bergantung pada desain *microservices* yang diciptakan (Watts, 2017). Meski *trade-off* dapat terjadi jika digunakan *middleware* pada *microservices* terutama pada kompleksitas penggunaan, penggunaan *middleware* dapat membantu manajemen *microservices* (Wöhner, 2015).

Dalam penelitian ini, permasalahan yang digarisbawahi adalah kebutuhan integrasi antar layanan yang semakin besar. Penelitian ini berupaya untuk menciptakan *middleware* dengan harapan mampu mengintegrasikan *microservices* yang ada. *Middleware* yang dibangun akan berfungsi sebagai *pool* sekaligus *orchestrator* untuk *microservices* yang dikaitkan pada *middleware*. Teknologi yang diterapkan pada *middleware* ini adalah teknologi *microservices* berbasis *Representational State Transfer* mengikuti perkembangan *microservices* yang ada.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang muncul pada latar belakang, terdapat beberapa masalah yang dapat diselesaikan yaitu sebagai berikut.

1. Apa saja yang diperlukan dalam membangun *middleware* dengan teknologi *microservices* berbasis REST?
2. Bagaimana cara agar *microservices* dapat dikenali oleh *middleware*?
3. Bagaimana cara untuk melakukan integrasi terhadap *microservices* yang sudah dikenali oleh *middleware*?

1.3 Tujuan Penelitian

Adapun tujuan yang harus dicapai dalam penelitian ini adalah sebagai berikut.

1. Mendefinisikan keperluan pembangunan *middleware* dengan teknologi *microservices* berbasis REST.
2. Mendefinisikan cara untuk memperkenalkan *microservices* terhadap *middleware*.
3. Menerapkan rancangan integrasi terhadap *microservices* yang sudah dikenali oleh *middleware*.

1.4 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut.

1. Layanan *microservices* yang dikenalkan dan diintegrasikan adalah layanan yang dikembangkan oleh tim peneliti dalam permasalahan *e-commerce*.
2. Kebutuhan yang akan dipenuhi hanya kebutuhan untuk pengenalan *microservices* dan orkestrasi *microservices*.
3. Kebutuhan integrasi dan orkestrasi didiskusikan dengan pemilik layanan secara manual (diluar sistem).

1.5 Sistematika Penulisan

Sistematika penyusunan skripsi ini adalah sebagai berikut.

BAB 1 PENDAHULUAN

Berisi pembahasan masalah umum yang diangkat pada penelitian, di dalamnya terdapat latar belakang, rumusan masalah, batasan masalah, tujuan penelitian dan sistematika penulisan.

BAB 2 KAJIAN PUSTAKA

Berisi mengenai kajian teori yang digunakan di dalam penelitian. Pada bab ini dibahas dasar teori mengenai layanan berbasis web (*Web Service*), *Representational State Transfer* (REST), *Service Oriented Architecture* (SOA), dan *Enterprise Service Bus* (ESB).

BAB 3 METODOLOGI PENELITIAN

Berisi dasar teori mengenai metodologi yang digunakan untuk melakukan penelitian, metodologi meliputi desain penelitian, alat dan bahan penelitian, dan metode penelitian yang di dalamnya terdapat pengumpulan data, dan proses pengembangan perangkat lunak.

BAB 4 HASIL PENELITIAN DAN PEMBAHASAN

Berisi penjelasan dari hasil penelitian yang dilakukan, hasil penelitian berupa objek yang akan ditampilkan.

BAB 5 KESIMPULAN DAN SARAN

Berisi kesimpulan dan saran yang didapat dari penelitian dari mulai merumuskan masalah sampai dengan selesai.