

## BAB III

### PELAKSANAAN KERJA

#### 3.1 Kedudukan dan Koordinasi

Bagian ini menguraikan tentang posisi mahasiswa magang dalam susunan organisasi perusahaan dan cara koordinasi yang digunakan selama proyek berlangsung. Diskusi ini mencakup tempat kerja peserta magang, tugas dan tanggung jawab yang mereka lakukan, serta cara komunikasi dan koordinasi dengan supervisor lapangan dan pihak-pihak lain yang terkait untuk mendukung kelancaran proses pengembangan sistem.

##### 3.1.1 Kedudukan

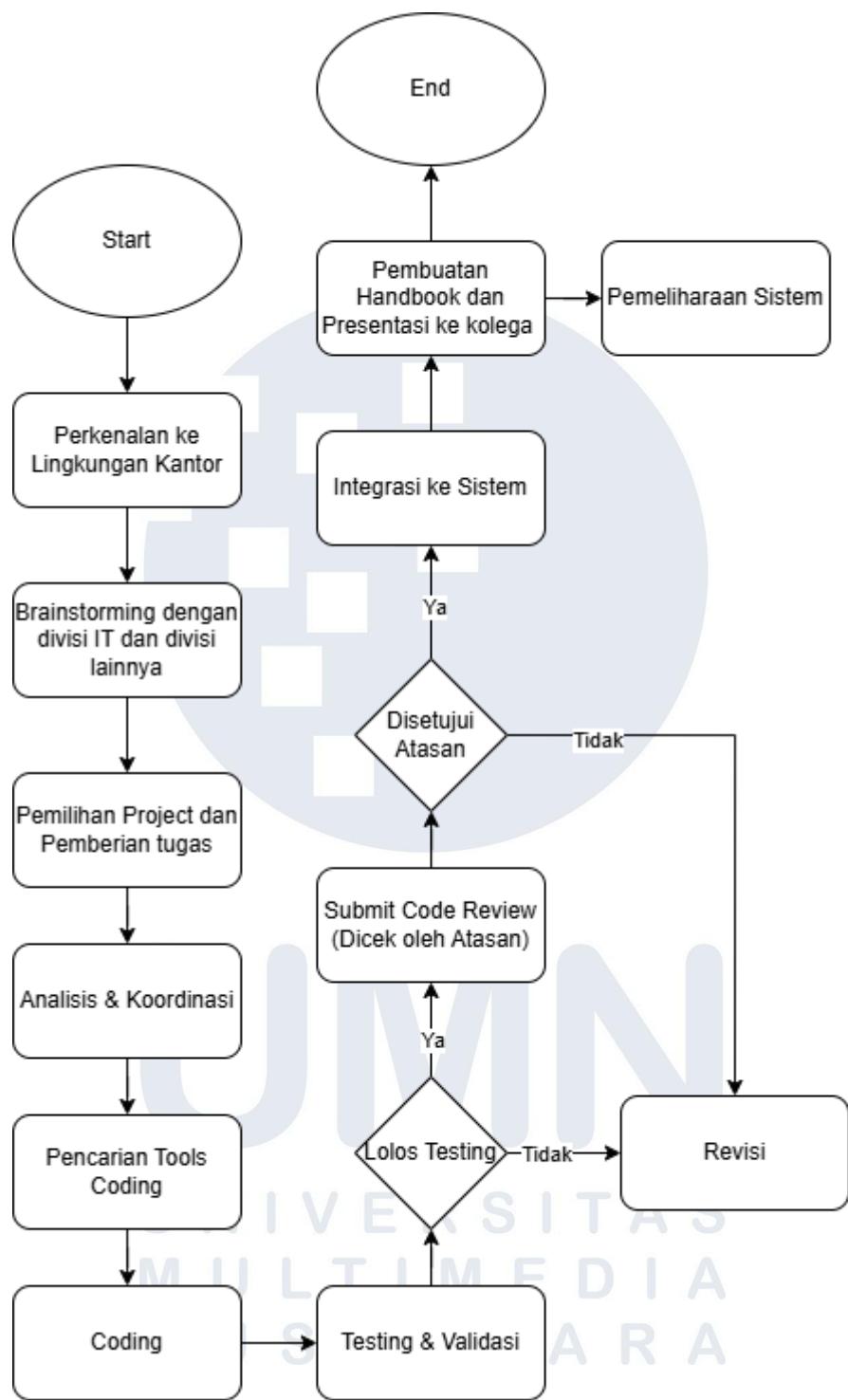
Selama menjalani program magang di PT Dayalima Abisatya, peserta magang ditempatkan di posisi IT Internship, yang berfokus pada pengembangan solusi digital untuk kebutuhan internal perusahaan. Posisi ini menawarkan ruang kerja yang strategis, mencakup mulai dari menemukan masalah di bagian administrasi, merancang proses otomatisasi, menulis kode pemrograman, hingga melaksanakan sistem berbasis web. Penempatan di posisi ini memberi kesempatan kepada peserta magang untuk memahami keseluruhan proses transformasi digital, terutama dalam memanfaatkan teknologi untuk meningkatkan efisiensi operasional.

Berdasarkan struktur organisasi yang telah dijelaskan sebelumnya pada gambar 2.1, posisi IT Internship berada di bawah bimbingan langsung dari Kepala Departemen IT. Kepala IT memiliki tanggung jawab utama memberikan petunjuk teknis, memantau kemajuan setiap proyek, serta memastikan bahwa solusi yang dibuat sesuai dengan standar teknologi dan kebutuhan operasional PT Dayalima Abisatya. Dalam konteks ini, peserta magang berfungsi sebagai dukungan teknis yang bertugas menjalankan proyek-proyek otomatisasi, seperti pembuatan sistem reservasi fasilitas dan peningkatan pengelolaan *database* perusahaan melalui pembersihan data.

Dalam menjalankan tugas sehari-hari, mahasiswa magang harus mampu beradaptasi dan bersikap aktif dalam belajar berbagai alat teknologi, mulai dari menggunakan *Google Apps Script* untuk mengintegrasikan *Google Workspace* hingga menggunakan bahasa pemrograman *Python* untuk pengolahan data mentah. Peserta magang didorong untuk memanfaatkan teknologi ini untuk mengubah proses administrasi yang tadinya manual menjadi sistem yang otomatis dan terpusat. Untuk memastikan ketepatan dalam hal-hal yang kompleks, seperti pengintegrasian *database CRM* atau pengolahan laporan data yang besar, peserta magang secara teratur berkoordinasi dan berkonsultasi dengan Kepala Departemen IT. Komunikasi yang intens ini bertujuan agar setiap sistem yang dikembangkan memiliki transparansi yang baik dalam pemantauan dan dapat mendukung kelancaran operasional semua staf di perusahaan.

### 3.1.2 Koordinasi

Agar proyek dapat berjalan dengan baik, PT Dayalima Abisatya menggunakan cara koordinasi yang teratur antara semua department yang ada. Dengan adanya koordinasi antar department ini diharapkan dapat mengurangi *misscommunication* antar department, pembagian tugas, dan pengecekan hasil kerja dapat dilakukan dengan baik dan teratur. Proses koordinasi ini dapat dilihat pada Gambar 3. 1.



Gambar 3.1 Flowchart Alur Koordinasi Magang

Berdasarkan alur kerja yang dijalankan selama masa magang, seluruh aktivitas pengembangan teknologi di PT Dayalima Abisatya mengikuti standar operasional yang sistematis untuk memastikan setiap solusi digital yang dihasilkan

benar-benar tepat guna. Proses ini diawali dengan fase orientasi melalui perkenalan mendalam terhadap lingkungan kantor dan budaya kerja perusahaan. Setelah memahami ekosistem kerja, langkah krusial berikutnya adalah melakukan sesi brainstorming bersama divisi IT serta divisi terkait lainnya untuk menjaring aspirasi dan kebutuhan riil di lapangan.

Dari hasil diskusi tersebut, dilakukan pemilihan proyek dan penetapan tugas spesifik yang mencakup tiga fokus utama: pembangunan Sistem Booking Ruangan, pelaksanaan *Data Cleansing* CRM, serta Pengolahan Data PINTAR. Tahapan ini kemudian berlanjut pada fase analisis dan koordinasi teknis untuk membedah kebutuhan sistem secara lebih mendalam, yang diikuti dengan riset serta pemilihan tools coding yang paling sesuai dengan infrastruktur teknologi perusahaan.

Memasuki tahap inti, aktivitas difokuskan pada proses koding atau pengembangan perangkat lunak. Sebagai contoh, dalam proyek Sistem Booking Ruangan yang dikerjakan selama bulan Juli hingga pertengahan Agustus, proses koding diarahkan pada pembuatan fitur reservasi otomatis untuk mencegah bentrok jadwal. Setelah fungsionalitas sistem terbentuk, dilakukan pengujian dan validasi (*Testing & Validation*) secara mandiri untuk mendeteksi adanya celah atau kesalahan teknis.

Sistem yang telah lolos pengujian internal kemudian masuk ke tahap pemeriksaan kualitas yang lebih ketat, yaitu Submit *Code Review* oleh atasan atau *Head of IT Department*. Jika terdapat catatan atau ketidaksesuaian, proses akan dikembalikan ke tahap revisi hingga mencapai standar yang diinginkan. Setelah mendapatkan persetujuan penuh, sistem baru tersebut akan diintegrasikan secara resmi ke dalam ekosistem digital perusahaan.

Sebagai penutup dari siklus pengembangan, dilakukan pembuatan buku panduan (*Handbook*) teknis serta presentasi kepada kolega dan departemen terkait untuk memastikan mereka dapat mengoperasikan sistem dengan lancar. Prosedur ini diakhiri dengan fase pemeliharaan sistem secara berkala untuk menjamin stabilitas dan keamanan data jangka panjang. Seluruh rangkaian proses ini

memastikan bahwa transformasi digital yang diusung oleh PT Dayalima Abisatya berjalan secara profesional, terukur, dan berdampak langsung pada efisiensi kerja harian.

### 3.2 Tugas yang Dilakukan

Selama menjalani program magang di PT Dayalima Abisatya, perhatian utama terfokus pada proses digitalisasi administrasi perusahaan dengan mengembangkan berbagai sistem otomatisasi yang menggunakan Google Apps Script. Semua sistem yang dibuat bertujuan untuk menggantikan metode manual sebelumnya yang menggunakan email, percakapan langsung, atau pencatatan di Excel, agar kegiatan administrasi dapat dilakukan dengan lebih efisien, terkelola dengan baik, dan mudah diawasi oleh setiap bagian terkait.

Rangkaian tugas selama magang dilaksanakan secara bertahap, diawali dengan mengidentifikasi kebutuhan operasional dari setiap divisi, merancang alur proses, menyusun struktur data, mengembangkan skrip, dan berlanjut ke tahap pengujian serta pelaksanaan. Semua langkah ini dilakukan dengan melibatkan komunikasi aktif dengan Kepala Departemen IT serta wakil pengguna yang merupakan pemilik proses dari masing-masing sistem. Dengan pendekatan ini, setiap sistem yang dikembangkan mampu memenuhi kebutuhan operasional dengan tepat dan selaras dengan alur kerja nyata perusahaan.

Untuk memberikan gambaran umum mengenai ruang lingkup tugas dan periode pelaksanaannya, daftar sistem yang telah dikembangkan disajikan dalam Tabel 3. 1 berikut.

Tabel 3.1 Detail Pekerjaan yang Dilakukan

No.	Minggu	Proyek	Keterangan
1	Sistem <i>Booking</i> Ruang	14 July 2025	7 Agustus 2025
2	<i>Data</i> <i>Cleansing</i> CRM	8 Agustus 2025	3 Oktober 2025

3	Olah Data PINTAR	10 Oktober 2025	4 November 2025
---	---------------------	-----------------	-----------------

### 3.2.1 Sistem Booking Ruangan

Sistem pemesanan ruangan dibuat berdasarkan permintaan yang diajukan oleh tim *General Affairs* (GA), yang bertugas mengelola tempat pertemuan dan mengatur penggunaan aset fisik kantor lainnya. Sebelum sistem otomatis ini ada, reservasi ruangan masih dilakukan dengan cara tradisional dan manual, yang sering menimbulkan masalah seperti bentrok jadwal (*double booking*) dan sulitnya memeriksa ketersediaan ruangan secara langsung bagi semua karyawan[4].

Sistem ini membuatnya lebih mudah bagi karyawan untuk mengajukan permohonan peminjaman ruangan melalui antarmuka yang terhubung dengan layanan *Google Workspace*. Data yang diajukan akan diproses secara otomatis oleh *Google Apps Script* untuk memeriksa jadwal dan menyimpan rekaman data di dalam database pusat. Dengan adanya sistem ini, seluruh catatan penggunaan fasilitas tersimpan dengan rapi, sehingga transparansi pemantauan aset meningkat dan kesalahan dalam pencatatan jadwal berkurang.

Dari segi teknis, pengembangan sistem ini mengikuti langkah yang teratur, dari analisis kebutuhan hingga integrasi terakhir ke dalam ekosistem digital perusahaan. Proses utamanya meliputi pemeriksaan input untuk menghindari jadwal yang tumpang tindih, pengiriman notifikasi otomatis, dan pembaruan status ketersediaan ruangan secara langsung. Keberadaan sistem ini sangat membantu tim GA untuk memastikan pembagian penggunaan fasilitas kantor lebih teratur, tepat, dan mendukung kelancaran operasional sehari-hari di PT Dayalima Abisatya.



Gambar 3.1 Logo Google Apps Script

Dalam upaya memperkuat transformasi digital di lingkungan PT Dayalima Abisatya, pemilihan instrumen teknologi menjadi faktor krusial yang menentukan keberhasilan efisiensi operasional. *Google Apps Script* hadir sebagai solusi utama yang bertindak sebagai infrastruktur pusat atau "jembatan" penghubung bagi seluruh layanan *Google Workspace* yang digunakan oleh perusahaan[5]. Sebagai bahasa pemrograman berbasis cloud yang ringan namun bertenaga, logo *Google Apps Script* yang dapat dilihat pada Gambar 3.1 dapat memungkinkan adanya komunikasi dua arah antar platform yang berbeda, seperti *Google Sheets*, *Google Form*, *Google Calendar*, dan *Gmail*. Dengan menggunakan *script* ini, seluruh layanan tersebut tidak lagi beroperasi sebagai entitas yang terpisah secara parsial, melainkan menjadi satu kesatuan ekosistem digital yang bekerja secara otomatis, terstruktur, dan minim intervensi manual[6].

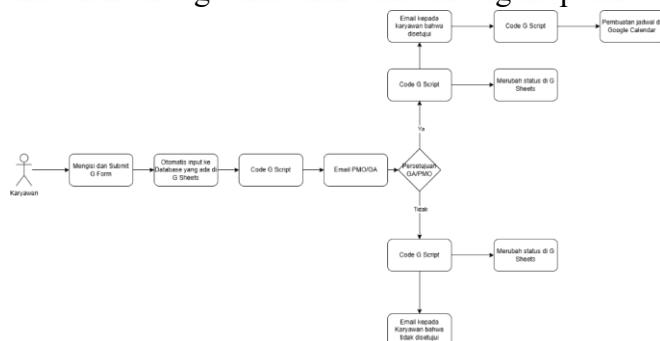
Peran integrasi ini dimulai sejak tahap awal pengumpulan data melalui Google Form. Dalam sistem booking ruangan yang dikembangkan, Google Form berfungsi sebagai gerbang utama bagi karyawan untuk memasukkan permohonan peminjaman fasilitas secara seragam. Begitu data dikirimkan, *Google Apps Script* secara instan menangkap input tersebut melalui mekanisme trigger otomatis. Proses ini jauh lebih efektif dibandingkan cara lama yang masih mengandalkan pesan singkat atau formulir fisik, karena skrip akan langsung melakukan validasi awal terhadap kelengkapan informasi sebelum data tersebut diteruskan ke langkah berikutnya dalam sistem. Hal ini memastikan bahwa data yang masuk sejak awal sudah memiliki standar kualitas yang baik untuk diproses lebih lanjut.

Setelah data berhasil divalidasi, *Google Apps Script* kemudian mengarahkan aliran informasi tersebut menuju *Google Sheets* yang berperan sebagai *database* terpusat. Di sini, skrip menjalankan fungsi yang lebih kompleks, yakni melakukan pemeriksaan silang terhadap jadwal yang sudah ada untuk mencegah terjadinya tumpang tindih atau *double booking*. *Script* secara otomatis mengatur penulisan data ke dalam kolom-kolom spesifik, merapikan formatnya, dan menjaga agar informasi tetap konsisten tanpa ada data ganda yang dapat menghambat proses administrasi tim *General Affairs* (GA). Melalui pemanfaatan *Google Sheets* sebagai

basis data yang dikelola oleh *script*, perusahaan memiliki rekam jejak digital yang transparan dan dapat diaudit sewaktu-waktu, yang merupakan peningkatan signifikan dari sistem pencatatan manual sebelumnya.

Konektivitas sistem ini mencapai puncaknya pada integrasi dengan *Google Calendar* dan *Gmail*. Ketika *script* mendeteksi bahwa slot waktu ruangan yang diminta masih tersedia di *database*, *Google Apps Script* akan secara otomatis membuat agenda pertemuan baru di *Google Calendar* milik departemen terkait. Penjadwalan otomatis ini menjamin akurasi waktu yang tinggi dan meminimalisir kesalahan manusia dalam memasukkan jadwal pemakaian ruangan. Bersamaan dengan terbentuknya jadwal tersebut, *script* memicu *Gmail* untuk mengirimkan notifikasi konfirmasi secara *real-time* kepada pihak pemohon maupun tim GA sebagai pengelola fasilitas. *Email* yang dikirimkan pun memiliki format yang dinamis dan informatif, mencakup rincian waktu, lokasi, hingga tujuan penggunaan ruangan.

Secara keseluruhan, pemanfaatan *Google Apps Script* sebagai pusat kendali teknologi di PT Dayalima Abisatya telah mengubah paradigma kerja dari proses manual yang lambat menjadi alur kerja digital yang cepat dan terukur. Integrasi yang mulus antar layanan *Google Workspace* ini tidak hanya mempercepat proses pembukuan bulanan, tetapi juga memberikan transparansi monitoring yang lebih baik bagi seluruh karyawan dan manajemen. Dengan dukungan *script* yang benar, sistem pemesanan fasilitas dan pengelolaan data perusahaan dapat berjalan secara sinkron, akurat, dan mampu mendukung kelancaran operasional harian seluruh staf secara berkelanjutan sesuai dengan visi transformasi digital perusahaan.



Gambar 3.2 Flow Penggunaan Booking Ruangan

Dari gambar 3.2 yang menunjukkan flow penggunaan sistem booking ruangan dapat dijabarkan bahwa mekanisme operasional dari sistem booking ruangan yang dikembangkan di PT Dayalima Abisatya dirancang untuk menciptakan efisiensi maksimal dengan meminimalkan hambatan birokrasi manual. Alur kerja sistem ini dimulai secara fundamental ketika seorang karyawan mengakses dan mengisi data melalui antarmuka *Google Form* yang telah distandarisasi. Penggunaan *Google Form* sebagai gerbang utama memastikan bahwa setiap informasi yang masuk, baik itu terkait permohonan peminjaman ruangan maupun kebutuhan operasional lainnya, memiliki format yang seragam dan terstruktur sejak awal. Begitu pengguna menekan tombol kirim, data tersebut tidak hanya sekadar tersimpan, melainkan secara instan terelasi ke dalam *database* pusat yang berbasis pada *Google Sheets*. Integrasi langsung ini menghilangkan risiko kehilangan data yang sering terjadi pada sistem pencatatan berbasis kertas atau pesan instan yang tidak terpusat.

Masuknya baris data baru ke dalam *database Google Sheets* bertindak sebagai pemicu utama (*trigger*) bagi infrastruktur *Google Apps Script* untuk memulai serangkaian prosedur logika pemrograman. Pada tahap ini, skrip bekerja di latar belakang untuk melakukan pemrosesan awal, termasuk mengirimkan notifikasi email secara otomatis kepada pihak pengelola, seperti departemen PMO atau *General Affairs* (GA). Notifikasi *real-time* ini sangat krusial agar pihak pengelola dapat segera menyadari adanya permohonan baru yang masuk tanpa harus melakukan pengecekan manual secara berkala pada lembar kerja *database*. Hal ini mempercepat waktu respon organisasi terhadap kebutuhan fasilitas karyawan, yang pada gilirannya akan mendukung produktivitas harian seluruh staf perusahaan.

Selanjutnya, efektivitas sistem ini sangat bergantung pada tahap peninjauan dan keputusan persetujuan yang dilakukan oleh pihak berwenang. Di sinilah *Google Apps Script* menunjukkan perannya sebagai pusat kendali otomatisasi yang dinamis. Jika setelah dilakukan peninjauan pengajuan tersebut dinyatakan disetujui, maka sistem akan menjalankan serangkaian perintah berantai yang kompleks namun otomatis. Pertama, sistem akan memperbarui status permohonan di dalam

*Google Sheets* untuk memberikan transparansi monitoring bagi semua pihak terkait. Kedua, skrip akan memerintahkan layanan *Gmail* untuk mengirimkan email konfirmasi resmi kepada karyawan pemohon, yang berisi rincian persetujuan dan instruksi penggunaan fasilitas. Terakhir, sistem secara otomatis akan berinteraksi dengan *Google Calendar* untuk membuat jadwal agenda pertemuan yang sesuai dengan waktu dan lokasi yang telah divalidasi. Sinkronisasi otomatis ke kalender ini adalah langkah kunci untuk mencegah terjadinya tumpang tindih jadwal atau *double booking* yang sering menjadi kendala utama dalam koordinasi aset perusahaan.

Sebaliknya, sistem juga dirancang untuk menangani skenario penolakan pengajuan secara profesional dan sistematis. Jika pihak pengelola memutuskan bahwa permohonan tidak dapat dipenuhi karena alasan tertentu, sistem tidak akan berhenti begitu saja, melainkan akan melakukan prosedur penutupan alur yang rapi. Dalam kondisi pengajuan tidak disetujui, *Google Apps Script* akan segera memperbarui status penolakan pada kolom *database* di *Google Sheets* sebagai bentuk dokumentasi dan transparansi riwayat administrasi. Bersamaan dengan itu, sistem akan mengirimkan *email* pemberitahuan otomatis kepada karyawan pemohon yang menginformasikan bahwa permohonannya belum dapat diproses. Pada skenario ini, skrip secara cerdas akan menghentikan alur kerja sehingga tidak terjadi pembuatan jadwal di *Google Calendar*. Hal ini memastikan bahwa kalender perusahaan tetap bersih dari agenda-agenda yang tidak valid atau batal, sehingga ketersediaan ruangan dapat segera diperebutkan kembali oleh pihak lain yang membutuhkan.

Melalui penerapan alur kerja yang sedemikian rupa, PT Dayalima Abisatya berhasil mentransformasi proses administrasi internalnya menjadi lebih responsif dan akurat. Seluruh tahapan, mulai dari input data, validasi, hingga output berupa jadwal kalender dan notifikasi email, terjadi dalam satu kesatuan ekosistem yang kohesif. Tidak hanya mempermudah tim GA dalam mengatur distribusi fasilitas kantor, sistem ini juga memberikan pengalaman pengguna yang lebih baik bagi karyawan karena adanya kepastian status permohonan yang dapat dipantau secara

langsung. Implementasi *Google Apps Script* sebagai "jembatan" utama benar-benar telah mengubah cara perusahaan dalam mengelola aset digital dan fisiknya, memastikan bahwa setiap aktivitas tetap selaras dengan upaya transformasi digital yang sedang dijalankan perusahaan.

The image shows a screenshot of a Google Form. At the top, there is a logo for 'DAYALIMA' with the subtitle 'A People Strategist Group'. Below the logo, the title of the form is 'Form Booking Ruangan Alamanda Tower'. A descriptive text states: 'Form ini digunakan untuk meminjam ruangan yang tersedia di Alamanda Tower dan akan di Acc oleh GA'. A note at the bottom of the form area indicates '\* Indicates required question'. The first field is labeled 'Nama Karyawan\*' and has a placeholder 'Your answer' below it. The entire form is contained within a light-colored box with a dark border.

Gambar 3.3 Tampilan G Form Booking Ruangan Kantor Alamanda

Dalam proses membuat antarmuka untuk sistem pemesanan ini yang dapat dilihat pada Gambar 3.3 yaitu menggunakan *Google Form*, pembuatan kolom input dilakukan dengan sangat hati-hati untuk memastikan bahwa informasi yang diperoleh adalah penting namun tetap singkat untuk pengguna[6]. Struktur formulir mencakup data penting seperti nama karyawan yang harus diisi, pilihan tipe ruang yang ada, penentuan tanggal penggunaan, dan lama waktu penggunaan fasilitas. Setiap data ini memiliki fungsi penting dalam database perusahaan untuk memastikan akurasi administratif dan keaslian laporan penggunaan aset di akhir periode.

Desain antarmuka ini sengaja dibuat dengan gaya yang sederhana dan menggunakan elemen yang sudah dikenal oleh semua staf di PT Dayalima Abisatya. Pendekatan ini diterapkan agar karyawan dari berbagai bagian, tanpa memikirkan kemampuan teknisnya, bisa melakukan pemesanan sendiri dan cepat

tanpa perlu belajar dari panduan teknis yang rumit atau mengikuti pelatihan khusus. Dengan menghilangkan kesulitan teknis bagi pengguna, tingkat penerimaan sistem otomatis ini dalam lingkungan internal perusahaan dapat meningkat secara signifikan sejak awal penerapannya.

Tujuan utama dari desain ini adalah untuk mengutamakan kemudahan penggunaan dalam kegiatan sehari-hari yang dinamis. Dalam hal produktivitas, antarmuka yang sederhana namun fungsional memungkinkan proses pemesanan selesai hanya dalam beberapa menit, sehingga waktu kerja staf tidak terbuang untuk urusan administratif yang rumit. Selain itu, desain yang bersih ini membantu mengurangi kemungkinan kesalahan saat mengisi data yang sering terjadi jika formulir terlalu rumit, sehingga informasi yang masuk ke dalam sistem tetap bersih dan siap diproses oleh Google Apps Script.

Lebih dari itu, kemudahan penggunaan ini adalah bagian dari komitmen untuk mendukung perubahan digital yang inklusif di PT Dayalima Abisatya. Sistem yang dibuat harus bisa menjawab kebutuhan operasional tim General Affairs (GA) sambil memberikan kemudahan akses bagi karyawan. Dengan menciptakan alur kerja yang mudah dimengerti mulai dari pengisian formulir, perusahaan tidak hanya mendapatkan sistem yang otomatis dari segi teknis, tetapi juga menciptakan budaya kerja yang lebih efisien dan terorganisir melalui penggunaan teknologi yang lebih ramah bagi pengguna.

Form Response	Timestamp	Nama Karyawan	Email Karyawan	Nama Ruangan	Tanggal Pemesanan	Waktu Mulai	Waktu Selesai	Kegiatan	Status	Event ID
2	8/6/2025 18:16:46	Akila Fachih	akila.fachih@dayalima.id	Raya	8/7/2025	9:00:00 AM	9:00:00 PM	Internal Meeting Trilind	Menunggu Persetujuan	910016548ba0ca08ff
3	8/15/2025 9:19:50	Ayu HC	ayu.haryani@dayalima.id	Kavanna	8/15/2025	10:00:00 AM	10:00:00 PM	Konferensi DIL + x HC	Ditunda	910016548ba0ca08ff
4	8/15/2025 10:45:45	Ridho Yulu Rahmadi	ridho.yulu@dayalima.id	Yema	8/16/2025	1:00:00 PM	2:00:00 PM	Meeting DOD + Prepre 1	Menunggu Persetujuan	910016548ba0ca08ff
5	8/16/2025 13:02:22	Iqbal	tasha.safri@dayalima.id	Indra	8/18/2025	9:00:00 AM	10:00:00 AM	GA training	Ditunda	910016548ba0ca08ff
6	8/26/2025 12:30:19	Ridho Yulu Rahmadi	ridho.yulu@dayalima.id	Banker	9/2/2025	8:00:00 AM	10:00:00 AM	Prepare Town Hall Untuk	Ditunda	910016548ba0ca08ff
7	8/27/2025 19:07:01	Ridho Yulu Rahmadi	ridho.yulu@dayalima.id	Surya/Aquarium	8/29/2025	3:00:00 PM	5:00:00 PM	Gaudi Bank Town Hall M.	Ditunda	910016548ba0ca08ff
8	9/3/2025 20:53	Yoga	choule.yoga@dayalima.id	Banker	9/8/2025	12:00:00 PM	3:00:00 PM	DMH Weekly Meeting	Ditunda	910016548ba0ca08ff
9	9/3/2025 14:02:54	Ridho Yulu	ridho.yulu@dayalima.id	Banker	9/9/2025	7:00:00 AM	11:00:00 AM	Muang khusus MC Town	Ditunda	910016548ba0ca08ff

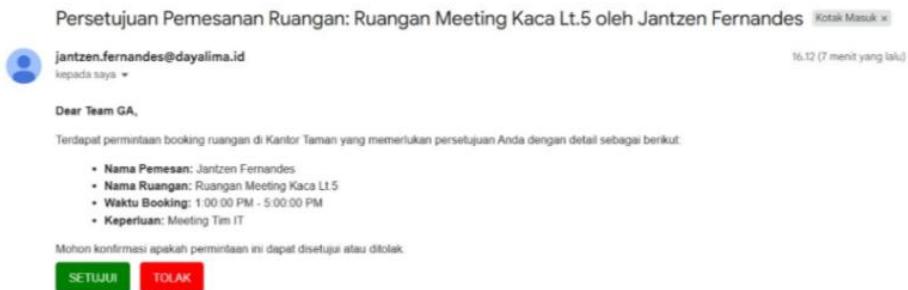
Gambar 3.4 Tampilan Spreadsheet pemesanan ruangan

Sistem ini menjadikan *Google Sheets* sebagai bagian penting dari infrastruktur, di mana perannya bukan hanya sebagai lembar kerja biasa, tetapi juga berfungsi sebagai pusat data utama untuk menyimpan semua catatan tentang aktivitas perusahaan sesuai pada Gambar 3.4 yang menunjukkan Spreadsheet sebagai database dari data pengisian form pemesanan ruangan. Begitu karyawan

menekan tombol untuk mengirim formulir reservasi, semua informasi yang dimasukkan akan terekam dengan rapi, secara otomatis, dan dalam urutan waktu yang tepat ke dalam baris-baris di Google Sheets[6]. Koneksi langsung ini sangat efisien, karena pengelolaan data tidak perlu lagi membuat tabel secara manual atau menyalin data berkali-kali yang bisa membuat kesalahan. Setiap permintaan peminjaman akan segera mengisi baris baru, menjamin bahwa data mentah tersimpan utuh dan aman.

Dalam membangun struktur data ini, kolom-kolom yang tersedia tidak hanya mengenai data dasar seperti nama karyawan, pilihan ruangan, atau lama waktu peminjaman. Ada satu kolom penting yang disebut "Status Persetujuan" yang berfungsi sebagai tanda untuk memantau proses administrasi. Kolom ini diatur untuk berfungsi secara otomatis melalui sistem skrip, di mana status dari permohonan akan diperbarui secara otomatis mulai dari "Menunggu", lalu menjadi "Disetujui", atau bisa juga "Ditolak" tergantung hasil pemeriksaan pihak berwenang. Pengaturan status yang otomatis ini sangat penting untuk kelancaran kerja tim administrasi, karena memberikan gambaran yang jelas mengenai penggunaan fasilitas selama periode tertentu dengan transparansi.

Penggunaan Google Sheets sebagai catatan utama juga sangat mendukung proses pengawasan dan pelaporan jangka panjang. Dengan data yang tersusun rapi, admin atau manajemen bisa dengan mudah melacak riwayat peminjaman jika suatu saat dibutuhkan untuk evaluasi atau audit internal perusahaan. Keterbukaan data yang ditawarkan memungkinkan tim General Affairs (GA) untuk melihat pola penggunaan ruangan, sehingga mereka bisa membuat keputusan yang lebih baik tentang pengelolaan aset kantor. Secara keseluruhan, penggunaan Sheets sebagai bagian belakang dari sistem ini membuat akses lebih mudah untuk staf IT dan admin tanpa perlu perangkat lunak database yang rumit, tetapi tetap dapat mendukung kelancaran operasional sehari-hari di PT Dayalima Abisatya.



Gambar 3.5 Tampilan Approval Pemesanan di Gmail General Affairs

Bagian ini adalah pusat dari inovasi sistem otomatisasi yang telah dibuat, dengan tujuan utama untuk meningkatkan efisiensi bagi para pengambil keputusan. Dalam sistem kerja yang lama, tim General Affairs (GA) biasanya harus memeriksa lembar kerja spreadsheet secara manual atau membuka laptop setiap kali mereka menerima permintaan untuk memastikan aset tersedia. Namun, dengan hadirnya sistem baru ini, proses tersebut telah diubah menjadi lebih cepat melalui pengiriman Email Notifikasi Cerdas yang langsung masuk ke kotak masuk tim pengelola.

Di Gambar 3.5, terlihat tampilan email notifikasi, di mana sistem ini tidak hanya mengirimkan pesan teks biasa, tapi juga ringkasan data yang lebih lengkap dan terorganisir. Di dalam email tersebut, terdapat informasi penting dari permintaan, seperti identitas pemohon, jenis ruangan yang dipilih, dan waktu pemakaian yang diminta. Salah satu fitur unggulan dari sistem ini adalah adanya tombol aksi interaktif atau Action Buttons yang bisa digunakan langsung di dalam email.

Ada dua tombol utama yang bisa dipilih oleh tim GA, yakni tombol "SETUJUI" berwarna hijau dan tombol "TOLAK" berwarna merah. Kedua tombol ini terhubung langsung dengan skrip di belakang layar menggunakan teknologi Google Apps Script. Ketika salah satu tombol ditekan, sistem akan secara otomatis menjalankan perintah untuk memperbarui database, mengirimkan balasan kepada pemohon, dan menjadwalkan di kalender tanpa perlu admin berpindah aplikasi atau membuka dokumen lain.

Proses ini secara efektif mengurangi birokrasi yang sebelumnya rumit dan memakan waktu. Keputusan manajerial kini bisa diambil dalam hitungan detik

dengan proses yang sangat mudah. Fleksibilitas ini memungkinkan tim GA untuk terus melakukan pengawasan dan persetujuan kapan saja dan di mana saja, bahkan hanya dengan smartphone. Dengan demikian, koordinasi dalam menggunakan fasilitas di PT Dayalima Abisatya menjadi lebih dinamis, transparan, dan mendukung mobilitas tinggi bagi staf di zaman digital ini.

```
const SPREADSHEET_ID = '1vzXKk_21Z0fyWB6jPORuIx3xam0Z1s9VkuEhy4AGzw'; // ID Spreadsheet
const CALENDAR_ID = 'c_4f2bcae21f7e042f00a2ad3b81c9beae71e73873e1c205e43a90880793d6b7b2@group.calendar.google.com'; // Calendar ID
const MANAGER_GA_EMAILS = ['gaservices@dayalima.id']; // Email untuk approval
const SHEET_Pemesanan = 'Form Responses 1'; // Nama sheet
const WEB_APP_URL = 'https://script.google.com/macros/s/AKfycbxSEVKB0AurR0qRPfnXICTlsRWgk2ndfPgd8G6L56NrrnEME2sPjNivDFP-UE9EYc0Qtw/exec'; // Web App URL
```

Gambar 3.6 Code bagian Konfigurasi

Tahap pertama dalam membuat kode program ini berfokus pada pengaturan dan pemulai, yang secara teknis berfungsi sebagai dasar dari seluruh sistem skrip yang dibuat. Di bagian ini, ditetapkan variabel-variabel tetap yang menyimpan "alamat" digital untuk setiap bagian layanan Google Workspace, agar sistem dapat menavigasi dengan benar dan menghindari kesalahan saat menjalankan perintah. Tanpa adanya peta yang jelas di tahap awal ini, skrip tidak akan tahu ke mana data harus dikirim atau layanan mana yang harus diaktifkan otomatis.

Seperti yang terlihat di Gambar 3. 6 mengenai kode konfigurasi, bagian ini bisa dibandingkan dengan "buku alamat" sentral yang memastikan program memiliki referensi yang tepat saat mulai beroperasi di belakang layar. Dalam kode tersebut, ada identitas-identitas penting yang terhubung satu sama lain. Pertama, ada ID Spreadsheet yang memberikan arahan spesifik tentang lokasi basis data utama tempat semua permohonan disimpan. Kedua, ada ID Kalender sebagai rujukan bagi sistem untuk mencatat setiap agenda yang telah disetujui ke kalender perusahaan dengan tepat.

Di samping itu, aspek koordinasi manusia juga dikaitkan langsung melalui penentuan Email Manager (gaservices@dayalima. id), yang punya otoritas tunggal untuk menerima pemberitahuan dan memberikan persetujuan terhadap setiap permohonan yang masuk. Kode ini juga secara khusus mengarahkan program untuk mengambil data dari lembar kerja bernama "Form Responses 1", serta menyertakan

Link Website agar sistem ini dapat diakses secara fungsional sebagai aplikasi web yang responsif.

Bagian konfigurasi ini bukan hanya sekedar kumpulan kode teknis, tetapi juga sebuah titik menghubungkam *Google Form* pemesanan ruangan, *Database Google Sheets* penjadwalan Kalender di *Google Calendar*, dan *Gmail Approval* dalam satu saluran komunikasi yang terintegrasi. Dengan menetapkan variabel-variabel ini dari awal, pemeliharaan sistem di masa depan menjadi jauh lebih mudah; jika ada perubahan pada alamat email admin atau lokasi database, perubahannya cukup dilakukan di satu titik konfigurasi ini tanpa harus merombak seluruh logika program. Hal ini memastikan bahwa sistem otomasi di PT Dayalima Abisatya memiliki fleksibilitas dan stabilitas yang tinggi untuk mendukung operasional harian yang dinamis.

```
function onFormSubmit(e) {
  if (!e || !e.namedValues) return;

  const data = e.namedValues;
  const nama = data['Nama Karyawan']?.[0] || "";
  const email = data['Email Karyawan']?.[0] || "";
  const ruangan = data['Nama Ruangan']?.[0] || "";
  const tanggal = data['Tanggal Pemakaian']?.[0] || "";
  const mulai = data['Waktu Mulai']?.[0] || "";
  const selesai = data['Waktu Selesai']?.[0] || "";
  const keperluan = data['Keperluan']?.[0] || "";

  const startTime = new Date(`${tanggal} ${mulai}`);
  const endTime = new Date(`${tanggal} ${selesai}`);
  if (isNaN(startTime) || isNaN(endTime)) return;

  const calendar = CalendarApp.getCalendarById(CALENDAR_ID);
  const event = calendar.createEvent(`[MENUNGGU] ${ruangan} - ${keperluan}`, startTime, endTime, {
    description: `Dipesan oleh: ${nama} (${email})\nKeperluan: ${keperluan}`,
    guests: email
  });
  event.setColor(CalendarApp.EventColor.GRAY);
```

Gambar 3.7 Code Bagian Trigger

Bagian ini adalah elemen penting dalam sistem pemrograman *Google Apps Script* yang berfungsi sebagai pemicu otomatis. Sesuai dengan panduan teknis yang disusun, fungsi *onFormSubmit* ini dibuat untuk langsung bekerja tepat setelah seorang karyawan mengklik tombol kirim pada formulir pemesanan ruang. Keberadaan fungsi ini memastikan bahwa tidak ada jeda waktu antara memasukkan data dan pemrosesan sistem, sehingga alur kerja tetap berjalan secara langsung tanpa perlu dilakukan secara manual oleh tim IT.

Seperti yang ditunjukkan dalam Gambar 3. 7 tentang struktur kode bagian pemicu, fungsi ini bisa diibaratkan sebagai "resepsonis otomatis" yang bertugas menerima dan mengatur setiap data yang masuk. Tugas utamanya dimulai dengan mengambil dan merapikan informasi dari formulir; sistem ini akan menentukan siapa yang memesan, mengidentifikasi ruang yang dipilih, serta memeriksa rincian waktu penggunaan seperti tanggal dan jam. Ketepatan dalam mengolah data ini sangat penting agar langkah sinkronisasi berjalan dengan baik.

Begitu data divalidasi dan dianggap lengkap, sistem akan mengambil tindakan pencegahan dengan segera membuat slot jadwal di Google Calendar. Namun, karena sistem ini mengikuti prinsip kontrol manajerial, jadwal ini tidak langsung dianggap final. Secara teknis, skrip akan memberikan tanda awal berupa "[MENUNGGU]" dan mengubah warna label acara tersebut menjadi abu-abu. Penandaan visual ini sangat penting bagi tim *General Affairs* dan karyawan lain yang melihat kalender, karena mengindikasikan bahwa ruang tersebut telah dipesan (di-tag), namun statusnya masih dalam antrean atau menunggu persetujuan resmi.

Selain menambahkan tanda pada kalender umum, sistem juga akan secara otomatis memasukkan alamat email pemesan sebagai undangan dalam jadwal tersebut. Tujuannya adalah agar karyawan yang bersangkutan menerima pemberitahuan langsung di kalender pribadi mereka sebagai konfirmasi bahwa permohonan mereka telah diterima dan sedang diproses. Dengan cara ini, transparansi dalam pemantauan ketersediaan ruang tetap terjaga dengan baik, sekaligus memberikan kepastian bagi pemohon bahwa waktu yang mereka inginkan sudah diamankan sementara dalam database perusahaan.

```

function doGet(e) {
  const { action, eventId, statusCell, requesterEmail, requesterName, ruangan, jam, keperluan } = e.parameter;

  if (!eventId || !statusCell) {
    return ContentService.createTextOutput("Parameter tidak lengkap.");
  }

  const calendar = CalendarApp.getCalendarById(CALENDAR_ID);
  const sheet = SpreadsheetApp.openById(SPREADSHEET_ID).getSheetByName(SHEET_Pemesanan);

  let event;
  try {
    event = calendar.getEventById(eventId);
  } catch (err) {
    return ContentService.createTextOutput("Event tidak ditemukan atau rusak.");
  }

  if (!sheet || !event) {
    return ContentService.createTextOutput("Data tidak valid.");
  }

  if (action === 'approve') {
    event.setTitle(event.getTitle().replace('[MENUNGGU]', '[DISETUJUI]'));
    event.setColor(CalendarApp.EventColor.GREEN);
    event.setDescription(event.getDescription() + "\nStatus: Disetujui");
  }
}

```

Gambar 3.8 Code bagian doGet

Tahapan ini adalah bagian yang sangat penting dalam sistem, di mana skrip bekerja untuk mengolah keputusan akhir yang diambil oleh manajemen melalui email. Fungsi doGet(e) pada Gambar 3.8 berperan sebagai penerima sinyal digital yang dikirim saat tombol aksi ("SETUJUI" atau "TOLAK") diklik di inbox email. Kode ini dibuat untuk memastikan bahwa semua parameter yang dikirim—seperti ID acara kalender, alamat sel status, hingga informasi pemesan—diterima dengan lengkap sebelum proses perubahan data dimulai. Jika ada data yang hilang atau rusak, sistem secara otomatis akan menolak permintaan untuk menjaga integritas database.

Keunggulan utama dari blok kode ini adalah kemampuannya untuk menyinkronkan data secara langsung antara kalender dan database pusat. Ketika perintah 'approve' (setuju) diterima, skrip akan mencari agenda yang sebelumnya memiliki status "[MENUNGGU]" di Google Calendar, dan secara otomatis mengubah judulnya menjadi "[DISETUJUI]". Selain mengganti teks, sistem juga memberi tanda visual yang jelas dengan mengubah warna label agenda dari abu-abu menjadi hijau (GREEN). Perubahan warna ini memudahkan tim operasional untuk dengan cepat mengetahui ruangan mana yang sudah resmi dipesan dan mana yang masih dalam antrean.

Selain memperbarui tampilan kalender, skrip ini juga bertugas memperbarui "catatan dalam" atau deskripsi acara untuk kebutuhan audit di masa depan. Sistem

akan menambahkan baris status baru di deskripsi kalender untuk mencatat bahwa permohonan telah melewati tahap validasi manajemen. Dengan adanya program yang terstruktur ini, seluruh proses birokrasi yang dulunya manual dan lambat kini dapat diselesaikan dalam hitungan detik. Hal ini memastikan bahwa ketersediaan aset perusahaan terpantau dengan tepat, meminimalisir kesalahan manusia, dan menjamin bahwa seluruh aktivitas di PT Dayalima Abisatya berjalan lebih efisien dan teratur.

Sebagai langkah terakhir yang sangat penting dalam serangkaian program magang ini, ada pemahaman mendalam bahwa kualitas sebuah sistem tidak hanya terlihat dari kecanggihan fitur teknis yang ada saat ini, tetapi juga dari seberapa baik sistem bisa berjalan di masa mendatang. Mengingat waktu kegiatan magang memiliki batasan tertentu, telah diambil langkah proaktif untuk memastikan bahwa sistem otomatisasi yang dibangun tidak ditinggalkan atau mengalami masalah teknis yang tidak bisa diatasi setelah magang selesai. Sebagai bentuk tanggung jawab profesional, dibuatlah sebuah dokumen lengkap yang berupa Buku Panduan Teknis yang dirancang khusus untuk membantu perusahaan dalam mentransfer pengetahuan.

Buku panduan ini disusun dengan cara yang sangat rinci tetapi tetap mudah dipahami, dengan membagi fokus bahasannya ke dalam dua sudut pandang pengguna yang berbeda untuk memastikan sistem dapat diadopsi dengan baik. Sudut pandang pertama adalah Panduan untuk Pengguna Umum yang ditujukan bagi semua karyawan. Bagian ini memuat tutorial praktis tentang cara menggunakan sistem dalam aktivitas sehari-hari, menjelaskan langkah demi langkah mulai dari cara mengakses tautan formulir, prosedur untuk mengisi data dengan benar, sampai cara melihat status ketersediaan ruangan di kalender perusahaan. Metode ini bertujuan agar setiap staf, termasuk karyawan baru, dapat dengan mudah memahami cara kerja sistem tanpa perlu pelatihan teknis yang memakan banyak waktu.

Sudut pandang kedua dalam dokumen ini adalah Panduan Teknis untuk Administrator, yang khusus ditujukan bagi tim General Affairs (GA) atau PMO

yang bertanggung jawab atas pengelolaan fasilitas. Bagian ini lebih mendalam dan teknis karena menjelaskan semua alur kerja di balik layar, termasuk cara mengelola database di Google Sheets, prosedur pemecahan masalah jika ada masalah dengan pengiriman notifikasi email, hingga pemahaman dasar tentang logika pemrograman Google Apps Script yang mengoperasikan semua otomatisasi. Dokumen ini berfungsi sebagai "buku resep" teknis yang memungkinkan administrator mengelola dan mengembangkan sistem secara mandiri.

Tujuan utama dari penyusunan Buku Panduan Teknis ini adalah memastikan bahwa proses transfer pengetahuan berlangsung dengan baik dan jelas. Ini sejalan dengan komitmen untuk memberikan PT Dayalima Abisatya kemandirian dalam mengelola aset digital barunya. Dengan adanya panduan yang lengkap, tim internal perusahaan tidak akan bergantung pada pengembang awal, sehingga sistem pemesanan ruangan ini bisa terus berfungsi dengan baik dalam jangka panjang untuk mendukung kelancaran operasional sehari-hari seluruh staf.

Upaya dokumentasi ini juga menunjukkan tahap akhir dalam diagram alur pelaksanaan magang, di mana pembuatan buku panduan dan presentasi kepada rekan kerja menjadi pengantar sebelum sistem mulai dalam tahap pemeliharaan rutin. Dengan dokumentasi yang teratur, setiap perbaikan atau perubahan yang mungkin diperlukan di kemudian hari dapat dilakukan dengan merujuk pada logika yang telah ditetapkan, sehingga transformasi digital yang sudah dimulai dapat memberikan manfaat yang berkelanjutan bagi produktivitas perusahaan.



Gambar 3.9 PPT Handbook

Setelah semua proses pengembangan kode dan uji coba internal selesai, proyek otomatisasi ini masuk ke tahap penting terakhir, yaitu sosialisasi dan penerapan di lapangan. Ada kesadaran bahwa sebaik apapun sistem yang dibuat, manfaatnya tidak akan maksimal jika pengguna akhir tidak tahu cara mengoperasikannya dengan benar. Oleh karena itu, diambil langkah untuk secara resmi memperkenalkan sistem baru ini kepada seluruh karyawan di PT Dayalima Abisatya agar transisi dapat berlangsung dengan baik.

Langkah pertama yang dilakukan adalah mengadakan sesi presentasi untuk karyawan dan manajemen perusahaan. Dalam pertemuan tersebut, materi yang disampaikan tidak hanya fokus pada aspek teknis yang rumit, tetapi lebih pada menjelaskan alur kerja sistem dari sudut pandang pengguna sehari-hari. Penjelasan termasuk bagaimana peralihan dari metode manual lama ke sistem otomatis baru ini akan sangat membantu dalam meningkatkan efisiensi kerja mereka. Tujuan utama adalah untuk memastikan semua orang memiliki pemahaman yang sama dan meyakinkan tim bahwa sistem baru ini aman, mudah digunakan, dan sangat efisien untuk jangka panjang.

Namun, hanya memberikan penjelasan teori tidak akan cukup untuk memberi jaminan penuh. Sebagai bukti nyata, agenda dilanjutkan dengan sesi demonstrasi langsung. Ini menjadi momen untuk menunjukkan kepada audiens bagaimana sistem ini digunakan secara langsung. Proses dimulai dengan pengisian Google Form, di mana ditunjukkan dengan jelas bagaimana data dengan cepat menjadi notifikasi email di kotak masuk admin. Selanjutnya, diperlihatkan juga bagaimana pengelola bisa memberikan persetujuan hanya dengan satu klik, dan akhirnya jadwal muncul secara otomatis di Google Calendar tanpa ada jeda.

Simulasi yang dilakukan tanpa rekayasa ini berhasil menunjukkan bahwa integrasi fitur Google Workspace menggunakan Google Apps Script yang telah dibuat berfungsi dengan baik dan stabil. Keberhasilan demonstrasi ini memberikan keyakinan pada perusahaan bahwa sistem sudah siap sepenuhnya untuk digunakan dalam kegiatan sehari-hari untuk mendukung kelancaran penggunaan fasilitas kantor. Setelah tahap sosialisasi ini selesai, tanggung jawab sebagai pengembang

sistem juga termasuk memastikan bahwa setiap umpan balik yang muncul selama demo langsung ditanggapi untuk penyempurnaan sistem sebelum pemeliharaan rutin dimulai.

Implementasi ini juga bagian dari strategi perusahaan untuk memperkuat posisi sebagai konsultan SDM yang responsif dalam menggunakan teknologi digital. Dengan sistem yang terpusat dan otomatis, kini transparansi pemantauan fasilitas untuk IT dan karyawan dapat terwujud, mengurangi kesalahan pencatatan manual yang sebelumnya menjadi masalah operasional. Tahap akhir ini menandakan keberhasilan proses digitalisasi internal yang tidak hanya mengandalkan kode program yang kuat, tetapi juga partisipasi aktif dari semua elemen organisasi untuk bergerak bersama menuju efisiensi kerja yang lebih modern.

Setelah tahap pelaksanaan selesai dan sistem pemesanan ruang telah resmi diluncurkan untuk digunakan oleh semua pegawai di PT Dayalima Abisatya, peran yang diambil beralih dari pengembang sistem menjadi pengelola operasional atau dukungan teknis. Diketahui dengan jelas bahwa meluncurkan aplikasi bukanlah akhir dari segalanya, melainkan awal dari fase penting berikutnya, yaitu pemeliharaan. Fase ini sangat penting untuk menentukan apakah sistem yang dirancang dapat bertahan dengan baik dan berjalan mulus di tengah kesibukan kantor, atau malah menimbulkan masalah baru yang bisa mengganggu kerja staf.

Dalam menjalankan peran pemeliharaan teknis, dilakukan pemeriksaan rutin terhadap kesehatan sistem pada ekosistem Google Workspace yang dipakai. Fokus utama berada pada Google Apps Script Dashboard untuk mengawasi aktivitas skrip yang berjalan di belakang layar. Secara berkala, menu Execution Logs diperiksa agar bisa dipastikan apakah skrip berfungsi dengan baik atau mengalami kegagalan. Ini sangat penting untuk mengantisipasi masalah yang tak terduga, seperti gangguan di server Google atau adanya kemungkinan melebihi batas kuota harian untuk pengiriman email otomatis. Selain itu, pengelolaan basis data di Google Sheets juga dilakukan secara teratur. Mengingat Google Sheets memiliki batas jumlah sel dan

bisa menjadi lambat jika terlalu banyak data, stabilitas spreadsheet dijaga tetap ringan dan cepat dengan proses pembersihan dan pengarsipan data lama.

Di sisi lain, tantangan terbesar pada fase ini sering kali datang dari aspek non-teknis, yaitu adaptasi pengguna. Sebagai garda terdepan dalam memberikan dukungan teknis, saluran komunikasi khusus dibuka untuk menampung semua keluhan, pertanyaan, dan masalah yang dihadapi karyawan. Pada minggu-minggu pertama setelah peluncuran, banyak laporan dari pengguna yang merasa bingung atau mengalami masalah teknis. Beberapa keluhan yang sering ditangani meliputi masalah dengan notifikasi email, di mana setelah dilakukan pemeriksaan ditemukan bahwa masalah sering kali disebabkan oleh email yang masuk ke folder Spam atau kesalahan penulisan alamat email saat mengisi formulir, bukan karena kesalahan pada sistem utama.

Masalah lain yang umum muncul adalah kebingungan mengenai status jadwal di Google Calendar. Ada pengguna yang bertanya mengapa jadwal mereka belum muncul meskipun formulir telah dikirim dari pagi. Dalam keadaan ini, dilakukan edukasi mengenai alur bisnis untuk menjelaskan kepada pengguna bahwa sistem ini menggunakan logika Persetujuan, sehingga jadwal baru hanya akan muncul secara otomatis setelah Admin GA menekan tombol "Setuju". Selain itu, kesalahan dalam format waktu (seperti perbedaan antara AM/PM atau format 24 jam) juga sering menjadi masalah yang menyebabkan jadwal tidak terdaftar dengan benar.

Menanggapi perubahan ini, penanganan tidak hanya dilakukan dengan menyelesaikan masalah satu per satu, tetapi juga melalui perbaikan sistem secara terus-menerus. Berdasarkan masukan dan pola keluhan yang ada, beberapa penyesuaian kecil dilakukan pada tampilan sistem. Contohnya, ditambahkan penjelasan yang lebih jelas pada Google Form untuk membantu cara mengisi jam yang benar, serta mengubah kalimat pada subjek email agar lebih menarik perhatian dan tidak dianggap sebagai spam oleh sistem email. Semua kegiatan pemeliharaan dan penanganan keluhan ini menunjukkan komitmen untuk memastikan bahwa transisi teknologi di PT Dayalima Abisatya berjalan dengan baik, di mana sistem

tidak hanya baik dari segi teknis, tetapi juga ramah dan memberikan solusi bagi semua penggunanya.

Proses pemeliharaan ini juga menjadi cara untuk memastikan bahwa sistem tetap sesuai dengan kebutuhan operasional yang mungkin berubah. Dengan pengawasan yang terus menerus, setiap kesalahan data atau celah dalam keamanan dapat ditemukan lebih awal, sehingga keandalan sistem tetap terjaga. Upaya ini dilakukan agar investasi di bidang teknologi yang telah dilakukan oleh perusahaan memberikan manfaat yang maksimal dan berkelanjutan, sekaligus menciptakan rasa aman bagi staf dalam menggunakan fasilitas digital perusahaan setiap hari.

### **3.2.2 Data Cleansing CRM**

Manajemen Hubungan Pelanggan atau CRM adalah sistem dan strategi yang terintegrasi yang digunakan oleh perusahaan untuk mengatur dan menganalisis semua jenis interaksi dengan pelanggan selama mereka bertransaksi[7]. PT Dayalima Abisatya, yang berfokus pada konsultasi sumber daya manusia, melihat CRM tidak hanya sebagai alat teknis, tetapi sebagai cara dasar untuk membangun dan mempertahankan hubungan yang kuat dengan mitra bisnis dan klien besar, seperti PT Antam dan PT Pertamina. Dengan memakai platform CRM yang terpusat, perusahaan dapat mengumpulkan, menyimpan, dan menyelaraskan berbagai informasi pelanggan dalam satu database, sehingga memudahkan semua departemen untuk mendapatkan akses informasi yang konsisten dan jelas.

Penerapan CRM menawarkan berbagai fungsi dan keuntungan yang mendukung kegiatan sehari-hari perusahaan. Pertama, sistem ini membantu mengelola data pelanggan secara menyeluruh, mencakup sejarah interaksi, preferensi layanan, dan catatan keluhan semuanya dalam satu tempat yang teratur. Hal ini memungkinkan tim untuk memahami kebutuhan klien perusahaan dengan lebih mendalam dan memberikan solusi pengembangan sumber daya manusia yang relevan. Kedua, pemahaman yang mendalam ini langsung berdampak pada peningkatan kepuasan dan kesetiaan pelanggan, yang dalam jangka waktu lama mendorong stabilitas penjualan dan pertumbuhan bisnis. Ketiga, fitur otomatis dalam CRM terbukti memudahkan tim dengan mengurangi tugas administratif,

sehingga tenaga kerja bisa lebih fokus pada pekerjaan yang lebih strategis dan kreatif. Akhirnya, CRM juga menyediakan kemampuan analisis dan pelaporan yang memungkinkan manajemen membuat keputusan bisnis yang lebih tepat berdasarkan data yang valid.

Mengingat pentingnya pelanggan strategis seperti PT Antam dan PT Pertamina sebagai aset berharga dalam layanan konsultasi, kualitas data yang ada di CRM menjadi sangat penting. Awalnya, tim penjualan di PT Dayalima Abisatya melakukan pembersihan data secara manual terhadap ribuan baris data. Namun, metode manual ini tidak hanya memakan waktu lama, tetapi juga berisiko tinggi untuk kesalahan manusia, seperti kesalahan dalam menghapus data ganda atau format yang tidak konsisten. Oleh karena itu, dengan adanya program magang TI, mereka melakukan inisiatif untuk mengubah proses pembersihan data ini dengan menggunakan bahasa pemrograman Python.

Penggunaan Python dalam proses pembersihan data ini memberikan peningkatan efisiensi yang besar bagi perusahaan. Dengan memanfaatkan pustaka pemrosesan data seperti Pandas, ribuan baris data CRM yang sebelumnya menjadi beban tim penjualan dapat dibersihkan hanya dalam beberapa detik dengan akurasi yang lebih tinggi[8][9]. Skrip Python yang dibuat mampu secara otomatis menemukan duplikasi, menstandarisasi format nama perusahaan, dan memvalidasi alamat email klien. Perubahan dari cara manual ke otomatis ini memastikan bahwa data yang digunakan untuk berinteraksi dengan pelanggan tetap tepat dan terkini, sehingga memperkuat posisi PT Dayalima Abisatya sebagai konsultan yang terbuka terhadap teknologi digital. Dengan data yang bersih, perusahaan dapat mengoptimalkan setiap interaksi dengan klien dan memastikan bahwa setiap keputusan strategis didukung oleh fakta yang akurat untuk mencapai tujuan bisnis jangka panjang.



```
[15]: import pandas as pd

# 1. Baca file
df = pd.read_excel("merged_clean.xlsx")

# 2. Rapikan nama kolom biar ga ada spasi tersembunyi
df.columns = df.columns.str.strip()

# 3. Normalisasi: treat "-" and "" as kosong
df = df.replace("-", pd.NA)
df = df.replace("", pd.NA)

# 4. Hitung jumlah kosong per baris
df["missing_count"] = df.isna().sum(axis=1)

# 5. Tentukan urutan prioritas kolom
sort_priority = ["missing_count", "Phone_number", "Fax_Number", "Email", "Company_Name"]

# 6. Urutkan
df = df.sort_values(by=sort_priority, ascending=[True]*len(sort_priority))

# 7. Hapus kolom bantu
df = df.drop(columns="missing_count")

# 8. Kalau mau isi kosong jadi "-"
df = df.fillna("-")

# 9. Simpan hasil
df.to_excel("merged_sorted.xlsx", index=False)
print("Data berhasil diurutkan sesuai prioritas")
```

Gambar 3.10 Code Python untuk awal data Cleansing

Dalam lingkungan bisnis PT Dayalima Abisatya yang memperhatikan konsultasi sumber daya manusia, informasi tentang pelanggan amat penting. Beberapa klien besar, seperti PT Antam dan PT Pertamina, memiliki ribuan catatan interaksi dan kontak yang terus berkembang setiap hari dalam sistem manajemen hubungan pelanggan (CRM). Namun, data tersebut sering kali tersimpan dalam keadaan yang kurang baik, dengan banyak pengulangan, format yang tidak konsisten, serta kolom-kolom yang kosong yang dapat mengganggu analisis laporan penjualan. Sebelum data ini dapat digunakan lebih lanjut oleh tim penjualan atau manajemen, proses pembersihan data menjadi langkah yang sangat penting untuk memastikan keakuratan informasi.

Dari sejarah pengelolaan data di perusahaan, terlihat bahwa awalnya tim penjualan perlu membersihkan ribuan catatan tersebut secara manual. Proses yang dilakukan dengan tangan ini tidak hanya membutuhkan waktu yang lama—bisa memakan beberapa hari tetapi juga sangat rentan terhadap kesalahan manusia. Kesalahan kecil dalam menghapus baris atau salah menstandarkan nama perusahaan dapat sangat mempengaruhi akurasi strategi pemasaran. Menyadari adanya masalah ini, melalui program magang IT, dikembangkan solusi berbasis teknologi menggunakan bahasa pemrograman Python dengan bantuan pustaka Pandas. Seperti yang ditunjukkan pada Gambar 3.10, skrip ini dibuat untuk

melakukan pekerjaan yang berulang dengan tingkat akurasi yang tinggi dan kecepatan yang sangat baik.

Tahap pertama dalam logika pemrograman ini adalah membaca file data dari luar secara teratur. Begitu file dimuat, skrip langsung melakukan pembersihan di tingkat sel, yaitu menghapus spasi tersembunyi pada nama kolom. Meskipun ini tampak sepele, spasi yang tidak terlihat sering kali menjadi masalah dalam database yang membuat data tidak tersedia saat dicari. Langkah selanjutnya adalah Normalisasi Data, di mana skrip secara otomatis menemukan karakter yang tidak konsisten, seperti penggunaan tanda strip atau sel yang kosong, dan mengubahnya menjadi format nilai null yang diterima secara teknis. Ini memastikan bahwa database perusahaan bebas dari karakter yang tidak memiliki nilai informatif.

Inovasi utama dalam pengembangan kode ini adalah kemampuannya untuk melakukan Analisis Kualitas Data secara otomatis. Skrip Python ini dirancang untuk menghitung jumlah kolom yang kosong di setiap baris data. Setelah jumlah data yang kosong diketahui, sistem akan mengurutkan berdasarkan prioritas yang telah ditetapkan, seperti adanya nomor telepon, nomor faks, alamat email, dan nama perusahaan. Dengan logika ini, data yang paling lengkap mengenai klien akan berada di urutan teratas, sementara data yang kurang informasi akan terkumpul di bagian bawah atau dipisahkan. Ini memberikan kemudahan yang besar bagi tim penjualan untuk langsung fokus pada prospek klien yang datanya sudah siap untuk dihubungi.

Setelah selesai dengan semua langkah pembersihan dan pengurutan, skrip akan menyelesaikan proses dengan mengisi sel-sel kosong menggunakan karakter standar (seperti "-") agar tampilan akhir di Excel tetap rapi dan terlihat profesional untuk pengguna yang tidak memiliki latar belakang teknis. File akhir kemudian akan disimpan dalam bentuk baru yang sepenuhnya sudah dioptimasi. Perubahan dari metode manual ke otomatisasi menggunakan Python ini bukan hanya soal kecepatan, tetapi juga menyangkut keakuratan data. Sebelumnya, tim penjualan harus berjuang dengan ribuan baris data selama berjam-jam, tetapi sekarang proses

yang sama dapat diselesaikan hanya dalam beberapa detik dengan hasil yang jauh lebih tepat.



```
[17]: import pandas as pd

# 1. Baca file
df = pd.read_excel("merged_clean.xlsx")

# 2. Rapihin nama kolom
df.columns = df.columns.str.strip()

# 3. Normalisasi " " dan "" jadi NaN
df = df.replace(" ", pd.NA).replace("", pd.NA)

# 4. Tambahkan flag kolom (1 kalau ada isi, 0 kalau kosong)
for col in ["Email", "Phone_number", "Fax_Number", "Company_Name"]:
    df[f"has_{col}"] = df[col].notna().astype(int)

# 5. Hitung total kelengkapan
df["filled_count"] = df[["has_Email", "has_Phone_number", "has_Fax_Number", "has_Company_Name"]].sum(axis=1)

# 6. Sort multi-Level (Email dulu, Lalu Phone, Lalu Fax, Lalu Company, Lalu total)
df = df.sort_values(
    by=["has_Email", "has_Phone_number", "has_Fax_Number", "has_Company_Name", "filled_count"],
    ascending=[False, False, False, False, False]
)

# 7. Hapus kolom bantu
df = df.drop(columns=[c for c in df.columns if c.startswith("has_")] + ["filled_count"])

# 8. Balikin NaN jadi "-"
df = df.fillna("-")

# 9. Simpan hasil
df.to_excel("merged_sorted.xlsx", index=False)
print("Data sudah diurutkan: paling lengkap & penting di atas, paling kosong di bawah")
```

Gambar 3.11 Code Python untuk mengurutkan data CRM sesuai kelengkapan

Melanjutkan proses pembersihan data yang sudah dilakukan sebelumnya, Gambar 3.11 menunjukkan langkah-langkah pengkodean yang lebih mendalam untuk memastikan bahwa informasi tentang klien berkualitas tinggi. Jika skrip sebelumnya hanya sebatas menghilangkan "sampah" visual, maka skrip di bagian ini berfungsi sebagai sistem pemeringkatan cerdas yang dapat memilih data paling berharga untuk tim penjualan. Metode ini adalah kemajuan besar dibandingkan dengan cara pembersihan manual yang sering kali melewatkannya detail penting.

Secara teknis, skrip ini menggunakan logika "Binary Flagging", di mana sistem secara otomatis memberi tanda (1 untuk ada isi, 0 untuk kosong) pada kolom penting seperti alamat email, nomor telepon, nomor faks, dan nama perusahaan[10]. Dengan cara ini, setiap baris data memiliki skor kelengkapan yang jelas. Skrip kemudian menghitung total skor tersebut dan menyimpannya dalam kolom baru yang disebut filled\_count. Dengan teknik ini, tim penjualan tidak perlu lagi menebak-nebak data mana yang siap dihubungi; sistem akan secara otomatis menampilkan data dengan informasi paling lengkap di bagian atas.

Keunggulan utama dari kode ini adalah penggunaan Sortir Multi-Level. Skrip dirancang untuk mengurutkan data secara hierarkis: mengutamakan ketersediaan

email terlebih dahulu, lalu nomor telepon, kemudian nomor faks, dan terakhir nama perusahaan. Pengurutan ini dilakukan dari yang paling lengkap ke yang paling sedikit informasinya, sehingga data dengan informasi terbaik akan muncul di atas laporan, sementara data yang kurang lengkap akan pindah ke bawah. Setelah proses pengurutan selesai, kolom tambahan akan dihapus agar tampilan akhir kembali bersih dan siap dipakai oleh pengguna non-teknis.

Langkah terakhir dari skrip ini adalah mengubah nilai kosong menjadi karakter strip ("") untuk meningkatkan kerapian tampilan, lalu menyimpannya kembali dalam format Excel yang telah dioptimalkan. Pengembangan kode ini bukan hanya untuk menghemat waktu, tetapi juga untuk memberikan Data Integrity yang dapat diandalkan. Dengan bantuan otomatisasi Python ini, ribuan baris data CRM yang sebelumnya kacau kini berubah menjadi aset informasi yang sangat tajam dan akurat, sehingga memudahkan divisi penjualan dalam menjaga hubungan baik dengan klien perusahaan secara terus-menerus.



```
[3]: import pandas as pd

# Baca file Excel
df = pd.read_excel("merged_sorted.xlsx")

# 1. Tambahan kolom Mobile_number (default kosong dulu)
df["Mobile_number"] = ""

# 2. Normalisasi Phone_number: kalau awalan "21-", ganti jadi "021-"
def fix_phone(num):
    if isinstance(num, str) and num.startswith("21-"):
        return "021-" + num[3:]
    return num

df["Phone_number"] = df["Phone_number"].apply(fix_phone)

# 3. Cek apakah Company_Name dan Legal Name nyambung
# Misal: kalau salah satunya ada tapi beda + kosth flag
def check_company_legal(row):
    if pd.isna(row["Company_Name"]) or pd.isna(row["Legal Name"]):
        return "Missing"
    if row["Company_Name"].lower() in row["Legal Name"].lower():
        return "Match"
    return "Mismatch"

df["Company_vs_Legal"] = df.apply(check_company_legal, axis=1)

# Simpan hasil ke file baru
df.to_excel("data_cleaned.xlsx", index=False)
print("Data sudah jadi")
Data sudah jadi
```

Gambar 3.12 Code Python untuk Normalisasi Detail dan Sinkronisasi Identitas Perusahaan

Tahap penting dari pengembangan skrip untuk membersihkan data CRM ini berfokus pada aspek teknis tertentu untuk memastikan data tetap konsisten sebelum digunakan oleh tim penjualan. Seperti yang ditunjukkan pada Gambar 3.12, skrip ini mengerjakan tugas penyempurnaan, termasuk menyelaraskan format komunikasi dan memastikan nama perusahaan yang digunakan secara komersial sesuai dengan nama resmi perusahaan. Pendekatan ini memastikan bahwa database

tidak hanya bebas dari data yang berulang tetapi juga memiliki standar penulisan yang profesional dan terukur.

Salah satu fitur utama dalam bagian kode ini adalah fungsi Normalisasi Nomor Telepon. Seringkali, data yang diterima dari berbagai sumber tidak memiliki format awalan yang konsisten, seperti penggunaan kode area yang tidak lengkap. Skrip ini secara otomatis mengidentifikasi nomor yang dimulai dengan format lama atau yang tidak standar, dan menggantinya dengan format yang benar (contohnya mengubah awalan "21-" menjadi "021-"). Proses otomatis ini sangat membantu tim operasional karena mereka tidak perlu lagi melakukan pengeditan manual satu per satu untuk membuat format kontak klien seragam, sehingga mengurangi risiko kesalahan saat melakukan panggilan bisnis.

Selain pengaturan format nomor, skrip ini juga menerapkan logika Sinkronisasi Identitas Perusahaan melalui fungsi *check\_company\_legal*. Fungsi ini membandingkan kolom "*Company Name*" (nama komersial) dengan "*Legal Name*" (nama resmi perusahaan). Sistem secara otomatis akan memberikan label atau tanda seperti "*Match*" jika kedua nama sesuai, "*Mismatch*" jika ada perbedaan yang jelas, atau "*Missing*" jika salah satu data tidak ada. Fitur ini sangat penting bagi PT Dayalima Abisatya untuk memastikan data akurat dalam surat menyurat resmi atau pembuatan dokumen kontrak, sehingga profesionalisme perusahaan tetap terjaga di mata klien.

Sebagai rangkuman dari seluruh proses pembersihan data, skrip ini akan menyimpan hasil akhir yang sudah bersih ke dalam file Excel baru yang siap digunakan. Dengan penerapan logika validasi yang lebih lanjut ini.

```
[4]: import pandas as pd

# Baca file Excel
df = pd.read_excel("merged_sorted.xlsx")

# 1. Tambahan kolom Mobile_number
df["Mobile_number"] = ""

# 2. Normalisasi Phone_number
def fix_phone(num):
    if pd.isna(num):
        return num, ""
    if isinstance(num, str):
        # Kalau diawali "21-", ganti jadi "021-"
        if num.startswith("21-"):
            return "021-" + num[3:], ""
        # Kalau bukan "21- > pindahkan ke Mobile_number
        else:
            return "", num
    return num, ""

df[["Phone_number", "Mobile_number"]] = df[["Phone_number"]].apply(
    lambda x: pd.Series(fix_phone(x))
)

# 3. Samaan Company_Name & Legal Name
def match_company_legal(row):
    cname = str(row["Company_Name"]) if not pd.isna(row["Company_Name"]) else ""
    lname = str(row["Legal Name"]) if not pd.isna(row["Legal Name"]) else ""

    if cname == "" or lname == "":
        return "Missing"
    if cname.lower() in lname.lower():
        return "Match"
    if lname.lower() in cname.lower():
        return "Match"
    return "Mismatch"

df["Company_vs_Legal"] = df.apply(match_company_legal, axis=1)

# 4. Kalau match > sinkronkan Company_Name dan Legal Name
def sync_names(row):
    cname = str(row["Company_Name"]) if not pd.isna(row["Company_Name"]) else ""
    lname = str(row["Legal Name"]) if not pd.isna(row["Legal Name"]) else ""

    if row["Company_vs_Legal"] == "Match":
        # pilih yang lebih panjang (biasanya Legal Name Lebih Lengkap)
        return lname if len(lname) > len(cname) else cname
    return lname

df["Legal Name"] = df.apply(sync_names, axis=1)

# Simpan hasil ke file baru
df.to_excel("data_cleaned.xlsx", index=False)
print("Data sudah dibersihkan")

```

✓ Data sudah dibersihkan

Gambar 3.13 Code Python untuk Cleansing

Tahap terakhir dari seluruh proses pembersihan data CRM ini dapat dilihat pada Gambar 3.13. Bagian kode ini merupakan fase perbaikan paling mendalam, di mana skrip tidak hanya membersihkan data, tetapi juga melakukan sinkronisasi cermat untuk memastikan bahwa identitas perusahaan klien tetap terjaga dengan standar profesional yang tinggi.

Salah satu elemen paling penting dalam fase ini adalah Logika Pembersihan Kontak Ganda. Skrip dibuat untuk secara otomatis mengelompokkan kolom nomor telepon. Dengan fungsi fix\_phone, sistem dapat mengenali nomor yang memiliki awalan tidak umum (seperti "21-") dan secara otomatis mengubahnya ke format area standar ("021-"). Yang menarik, jika sistem menemukan nomor telepon yang tidak sesuai dengan kriteria utama, nomor tersebut akan secara otomatis dipindahkan ke kolom cadangan (Mobile\_number). Pendekatan ini memastikan tidak ada informasi kontak yang hilang, sambil menjaga agar database utama tetap rapi dan teratur.

Selanjutnya, skrip menerapkan fungsi Validasi Nama Legal Komprehensif menggunakan metode `match_company_legal`. Fungsi ini berfungsi untuk membandingkan kesamaan antara "Company Name" (nama perusahaan) dengan "Legal Name" (nama resmi PT/badan hukum). Hasil perbandingannya akan diklasifikasikan menjadi tiga label: "Match" jika nama-nama tersebut sesuai, "Mismatch" jika ada perbedaan, atau "Missing" jika salah satu kolom tidak terisi. Langkah ini sangat penting untuk mencegah kesalahan administratif saat perusahaan perlu mengeluarkan dokumen kontrak atau komunikasi resmi.

Kehebatan dari skrip ini terlihat pada fungsi Sinkronisasi Nama Otomatis (`sync_names`). Jika sistem menemukan status "Match", skrip secara otomatis akan memilih nama yang paling lengkap dan panjang untuk disimpan dalam kolom "Legal Name". Tujuannya adalah untuk memastikan bahwa database perusahaan selalu menyimpan nama klien yang paling formal dan tepat.

Sebagai akhir, semua hasil pengolahan yang telah melalui tahap validasi berlapis ini disimpan dalam file Excel final yang berjudul `data_cleaned.xlsx`. Dengan selesainya tahap yang terlihat pada Gambar 3. 12 ini, transformasi data dari ribuan baris data mentah yang tidak teratur menjadi informasi yang siap digunakan telah sepenuhnya tercapai. Keberhasilan otomatisasi yang dilakukan dengan Python ini tidak hanya memberikan efisiensi waktu yang signifikan untuk divisi penjualan, tetapi juga menjaga profesionalisme PT Dayalima Abisatya dalam mengelola data klien strategisnya secara berkelanjutan.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

### 3.2.3 Olah Data PINTAR



Gambar 3.14 website PINTAR

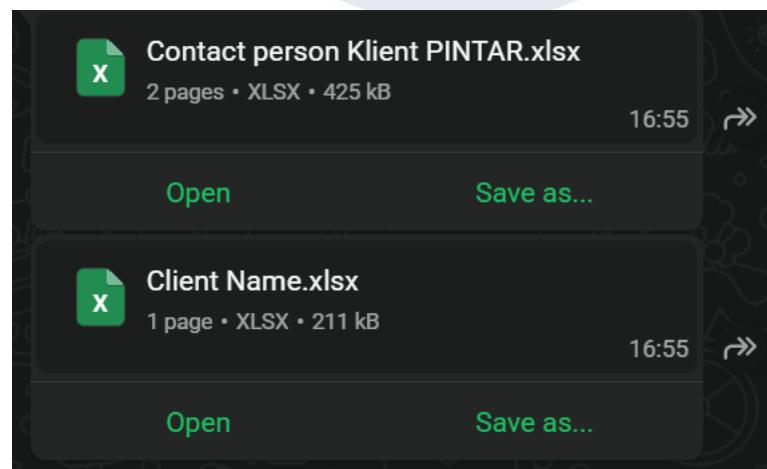
Dalam ekosistem operasional PT Dayalima Abisatya, keberadaan platform digital yang kuat adalah fondasi untuk efisiensi dalam layanan konsultasi SDM. Salah satu sistem utama yang dipakai adalah PINTAR, yang merupakan singkatan dari People Interactive Tools for Administration & Resourcing. Seperti yang ditunjukkan pada Gambar 3.14, platform ini berfungsi sebagai pusat manajemen administratif dan pengelolaan sumber daya yang dirancang khusus untuk menangani beban kerja yang tinggi sambil memastikan tingkat akurasi tetap terjaga.

PINTAR adalah solusi berbasis web yang memungkinkan perusahaan untuk menggabungkan berbagai proses bisnis dalam satu akses. Dengan antarmuka login yang aman, sistem ini menjaga agar semua data sensitif yang berhubungan dengan klien dan kandidat tetap terlindungi. Sebagai alat interaktif, PINTAR memudahkan kerja sama antara department, mulai dari manajemen basis data individu hingga penanganan tugas administratif yang rumit. Adanya logo Daya Dimensi Indonesia pada sistem ini menegaskan bahwa PINTAR adalah bagian penting dari standar layanan profesional yang disediakan oleh grup perusahaan kepada mitra bisnisnya[11].

Selama program magang, penggunaan sistem PINTAR berfokus pada pemanfaatan data yang tersimpan untuk mendukung transformasi digital. Sebagai

IT Intern, tugas yang dilakukan termasuk menganalisis struktur data administrasi dan resourcing yang sudah ada untuk diproses lebih lanjut. Data mentah yang diambil dari PINTAR sering kali menjadi objek dalam proses pengolahan data menggunakan Python, untuk menjamin bahwa informasi yang disampaikan kepada manajemen memiliki tingkat kevalidan yang tinggi. Penggabungan kemampuan sistem PINTAR dengan metode pengolahan data modern ini adalah kunci dalam mempercepat birokrasi internal.

Secara keseluruhan, PINTAR lebih dari sekadar aplikasi database, tetapi merupakan aset strategis yang memungkinkan PT Dayalima Abisatya untuk tetap fleksibel dan responsif terhadap kebutuhan pasar. Dengan sistem yang terstruktur, perusahaan dapat mengurangi kesalahan manual dalam pencatatan administratif dan lebih fokus pada pengembangan potensi manusia yang menjadi visi utama organisasi. Penggunaan PINTAR menunjukkan komitmen perusahaan dalam menerapkan teknologi yang tepat guna untuk menyediakan layanan konsultasi yang profesional, terukur, dan berkelanjutan di era digital.



Gambar 3.15 Raw Data PINTAR

Dalam langkah awal pengolahan data yang lebih jelas, perhatian difokuskan pada data yang berasal langsung dari sistem perusahaan. Seperti yang terlihat di Gambar 3.15, terdapat dua file utama dalam format Microsoft Excel yang dikirim oleh direksi untuk proses lebih lanjut. File-file tersebut adalah "Contact person Klient PINTAR. xlsx" dan "Client Name. xlsx". Kedua berkas ini merupakan data

mentah yang diperoleh langsung dari sistem PINTAR (People Interactive Tools for Administration & Resourcing), yang menyimpan informasi dasar mengenai jaringan klien dan kontak penting di PT Dayalima Abisatya.

Pada saat pertama kali diterima, kondisi data masih belum terorganisir dengan baik, yang berarti informasi yang ada belum siap untuk analisis mendalam. File "Contact person Klient PINTAR. xlsx" menyimpan daftar orang-orang yang menjadi kontak utama di setiap perusahaan klien, sementara file "Client Name. xlsx" berisi nama resmi dan komersial dari organisasi yang menggunakan layanan perusahaan. Tantangan utama dalam memproses data ini adalah kemungkinan adanya perbedaan antara nama klien yang tercatat di sistem dengan data kontak yang terpisah, serta kemungkinan format penulisan yang tidak seragam karena berasal dari input sistem yang besar.

Penyampaian data langsung dari direksi ini menunjukkan betapa pentingnya proses pengolahan data untuk pengambilan keputusan strategis perusahaan. Sebagai seorang IT Internship, langkah pertama yang diambil adalah memeriksa keutuhan data tersebut sebelum melanjutkan ke tahap pembersihan menggunakan Python. Proses ini melibatkan penghitungan jumlah baris, identifikasi kolom-kolom kunci, dan perencanaan awal mengenai cara menggabungkan kedua file ini menjadi satu database yang utuh.

Proses pengolahan data PINTAR ini sangat penting untuk meningkatkan efisiensi departemen penjualan dan bagian lainnya. Dengan mengubah data mentah yang terpisah menjadi informasi yang terintegrasi, PT Dayalima Abisatya dapat memiliki pemahaman yang lebih jelas tentang profil klien mereka. Ini membuktikan bahwa peran teknologi informasi tidak hanya terbatas pada pengembangan sistem, tetapi juga pada kemampuan untuk mengolah aset data internal menjadi informasi yang sangat berguna bagi kemajuan organisasi secara keseluruhan.



```
[8]: import pandas as pd  
file_a = pd.read_excel("Contact_person_Klient_PINTAR.xlsx")  
file_b = pd.read_excel("Copy of Client_name.xlsx")  
  
# Biar sama-sama bisa concat, tambahkan kolom kosong biar kolom match  
for col in file_a.columns:  
    if col not in file_b.columns:  
        file_b[col] = None  
  
for col in file_b.columns:  
    if col not in file_a.columns:  
        file_a[col] = None  
  
combined = pd.concat([file_a, file_b], ignore_index=True)  
combined.to_excel("merged_clean.xlsx", index=False)  
print("File berhasil digabung jadi 1 sheet")
```

File berhasil digabung jadi 1 sheet

Gambar 3.15 Olah Data PINTAR

Setelah melakukan pemeriksaan awal terhadap dokumen mentah yang diperoleh dari direksi, langkah teknik selanjutnya adalah menggabungkan data. Proses ini sangat penting karena informasi tentang klien dan kontak dalam sistem PINTAR terpisah dalam dua file yang berbeda. Seperti ditunjukkan pada Gambar 3.15 , sebuah skrip Python dibuat menggunakan library Pandas untuk menyatukan file "Contact person Klient PINTAR. xlsx" dan "Copy of Client\_name. xlsx" menjadi satu data yang utuh.

Secara teknis, skrip ini memulai dengan membaca kedua file tersebut ke dalam bentuk data yang disebut DataFrame. Namun, ada tantangan teknis karena kedua file tersebut memiliki struktur kolom yang berbeda. Untuk mengatasi masalah ini, skrip menggunakan logika untuk Menyamakan Kolom Secara Otomatis. Sebelum penggabungan dilakukan, sistem cerdas akan membandingkan daftar kolom dari kedua file. Jika kolom ada di file pertama tetapi tidak ada di file kedua (atau sebaliknya), skrip secara otomatis akan menambahkan kolom kosong dengan nilai None. Tindakan pencegahan ini memastikan bahwa proses penggabungan tidak mengalami kesalahan dan tidak ada informasi yang hilang atau berpindah posisi.

Setelah kolom diselaraskan, skrip menggunakan fungsi pd. concat untuk menggabungkan kedua sumber data secara vertikal menjadi satu lembar tunggal yang lengkap. Penggunaan parameter ignore\_index=True memastikan bahwa nomor baris dalam database baru ini teratur dari awal sampai akhir. Hasil dari penggabungan ini kemudian disimpan ke dalam file baru bernama "merged\_clean.

xlsx", yang menunjukkan bahwa data sekarang telah disatukan dalam satu akses tanpa ada data yang terpisah lagi.

Keberhasilan penggabungan data ini merupakan dasar yang penting bagi efisiensi kerja di PT Dayalima Abisatya. Dengan menyatukan data PINTAR yang sebelumnya terpisah, tim operasional kini memiliki satu basis data yang memudahkan pencarian dan analisis informasi klien. Proses digital melalui skrip otomatisasi ini terbukti jauh lebih efektif dibandingkan dengan cara copy-paste manual, karena membantu menjaga integritas data, mengurangi risiko kehilangan data selama pemindahan, dan mempercepat proses administrasi perusahaan secara signifikan.

### **3.3 Kendala yang Ditemukan**

Sewaktu meniti karier profesional semasa magang di PT Dayalima Abisatya, alur perancangan sistem otomatisasi tidak senantiasa lancar tanpa kendala. Ada berbagai kesulitan yang cukup pelik, baik dari aspek teknis maupun pengelolaan proyek, yang menuntut daya lenting dan ketekunan yang tinggi. Salah satu rintangan utama adalah keharusan untuk menguasai Google Apps Script sebagai bahasa coding utama dalam menyusun segenap ekosistem otomatisasi kantor. Mengingat Google Apps Script merupakan lingkungan kerja yang benar-benar baru dan belum pernah dipelajari sebelumnya, terdapat jurang pemahaman yang lumayan lebar di permulaan. Proses penyesuaian ini menghabiskan kurun waktu yang lumayan besar sebab sintaks dan petunjuk yang tersedia memiliki ciri khas tersendiri yang berbeda dari bahasa pemrograman lainnya.

Keadaan teknis tersebut bertambah rumit dengan situasi bahwa tidak ada pembimbing atau mentor teknis khusus yang dapat menyalurkan arahan langsung secara intensif. Hal ini terjadi sebab tiap staf IT di PT Dayalima Abisatya sudah memikul beban tugas pada rencana penting mereka masing-masing, sehingga waktu untuk mengawasi aspek teknis menjadi amat terbatas. Akibatnya, segenap proses belajar harus dilakukan secara mandiri melalui metode belajar mandiri dan berbagi dengan sesama peserta magang. Kondisi ini memunculkan kesulitan tersendiri dalam menanggulangi persoalan teknis yang muncul, sebab setiap penyelesaian

harus dicari lewat penelusuran petunjuk mandiri dan diskusi kelompok yang padat, yang sering kali memerlukan upaya tambahan untuk memastikan alur program yang dibangun sudah selaras dengan aturan operasional perusahaan.

Selain hambatan dari penguasaan teknologi, interaksi antar kelompok juga menjadi faktor yang amat menentukan kelancaran pembangunan sistem. Komunikasi antar seksi terkadang kurang berjalan teratur, dengan arahan atau keperluan dari beberapa unit pemakai (user) kadang kala kurang lengkap atau bahkan berganti di pertengahan jalan. Ketidakseragaman informasi ini berdampak langsung pada kerangka kode yang telah dirangkai, sehingga sering timbul tafsiran yang berbeda antara tim pengembang dan tim pelaksana. Imbas nyata dari minimnya detail keterangan keperluan ini adalah munculnya perbaikan berulang yang pada akhirnya mengacaukan jadwal penyelesaian proyek secara menyeluruh. Keterlambatan pada satu tahap penerapan akibat masukan ulang ini tentu memberikan efek berantai pada dimulainya rencana-rencana berikutnya, yang membuat pengelolaan proyek menjadi lebih berat.

Keterbatasan pengalaman dalam memakai teknologi baru seperti Google Apps Script juga secara wajar berujung pada timbulnya bermacam masalah teknis dalam wujud bug atau kekeliruan sistem. Di fase awal pengembangan, sistem acap kali mengalami gagal fungsi, mulai dari kekhilafan logika pemrograman, ketidakmampuan fungsi skrip, hingga hambatan dalam penyatuan antara Google Forms, Spreadsheets, dan pengiriman surel pemberitahuan otomatis yang tidak sesuai dengan dugaan awal. Proses penanggulangan bug (debugging) ini memerlukan cara coba-gagal yang sangat padat dan menyita urusan. Karena terbatasnya rujukan teknis yang dapat dijadikan acuan, proses pembersihan kode dan pemulihannya kerap memakan waktu yang jauh lebih panjang dari perkiraan semula. Meskipun demikian, rentetan kegagalan teknis ini sejatinya menjadi ajang pembelajaran yang sangat penting untuk menajamkan firasat pemecahan masalah dan pemahaman mendalam mengenai urutan penyatuan sistem digital.

Semua kendala tersebut kemudian bersua dengan tekanan dari batas waktu (deadline) penyelesaian proyek yang amat mendesak. Beberapa rencana otomatisasi ditargetkan untuk segera dipakai dalam kegiatan kantor dalam tempo pendek, sementara proses desain sistem yang mapan memerlukan wawasan mendalam terhadap alur kerja dalam perusahaan yang cukup rumit. Adanya tekanan waktu ini, ditambah dengan keharusan perbaikan dari pengguna dan proses debugging yang tak terduga, membentuk kesulitan besar dalam pengelolaan urutan kerja. Saat beberapa sistem harus dikerjakan secara bersamaan, pembagian konsentrasi menjadi poin utama meski beban yang dirasakan cukup tinggi. Desakan batas waktu ini pada akhirnya mendorong kelompok untuk bekerja dengan tingkat performa puncak, sekaligus menjadi ujian kematangan profesional dalam menjaga mutu hasil pekerjaan di tengah keterbatasan waktu dan sarana yang ada..

### **3.4 Solusi atas Kendala yang Ditemukan**

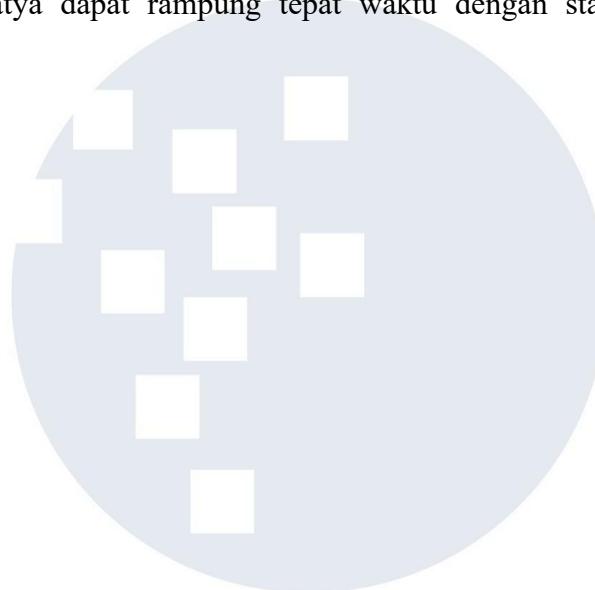
Demi menyajikan tanggapan yang menyeluruh atas tantangan selama program magang di PT Dayalima Abisatya, serangkaian jalan keluar strategis telah diterapkan untuk menjamin seluruh tatanan yang diciptakan tetap bermutu tinggi serta dapat dijalankan dengan mantap. Jalan keluar pertama yang dipilih untuk mengatasi keterbatasan pendamping teknis serta kurangnya pengalaman awal adalah dengan mendalami Google Apps Script secara mandiri dan berkesinambungan. Strategi ini mencakup pemanfaatan panduan resmi Google secara intensif, penjelajahan forum perkumpulan perancang, serta analisis contoh kasus dari pelbagai rancangan sejenis yang terdapat di dunia maya. Pembelajaran bukan hanya dilakukan secara teori, akan tetapi langsung dipraktikkan melalui telaah fitur-fitur dasar, penggunaan pemicuan otomatis, hingga pengelolaan kesinambungan antara Google Forms, Spreadsheets, dan Gmail. Corak belajar sambil berkarya (learning by doing) ini terbukti amat berguna dalam mengakselerasi proses penyesuaian diri sekaligus memberikan pemahaman mendalam tentang alur pemrograman yang khas dalam lingkungan Google Workspace.

Selain sisi teknis, penernalan alur komunikasi juga menjadi hal utama untuk menanggulangi informasi yang tidak sejajar antar bagian. Jalan keluar yang diwujudkan adalah dengan memprakarsai penyelarasan berulang dan sesi penjelasan tambahan bersama bagian pemakai, seperti tim Urusan Umum (GA), PMO, SDM, maupun dukungan TI internal lainnya. Rapat tambahan ini berperan untuk mengurai setiap keperluan sistem secara rinci hingga ke unit terkecil sebelum fase penulisan kode dimulai. Guna mengurangi bahaya salah tafsir, setiap pergantian keperluan maupun hasil diskusi langsung dicatat dalam notula tertulis yang dikonfirmasi ulang kepada pihak terkait. Langkah antisipatif ini secara nyata sanggup menekan kebutuhan perombakan besar di tengah rentang pengembangan, sehingga alur pengerjaan menjadi jauh lebih teratur dan efektif.

Untuk menuntaskan persoalan ketidak sempurnaan atau eror teknis yang muncul karena kurikulum teknologi baru, diterapkan cara perbaikan kerusakan bertahap dan pemeriksaan berulang pada tiap bagian sistem. Daripada menguji sistem seluruhnya di akhir, setiap fungsi kecil diuji terpisah (unit testing) untuk memastikan kestabilannya sebelum digabungkan ke dalam kumpulan final. Proses peniruan pengisian formulir dilakukan secara besar-besaran, diikuti dengan pemantauan balasan surel secara langsung dan pembuktian ketepatan data pada Google Sheets. Pendekatan ini diperkuat dengan pengujian bersama pemakai akhir (User Acceptance Testing) untuk menemukan potensi kegagalan sistem sedini mungkin sebelum dilepas secara resmi. Melalui ragam ini, setiap celah kekhawatiran atau kekeliruan logika dapat seketika diperbaiki, sehingga kemantapan sistem terpelihara pada tingkatan puncak.

Akhirnya, dalam menghadapi desakan batas waktu yang sempit, pengelolaan waktu yang lebih teratur menjadi penyelesaian penting yang diterapkan. Setiap rancangan besar dibagi menjadi segmen-segmen kerja yang lebih kecil (task breakdown) dengan penentuan prioritas berdasarkan tingkat kegentingan dan kerumitannya. Penentuan agenda harian dibuat secara tertib, contohnya dengan mengalokasikan waktu pagi hari khusus untuk telaah kebutuhan dan perbincangan, sedangkan waktu petang hari dipusatkan sepenuhnya untuk perancangan kode agar

konsentrasi tetap terjaga. Penyelarasan rutin dengan Pemimpin TI juga dijalankan secara periodik untuk menyamakan prioritas apabila ada perubahan keperluan mendadak atau penyesuaian jadwal dari pihak pengelola. Melalui perpaduan seluruh solusi ini, beban kerja tetap dapat ditangani dengan profesional, yang kemudian memastikan seluruh perangkat digital yang dirancang untuk PT Dayalima Abisatya dapat rampung tepat waktu dengan standar kinerja yang diharapkan.



**UMN**  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA