



Analisis Perbandingan Strategi *Reuse Service* dalam Implementasi *Service-Oriented Architecture* (SOA) untuk Efisiensi Proyek Teknologi Informasi

Andika Noor Ismawan¹✉, Bagus Rosyid Hamdani¹, Sahri Ramadan¹, Unique Scovi Egnel¹, Theodora Maria Putri Komul¹

⁽¹⁾Teknik Informatika Universitas Esa Unggul, Bekasi

DOI: 10.31004/jutin.v8i3.46348

✉ Corresponding author:
[noorismawanandika@gmail.com]

Article Info	Abstrak
<p><i>Kata kunci:</i> <i>Reuse Service;</i> <i>Service-Oriented</i> <i>Architecture;</i> <i>Efisiensi Pengembangan</i> <i>Sistem;</i></p>	<p>Perkembangan teknologi digital yang pesat mendorong kebutuhan akan strategi pengembangan sistem yang efisien dan berkelanjutan. Penelitian ini bertujuan untuk melakukan analisis komparatif terhadap empat pendekatan reuse service dalam arsitektur <i>Service-Oriented Architecture</i> (SOA), yaitu Standardisasi API & Service, Arsitektur <i>Microservices</i>, Pengembangan Berbasis <i>API-First</i>, dan <i>Enterprise Service Bus</i> (ESB). Metode yang digunakan berupa studi pustaka terhadap berbagai sumber ilmiah terkini serta evaluasi terhadap efektivitas masing-masing pendekatan ditinjau dari aspek efisiensi waktu, biaya, dan pemanfaatan sumber daya. Hasil kajian menunjukkan bahwa setiap strategi memiliki karakteristik dan keunggulan tersendiri, namun pendekatan <i>API-First</i> dianggap paling fleksibel dalam mendukung proses integrasi layanan dan pengembangan sistem yang bersifat modular. Berdasarkan temuan tersebut, disarankan pemilihan strategi disesuaikan dengan kebutuhan organisasi dan kompleksitas sistem yang dikembangkan.</p>

Abstract

Keywords:
Reuse Service;
Service-Oriented
Architecture;
System Development
Efficiency

The rapid advancement of digital technology necessitates the adoption of efficient and sustainable system development strategies. This study aims to conduct a comparative analysis of four reuse service approaches within the framework of Service-Oriented Architecture (SOA), namely API & Service Standardization, Microservices Architecture, API-First Development, and Enterprise Service Bus (ESB). The research method employed is a literature review of recent scholarly sources, along with an evaluation of the effectiveness of each approach in terms of time efficiency, cost, and resource utilization. The findings indicate that each strategy possesses distinct characteristics and contextual advantages, with the API-First approach deemed the most flexible in supporting service integration and modular

system development. Based on these findings, it is recommended that the selection of reuse strategies be aligned with organizational needs and the complexity of the systems being developed.

1. PENDAHULUAN

Perkembangan teknologi informasi yang sangat cepat telah mendorong organisasi untuk mengadopsi pendekatan arsitektur sistem yang adaptif, efisien, dan siap menghadapi perubahan. Dalam konteks pengembangan sistem skala menengah hingga besar, pendekatan Service-Oriented Architecture (SOA) menjadi salah satu solusi yang banyak digunakan karena mampu membentuk sistem yang modular dan dapat digunakan kembali (reusable) untuk berbagai kebutuhan (Niknejad et al., 2020).

Salah satu prinsip utama dalam implementasi SOA adalah reuse service, yaitu penggunaan kembali layanan yang telah dirancang agar dapat dimanfaatkan di berbagai aplikasi atau domain bisnis tanpa perlu melakukan modifikasi besar. Pendekatan ini tidak hanya mendorong efisiensi dari sisi waktu dan biaya pengembangan, tetapi juga meningkatkan konsistensi antar layanan dan mempermudah proses integrasi sistem yang kompleks (Desina, 2020). Namun demikian, strategi reuse yang diterapkan harus dirancang dengan hati-hati, sebab tanpa perencanaan yang matang, reuse justru berpotensi menimbulkan tumpang tindih fungsi layanan dan hambatan dalam pengelolaan siklus hidup sistem.

Beberapa penelitian terdahulu telah membahas efektivitas dari pendekatan reuse service dalam SOA. Misalnya, Muslih & Hasanah, (2019) meneliti penerapan SOA untuk monitoring layanan beasiswa dan menemukan bahwa layanan berbasis reusability meningkatkan fleksibilitas integrasi antar modul. Sementara itu, Hadian, Hakim, & Fanani, (2023) mengevaluasi keberhasilan SOA dalam konteks sistem informasi UMKM, dan menyimpulkan bahwa keberhasilan reuse sangat dipengaruhi oleh standar API dan pemetaan kebutuhan layanan.

Penelitian oleh Lercher et al., (2024) lebih lanjut menyoroti tantangan dalam evolusi API pada arsitektur mikroservis, terutama pada pengelolaan kompatibilitas layanan jangka panjang. Adapun Dixit, (2024) menekankan pentingnya desain API-First dalam mendukung reuse service secara berkelanjutan dalam organisasi modern berbasis digital.

Namun, meskipun banyak studi membahas strategi reuse secara individual, belum banyak yang melakukan analisis komparatif menyeluruh terhadap keempat pendekatan utama reuse service dalam SOA, yaitu: Standardisasi API & Service, Microservices Architecture, API-First Development, dan Enterprise Service Bus (ESB). Cela inilah yang menjadi landasan penelitian ini untuk memberikan kontribusi lebih lanjut.

Melalui studi pustaka dan evaluasi terhadap referensi teknis serta studi kasus dari berbagai industri, penelitian ini bertujuan untuk memberikan analisis komparatif terhadap keempat strategi reuse service tersebut. Fokus kajian diarahkan pada tiga indikator utama, yaitu efisiensi waktu pengembangan, penghematan biaya, dan optimalisasi penggunaan sumber daya. Hasil analisis ini diharapkan dapat memberikan rekomendasi strategi reuse yang paling relevan dan kontekstual, sesuai dengan kebutuhan organisasi dan tingkat kompleksitas sistem yang dibangun.

2. METODE

Penelitian ini menggunakan pendekatan deskriptif-kualitatif dengan metode studi pustaka dan analisis komparatif. Tujuan utama dari pendekatan ini adalah untuk mengevaluasi serta membandingkan empat strategi reuse service dalam konteks implementasi Service-Oriented Architecture (SOA), yaitu Standardisasi API & Service, Microservices Architecture, API-First Development, dan Enterprise Service Bus (ESB).

Jenis dan Pendekatan Penelitian

Jenis penelitian ini adalah studi kepustakaan (library research), yang mengkaji literatur ilmiah, jurnal, artikel teknis, serta dokumentasi dari studi kasus implementasi nyata di industri. Pendekatan ini dipilih karena memungkinkan peneliti untuk menganalisis secara mendalam konsep, prinsip, serta penerapan dari masing-masing strategi reuse service berdasarkan data sekunder yang tersedia secara valid dan terpercaya (Hadian et al., 2023); Rasyidi & Pratama, 2024).

Objek Penelitian

Fokus utama dalam penelitian ini adalah empat strategi reuse service yang paling umum dalam penerapan arsitektur SOA:

- Standardisasi API & Service
- Microservices Architecture
- API-First Development
- Enterprise Service Bus (ESB)

Keempat strategi ini dianalisis berdasarkan kelebihan, tantangan, dan dampaknya terhadap efisiensi proyek teknologi informasi.

Teknik Pengumpulan Data

Data dikumpulkan melalui studi literatur terhadap jurnal akademik selama 10 tahun terakhir (2015-2025), white paper, dokumentasi teknologi, serta laporan implementasi dari organisasi yang menerapkan arsitektur layanan. Beberapa jurnal diambil sebagai referensi utama yang membahas tentang efektivitas API standard (Dixit, 2024), performa ESB dalam sistem enterprise (Aziz et al., 2020), serta keunggulan microservices dalam pengembangan system terdistribusi (Lercher et al., 2024b).

Teknik Analisis Data

Analisis data dilakukan dengan langkah-langkah sebagai berikut:

- Reduksi Data: Menyaring informasi yang tidak relevan dan fokus terhadap data yang terkait dengan empat strategi reuse service dalam SOA.
- Kategorisasi: Mengelompokan data yang telah terkumpul, berdasarkan tiga indikator evaluasi, yaitu efisiensi waktu, efisiensi biaya, dan optimalisasi sumber daya.
- Perbandingan: Menyusun matriks perbandingan untuk mengidentifikasi keunggulan dan kelemahan dari setiap strategi.
- Interpretasi: Menganalisis secara mendalam untuk menentukan strategi yang paling efektif dan sesuai dengan kebutuhan proyek teknologi informasi sesuai dengan studi literatur dan studi kasus.

Indikator Evaluasi

Tabel 1. Indikator Evaluasi

Indikator	Deskripsi
Efisiensi Waktu	Kecepatan dalam pengembangan, integrasi, dan penyebaran (deployment) layanan.
Efisiensi Biaya	Penghematan biaya pengembangan awal, pemeliharaan sistem, dan integrasi layanan.
Efisiensi Sumber Daya	Optimalisasi pemanfaatan sumber daya manusia, perangkat keras, dan infrastruktur teknologi.

3. HASIL DAN PEMBAHASAN

Pemaparan Strategi Reuse Service dalam SOA

Service-Oriented Architecture (SOA) merupakan pendekatan arsitektur yang digunakan untuk membangun sistem yang modular, fleksibel, dan mudah diintegrasikan. Salah satu prinsip penting dalam SOA adalah reuse service, yaitu penggunaan kembali layanan yang sudah dikembangkan untuk berbagai kebutuhan aplikasi, proyek, atau unit bisnis. Reuse tidak hanya mempercepat proses pengembangan, tetapi juga meningkatkan efisiensi sumber daya dan konsistensi sistem (Desina, 2020).

Penelitian ini mengevaluasi empat strategi reuse service yang umum diterapkan pada SOA, yaitu:

- Standardisasi API & Service
- Microservices Architecture
- API-First Development
- Enterprise Service Bus (ESB)

Setiap strategi dianalisis sesuai dengan keunggulan, tantangan, dan relevansinya terhadap efisiensi waktu, biaya dan pemanfaatan sumber daya dalam proyek teknologi informasi.

Analisi Perbandingan Strategi

a. Standarisasi API & Service

Strategi ini berfokus pada penggunaan protokol dan antarmuka yang seragam seperti REST, SOAP, serta dokumentasi berbasis OpenAPI atau Swagger. Pendekatan ini mempercepat integrasi antar sistem dan memudahkan pengembang dalam memahami serta memanfaatkan layanan yang tersedia (Dixit, 2024). Standardisasi API juga mendukung interoperabilitas dan pengurangan redundansi layanan.

Keunggulannya berupa interoperabilitas tinggi, mudah dipelajari, mempercepat pengembangan. Sedangkan untuk kelemahannya membutuhkan komitmen dokumentasi dan konsistensi antar tim.

b. Microservices Architecture

Pendekatan ini membagi sistem menjadi layanan-layanan kecil dan independen yang dapat dikembangkan, di-deploy, dan diskalakan secara terpisah. Setiap microservice menjalankan fungsi spesifik dan berkomunikasi melalui API (Lercher et al., 2024b). Pendekatan ini sangat cocok untuk sistem kompleks yang terus berkembang.

Keunggulan strategi ini merupakan memiliki skalabilitas tinggi, fleksibel, mendukung pengembangan paralel. Sedangkan yang akan menjadi tantangan strategi ini yaitu memiliki kompleksitas integrasi, kebutuhan manajemen versi layanan yang ketat.

c. API-First Development

Dalam strategi ini, desain API dilakukan sebelum pengembangan sistem dimulai. API berperan sebagai kontrak antara berbagai tim pengembang. Strategi ini memungkinkan pengembangan paralel antara tim frontend dan backend, serta meningkatkan kemungkinan reuse layanan dalam berbagai proyek (Dixit, 2024).

Keunggulan dari penerapan strategi ini yaitu memiliki konsistensi layanan, dukungan integrasi yang cepat, dan sangat flexibel. Sedangkan kelemahannya antara lain membutuhkan alat bantu desain API dan pengujian kontrak sejak awal.

d. Enterprise Service Bus (ESB)

ESB berfungsi sebagai infrastruktur perantara yang menangani komunikasi dan transformasi pesan antar layanan. Strategi ini efektif untuk integrasi sistem heterogen, termasuk sistem legacy. ESB mampu mengatur aliran data, transformasi format, dan pengelolaan keamanan secara terpusat (Aziz et al., 2020).

Enterprise Service Bus (ESB) memiliki keunggulan, yaitu terstruktur, cocok untuk lingkungan enterprise, dan mendukung pengaturan secara otomatis dalam alur kerja bersama sistem lain.

Perbandingan Strategi Reuse Service

Berikut perbandingan dari keempat strategi berdasarkan tiga indikator evaluasi utama:

Tabel 2. Perbandingan Strategi Reuse Service

Strategi	Efisiensi Waktu	Efisiensi Biaya	Efisiensi Sumber Daya
Standardisasi API & Service	Tinggi	Sedang	Tinggi
Microservices Architecture	Tinggi	Sedang–Rendah (awal)	Tinggi (jika sudah stabil)
API-First Development	Tinggi	Tinggi	Tinggi
Enterprise Service Bus (ESB)	Sedang	Tinggi (untuk enterprise)	Sedang

Tabel perbandingan keempat strategi ini diolah dari hasil studi pustaka (Aziz et al., 2020; Lercher et al., 2024; Dixit, 2024; Desina, 2020).

Pembahasan Temuan

Berdasarkan hasil studi pustaka yang dilakukan terhadap berbagai referensi ilmiah dan dokumentasi teknis, ditemukan bahwa masing-masing strategi *reuse service* dalam kerangka arsitektur Service-Oriented Architecture (SOA) telah diimplementasikan dalam berbagai dunia industri. Setiap strategi menunjukkan karakteristik dan keunggulan tersendiri dalam menjawab tantangan integrasi sistem serta efisiensi pengembangan teknologi informasi.

Strategi	Konteks Implementasi	Temuan Utama	Sumber
API-First Development	Sistem Informasi UMKM	Waktu pengembangan berkurang 30%, integrasi frontend-backend lebih lancar.	Hadian, Hakim, & Fanani, (2023)
Microservices Architecture	Sistem perbankan digital	Deployment harian, peningkatan performa, tetapi memerlukan pengujian intensif dan manajemen dependensi.	Lercher et al., (2024)
Enterprise Service Bus (ESB)	Sistem informasi rumah sakit di Pakistan	Integrasi 15+ sistem legacy, pengurangan beban manual, peningkatan orkestrasi dan konsistensi data.	Aziz et al., (2020)
Standardisasi API & Service	Sistem monitoring beasiswa berbasis REST API	Efisiensi pemanggilan data meningkat 45%, dokumentasi lebih rapi dengan Swagger/OpenAPI.	Muslih & Hasanah, (2019)

Berikut adalah ringkasan hasil temuan implementasi dari keempat strategi utama *reuse service* yang dianalisis, hasil temuan ini mendukung pembahasan bahwa tidak ada satu pendekatan tunggal yang paling unggul dalam segala konteks. Keberhasilan implementasi strategi *reuse service* sangat bergantung pada kesiapan teknologi, kompleksitas sistem, serta kebutuhan jangka panjang organisasi. Oleh karena itu, pemilihan strategi perlu mempertimbangkan kondisi spesifik dari proyek dan tujuan bisnis yang ingin dicapai.

4. KESIMPULAN

Penelitian ini telah melakukan analisis komparatif terhadap empat strategi *reuse service* dalam implementasi Service-Oriented Architecture (SOA), yaitu Standardisasi API & Service, Microservices Architecture, API-First Development, dan Enterprise Service Bus (ESB). Hasil kajian menunjukkan bahwa masing-masing strategi memiliki kekuatan dan tantangan yang unik, tergantung pada kebutuhan organisasi serta kompleksitas sistem yang dibangun.

Strategi API-First Development dinilai paling fleksibel dan relevan untuk mendukung efisiensi pengembangan sistem modern karena memungkinkan pengembangan paralel, mempercepat integrasi, serta meningkatkan kualitas dokumentasi layanan. Di sisi lain, Microservices Architecture unggul dalam hal skalabilitas dan modularitas, namun membutuhkan pengelolaan yang cermat. Standardisasi API memberikan fondasi kuat untuk interoperabilitas, sedangkan ESB masih sangat efektif dalam skenario integrasi lintas sistem kompleks, khususnya pada organisasi berskala besar dengan infrastruktur heterogen.

Berdasarkan temuan ini, disarankan agar pemilihan strategi *reuse service* disesuaikan dengan karakteristik proyek, tingkat kesiapan teknologi, dan tujuan jangka panjang dari organisasi. Pendekatan yang tepat tidak hanya akan meningkatkan efisiensi waktu dan biaya, tetapi juga memperkuat fondasi arsitektur sistem yang berkelanjutan dan adaptif terhadap perubahan teknologi.

5. REFERENSI

- Aziz, O., Farooq, M. S., Abid, A., Saher, R., & Aslam, N. (2020). Research Trends in Enterprise Service Bus (ESB) Applications: A Systematic Mapping Study. *IEEE Access*, 8, 31180–31197. <https://doi.org/10.1109/ACCESS.2020.2972195>
- Desina, G. C. (2020). (*PDF*) *Evaluating The Impact Of Cloud-Based Microservices Architecture On Application Performance*. https://www.researchgate.net/publication/371040409_Evaluating_The_Impact_Of_Cloud-Based_Microservices_Architecture_On_Application_Performance

- Dixit, P. (2024). *The Truth About Reusable APIs For Sustainable Success*. https://www.researchgate.net/publication/385272905_The_Truth_About_Reusable_APIs_For_Sustainable_Success
- Hadian, N., Hakim, M., & Fanan, M. R. (2023). *Implementasi Model Service-Oriented Architecture (SOA) dalam Perancangan Sistem Informasi UMKM*. <https://jurnal.unidha.ac.id/index.php/jtekstis/article/view/879/568>
- Hantana, J. S. P., & Hantana, J. S. P. (2013). Pendekatan Service Oriented Architecture (SOA) Pada Pelaksanaan E-Government di Kementerian Hukum dan HAM RI. *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*. <https://doi.org/10.23887/janapati.v2i3.9813>
- Lercher, A., Glock, J., Macho, C., & Pinzger, M. (2024). Microservice API Evolution in Practice: A Study on Strategies and Challenges. *Journal of Systems and Software*, 215. <https://doi.org/10.1016/j.jss.2024.112110>
- Muslih, M., & Hasanah, M. I. (2019). *PENERAPAN SERVICE ORIENTED ARCHITECTURE (SOA) GUNA MENINGKATKAN PELAYANAN MONITORING MAHASISWA BEASISWA DI PERGURUAN TINGGI BERBASIS WEB SERVICE*. *Jurnal Tekno Insentif*. <https://jurnal.lldikti4.or.id/index.php/jurnalteknoinsentif/article/view/137/69>
- Niknejad, N., Ismail, W., Ghani, I., Nazari, B., Bahari, M., & Hussin, A. R. B. C. (2020). Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation. In *Information Systems* (Vol. 91). Elsevier Ltd. <https://doi.org/10.1016/j.is.2020.101491>
- Rasyidi, B., & Pratama, F. (2024). *View of Sistem Monitoring Server di PT. XYZ Media Indonesia Berbasis Grafana dan Prometheus*. <https://journal.irpi.or.id/index.php/malcom/article/view/1546/793>