

Implementasi ICA Framework Untuk Peningkatan Efisiensi Waktu Dan Sumber Daya Dalam Proses Data Ingestion Di Informatica Dei (Studi Kasus Bank XYZ)

Muhamad Iqbal Farhan Maulana¹, Patah Herwanto²

^{1,2} Program Studi Teknik Informatika STMIK IM, Jl. Belitung No.7 Bandung
Iqbalfm78@gmail.com¹, pherwanto@stmik-im.ac.id²

Abstract

The Informatica Data Engineering Integration (DEI) platform plays a crucial role in managing data flow at Bank XYZ, particularly in data ingestion, the initial transfer of data from various sources to the centralized repository, Datalake. The development and testing of ingestion pipelines are often time-consuming, and high demand for new pipeline creation leads to significant bottlenecks. To address this challenge, a composable parameterized job (a reusable data processing task configurable with varying inputs) is initiated within Informatica DEI. This parameterized job is accessed and managed via a web application, forming an end-to-end framework (covering the complete process from input to execution) called the ICA Framework. This framework aims to streamline the data ingestion process by transforming manual and repetitive tasks into a simple input & click operation via the ICA website. This research encompasses the design of the ICA Framework, identification of existing process inefficiencies, development of the proposed process, and its implementation and evaluation. Implementation of the ICA Framework demonstrated significant positive results in process time and human resource efficiency for data ingestion pipeline development and testing. Specifically, the average development time per pipeline was reduced from 240 minutes to 5 minutes (97.91% efficiency), and testing time from 90 minutes to 1 minute (98.9% efficiency), resulting in an overall process time efficiency of 98.4%. Human resource utilization efficiency increased by 65% (from 5 to 1 person for development, and 2 to 1 person for testing). Thus, the ICA Framework is proven effective in overcoming data ingestion bottlenecks at Bank XYZ, significantly improving operational efficiency and data delivery speed.

Keywords: Data Ingestion, Informatica DEI, ICA Framework, Process Efficiency, Automation.

Abstrak

Platform Informatica Data Engineering Integration (DEI) memainkan peran penting dalam pengelolaan aliran data di Bank XYZ, terutama dalam proses data ingestion, yaitu perpindahan data awal dari berbagai sumber ke penyimpanan terpusat, Datalake. Proses pengembangan ingestion pipeline dan pengujian secara tradisional membutuhkan waktu signifikan, yaitu sekitar 240 menit untuk pengembangan dan 90 menit untuk pengujian, sehingga total 330 menit per pipeline dalam kondisi ideal. Tingginya permintaan pembuatan pipeline baru seringkali menyebabkan bottleneck. Untuk mengatasi tantangan ini, sebuah parameterized job yang bersifat composable diinisiasikan dalam Informatica DEI guna mencapai efisiensi. Parameterized job ini, yang mampu mengambil parameter terkait ingestion dan dapat digunakan kembali, diimplementasikan dalam sebuah aplikasi web untuk kemudahan akses oleh developer dan tester. Gabungan ini membentuk sebuah framework end-to-end yang disebut Integration Composable Architecture (ICA) Framework. ICA Framework bertujuan menyederhanakan proses data ingestion dengan mengubah tugas manual dan repetitif menjadi proses input dan klik yang sederhana melalui situs web ICA. Penelitian ini mencakup perancangan ICA Framework, identifikasi kekuatan dan kelemahan proses yang ada, pengembangan proses yang diusulkan, serta implementasi dan evaluasi. Metode penelitian yang digunakan adalah eksploratif dengan pendekatan kualitatif dan kuantitatif, melibatkan observasi, wawancara, dan analisis log sistem. Hasil implementasi menunjukkan bahwa ICA Framework berhasil mengurangi waktu pengembangan pipeline dari 240 menit menjadi 5 menit (efisiensi 97,91%) dan waktu pengujian dari 90 menit menjadi 1 menit (efisiensi 98,9%), sehingga mencapai efisiensi waktu proses keseluruhan sebesar 98,4%. Selain itu, terjadi peningkatan efisiensi penggunaan sumber daya manusia sebesar 65%. Dengan demikian, ICA Framework terbukti efektif dalam

mengurangi waktu pengembangan, mengefisienkan penggunaan SDM, dan berpotensi meningkatkan kualitas serta akurasi data ingestion.

Kata kunci: Data Ingestion, Informatica DEI, ICA Framework, Efisiensi Proses, Automasi.

I. PENDAHULUAN

Seiring kemajuan teknologi digital dan peningkatan volume Big Data, sektor perbankan dituntut untuk mengoptimalkan pengelolaan data guna mendukung pengambilan keputusan strategis yang lebih baik. Kemampuan mengolah data secara cepat dan akurat menjadi krusial, terutama ketika berhadapan dengan berbagai sistem operasional yang kompleks. Pemanfaatan Big Data diyakini dapat memberikan *value creation* signifikan, mulai dari peningkatan layanan, efisiensi biaya, inovasi produk, hingga perumusan strategi bisnis (Amana, Liu, & Alariqi, 2022). Hal ini sejalan dengan upaya mencapai keunggulan kompetitif di pasar (Rahardja & Hariyanto, 2023).

Di Bank XYZ, proses pemindahan data atau *data ingestion* yang membawa data awal ke dalam Datalake menghadapi tantangan berarti. Data internal menunjukkan bahwa dalam enam bulan terakhir, permintaan untuk *pipeline* data meningkat sebesar 40%, menyebabkan keterlambatan hingga tiga hari. Keterlambatan ini tidak hanya menghambat efisiensi operasional tetapi juga berdampak buruk pada kualitas informasi yang digunakan untuk pengambilan keputusan. Permasalahan utama dalam penelitian ini adalah tingginya ketergantungan pada proses manual yang menyebabkan keterlambatan dan potensi kesalahan dalam *pipeline data ingestion*, yang berakibat langsung pada proses bisnis dan kualitas pengambilan keputusan.

Meskipun beberapa solusi otomatisasi telah diterapkan di industri perbankan, sebagian besar masih bergantung pada proses manual yang rentan kesalahan dan memakan banyak waktu. Kebutuhan mendesak muncul untuk mengurangi proses manual serta meningkatkan kecepatan dan keakuratan pengolahan data. Sejalan dengan tantangan tersebut, sejumlah studi menunjukkan pentingnya pendekatan otomatisasi. Chen, Chiang, dan Storey (2012) menunjukkan bahwa penerapan *business intelligence* dan otomatisasi analitik bisa memberikan dampak besar bagi efisiensi operasional. Marx (2013), sebagaimana dikutip dalam Archibald et al. (2013), menekankan bahwa tantangan dalam mengelola Big Data memerlukan pendekatan inovatif untuk menangani kompleksitas dan volume data yang terus bertambah. Konsep *composable architecture* dan *parameterized job*, seperti yang dijelaskan dalam konteks pengembangan *framework* (Cwalina & Abrams, 2008) dan implementasi teknis di Informatica DEI (Informatica LLC Inc., 2021), menawarkan dasar teoritis untuk solusi yang diusulkan. Peralatan untuk *data ingestion* dan preparasi juga terus berkembang untuk mengatasi tantangan ini (Alwidian, Rahman, Gnaim, & Al-Taharwah, 2020).

Dalam konteks ini, penelitian ini memperkenalkan Kerangka ICA (Integration Composable Architecture) sebagai solusi inovatif. Kerangka kerja ini mengubah cara konvensional yang bergantung pada interaksi manual

dengan sistem Informatica DEI menjadi proses berbasis web yang lebih sederhana dengan konsep input dan klik. Pendekatan ini diharapkan dapat mengurangi risiko kesalahan manusia dan mempercepat proses pengolahan data. Sun et al. (2014) menyatakan bahwa penerapan kerangka kerja berbasis automasi dalam analitik data pelanggan dapat meningkatkan efisiensi dan akurasi, yang sejalan dengan tujuan penelitian ini. Lebih jauh lagi, literatur terbaru juga menekankan bahwa penerapan analitik data di sektor keuangan memberikan kontribusi pada pengambilan keputusan yang lebih tepat dan strategis (Nobanee et al., 2021; Hasan, Hoque, & Le, 2023).

Oleh karena itu, penelitian ini bertujuan untuk mengidentifikasi permasalahan utama dalam proses pengolahan data di Bank XYZ, merancang dan mengimplementasikan Kerangka ICA, serta mengevaluasi efektivitasnya berdasarkan pengukuran efisiensi waktu dan penurunan tingkat kesalahan manusia. Kontribusi penelitian ini adalah menyediakan sebuah *framework* praktis yang dapat meningkatkan efisiensi operasional secara signifikan dalam *data ingestion*, serta memberikan dasar bagi bank untuk mengambil keputusan yang lebih baik berdasarkan data yang lebih cepat tersedia dan andal. Dengan mengintegrasikan teknologi aplikasi web dan sistem yang terparameter di Informatica DEI, diharapkan ICA Framework dapat menjadi solusi inovatif yang mudah diadaptasi oleh institusi perbankan lain yang menghadapi tantangan serupa.

II. METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksploratif dengan metode campuran (kualitatif dan kuantitatif) untuk merancang, mengimplementasikan, dan mengevaluasi Kerangka Kerja ICA. Pendekatan ini dipilih karena penelitian bertujuan mengembangkan solusi baru (ICA Framework) untuk masalah praktis di Bank XYZ, yang sejalan dengan prinsip-prinsip Design Science Research (DSR) yang umum digunakan dalam pengembangan sistem informasi berbasis solusi.

1. Desain Penelitian

Penelitian ini dibagi menjadi beberapa tahap utama:

a. Analisis dan Ide

Melakukan observasi mendalam terhadap proses *data ingestion* yang sedang berjalan di Bank XYZ melalui wawancara dengan *developer*, *tester*, dan tim Data Operations, serta telaah dokumen dan log sistem. Pada tahap ini juga dilakukan identifikasi masalah yang disebabkan oleh proses manual dan perumusan kebutuhan pengguna.

b. Desain

Mengubah temuan analisis menjadi desain sistem untuk Kerangka Kerja ICA. Hal ini termasuk membuat model alur kerja baru (Gambar 2), merancang arsitektur aplikasi berbasis web (Gambar

3), dan spesifikasi teknis *parameterized job* di Informatica DEI.

c. Pengembangan

Menerapkan desain ke dalam sistem fungsional. Hal ini melibatkan pembuatan *parameterized job* di Informatica DEI versi 10.5.5, pengembangan *server-side* menggunakan PHP versi 8.3 dan database Microsoft SQL Server 2019, serta *client-side* menggunakan Bootstrap versi 5.1. Detail spesifikasi alat dan skema integrasi dijelaskan lebih lanjut pada bagian Hasil Penelitian.

d. Implementasi

Meluncurkan sistem dalam lingkungan pengembangan secara terbatas (pilot testing).

e. Evaluasi

Mengukur kinerja sistem dengan membandingkan efisiensi waktu dan penggunaan sumber daya manusia sebelum dan sesudah Kerangka Kerja ICA digunakan. Data "Before ICA" diperoleh dari catatan historis proyek dan estimasi rata-rata dari tim terkait. Data "After ICA" diukur melalui observasi langsung dan pencatatan log sistem selama pilot testing. Umpan balik kualitatif juga dikumpulkan dari pengguna melalui kuesioner terstruktur yang menilai aspek kemudahan penggunaan (*usability*) dan manfaat yang dirasakan (*perceived benefit*).

Pendekatan ini sejalan dengan literatur yang ada tentang pengembangan sistem informasi, yang menekankan pentingnya memvalidasi aspek kualitatif dan kuantitatif secara bersamaan.

2. Teknik Pengumpulan Data dan Sampling

Data dikumpulkan melalui:

- **Observasi Langsung dan Wawancara:** Dilakukan pada fase Analisis & Ide untuk menggali permasalahan dan kebutuhan pengguna. Responden dipilih secara *purposive sampling*, yaitu individu yang terlibat langsung dalam proses *data ingestion* (pengembang, *tester*, tim Data Operations) untuk memastikan relevansi informasi.
- **Studi Dokumen dan Log Sistem:** Digunakan untuk mendapatkan data kuantitatif mengenai waktu pengerjaan, jumlah sumber daya yang terlibat, dan frekuensi permintaan *pipeline* sebelum implementasi ICA Framework, mendukung analisis kondisi awal.
- **Pengukuran Langsung dan Pencatatan Log Sistem:** Dilakukan selama fase Evaluasi untuk mengukur waktu pengerjaan dan sumber daya setelah implementasi ICA Framework. Waktu pengerjaan diukur dari log sistem aplikasi ICA dan Informatica DEI. Log sistem diekstraksi dan dibersihkan dari anomali (misalnya, waktu eksekusi yang tidak wajar karena gangguan sistem eksternal) sebelum dianalisis.
- **Kuesioner:** Digunakan untuk mengumpulkan *feedback* kualitatif dari pengguna (pengembang dan *tester*) mengenai sistem ICA Framework setelah periode pilot testing. Instrumen kuesioner mencakup pertanyaan tertutup dengan skala Likert (misalnya, menilai kemudahan penggunaan, kepuasan) dan pertanyaan terbuka untuk menggali persepsi

pengguna terkait kelebihan, kekurangan, dan saran perbaikan. Contoh item: "Seberapa mudah Anda membuat *pipeline data ingestion* baru menggunakan ICA Framework?" (Skala 1-Sangat Sulit hingga 5-Sangat Mudah).

3. Teknik Validasi dan Analisis Data

Validasi hasil penelitian dilakukan dengan:

- **Uji Coba Sistem:** Melakukan simulasi penggunaan ICA Framework di lingkungan pengembangan, dilanjutkan dengan uji coba terbatas (pilot testing) pada kasus nyata di Bank XYZ untuk memvalidasi fungsionalitas dan alur kerja.
- **Analisis Statistik Deskriptif:** Digunakan untuk membandingkan efisiensi sistem (waktu proses dan SDM) sebelum dan sesudah implementasi. Efisiensi dihitung sebagai persentase pengurangan waktu atau sumber daya. Perangkat lunak analisis yang digunakan adalah Microsoft Excel.
- **Analisis Data Kualitatif:** *Feedback* dari kuesioner dan catatan wawancara dianalisis secara tematik untuk mengidentifikasi pola, kelebihan, kekurangan sistem, dan area perbaikan dari perspektif pengguna.
- **Triangulasi:** Mengintegrasikan data kuantitatif (log sistem, pengukuran waktu) dan kualitatif (wawancara, kuesioner) untuk meningkatkan keandalan dan keakuratan temuan. Validitas instrumen kuesioner diuji melalui *expert review* oleh praktisi data di Bank XYZ dan *pilot testing* pada sekelompok kecil pengguna sebelum disebarkan secara luas.

III. HASIL PENELITIAN

Bagian ini menyajikan temuan dari implementasi ICA Framework dan membahas dampak serta implikasinya terhadap efisiensi proses data ingestion di Bank XYZ. Data kuantitatif mengenai efisiensi waktu dan sumber daya manusia akan diuraikan, diikuti dengan interpretasi mendalam terhadap temuan tersebut.

1. Analisis Proses Data Ingestion & Extraction

Sebelum implementasi ICA Framework, proses *data ingestion* di Divisi Data Bank XYZ melibatkan alur kerja yang kompleks dan memakan waktu, dengan beberapa tahapan manual. Alur kerja ini melibatkan tiga peran utama (*stakeholder*): *Developer*, *Tester* (QC), dan *Data Operations*. **Gambar 1** (Diagram Alir Proses Eksisting) menyajikan diagram alir proses *data ingestion* yang ada di Divisi Data Bank XYZ sebelum implementasi ICA Framework. Proses ini melibatkan langkah-langkah manual yang signifikan bagi setiap peran, dimulai dari pembuatan *mapping* oleh *developer*, melalui berbagai tahap pengujian oleh *tester*, hingga *re-deploy application* oleh *Data Operations*.



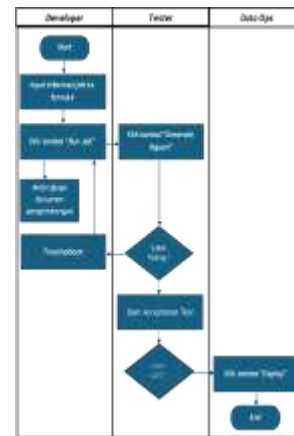
Gambar 1 : Eksisting proses data Ingestion

Gambar 1 menggambarkan secara detail alur kerja sekuensial dan pembagian tugas antara *Developer*, *Tester*, dan *Data Operations* dalam proses *data ingestion* sebelum adanya ICA Framework. Alur ini menunjukkan banyaknya intervensi manual dan potensi *bottleneck* pada setiap transisi antar peran.

- Fase Pengembangan:** Pengembang bertanggung jawab untuk membuat *Mapping*, mengintegrasikannya ke dalam *workflow*, melakukan *deploy application*, menjalankan *workflow*, dan melengkapi dokumentasi.
- Fase Pengujian:** *Tester* melakukan review terhadap *job*, termasuk menghitung jumlah baris, memeriksa tipe data, mengambil sampel data, dan melaksanakan *User Acceptance Test (UAT)*.
- Fase Deployment ke Production:** Tim *Data Operations* mengalihkan koneksi dari lingkungan pengembangan ke lingkungan produksi dan melakukan *re-deploy application*.

Proses *existing* ini memiliki beberapa kelebihan, antara lain dokumentasi yang baik dan pengujian yang seksama. Namun, terdapat juga kekurangan signifikan, yaitu pengembangan yang repetitif dan manual, review *tester* yang redundant, dan keterlibatan pengembang dalam proses *deployment*.

Sebagai perbandingan, ICA Framework mengusulkan proses baru yang lebih otomatis, mengubah proses manual menjadi input dan klik melalui antarmuka web. Dalam proses yang diusulkan:



Gambar 2 : Usulan proses data Ingestion dengan ICA Framework

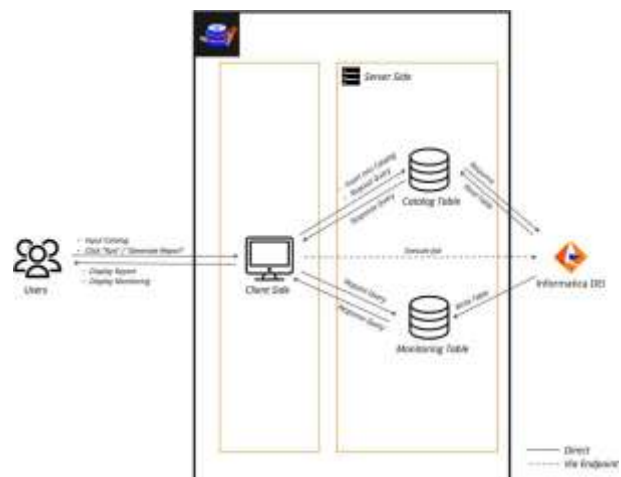
Gambar 2 menunjukkan penyederhanaan signifikan pada langkah-langkah yang dilakukan oleh *Developer*, *Tester*, dan *Data Operations*. Interaksi manual dengan sistem Informatica DEI digantikan dengan penggunaan formulir dan tombol aksi pada antarmuka web ICA, mengurangi jumlah langkah dan potensi kesalahan.

- Pengembang memasukkan detail pekerjaan dan mengeksekusi dengan tombol "Run Mapping".
- Tester sudah di buatkan Report Dengan Klik "Generate Report"
- Data Operations* melakukan *deployment* secara mandiri.

ICA Framework menyederhanakan dokumentasi dan mengatasi kekurangan proses *existing*, sambil mempertahankan dan meningkatkan aspek positifnya.

2. Arsitektur ICA Framework

ICA Framework dirancang sebagai aplikasi *web-based* yang mendukung pengembangan *pipeline ingestion* data secara *end-to-end*, baik di lingkungan *development* maupun *production*. **Gambar 3** (Arsitektur ICA Framework) menggambarkan arsitektur ICA Framework yang terdiri dari tiga komponen utama: *Client-Side*, *Server-Side*, dan *Informatica DEI* (eksternal)..



Gambar 3 : Arsitektur ICA Framework

Arsitektur ICA Framework pada Gambar 3 menunjukkan interaksi antara pengguna (melalui *Client-Side*), server aplikasi (yang mengelola *Catalog Table* dan *Monitoring Table*), dan platform Informatica DEI. Panah menunjukkan alur permintaan dan respons, baik melalui interaksi langsung maupun *endpoint API*.

- a. **Client-Side:** Ini adalah *interface* utama bagi pengguna dalam case ini developer data ingestion untuk memasukkan informasi *ingestion*, berinteraksi dengan *Informatica DEI* (melalui tombol "Run" dan "Generate Report"), dan memantau pekerjaan melalui *dashboard*.
- b. **Server-Side:** Komponen ini menyimpan informasi dari *Client-Side* dalam dua tabel: *Catalog Table* (menyimpan detail *ingestion*) dan *Monitoring Table* (menyimpan status pekerjaan). *Server-Side* juga menangani komunikasi dengan *Client-Side* dan *Informatica DEI*.
- c. **Informatica DEI:** Ini adalah *platform* eksternal yang menjalankan pekerjaan *ingestion* dan laporan. ICA Framework berkomunikasi dengan *Informatica DEI* melalui *API* untuk mengeksekusi *parameterized job*.

Dalam lingkungan *development*, pengguna memicu eksekusi *job* melalui *Client-Side*. Alur *parameterized job* untuk pengembangan *ingestion* dari sumber *RDBMS* melibatkan *API trigger mapping*, *workflow* yang membaca *Catalog Table*, dan *workflow* dinamis untuk memetakan sumber dan target. Alur untuk sumber *Flat File* serupa, tetapi mencakup *workflow* tambahan untuk mengambil *Flat File* ke *Informatica DEI*. Di lingkungan *production*, eksekusi *job* didasarkan pada *schedule* di *Informatica DEI*. *Informatica DEI* menjalankan *job* terjadwal setelah memeriksa *Catalog Table*, dan status *job* ditampilkan di *dashboard* web ICA.

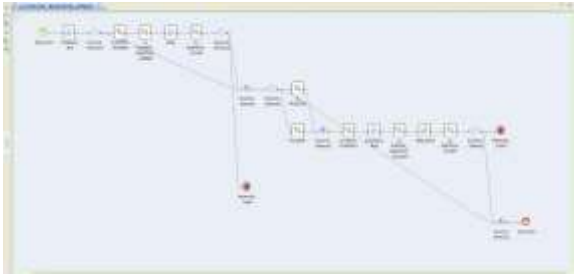
3. Pengembangan Sistem ICA Framework

Pengembangan ICA Framework melibatkan realisasi rancangan sistem, yang mencakup pengembangan *parameterized job* di *Informatica DEI*, *Server-Side*, dan *Client-Side* (web ICA)

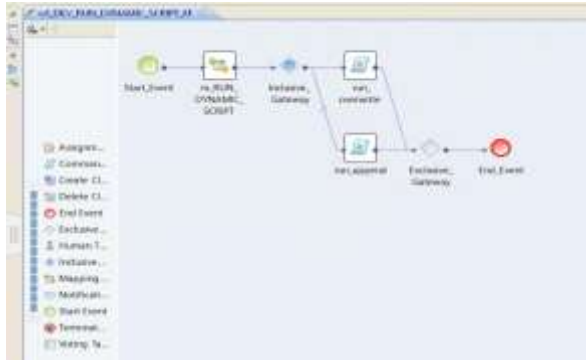
- 1. **Pengembangan Parameterized Job di Informatica DEI:** *Parameterized job* dikembangkan dalam beberapa jenis untuk lingkungan *development* dan *production*, berdasarkan tipe sumber (*RDBMS* dan *Flat File*) dan mode penulisan (*Append* dan *Overwrite*).



Gambar 4 : Pengembangan Workflow Get Catalog



Gambar 5 : Pengembangan Workflow Dynamic Ingestion



Gambar 6 : Pengembangan Workflow Get Flat File

- 2. **Pengembangan Server-Side:** *Server-Side* melibatkan pembuatan *Catalog Table* dan *Monitoring Table*, serta pengembangan *API* untuk komunikasi dengan *Client-Side* dan *Informatica DEI*.

Tabel 1: Sampel API Endpoint untuk Run Job dari Web ICA

Nama API	API Run Dev Append
Method	POST
URL	curl -X POST "http://60.0.98.130:8098/DataIntegrationService/modules/ms/v1/applications/App_DEV_CATALOG_INGESTION/mappings/m_CALL_JOB_DEV_APPEND" -H "accept: application/json" -H "username: {username}" -H "encryptedpassword: {password}" -H "securitydomain: Native" -H "servicename: DIS_ING_XYZ" -H "Content-Type: application/json" -d '{"action": "RunDeployedMapping", "customParameters": [{"name": "p_id", "value": "{value_id}"}]}'
Authentication	Basic

3. **Pengembangan Web ICA (Client-Side):** Pengembangan *front-end* web ICA mencakup perancangan halaman *login*, *dashboard*, dan formulir pengisian *catalog*.



Gambar 7 : Tampilan Halaman Login Website ICA



Gambar 8 : Tampilan ICA Dashboard

Gambar 9 : Tampilan Catalog Job di Website ICA

Gambar 10 : Tampilan Catalog Form di Website ICA

4. Manajemen Risiko ICA Framework

Pengembangan ICA Framework melibatkan potensi risiko yang perlu dikelola. Manajemen risiko dilakukan dengan mengacu pada pedoman yang berlaku di Bank XYZ (Pedoman Pelaksanaan SE. 09-DIR/DRD/04/2024),

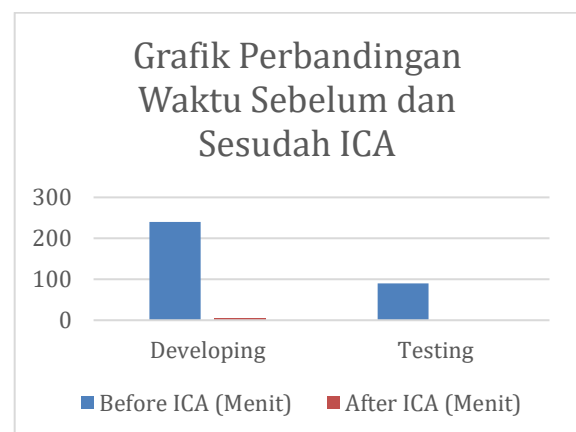
meliputi identifikasi, pengukuran, penanganan, dan pemantauan risiko.

1. **Identifikasi Risiko:** Risiko yang diidentifikasi mencakup risiko bank (risiko operasional, risiko strategis, risiko kepatuhan) dan risiko digital (risiko keamanan informasi, risiko resiliensi, risiko kepatuhan, risiko data, risiko sumber daya manusia, risiko pihak ketiga).
2. **Pengukuran Risiko:** Risiko-risiko tersebut diukur menggunakan *Inherit Risk Score (IRS)* berdasarkan tingkat dampak (*impact*) dan probabilitas (*likelihood*).
3. **Penanganan Risiko:** Metode penanganan risiko (*mitigate, avoid, accept*) ditentukan untuk setiap risiko, bersama dengan Nilai Efektivitas Kontrol (NEK).
4. **Pemantauan Risiko:** Risiko-risiko tersebut dipantau secara berkala, dengan periode pemantauan dan pelaporan yang telah ditentukan.
5. **Efisiensi Waktu Proses**

Implementasi ICA Framework secara signifikan mempengaruhi waktu yang dibutuhkan untuk menyelesaikan tugas *data ingestion*. Tabel 2 (Perbandingan Waktu Sebelum dan Sesudah ICA) menyajikan perbandingan waktu proses sebelum dan sesudah implementasi *framework*. Data "Before ICA" diperoleh dari rata-rata waktu pengerjaan proyek serupa yang terdokumentasi dalam log sistem dan dokumen proyek historis, serta estimasi dari tim *developer* dan *tester* berdasarkan pengalaman mereka pada proses manual. Data "After ICA" diukur melalui observasi langsung dan pencatatan log sistem selama periode pilot testing ICA Framework.

Tabel 2: Perbandingan Waktu Sebelum dan Sesudah ICA

Proses	Before ICA (Menit)	After Ica (Menit)	Tingkat Efisiensi
Developing	240	5	97,91%
Testing	90	1	98,9%



Gambar 11 : Grafik perbandingan Waktu Sebelum dan Sesudah ICA

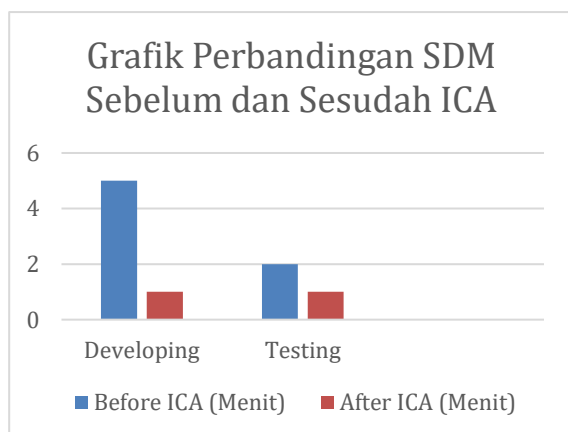
Hasil yang tertera pada Tabel 2 dan Gambar 11 memperlihatkan adanya pengurangan waktu proses yang sangat signifikan pada kedua tahapan. Waktu yang dibutuhkan untuk mengembangkan *pipeline* data berkurang dari 240 menit menjadi hanya 5 menit, mencerminkan peningkatan efisiensi sebesar 97,91%. Serupa dengan itu, waktu pengujian juga mengalami penurunan yang drastis dari 90 menit menjadi 1 menit, dengan peningkatan efisiensi mencapai 98,9%. Secara keseluruhan, implementasi ICA Framework menghasilkan efisiensi waktu proses rata-rata sebesar 98,4%. Pengurangan waktu proses ini mengindikasikan bahwa ICA Framework berhasil mengotomatiskan tugas-tugas manual dan repetitif, sehingga pengembang dan penguji dapat menyelesaikan pekerjaan mereka dengan lebih cepat. Otomatisasi ini mengurangi waktu tunggu dan mempercepat keseluruhan siklus pengembangan *data ingestion*. Dalam konteks Bank XYZ, yang merupakan institusi keuangan besar dengan sistem IT yang kompleks dan volume data yang tinggi, efisiensi waktu ini sangat krusial untuk memastikan ketersediaan data yang tepat waktu untuk berbagai kebutuhan bisnis.

6. Efisiensi Sumber Daya Manusia

Selain efisiensi waktu, implementasi ICA Framework juga berdampak positif pada penggunaan sumber daya manusia. Tabel 3 (Perbandingan Penggunaan SDA Sebelum dan Sesudah ICA) membandingkan jumlah sumber daya manusia yang terlibat dalam proses *data ingestion* sebelum dan sesudah implementasi *framework*, dengan asumsi volume pekerjaan yang sama (misalnya, pengembangan 10 *pipeline*).

Tabel 3: Perbandingan Penggunaan SDM Sebelum dan Sesudah ICA

Proses	Before ICA(Orang)	After Ica (Orang)	Tingkat Efisiensi
<i>Developing</i>	5	1	80%
<i>Testing</i>	2	1	50%



Gambar 12 : Grafik perbandingan SDA Sebelum dan Sesudah ICA

Hasil pada Tabel 3 dan Gambar 12 menunjukkan bahwa ICA Framework memungkinkan pengurangan jumlah sumber daya manusia yang dibutuhkan untuk kedua tahapan proses. Tahap pengembangan yang sebelumnya membutuhkan 5 orang untuk menangani

volume pekerjaan tertentu, setelah implementasi *framework* hanya membutuhkan 1 orang. Tahap pengujian juga mengalami pengurangan dari 2 orang menjadi 1 orang. Secara rata-rata, implementasi ICA Framework menghasilkan efisiensi sumber daya manusia sebesar 65%. Pengurangan ini menunjukkan bahwa ICA Framework memungkinkan optimasi alokasi sumber daya manusia. Dengan otomatisasi tugas, lebih sedikit orang yang dibutuhkan untuk menyelesaikan pekerjaan yang sama, sehingga sumber daya manusia dapat dialokasikan ke tugas lain yang lebih strategis, seperti pengembangan model analitik atau integrasi data untuk aplikasi bisnis baru. Ini tidak hanya meningkatkan efisiensi, tetapi juga berpotensi mengurangi biaya operasional terkait dengan tenaga kerja.

IV. PEMBAHASAN

Temuan dari penelitian ini secara konsisten menunjukkan bahwa ICA Framework secara signifikan meningkatkan efisiensi proses *data ingestion* di Bank XYZ. Peningkatan efisiensi waktu sebesar 98,4% dan efisiensi sumber daya manusia sebesar 65% menunjukkan keberhasilan otomatisasi tugas-tugas manual dan repetitif. Penyebab utama peningkatan efisiensi ini dari sudut pandang sistem adalah standarisasi alur kerja melalui *parameterized job* di Informatica DEI dan penyederhanaan interaksi pengguna melalui antarmuka web. *Parameterized job*, yang merupakan inti dari otomatisasi ini, dirancang untuk menerima parameter dinamis (seperti nama tabel sumber, tabel target, mode penulisan) dari aplikasi web ICA. Ketika pengguna mengklik "Run Job" pada web ICA, *endpoint* API di *server-side* akan dipanggil. API ini kemudian memicu *workflow* di Informatica DEI yang telah dikonfigurasi untuk menjalankan *mapping* generik yang menggunakan parameter tersebut untuk melakukan proses *ingestion* secara otomatis tanpa perlu pembuatan atau modifikasi manual *mapping* untuk setiap *job* baru. Ini adalah kunci bagaimana eksekusi otomatis dilakukan dan mengapa *parameterized job* sangat penting untuk mencapai efisiensi yang diamati.

Hasil ini selaras dengan temuan Chen, Chiang, & Storey (2012) yang menunjukkan dampak besar otomatisasi analitik bagi efisiensi operasional, dan Sun et al. (2014) yang menyatakan bahwa kerangka kerja berbasis automasi meningkatkan efisiensi dan akurasi. ICA Framework memberikan contoh praktis bagaimana otomatisasi diterapkan dalam *data ingestion*, sejalan dengan praktik *state-of-the-art* dalam *data engineering* yang menekankan penggunaan *template* dan *framework* untuk orkestrasi *pipeline* (misalnya, pendekatan yang mirip dengan yang ditemukan pada alat seperti Apache Airflow atau dbt) (Oussous, Benjelloun, Lahcen, & Belfkih, 2018; Patgiri, 2018). Pentingnya alat yang efisien untuk *data ingestion* juga ditekankan oleh Kona (2018) dan Alwidian et al. (2020).

Perbandingan antara *existing process* dan *proposed process* menyoroti penyederhanaan alur kerja. Proses manual yang kompleks diubah menjadi alur kerja efisien berbasis web dan eksekusi otomatis. Pengurangan risiko

kesalahan manusia juga menjadi fokus, yang mendukung pernyataan Veldkamp (2023) mengenai pentingnya tata kelola data yang baik untuk memaksimalkan nilai data. Jika dianalisis menggunakan kerangka seperti Technology Acceptance Model (TAM), kemudahan penggunaan (*perceived ease of use*) yang ditawarkan oleh antarmuka web ICA dan manfaat yang dirasakan (*perceived usefulness*) dalam bentuk percepatan proses kemungkinan besar berkontribusi pada penerimaan positif oleh *developer* dan *tester*, yang pada akhirnya mendorong peningkatan efisiensi. Hasil ini membuktikan dan memperluas pemahaman tentang bagaimana penerapan *framework* otomatisasi dapat secara langsung meningkatkan produktivitas dalam konteks spesifik *data ingestion* di lingkungan korporat.

Arsitektur ICA Framework yang terdiri dari *Client-Side*, *Server-Side*, dan Informatica DEI memungkinkan alur kerja yang terstruktur dan terkontrol. Manajemen risiko yang komprehensif, termasuk identifikasi risiko operasional, strategis, dan kepatuhan (sesuai Pedoman Pelaksanaan SE. 09-DIR/DRD/04/2024 dari Bank XYZ), memastikan implementasi yang aman dan efektif. Peningkatan efisiensi dan potensi peningkatan akurasi *data ingestion* (melalui pengurangan intervensi manual) ini mendukung penyediaan data yang andal untuk pengambilan keputusan strategis di sektor keuangan (Nobanee et al., 2021; Hasan, Hoque, & Le, 2023). Arsitektur ICA Framework yang terdiri dari *Client-Side*, *Server-Side*, dan Informatica DEI (eksternal) memungkinkan alur kerja *data ingestion* yang terstruktur dan terkontrol. *Parameterized job* di Informatica DEI, yang dipicu oleh input pengguna melalui *Client-Side*, menyediakan mekanisme yang efisien untuk mengeksekusi tugas *ingestion*.

V. KESIMPULAN

Penelitian ini menunjukkan bahwa implementasi ICA Framework memberikan dampak positif signifikan terhadap efisiensi proses *data ingestion* di Bank XYZ. Temuan utama adalah:

1. ICA Framework berhasil mengatasi masalah utama dalam proses *data ingestion* yang ada, yaitu proses manual dan repetitif yang memakan waktu dan rentan kesalahan. Ini dicapai dengan mengubah proses tersebut menjadi alur kerja yang lebih efisien berbasis input dan klik melalui aplikasi web yang berinteraksi dengan sistem Informatica DEI melalui *parameterized job*, sehingga memudahkan pengguna dari tim *developer*, *tester*, dan *data operations*.
2. Implementasi ICA Framework menghasilkan peningkatan efisiensi waktu proses yang sangat signifikan, yaitu sebesar 98,4% (dari rata-rata total 330 menit per *pipeline* untuk pengembangan dan pengujian, menjadi kurang lebih 6 menit per *pipeline*). Selain itu, tercatat peningkatan efisiensi penggunaan sumber daya manusia sebesar 65% untuk volume pekerjaan yang sama.
3. Peningkatan efisiensi ini secara langsung berkontribusi pada pengurangan potensi biaya operasional, peningkatan produktivitas tim data, dan yang terpenting, penyediaan data yang lebih cepat

dan akurat untuk mendukung pengambilan keputusan bisnis di Bank XYZ. Dengan demikian, ICA Framework mendukung upaya transformasi digital dan implementasi tata kelola data yang lebih adaptif dan responsif di dalam organisasi.

4. Kontribusi unik dari ICA Framework adalah penyediaan solusi *composable* dan terstandarisasi yang dirancang khusus untuk kebutuhan dan lingkungan *data ingestion* menggunakan Informatica DEI di Bank XYZ, yang mana prinsip-prinsipnya berpotensi untuk diadaptasi dan dikembangkan lebih lanjut oleh institusi lain yang menghadapi tantangan serupa dalam orkestrasi alur kerja data.

Berdasarkan hasil penelitian dan keterbatasan yang ada, saran untuk penelitian selanjutnya atau penerapan praktis adalah:

Saran Akademis/Penelitian Lanjutan:

1. Evaluasi Kualitas Data dan Kepuasan Pengguna: Melakukan evaluasi kuantitatif terhadap dampak ICA Framework pada kualitas data aktual yang di-*ingest* (misalnya, penurunan tingkat kesalahan, peningkatan konsistensi) dan mengukur tingkat kepuasan pengguna secara lebih mendalam menggunakan instrumen standar.
2. Studi Komparatif dan Generalisasi: Mengeksplorasi implementasi atau adaptasi prinsip ICA Framework di berbagai institusi keuangan lain atau bahkan industri berbeda untuk menguji generalisasi temuan dan mengidentifikasi faktor kontekstual yang mempengaruhi keberhasilan.
3. Pengembangan Fitur Lanjutan: Mengembangkan dan menguji fitur tambahan untuk ICA Framework, seperti integrasi dengan sistem *data quality* (misalnya Informatica DQ) untuk *data profiling* dan validasi data otomatis sebelum proses *ingestion*, atau eksplorasi penggunaan elemen kecerdasan buatan (AI) untuk validasi parameter input atau deteksi anomali.
4. Analisis Penerimaan Teknologi: Menganalisis lebih dalam aspek penerimaan pengguna terhadap ICA Framework menggunakan model teoritis seperti Technology Acceptance Model (TAM) atau Unified Theory of Acceptance and Use of Technology (UTAUT) untuk memahami faktor-faktor kunci yang mempengaruhi adopsi dan penggunaan berkelanjutan.

Saran Praktis:

1. Pelatihan dan Dokumentasi Komprehensif: Melakukan pelatihan berkelanjutan bagi seluruh pengguna ICA Framework (pengembang, *tester*, *data operations*) dan menyediakan dokumentasi pengguna yang lengkap, jelas, serta mudah diakses untuk memastikan kelancaran adopsi dan penggunaan yang optimal.
2. Integrasi dengan Tata Kelola Data Perusahaan: Mempertimbangkan integrasi ICA Framework dengan *platform* atau inisiatif tata kelola data

yang lebih luas di Bank XYZ untuk memastikan kepatuhan terhadap kebijakan data, meningkatkan visibilitas metadata, dan memperkuat kontrol atas alur data.

3. Migrasi Bertahap Job Eksisting: Mengeksplorasi strategi untuk migrasi *job-job ingestion* yang sudah ada (sebelum ICA Framework) ke dalam *framework* ini secara bertahap, guna memaksimalkan manfaat efisiensi dan standardisasi di seluruh operasional *data ingestion* Bank XYZ.
4. Pengembangan Komunitas Pengguna Internal: Mendorong pembentukan komunitas pengguna internal untuk berbagi praktik terbaik, tips, dan solusi atas tantangan yang mungkin timbul selama penggunaan ICA Framework, sehingga dapat mempercepat pembelajaran dan inovasi berkelanjutan.

VI. REFERENSI

- [1] Alwidian, J., Rahman, S. A., Gnaim, M., & Al-Taharwah, F. (2020). Big Data Ingestion and Preparation Tools. *Modern Applied Science*, 14(1), 12-27
- [2] Amana, M., Liu, P., & Alariqi, M. (2022). Value Creation and Capture with Big Data in Smart Phones Companies. *Sustainability*, 14(19), 12004.
- [3] Archibald, R. F., Kaspi, V. M., Ng, C.-Y., Gourgouliatos, K. N., Tsang, D., Scholz, P., Beardmore, A. P., Gehrels, N., & Kennea, J. A. (2013). An anti-glitch in a magnetar. *Nature*, 497(7451), 591–593.
- [4] Chen, H., Chiang, R. H. L., & Storey, V. C. (2012). Business Intelligence and Analytics: From Big Data to Big Impact. *MIS Quarterly*, 36(4), 1165-1188.
- [5] Cwalina, K., & Abrams, B. (2008). *Framework Design Guidelines: Conventions, Idioms, and Patterns for Reusable .NET Libraries Second Edition*. Boston: Addison-Wesley Professional.
- [6] Gupta, D., & Rani, R. (2018). A Study of Big Data Evolution and Research Challenges. *Journal of Information Science*, 45(3), 322–340.
- [7] Hasan, M. M., Hoque, A., & Le, T.-T. (2023). Big Data-Driven Banking Operations: Opportunities, Challenges, and Data Security Perspectives. *FinTech*, 2(4), 494-515.
- [8] Informatica LLC Inc. (2021). *Informatica Data Engineering Integration User Guide 10.5*. New York: Informatica LLC Inc.
- [9] Kona, S. S. (2018). Enhancing Efficiency in File-Based Data Ingestion Pipelines for Big Data. *European Journal of Advances in Engineering and Technology*, 5(8), 572-575.
- [10] Oussous, A., Benjelloun, F.-Z., Lahcen, A. A., & Belfkih, S. (2018). Big Data technologies: A Survey. *Journal of King Saud University - Computer and Information Sciences*, 30(4), 431-448.
- [11] Patgiri, R. (2018). Taxonomy of Big Data: A Survey. *arXiv preprint arXiv:180*.
- [12] Rahardja, A. S., & Hariyanto, W. (2023, February). *Pengaruh Teknologi Big Data terhadap Financial Performance dan Competitive Advantage: Studi pada Perusahaan Go Public di Indonesia Sektor Perbankan*.
- [13] Sun, N., Morris, J. G., Xu, J., Zhu, X., & Xie, M. (2014). iCARE: A framework for big data-based banking customer analytics. *IBM Journal of Research and Development*, 58(5/6), 4:1-4:9.
- [14] Veldkamp, L. (2023). Valuing Data as an Asset. *Review of Finance*, 27(5), 1545-1562.
- [15] SE. 09-DIR/DRD/04/2024 dari Bank XYZ
- [16] Nobanee, H., Dilshad, M. N., Al Dhanhani, M., Al Neyadi, M., Al Qubaisi, S., & Al Shamsi, S. (2021). Big Data Applications in the Banking Sector: A Bibliometric Analysis Approach. *Sage Open*, 11(4).