

PERAN DATA ANALYST DALAM MENINGKATKAN EFISIENSI PENGOLAHAN DATA SUMBER DAYA MANUSIA PADA PERUSAHAAN OTOMOTIF

THE ROLE OF DATA ANALYSTS IN IMPROVING THE EFFICIENCY OF HUMAN RESOURCE DATA PROCESSING IN AUTOMOTIVE COMPANIES

Nanda Kanmario¹, Jap Tji Beng^{2*}, Leny Marshalina³, Felicia June⁴, Sri Tiatri⁵, Rahmiyana Nurkholiza⁶, Vienchenzia Oeyta Dwitama Dinatha⁷

Fakultas Teknologi Informasi, Program Studi Sarjana Sistem Informasi, Universitas Tarumanagara, Jakarta^{1,2,3,4}

Fakultas Psikologi, Program Studi Psikologi, Universitas Tarumanagara, Indonesia⁵

Binus Graduate Program, Universitas Bina Nusantara, Jakarta, Indonesia⁶

Faculty of Business & Communications, Doctor of Business Administration Programme, INTI International University, Malaysia⁷

Laboratorium Kognisi Edukasi dan Inovasi Teknologi, Universitas Tarumanagara, Indonesia^{2,5,6,7}

t.jap@untar.ac.id^{2*}

ABSTRACT

The advancement of information technology encourages automotive companies to improve the efficiency of human resource data processing. The previous manual approach using Microsoft Excel was considered inefficient due to its time-consuming nature and high risk of errors. This study aims to analyze the role of Data Analysts in improving data processing efficiency through the implementation of Structured Query Language (SQL). A descriptive method with a case study approach was applied, based on direct implementation using company employee data. The results indicate that SQL enables automated data processing in a faster, more structured, and more accurate manner compared to manual methods. Therefore, the role of Data Analysts is proven essential in supporting digital transformation and data-driven decision making in the automotive industry.

Keywords: *Data Analyst, SQL, Work Efficiency, Data Processing, Automotive Industry*

ABSTRAK

Perkembangan teknologi informasi mendorong perusahaan otomotif untuk meningkatkan efisiensi dalam pengolahan data sumber daya manusia. Pengolahan data yang sebelumnya dilakukan secara manual menggunakan Microsoft Excel dinilai kurang efisien karena membutuhkan waktu yang lama dan rentan terhadap kesalahan. Penelitian ini bertujuan untuk menganalisis peran Data Analyst dalam meningkatkan efisiensi pengolahan data melalui penerapan *Structured Query Language* (SQL). Metode yang digunakan adalah deskriptif dengan pendekatan studi kasus berdasarkan implementasi langsung pada data karyawan perusahaan. Hasil penelitian menunjukkan bahwa penerapan SQL mampu mengotomatisasi proses pengolahan data secara lebih cepat, terstruktur, dan akurat dibandingkan metode manual. Dengan demikian, peran Data Analyst terbukti penting dalam mendukung transformasi digital dan pengambilan keputusan berbasis data pada perusahaan otomotif.

Kata Kunci: *Data Analyst, SQL, Efisiensi Kerja, Pengolahan Data, Industri Otomotif*

PENDAHULUAN

Perkembangan teknologi informasi pada era digital telah mendorong perusahaan untuk meningkatkan efektivitas dan efisiensi dalam pengelolaan data. Volume data yang terus meningkat menyebabkan perusahaan membutuhkan metode pengolahan data yang cepat, akurat, dan terintegrasi guna mendukung proses pengambilan keputusan Perusahaan (Japit et al., 2025). Dalam industri otomotif, data menjadi salah satu komponen penting yang

digunakan untuk mendukung kegiatan operasional perusahaan, pengelolaan kinerja karyawan, serta pengembangan sumber daya manusia (Mayasari & Agussalim, 2023). Oleh karena itu, perusahaan membutuhkan sistem pengolahan data yang mampu menghasilkan informasi secara efektif dan efisien agar dapat menunjang produktivitas kerja perusahaan.

Pada proses bisnis perusahaan, pengolahan data yang masih dilakukan

secara manual menggunakan Microsoft Excel sering kali menimbulkan berbagai kendala. Pengolahan data dalam jumlah besar membutuhkan waktu yang relatif lama, terutama pada proses penggabungan data, pencarian informasi, penyusunan laporan, serta analisis data yang memiliki hubungan kompleks antar data (Kamila & Kurniawati, 2026). Selain itu, penggunaan Microsoft Excel sangat bergantung pada ketelitian pengguna sehingga berpotensi menimbulkan kesalahan input atau *human error* dalam proses pengolahan data (Meytha & Delfin, 2023). Kondisi tersebut menyebabkan proses kerja menjadi kurang efisien dan dapat memengaruhi kualitas informasi yang digunakan dalam pengambilan keputusan perusahaan.

Seiring dengan perkembangan teknologi database, penggunaan Structured Query Language (SQL) menjadi salah satu solusi dalam meningkatkan efisiensi pengolahan data perusahaan (Aulia & Nasution, 2025). SQL mampu menangani volume data yang besar dengan kompleksitas *query* yang tinggi secara lebih terstruktur dan efisien (Aritonang et al., 2025). Melalui penggunaan *query execution*, *join table*, *filtering data*, dan *recursive query* proses pengolahan data yang sebelumnya dilakukan secara manual dapat diselesaikan dengan lebih cepat dan akurat. Selain itu, penggunaan SQL juga membantu proses integrasi data dari berbagai sumber sehingga analisis data dapat dilakukan secara lebih efektif dibandingkan metode manual menggunakan Microsoft Excel (Maruf et al., 2022).

Peran Data Analyst menjadi penting dalam membantu perusahaan menghasilkan informasi yang mendukung pengambilan keputusan berbasis data (Waloyo et al., 2024). Data Analyst tidak hanya bertanggung jawab dalam mengelola dan menganalisis data, tetapi juga membantu perusahaan dalam menyusun laporan yang lebih akurat dan efisien (Sergio et al., 2021). Penggunaan SQL dalam proses analisis data membantu perusahaan dalam

menerapkan *data-driven decision making* secara lebih efektif karena proses pengolahan data dapat dilakukan dengan lebih cepat, terstruktur, dan mampu menangani data yang kompleks (Kuntadi & Feriandi, 2023). Dengan demikian, implementasi SQL dapat membantu meningkatkan efisiensi kerja serta mendukung transformasi digital pada perusahaan otomotif.

METODE

Penelitian ini menggunakan metode kualitatif dengan pendekatan deskriptif untuk menganalisis peran Data Analyst dalam meningkatkan efisiensi pengolahan data sumber daya manusia pada perusahaan otomotif melalui penggunaan Structured Query Language (SQL) (Hermanto et al., 2023). Pendekatan ini digunakan untuk menggambarkan proses pengolahan data sebelum dan sesudah penerapan SQL dalam aktivitas pengolahan data perusahaan.

Objek dan Fokus Penelitian

Penelitian dilakukan pada perusahaan otomotif yang memiliki aktivitas pengolahan data sumber daya manusia dengan volume data yang besar dan kompleks. Fokus penelitian diarahkan pada proses pengolahan data yang sebelumnya dilakukan secara manual menggunakan Microsoft Excel dan dibandingkan dengan pengolahan data menggunakan SQL.

Dokumentasi

Dokumentasi dilakukan dengan mengumpulkan berbagai dokumen yang berkaitan dengan proses penelitian. Dokumen-dokumen tersebut meliputi struktur data karyawan yang digunakan sebagai acuan dalam pengelolaan informasi, catatan penggunaan Microsoft Excel sebagai alat bantu pengolahan data awal, *query* SQL yang diterapkan dalam proses ekstraksi dan manipulasi data, serta hasil pengolahan data yang diperoleh dari setiap tahapan analisis. Keseluruhan dokumentasi ini berfungsi sebagai bukti empiris sekaligus referensi sistematis yang

mendukung validitas dan keterlacakan proses penelitian yang dilakukan.

Studi Literatur

Studi literatur dilakukan dengan mengumpulkan referensi yang bersumber dari jurnal ilmiah, buku, serta penelitian-penelitian terdahulu yang relevan dengan topik yang dikaji. Referensi-referensi tersebut mencakup pembahasan mengenai peran Data Analyst dalam organisasi modern, konsep dan strategi efisiensi pengolahan data, penerapan SQL sebagai alat utama dalam proses ekstraksi dan manipulasi data, serta implementasi teknologi informasi dalam mendukung pengambilan keputusan berbasis data di lingkungan perusahaan. Kajian literatur ini bertujuan untuk membangun landasan teoretis yang kuat serta memberikan konteks akademis terhadap permasalahan dan pendekatan yang digunakan dalam penelitian ini.

HASIL DAN PEMBAHASAN

Pada tahap hasil dan pembahasan, penelitian ini berfokus pada proses pengolahan data organisasi karyawan untuk menghasilkan struktur data hierarki yang terintegrasi dan terorganisir. Pengolahan data dilakukan dengan studi kasus pembentukan struktur organisasi perusahaan berdasarkan hubungan atasan dan bawahan yang terdapat pada data karyawan perusahaan.

Hasil pengolahan data yang diinginkan Adalah untuk menghasilkan tabel organisasi bertingkat yang tetap mampu:

- Membentuk jalur hierarki organisasi
- Mencegah kesalahan struktur organisasi
- Pemecahan *multi-value*
- Melakukan deduplikasi data
- Menghasilkan tabel siap analisis dan visualisasi

Alur proses penelitian dapat dilihat pada Gambar.1.



Gambar 1. Flowchart Penelitian

Sumber: Desain Pribadi

Proses Pengumpulan Data

Tahap awal penelitian dilakukan melalui proses pengumpulan data yang bersumber dari database perusahaan. Data yang diperoleh digunakan untuk mendukung analisis terkait pengolahan data sumber daya manusia. Berdasarkan hasil proses pengumpulan data, diperoleh dataset Karyawan sebanyak 4.407 baris data yang terdiri atas beberapa kolom yang digunakan dalam penelitian. Informasi mengenai struktur data Karyawan yang didapatkan dapat dilihat pada Tabel 1.

Table 1. Struktur Data Karyawan

No	Nama Table	Deskripsi Data
1	Employee_No	Nomor identitas unik karyawan yang digunakan sebagai <i>primary key</i> atau kode pembeda setiap karyawan dalam sistem perusahaan.
2	Employee_Name	Nama lengkap karyawan yang tercatat dalam database Perusahaan.
3	Length_Of_Service	Lama masa kerja karyawan di perusahaan yang dihitung berdasarkan durasi waktu bekerja sejak tanggal bergabung
4	Employee_Position	Jabatan atau posisi pekerjaan yang dimiliki oleh karyawan dalam struktur organisasi Perusahaan
5	Organization_Unit	Unit organisasi karyawan bekerja sesuai dengan struktur departemen perusahaan.

6	Job_Status	Unit perusahaan atau entitas tempat karyawan bekerja. Kolom ini menunjukkan afiliasi perusahaan dari karyawan tersebut.
7	Grade	Tingkatan jabatan atau level karier karyawan dalam perusahaan, seperti staff, supervisor, manager, dan director.
8	Work_Location	Lokasi atau tempat kerja utama karyawan yang menjalankan aktivitas pekerjaannya
9	Cost_Center	(Multi-Value) Unit Perusahaan dan Divisi dari karyawan
10	Direct_Supervisor	(Celah Kesalahan) Nama atasan langsung karyawan yang bertanggung jawab melakukan pengawasan pekerjaan sehari-hari.
11	Immediate_Manager	(Celah Kesalahan) Nama manajer yang memiliki tanggung jawab manajerial terhadap karyawan dalam struktur organisasi Perusahaan (Level kedua setelah Direct_Supervisor).

Sumber: Data center Perusahaan

Proses Pengolahan Data

Setelah proses pengumpulan data selesai dilakukan, tahap berikutnya adalah proses pengolahan data. Pengolahan data dilakukan untuk mencapai hasil yang diinginkan serta dapat membuat struktur data baru yang menjadi informasi yang lebih terstruktur dan mudah dianalisis (Arsyad, 2021). Pada penelitian ini, proses pengolahan dilakukan menggunakan dua metode, yaitu pengolahan manual menggunakan Microsoft Excel dan pengolahan otomatis menggunakan *Query SQL*.

1. Menggunakan Microsoft Excel

Proses pengolahan data organisasi dilakukan secara manual menggunakan Microsoft Excel untuk menghasilkan struktur data organisasi bertingkat yang lebih terorganisir dan siap digunakan dalam proses analisis maupun visualisasi data perusahaan. Tahapan pengolahan data dilakukan secara sistematis mulai dari

normalisasi struktur data, pembersihan data, pembentukan hierarki organisasi, hingga validasi data akhir.

A. Normalisasi Struktur Data

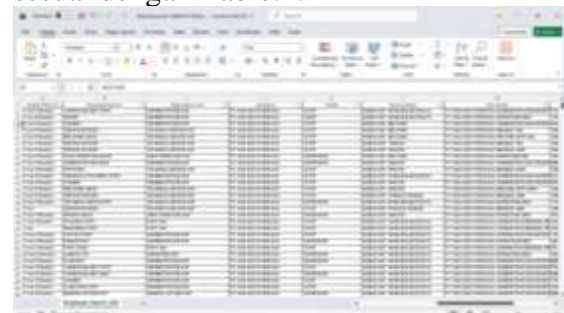
Tahap ini dilakukan untuk mencegah kesalahan relasi organisasi akibat penggunaan nama supervisor dan manager secara langsung. Oleh karena itu, dilakukan penambahan kolom ID supervisor dan manager menggunakan *Employee_No* sebagai identitas unik karyawan. Untuk mengatasi permasalahan tersebut, dilakukan penambahan kolom baru berupa, Supervisor ID dan Manager ID.

Penambahan kolom ID bertujuan untuk memastikan bahwa hubungan organisasi dibentuk menggunakan identitas unik karyawan *Employee_No* sehingga proses penggabungan data menjadi lebih akurat dan konsisten. Proses pencarian ID supervisor dan manager dilakukan menggunakan fungsi *lookup* pada Microsoft Excel, seperti *XLOOKUP* maupun *VLOOKUP*, dengan mencocokkan nama atasan terhadap nomor identitas karyawan.

Hasil dari tahap ini menghasilkan struktur relasi organisasi yang lebih stabil dan terhindar dari kesalahan penggabungan data berdasarkan nama.

1. Langkah 1

Siapkan tabel data utama karyawan, sesuai dengan Table.2.

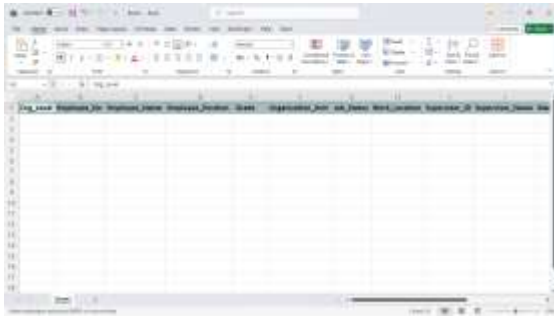


Gambar 2. Data Karyawan

Sumber: Dokumentasi Perusahaan

2. Langkah 2

Tambahkan kolom baru yaitu, Supervisor_ID, Manager_ID.

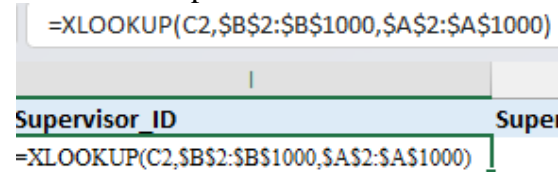


Gambar 3. Struktur Data Hirarki

Sumber: Dokumentasi Pribadi

3. Langkah 3

Langkah ketiga dalam proses pengolahan data adalah penggunaan rumus XLOOKUP pada Microsoft Excel untuk mencari ID supervisor.



Gambar 4. Rumus XLOOKUP

Sumber: Dokumentasi Pribadi

Rumus ini digunakan untuk mencocokkan nama supervisor yang terdapat pada sel C2 dengan data yang tersedia pada Kolom B (*Employee Name*), kemudian mengambil nilai yang bersesuaian dari Kolom A (*Employee No*) sebagai hasil pencarian. Dengan demikian, rumus XLOOKUP berfungsi untuk mengidentifikasi nomor karyawan (*Employee No*) yang merupakan ID unik dari supervisor yang dimaksud secara otomatis dan efisien. Penggunaan rumus ini bertujuan untuk meminimalkan kesalahan manual serta mempercepat proses pencarian data dalam jumlah besar.

4. Langkah 4

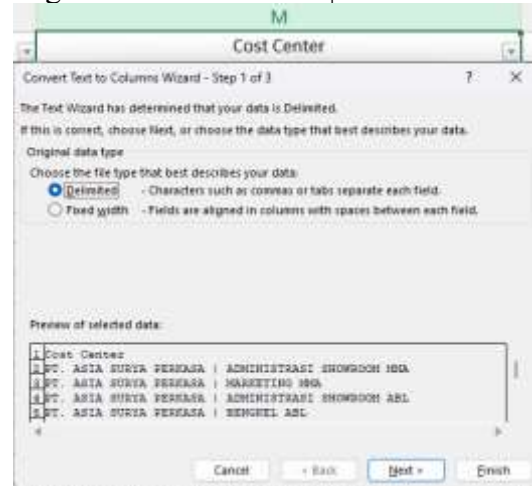
Salin formula ke seluruh baris data.

B. Pemecahan data *multi-value*

Tahap berikutnya dilakukan pemecahan data *multi-value* pada kolom Cost Center. Pada struktur data awal, kolom tersebut berisi dua informasi sekaligus dalam satu sel, yaitu nama perusahaan dan nama divisi atau unit kerja karyawan.

Kondisi tersebut menyebabkan data sulit digunakan untuk proses analisis

maupun visualisasi karena satu kolom mengandung lebih dari satu atribut informasi. Oleh karena itu, dilakukan proses pemisahan data menggunakan fitur Text to Columns pada Microsoft Excel dengan delimiter simbol '|'.



Gambar 5. Menggunakan Text to Column

Sumber: Dokumentasi Pribadi

Hasil pemecahan menghasilkan dua kolom baru yaitu, Company Name dan Division Name.

Tahap ini bertujuan untuk meningkatkan kualitas struktur data sehingga lebih mudah digunakan pada proses *filtering*, *grouping*, visualisasi, dan analisis data organisasi perusahaan.

C. Pembentukan Jalur Hierarki

Setelah struktur data dinormalisasi, dilakukan proses pembentukan jalur hierarki organisasi secara bertingkat. Tahap ini bertujuan untuk mengidentifikasi hubungan antara atasan dan bawahan berdasarkan relasi Direct Supervisor_ID dan Manager_ID.

Proses pembentukan hierarki organisasi dilakukan dengan mengelompokkan posisi-posisi yang terdapat dalam struktur organisasi perusahaan ke dalam beberapa tingkatan level yang mencerminkan alur kewenangan dan tanggung jawab secara vertikal. Pengelompokan tersebut terdiri atas empat level utama, yaitu Level 1 yang merepresentasikan posisi Direktur sebagai pemegang otoritas tertinggi dalam

organisasi, Level 2 yang merepresentasikan posisi Manager sebagai pengelola divisi atau departemen, Level 3 yang merepresentasikan posisi Supervisor sebagai pengawas operasional di lapangan, serta Level 4 yang merepresentasikan posisi Staff sebagai pelaksana tugas teknis sehari-hari.

Data kemudian disusun secara bertingkat berdasarkan hubungan antar karyawan sehingga membentuk jalur organisasi dari level tertinggi hingga level terendah. Proses identifikasi bawahan dilakukan menggunakan fitur filter dan fungsi pencarian data pada Microsoft Excel.

Hasil dari tahap ini menghasilkan tabel organisasi bertingkat yang mampu menunjukkan struktur organisasi perusahaan secara lebih jelas dan sistematis. Struktur tersebut mempermudah proses analisis organisasi, identifikasi hubungan kerja, serta mendukung kebutuhan visualisasi data perusahaan.

Namun, metode manual menggunakan Microsoft Excel masih memiliki beberapa kelemahan, terutama ketika digunakan pada data organisasi dengan jumlah besar dan struktur yang kompleks. Selain itu, proses pencarian hubungan antar karyawan menggunakan filter dan formula Microsoft Excel memiliki risiko terjadinya kesalahan input maupun kesalahan relasi apabila terdapat perubahan data organisasi secara berkala.

Kelemahan lainnya adalah keterbatasan Microsoft Excel dalam menangani struktur hierarki yang sangat dalam atau bersifat dinamis. Apabila terdapat perubahan supervisor atau perpindahan jabatan karyawan, maka proses penyusunan level organisasi perlu diperbarui kembali secara manual. Hal tersebut dapat meningkatkan kemungkinan terjadinya inkonsistensi data dan memperbesar risiko *human error* dalam proses pengolahan data organisasi.

D. Validasi Kesalahan Struktur

Tahap berikutnya dilakukan validasi terhadap struktur organisasi guna mencegah terjadinya kesalahan dalam hubungan hierarki antar karyawan. Proses validasi dilaksanakan dengan memastikan bahwa setiap karyawan memiliki relasi supervisor yang valid, tidak terdapat hubungan organisasi yang berulang (*circular hierarchy*), karyawan tidak terdaftar sebagai supervisornya sendiri, serta struktur level organisasi telah tersusun dengan benar dan konsisten.

Penggunaan *Supervisor_ID* dan *Manager_ID* pada tahap sebelumnya berkontribusi dalam meningkatkan akurasi proses validasi dibandingkan apabila menggunakan nama karyawan sebagai acuan. Hal ini dikarenakan penggunaan identitas unik karyawan berupa *Employee_No* mampu meminimalkan risiko kesalahan yang disebabkan oleh kesamaan nama, perbedaan penulisan, maupun perubahan data karyawan yang terjadi sewaktu-waktu. Selain itu, dilakukan pula pemeriksaan secara manual dengan memanfaatkan formula logika pada Microsoft Excel untuk mendeteksi kemungkinan adanya kesalahan dalam struktur organisasi, seperti hubungan supervisor yang tidak sesuai maupun relasi organisasi yang bersifat melingkar (*circular*).

Tahap validasi ini bertujuan untuk menjaga konsistensi dan validitas struktur organisasi yang dihasilkan, sehingga data yang diperoleh tetap akurat dan dapat digunakan secara optimal pada proses analisis maupun visualisasi data perusahaan. Sebagai ilustrasi, Gambar 6 menyajikan contoh rumus yang digunakan dalam proses validasi tersebut, di mana rumus tersebut secara spesifik memeriksa apakah seorang karyawan terdaftar sebagai supervisor bagi dirinya sendiri.

=IF(A2=H2,"ERROR","VALID")

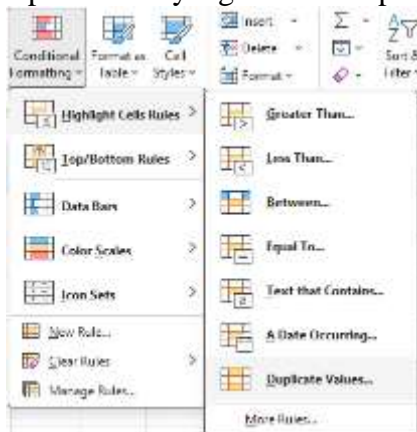
Gambar 6. Rumus Validasi

Sumber: Dokumentasi Pribadi

Namun demikian, proses validasi manual menggunakan Microsoft Excel masih memiliki beberapa keterbatasan. Pemeriksaan dilakukan dengan menggunakan rumus yang terbatas, yang berarti jika menginginkan hasil validasi yang baik harus menggunakan banyak rumus validasi. Selain itu, proses validasi manual memiliki risiko *human error*, seperti kesalahan penggunaan formula. Microsoft Excel juga memiliki keterbatasan dalam melakukan validasi hierarki secara dinamis apabila terjadi perubahan struktur organisasi secara berkala, sehingga proses pengecekan dan pembaruan data perlu dilakukan kembali secara manual untuk menjaga konsistensi struktur organisasi perusahaan.

E. Deduplikasi Data Karyawan

Tahap selanjutnya adalah pelaksanaan deduplikasi data yang bertujuan untuk menghapus entri data karyawan yang muncul lebih dari satu kali dalam dataset. Proses deduplikasi dilakukan dengan memanfaatkan fitur *Remove Duplicates* pada Microsoft Excel, dengan menjadikan *Employee_No* sebagai identitas utama yang digunakan sebagai acuan dalam pendeteksian dan penghapusan data yang bersifat duplikat.



Gambar 7. Menggunakan fitur Text to Column

Sumber: Dokumentasi Pribadi

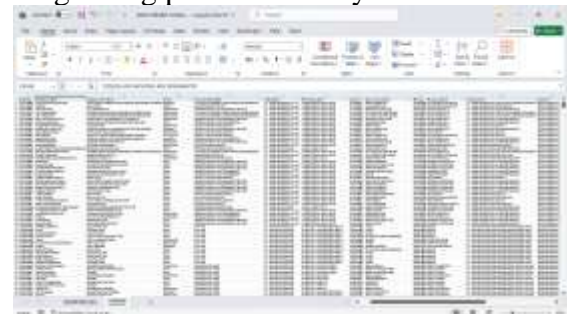
Tahap ini memiliki beberapa tujuan utama, yaitu mengurangi redundansi data yang dapat mengganggu keakuratan analisis, meningkatkan konsistensi dataset secara keseluruhan, memastikan bahwa

setiap karyawan hanya memiliki satu representasi data yang valid dalam sistem, serta mempermudah proses analisis dan visualisasi data pada tahap selanjutnya.

Hasil dari proses deduplikasi ini adalah terbentuknya dataset organisasi yang lebih bersih, terstruktur, dan siap digunakan untuk keperluan pengolahan data lebih lanjut. Penggunaan *Employee_No* sebagai kunci identifikasi utama dalam proses ini dinilai lebih andal dibandingkan penggunaan nama karyawan, mengingat nomor karyawan bersifat unik dan tidak mengalami perubahan meskipun terdapat kesamaan nama antar individu dalam organisasi.

F. Hasil Pengolahan Manual

Hasil akhir dari proses pengolahan data menggunakan Microsoft Excel adalah terbentuknya tabel organisasi bertingkat yang telah melalui proses normalisasi dan validasi data. Tabel tersebut memuat informasi hubungan hierarki organisasi berdasarkan *Supervisor_ID* dan *Manager_ID*, sehingga relasi antar karyawan menjadi lebih akurat dan tidak bergantung pada nama karyawan.



Gambar 8. Hasil Akhir Menggunakan Metode manual Excel

Sumber: Dokumentasi Pribadi

Selain itu, data *multi-value* pada kolom *Cost Center* berhasil dipisahkan menjadi informasi perusahaan dan divisi yang berbeda sehingga struktur data menjadi lebih rapi dan mudah digunakan dalam proses analisis. Proses deduplikasi juga menghasilkan dataset yang lebih konsisten dengan memastikan setiap karyawan hanya memiliki satu representasi data.

Hasil pengolahan tersebut menghasilkan tabel organisasi yang siap digunakan untuk kebutuhan analisis data, *dashboard* organisasi, visualisasi struktur perusahaan, maupun pengolahan lanjutan.

G. Kesimpulan

Meskipun Microsoft Excel dapat digunakan untuk membangun struktur organisasi secara manual, metode ini memiliki beberapa kelemahan, seperti proses pengolahan yang memerlukan waktu lebih lama, tingginya risiko *human error*, serta keterbatasan dalam menangani struktur hierarki yang kompleks. Selain itu, proses validasi dan pembaruan data masih dilakukan secara manual sehingga kurang efisien apabila digunakan pada dataset dengan jumlah data yang besar. Performa Excel juga cenderung menurun ketika digunakan untuk pengolahan data dalam skala besar dan memiliki banyak formula maupun relasi data.

2. Menggunakan SQL

Pengolahan data organisasi menggunakan SQL Server dilakukan untuk menghasilkan struktur data organisasi yang lebih terintegrasi, otomatis, dan siap digunakan dalam proses analisis maupun visualisasi data perusahaan. Berbeda dengan metode manual menggunakan Microsoft Excel, pengolahan menggunakan SQL memungkinkan proses transformasi data dilakukan secara lebih cepat, konsisten, dan efisien melalui penggunaan *query* terstruktur.

Pada penelitian ini, proses pengolahan data dilakukan langsung pada database perusahaan menggunakan SQL Server dengan memanfaatkan *Recursive Common Table Expression (Recursive CTE)* untuk membangun struktur organisasi bertingkat berdasarkan hubungan supervisor dan manager pada data karyawan. Selain pembentukan hierarki organisasi, proses pengolahan juga mencakup validasi data, pembersihan data, pemecahan data *multi-value*, deduplikasi

data, hingga pembentukan tabel final organisasi.

A. Normalisasi Struktur Data

Pada tahap normalisasi struktur data, dilakukan penambahan atribut baru berupa Supervisor_ID dan Manager_ID pada tabel karyawan untuk memperbaiki struktur relasi organisasi. Sebelumnya, hubungan antara karyawan dan atasannya masih menggunakan nama supervisor maupun manager, sehingga berpotensi menimbulkan kesalahan relasi akibat adanya nama yang sama, perbedaan penulisan nama, maupun perubahan data karyawan pada sistem perusahaan.

```
--TAMBAHIN ID KE IMMEDIATE MANAGER SAMA DIRECT SUPERVISOR DULU
ALTER TABLE [EPMS NEW].[dbo].[Data Employee]
ADD Supervisor_ID VARCHAR(50),
    Manager_ID VARCHAR(50);
```

Gambar 9. Query Penambahan Kolom

Sumber: Dokumentasi Pribadi

Perintah tersebut digunakan untuk menambahkan, Supervisor_ID dan Manager_ID ke dalam struktur tabel karyawan. Kedua kolom tersebut digunakan untuk menyimpan nomor identitas unik Employee_No dari supervisor dan manager setiap karyawan.

Setelah kolom baru berhasil ditambahkan, dilakukan proses pengisian data Supervisor_ID menggunakan perintah *UPDATE* dan *LEFT JOIN*.

```
-- Update Supervisor_ID
UPDATE emp
SET Supervisor_ID = sup.Employee_No
FROM [EPMS NEW].[dbo].[Data Employee] emp
LEFT JOIN [EPMS NEW].[dbo].[Data Employee] sup
ON emp.Direct_Supervisor = sup.Employee_Name;

-- Update Manager_ID
UPDATE emp
SET Manager_ID = mgr.Employee_No
FROM [EPMS NEW].[dbo].[Data Employee] emp
LEFT JOIN [EPMS NEW].[dbo].[Data Employee] mgr
ON emp.Immediate_Manager = mgr.Employee_Name;

UPDATE emp
SET
    Supervisor_ID = sup.Employee_No,
    Manager_ID = mgr.Employee_No
FROM [EPMS NEW].[dbo].[Data Employee] emp
LEFT JOIN [EPMS NEW].[dbo].[Data Employee] sup
ON emp.Direct_Supervisor = sup.Employee_Name
LEFT JOIN [EPMS NEW].[dbo].[Data Employee] mgr
ON emp.Immediate_Manager = mgr.Employee_Name;
```

Gambar 10. Query Update Table

Sumber: Dokumentasi Pribadi

Query tersebut digunakan untuk memperbarui kedua kolom relasi organisasi sekaligus dalam satu proses. Penggunaan *LEFT JOIN* memungkinkan data tetap

diproses meskipun terdapat supervisor atau manager yang bernilai kosong (*NULL*).

Melalui proses normalisasi ini, hubungan organisasi tidak lagi bergantung pada nama karyawan, melainkan menggunakan identitas unik berupa *Employee_No*. Hal tersebut membantu meningkatkan konsistensi relasi data, mempermudah proses pembentukan hierarki organisasi, serta mengurangi risiko kesalahan relasi organisasi pada tahap pengolahan data selanjutnya.

B. Pembuatan Query lengkap

Proses pengolahan data organisasi dilakukan menggunakan pendekatan *Recursive Common Table Expression (Recursive CTE)* untuk membangun struktur hierarki organisasi secara otomatis. *Query* lengkap ini dibagi menjadi beberapa bagian utama sebagai berikut:

1. Root Hierarchy

```
WITH OrgHierarchy AS
(
    -- ROOT: Director
    SELECT
        Employee_No,
        Employee_Name,
        Supervisor_ID,
        Manager_ID,
        Employee_Position,
        Grade,
        Organization_Unit,
        Job_Status,
        Work_Location,
        @ AS Org_Level,
        CAST(Employee_No AS VARCHAR(100)) AS Hierarchy_Path
    FROM [PMS_MW].[dbo].[Data_Employee]
    WHERE Grade = 'DIRECTOR'
```

Gambar 11. Root Hierarchy

Sumber: Dokumentasi Pribadi

Bagian ini merupakan titik awal (*anchor*) dari proses rekursif. *Query* memilih karyawan dengan *grade* DIRECTOR sebagai level tertinggi organisasi (Level 0) dan membentuk atribut *Hierarchy_Path* yang digunakan untuk menyimpan jalur hierarki organisasi secara berurutan.

2. Recursive Hierarchy

```
UNION ALL
-- Recursive Hierarchy
SELECT
    e.Employee_No,
    e.Employee_Name,
    e.Supervisor_ID,
    e.Manager_ID,
    e.Employee_Position,
    e.Grade,
    e.Organization_Unit,
    e.Job_Status,
    e.Work_Location,
    eh.Org_Level + 1,
    CAST(e.Hierarchy_Path + '-' + CAST(e.Employee_No AS VARCHAR(10)) AS VARCHAR(100))
FROM [PMS_MW].[dbo].[Data_Employee] e
INNER JOIN OrgHierarchy eh
ON
    e.Supervisor_ID = eh.Employee_No
OR e.Manager_ID = eh.Employee_No
OR e.Manager_ID IS NULL AND e.Supervisor_ID = eh.Employee_No
```

Gambar 12. Query Recursive Hierarchy

Sumber: Dokumentasi Pribadi

Bagian ini menelusuri hubungan atasan dan bawahan secara bertingkat berdasarkan *Supervisor_ID* dan *Manager_ID*. Setiap iterasi rekursif menambahkan satu level organisasi (*Org_Level + 1*) dan memperpanjang jalur hierarki (*Hierarchy_Path*) dengan menambahkan *Employee_No* karyawan berikutnya.

3. Pencegahan Circular Hierarchy

```
WHERE CHARINDEX('-',@ + CAST(e.Employee_No AS VARCHAR(50)) + '-',
    '-' + eh.Hierarchy_Path + '-') = 0
```

Gambar 13. Query Pencegahan Circular Hierarchy

Sumber: Dokumentasi Pribadi

Bagian ini mencegah terjadinya perulangan hubungan organisasi (*circular hierarchy*) menggunakan fungsi CHARINDEX. Fungsi ini memastikan bahwa *Employee_No* seorang karyawan tidak muncul kembali pada jalur hierarki yang sama, sehingga proses rekursif tidak berjalan tanpa batas.

4. Deduplikasi Data

```
DedupHierarchy AS
(
    SELECT *,
        ROW_NUMBER() OVER (
            PARTITION BY Employee_No
            ORDER BY Org_Level
        ) AS rn
    FROM OrgHierarchy
)
```

Gambar 14. Query Deduplikasi Data

Sumber: Dokumentasi Pribadi

Bagian ini melakukan deduplikasi data menggunakan fungsi ROW_NUMBER dengan partisi berdasarkan *Employee_No*. Setiap karyawan yang muncul lebih dari satu kali dalam hasil rekursif akan diurutkan berdasarkan *Org_Level* terendah, sehingga hanya satu representasi data yang dipertahankan pada hasil akhir.

5. Pemecahan Hierarchy Path menjadi Level Organisasi

```

CROSS APPLY
(
  SELECT
    ROW_NUMBER() OVER (ORDER BY (SELECT NULL)) AS lvl,
    x.value('varchar(255) AS emp_no'
    FROM (
      SELECT CAST('00' + REPLACE(oh.hierarchy_path, '-', '000000') AS XML) AS xml
    ) x(xml)
    CROSS APPLY xml.nodes('/x') bc)
) s

```

Gambar 15. Query Pemecah Hierarchy Data

Sumber: Dokumentasi Pribadi

Bagian ini memecah atribut *Hierarchy_Path* menjadi beberapa level organisasi menggunakan *CROSS APPLY* dan XML parsing. Jalur hierarki yang disimpan dalam format Employee_No1->Employee_No2->Employee_No3 dipecah menjadi kolom-kolom Level_1 hingga Level_7 yang merepresentasikan nama karyawan pada setiap tingkatan organisasi.

6. Penggabungan Data (*LEFT JOIN*)

```

LEFT JOIN [EPMS NEM].[dbo].[Data Employee] emp
  ON emp.Employee_No = s.emp_no

LEFT JOIN [EPMS NEM].[dbo].[Data Employee] emp_main
  ON emp_main.Employee_No = oh.Employee_No

LEFT JOIN [EPMS NEM].[dbo].[Data Employee] sup
  ON sup.Employee_No = oh.Supervisor_ID --  tambahan

LEFT JOIN [EPMS NEM].[dbo].[Data Employee] mgr
  ON mgr.Employee_No = oh.Manager_ID --  tambahan
WHERE oh.rn = 1

```

Gambar 16. Query JOIN Data

Sumber: Dokumentasi Pribadi

Bagian ini melakukan beberapa proses *LEFT JOIN* terhadap tabel *Data Employee* untuk melengkapi informasi organisasi, meliputi nama supervisor, nama manager, *cost center*, posisi karyawan, unit organisasi, status pekerjaan, dan lokasi kerja. Filter *WHERE oh.rn = 1* diterapkan untuk memastikan hanya data karyawan tanpa duplikasi yang diproses pada tahap ini.

7. Pembentukan Tabel Final

```

MAX(CASE WHEN lvl = 1 THEN emp.Employee_Name END) AS Level_1,
MAX(CASE WHEN lvl = 2 THEN emp.Employee_Name END) AS Level_2,
MAX(CASE WHEN lvl = 3 THEN emp.Employee_Name END) AS Level_3,
MAX(CASE WHEN lvl = 4 THEN emp.Employee_Name END) AS Level_4,
MAX(CASE WHEN lvl = 5 THEN emp.Employee_Name END) AS Level_5,
MAX(CASE WHEN lvl = 6 THEN emp.Employee_Name END) AS Level_6,
MAX(CASE WHEN lvl = 7 THEN emp.Employee_Name END) AS Level_7
INTO [EPMS NEM].[dbo].[Org_Hierarchy_Result]

```

Gambar 17. Query Pembentukan Table

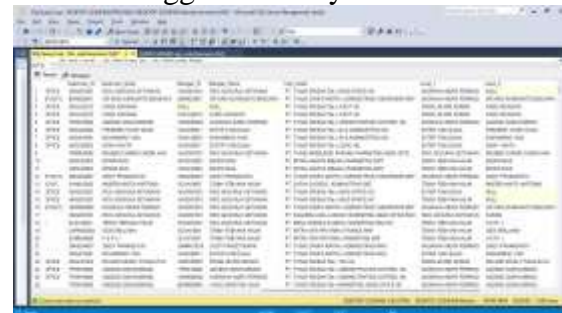
Sumber: Dokumentasi Pribadi

Bagian akhir *query* membentuk kolom Level_1 hingga Level_7 menggunakan fungsi agregasi *MAX* dengan kondisi *CASE*, yang merepresentasikan nama

karyawan pada setiap tingkatan hierarki organisasi. Seluruh hasil pengolahan kemudian disimpan ke dalam tabel baru bernama *Org_Hierarchy_Result* sebagai tabel final organisasi. Parameter *MAXRECURSION 100* diterapkan untuk membatasi kedalaman rekursif maksimal hingga 100 level guna mencegah perulangan tak terbatas.

C. Hasil Pengolahan SQL

Berdasarkan proses pengolahan data yang telah dilakukan menggunakan SQL Server, berhasil dibentuk tabel organisasi bertingkat yang memuat hubungan hierarki antar karyawan secara lebih terstruktur dan terintegrasi. *Query* yang dibangun menggunakan pendekatan *Recursive Common Table Expression (Recursive CTE)* mampu mengidentifikasi hubungan antara atasan dan bawahan berdasarkan *Supervisor_ID* dan *Manager_ID*, sehingga struktur organisasi perusahaan dapat tersusun secara otomatis mulai dari level direktur hingga level karyawan terbawah.



Gambar 18. Hasil Akhir Menggunakan Metode SQL

Sumber: Dokumentasi Pribadi

Hasil pengolahan data menghasilkan sebuah tabel yang dinamakan *Org_Hierarchy_Result*, yang memuat informasi utama organisasi secara komprehensif. Tabel tersebut mencakup berbagai atribut penting, meliputi identitas karyawan, nama supervisor, nama manager, posisi karyawan, unit organisasi, lokasi kerja, hingga struktur level organisasi bertingkat yang terdiri dari Level-1 hingga Level-7. Di samping itu, proses pengolahan data juga berhasil membentuk jalur hierarki organisasi (*Hierarchy_Path*) yang digunakan untuk

merepresentasikan hubungan antar posisi dalam organisasi secara berurutan, mulai dari level tertinggi hingga level terbawah dalam struktur organisasi perusahaan.

Dalam proses pengolahan tersebut, *query* yang diterapkan mampu menjalankan beberapa proses pengolahan data secara simultan dalam satu eksekusi, sehingga meningkatkan efisiensi dan efektivitas pengolahan secara keseluruhan. Proses-proses tersebut meliputi pembentukan hierarki organisasi, validasi struktur organisasi, pencegahan *circular hierarchy*, deduplikasi data karyawan, penggabungan data menggunakan metode *JOIN*, serta pembentukan tabel final organisasi sebagai output akhir dari keseluruhan proses pengolahan data yang dilakukan.

Perbandingan Hasil

Berdasarkan hasil pengolahan data organisasi yang telah dilakukan, penggunaan SQL Server terbukti lebih efektif dan efisien dibandingkan metode manual menggunakan Microsoft Excel (Nuriev et al., 2023). SQL mampu mengotomatisasi proses pengolahan data organisasi yang kompleks melalui penggunaan *query* terstruktur dan *Recursive CTE*, sehingga proses pembentukan hierarki organisasi dapat dilakukan secara lebih cepat, akurat, dan konsisten (Carvalho et al., 2026).

Selain itu, SQL memiliki kemampuan untuk menjalankan berbagai proses pengolahan data dalam satu *query* secara bersamaan, seperti proses *data cleansing*, validasi relasi organisasi, deduplikasi data, pembentukan hierarki organisasi, hingga penggabungan data antar table (Budi & Syahrial, 2023). Kemampuan tersebut membantu mengurangi proses manual, meminimalkan risiko *human error*, serta meningkatkan efisiensi pengolahan data perusahaan.

Dengan demikian, penggunaan SQL Server menjadi solusi yang lebih optimal dalam pengolahan data organisasi perusahaan, terutama untuk dataset berskala besar dan struktur organisasi yang

kompleks, karena mampu menghasilkan data yang lebih terintegrasi, terstruktur, dan siap digunakan untuk kebutuhan analisis maupun visualisasi data perusahaan (Firmansyah et al., 2025).

SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, peran Data Analyst terbukti berkontribusi signifikan dalam meningkatkan efisiensi pengolahan data sumber daya manusia pada perusahaan otomotif. Melalui penerapan *Structured Query Language* (SQL), proses pengolahan data yang sebelumnya dilakukan secara manual dapat dijalankan secara lebih terstruktur, cepat, dan akurat. Penggunaan SQL memungkinkan berbagai proses seperti *filtering* data, penggabungan tabel, validasi struktur organisasi, hingga pembentukan laporan dilakukan secara otomatis dalam satu eksekusi, sehingga mampu meminimalkan risiko *human error* serta mengatasi keterbatasan dalam menangani data bervolume besar dengan relasi yang kompleks.

Di samping peran teknisnya, Data Analyst juga berkontribusi dalam mendukung pengambilan keputusan berbasis data (*data-driven decision making*) melalui penyediaan informasi yang lebih akurat, relevan, dan mudah dipahami oleh perusahaan. Dengan demikian, kombinasi antara kemampuan Data Analyst dan pemanfaatan SQL terbukti mampu meningkatkan kualitas pengelolaan data sumber daya manusia secara efektif, efisien, dan terintegrasi, sekaligus mendukung transformasi digital dan peningkatan efisiensi operasional perusahaan otomotif secara menyeluruh.

DAFTAR PUSTAKA

Aritonang, G. R. M. L. B., Hasan, M. A., Ridwan, M. K., Iman, M. N., Silalahi, R. C. P., & Sipayung, R. S. (2025). Optimasi Query SQL Server dengan Teknik Indexing dan Performance Monitoring. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(2), 3094-

3099.
<https://doi.org/10.36040/jati.v9i2.13179>
- Arsyad, Z. (2021). Analisis Dynamic ETL Incremental Load untuk Data Integration Datawarehouse. *INTERNAL (Information System Journal)*, 4(2), 102–112.
<https://doi.org/10.32627/internal.v4i2.260>
- Aulia, H. I., & Nasution, M. I. P. (2025). Implementasi Database Management System (DBMS) dalam Meningkatkan Efisiensi Operasional Perusahaan. *Epsilon Journal of Management*, 3(1), 17–26.
<https://doi.org/10.62951/epsilon.v3i1.125>
- Budi, D. S., & Syahrial, H. (2023). Pengoptimalan Performa Database Pada Proses Transformasi Data Pada SQL Server. *Technomedia Journal*, 8(3), 78–90.
<https://doi.org/10.33050/tmj.v8i3.2167>
- Carvalho, G., Rasteiro, D., Dorgham, N., Cabral, B., Bernardino, J., & Pereira, V. (2026). Automating the generation of database artifacts: From ER+ to SQL. *Information Systems*, 141, 102741.
<https://doi.org/10.1016/j.is.2026.102741>
- Firmansyah, E., Firdaus, H., & Samidi, S. (2025). Optimasi Performa Query Subsidi Debitur dengan Index and Table Partition, Subquery and Indexing, dan Parallel Query Execution. *Jurnal Pendidikan Dan Teknologi Indonesia*, 5(6), 1769–1785.
<https://doi.org/10.52436/1.jpti.824>
- Hermanto, E. A., Gemintang, S. A., Ariansyah, R., & Giovanny, M. A. (2023). Analisis Perbandingan Penerapan Business Intelligence di Indonesia Menggunakan Metode Systematic Literature Review. *Djtechno: Jurnal Teknologi Informasi*, 4(2), 344–354.
<https://doi.org/10.46576/djtechno.v4i2.3412>
- Kamila, R. S., & Kurniawati, Y. K. (2026). Analisis Perbandingan Penggunaan Microsoft Excel dan Accurate Terhadap Efisiensi dan Akurasi Penyusunan Laporan Keuangan (Studi Kasus PT Fadsan Indonesia Group). *Bilancia: Jurnal Ilmiah Akuntansi*, 10(1), 70–78.
<https://doi.org/10.35145/bilancia.v10i1.5799>
- Kuntadi, A., & Feriandi, Y. (2023). Penggunaan SQL Server dalam Pengolahan Data Warehouse yang Praktis dan Berkelanjutan. *Jurnal Algoritma*, 20(2), 457–468.
<https://doi.org/10.33364/algoritma/v.20-2.1522>
- Maruf, A. Al, Paul, R., Imam, M. H., & Babar, Z. (2022). A Systematic Review of The Role of SQL and Excel in Data-Driven Business Decision-Making for Aspiring Analysts. *American Journal of Scholarly Research and Innovation*, 01(01), 249–269.
<https://doi.org/10.63125/n142cg62>
- Mayasari, E., & Agussalim, A. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, 3(3), 171–187.
<https://doi.org/10.55606/juisik.v3i3.680>
- Meytha, S., & Delfin, M, V. O. (2023). Perbandingan Penginputan Transaksi Secara Manual Menggunakan Accurate pada Laporan Keuangan Input Transaksi Manual dengan Accurate di Laporan Keuangan. *El-Mujtama: Jurnal Pengabdian Masyarakat*, 4(2), 1426–1434.
<https://doi.org/10.47467/elmujtama.v4i2.5412>
- Nuriev, M., Zaripova, R., Potapov, A., & Kuznetsov, M. (2023). Achieving new SQL query performance levels through parallel execution in SQL Server. *E3S Web of Conferences*, 460,

04005.

<https://doi.org/10.1051/e3sconf/202346004005>

- Sergio, M., Franciosi, C., & Iannone, R. (2021). A model for the economic assessment of disassembly-line integration in traditional manufacturing processes. *Procedia Computer Science*, 180, 308–317. <https://doi.org/10.1016/j.procs.2021.01.168>
- Japit, S., Risyani, Y., Bombongan, C., Selamat, T., & Yuliana, Y. (2025). Pemanfaatan Pandas, NumPy, Spark, dan TableauPower BI dalam Pengolahan dan Visualisasi Big Data untuk Mendukung Pengambilan Keputusan. *Jurnal Minfo Polgan*, 14(1), 452–456. <https://doi.org/10.33395/jmp.v14i1.14762>
- Waloyo, W., Suwaji, R., Sholihin, A., Hepiyanto, H., & Sabella, R. I. (2024). Revolusi Manajemen SDM: Menerapkan Big Data untuk Efisiensi dan Strategi HR di Era Data-Driven. *EKOMA: Jurnal Ekonomi, Manajemen, Akuntansi*, 4(1), 2185–2192. <https://doi.org/10.56799/ekoma.v4i1.6136>