

## **BAB III**

### **PELAKSANAAN KERJA**

#### **3.1 Kedudukan dan Koordinasi**

Dalam pelaksanaan program magang, posisi sebagai Intern Sistem Informasi Manajemen (MIS) yang berada di bawah Divisi Teknologi Informasi (IT) PT Trans Retail Indonesia. Divisi IT adalah bagian dari kelompok Pendukung, yang memiliki peran penting dalam mendukung semua aktivitas operasional dan bisnis perusahaan dengan menggunakan sistem informasi dan teknologi. Posisi ini dalam struktur organisasi tidak bersifat manajerial, tetapi tetap memiliki peran penting dalam mendukung pengembangan dan pelaksanaan sistem informasi internal perusahaan.

Secara struktural, berada di bawah koordinasi Manajer Aplikasi TI dan Sistem Informasi Manajemen (MIS), yakni Bapak Idham Fahrian, yang memiliki tanggung jawab sebagai pemimpin divisi dalam mengelola pengembangan, pemeliharaan, dan pengawasan atas sistem informasi serta aplikasi internal perusahaan. Semua tugas utama yang diberikan, termasuk pengembangan website Data Query Builder, berada dalam area tanggung jawab dan persetujuan kepala divisi. Dalam proses pengambilan keputusan strategis yang berkaitan dengan kebutuhan sistem serta penyesuaian fitur, harus dipatuhi terutama petunjuk dan kebijakan yang telah ditentukan oleh Manajer Aplikasi TI dan MIS.

Dalam kegiatan teknis sehari-hari, koordinasi dilakukan secara langsung dengan mentor, yaitu Bapak Axel, yang berfungsi sebagai pembimbing magang. Insinyur TI memiliki tanggung jawab untuk memberikan petunjuk teknis mengenai penerapan sistem, standar dalam pengembangan aplikasi, serta mendukung dalam mengatasi masalah teknis yang timbul selama proses pembuatan situs web. Koordinasi dengan pembimbing dilaksanakan secara intensif, baik melalui perbincangan langsung, pertemuan internal, maupun komunikasi tidak resmi, agar sistem yang dibangun sesuai dengan kebutuhan operasional serta standar teknis perusahaan.

### **3.2 Tugas Kerja Magang**

Setelah telah menyelesaikan program magang selama 640 jam kerja yang telah disesuaikan dengan aturan dalam kurikulum Program Studi Sistem Informasi. Selama masa magang ini, sebagai Intern Sistem Informasi Manajemen (MIS) yang memiliki tugas utama dalam melakukan pengawasan sistem serta melaksanakan kegiatan operasional sehari-hari yang berhubungan dengan sistem informasi perusahaan. Tugas inti ini dilaksanakan secara berkala, terutama pada bulan-bulan awal sampai pertengahan periode magang

Dibulan terakhir program magang, Mentor memberikan tanggung jawab baru berupa aktivitas pemantauan terhadap sistem Transmart dan Bank Mega. Tujuan dari kegiatan ini adalah untuk memastikan bahwa kedua sistem tersebut berfungsi dengan baik dan tidak menghadapi masalah operasional. Kegiatan pemantauan dan pelaksanaan harian biasanya dapat diselesaikan hingga sekitar pukul 12. 00 siang setiap harinya dan akan dilanjutkan di jam 13.00 jika ada tugas tambahan lainnya tentunya tugas tersebut akan tetap berada dalam jam kerja magang dan berada di bawah pengawasan serta koordinasi tim IT.

Agar memaksimalkan penggunaan waktu magang yang masih ada, dari mentor yang ditempatkan di divisi IT Engineer memberikan tugas tambahan, Bapak Axel, yang berupa pengembangan aplikasi berbasis web yang dinamakan Data Query Builder. Aplikasi ini dibuat untuk membantu pengguna dalam membuat dan menjalankan query pengambilan data secara mandiri tanpa perlu menulis query secara manual. Sebelum aplikasi ini dikembangkan, tim IT melakukan proses pengambilan data dengan cara memasukkan query secara manual sesuai dengan kebutuhan setiap pengguna. Dengan adanya Data Query Builder, diharapkan pengguna internal dapat menarik data secara mandiri melalui antarmuka web, sehingga dapat mengurangi ketergantungan terhadap tim TI serta meningkatkan efisiensi dan

kecepatan dalam memenuhi kebutuhan data.

Dalam pelaksanaan proyek Data Query Builder, intern berperan dalam analisis kebutuhan pengguna internal, perancangan alur sistem, dan penerapan fitur utama aplikasi sesuai dengan petunjuk supervisor serta standar tim MIS. Proyek ini dilaksanakan secara bertahap dan terintegrasi dengan kegiatan magang harian, sehingga intern tetap melaksanakan tugas utamanya sebagai Intern MIS sekaligus mengerjakan proyek sistem informasi sebagai tanggung jawab tambahan. Penjelasan yang lebih mendetail tentang langkah-langkah pelaksanaan proyek serta aktivitas yang dilakukan selama periode magang akan disampaikan pada poin-poin yang dibahas dalam bagian berikut :

Table 2 Detail Pekerjaan yang Dilakukan Selama Magang

No	Kegiatan	Mulai	Berakhir
1	Orientasi dan Pengenalan Lingkungan Kerja	20 Agustus 2025	20 Agustus 2025
2	Melakukan Monitoring dan Daily Execute	21 Agustus 2025	20 Desember 2025
3	Eksplorasi Kemampuan Secara Mandiri	20 Agustus 2025	31 Agustus 2025
4	Melakukan Penarikan Data sesuai kebutuhan User	21 Agustus 2025	24 November 2025
5	Pengenalan Proyek	29 Agustus 2025	29 Agustus 2025
6	Penyusunan FSD	1 September 2025	5 September 2025
7	Pengerjaan Proyek	8 Sepetember 20225	20 Desember 2025
8	Melakukan <i>Monitoring Dashboard</i> Transmart Bank Mega	6 November 2025	20 Desember 2025

### **3.3 Uraian Pelaksanaan**

#### **3.3.1 Proses Pelaksanaan**

##### **3.3.1.1 Orientasi dan Pengenalan Lingkungan Kerja**

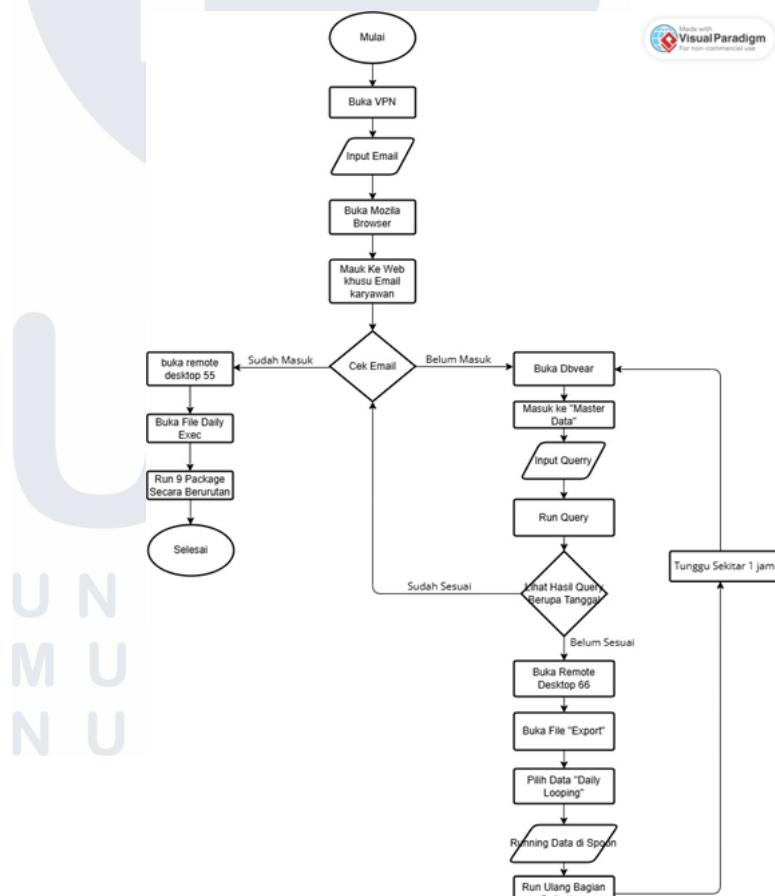
Pada awal pelaksanaan kerja magang, kegiatan orientasi dan pengenalan lingkungan kerja di PT Trans Retail Indonesia. Tujuan dari kegiatan ini adalah untuk membantu dalam memahami struktur organisasi, budaya kerja, serta sistem operasional yang berlaku di perusahaan, terutama di Divisi Information Technology (IT). Dalam proses orientasi tersebut, diperkenalkan juga dengan peran dan fungsi divisi IT sebagai unit pendukung utama dalam menjaga kelancaran sistem informasi dan operasional perusahaan ritel.

Pada tahap pengenalan lingkungan kerja, diperkenalkan secara langsung dengan sub-divisi IT Application and Management Information System (MIS), yang merupakan unit tempat melakukan kegiatan magang. Dijelaskan mengenai peran IT Application and MIS Manager sebagai penanggung jawab sistem aplikasi dan informasi manajemen, serta peran IT Engineer sebagai pembimbing teknis yang memberikan panduan dalam menjalankan tugas sehari-hari. Dengan pengenalan ini, harapannya alur kerja, pola komunikasi, serta batasan tanggung jawab yang harus dijalani selama masa magang dapat dipahami. Selain itu, penjelasan mengenai tugas pokok sebagai Intern MIS juga diberikan pada masa ini, yang mencakup kegiatan pemantauan sistem, pelaksanaan tugas harian, serta pengambilan dan pemeriksaan data menggunakan alat yang digunakan oleh perusahaan.

Pada tahap ini, pengenalan dengan sistem dan perangkat lunak pendukung dalam operasional, seperti akses jaringan internal, penggunaan Virtual Private Network (VPN), serta tools database untuk mendukung kegiatan pemantauan data. Proses orientasi juga mencakup pemahaman

mengenai alur kerja harian serta standar operasional prosedur (SOP) yang berlaku di lingkungan kerja. Proses pengamatan dan mempelajari bagaimana proses pemantauan sistem dilakukan juga sudah mulai dilakukan, bagaimana data diproses dan diverifikasi, serta bagaimana hasil pemantauan dilaporkan kepada pihak yang berkepentingan. Dengan adanya tahapan orientasi dan pengenalan lingkungan kerja, proses adaptasi berjalan secara perlahan terhadap ritme kerja profesional dan siap menjalankan tugas sebagai Intern MIS sesuai kebutuhan operasional perusahaan.

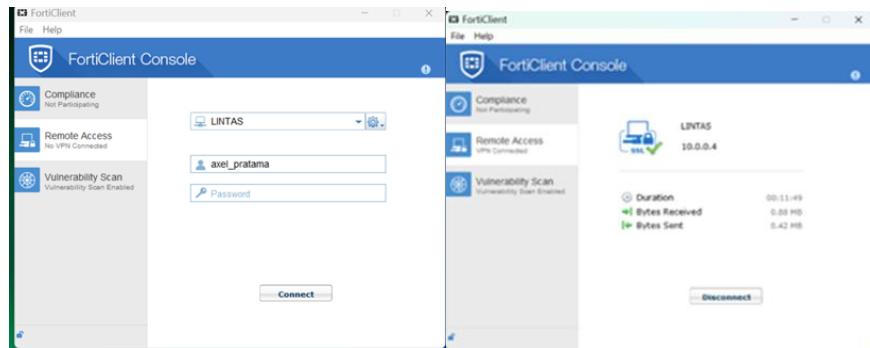
### 3.3.1.2 Melakukan Monitoring Serta menjalankan *Daily Exec*



Gambar 3.1 Diagram alur monitoring

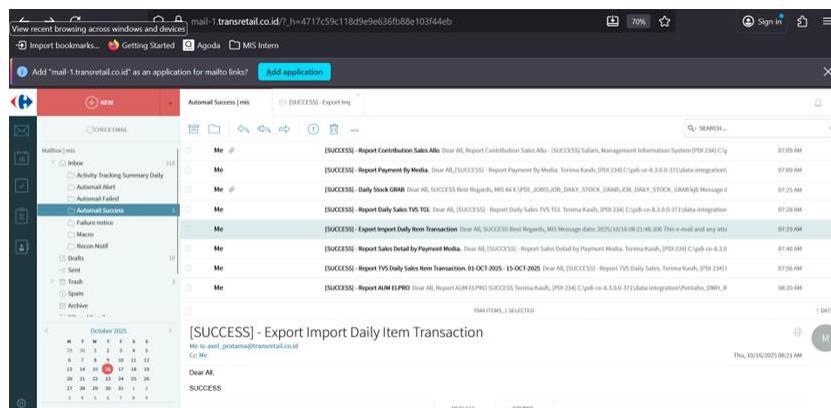
Diagram tersebut menunjukkan proses pemantauan dan pelaksanaan eksekusi harian yang dilakukan untuk memastikan bahwa operasional sistem berjalan dengan baik setiap hari. Proses dimulai dari langkah persiapan agar dapat mengakses sistem, yaitu dengan mengaktifkan

koneksi Virtual Private Network (VPN) untuk menjamin bahwa akses ke jaringan internal perusahaan dilakukan dengan aman. Setelah VPN diaktifkan, lalu memasukkan email karyawan sebagai langkah autentikasi, kemudian membuka peramban Mozilla untuk mengakses sistem internal perusahaan.



Gambar 3.2 Gambar diatas menunjukan VPN sebelum (kiri) dan sesudah (kanan) Connect

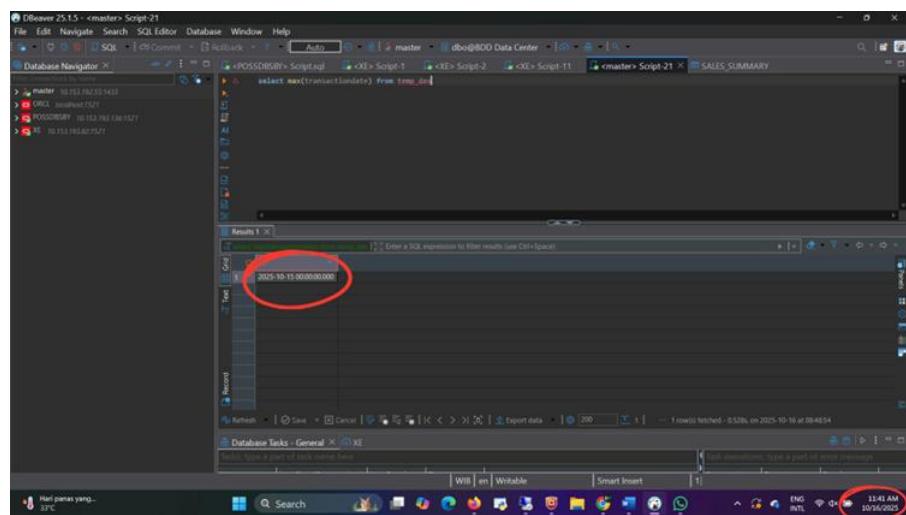
Langkah berikutnya adalah melakukan akses ke situs web internal yang ditujukan khusus untuk email karyawan. Pada tahap ini, dilakukan pemeriksaan untuk memastikan apakah email telah berhasil masuk ke dalam sistem atau tidak. Jika proses login berhasil, langkah selanjutnya adalah melanjutkan ke kegiatan pemantauan utama.



Gambar 3.3 Gambar diatas menunjukan email sudah masuk

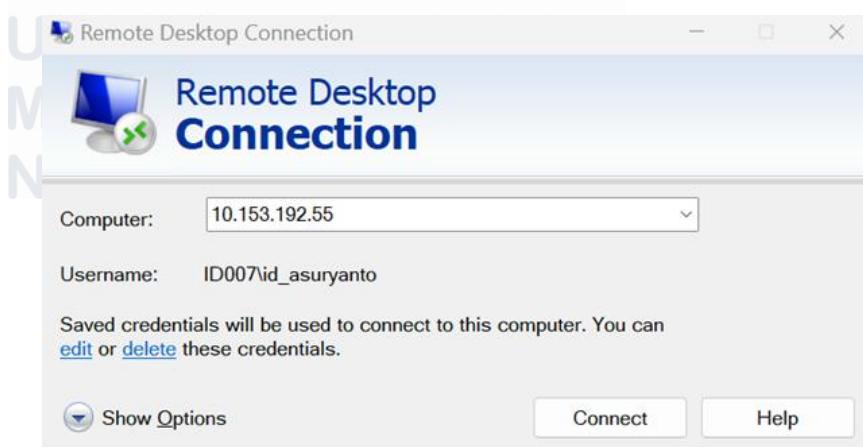
Namun, jika email belum berhasil diterima, pengecekan dilakukan Kembali lalu di cek apakah data nya sudah berjalan atau tersangkut dengan menggunakan DBeaver dengan mengakses Master Data, selanjutnya

melakukan input dan menjalankan query sesuai kebutuhan. Proses ini dilakukan untuk menjamin bahwa informasi yang diperlukan tetap dapat diakses meskipun terdapat masalah pada sistem utama.



Gambar 3.4 Halaman Dbvear untuk cek Periode Waktu

Setelah email diterima dengan sukses, Langkah selanjutnya adalah melanjutkan langkah berikut dengan mengakses remote desktop server (Desktop 55) untuk menggunakan sistem internal yang diperlukan dalam pelaksanaan kegiatan harian. Selanjutnya, membuka file Daily Execution, lalu menjalankan paket satu per satu sesuai dengan prosedur yang telah ditentukan. Tahap ini bertujuan untuk memastikan bahwa semua proses harian dilaksanakan sesuai rencana dan tidak ada kesalahan dalam sistem.



Gambar 3.5 Tampilan Desktop 55

Dalam pelaksanaan penarikan dan pengolahan data, dilakukan verifikasi terhadap hasil pencarian, terutama berkaitan dengan kesesuaian

data yang berdasarkan tanggal. Jika hasil kueri telah dinyatakan sesuai, maka proses pelaksanaan harian dapat diselesaikan. Namun, jika hasil data belum memuaskan, maka akan dilakukan proses perbaikan dengan membuka remote desktop yang lain (Desktop 66), kemudian mengakses file Export dan memilih data Daily Looping untuk menjalankan proses pengolahan data di Spoon (Pentaho Data Integration).



Gambar 3.6 Tampilan remot Desktop 66

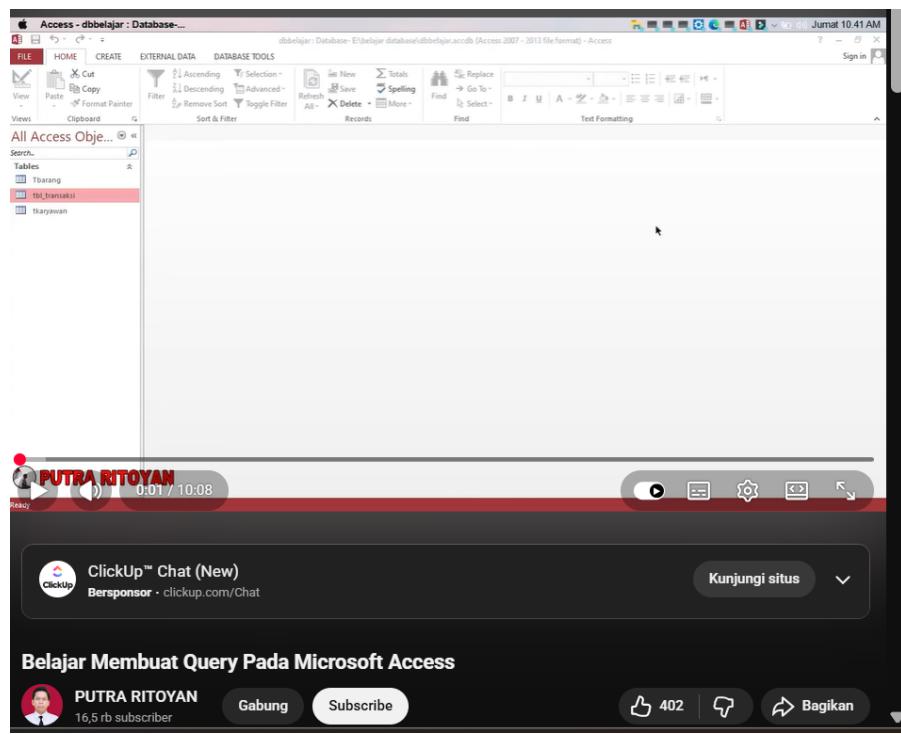
Setelah itu, akan mengulangi bagian Daily Item hingga data yang diperoleh memenuhi kebutuhan yang diinginkan. Proses *running* ulang data memerlukan waktu lebih, biasanya memerlukan waktu sekitar satu jam sebelum memeriksa kembali hasil datanya. Proses ini dilakukan cukup kali sampai semua tanggal yang dihasilkan sesuai dengan apa yang telah ditentukan supervisi. Setelah semua tahap selesai dan tanggal dinyatakan valid, lalu akan melakukan cek ulang email yang dimaksud sudah masuk atau belum. Jika sudah maka akan menjalankan daily execute di desktop 55.

Secara umum, diagram tersebut menggambarkan bahwa proses pemantauan pelaksanaan harian dilakukan dengan cara yang terstruktur dan bertahap, dimulai dari persiapan akses sistem, pelaksanaan pemantauan, pengambilan dan validasi data, hingga penyelesaian aktivitas harian. Proses ini menjamin bahwa semua kegiatan operasional sehari-hari dapat berlangsung dengan baik, tepat, dan sesuai dengan aturan yang ada di perusahaan

### **3.3.1.3 Eksplorasi Kemampuan Secara mandiri**

Pada tahap ini, proses menjelajah dan memperdalam kemampuan dalam menyusun perintah basis data sebagai bagian dari tugas utama di posisi Manajemen Informasi Sistem (MIS). Di awal masa magang, sebenarnya sudah memiliki pengetahuan dasar mengenai Structured Query Language (SQL) yang diperoleh selama masa perkuliahan. Namun, pengetahuan tersebut belum digunakan secara intensif dalam kegiatan magang sebelumnya karena fokus pekerjaan waktu itu lebih berorientasi pada bidang analisis bisnis. Situasi tersebut menyebabkan sebagian pemahaman teknis tentang SQL perlu dipelajari ulang dan disesuaikan dengan kebutuhan lingkungan kerja yang baru.

Lingkungan kerja di PT Trans Retail Indonesia menggunakan sistem manajemen basis data yang diakses melalui perangkat lunak DBeaver. Maka proses untuk beradaptasi dengan alat ini karena selama pembelajaran di kampus, mereka lebih terbiasa menggunakan MySQL dan SQL Server. Proses adaptasi ini menjadi bagian penting dalam upaya menjelajahi kemampuan, karena harus memahami perbedaan lingkungan, struktur database, serta cara menghubungkan ke basis data perusahaan melalui jaringan Virtual Private Network (VPN). Perintah yang disusun bukan hanya sederhana, tetapi juga mencakup perintah yang cukup rumit, seperti penggabungan beberapa tabel (JOIN), penggunaan fungsi agregasi, serta penyaringan data sesuai dengan kebutuhan operasional perusahaan.



Gambar 3.7 Chanel yang digunakan untuk belajar query

Dalam proses menjelajahi kemampuan menyusun perintah, penerapan berbagai pendekatan pembelajaran secara mandiri. Salah satu caranya adalah dengan mempelajari dokumen dan referensi dari berbagai sumber internet, seperti artikel teknis, tutorial berbasis web, serta video pembelajaran di platform YouTube. Selain itu, memanfaatkan teknologi kecerdasan buatan (AI) untuk berlatih menyelesaikan soal perintah, di mana AI digunakan untuk memberikan contoh kasus, latihan soal, serta membantu mengevaluasi hasil perintah yang telah disusun. Pendekatan ini membantu untuk mengasah logika penyusunan perintah secara bertahap dan sistematis.

Selain belajar mandiri, eksplorasi melalui modifikasi perintah yang

telah ada juga cukup membantu untuk berlatih. Perintah-perintah yang sebelumnya pernah digunakan, baik yang dibuat sendiri maupun yang diberikan oleh mentor (Mas Axel), dijadikan bahan pembelajaran untuk memahami pola penulisan, struktur logika, serta fungsi-fungsi yang digunakan dalam penarikan data. Dengan memodifikasi perintah tersebut sesuai dengan kebutuhan baru, sehingga kedepannya dapat memahami bagaimana perubahan kecil dalam perintah bisa mempengaruhi hasil data yang ditampilkan. Proses trial and error menjadi bagian yang tidak terpisahkan dalam tahap ini, terutama ketika menghadapi kesalahan atau hasil perintah yang tidak sesuai dengan harapan.

Data yang paling umum diproses selama tahap eksplorasi adalah data Sales Summary, yang digunakan untuk mengambil informasi. Data ini membutuhkan tingkat keakuratan yang tinggi karena berkaitan langsung dengan performa penjualan dan menjadi dasar dalam pengambilan keputusan operasional. Salah satu tantangan utama yang dihadapi adalah munculnya kesalahan atau ketidaksesuaian hasil dari query, yang sering terjadi akibat ketidaktelitian dalam menulis sintaks atau karena penggunaan fungsi tertentu yang belum sepenuhnya dipahami. Tantangan ini mendorong untuk lebih teliti dalam menyusun query, memahami struktur tabel, serta mempelajari hubungan antar data.

### **3.3.1.4 Melakukan Penarikan Data Sesuai Kebutuhan User**



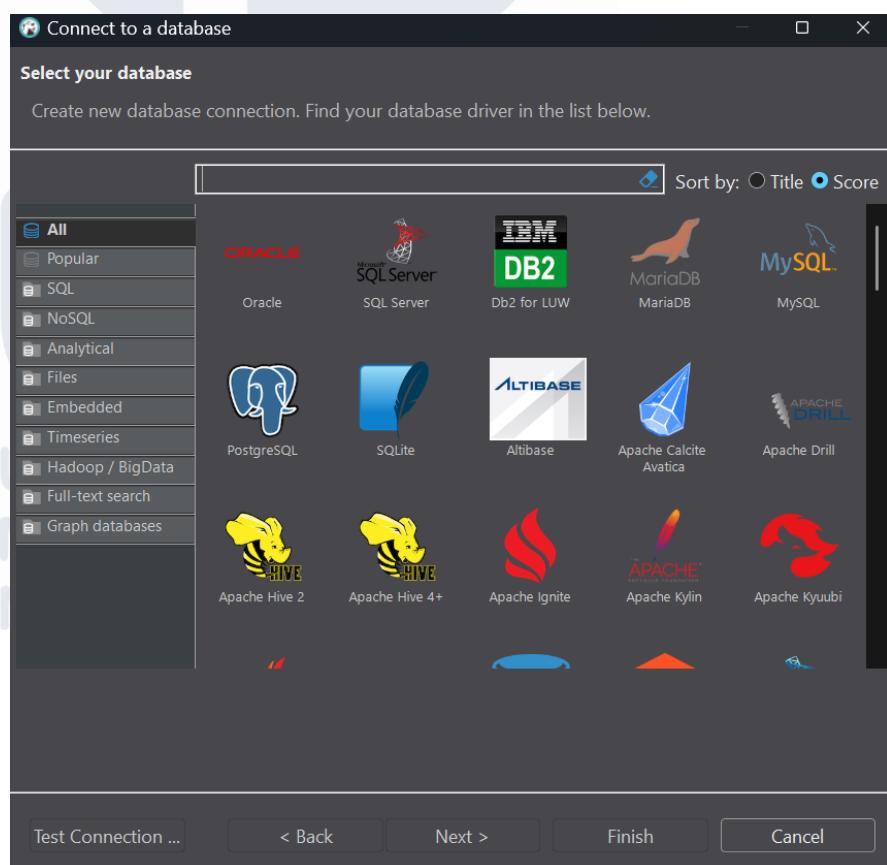
Gambar 3.8 Logo Dbeaver

Dalam pelaksanaan kegiatan magang, akan memanfaatkan DBeaver sebagai salah satu perangkat lunak utama untuk mendukung aktivitas pengelolaan dan pengambilan data. DBeaver adalah program untuk mengelola basis data yang bersifat sumber terbuka dan dapat dijalankan di berbagai sistem operasi. Perangkat lunak ini dibuat untuk membantu pengguna dalam mengelola database, termasuk menjalankan query, mengatur struktur database, serta menampilkan dan mengekspor data dengan cara yang visual.

Penggunaan DBeaver selama proses magang sangat mendukung dalam mengakses serta mengelola basis data relasional, terutama basis data yang berbasis PostgreSQL yang diterapkan oleh perusahaan. Dengan antarmuka grafis yang mudah dipahami, DBeaver memungkinkan pengguna untuk menulis dan menjalankan perintah SQL tanpa perlu menggunakan command line secara langsung. Fitur editor SQL yang ada juga menyediakan penyorotan sintaks dan pelaksanaan query secara langsung, hal ini mempermudah proses pemeriksaan dan

validasi data.

Selain menjalankan kueri, DBeaver juga digunakan dalam aktivitas pemantauan data harian, terutama untuk memastikan bahwa data yang diambil sesuai dengan kebutuhan operasional. DBeaver dimanfaatkan untuk mengakses data utama, memeriksa hasil dari kueri dalam periode tertentu, serta melakukan penyesuaian pada kueri jika terdapat ketidaksesuaian data. Fitur visualisasi data yang disajikan dalam bentuk tabel membantu dalam melakukan analisis awal sebelum data digunakan atau diproses lebih lanjut.



Gambar 3.9 Pilihan Database di Dbvear

DBeaver juga dapat mengekspor data ke berbagai format file, seperti CSV, yang mendukung integrasi data dengan sistem lain atau alat tambahan[9]. Kemampuan ini sangat penting bagi dalam menjamin bahwa data yang dihasilkan dapat dimanfaatkan oleh unit terkait tanpa harus melalui proses pengolahan ulang yang rumit. Selain itu, kemampuan DBeaver dalam mendukung berbagai jenis basis data memberikan kenyamanan bagi tim IT untuk beroperasi dalam satu platform yang terpadu.

Secara umum, pemanfaatan DBeaver dalam aktivitas magang memiliki peran yang signifikan dalam mendukung proses pengelolaan database yang efisien, tepat, dan teratur. Perangkat lunak ini mendukung dalam melakukan pemantauan sistem, pengambilan data, serta membantu dalam pelaksanaan proyek Data Query Builder, dengan tujuan untuk mengurangi ketergantungan pengguna internal terhadap penulisan kueri secara manual oleh tim TI.

```
• SELECT
  SALES_DATE, STORE_CODE, STORE_NAME,BU_NAME, pos ,SUM(CUST_DEBIT) TOTAL_TRX,SUM (GROSS_SALES) TOTAL_AMOUNT
FROM apps.SALES_SUMMARY
WHERE TRUNC(SALES_DATE) BETWEEN '1 AUG 2025' AND '31 AUG 2025'
GROUP BY SALES_DATE, STORE_CODE, STORE_NAME, BU_NAME, POS
ORDER BY SALES_DATE
```

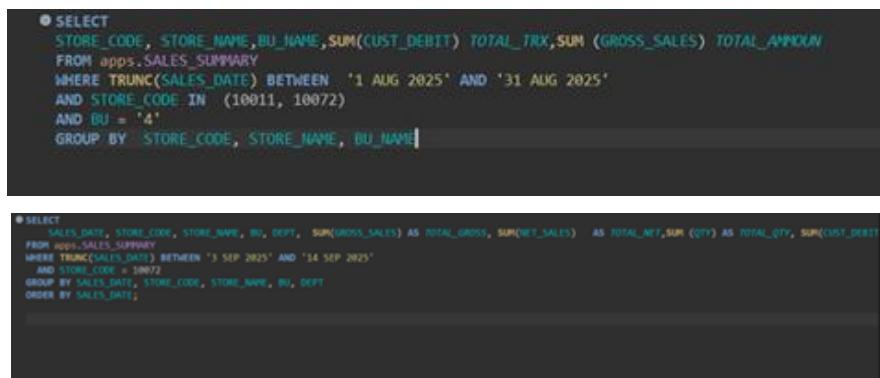
Gambar 3.10 Contoh Query Penarikan Data

Gambar ini menunjukkan contoh query Structured Query Language (SQL) yang dipakai untuk memantau dan menarik data transaksi penjualan selama periode magang. Permintaan ini dilaksanakan

melalui aplikasi DBeaver untuk mengambil informasi transaksi dari basis data perusahaan sesuai dengan kebutuhan operasional dan permintaan pengguna internal. Dalam bagian SELECT, kueri memperoleh sejumlah atribut penting yang berkaitan dengan informasi penjualan, yaitu SALES\_DATE (tanggal transaksi), STORE\_CODE (kode toko), STORE\_NAME (nama toko), BU\_NAME (nama unit bisnis), serta POS (titik penjualan). Selain itu, kueri juga memanfaatkan fungsi agregasi SUM() untuk menghitung total transaksi berdasarkan nilai CUST\_DEBIT dan total nilai penjualan berdasarkan GROSS\_SALES, yang masing-masing diberi nama alias sebagai TOTAL\_TRX dan TOTAL\_AMOUNT. Penerapan fungsi agregasi ini dimaksudkan untuk mendapatkan data dalam format ringkas yang lebih sederhana untuk dianalisis.

Dalam bagian FROM, kueri mengambil data dari tabel APPS.SALES\_SUMMARY, yang berfungsi sebagai sumber data. Selanjutnya, pada bagian WHERE, kueri membatasi data yang diambil berdasarkan rentang tanggal tertentu, yakni dari 1 Agustus 2025 hingga 31 Agustus 2025, sehingga data yang diperoleh hanya meliputi transaksi dalam periode tersebut. Bagian GROUP BY berfungsi untuk mengorganisir data berdasarkan tanggal transaksi, kode toko, nama toko, unit bisnis, serta POS. Pengelompokan ini diperlukan agar hasil perhitungan fungsi agregasi dapat disajikan secara teratur sesuai dengan setiap kombinasi atribut yang telah ditentukan. Selanjutnya, pada bagian ORDER BY, data diatur menurut SALES\_DATE agar proses pembacaan

dan analisis data secara kronologis menjadi lebih mudah. Secara keseluruhan, kueri SQL yang terdapat pada gambar itu digunakan sebagai dasar untuk memantau transaksi harian dan bulanan, serta merupakan salah satu contoh aktivitas pengambilan data secara manual yang sebelumnya dilakukan oleh tim IT. Proses ini kemudian menjadi dasar pembuatan aplikasi Data Query Builder, yang bertujuan untuk membantu pengguna internal dalam menarik data tanpa perlu menulis query SQL secara langsung.



```
④ SELECT
    STORE_CODE, STORE_NAME, BU_NAME, SUM(CUST_DEBIT) TOTAL_TRX, SUM(GROSS_SALES) TOTAL_AMOUN
    FROM apps.SALES_SUMMARY
    WHERE TRUNC(SALES_DATE) BETWEEN '1 AUG 2025' AND '31 AUG 2025'
    AND STORE_CODE IN (10011, 10072)
    AND BU = 'A'
    GROUP BY STORE_CODE, STORE_NAME, BU_NAME|
```

```
⑤ SELECT
    SALES_DATE, STORE_CODE, STORE_NAME, BU, DEPT, SUM(GROSS_SALES) AS TOTAL_GROSS, SUM(NET_SALES) AS TOTAL_NET, SUM(QTY) AS TOTAL_QTY, SUM(CUST_DEB|
```

Gambar 3.11 Beberapa Query lainnya untuk Penarikan Data

### 3.3.1.5 Pengenalan Proyek

Pada tahap ini, briefing dengan proyek pembuatan aplikasi berbasis web yang disebut Data Query Builder, sebuah aplikasi baru yang dibuat dari awal sesuai dengan kebutuhan operasional perusahaan. Proyek ini diberikan oleh seorang IT Engineer sebagai tugas tambahan selama masa magang, dengan tujuan agar waktu kerja dapat dimanfaatkan secara optimal serta memberikan pengalaman praktis dalam proses perancangan sistem informasi.

Sebelum adanya aplikasi ini, proses pengambilan data dilakukan secara manual dengan cara menulis query SQL langsung sesuai dengan kebutuhan pengguna. Proses tersebut hanya bisa dilakukan oleh tim IT, sehingga para pengguna harus mengajukan permintaan data terlebih dahulu dan menunggu hingga tim IT selesai memprosesnya. Kondisi ini dinilai tidak efisien, terutama dalam hal pengambilan data yang bersifat rutin.

Dengan adanya aplikasi Data Query Builder, perusahaan berencana menyediakan alat bantu internal yang memungkinkan pengguna menarik data secara mandiri melalui antarmuka berbasis web, tanpa tergantung sepenuhnya pada tim IT. Aplikasi ini dirancang untuk membantu pengguna dalam menyusun query secara terstruktur berdasarkan kebutuhan yang telah ditentukan. Hal ini bertujuan meminimalkan kesalahan dalam penulisan query dan mempercepat proses pengambilan data.

Dalam tahap pengenalan proyek, pemahaman mengenai tujuan utama aplikasi, masalah yang ingin diatasi, serta batasan-batasan sistem yang akan dibangun dijelaskan secara umum oleh mentor, lalu dijelaskan sedikit terkait dengan alur kerja umum aplikasi, mulai dari input kebutuhan data oleh pengguna, proses pembentukan query, hingga penarikan data dari database perusahaan. Tahap ini menjadi dasar awal sebelum melanjutkan ke tahap perancangan dan pengembangan sistem

### **3.3.1.6 Penyusunan FSD**

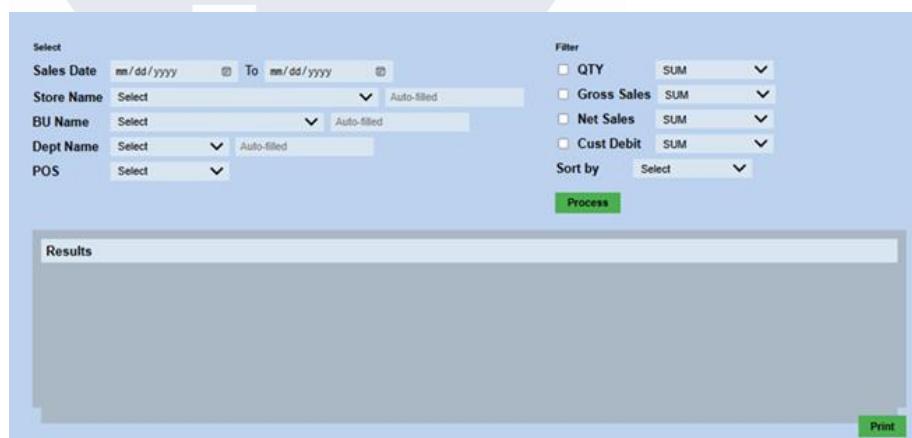
Setelah memahami tujuan dan lingkup proyek, dilanjutkan ke tahap pembuatan Functional Specification Document (FSD) sebagai dokumen perancangan awal aplikasi Data Query Builder. FSD ini dibuat untuk menjelaskan secara pasti fungsi-fungsi utama sistem yang akan dikembangkan sejak awal pembuatan aplikasi. Dalam pembuatan FSD, dilakukan sesi diskusi dengan mentor untuk mengidentifikasi kebutuhan pengguna serta menentukan fitur yang diperlukan dalam aplikasi. Dokumen FSD mencakup penjelasan tentang fungsi sistem, seperti cara memilih parameter data, pembuatan query secara otomatis berdasarkan masukan pengguna, proses eksekusi query ke database perusahaan, serta tampilan hasil data dalam bentuk yang mudah dipahami.

Selain fungsi utama, FSD juga mencakup ketentuan mengenai hak akses pengguna, validasi masukan, serta batasan penggunaan aplikasi untuk menjaga keamanan data perusahaan. Pembuatan FSD dilakukan secara bertahap, dan dalam pelaksanaannya terjadi penyesuaian kebutuhan sesuai dengan perkembangan proses perancangan dan pengembangan aplikasi dari awal.

Dengan adanya FSD, pengembangan proyek memiliki pedoman yang jelas dalam membangun aplikasi Data Query Builder sesuai dengan kebutuhan operasional perusahaan. Proses pembuatan dokumen ini juga membantu dalam memahami pentingnya perancangan dan dokumentasi sistem sebelum masuk ke tahap implementasi

### 3.3.1.7 Pengerjaan Proyek

Proyek Data Query Builder dikerjakan secara mandiri dengan menerapkan metode trial and error yang disesuaikan dengan kondisi yang ada di lapangan. Dalam proses pengerjaan, di tahap ini akan lebih banyak melakukan eksperimen langsung dalam pengembangan aplikasi, kemudian melaporkan kepada pembimbing jika menemukan hambatan atau masalah yang cukup signifikan. Metode ini dipilih karena tujuan utama proyek cukup jelas, namun tidak diberikan spesifikasi teknis yang terlalu rinci mengenai cara penerapannya.



Gambar 3.12 Contoh Desain UI pertama Website Data Query Builder

Pada tahap awal, penggunaan referensi berbasis kecerdasan buatan sebagai bahan pemahaman dalam pembuatan website. Namun, pendekatan awal yang terlalu rumit justru menyulitkan proses perbaikan ketika terjadi kesalahan dalam sistem. Karena itu, mengubah pendekatan dengan menggunakan referensi yang lebih sederhana dan mudah dipahami. Dengan demikian, jika terjadi masalah, hal tersebut masih bisa dicari sumbernya dan diperbaiki melalui berbagai sumber belajar tambahan

seperti dokumentasi daring, video pembelajaran, atau pencarian mandiri di internet.

Dalam menentukan urutan pekerjaan, prioritas pengembangan fitur utama didahului, sedangkan fitur tambahan dan perbaikan dilakukan di tahap berikutnya. Pendekatan ini bertujuan agar fungsi inti aplikasi dapat berjalan sesuai dengan tujuan proyek sebelum dilanjutkan ke tahap pengembangan berikutnya. Dengan cara ini, aplikasi dapat diuji secara bertahap sejak fitur dasarnya selesai.

Metode kerja yang digunakan bersifat iteratif, yaitu melalui tahapan pembuatan, pengujian, dan perbaikan secara berulang. Setiap bagian aplikasi yang sudah dikembangkan akan diuji terlebih dahulu untuk memastikan fungsinya berjalan dengan baik. Jika ditemukan kekurangan atau potensi perbaikan, akan dilakukan penyesuaian sebelum melanjutkan ke tahap berikutnya. Metode ini memungkinkan aplikasi berkembang secara bertahap sesuai dengan kebutuhan, tanpa harus menunggu seluruh sistem selesai terlebih dahulu.

Selama proses penggeraan, revisi dari pembimbing lapangan cukup jarang karena pembimbing lebih berfokus pada pencapaian tujuan utama aplikasi dibandingkan detail teknis implementasi tertentu. Revisi biasanya dilakukan setelah aplikasi sudah berjalan, tetapi ditemukan masalah yang cukup sulit diperbaiki, sehingga dibutuhkan penyesuaian pendekatan atau penyelesaian alternatif agar fungsi inti aplikasi tetap dapat berjalan dengan

baik.

Dalam pembuatan aplikasi Data Query Builder, berbagai bahasa pemrograman dan teknologi pendukung digunakan sesuai dengan kebutuhan sistem serta lingkungan kerja internal perusahaan. Pemilihan teknologi dilakukan dengan memperhatikan kemudahan dalam pengembangan serta kemampuan aplikasi untuk menjalankan fungsi utamanya sebagai alat bantu dalam menarik data.

Pada bagian antarmuka pengguna (frontend), aplikasi dikembangkan menggunakan HTML, CSS, dan JavaScript. HTML digunakan untuk membuat struktur dasar halaman web, CSS digunakan untuk mengatur tampilan dan tata letak antarmuka agar lebih terorganisir dan mudah dipahami, sedangkan JavaScript digunakan untuk mengelola interaksi pengguna serta memproses input yang diberikan. Selain itu, format data JSON digunakan sebagai media pertukaran dan pengolahan data antara sistem dan tampilan web.

Pada bagian pemrosesan (backend), aplikasi dikembangkan menggunakan Node.js sebagai lingkungan eksekusi untuk menjalankan proses pengolahan data. Untuk menghubungkan aplikasi dengan database perusahaan, diperlukan instalasi client database khusus, yaitu Instant Client, yang berfungsi sebagai penghubung antara aplikasi dan sistem database internal perusahaan. Penggunaan Instant Client merupakan salah satu syarat agar aplikasi dapat mengakses data yang tersedia di lingkungan internal.

Database yang digunakan merupakan database milik perusahaan yang bersifat tertutup dan hanya dapat diakses melalui jaringan internal kantor. Karena itu, akses ke database hanya dapat dilakukan jika perangkat terhubung ke jaringan internal perusahaan melalui koneksi WiFi kantor atau menggunakan Virtual Private Network (VPN) yang disediakan perusahaan, yaitu FortiClient. Pembatasan ini diterapkan sebagai bagian dari kebijakan keamanan data perusahaan.

Dalam proses pengembangan aplikasi, perangkat lunak Visual Studio Code digunakan sebagai text editor utama untuk menulis dan mengelola kode program. Selain itu, web browser digunakan untuk melakukan pengujian dan memastikan aplikasi dapat berjalan sesuai dengan fungsinya. Secara keseluruhan, teknologi yang digunakan dalam proyek ini tergolong pada tingkat dasar hingga menengah dan dipilih agar proses pengembangan tetap sederhana namun mampu memenuhi tujuan utama aplikasi.

Proses pembuatan website Data Query Builder dilakukan secara bertahap dan disesuaikan dengan kondisi kerja di lapangan. Pada tahap awal, membuat tampilan dasar website agar aplikasi memiliki bentuk visual yang jelas. Tujuan dari pembuatan tampilan awal ini adalah memberikan gambaran mengenai alur penggunaan aplikasi sebelum fungsi-fungsi utama dikembangkan lebih lanjut.

Setelah tampilan awal terbentuk, dilakukan penyusunan struktur folder

dan file website agar pengelolaan proyek menjadi lebih terorganisir. Penyusunan struktur ini membantu memisahkan komponen tampilan, logika aplikasi, serta berkas pendukung lainnya, sehingga memudahkan proses pengembangan dan perbaikan di tahap berikutnya. Selanjutnya, melakukan pengujian koneksi ke database perusahaan. Pengujian ini tidak langsung dilakukan pada proyek website, melainkan menggunakan extension Oracle Developer di lingkungan pengembangan untuk memastikan bahwa koneksi ke database internal dapat berjalan dengan baik melalui jaringan internal perusahaan.

Setelah koneksi ke database berhasil dipastikan, proses pengembangan dilanjutkan dengan pembuatan fungsi dasar aplikasi. Pada tahap ini, fokus utama adalah memastikan website mampu menerima input dari pengguna dan menjalankan proses utama sesuai dengan tujuan aplikasi. Website kemudian mulai dikembangkan agar memiliki elemen interaktif, seperti pemilihan dan pengaturan parameter data, sehingga pengguna dapat menentukan kebutuhan data secara mandiri melalui antarmuka web.

Selama proses pengembangan tersebut, pemanfaatan bantuan teknologi berbasis kecerdasan buatan sebagai alat bantu dalam penulisan kode, pencarian kesalahan yang tidak terlihat secara langsung, serta memahami logika penarikan data. Namun, seiring berjalannya waktu, pendekatan awal yang terlalu kompleks dinilai kurang efektif. Oleh karena itu, diputuskan untuk menyederhanakan struktur dan logika aplikasi agar lebih mudah dipahami dan diperbaiki. Keputusan ini menyebabkan

pengulangan proses pembuatan website dari awal dengan pendekatan yang lebih sederhana dan terstruktur.

Selama proses pembuatan website, pengujian dilakukan secara langsung dengan menjalankan aplikasi melalui web browser setiap kali terdapat perubahan atau penambahan fungsi. Pendekatan ini memungkinkan untuk segera mengetahui apakah fitur yang dibuat berjalan sesuai dengan yang diharapkan. Proses pembuatan website dilakukan secara iteratif, yaitu melalui tahapan pembuatan, pengujian, dan perbaikan secara berulang hingga aplikasi dapat menarik data secara aktual dari database perusahaan dan menampilkan hasilnya dengan baik.

```
query-builder-app/
├── config/
│   └── database.js          # Konfigurasi database
├── js/
│   ├── query-builder.js     # Frontend utama
│   ├── csv-downloader.js    # Export ke CSV
│   └── app.js                # Inisialisasi
├── css/
│   └── style.css            # Styling
├── data/
│   └── filter-options.json  # Data dropdown
├── instantclient_23_0/      # Oracle Client
└── node_modules/
    ├── .env                  # Environment variables
    ├── server.js             # Backend server
    ├── Home.html              # Halaman utama
    ├── package.json           # Penyusunan
    └── package-lock.json
```

Gambar 3.13 Struktur Halaman Website

Struktur halaman website pada aplikasi Data Query Builder disusun secara terorganisir agar memudahkan proses pengembangan, pemeliharaan, dan pemisahan fungsi antar komponen sistem. Penyusunan

struktur tersebut bertujuan agar setiap bagian aplikasi memiliki peran yang jelas, baik di sisi tampilan, pemrosesan data, maupun konfigurasi sistem pendukung. Direktori config digunakan untuk menyimpan berkas konfigurasi yang berhubungan dengan koneksi database perusahaan. Pemisahan konfigurasi pada direktori khusus ini bertujuan agar pengaturan sistem tidak tercampur dengan logika aplikasi maupun tampilan website, sehingga lebih aman dan mudah dikelola.

Direktori js berisi berkas JavaScript yang digunakan menjalankan logika aplikasi di sisi antarmuka pengguna. Berkas-berkas dalam direktori ini mengelola proses utama aplikasi, seperti menangani interaksi pengguna, permintaan penarikan data, serta fitur tambahan seperti ekspor data ke dalam format CSV. Pemisahan berkas JavaScript berdasarkan fungsinya bertujuan menjaga struktur kode tetap rapi dan mudah dipahami. Direktori css digunakan untuk menyimpan berkas pengaturan tampilan website. Berkas ini mengatur tata letak, warna, dan gaya visual aplikasi agar tampilan website lebih konsisten dan lebih mudah digunakan oleh pengguna internal. Dengan memisahkan tampilan dan logika aplikasi, perubahan desain dapat dilakukan tanpa mengganggu fungsi sistem.

Direktori data berisi berkas pendukung berformat JSON yang digunakan menyimpan data statis, seperti opsi filter atau pilihan pada dropdown. Penggunaan berkas JSON memungkinkan data tersebut dikelola dengan terpisah dari logika utama aplikasi, sehingga memudahkan proses pembaruan saat terjadi perubahan kebutuhan.

Direktori instantclient\_23\_0 merupakan komponen pendukung yang digunakan untuk menghubungkan aplikasi dengan database Oracle milik perusahaan. Keberadaan direktori ini menjadi syarat agar aplikasi dapat berkomunikasi dengan sistem database internal sesuai dengan kebijakan infrastruktur perusahaan. Direktori node\_modules berisi dependensi yang diperlukan agar aplikasi dapat berjalan dengan baik. Direktori ini dihasilkan secara otomatis berdasarkan konfigurasi proyek dan digunakan untuk mendukung fungsi aplikasi di sisi server

Berkas .env digunakan untuk menyimpan variabel lingkungan yang bersifat sensitif, seperti konfigurasi koneksi sistem, sehingga tidak dituliskan langsung di dalam kode aplikasi. Pendekatan ini membantu menjaga keamanan dan fleksibilitas pengaturan sistem. Berkas server.js berfungsi sebagai pengelola utama proses di sisi server. Berkas ini mengelola jalannya aplikasi backend dan menjadi penghubung antara antarmuka website dengan database perusahaan. Sementara itu, berkas Home.html merupakan halaman utama website yang menjadi titik awal interaksi pengguna dengan aplikasi Data Query Builder. Selain itu, berkas package.json dan package-lock.json berfungsi untuk mengatur konfigurasi proyek serta dependensi yang diperlukan dalam proses pengembangan. Kehadiran kedua berkas tersebut memastikan bahwa lingkungan pengembangan aplikasi dapat berjalan secara konsisten.

Secara keseluruhan, struktur halaman dan direktori website Data Query

Builder dirancang secara modul dan terpisah sesuai dengan fungsi masing-masing. Pendekatan ini memudahkan proses pengembangan, pengujian, serta pemeliharaan aplikasi, sekaligus mendukung tujuan utama sistem sebagai alat bantu dalam menarik data internal perusahaan.

Lalu Untuk bagian Tampilan atau user interface. Tampilan antarmuka pengguna (User Interface/UI) dari website Data Query Builder dirancang dengan tujuan utama yaitu memudahkan pengguna yang bukan berlatar belakang teknologi informasi (non-IT) dalam mengambil data dari database perusahaan. Dalam praktiknya, kebutuhan data operasional tidak hanya berasal dari tim IT, tetapi juga dari pengguna internal lainnya seperti bagian operasional, analis, atau supervisor. Karena itu, UI dibuat agar pengguna dapat menentukan kriteria data secara mandiri tanpa harus mengerti bahasa pemrograman SQL secara teknis.

Secara tampilan, halaman utama website menampilkan status koneksi database di bagian atas sebagai indikator bahwa sistem telah terhubung langsung dengan database perusahaan. Informasi ini penting karena memberikan rasa aman dan percaya diri kepada pengguna bahwa data yang dihasilkan berasal dari sumber yang sah dan terkini. Akses ke website ini bersifat internal, yaitu pengguna hanya dapat mengaksesnya melalui jaringan dalam perusahaan atau menyambungkan ke jaringan internal melalui VPN, sehingga tidak diperlukan proses login tambahan di halaman UI.

Struktur UI dibagi menjadi beberapa bagian utama yang disusun secara

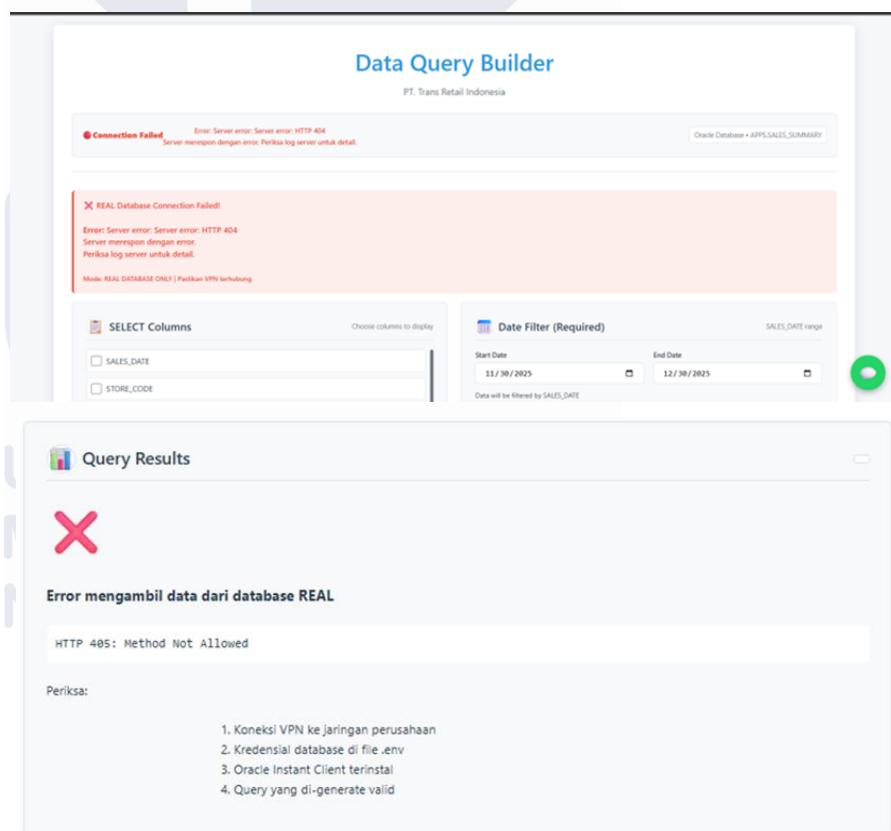
terarah. Bagian “Select Columns” memungkinkan pengguna untuk memilih kolom data yang ingin ditampilkan sesuai dengan kebutuhan analisisnya. Pendekatan ini bertujuan untuk mengurangi kesalahan dalam pembuatan query, karena pengguna hanya dapat memilih kolom yang tersedia di database. Selanjutnya, bagian “Date Filter” disediakan sebagai filter wajib, mengingat sebagian besar kebutuhan data bersifat berkala dan berbasis waktu.

Untuk kebutuhan pengolahan data singkat, UI menyediakan bagian “Aggregate Functions” seperti fungsi penjumlahan (SUM). Pengguna dapat dengan mudah menentukan agregasi data tanpa harus menulis sintaks query secara manual. Selain itu, tersedia bagian “Additional Filters” sebagai filter tambahan yang memungkinkan pengguna mempersempit data sesuai dengan kriteria tertentu. Jika kriteria yang dimasukkan tidak menghasilkan data, sistem akan memberikan umpan balik yang jelas. Sebagai rencana cadangan, pengguna dapat menghubungi tim IT melalui ikon WhatsApp yang tersedia di bagian kanan atas halaman.

Salah satu komponen penting di dalam UI adalah “SQL Query Preview”. Fitur ini berfungsi menampilkan hasil query SQL yang terbentuk berdasarkan kriteria yang dipilih pengguna. Keberadaan fitur ini memberikan transparansi kepada pengguna sekaligus berperan sebagai bentuk validasi sebelum data benar-benar diambil dari database. Dengan

demikian, pengguna dapat memastikan bahwa query yang dihasilkan sudah sesuai dengan kebutuhan dan logika data yang diinginkan.

Secara keseluruhan, alur penggunaan UI bersifat terarah namun tetap fleksibel. Umumnya, pengguna memulai dengan memilih kolom data yang diperlukan, kemudian menentukan fungsi agregasi yang diperlukan, dan lanjut dengan menambahkan filter tambahan. Desain tampilan dibuat sederhana dan fungsional agar proses penarikan data dapat dilakukan dengan lebih cepat, efisien, serta mengurangi ketergantungan pengguna terhadap tim IT.

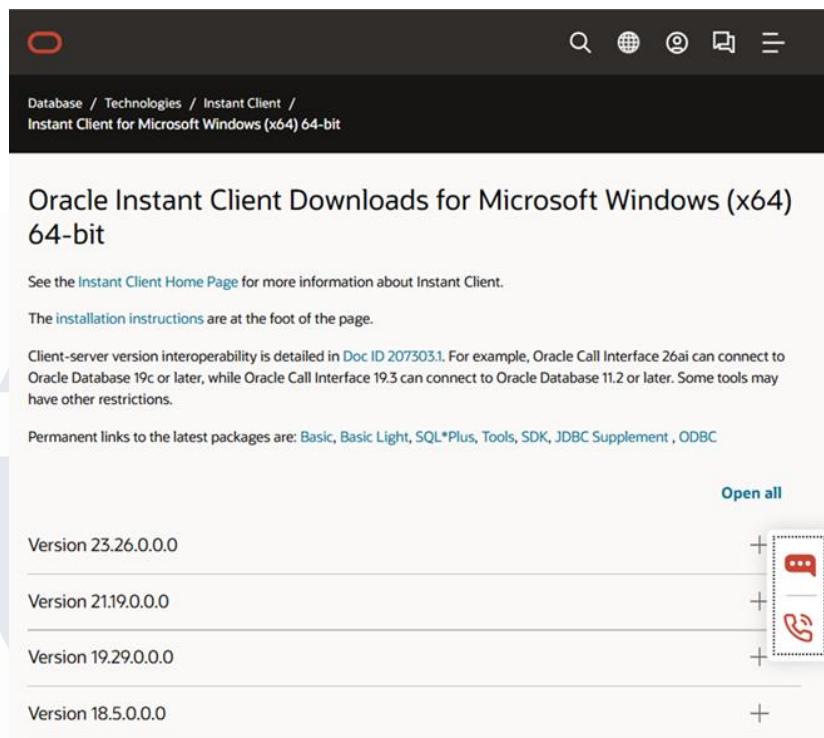


Gambar 3.14 Gambar Diatas Menunjukkan Bawa Database belum bisa Tersambung

Dalam prosesnya pengembangan website Data Query Builder,

menghadapi beberapa hambatan yang muncul dari sisi teknis maupun proses pengembangan. Kendala-kendala ini merupakan bagian dari kegiatan pembelajaran selama masa kerja magang. Salah satu hambatan utama yang dihadapi adalah keterbatasan akses terhadap database perusahaan. Akses hanya bisa dilakukan melalui jaringan internal kantor atau menggunakan aplikasi VPN perusahaan. Situasi ini menyebabkan proses pengujian aplikasi tidak selalu berjalan fleksibel, terutama ketika koneksi VPN tidak stabil atau belum terhubung, sehingga aplikasi menampilkan pesan kesalahan koneksi ke database.

Kendala berikutnya berkaitan dengan koneksi serta pengaturan database Oracle, termasuk penggunaan Oracle Instant Client dan penyimpanan kredensial di file environment. Kesalahan dalam pengaturan atau perbedaan antara lingkungan kerja sering menyebabkan munculnya error saat menarik data, seperti gagalnya respons dari server atau metode request yang tidak sesuai. Untuk menyelesaikan masalah tersebut, melakukan pemantauan ulang terhadap koneksi, memeriksa kembali validitas query yang dihasilkan, serta memastikan lingkungan kerja sudah sesuai dengan kebutuhan sistem.



Gambar 3.15 Situs Resmi Oracle Untuk Download Instant Client

Untuk mengatasi masalah yang terjadi saat membangun website Data Query Builder, dilakukan beberapa penyesuaian sesuai dengan kondisi di lapangan. Keterbatasan akses ke database berhasil diatasi dengan memastikan koneksi ke jaringan internal perusahaan atau VPN sudah aktif sebelum melakukan pengujian aplikasi. Dengan demikian, proses pengujian dapat dilakukan dalam lingkungan yang sesuai dengan sistem operasional perusahaan.

Selain itu, masalah pada koneksi dan konfigurasi database Oracle ditangani dengan melakukan pembaruan lingkungan pengembangan, salah satunya dengan mengunduh dan menginstal ulang Oracle Instant Client versi terbaru. Lalu dilakukan pengecekan kembali terhadap konfigurasi koneksi dan kredensial database agar sesuai dengan kebutuhan sistem.

Langkah-langkah tersebut berhasil mengurangi terjadinya error dan memastikan aplikasi dapat berjalan serta menarik data sesuai dengan kondisi database perusahaan.

The screenshot shows a web-based application for executing SQL queries. At the top, there's a 'SQL Query Preview' section containing a complex SELECT statement. Below this are several buttons: 'Generate Query', 'Execute Query', 'Generating CSV...', 'Test Connection', and 'Reset Form'. The main area is titled 'Query Results: 11 rows from database (EAI)'. It displays a table with columns: SALES\_DATE, STORE\_CODE, STORE\_NAME, BU, SUM(GROSS\_SALES), and SUR(VAT). The data shows sales for various stores across different business units on January 12, 2025.

SALES_DATE	STORE_CODE	STORE_NAME	BU	SUM(GROSS_SALES)	SUR(VAT)
1/12/2025	T03111	Transnet Cempaka Putih	1	Rp 17.469.153,00	-
1/12/2025	T03111	Transnet Cempaka Putih	2	Rp 3.499.239,00	-
1/12/2025	T03111	Transnet Cempaka Putih	4	Rp 61.867.200,00	-
1/12/2025	T03111	Transnet Cempaka Putih	7	Rp 2.668.000,00	-
1/12/2025	T03111	Transnet Cempaka Putih	A	Rp 444.210,00	-
1/12/2025	T03111	Transnet Cempaka Putih	B	Rp 9.716.345,00	-
1/12/2025	T03111	Transnet Cempaka Putih	D	Rp 107.000,00	-
1/12/2025	T03111	Transnet Cempaka Putih	E	Rp 5.774.507,00	-
1/12/2025	T03111	Transnet Cempaka Putih	F	Rp 227.500,00	-

Gambar 3.16 Gambar Diatas Menunjukan Web bisa mengeluarkan data sesuai database

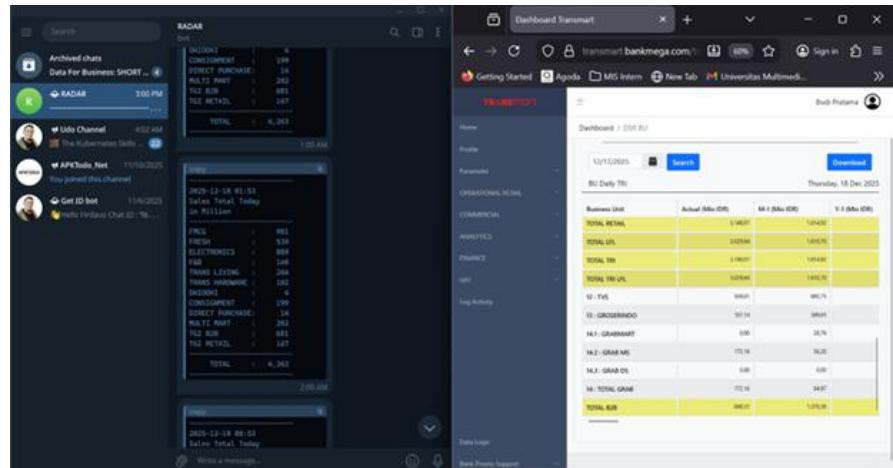
Setelah melalui proses penyesuaian lingkungan pengembangan dan menyelesaikan berbagai kendala teknis, website Data Query Builder berhasil terhubung ke database perusahaan. Pembaruan Oracle Instant Client dan konfigurasi koneksi yang tepat memungkinkan aplikasi mengakses database baik melalui jaringan internal perusahaan maupun melalui VPN secara lebih stabil.

Pada tahap akhir, website mampu menampilkan data langsung dari database sesuai dengan kriteria yang disediakan oleh pengguna. Data yang ditampilkan pada website selaras dengan isi database perusahaan, sehingga hasil yang diperoleh dapat digunakan untuk keperluan analisis maupun operasional. Dengan tercapainya hasil tersebut, aplikasi Data Query Builder dinilai telah memenuhi tujuan utama proyek, yaitu membantu pengguna non-IT dalam mengambil data secara mandiri melalui antarmuka berbasis web tanpa harus bergantung sepenuhnya pada tim IT.

### **3.3.1.8 Melakukan Monitoring Dashboard Bank Mega**

Berdasarkan gambar dibawah, terdapat dua sumber data utama yang digunakan dalam proses pemantauan dan validasi kinerja penjualan, yaitu data penjualan yang dikirimkan melalui aplikasi Telegram serta data penjualan yang ditampilkan di situs web dashboard Transmart Bank Mega. Kedua sumber informasi ini memiliki peran yang saling mendukung dalam membantu kegiatan pengawasan operasional harian oleh tim MIS. Informasi yang disajikan dalam aplikasi Telegram berasal dari sistem internal yang secara otomatis mengirimkan ringkasan penjualan harian. Pengiriman data ini direncanakan dan diperbaharui secara teratur setiap hari pada pukul 02. 00 WIB. Informasi yang disampaikan mencakup keseluruhan penjualan serta detail kontribusi dari berbagai kategori usaha, seperti FMCG, Fresh, Elektronik, hingga B2B. Data Telegram berperan sebagai laporan awal yang memberikan gambaran singkat mengenai kinerja penjualan sebelum data resmi sepenuhnya disinkronkan dalam

sistem eksternal.



Gambar 3.17 Dua Halaman yang Digunakan untuk Monitor

Sementara itu, data perbandingan diperoleh dari situs web resmi dashboard Transmart yang terhubung dengan sistem Bank Mega. Dashboard ini menunjukkan data Unit Bisnis Daily TRI yang diperbarui secara berkala pada waktu-waktu tertentu, dengan pembaruan terakhir dilaksanakan sekitar pukul 09. 45 WIB. Karena proses pembaruan informasi di situs web melibatkan sistem antar lembaga, terdapat jeda waktu antara data yang ditampilkan di Telegram dan data akhir yang terlihat di papan dasbor situs web. Dalam situasi ini, sebagai intern MIS (Sistem Informasi Manajemen) memiliki tanggung jawab utama untuk melakukan pencocokan data yang dikirim melalui Telegram dengan data yang terdapat di situs web Transmart Bank Mega. Proses pencocokan ini bertujuan untuk memastikan bahwa angka penjualan yang dilaporkan adalah konsisten, tepat, dan tidak terdapat perbedaan yang berarti. Jika terdapat perbedaan data dalam batas wajar, hal

itu masih dapat diterima karena dipengaruhi oleh variasi waktu pembaruan sistem.

Namun, jika terdapat perbedaan data yang sangat signifikan atau tidak wajar, maka harus melaporkan temuan ini kepada pengawas atau mentor. Selanjutnya, mentor yang bersangkutan (yaitu Mas Axel) akan menyampaikan informasi tersebut kepada Bank Mega, yang merupakan pemilik dan pengelola sistem dashboard. Proses ini mengindikasikan adanya jalur komunikasi resmi yang perlu diikuti dalam menangani anomali data untuk mencegah kesalahan dalam interpretasi saat membuat keputusan manajerial.

Harus dinyatakan dengan jelas bahwa perbedaan data itu tidak diperbaiki secara langsung oleh tim IT internal, karena masalah ini berada di luar batas tanggung jawab tim IT internal perusahaan. Sistem Telegram dan sistem dashboard Bank Mega merupakan entitas yang berbeda dan memiliki kepemilikan yang terpisah, sehingga setiap perubahan atau pembaruan data hanya dapat dilakukan oleh Bank Mega sebagai pengelola utama sistem tersebut. Dengan demikian, fungsi MIS terbatas pada pengawasan, verifikasi, dan pelaporan, bukan pada intervensi teknis terhadap sistem yang berada di luar. Secara umum, langkah-langkah yang terlihat pada gambar menggambarkan metode pengelolaan dan pemantauan data operasional yang krusial dalam konteks perusahaan ritel besar. Dengan melakukan pemeriksaan data secara teratur dan melaporkan perbedaan yang penting, perusahaan dapat mempertahankan keandalan informasi manajerial serta

memastikan bahwa data yang digunakan sebagai dasar untuk pengambilan keputusan adalah akurat dan dapat dipertanggungjawabkan.

### **3.3.2 Kendala yang Ditemukan**

Selama masa kerja magang, terdapat berbagai kendala yang mempengaruhi proses kerja dan kemampuan dalam menyelesaikan tugas. Kendala tersebut muncul dari berbagai faktor teknis maupun non-teknis yang berkaitan dengan kondisi lingkungan kerja serta adaptasi terhadap sistem kerja di Perusahaan

Kendala pertama yang dialami adalah jarak lokasi kantor yang cukup jauh dari tempat tinggal. Hal ini menyebabkan waktu tempuh yang relatif lama setiap hari, serta berpotensi mempengaruhi fokus dan kinerja kerja. Jarak yang jauh juga membatasi fleksibilitas waktu, terutama ketika harus menyesuaikan jadwal kerja magang dengan aktivitas akademik di kampus.

Kendala kedua terkait dengan penggunaan perangkat lunak baru, yaitu DBeaver, yang sebelumnya belum dikuasai. DBeaver digunakan sebagai alat utama dalam proses penarikan, pengelolaan, dan pemantauan data dari database perusahaan. Pada awal penggunaan, memerlukan waktu untuk memahami antarmuka, fungsi utama, serta cara menulis dan menjalankan query sesuai struktur database perusahaan. Proses adaptasi ini menjadi tantangan, terutama saat harus menjalankan tugas daily execution yang menuntut ketelitian dan akurasi data.

Kendala ketiga adalah tidak adanya teman magang dalam sub divisi Management Information System (MIS). Hal ini menyebabkan penggerakan tugas dan proyek secara mandiri, termasuk pembuatan proyek Data Query Builder. Tidak adanya rekan magang dalam subdivisi yang sama membuat intern sulit berdiskusi langsung dengan sesama peserta magang, sehingga lebih diharapkan untuk lebih mandiri dalam mencari solusi, memahami kebutuhan sistem, serta menyelesaikan permasalahan teknis yang muncul

selama proses magang.

### **3.3.3 Solusi atas Kendala yang Ditemukan**

Berdasarkan masalah-masalah yang muncul selama proses magang, diambilah beberapa langkah penyelesaian agar pelaksanaan magang tetap berjalan lancar dan tujuan pembelajaran dapat tercapai. Untuk mengatasi masalah jarak antara tempat tinggal dan lokasi kantor, dipilih untuk tinggal di kos yang lebih dekat dengan tempat kerja. Hal ini bertujuan untuk mengurangi waktu perjalanan, mengurangi kelelahan, serta meningkatkan konsistensi dan efektivitas kehadiran selama masa magang. Dengan jarak yang lebih dekat, sehingga lebih fokus dalam menjalankan tugas harian.

Untuk mengatasi kesulitan dalam menggunakan perangkat lunak baru, yaitu DBeaver, intern diberikan waktu dan kesempatan untuk belajar secara mandiri. Pembelajaran dilakukan melalui eksplorasi fitur, pembacaan dokumentasi, serta praktik langsung dalam menjalankan query dari yang sederhana hingga kompleks. Selain itu, kesempatan untuk berdiskusi dengan mentor bila menemui masalah teknis juga diperbolehkan, sehingga pemahaman tentang penggunaan DBeaver dapat meningkat secara bertahap dan mendukung kelancaran dalam tugas monitoring data.

Dalam situasi tidak adanya teman magang dari satu subdivisi yang sama, secara tidak langsung mengembangkan sikap kerja yang lebih mandiri dan proaktif. Dalam proses pembuatan website Data Query Builder, diperbolehkan oleh mentor untuk memanfaatkan teknologi kecerdasan buatan (Artificial Intelligence/AI) sebagai alat bantu dalam memahami konsep teknis, menyusun struktur sistem, serta mempercepat proses pengembangan. Penggunaan AI ini tetap dalam bimbingan mentor dan hanya digunakan sebagai sarana pendukung pembelajaran, bukan pengganti proses analisis dan pengambilan keputusan.

Dengan menerapkan langkah-langkah penyelesaian tersebut, kendala yang dialami selama magang dapat ditangani dan tetap menjalankan tugas serta tanggung jawab sebagai Intern MIS secara optimal