

DESAIN SISTEM TERFEDERASI DENGAN PENDEKATAN MICROSERVICE ARCHITECTURE PADA KASUS STUDI SISTEM PELAPORAN PAJAK

Astie Darmayantie
Fakultas Teknologi Industri Universitas Gunadarma,
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
astie@staff.gunadarma.ac.id

Abstrak

Trend ekonomi global saat ini telah berubah dari ekonomi berbasis tenaga kerja menjadi ekonomi berbasis informasi. Jumlah aplikasi yang digunakan menunjukkan semakin banyaknya sirkulasi data dan informasi, sehingga menciptakan lingkungan pasar berbasis data. Dalam pengembangannya terdapat beberapa pendekatan dalam mengembangkan sebuah sistem yaitu pendekatan terpusat (*centralized*), terdistribusi (*distributed*) dan terfederasi (*decentralized*). Meskipun pendekatan terpusat lebih disukai dalam banyak kasus, namun, dengan semakin banyaknya pengguna, pendekatan ini rentan terhadap permasalahan *single-point of failure*. Makalah ini menyajikan sebuah desain sistem terfederasi berbasis Arsitektur Berorientasi Layanan-mikro. Studi penggunaan dari desain sistem ini diimplementasi pada sistem pelaporan pajak bulanan dalam suatu organisasi X yang melibatkan 5 stakeholder yang berbeda. Solusi teknologi berbasis web service, RESTful-API, diusulkan untuk menjamin interoperabilitas dari sistem terfederasi. Desain sebuah sistem manajemen layanan dan integrasi juga diusulkan untuk mengelola layanan yang tersedia, sehingga menjadikan desain yang diusulkan untuk dapat dikembangkan pada pengembangan lanjutan di masa mendatang.

Kata Kunci: REST-API, Repositori Service, Service Oriented Architecture, Web Service.

Abstract

The global economy rapidly evolves from a labor-based economy to an information-based economy. The growing number of applications being used infers to a large number of data and information circulation, thus creating data-driven market environment. In this era, there are three approaches can be conducted in developing a system: centralize, distributed, and decentralized. Though centralized-approach is more preferable in most cases, however, with the growing number of usage, it is prone to single point of failure. This paper present a decentralized system design using Microservice Architecture approach. The use case implementation of the design is for financial reporting system within an organization. REST-API based web service is adopted to ensure the interoperability of each systems. A design of Service integration and Management System is also proposed to manage available services, thus making the proposed design to be scalable for future development.

Keywords: Service Repository, REST-API, Service Oriented Architecture, Web Service.

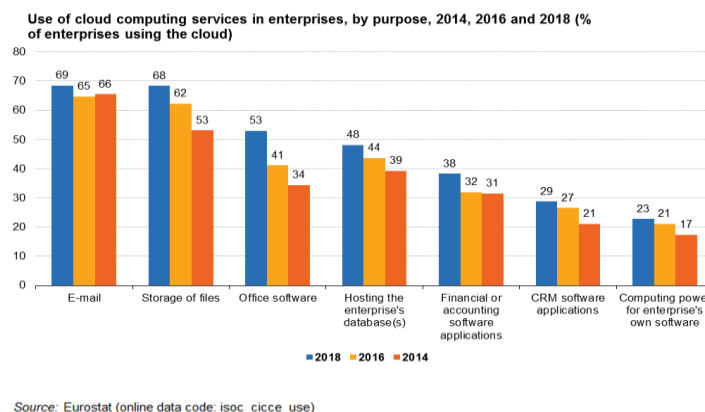
PENDAHULUAN

Trend ekonomi global saat ini telah berkembang dari ekonomi berbasis tenaga kerja menjadi ekonomi berbasis informasi [1].

Teknologi internet dan otomasi berkembang dengan kecepatan yang sangat tinggi. Teknologi digunakan pada sebagian besar kehidupan manusia mulai dari sektor kesehatan hingga dalam menyelesaikan tugas

sehari-hari seperti berbelanja bahan makanan. Kemajuan teknologi membentuk sifat perilaku masyarakat dalam menggunakan teknologi. Dua puluh tahun yang lalu pengguna Web hanya mengakses informasi melalui aplikasi Web statis. Saat ini, pengguna dapat dengan mudah berpartisipasi secara aktif pada keberhasilan Web dengan menghasilkan konten, umpan balik, peringkat, dan interaksi sejenis lainnya. Seiring dengan berbagai langkah digitalisasi ditempuh baik oleh pihak swasta maupun pemerintah dan jumlah aplikasi perangkat lunak yang meningkat tiap tahunnya, dapat disimpulkan bahwa jumlah volume dan jenis informasi yang beredar saat ini sangat besar. Berkembangnya teknologi internet menjadi salah satu pendorong terobosan teknologi seperti *Big data*, *artificial intelligence*, *Internet of Things (IoT)* dan *Cloud Computing*. Aplikasi berbasis cloud lebih disukai karena fleksibilitas dan kemudahannya. Pergeseran trend penggunaan aplikasi,

menginduksi pendekatan yang berbeda dalam pengembangan teknologi guna mengakomodasi permintaan *user*. Terobosan teknologi ini tersebut juga semakin memudahkan proses pengumpulan dan pengarsipan data. Terlihat pada Gambar 1 [2], jumlah perusahaan yang menggunakan jasa Cloud Computing terus meningkat tiap tahunnya. Terdapat tiga jenis pendekatan dalam pembangunan sistem yaitu terpusat, terdistribusi dan terfederasi [1]. Pendekatan terdistribusi merujuk pada komputasi yang tersebar di beberapa simpul (node), sedangkan konsep terfederasi menitikberatkan pada kebebasan setiap node untuk menjalankan prosesnya. Pada konsep federasi masing-masing node bekerja secara independent tanpa bergantung pada simpul lain sehingga keseluruhan sistem menjadi otonom. Melalui mekanisme ini, setiap proses yang didukung oleh pendekatan federasi menjadi tidak rentan terhadap resiko *single-point of failure* seperti yang terjadi konsep sistem tersentralisasi dapat diatasi [3].

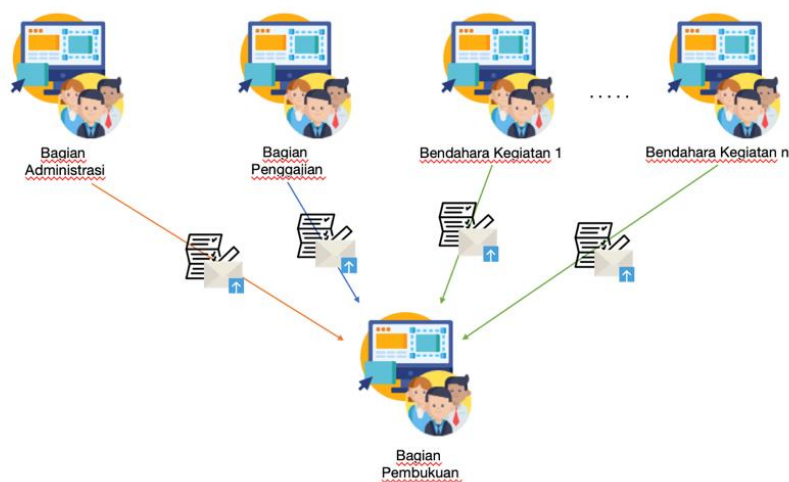


Gambar 1. Jumlah Perusahaan yang Menggunakan Jasa Cloud Computing Tahun 2014, 2016, 2018

Dengan pendekatan terfederasi, pemangku kepentingan dari sistem yang ada dapat bekerja secara independen dalam menjalankan proses bisnis yang dimiliki, tanpa bergantung pada suatu simpul atau pemangku kepentingan tertentu [4]. Pendekatan sistem terfederasi juga dapat menjadi solusi terhadap isu mengenai kepemilikan data atau data *ownership*. Sistem dengan pendekatan terfederasi tidak bergantung pada satu entitas tertentu dalam menangani data dan informasi yang bersirkulasi pada sistemnya.

Pada sistem federasi memungkinkan setiap stakeholder untuk memiliki wewenang penuh terhadap pengelolaan sistem, data, maupun informasi yang dimiliki sesuai dengan standar yang ditentukan. Dengan pendekatan ini juga memungkinkan para pemangku kepentingan sistem untuk dapat mengembangkan layanan, proses, produk, ataupun tools sesuai dengan permasalahan

yang dimiliki pada masing-masing divisi. Bukti potong pajak merupakan sebuah dokumen yang diterbitkan oleh pemotong pajak/bendahara maupun divisi yang ditunjuk oleh suatu instansi dan akan digunakan untuk pelaporan SPT Tahunan orang pribadi yang menerima penghasilan. Setiap organisasi memiliki keunikan dalam menjalankan proses bisnisnya. Penelitian ini mengambil studi kasus proses bisnis penerbitan bukti potong pajak bulanan pada organisasi X. Dalam proses penerbitan bukti potong ini melibatkan 5 *stakeholder* seperti tergambar pada Gambar 2. Terlihat pada Gambar 2, proses yang berjalan saat ini telah memanfaatkan beberapa teknologi seperti dokumen digital dan media komunikasi, namun, secara pelaksanaan masih dilakukan secara manual. Setiap bulan, bagian pembukuan akan mengumpulkan dokumen digital yang diperlukan kepada setiap bagian menggunakan email.



Gambar 2. Alur Relasi Pemangku Kepentingan

Bagian pembukuan sendiri belum menggunakan sistem basis data dalam mengarsipkan data dan informasi yang ada, sehingga semua data tersimpan dalam file excel. Selain permasalahan manajemen pengarsipan informasi, proses integrasi informasi yang diterima dilakukan secara manual dengan melakukan *copy-paste*. Hal ini menyebabkan proses bisnis yang berjalan dirasa kurang efisien dan memakan waktu. Dengan skala, keragaman, dan kecepatan perubahan teknologi web mengakibatkan permasalahan integrasi antar arsitektur Web. Arsitektur Web terus berubah ketika terdapat klien, server, proksi, dan gateway baru, maupun dihapus dari ekosistem sistem. Komponen penyusun sistem ini, akan terus berubah untuk memberikan kemampuan baru, seperti menambahkan sumber daya baru dalam bentuk situs web baru dan layanan web, mendukung representasi baru untuk sumber daya (misalnya, gambar novel dan format video), mengintegrasikan perangkat keras novel, menambahkan novel fitur untuk agen pengguna (browser web), dan sebagainya. Tahun 2011 B. David, M. Klems, S. Tai, dan M. Menzel [5], memperkenalkan sebuah sistem *hashtable* terdistribusi yang bernama *MetaStorage*. *Metastorage* menggunakan arsitektur terdistribusi yang memungkinkan penggunaan beberapa *Cloud Storage Service*. *MetaStorage* menerapkan konsep *reuse* dan skema replikasi seperti yang diterapkan oleh Amazon Dynamo [6], dan menggunakan

sebuah protocol pesan asinkronus bernama *Stage Event-Driver Architecture* (SEDA). Penelitian lainnya [7] menjelaskan bahwa style arsitektur Representational State Transfer (REST) memungkinkan seluruh industri *Software-as-a-Service* (SaaS) untuk membangun sistem terdesentralisasi dengan memberikan kemampuan *malleability* pada arsitektur sebuah perangkat lunak. Berdasarkan uraian tersebut, penelitian ini menjelaskan tentang desain model sistem terfederasi dengan pendekatan arsitektur berbasis *microservice*. Desain system kemudian diimplementasi pada kasus studi ini adalah pembuatan bukti potong pajak bulanan pada sebuah organisasi X yang melibatkan lima stakeholder. Model ini dikembangkan dengan mempertimbangkan kepentingan serta fungsi dari masing-masing *stakeholder*. Integrasi aplikasi dari masing-masing *stakeholder* sistem memanfaatkan teknologi *web service* berbasis RESTful-API [8][9].

Layanan yang dimiliki oleh masing-masing *stakeholder* dari system, kemudian, dapat diorkestrasi menjadi sebuah layanan informasi keuangan yang dapat diakses oleh divisi lain yang membutuhkan informasi, contohnya oleh sistem dashboard pimpinan organisasi. Sebuah desain sistem manajemen layanan dan integrasi (*Service integration and Management System*) juga diperkenalkan dalam pengolaan layanan web services yang tersedia, untuk menjamin skalabilitas desain sistem untuk pengembangan sistem lanjutan.

METODE PENELITIAN

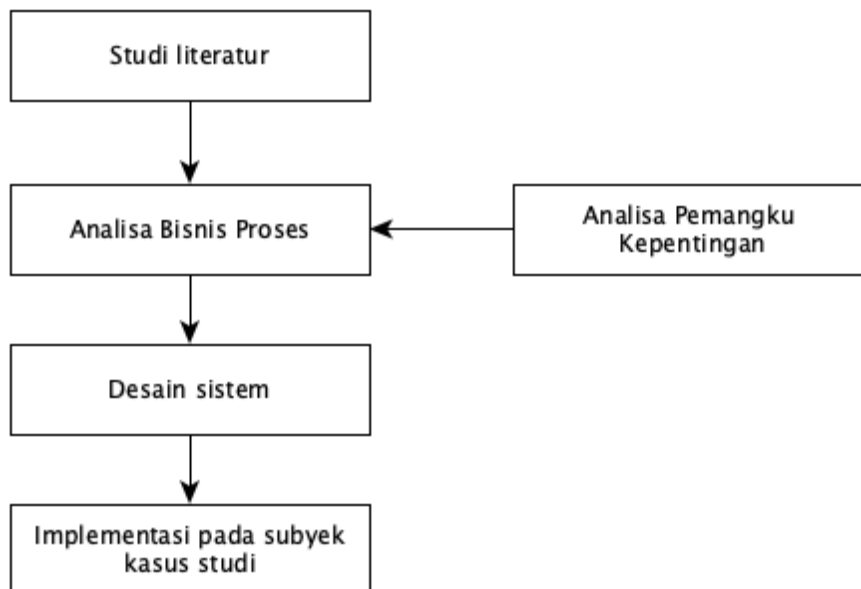
Tujuan utama dari pengembang sistem adalah untuk mengembangkan sistem yang memenuhi tujuan utama yang telah ditetapkan oleh para pemangku kepentingan, termasuk tujuan-tujuan yang terkait dengan penciptaan, produksi, dan terminasi sistem [10]. Sehingga, pendekatan pengembangan sistem untuk masing-masing permasalahan menjadi unik. Sebagian besar aplikasi perangkat lunak web mengikuti model *client-server* terpusat, beberapa menerapkan terdis-tribusi, dan beberapa perusahaan maju telah menggunakan pendekatan terfederasi.

Solusi permasalahan sistem sangat bergantung pada kebutuhan dari masing-masing pemangku kepentingan. Hal ini

menyebabkan tahapan analisis pemangku kepentingan yang memberikan informasi secara holistik, memegang peranan kunci dalam proses desain suatu sistem yang baik. Sub-bagian ini akan menjelaskan metode penelitian yang digunakan dalam menganalisa permasalahan yang ada serta mengembangkan solusi yang diajukan berupa desain sistem terfederasi dengan pendekatan arsitektur *microservice*. Adapun tahapan penelitian yang dilakukan terlihat pada diagram Gambar 3.

Studi Literatur

Integrasi data merupakan serangkaian metode yang bertujuan untuk menggabungkan data dari berbagai sumber untuk memberikan representasi tunggal yang seragam [11].



Gambar 3. Diagram Alur Penelitian

Kegiatan integrasi data mencakup berbagai bentuk penggunaan kembali informasi, seperti memindahkan data dari satu basis data ke basis data lain, menerjemahkan pesan untuk transaksi bisnis-ke-bisnis, atau menyediakan akses ke data dan dokumen terstruktur melalui portal Web. Masalah utama yang harus dipecahkan dalam melakukan integrasi data adalah dalam melakukan definisi serta pemeliharaan skema terintegrasi, pemetaan antara skema global dan skema lokal dari sumber data yang terlibat, serta mencari solusi terhadap ketidaksesuaian antara model dan skema data pada skema heterogen. Pertumbuhan jumlah aplikasi yang tersedia menjadi salah satu faktor pendorong konsep data mashup.

Integrasi Aplikasi merupakan topik yang dieksplorasi dan diteliti dalam beberapa tahun terakhir dengan seiringnya perkembangan teknologi dan aplikasi [11]. Beberapa teknologi yang diterapkan seperti Remote Procedure Calls (RPCs), broker objek, dan, layanan Web. Dalam skema integrasi aplikasi, lapisan logika proses bisnis dari aplikasi dikembangkan dengan mengintegrasikan fungsi-fungsi yang dimiliki oleh komponen sistem lain. Skema integrasi ini dapat berjalan dengan baik jika masing-masing komponen sistem memiliki antarmuka pemrograman dan mekanisme pertukaran yang stabil. Web Service merupakan perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. Web service memungkinkan aplikasi web

yang ada untuk saling berinteraksi melalui sebuah layanan. REST merupakan sebuah akronim dari *Representational State Transfer* [9], sebuah konsep interaksi antar perangkat lunak yang diperkenalkan oleh Roy Fielding pada tahun 2000. Pendekatan REST menggunakan sebuah *identifier* dalam mengidentifikasi sumber daya tertentu yang terlibat dalam sebuah interaksi antara komponen sistem. *State* dari sebuah *resource* pada suatu waktu tertentu disebut sebagai *resource representation* [12]. Representasi terdiri dari data, metadata yang menggambarkan data dan tautan hypermedia. Sedangkan, API (*Application Programming Interface*) merupakan sebuah antarmuka, yang terdiri atas kumpulan instruksi yang disimpan dalam bentuk library, yang menjelaskan mekanisme berinteraksi antara suatu perangkat lunak dengan perangkat lunak lainnya.

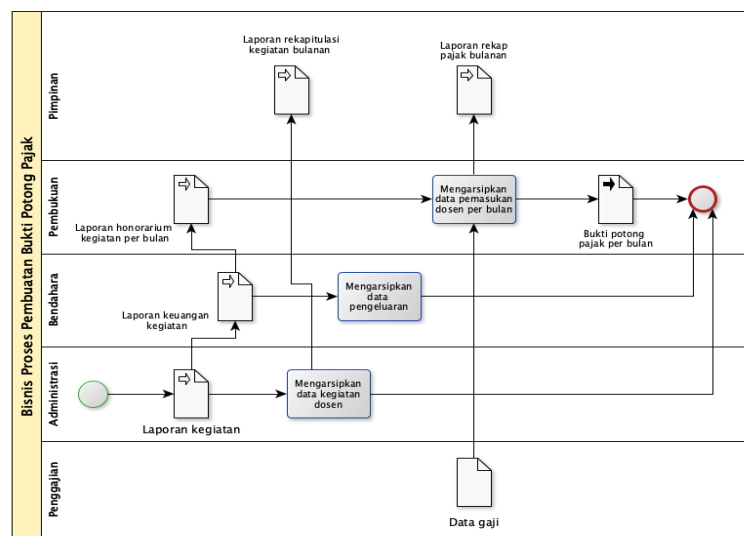
Analisa Bisnis Proses

Bukti potong pajak merupakan sebuah dokumen yang diterbitkan oleh pemotong pajak/bendahara maupun divisi yang ditunjuk oleh suatu instansi dan akan digunakan untuk pelaporan SPT Tahunan orang pribadi yang menerima penghasilan. Dalam studi kasus yang dikaji, bukti potong pajak pada organisasi X diterbitkan oleh bagian pembukuan. Seperti tersaji pada diagram BPMN (*Business Process Modelling Notation*) - Gambar 4, terdapat beberapa stakeholder yang terlibat dalam proses bisnis ini yaitu pimpinan institusi, bagian pembu-

kuan, bendahara kegiatan, bagian administrasi, dan bagian penggajian. Bagian pembukuan membutuhkan beberapa informasi yang dimiliki oleh bagian lain dalam menyusun bukti potong untuk karyawan diantaranya dari: bendahara kegiatan, bagian administrasi, dan bagian penggajian. Proses pengumpulan yang saat ini dilakukan secara manual dengan mengumpulkan file dokumen dalam bentuk excel sebelum tenggat waktu tertentu. Meskipun telah memanfaatkan teknologi, terdapat beberapa kendala yang dihadapi pada proses bisnis yang berjalan saat ini, salah satunya adalah proses integrasi informasi. Salah satu permasalahan yang timbul adalah ketidakseragaman nama karyawan yang dicantumkan pada dokumen yang diberikan karena kesalahan ejaan. Hal ini disebabkan karena proses penginputan nama masih dilakukan secara manual dan tidak menggunakan referensi. Keberagaman

format dokumen yang dikumpulkan juga menjadi salah satu tantangan lain yang menyebabkan proses integrasi informasi yang ada dilakukan satu persatu secara manual. Masalah lain adalah keberagaman format informasi yang diberikan oleh masing-masing stakeholder. Proses integrasi informasi dari masing-masing *stakeholder* yang ada saat ini dilakukan dengan proses *copy-paste*. Resiko kesalahan terjadi dalam penginputan data menjadi besar dengan jumlah kegiatan yang sangat banyak.

Berdasarkan permasalahan tersebut, diperlukan sebuah pendekatan yang memungkinkan otomatisasi pada proses integrasi data/informasi pada bagian terkait untuk menyederhanakan tugas. Pendekatan yang dibutuhkan adalah pendekatan yang mempertimbangkan tugas dan fungsi masing-masing pemangku kepentingan.



Gambar 4. Diagram Bisnis Proses Penerbitan Bukti Potong Pajak Bulanan

HASIL DAN PEMBAHASAN

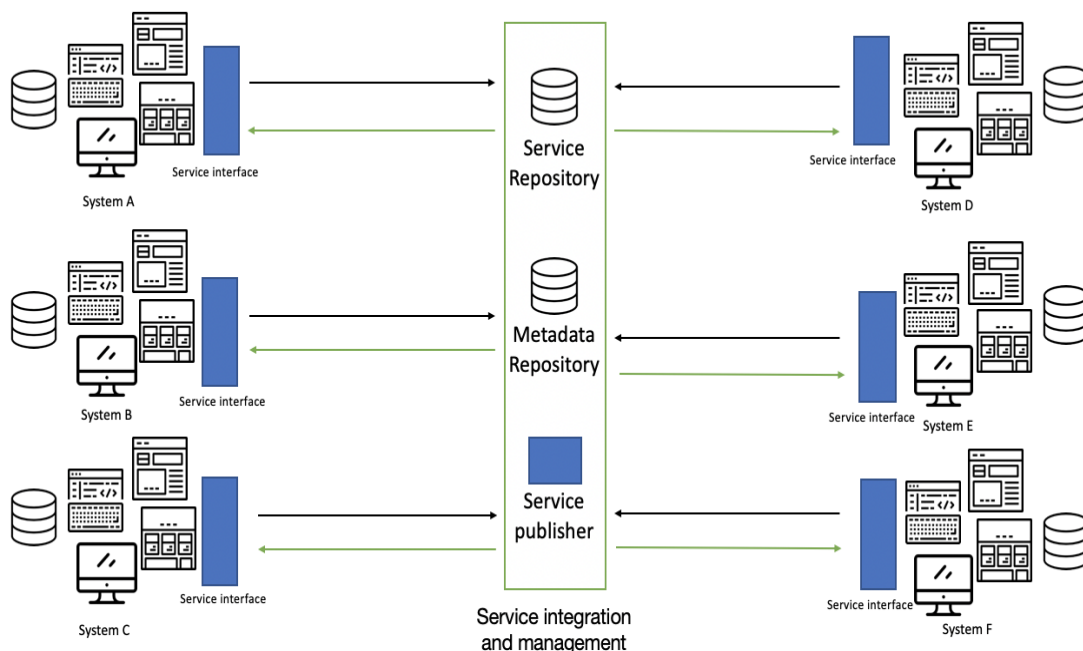
Berdasarkan uraian tersebut, sebuah desain sistem terfederasi dengan pendekatan *microservice* architecture [13] dikembangkan sebagai solusi dari permasalahan yang ada. Adapun hasil dari penelitian ini adalah sebuah desain sistem terfederasi dengan pendekatan arsitektur *microservice*. Desain ini kemudian diimplementasi (pilot project) dalam studi kasus penerbitan bukti potong pajak bulanan pada organisasi X.

Desain Umum Sistem

Sistem A, B, sampai F pada Gambar 5 merupakan ilustrasi sistem yang dimiliki oleh masing-masing *stakeholder*. *Stakeholder* ini

mencakup *stakeholder* internal maupun *stakeholder* eksternal. Setiap sistem beroperasi secara independen yang dilengkapi dengan sebuah antar-muka layanan (*service interface*).

Antar-muka layanan ini merupakan mekanisme yang digunakan oleh sistem yang ada untuk saling bertukar informasi. *Service interface* ini dapat berupa RPC, REST-API, SOAP maupun bentuk lain sesuai dengan kesepakatan atau standar yang berlaku pada suatu organisasi. Sistem A, B, sampai F pada Gambar 5 merupakan ilustrasi sistem yang dimiliki oleh masing-masing *stakeholder*. *Stakeholder* ini mencakup *stakeholder* internal maupun *stakeholder* eksternal.



Gambar 5. Desain Arsitektur Sistem Terfederasi Berbasis Arsitektur *Microservice*

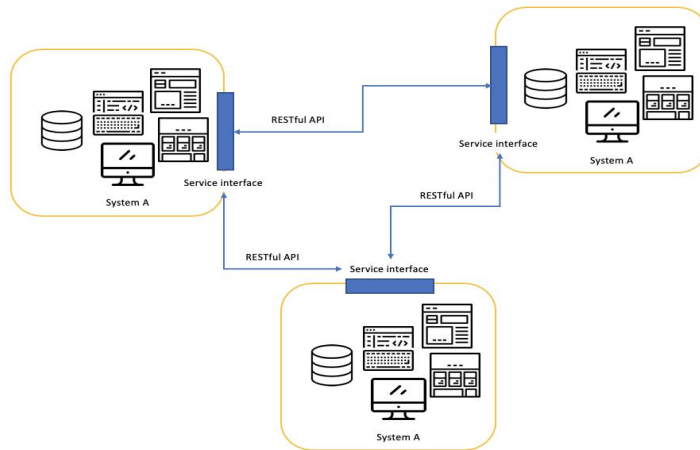
Setiap sistem beroperasi secara independen yang dilengkapi dengan sebuah antar-muka layanan (*service interface*). Antar muka layanan ini merupakan mekanisme yang digunakan oleh system yang ada untuk saling bertukar informasi. Service interface ini dapat berupa RPC, REST-API, SOAP maupun bentuk lain sesuai dengan kesepakatan atau standar yang berlaku pada suatu organisasi. Dalam desain yang diajukan service interface yang digunakan ialah berbasis REST-API. Upan balik dari web service berbasis REST-API ini berupa JSON (JavaScript Object Notation) yang merupakan sebuah format pertukaran data berbasis teks.

Dengan kebebasan yang diberikan kepada masing-masing *stakeholder*, dalam implementasinya, desain ini membutuhkan sebuah standar yang mendefinisikan bagaimana *stakeholder* dalam membangun sebuah sistem. Pendekatan sistem yang dirancang tidak menitik beratkan pada keseragaman teknologi, tetapi pada standar bagaimana sebuah sistem dapat dibangun dan beroperasi.

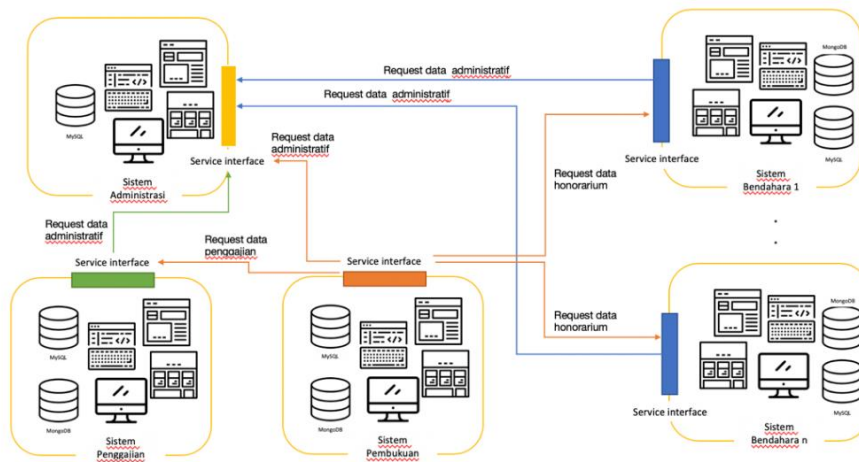
Seperti yang telah dijabarkan sebelumnya, salah satu masalah krusial dari proses integrasi adalah dalam proses definisi sistem dan pemeliharaan layanan. Penelitian ini

memperkenalkan sebuah sub-system *Service integration and Management System* yang bertugas untuk mengarsipkan serta mengelola seluruh layanan yang dimiliki oleh aplikasi. Beberapa entitas yang menyusun sistem ini antara lain adalah: (1) **Service repository** - merupakan database API dan layanan yang dimiliki sistem internal organisasi. (2) **Metadata repository** - menyimpan informasi skema data yang dimiliki oleh masing-masing sistem. (3) **Service publisher** - merupakan satu entitas tambahan jika diperlukan. Entitas ini bertugas untuk layanan yang dapat diakses oleh pihak eksternal divisi informasi. Gambar 6 merupakan ilustrasi interaksi dari masing-masing sistem yang terfederasi dengan pendekatan *microservice* architecture. Seperti terlihat pada Gambar 5, masing-masing sistem berdiri secara independen yang selanjutnya dapat diilustrasikan sebagai node/simpul. Masing-masing node kemudian dapat berinteraksi dengan node lain melalui sebuah layanan atau *web service* berbasis RESTful API yang dimiliki oleh masing-masing node.

Terlihat pada Gambar 6, sistem dimiliki oleh pemangku kepentingan memiliki sebuah antar-muka layanan sebagai media penghubung.



Gambar 6. Desain Interaksi antar Sistem dengan Pendekatan *Microservice*



Gambar 7. Alur Informasi pada 4 Komponen Stakeholder dengan Pendekatan Arsitektur *Microservice*

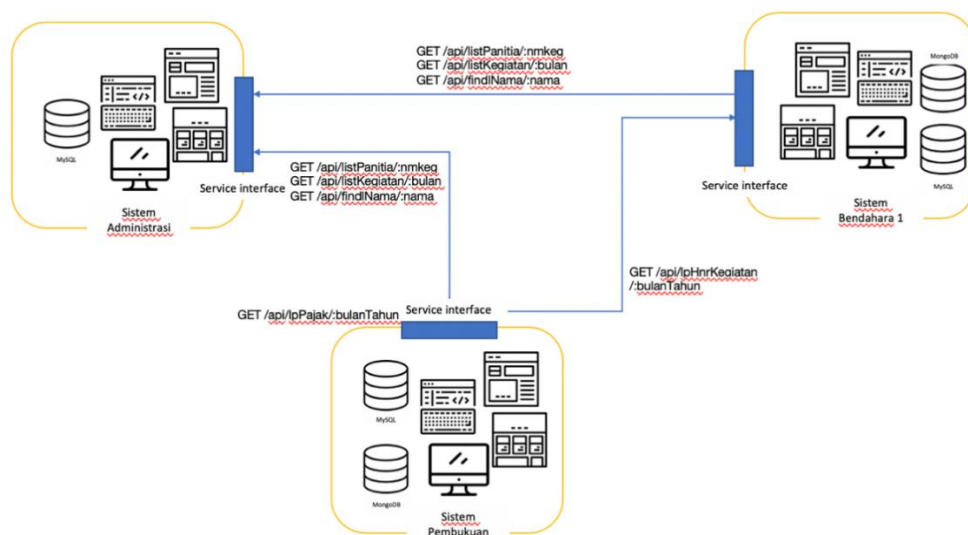
Masing-masing sistem dapat bertukar data dan informasi yang dimiliki melalui *service* berbasis RESTful API. Masing-masing sistem, secara independen, dapat mendefinisikan *service* yang disediakan tanpa bergantung pada entitas luar sistem tersebut sesuai dengan standar yang telah disepakati.

Implementasi pada Kasus Studi

Sebagai pilot-project, beberapa komponen pada desain sistem yang diajukan telah berhasil diimplementasi. Diagram alur interaksi aplikasi masing-masing *stakeholder* digambarkan pada Gambar 7. Pada Gambar 7 terlihat setiap sistem akan melakukan *request*

data kepada bagian administrasi ketika membutuhkan data terkait administrasi. Sebagai contoh pada data nama, sistem lain akan mendapat notifikasi dari sistem jika terdapat penambahan data nama pada sistem administrasi yang *mentrigger* masing-masing sistem untuk melakukan *request* data nama baru dan mengupdate data yang tersimpan pada masing-masing sistem kepada secara otomatis. Pada skema ini data yang digunakan pada suatu sistem mengacu pemilik data utama dan diakses melalui web service berbasis REST-API. Gambar 8 menjelaskan ilustrasi layanan *microservice* dari tiga *stakeholder* sistem yaitu pada sistem bendahara, sistem pembukuan dan sistem administrasi. Setiap sistem dapat mengakses tiga layanan yang dimiliki oleh sistem administrasi seperti informasi nama kegiatan per bulan, list karyawan yang terlibat pada suatu kegiatan dan nama karyawan.

Selanjutnya, Sistem pembukuan dapat mengakses informasi laporan honorarium yang diterima oleh karyawan dengan memanfaatkan *service* yang disediakan oleh sistem bendahara. Permasalahan utama disebabkan oleh data nama karyawan. Dengan menerapkan pendekatan layanan-mikro, sistem lain mereferensi data kepada sumber data utama. Masing-masing *stakeholder* data mendefinisikan sebuah *service* sesuai dengan kebutuhan pihak eksternal tanpa mengganggu proses maupun infrastruktur pada sistem utama. Pihak eksternal dapat mengakses informasi yang dimiliki oleh sistem tersebut melalui perantara *service*, tanpa mengakses langsung kedalam sistem utama. Dengan demikian, isu *ownership* maupun validitas data maupun informasi dapat ditelusuri dengan menganalisa interaksi antara layanan-mikro yang ada.



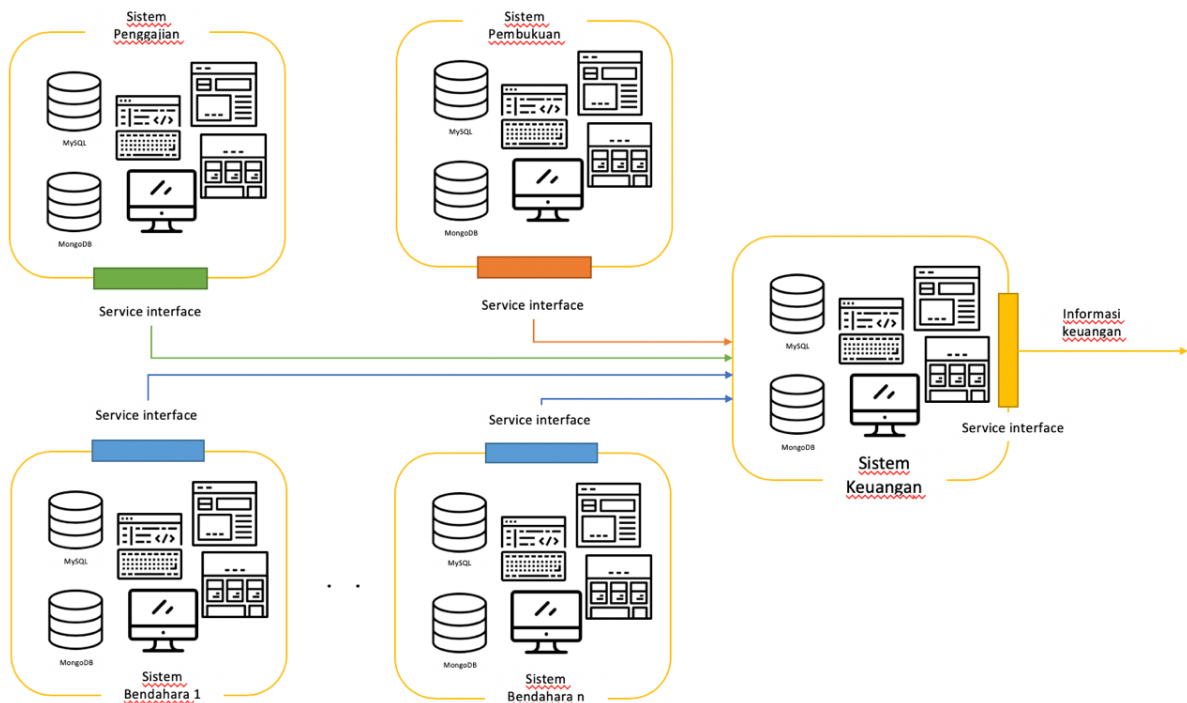
Gambar 8. Implementasi Pilot dari Perancangan Sistem Terfederasi dengan Pendekatan Arsitektur *Microservice*

Mekanisme komunikasi antar sistem ini memungkinkan proses integrasi informasi dilakukan secara otonom sesuai dengan tupoksi dan kepentingan yang dimiliki oleh masing-masing pemangku kepentingan.

Pengembangan Lanjutan

Gambar 9 menjelaskan sebagai pengembangan lanjutan, pelayanan yang disediakan oleh bagian penggajian, pembukuan dan bendahara dapat di wrap-up menjadi sebuah layanan informasi keuangan yang disusun oleh *microservice* penggajian, *microservice* honorarium dan *microservice* pajak. Layanan informasi keuangan ini dapat diakses oleh divisi lain yang membutuhkan

informasi ini melalui sebuah web service. *Service integration and Management System* kemudian dapat dikembangkan dalam mengelola dan mnginven-tarisir informasi mengenai layanan yang tersedia pada sistem yang dimiliki oleh masing-masing stakeholder. Salah satu masalah utama dalam mengembangkan sistem yang tidak terintegrasi adalah sumber data. Para pengembang dengan menggunakan skema ini sistem dapat mengakses informasi mengenai data dan informasi yang tersedia pada sistem yang ada dan bagaimana cara mengakses data tersebut melalui *platform Service integration and Management System* ini.



Gambar 9. Alur Informasi pada Layanan Sistem Keuangan dengan Pendekatan Arsitektur *Microservice*

KESIMPULAN DAN SARAN

Pada penelitian ini telah dijabarkan sebuah desain arsitektur untuk sistem terfederasi dengan pendekatan *microservice architecture*. Pendekatan desain sistem terfederasi telah berhasil diimplementasi dengan mempertimbangkan kebutuhan serta kewenangan masing-masing *stakeholder*. Teknologi *web service* berbasis REST-API digunakan untuk menjamin interoperabilitas data dan informasi pada masing-masing sistem. Sebuah desain *service integrator and management system* juga diusulkan untuk memudahkan proses pengarsipan *service* yang dimiliki, sehingga desain sistem terfederasi ini menjadi *scalable* untuk pengembangan selanjutnya.

DAFTAR PUSTAKA

- [1] S. Ravaal, *Decentralize Application*, 1st ed., O'Riley, 2016. [E-book] Available: Safari e-book.
- [2] Smihima, *Use of cloud computing services in enterprises, by purpose, 2014, 2016 and 2018 (% of enterprises using the cloud)*, Eurostats Statistics Explained, Dec 2018. [Online]. Available: <http://www.ec.europa.eu>. [Accessed: Apr. 13, 2020].
- [3] B. George, H. Hinton, and A. Nadalin. "Local architecture for federated heterogeneous system." U.S. Patent Application 10/334,273, filed July 1, 2004.
- [4] Afsarmanesh, Hamideh, and Luis M. Camarinha-Matos, "Federated Information Management for Cooperative Virtual Organizations", In *International Conference on Database and Expert Systems Applications*, pp. 561-572. Springer, Berlin, Heidelberg, 1997.
- [5] B. David, M. Klems, S. Tai, and Michael Menzel, "Metastorage: A federated cloud storage system to manage consistency-latency tradeoffs", In *2011 IEEE 4th International Conference on Cloud Computing*, pp. 452-459. IEEE, 2011.
- [6] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, "Dynamo: amazon's highly available key-value store," in *Proc. SOSP*, 2007.
- [7] R.T. Fielding, R. N. Taylor, J.R. Erenkrantz, M. M. Gorlick, J.Whitehead, R. Khare, and P. Oreizy, "Reflections on the REST architectural style and" principled design of the modern web architecture"(impact paper award)", In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, pp. 4-14. 2017.

- [8] S. Allamaraju, *RESTful Web Services Cookbook*, 1st ed., O’Riley, 2010. [E-book] Available: Safari e-book.
- [9] “What is REST”, Available: <https://restfulapi.net> [Accessed: Apr. 13, 2020].
- [10] D. M. Buede, *The Engineering Design of Systems Models and Methods*, 2nd ed., John Wiley and Sons, 2009.
- [11] F. Daniel and M. Matera, *Mashups, Data-Centric Systems and Applications*, DOI 10.1007/978-3-642-55049-2_2, Springer-Verlag Berlin Heidelberg, 2014
- [12] RT. Fielding, *Architectural Styles and the Design of Network-based Software Architectures*, University of California, 2000 Available: <https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm> [Accessed: Apr. 14, 2020]
- [13] I. Nadareishvili, R. Mitra, M. Mc.Larty, and M. Amundsen, *Microservice Architecture*, 1st ed., O’Riley, 2016. [E-book] Available: Safari e-book