
Penerapan Service-Oriented Architecture Pada Sistem Informasi Ekspedisi

Tony Wijaya

Sekolah Tinggi Manajemen Informatika dan Komputer Pontianak;
Jalan Merdeka Barat No. 372 Pontianak, Kalimantan Barat,
Telp (0561) 735555, Fax (0561) 737777
Program Studi Sistem Informasi
e-mail: mail.tonywijaya@gmail.com

Abstrak

Service-Oriented Architecture (SOA) merupakan arsitektur dalam pengembangan sistem informasi yang berorientasi pada service. Service dirancang untuk dieksekusi di sisi server untuk melayani semua client. Tidak hanya melayani, aplikasi yang berorientasi service juga wajib memonitor aktivitas-aktivitas client. Selain untuk alasan keamanan, proses monitoring juga dapat dimanfaatkan untuk menerapkan aplikasi messaging / chatting. Informasi yang diperoleh dari proses monitoring dapat digunakan untuk mengidentifikasi siapa pengirim dan penerima pesan. PT. Mitra Laut memiliki beberapa kantor di beberapa kota yang berbeda karena melayani pengiriman barang menggunakan kapal laut. Semua kantor selalu berkomunikasi untuk memastikan aktivitas operasional berjalan lancar. PT. Mitra Laut dapat memanfaatkan SOA untuk mengembangkan sistem informasi ekspedisi yang memiliki fasilitas messaging internal yang efektif.

Kata kunci: Service-Oriented Architecture, SOA

Abstract

Service-Oriented Architecture (SOA) is an architecture in information system development based on service. Service is designed to run on server-side to server all clients. Not just serving, service-oriented applications should also monitor client's activities. Despite for security reason, monitoring can also be used to implement a messaging / chatting application. Information gathered from monitoring process can be used to identify sender and receiver of the message. PT. Mitra Laut has several offices in different cities because it delivers packages using ships. All offices always communicating to ensure stability in operational activities. PT. Mitra Laut can make use of SOA to develop an expedition information system which has effective internal messaging facility.

Keywords: Service-Oriented Architecture, SOA

1. Pendahuluan

PT. Mitra Laut sebagai salah satu perusahaan yang bergerak di bidang ekspedisi harus dapat mengelola proses operasional dengan baik. Proses mengirim sebuah kontainer dari sebuah kota ke kota lain, mulai dari administrasi hingga buruh pelabuhan bukanlah sebuah proses yang sederhana. Kesalahpahaman antar kantor-kantor yang berlokasi di kota yang berbeda akan berakibat inefisiensi dan penurunan kualitas pelayanan kepada konsumen. Hal ini terjadi karena PT. Mitra Laut saat ini masih mengandalkan fasilitas-fasilitas komunikasi berupa telepon langsung, pengiriman dokumen fisik via pos, dan pengiriman dokumen *excel* via e-mail. Fasilitas-fasilitas komunikasi yang digunakan memang memungkinkan pertukaran informasi antar kantor di kota yang berbeda, tetapi belum efektif karena informasi tersebut memerlukan waktu yang cukup lama untuk sampai pada penerima informasi. Hal tersebut secara langsung dapat memperlambat proses operasional perusahaan sehingga berdampak pula pada pelayanan kepada konsumen.

Service-Oriented Architecture (SOA) merupakan arsitektur dalam mengembangkan sistem informasi yang berorientasi pada *service*. SOA menawarkan arsitektur dengan proses bisnis yang tersentralisasi. Hal ini berarti PT Mitra Laut yang memiliki cabang di beberapa kantor di Indonesia tidak mengolah transaksi bisnis secara terpisah, melainkan terpusat di satu server di kantor pusat.

Semua transaksi bisnis di setiap kantor dikirim dan diolah di komputer pusat. Dengan demikian data tersebut diolah menjadi informasi dan dapat diakses oleh semua cabang pula. Kantor cabang di Pontianak dapat mengakses secara langsung informasi kontainer yang dikirim dari Jakarta tanpa perlu menelepon ke kantor pusat di Jakarta. Akses informasi menjadi lebih mudah, lebih cepat. Frekuensi penggunaan telepon pun berkurang. Hal ini berarti SOA dapat meningkatkan efisiensi dan efektivitas dalam mengakses informasi.

Proses bisnis yang tersentralisasi sangat memudahkan perawatan dan pengembangan sistem informasi. Semua client tidak memiliki hak untuk mengakses langsung ke basis data server sehingga keamanan data pun lebih terjamin. SOA-lah yang menjadi *middleware* bagi semua client. Semua aktivitas client pun dipantau dan dicatat ke dalam log oleh server. Hal ini sangat penting dilakukan dalam mengelola keamanan data karena banyaknya client yang tersebar di beberapa kota.

Karena kemampuan SOA yang dapat mengidentifikasi client yang mengakses *service*, maka dapat pula dikembangkan sebuah aplikasi *messaging / chatting* antar client di seluruh kantor baik pusat maupun cabang. *Messaging* sangat cocok digunakan untuk mengirim informasi yang bersifat data, seperti *no invoice*, tanggal kirim kontainer, nomor kontainer, dan lain-lain. Bahkan dengan SOA, *messaging* dapat digunakan untuk mengirim *query* data transaksi ataupun laporan karena aplikasi *messaging* terintegrasi dengan proses bisnis PT. Mitra Laut menjadi satu paket sistem informasi yang komplit.

Dari uraian-uraian di atas, maka dapat disimpulkan bahwa SOA dapat memberikan solusi yang baik kepada PT. Mitra Laut dalam mengelola data menjadi informasi sekaligus meningkatkan efisiensi dan efektivitas dalam berkomunikasi guna memberikan pelayanan yang optimal kepada konsumen.

2. Metode Penelitian

Inti dari SOA adalah *service*. Inti dari *service* adalah komponen. Komponen berisi fungsi-fungsi yang merupakan perwujudan dari proses-proses bisnis PT. Mitra Laut. Penulis menyimpulkan proses-proses bisnis perusahaan tersebut berdasarkan hasil wawancara secara langsung dengan manager di Pontianak. Langkah selanjutnya yang penulis lakukan dalam metode penelitian ini adalah:

- a. Menguraikan komponen-komponen yang diperlukan serta fungsi-fungsi yang terdapat di dalam komponen-komponen tersebut, di antaranya:

Tabel 1 Daftar Fungsi-fungsi Utama dalam Proses Bisnis PT. Mitra Laut

| Nama Fungsi | Deskripsi |
|---------------------------|-----------------------------------|
| Kontainer.Tambah() | Menambah record kontainer |
| Kontainer.Ubah() | Mengubah record kontainer |
| Kontainer.Hapus() | Menghapus record kontainer |
| Kontainer.Tampil() | Menampilkan daftar kontainer |
| Pelanggan.Tambah() | Menambah record pelanggan |
| Pelanggan.Ubah() | Mengubah record pelanggan |
| Pelanggan.Hapus() | Menghapus record pelanggan |
| Pelanggan.Tampil() | Menampilkan data pelanggan |
| Kota.Tambah() | Menambah record kota |
| Kota.Ubah() | Mengubah record kota |
| Kota.Hapus() | Menghapus record kota |
| Kota.Tampil() | Menampilkan data kota |
| Ukuran Kontainer.Tambah() | Menambah record ukuran kontainer |
| Ukuran Kontainer.Ubah() | Mengubah record ukuran kontainer |
| Ukuran Kontainer.Hapus() | Menghapus record ukuran kontainer |
| Ukuran Kontainer.Tampil() | Menampilkan data ukuran kontainer |
| Kade.Tambah() | Menambah record kade |
| Kade.Ubah() | Mengubah record kade |
| Kade.Hapus() | Menghapus record kade |

| | |
|-----------------------------|---------------------------------------|
| Kade.Tampil() | Menampilkan daftar kade |
| CYC.Tambah() | Menambah record biaya CYC |
| CYC.Ubah() | Mengubah record biaya CYC |
| CYC.Hapus() | Menghapus record biaya CYC |
| CYC.Tampil() | Menampilkan biaya CYC |
| Opt.Tambah() | Menambah record biaya Opt |
| Opt.Ubah() | Mengubah record biaya Opt |
| Opt.Hapus() | Menghapus record biaya Opt |
| Opt.Tampil() | Menampilkan biaya Opt |
| Arrival.Tambah() | Menambah record kedatangan kapal |
| Arrival.Ubah() | Mengubah record kedatangan kapal |
| Arrival.Hapus() | Menghapus record kedatangan kapal |
| Arrival.Tampil() | Menampilkan kedatangan kapal |
| KVT.Tambah() | Menambah record kapal-voy-tahun |
| KVT.Ubah() | Mengubah record kapal-voy-tahun |
| KVT.Hapus() | Menghapus record kapal-voy-tahun |
| KVT.Tampil() | Menampilkan daftar kapal-voy-tahun |
| Penumpukan. Tambah() | Menambah record biaya penumpukan |
| Penumpukan. Ubah() | Mengubah record biaya penumpukan |
| Penumpukan. Hapus() | Menghapus record biaya penumpukan |
| Penumpukan. Tampil() | Menampilkan data biaya penumpukan |
| Tagihan Penumpukan.Tambah() | Menambah record tagihan penumpukan |
| Tagihan Penumpukan.Ubah() | Mengubah record tagihan penumpukan |
| Tagihan Penumpukan.Hapus() | Menghapus record tagihan penumpukan |
| Tagihan Penumpukan.Tampil() | Menampilkan daftar tagihan penumpukan |
| Biaya Kapal.Tambah() | Menambah record biaya kapal |
| Biaya Kapal.Ubah() | Mengubah record biaya kapal |
| Biaya Kapal.Hapus() | Menghapus record biaya kapal |
| Biaya Kapal.Tampil() | Menampilkan daftar biaya kapal |
| Tipe CYC.Tambah() | Menambah record tipe CYC |
| Tipe CYC.Ubah() | Mengubah record tipe CYC |
| Tipe CYC.Hapus() | Menghapus record tipe CYC |
| Tipe CYC.Tampil() | Menampilkan daftar tipe CYC |
| Volume.Tambah() | Menambah record volume barang |
| Volume.Ubah() | Mengubah record volume barang |
| Volume.Hapus() | Menghapus record volume barang |
| Volume.Tampil() | Menampilkan daftar volume barang |
| Kas.Tambah() | Menambah record transaksi kas |
| Kas.Ubah() | Mengubah record transaksi kas |
| Kas.Hapus() | Menghapus record transaksi kas |
| Kas.Tampil() | Menampilkan daftar transaksi kas |
| SaldoKas.Tambah() | Menambah record saldo kas |
| SaldoKas.Ubah() | Mengubah record saldo kas |
| SaldoKas.Hapus() | Menghapus record saldo kas |
| SaldoKas.Tampil() | Menampilkan daftar saldo kas |
| PiutangKaryawan.Tambah() | Menambah record piutang karyawan |
| PiutangKaryawan.Ubah() | Mengubah record piutang karyawan |
| PiutangKaryawan.Hapus() | Menghapus record piutang karyawan |

| | |
|--------------------------|-------------------------------------|
| PiutangKaryawan.Tampil() | Menampilkan record piutang karyawan |
| Beban.Tambah() | Menambah record beban usaha |
| Beban.Ubah() | Mengubah record beban usaha |
| Beban.Hapus() | Menghapus record beban usaha |
| Beban.Tampil() | Menampilkan laporan beban usaha |

- b. Menguraikan secara detil fungsi-fungsi yang telah disebutkan pada poin (a).
- c. Konversi fungsi-fungsi yang spesifik di dalam komponen-komponen menjadi *service*.
- d. Mengintegrasikan *service-service* yang ada menjadi *Web Service* dan diimplementasikan secara sentralistik pada komputer server PT. Mitra Laut.
- e. Pengujian dengan mengakses basis data dari *client* ke *server* melalui perantara SOA.

Tahapan-tahapan yang diperlukan untuk menerapkan SOA pada sistem informasi akademik:

a. Service

Fungsi-fungsi yang telah disebutkan pada analisa kebutuhan di atas selanjutnya siap untuk dikonversi menjadi *service*. *Service* menggunakan XML dalam pertukaran data sehingga dapat diakses oleh semua bahasa pemrograman. Hal ini dikarenakan XML adalah sebuah standar yang dimengerti oleh semua bahasa pemrograman. Sebuah *service* pada dasarnya adalah fungsi yang diberi *interface*. Tanpa *interface*, sebuah fungsi hanya dapat diakses oleh bahasa program yang sama dengan bahasa yang digunakan untuk menulis fungsi tersebut.

Siklus hidup sebuah *service* ada 4 tahap: *Uninstalled, Shutdown, Stopped, Started*. *Uninstalled* berarti *service* belum diinstall atau sudah dibuang pada/dari sebuah komputer. *Shutdown* berarti *service* sudah dimatikan dan semua *resource* dihancurkan dari memori komputer. *Stopped* berarti *service* dihentikan tetapi masih aktif di dalam memori sehingga dapat diaktifkan kembali dalam waktu singkat. *Started* berarti keadaan di mana *service* sedang aktif dan melayani eksekusi *service* oleh *client-client*.

Layaknya sebuah sistem operasi server, sebuah *service* dirancang untuk selalu aktif sepanjang waktu (24 jam). Hal ini disebabkan SOA melayani transaksi perusahaan skala besar sehingga ketidakterediaan *service* akan menurunkan kestabilan transaksi perusahaan. Penghentian *service* dilakukan apabila memang dibutuhkan, misalnya pada saat komputer memerlukan restart karena update tertentu.

b. Registry

Langkah kedua ialah mendaftarkan *service-service* yang telah dibuat ke dalam *registry*. Hal ini dilakukan supaya *service consumer* atau *client* dapat menemukan *service* yang mereka perlukan di dalam *registry* tersebut. Jadi *registry* ini adalah ibarat sebuah katalog buku di dalam perpustakaan.

c. Messaging

Tahap ketiga adalah menyiapkan jaringan yang cukup handal untuk menerapkan sistem informasi yang berorientasi pada *service*. Jaringan ini biasa disebut *Enterprise Service Bus (ESB)* yang akan digunakan untuk pengiriman dan penerimaan data XML. Sebuah ESB mampu menjamin transmisi data untuk perusahaan berskala nasional maupun internasional.

ESB yang sudah dibangun dapat digunakan pula untuk membuat sebuah aplikasi messaging yang digunakan untuk pertukaran pesan antar *client*. Layaknya aplikasi messaging seperti *Blackberry Messenger, WhatsApp*, dan lain-lain, aplikasi ini bertugas mengirim dan menerima pesan teks maupun gambar. Bedanya adalah aplikasi ini terintegrasi dengan sistem informasi ekspedisi milik PT. Mitra Laut. Aplikasi messaging yang terintegrasi tentunya memiliki keuntungan tersendiri, misalnya dapat mengirimkan *query* data antar *client*. *Client* di Jakarta pusat dapat mengirimkan *query* biaya kapal kepada *client* di Pontianak hanya dengan sebuah link di layar monitor. Fitur messaging ini juga dapat digunakan untuk melihat *client* mana yang sudah aktif (sudah login) ke dalam SOA. Setiap *client* selalu mengirim sinyal ke sever setiap 5 menit. Apabila dalam 5 menit tidak ada sinyal dari *client* A, maka *client* A dinyatakan offline.

d. Management

Ini merupakan tahapan manajerial terhadap SOA. Pengaturan *Web Service* sekuritas dan data dilakukan secara sentralistik pada tahap ini. Manajemen SOA harus dapat mengelola ESB dengan baik. Tahapan ini layaknya sebuah tahapan manajemen jaringan komputer. Di dalamnya seorang manajer jaringan harus mampu mengorganisasi hardware dan software jaringan, mengkonfigurasi jaringan sedemikian rupa sehingga memenuhi kebutuhan pemakai, memonitor performa jaringan, mencegah terjadinya gangguan, mendeteksi gangguan, mengkoreksi error pada jaringan, memberikan dukungan

kepada pemakai yang mengalami masalah, memberikan pelatihan dasar jaringan kepada pemakai, mengelola biaya jaringan yang meliputi pengadaan, perawatan, konfigurasi, dan pelatihan.

e. Orchestration

Tahap ini berfungsi untuk menjembatani SOA PT. Mitra Laut dengan aplikasi client. Pada tahapan ini dilakukan coding untuk mengkonsumsi *service* yang sudah dibuat dan dijalankan di komputer server pusat. Aplikasi client dapat dikembangkan dengan berbagai macam bahasa pemrograman, karena SOA menggunakan teknologi web service yang didasarkan pada standar terbuka (open standar) yaitu SOAP, XML, WSDL, dan UDDI. Keempat standar ini didukung oleh semua vendor seperti Microsoft, Oracle, Apple, Google yang menciptakan bahasa pemrograman dan basis data.

f. Analytics

Pada tahap ini penulis menyediakan fasilitas *monitoring* terhadap *event-event* yang terjadi pada SOA. *Event* yang dimaksud adalah pemanggilan *service* oleh *client*, waktu akses, identitas dan lokasi *client* yang memanggil *service* tersebut.

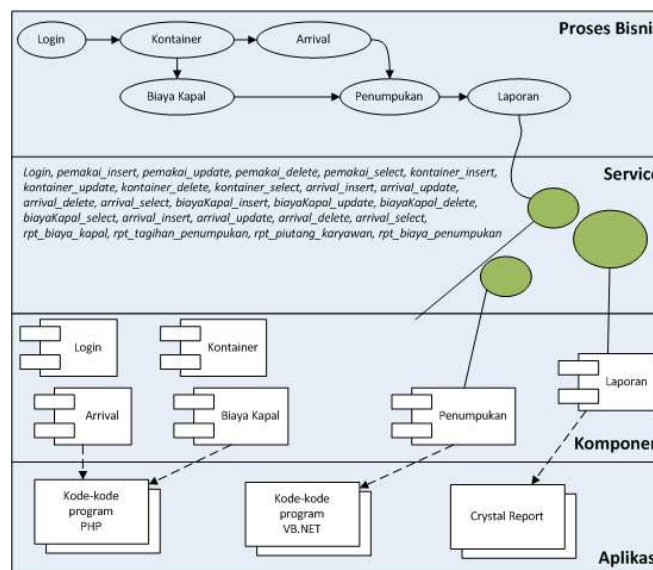
g. User Interface

Tahapan ini merupakan pengujian terhadap SOA melalui *User Interface* di sisi *client*. Sisi *client* hanya berisi tampilan interaksi dengan pemakai. Tidak ada proses bisnis karena semuanya telah disimpan di sisi *server*. Tahap ini membahas pemilihan user interface yang dibutuhkan oleh *client*. Beberapa pilihan yang biasa digunakan adalah desktop, web-based, dan mobile. Model desktop digunakan apabila pemakai butuh mengakses *resource* komputer secara penuh misalnya segala merk dan jenis printer (baik dot matrix, inkjet, maupun laser jet), membuat tampilan model multiple window, hardware-hardware khusus seperti alat pacu jantung, barcode reader, robotik, multi-monitor, dan lain-lain. Model web-based digunakan untuk menyerderhanakan proses implementasi karena hanya dibutuhkan sebuah web browser untuk mengaksesnya, tidak membutuhkan proses instalasi apapun pada sisi *client*. Model mobile digunakan apabila target dari aplikasi adalah pengguna smartphone.

3. Hasil dan Analisis

Proses penerapan *Service-Oriented Architecture* (SOA) dalam membangun sistem informasi ekspedisi dilakukan dengan langkah-langkah sebagai berikut:

Penerapan SOA pada pengembangan sistem informasi ekspedisi dilakukan berdasarkan *framework* standar SOA yaitu berupa komponen-komponen seperti: *Service*, *Registry*, *Messaging*, *Management*, *Orchestration*, *Analytics*, dan *User Interface*. Adapun implementasi *framework* SOA pada penelitian ini menggunakan teknologi *Windows Communication Foundation* (WCF) dari *Microsoft* yang merupakan bagian dari *.NET Framework 4.0*.

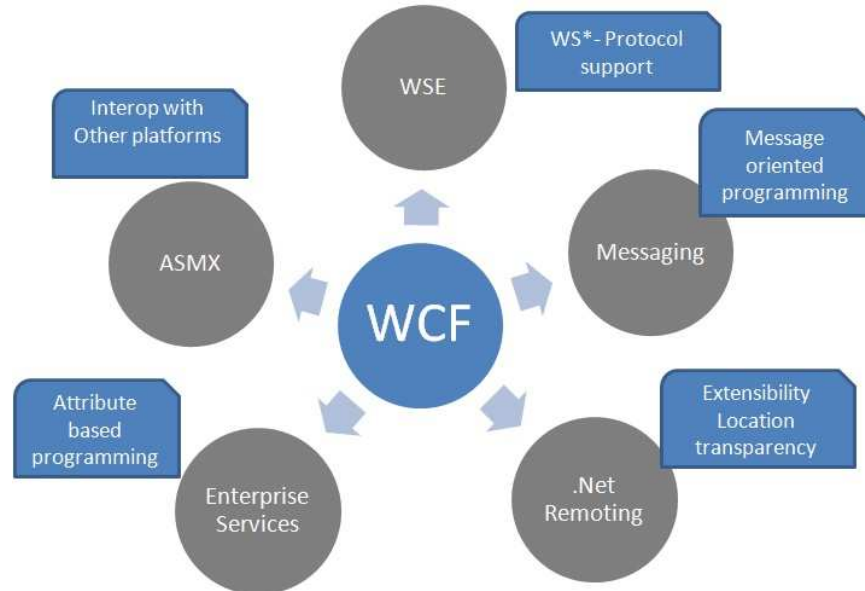


Gambar 1 SOA dan Proses Bisnis PT. Mitra Laut

Komponen utama dari SOA adalah *service*, yang merupakan kumpulan fungsi-fungsi yang mempunyai *interface* standar sehingga dapat diakses oleh berbagai bahasa pemrograman standar, seperti

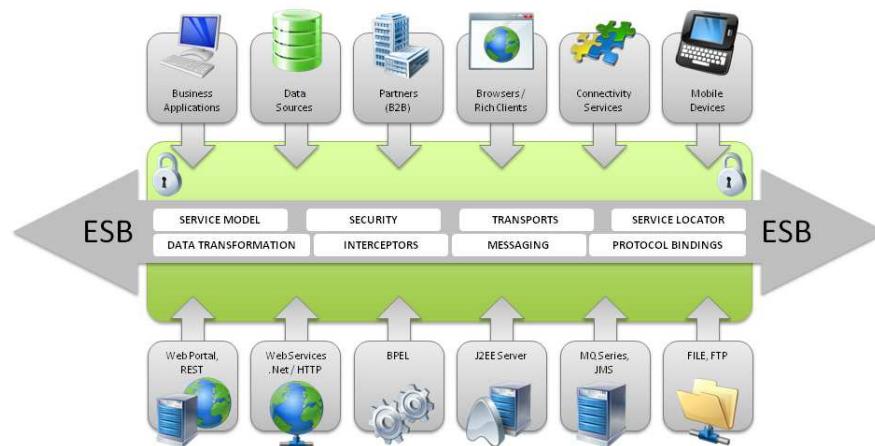
VB.NET, C#, PHP, dan lain-lain. Pada penelitian ini, penulis menerjemahkan proses-proses bisnis PT. Mitra Laut ke dalam *service* menggunakan bahasa pemrograman C# yang juga merupakan bagian dari *.NET Framework 4.0*. Kode-kode program tersebut kemudian di-*compile* menjadi komponen berupa *Dynamic Link Library (DLL)*. Komponen ini kemudian diberi *interface* oleh WCF sehingga menjadi *service*. Jadi dengan *service*, sangat membantu penulis dalam proses pengembangan sistem informasi sekolah supaya dapat diselesaikan dalam waktu yang relatif cepat dan efisien.

WCF menyediakan semacam pustaka bagi *client* yang ingin mengakses *service-service* yang dimiliki dengan *registry*. Dengan fasilitas ini, *developer* di sisi *client* dapat mengetahui nama *service*, parameter-parameter masukan yang diperlukan, dan proses yang dimiliki oleh *service* yang bersangkutan.



Gambar 2 Fitur-fitur *Windows Communication Foundation (WCF)* dari *Microsoft*

Implementasi SOA membutuhkan jaringan komputer yang memadai karena merupakan sebuah *framework* skala *enterprise*. Jaringan komputer yang dibutuhkan oleh standar SOA disebut *Enterprise Service Bus (ESB)*. ESB merupakan infrastruktur gabungan berupa jaringan komputer skala besar (*enterprise*) yang dapat menghubungkan semua workstation dalam sebuah organisasi, baik skala nasional maupun internasional. Sebuah *service* tanpa ESB hanyalah sebuah *service* biasa. Dengan dukungan ESB maka *service* tersebut dapat disebut SOA karena SOA selalu berbicara skala *enterprise*. Sebuah ESB bertanggung jawab untuk memastikan bahwa *service* tersedia secara global. Ini berarti dapat dengan mudah ditemukan, dikoneksi, dan dikonsumsi tanpa dipengaruhi oleh jenis teknologi yang digunakan. Dengan mudahnya diakses, maka ESB dapat membuat proses-proses bisnis digunakan kembali (*reuse*) dan menyederhanakan komposisi komponen di sisi server. Pada akhirnya, ESB juga wajib menyediakan kemampuan atau fitur untuk mengelola (*manage*), memantau (*monitoring*), *routing*, interaksi, dan transformasi data XML menjadi berbagai variasi penyampaian informasi sesuai dengan kebutuhan masing-masing *client*.



Gambar 3 Enterprise Service Bus (ESB)

Tata kelola SOA bersifat sentralisasi, yaitu di komputer *server*. Adapun tata kelola yang dimaksud di sini adalah pengelolaan *web service*, basis data, dan keamanan data. Ketiga pengelolaan ini merupakan tulang punggung atau *backbone* yang menopang SOA pada sistem informasi ekspedisi. WCF mengimplementasikan standar *web service* untuk mendistribusikan *service* yang dimiliki sehingga bersifat global dan dapat diakses baik oleh komputer, laptop, maupun *mobile device*. Oleh karena bersifat global, keamanan data perlu diperhatikan dengan baik. Keamanan yang diterapkan pada sistem informasi akademik berupa autentikasi *user id*, alamat *IP*, dan nama *client* yang mengakses. Di dalam SOA, tidak ada *client* yang dapat langsung mengakses ke basis data karena basis data hanya boleh diakses oleh *service* itu sendiri.

Untuk pengembangan ke depan, SOA dapat dihubungkan dengan aplikasi lain seperti website bagi pelanggan yang memakai jasa dari PT. Mitra Laut untuk mengirim barang. Website ini dapat dikembangkan supaya dapat mengakses posisi kontainer, biaya pengiriman, tanggal tiba di pelabuhan, berapa lama penumpukan di pelabuhan, dan lain-lain. Hal ini merupakan salah satu komponen dari SOA yaitu *Orchestration*.

Untuk pengelola SOA, yaitu admin, dapat memantau aktivitas-aktivitas atau *event-event* SOA melalui *analytics* SOA. Pemantauan yang dimaksud di sini adalah nama *service* yang diakses, waktu akses, nama *client* yang mengakses, dan *user id* yang mengakses.

User interface (UI) pada sistem informasi ekspedisi dibuat dengan dua jenis yaitu berupa *form desktop* dan halaman *web*. *Form desktop* digunakan oleh kasir untuk memanipulasi data karena bersifat *rich control* yang memudahkan untuk berinteraksi dengan multi dokumen, printer-printer dot matrix untuk mencetak struk, dan keamanan data yang lebih terjamin. Sedangkan manager menggunakan aplikasi berbasis web karena kebutuhan mereka lebih bersifat pelaporan atau menampilkan data rekapitulasi dan jarang berhubungan dengan hardware-hardware khusus sehingga format HTML akan lebih efektif dan efisien dalam menyampaikan informasi

4. KESIMPULAN

Setelah penulis melakukan penelitian mengenai penerapan *Service-Oriented Architecture* (SOA) pada pengembangan sistem informasi ekspedisi PT. Mitra Laut, maka kesimpulan yang dapat penulis ambil adalah sebagai berikut:

- Penerapan SOA dalam mengembangkan Sistem Informasi Ekspedisi PT. Mitra Laut memberikan solusi dalam mengatasi permasalahan jarak geografis antar kantor, di mana terdapat kesulitan dalam pertukaran informasi antar cabang yang masih mengandalkan fasilitas telepon, pengiriman dokumen via pos, dan pengiriman dokumen *excel* via e-mail.
- SOA menggunakan arsitektur yang bersifat sentralisasi sehingga pertukaran informasi antar cabang dapat dilakukan secara *real-time* kapan pun dibutuhkan oleh penerima informasi. Hal ini dapat meningkatkan efektivitas komunikasi antar kantor cabang PT. Mitra Laut.
- Arsitektur SOA yang tersentralisasi dan memiliki fitur *analytics* dapat dimanfaatkan untuk mengembangkan aplikasi *messaging* yang juga dapat meningkatkan efektivitas maupun efisiensi komunikasi antar kantor cabang PT. Mitra Laut.

- d. Penerapan SOA pada pengembangan Sistem Informasi Ekspedisi dengan menggunakan teknologi *NET Framework 4.0* yang mencakup *Windows Communication Foundation (WCF)* dan bahasa pemrograman *C#* menghasilkan sistem informasi ekspedisi PT Mitra Laut yang berorientasi pada *service*.
- e. Perancangan UI menggunakan *Microsoft Visual Studio 2013*. Sedangkan data provider *MySQL* dapat menggunakan RDBMS berupa *MySQL Workbench*.
- f. *Service* yang telah dibuat merupakan komponen yang diberi *interface* standar (XML) melalui fasilitas dari WCF. Komponen itu sendiri ditulis dengan bahasa pemrograman *C#*.
- g. Dengan penerapan SOA, memungkinkan *client* tidak langsung mengakses kepada basis data sehingga keamanan data lebih terjamin, stabilitas lebih terjaga, integritas data lebih terjamin.

5. SARAN

Penulis menyadari bahwa penelitian ini masih belum sempurna. Oleh karena itu penulis ingin menyampaikan saran untuk penelitian selanjutnya supaya:

- a. Sistem Informasi Ekspedisi PT. Mitra Laut yang berbasis *service* (SOA) masih dapat dikembangkan dengan menambahkan fitur-fitur yang dapat meningkatkan efisiensi dan efektivitas operasional perusahaan.
- b. *Platform* mobile dapat menjadi titik pengembangan selanjutnya, mengingat perkembangan teknologi *smartphone* yang sedang *booming*. Hal ini dikarenakan SOA juga dapat diakses oleh perangkat *mobile* seperti *smartphone*.
- c. Faktor keamanan dapat ditingkatkan lagi dengan membahasnya secara lebih detail.
- d. *Enterprise Service Bus* juga merupakan salah satu bagian yang dapat dibahas lebih rinci karena merupakan *backbone* dari SOA PT. Mitra Laut yang memiliki beberapa kantor cabang di beberapa kota yang berbeda.

UCAPAN TERIMA KASIH

Penulis mengambil kesempatan ini untuk berterima kasih kepada:

- a. Direktur PT. Mitra Laut yang telah mendukung penelitian ini dengan menyetujui pendanaan kepada penelitian ini.
- b. Bapak Sandy Kosasi, S.E., M.M., M.Kom. selaku Ketua STMIK Pontianak yang telah memberi masukan mengenai konsep SOA, mendanai penelitian ini dan melakukan pengadaan infrastruktur yang dibutuhkan dalam penelitian ini.
- c. Staf PT. Mitra Laut Pontianak seperti inkaso, admin, dan manager operasional yang membantu menjelaskan proses bisnis PT. Mitra Laut.

DAFTAR PUSTAKA

- [1]. Bertino, E., Martino, L., Paci, F., & Squicciarini, A. (2010). *Security For Web Services And Service-Oriented Architecture*. Berlin: Springer.
- [2]. Erl, T. (2008). *SOA Principles of Service Design*. Prentice Hall.
- [3]. Hafner, M., & Breu, R. (2009). *Security Engineering for Service-Oriented Architecture*. Berlin: Springer.
- [4]. Lam, W. (2007). *Enterprise Architecture and Integration: Methods, Implementation, and Technologies*. IGI Global.
- [5]. Lawler, J. P., & Howell-Barber, H. (2008). *Service-Oriented Architecture: SOA Strategy, Methodology, and Technology*. New York: Auerbach Publications.
- [6]. Mehta, M. R., Lee, S., & Shah, J. R. (2006). *Service-Oriented Architecture. Concepts and Implementation*, 3.
- [7]. Minoli, D. (2008). *Enterprise Architecture A to Z*. CRC Press.
- [8]. Schelp, J., & Aier, S. (2009). *SOA and EA. Sustainable Contributions for Increasing Corporate Agility*, 8.
- [9]. Schmutz, G., Welkenbach, P., & Liebhart, D. (2010). *Service-Oriented Architecture: An Integration Blueprint*. Birmingham: Packt Publishing.
- [10]. White, L., Wilde, N., Reichherzer, T., El-Sheikh, E., Goehring, G., Baskin, A., et al. (2012). *Understanding Interoperable Systems. Challenges for the Maintenance of SOA Applications*.