

LAPORAN MAGANG

DINAS PERHUBUNGAN KABUPATEN BENGKALIS

**RANCANG BANGUN SISTEM RO-RO ALERT BERBASIS
FLUTTER MOBILE, PC LOKAL, DAN CLOUD LARAVEL
UNTUK MONITORING OPERASIONAL KAPAL RO-RO
PADA DINAS PERHUBUNGAN KABUPATEN
BENGKALIS**

**RISKI AFENDI
6304221510**



**PROGRAM STUDI REKAYASA PERANGKAT LUNAK
JURUSAN TEKNIK INFORMATIKA POLITEKNIK NEGERI
BENGKALIS**

2026

LEMBAR PENGESAHAN

LAPORAN MAGANG DINAS PERHUBUNGAN KABUPATEN BENGKALIS

RANCANG BANGUN SISTEM RO-RO ALERT BERBASIS FLUTTER MOBILE, PC LOKAL, DAN CLOUD LARAVEL UNTUK MONITORING OPERASIONAL KAPAL RO-RO PADA DINAS PERHUBUNGAN KABUPATEN BENGKALIS

Ditulis sebagai salah satu syarat untuk menyelesaikan Magang

Oleh:

RISKI AFENDI
6304221510

Bengkalis, 29 Juni 2026

Kepala Bidang Pelayaran
Dinas Perhubungan Kabupaten Bengkalis



Reza Nurhawan, ST., M.T
NIP. 198206162015031001

Dosen Pembimbing Program Studi
Rekayasa Perangkat Lunak



Ryci Rahmatli Fiska, M.Kom
NIP. 199107112020122022

Mengetahui
Ketua Program Studi Rekayasa
Perangkat Lunak Politeknik Negeri Bengkalis



Fajri Profeso Putra, M. Cs.
NIP. 198805072015041003

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah SWT atas limpahan rahmat dan karunia-Nya sehingga kegiatan magang dan penyusunan Laporan Magang yang berjudul “Rancang Bangun Sistem Ro-Ro Alert Berbasis Flutter Mobile, PC Lokal, dan Cloud Laravel untuk Monitoring Operasional Kapal Ro-Ro pada Dinas Perhubungan Kabupaten Bengkalis” dapat diselesaikan.

Laporan ini disusun sebagai bentuk pertanggungjawaban atas pelaksanaan magang pada Dinas Perhubungan Kabupaten Bengkalis, khususnya Bidang Pelayaran, serta sebagai bagian dari kegiatan akademik Program Studi Sarjana Terapan Rekayasa Perangkat Lunak, Jurusan Teknik Informatika, Politeknik Negeri Bengkalis. Selama pelaksanaan magang, penulis memperoleh pengalaman dalam memahami kegiatan operasional pelabuhan, mendukung kegiatan TRIPS, menganalisis kebutuhan, serta merancang, mengembangkan, mengintegrasikan, menguji, dan mendokumentasikan Ro-Ro Alert.

Penyelesaian kegiatan magang dan laporan ini tidak terlepas dari arahan, bimbingan, dukungan, dan doa dari berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Bapak Ir. Johny Custer, S.T., M.T., selaku Direktur Politeknik Negeri Bengkalis;
2. Bapak Kasmawi, M.Kom., selaku Ketua Jurusan Teknik Informatika Politeknik Negeri Bengkalis;
3. Bapak Fajri Profesio Putra, S.Kom., M.Cs., selaku Ketua Program Studi Sarjana Terapan Rekayasa Perangkat Lunak Politeknik Negeri Bengkalis;
4. Ibu Ryci Rahmatil Fiska, M.T., selaku Koordinator Kerja Praktik Program Studi Diploma IV Rekayasa Perangkat Lunak Politeknik Negeri Bengkalis sekaligus dosen pembimbing yang telah memberikan waktu, arahan, dan masukan selama pelaksanaan magang dan penyusunan laporan;

5. Bapak dan Ibu dosen Jurusan Teknik Informatika Politeknik Negeri Bengkalis yang telah memberikan pengetahuan dan pengalaman akademik kepada penulis;
6. Bapak Edi Kurniawan, S.T., M.T., selaku Kepala Bidang Pelayaran, beserta seluruh pegawai Bidang Pelayaran Dinas Perhubungan Kabupaten Bengkalis yang telah memberikan kesempatan, penjelasan mengenai alur operasional yang dibutuhkan, bantuan selama kegiatan magang, serta kesempatan untuk mempresentasikan hasil Ro-Ro Alert;
7. Bapak Safrizal dan Ibu Agustina selaku orang tua penulis yang senantiasa memberikan doa, semangat, nasihat, dan dukungan selama pendidikan, pelaksanaan magang, dan penyusunan laporan; dan
8. Teman-teman Program Studi Rekayasa Perangkat Lunak serta seluruh pihak yang telah memberikan dukungan secara langsung maupun tidak langsung selama pelaksanaan magang dan penyusunan laporan.

Penulis menyadari bahwa laporan ini masih memiliki keterbatasan. Kritik dan saran yang membangun sangat diharapkan sebagai bahan perbaikan. Penulis berharap laporan ini dapat memberikan manfaat bagi pembaca, Program Studi Rekayasa Perangkat Lunak, dan Dinas Perhubungan Kabupaten Bengkalis, serta menjadi bahan pengembangan Ro-Ro Alert pada tahap berikutnya.

Bengkalis, 5 Agustus 2026
Penulis,



Riski Afendi
6304221510

DAFTAR ISI

LEMBAR PENGESAHAN	i
KATA PENGANTAR.....	ii
DAFTAR ISI	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	ix
DAFTAR LAMPIRAN	x
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Pemikiran Magang.....	1
1.2. Tujuan dan Manfaat Magang	4
1.2.1. Tujuan.....	4
1.2.2. Manfaat	5
1.3. Hasil Proyek Magang.....	6
BAB II GAMBARAN UMUM DINAS PERHUBUNGAN KABUPATEN BENGKALIS	9
2.1. Sejarah Singkat.....	9
2.2. Visi dan Misi	11
2.3. Struktur Organisasi.....	12
2.4. Ruang Lingkup dan Bidang Pelayaran.....	14
BAB III BIDANG PEKERJAAN SELAMA MAGANG	18
3.1. Orientasi dan Dukungan Operasional TRIPS	19
3.2. Analisis Kebutuhan dan Perancangan Ro-Ro Alert	21
3.3. Pengembangan <i>Cloud</i> Laravel	23

3.4. Pengembangan PC Lokal dan HMI.....	26
3.5. Pengembangan Aplikasi Flutter	29
3.6. Integrasi, Pengujian, Dokumentasi, dan Presentasi	32
BAB IV RANCANG BANGUN SISTEM RO-RO ALERT BERBASIS FLUTTER MOBILE, PC LOKAL, DAN CLOUD LARAVEL UNTUK MONITORING OPERASIONAL KAPAL RO-RO PADA DINAS PERHUBUNGAN KABUPATEN BENGKALIS.....	
4.1. Metodologi	37
4.1.1. Prosedur Pembuatan Sistem/Solusi.....	37
4.1.2. Metodologi Pengumpulan Data	39
4.1.3. Proses Perancangan	41
4.1.4. Tahapan dan Jadwal Pelaksanaan.....	43
4.2. Perancangan dan Implementasi.....	45
4.2.1. Analisis Data dan Kebutuhan.....	45
4.2.2. Rancangan Sistem/Solusi	50
1. Arsitektur Sistem.....	50
2. Alur Sistem.....	54
3. <i>Use Case</i>	57
4. <i>Activity Diagram</i>	60
5. Perancangan Basis Data	89
4.2.3 Implementasi Sistem/Solusi	96
1. Implementasi <i>Cloud</i> Laravel	97
2. Implementasi PC Lokal dan HMI	102
3. Implementasi Aplikasi Flutter	106
4. Implementasi Integrasi dan Sinkronisasi	113
5. Pengujian Sistem.....	115

4.2.4 Dampak Implementasi Sistem/Solusi	121
4.2.5 Kendala Implementasi.....	123
BAB V PENUTUP.....	127
5.1 Kesimpulan	127
5.2 Saran.....	130
DAFTAR PUSTAKA	133
LAMPIRAN.....	134

DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi Dinas Perhubungan Kabupaten Bengkalis	13
Gambar 2.2 Struktur Organisasi Bidang Pelayaran	14
Gambar 2.3 Lokasi Kantor Dinas Perhubungan Kabupaten Bengkalis	17
Gambar 3.1 Kegiatan Orientasi di Dinas Perhubungan Kabupaten Bengkalis	21
Gambar 3.2 Kegiatan Pengembangan <i>Cloud</i> Laravel	26
Gambar 3.3 Kegiatan Pengembangan PC Lokal dan HMI	29
Gambar 3.4 Kegiatan Pengembangan Aplikasi Flutter	32
Gambar 3.5 Demonstrasi dan Presentasi Hasil Proyek	34
Gambar 4.1 Arsitektur Sistem Ro-Ro Alert	51
Gambar 4.2 Alur Sistem <i>Monitoring</i> Operasional Kapal Ro-Ro	54
Gambar 4.3 <i>Use Case</i> Sistem Ro-Ro Alert	58
Gambar 4.4 <i>Activity Diagram</i> Login Admin	61
Gambar 4.5 <i>Activity Diagram</i> Login Petugas	63
Gambar 4.6 <i>Activity Diagram</i> Logout Admin	64
Gambar 4.7 <i>Activity Diagram</i> Logout Petugas	65
Gambar 4.8 <i>Activity Diagram</i> Kelola Profil Petugas Lapangan	67
Gambar 4.9 <i>Activity Diagram</i> Manajemen Akun Pengguna	69
Gambar 4.10 <i>Activity Diagram</i> Konfigurasi Target TRT	71
Gambar 4.11 <i>Activity Diagram</i> Monitoring Dashboard Real-Time	72
Gambar 4.12 <i>Activity Diagram</i> Remote Control Operasional	73
Gambar 4.13 <i>Activity Diagram</i> Laporan KPI Gross TRT dan Net TRT	75
Gambar 4.14 <i>Activity Diagram</i> Pencatatan Sandar (ATA)	77
Gambar 4.15 <i>Activity Diagram</i> Pelaporan Kendala (<i>Pause</i>)	79
Gambar 4.16 <i>Activity Diagram</i> Penyelesaian Kendala (<i>Resume</i>)	80
Gambar 4.17 <i>Activity Diagram</i> Pencatatan Berangkat (ATD)	82
Gambar 4.18 <i>Activity Diagram</i> Pengaturan Koneksi PC Lokal	83
Gambar 4.19 <i>Activity Diagram</i> Audio Peringatan dan Sirine	85
Gambar 4.20 <i>Activity Diagram</i> Sinkronisasi Operasional, Foto, dan Kendala	86

Gambar 4.21 <i>Activity Diagram</i> Rekonsiliasi <i>Fallback</i> 4G	88
Gambar 4.22 Skema Basis Data <i>Cloud</i> Laravel.....	90
Gambar 4.23 Skema Basis Data PC Lokal.....	92
Gambar 4.24 Skema Basis Data Flutter	94
Gambar 4.25 Halaman <i>Login Admin</i> Ro-Ro Alert <i>Web</i>	98
Gambar 4.26 <i>Dashboard Monitoring</i> Operasional Ro-Ro Alert <i>Web</i>	98
Gambar 4.27 Halaman Manajemen Petugas Ro-Ro Alert <i>Web</i>	99
Gambar 4.28 Halaman Manajemen Kapal Ro-Ro Alert <i>Web</i>	99
Gambar 4.29 Halaman Manajemen Dermaga Ro-Ro Alert <i>Web</i>	100
Gambar 4.30 Halaman Laporan KPI <i>Gross</i> TRT dan <i>Net</i> TRT	100
Gambar 4.31 Detail Operasional pada Laporan KPI <i>Gross</i> TRT dan <i>Net</i> TRT ..	101
Gambar 4.32 Halaman <i>Remote Control</i> dan <i>Log</i> Perintah Operasional	102
Gambar 4.33 Halaman Instalasi dan Registrasi PC Lokal	103
Gambar 4.34 <i>Dashboard</i> HMI PC Lokal	104
Gambar 4.35 Status Operasional Kapal dalam Kondisi Dijeda pada HMI PC Lokal	104
Gambar 4.36 Status Sinkronisasi, <i>Heartbeat</i> , dan Audio Peringatan PC Lokal .	105
Gambar 4.37 Halaman <i>Login</i> Aplikasi Ro-Ro Alert	106
Gambar 4.38 Halaman Beranda Aplikasi Ro-Ro Alert	107
Gambar 4.39 Halaman Pencatatan Sandar (ATA).....	108
Gambar 4.40 Halaman Pelaporan Kendala Operasional.....	109
Gambar 4.41 Halaman Penyelesaian Kendala (<i>Resume</i>)	110
Gambar 4.42 Halaman Pencatatan Berangkat (ATD)	111
Gambar 4.43 Halaman Riwayat Operasional.....	112
Gambar 4.44 Halaman Profil dan Pengaturan Koneksi	113
Gambar 4.45 Sinkronisasi Normal Flutter-PC Lokal- <i>Cloud</i>	114
Gambar 4.46 <i>Fallback</i> 4G dan Rekonsiliasi Data ke PC Lokal.....	115

DAFTAR TABEL

Tabel 1.1 Hasil Proyek Magang	7
Tabel 3.1 Rekapitulasi Bidang Pekerjaan Selama Magang.....	18
Tabel 3.2 Perangkat Lunak dan Data Pendukung	35
Tabel 3.3 Kendala dan Penyelesaian Selama Magang	35
Tabel 4.1 Tahapan dan Jadwal Pelaksanaan Proyek Ro-Ro Alert.....	44
Tabel 4.2 Kebutuhan Fungsional Sistem Ro-Ro Alert	47
Tabel 4.3 Kebutuhan Nonfungsional Sistem Ro-Ro Alert	48
Tabel 4. 4 Skenario Pengujian Fungsional	117
Tabel 4.5 Hasil Pengujian Integrasi Flutter, PC Lokal, dan Cloud	118
Tabel 4.6 Hasil Pengujian Audio, Sinkronisasi, dan Fallback	120
Tabel 4.7 Perbandingan Proses Sebelum dan Sesudah Implementasi.....	122
Tabel 4.8 Kendala Implementasi dan Tindak Lanjut	124

DAFTAR LAMPIRAN

Lampiran 1. Surat Keterangan Selesai Magang	134
Lampiran 2. Surat Penerimaan Magang	135
Lampiran 3. Logbook yang Telah Disahkan	136
Lampiran 4. Lembar Penilaian dari Instansi	152
Lampiran 5. Sertifikat Magang	153
Lampiran 6. Absensi Kehadiran	154
Lampiran 7. Dokumentasi Kegiatan Kerja Praktek	155

BAB I PENDAHULUAN

1.1. Latar Belakang Pemikiran Magang

Kegiatan operasional transportasi memerlukan pencatatan yang tertib agar informasi dapat digunakan untuk pemantauan, koordinasi, dan penyusunan laporan. Dalam lingkungan pemerintahan, pemanfaatan teknologi informasi tidak hanya berkaitan dengan perubahan media pencatatan, tetapi juga dengan keterpaduan proses dan ketersediaan data bagi pihak yang menjalankan pelayanan. Peraturan Presiden Nomor 95 Tahun 2018 menempatkan Sistem Pemerintahan Berbasis Elektronik sebagai landasan untuk membangun tata kelola yang efektif, transparan, akuntabel, dan didukung pelayanan publik yang berkualitas (Republik Indonesia 2018). Oleh karena itu, sistem informasi pada bidang transportasi perlu dirancang sesuai kondisi operasional, pola kerja petugas, serta ketersediaan jaringan di lokasi pelayanan.

Kebutuhan tersebut juga berkaitan dengan penyelenggaraan pelayaran dan angkutan penyeberangan. Undang-Undang Nomor 17 Tahun 2008 menempatkan angkutan di perairan, kepelabuhanan, keselamatan, dan keamanan pelayaran sebagai bagian dari sistem transportasi nasional yang harus dikembangkan secara efektif dan efisien dengan tetap mengikuti perkembangan ilmu pengetahuan dan teknologi (Republik Indonesia 2008). Pada tingkat daerah, Dinas Perhubungan Kabupaten Bengkalis melaksanakan urusan pemerintahan di bidang perhubungan melalui struktur dan fungsi yang diatur dalam Peraturan Bupati Bengkalis Nomor 99 Tahun 2021 sebagaimana diubah dengan Peraturan Bupati Bengkalis Nomor 44 Tahun 2024 (Pemerintah Kabupaten Bengkalis 2024). Dalam konteks laporan ini, Bidang Pelayaran menjadi lingkungan kerja yang berkaitan langsung dengan aktivitas pelabuhan, penyeberangan, dan pelayanan kapal Ro-Ro.

Penulis melaksanakan magang pada Dinas Perhubungan Kabupaten Bengkalis dan ditempatkan pada Bidang Pelayaran. Penempatan tersebut memberikan kesempatan untuk mengenali alur kerja instansi, mengamati aktivitas

pelayanan di lapangan, serta memahami bentuk koordinasi antara petugas pada kegiatan operasional pelabuhan. Bagi mahasiswa Program Studi Rekayasa Perangkat Lunak, pengalaman tersebut menjadi ruang untuk membandingkan konsep yang dipelajari di perguruan tinggi dengan kebutuhan pengguna di lingkungan kerja. Pengamatan tidak hanya diarahkan pada penggunaan aplikasi, tetapi juga pada hubungan antara prosedur pelayanan, perangkat yang digunakan, ketersediaan data, dan cara petugas mempertahankan kelangsungan pekerjaan ketika kondisi lapangan berubah.

Pada tahap awal magang, penulis mengikuti orientasi dan mendukung kegiatan posko yang berkaitan dengan penggunaan TRIPS. Portal resmi Pemerintah Kabupaten Bengkalis menjelaskan bahwa TRIPS digunakan sebagai layanan pemesanan tiket daring bagi kendaraan roda empat pada lintasan Air Putih-Sungai Selari dalam periode pengaturan arus mudik dan arus balik Maret 2026 (Pemerintah Kabupaten Bengkalis 2026). Berdasarkan catatan kegiatan, keterlibatan penulis mencakup pengaturan jalur kendaraan, pendampingan penggunaan kode QR dan pembayaran nontunai, pelayanan validasi tiket, serta distribusi *boarding pass*. Kegiatan tersebut memperlihatkan bahwa penggunaan sistem digital di pelabuhan tetap berhubungan erat dengan pengaturan fisik kendaraan dan pendampingan langsung kepada pengguna. TRIPS dalam laporan ini digunakan sebagai konteks pengalaman lapangan, bukan sebagai objek penilaian keberhasilan atau kegagalan sistem.

Catatan kegiatan juga menunjukkan beberapa kondisi yang memengaruhi pelayanan pada saat penulis bertugas. Pada sebagian kegiatan, pengguna memerlukan bantuan karena informasi kuota belum terlihat secara langsung pada antarmuka atau karena koneksi seluler di sekitar pelabuhan tidak stabil. Mekanisme digital dan mekanisme fisik atau manual digunakan sesuai kondisi operasional yang dicatat pada hari terkait. Temuan tersebut dibatasi pada pengamatan dan penugasan penulis, sehingga tidak digunakan untuk menyatakan kondisi seluruh pengguna maupun kinerja TRIPS secara keseluruhan. Meskipun demikian, pengalaman tersebut memperlihatkan pentingnya akses informasi yang jelas, jalur kerja

alternatif saat jaringan bermasalah, dan pencatatan yang tetap dapat ditelusuri ketika proses pelayanan berpindah dari satu mekanisme ke mekanisme lainnya.

Setelah kegiatan posko, catatan magang memuat proses rekapitulasi temuan operasional dan pembahasan kebutuhan sistem bersama Bidang Pelayaran. Kebutuhan yang dirumuskan meliputi pencatatan waktu kapal sandar atau *Actual Time of Arrival* (ATA), waktu kapal berangkat atau *Actual Time of Departure* (ATD), target waktu penyelesaian operasional atau *Turnaround Time* (TRT), serta kendala yang dapat menyebabkan proses dijeda. Dokumentasi proyek juga menunjukkan kebutuhan peringatan waktu, bukti foto dan lokasi, pemantauan melalui dasbor, dan sinkronisasi data antara perangkat petugas, komputer lokal, dan penyimpanan terpusat. Unsur-unsur tersebut dipandang sebagai kebutuhan rancangan karena status penerapan penuh dan dampak operasionalnya masih memerlukan pengujian serta konfirmasi dari instansi.

Kondisi jaringan menjadi salah satu pertimbangan dalam penyusunan solusi. Jalur utama dirancang menggunakan jaringan lokal atau LAN agar aplikasi petugas dapat berkomunikasi dengan PC Lokal di pelabuhan. Apabila PC Lokal tidak dapat dijangkau tetapi perangkat masih memiliki akses internet, aplikasi menyediakan jalur cadangan (*fallback*) melalui koneksi 4G menuju *server Cloud*. Penyimpanan lokal, penggunaan identitas transaksi yang konsisten, dan proses sinkronisasi diperlukan untuk mengurangi risiko pencatatan ganda serta menjaga hubungan antara data lapangan dan data terpusat. Pendekatan tersebut tidak dimaksudkan untuk menjamin sistem selalu tersedia, melainkan sebagai rancangan kesinambungan proses yang perlu dibuktikan melalui pengujian pada kondisi yang sesuai.

Berdasarkan kebutuhan tersebut, proyek magang diarahkan pada Rancang Bangun Sistem Ro-Ro Alert Berbasis *Flutter Mobile*, PC Lokal, dan *Cloud Laravel* untuk *Monitoring* Operasional Kapal Ro-Ro pada Dinas Perhubungan Kabupaten Bengkalis. Ro-Ro Alert dirancang sebagai sistem dengan tiga komponen utama. Aplikasi *Flutter* mendukung petugas lapangan dalam mencatat ATA, ATD, dan kendala beserta bukti foto dan lokasi. PC Lokal dan *Human-Machine Interface* (HMI) berperan sebagai pusat pencatatan serta pemantauan di lokasi, termasuk

pengelolaan waktu dan peringatan audio. Sementara itu, *Cloud* Laravel menyediakan penyimpanan data terpusat, dasbor administrasi, laporan, dan integrasi antarkomponen. Penjelasan teknis mengenai arsitektur, basis data, antarmuka, dan pengujian ditempatkan pada BAB IV.

Pengembangan Ro-Ro Alert memberi ruang bagi penulis untuk menerapkan kompetensi Rekayasa Perangkat Lunak melalui analisis kebutuhan, perancangan alur dan data, pengembangan aplikasi, integrasi antarkomponen, pengujian internal, serta penyusunan dokumentasi teknis. Setiap kegiatan tersebut berhubungan dengan masalah nyata yang ditemui dalam lingkungan kerja, tetapi hasil proyek tetap disajikan sesuai bukti yang tersedia tanpa menyatakan dampak yang belum diukur. Rangkaian kegiatan magang dan pengembangan sistem tersebut menjadi dasar penyusunan Laporan Kerja Praktik ini. Tujuan pelaksanaan magang serta hasil proyek dijelaskan pada subbab berikutnya.

1.2. Tujuan dan Manfaat Magang

1.2.1. Tujuan

Pelaksanaan magang diarahkan untuk menghubungkan pengalaman kerja di Bidang Pelayaran dengan penerapan kompetensi Rekayasa Perangkat Lunak. Berdasarkan kegiatan yang terdokumentasi, tujuan magang ini adalah sebagai berikut.

1. Memperoleh pengalaman kerja secara langsung di lingkungan Dinas Perhubungan Kabupaten Bengkalis, khususnya pada Bidang Pelayaran;
2. Memahami alur kerja, koordinasi petugas, dan kebutuhan informasi yang berkaitan dengan kegiatan operasional pelabuhan dan pelayanan kapal Ro-Ro;
3. Mengidentifikasi dan menganalisis kebutuhan pencatatan serta pemantauan ATA, ATD, TRT, dan kendala operasional berdasarkan kegiatan yang terdokumentasi;

4. Merancang dan mengembangkan komponen *Cloud* Laravel, PC Lokal/HMI, dan aplikasi Flutter dalam proyek Ro-Ro Alert;
5. Mengintegrasikan pertukaran data antara aplikasi petugas, PC Lokal, dan *Cloud* melalui jalur LAN serta jalur cadangan 4G;
6. Melaksanakan pengujian internal terhadap fungsi utama dan alur integrasi sesuai ruang lingkup proyek; dan
7. Menyusun dokumentasi teknis serta laporan magang sebagai bahan evaluasi dan pengembangan lebih lanjut.

1.2.2. Manfaat

Kegiatan magang dan pengembangan Ro-Ro Alert diharapkan memberikan manfaat yang dapat ditinjau dari kepentingan mahasiswa, institusi pendidikan, dan instansi tempat magang.

Bagi mahasiswa atau penulis. Magang memberikan pengalaman untuk memahami kondisi kerja pada instansi pemerintah, melatih analisis kebutuhan, mengembangkan dan mengintegrasikan perangkat lunak, serta memperkuat kemampuan dokumentasi dan komunikasi teknis. Keterlibatan pada kegiatan lapangan juga membantu penulis melihat hubungan antara rancangan sistem dan prosedur operasional yang didukungnya.

Bagi Politeknik Negeri Bengkalis dan Program Studi Rekayasa Perangkat Lunak. Hasil kegiatan dapat menjadi bahan untuk menilai penerapan kompetensi akademik mahasiswa pada lingkungan kerja. Dokumentasi proyek dan pengalaman pelaksanaan magang juga dapat digunakan sebagai referensi bagi evaluasi pembelajaran, penguatan hubungan dengan instansi, dan pelaksanaan proyek sejenis pada masa mendatang.

Bagi Dinas Perhubungan Kabupaten Bengkalis. Proyek menyediakan rancangan dan prototipe Ro-Ro Alert beserta dokumentasi teknis sebagai alternatif mekanisme pencatatan dan pemantauan operasional kapal Ro-Ro. Hasil tersebut dapat digunakan sebagai bahan evaluasi untuk menentukan kelayakan, kebutuhan

penyempurnaan, dan tahapan pengembangan selanjutnya tanpa menganggap sistem telah diterapkan secara penuh.

1.3. Hasil Proyek Magang

Hasil proyek magang mencakup produk kerja yang dibentuk selama rangkaian analisis, perancangan, pengembangan, integrasi, pengujian internal, dan dokumentasi. Pada proyek ini, hasil yang disusun tidak hanya berupa aplikasi, tetapi juga artefak teknis yang menjelaskan cara setiap komponen Ro-Ro Alert dirancang dan saling berkomunikasi.

Berdasarkan dokumentasi proyek, hasil pengembangan perangkat lunak mencakup *Cloud* Laravel, PC Lokal/HMI, dan aplikasi Flutter. Ketiga komponen tersebut didukung oleh dokumentasi GitBook, kontrak API, diagram arsitektur dan alur sistem, *use case*, *activity diagram*, rancangan basis data, serta skenario pengujian. Keberadaan hasil tersebut diverifikasi dari *folder* dokumentasi, berkas diagram, dan catatan kegiatan pengembangan.

Status setiap hasil disajikan berdasarkan kondisi terbaru. Seluruh komponen Ro-Ro Alert telah dikembangkan secara mandiri oleh penulis dan seluruh UML telah diverifikasi serta disetujui oleh Kepala Bidang Pelayaran. Tahap berikutnya adalah *hosting web admin*, *setup PC Control* pada dua pelabuhan, dan penerapan sistem di lapangan. Integrasi *LED timer display* masih ditunda karena modul penghubung antara *website* dan perangkat LED belum tersedia. Rincian hasil proyek ditunjukkan pada Tabel 1.1.

Tabel 1.1 Hasil Proyek Magang

No	Hasil	Deskripsi	Bentuk Hasil	Status/Keterangan
1	Cloud Laravel Ro-Ro Alert	Komponen terpusat untuk autentikasi, <i>master</i> data, penyimpanan operasional, dasbor administrasi, laporan, dan integrasi data.	Prototipe aplikasi <i>web</i> dan layanan API	Pengembangan telah selesai pada lingkungan lokal; tahap berikutnya adalah <i>hosting web admin</i> dan penerapan operasional.
2	PC Lokal dan HMI	Komponen lokal untuk menerima pencatatan operasional, menampilkan status dermaga, mengelola waktu TRT, peringatan audio, dan sinkronisasi ke <i>Cloud</i> .	Prototipe aplikasi lokal berbasis PHP, JavaScript, dan MySQL/MariaDB	Pengembangan telah selesai; <i>setup</i> dan konfigurasi PC <i>Control</i> masih perlu dilakukan pada dua pelabuhan.
3	Aplikasi Flutter Ro-Ro Alert	Aplikasi petugas untuk pencatatan ATA, ATD, dan kendala dengan dukungan foto, lokasi, jaringan LAN, serta jalur cadangan 4G.	Prototipe aplikasi Android dan APK demo	Aplikasi telah dikembangkan; penerapan bersama komponen <i>Cloud</i> dan PC <i>Control</i> di lapangan masih perlu dilakukan.
4	Dokumentasi GitBook dan kontrak API	Dokumentasi arsitektur, konfigurasi, alur integrasi, API, sinkronisasi, dan batas tanggung jawab setiap komponen.	Dokumentasi teknis digital	Terdokumentasi; pembaruan setelah periode magang perlu dipisahkan dari versi yang digunakan selama magang.
5	Diagram dan rancangan basis data	Diagram arsitektur, alur sistem, <i>use case</i> , <i>activity diagram</i> , serta rancangan basis data <i>Cloud</i> , PC Lokal, dan Flutter.	Berkas Draw.io dan gambar PNG	Seluruh UML telah diverifikasi dan disetujui oleh Kepala Bidang Pelayaran.

No	Hasil	Deskripsi	Bentuk Hasil	Status/Keterangan
6	Skenario dan dokumentasi pengujian	Dokumen pengujian fungsi, sinkronisasi, audio, jaringan, dan integrasi yang digunakan selama pengembangan internal.	Dokumen skenario dan catatan uji internal	Pengujian internal terdokumentasi; validasi operasional lapangan dilakukan setelah <i>hosting</i> , <i>setup</i> PC <i>Control</i> pada dua pelabuhan, dan penerapan sistem.

Rincian proses perancangan, implementasi, integrasi, dan pengujian setiap hasil Ro-Ro Alert akan dibahas pada BAB IV. Pembahasan tersebut akan membedakan hasil pengembangan internal, hasil pengujian yang telah memiliki bukti, dan data yang masih memerlukan validasi dari instansi.

BAB II

GAMBARAN UMUM DINAS PERHUBUNGAN KABUPATEN BENGKALIS

2.1. Sejarah Singkat

Sejarah singkat Dinas Perhubungan Kabupaten Bengkalis dapat ditelusuri melalui perkembangan dasar hukum yang mengatur kedudukan dan susunan organisasinya. Sumber resmi yang tersedia tidak memuat tanggal pendirian dinas secara khusus. Oleh karena itu, pembahasan pada subbab ini tidak menetapkan tahun berdiri yang tidak dapat diverifikasi, tetapi berfokus pada perubahan kelembagaan yang dapat dibuktikan melalui peraturan daerah. Pendekatan tersebut menempatkan sejarah organisasi sebagai rangkaian penataan tugas, fungsi, susunan unit kerja, dan tata kerja dalam lingkungan Pemerintah Kabupaten Bengkalis.

Pada tahun 2021, Pemerintah Kabupaten Bengkalis menetapkan Peraturan Bupati Bengkalis Nomor 99 Tahun 2021 tentang Kedudukan, Susunan Organisasi, Tugas dan Fungsi serta Tata Kerja Dinas Perhubungan Kabupaten Bengkalis. Peraturan ini menjadi dasar penataan Dinas Perhubungan sebagai perangkat daerah yang menangani urusan pemerintahan di bidang perhubungan. Pengaturan tersebut penting karena pelayanan perhubungan di Kabupaten Bengkalis tidak hanya berkaitan dengan lalu lintas dan angkutan jalan, tetapi juga kepelabuhanan, pelayaran, serta angkutan sungai, danau, dan penyeberangan yang mendukung hubungan antarkawasan.

Penataan organisasi kemudian disesuaikan melalui Peraturan Bupati Bengkalis Nomor 44 Tahun 2024. Peraturan tersebut mengubah beberapa ketentuan dalam Peraturan Bupati Nomor 99 Tahun 2021 untuk menyesuaikan sistem kerja instansi pemerintah dengan kebijakan penyederhanaan birokrasi. Perubahan mencakup rumusan susunan organisasi, kedudukan jabatan fungsional, pembentukan Unit Pelaksana Teknis Daerah, dan hubungan kerja di lingkungan dinas. Peraturan ini ditetapkan pada 17 September 2024, diundangkan pada 18

September 2024, dan berstatus berlaku pada periode magang penulis (Pemerintah Kabupaten Bengkalis 2024).

Struktur hasil perubahan tersebut memperlihatkan pembagian kerja yang lebih terarah. Kepala Dinas didukung oleh Sekretariat, Bidang Lalu Lintas Jalan, Bidang Angkutan Jalan dan Pemadu Moda, Bidang Kepelabuhanan, Bidang Pelayaran, Kelompok Jabatan Fungsional, dan UPTD. Dalam susunan ini, Bidang Pelayaran tetap menjadi unit yang secara formal menempatkan keselamatan pelayaran serta angkutan sungai, danau, dan penyeberangan sebagai dua lingkup kerja internal. Keberadaan kedua seksi tersebut menunjukkan pentingnya pelayanan transportasi perairan bagi wilayah Kabupaten Bengkalis yang memiliki kebutuhan konektivitas antarpulau dan kawasan pesisir. Peraturan tahun 2024 juga menegaskan pola hubungan kerja yang bersifat konsultatif dan koordinatif, sedangkan tanggung jawab pelaksanaan tugas tetap mengikuti jenjang organisasi. UPTD dapat dibentuk berdasarkan kebutuhan dan beban kerja untuk menjalankan kegiatan teknis operasional atau teknis penunjang. Pengaturan ini memperlihatkan bahwa perkembangan kelembagaan tidak hanya berupa perubahan nama unit, tetapi juga penyesuaian cara kerja dan jalur pertanggungjawaban.

Kantor Dinas Perhubungan berada di Jalan Pramuka, Senggoro, Kecamatan Bengkalis, Kabupaten Bengkalis, Riau. Penulisan alamat ini menggunakan unsur yang konsisten pada portal resmi pemerintah dan PPID. Kata "Pekanbaru" yang muncul pada salah satu halaman daftar perangkat daerah tidak digunakan karena tidak sesuai dengan lokasi administratif kantor dan tidak didukung oleh halaman PPID Kabupaten Bengkalis (PPID Kabupaten Bengkalis n.d.). Pada periode 10 Maret sampai 28 Juni 2026, penulis melaksanakan magang di Dinas Perhubungan Kabupaten Bengkalis dengan penempatan pada Bidang Pelayaran. Dengan demikian, dasar hukum dan susunan organisasi yang digunakan dalam laporan ini adalah ketentuan yang berlaku selama periode tersebut.

2.2. Visi dan Misi

Dokumen Indikator Kinerja Utama Dinas Perhubungan Kabupaten Bengkalis menjelaskan bahwa dinas mendukung visi dan misi Pemerintah Kabupaten Bengkalis yang ditetapkan dalam Rencana Pembangunan Jangka Menengah Daerah (RPJMD) Kabupaten Bengkalis Tahun 2021-2026. Dengan demikian, rumusan berikut merupakan visi dan misi pemerintah daerah yang menjadi arah kerja perangkat daerah, bukan visi dan misi terpisah yang disusun khusus oleh Dinas Perhubungan (Dinas Perhubungan Kabupaten Bengkalis 2021; Pemerintah Kabupaten Bengkalis 2021).

Visi: Terwujudnya Kabupaten Bengkalis yang Bermarwah, Maju dan Sejahtera.

Misi:

1. Mewujudkan pengelolaan potensi keuangan daerah, sumber daya alam dan sumber daya manusia yang efektif dalam memajukan perekonomian.
2. Mewujudkan reformasi birokrasi serta penguatan nilai-nilai agama dan budaya Melayu menuju tata kelola pemerintahan yang baik dan masyarakat yang berkarakter.
3. Mewujudkan penyediaan infrastruktur yang berkualitas dan mengembangkan potensi wilayah perbatasan untuk kesejahteraan rakyat.

Bagi Dinas Perhubungan, misi pertama berkaitan dengan dukungan transportasi terhadap kegiatan ekonomi dan pemanfaatan potensi pesisir. Misi kedua menjadi landasan untuk menjalankan pelayanan dan administrasi secara tertib, akuntabel, serta berorientasi kepada masyarakat. Sementara itu, misi ketiga berhubungan langsung dengan penyediaan infrastruktur perhubungan dan penguatan konektivitas wilayah, termasuk kawasan yang bergantung pada angkutan penyeberangan.

Penempatan penulis pada Bidang Pelayaran berada dalam konteks pencapaian arah pembangunan tersebut. Kegiatan pengamatan layanan penyeberangan, orientasi lingkungan kerja, dan identifikasi kebutuhan informasi operasional memberikan gambaran tentang peran pelayanan perhubungan dalam menghubungkan wilayah. Hubungan ini digunakan sebagai konteks magang, tanpa

menyatakan bahwa proyek yang dikerjakan telah menghasilkan perubahan kinerja instansi yang belum diukur.

2.3. Struktur Organisasi

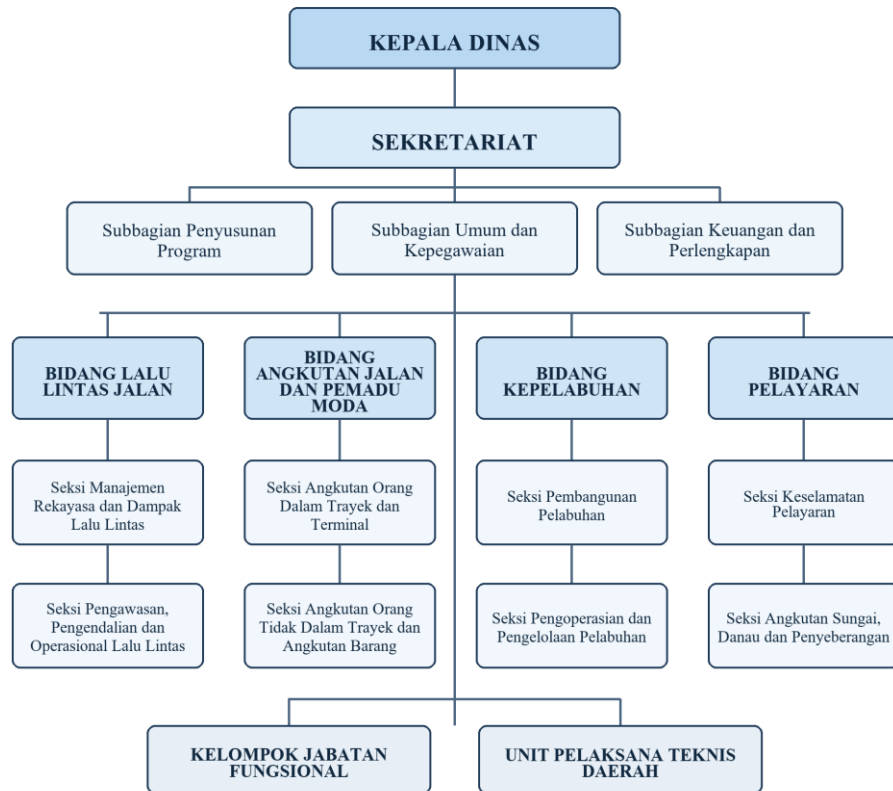
Susunan organisasi Dinas Perhubungan Kabupaten Bengkalis mengacu pada Peraturan Bupati Bengkalis Nomor 44 Tahun 2024. Dinas dipimpin oleh Kepala Dinas yang bertanggung jawab atas pelaksanaan tugas perangkat daerah di bidang perhubungan. Dalam menjalankan organisasi, Kepala Dinas didukung oleh unsur sekretariat, bidang teknis, kelompok jabatan fungsional, dan UPTD sesuai kebutuhan serta beban kerja (Pemerintah Kabupaten Bengkalis 2024).

Sekretariat merupakan unsur pendukung administrasi dinas dan terdiri atas Subbagian Penyusunan Program, Subbagian Umum dan Kepegawaian, serta Subbagian Keuangan dan Perlengkapan. Pembagian ini menempatkan kegiatan perencanaan, administrasi umum, kepegawaian, keuangan, dan pengelolaan perlengkapan dalam satu jalur koordinasi di bawah Sekretaris. Sekretariat mendukung bidang teknis agar pelaksanaan program dan pelaporan dapat dilakukan secara terkoordinasi.

Unsur teknis Dinas Perhubungan terdiri atas empat bidang. Bidang Lalu Lintas Jalan membawahi Seksi Manajemen Rekayasa dan Dampak Lalu Lintas serta Seksi Pengawasan, Pengendalian dan Operasional Lalu Lintas. Bidang Angkutan Jalan dan Pemas Angkutan Orang membawahi Seksi Angkutan Orang Dalam Trayek dan Terminal serta Seksi Angkutan Orang Tidak Dalam Trayek dan Angkutan Barang. Bidang Kepelabuhanan terdiri atas Seksi Pembangunan Pelabuhan dan Seksi Pengoperasian dan Pengelolaan Pelabuhan.

Bidang Pelayaran ditempatkan sejajar dengan ketiga bidang teknis lainnya dan dipimpin oleh Kepala Bidang. Struktur internalnya terdiri atas Seksi Keselamatan Pelayaran serta Seksi Angkutan Sungai, Danau dan Penyeberangan. Susunan ini membedakan Bidang Pelayaran dari Bidang Kepelabuhanan, sekaligus menunjukkan bahwa aspek keselamatan dan pelayanan angkutan perairan menjadi

bagian formal dari organisasi. Gambaran keseluruhan susunan dinas ditunjukkan pada Gambar 2.1.



Gambar 2.1 Struktur Organisasi Dinas Perhubungan Kabupaten Bengkulu
 Sumber: Disusun kembali berdasarkan Peraturan Bupati Bengkulu Nomor 44 Tahun 2024, Lampiran 1.

Selain sekretariat dan bidang, struktur organisasi juga memuat Kelompok Jabatan Fungsional dan UPTD. Jabatan fungsional memberikan pelayanan berdasarkan keahlian dan keterampilan tertentu serta berkedudukan sesuai keterkaitan tugasnya. UPTD dapat dibentuk berdasarkan kebutuhan dan beban kerja untuk melaksanakan kegiatan teknis operasional atau teknis penunjang pada satu atau beberapa kecamatan. Dalam pelaksanaannya, Kepala UPTD bertanggung jawab kepada Kepala Dinas dan melakukan koordinasi operasional dengan camat.

Posisi penulis selama magang berada pada Bidang Pelayaran. Penempatan ini menghubungkan kegiatan magang dengan dua seksi yang berada di bawah bidang tersebut, terutama konteks keselamatan pelayanan dan angkutan

penyeberangan. Struktur ringkas Bidang Pelayaran disajikan pada Gambar 2.2 agar hubungan antara Kepala Bidang dan kedua seksi dapat dilihat dengan jelas.



Gambar 2.2 Struktur Organisasi Bidang Pelayaran
Sumber: Disusun kembali berdasarkan Peraturan Bupati Bengkalis Nomor 44 Tahun 2024, Pasal 3 dan Lampiran I.

Pembimbing lapangan penulis adalah Edi Kurniawan, S.T., M.T., selaku Kepala Bidang Pelayaran. Nama, gelar, dan jabatan tersebut tercantum dalam Perjanjian Kinerja Dinas Perhubungan Kabupaten Bengkalis Tahun 2024 dan kembali disebut dalam publikasi resmi pemerintah pada Maret 2025. Catatan kegiatan magang tahun 2026 juga merekam koordinasi penulis dengan Kepala Bidang Pelayaran selama penempatan. Data ini digunakan tanpa mencantumkan NIP karena tidak diperlukan dalam narasi laporan (Dinas Perhubungan Kabupaten Bengkalis 2024; Dinas Komunikasi, Informatika dan Statistik Kabupaten Bengkalis 2025).

2.4. Ruang Lingkup dan Bidang Pelayaran

Dinas Perhubungan Kabupaten Bengkalis merupakan perangkat daerah yang menjalankan urusan pemerintahan bidang perhubungan sesuai kewenangan kabupaten. Ruang lingkup tersebut meliputi perumusan dan pelaksanaan kebijakan daerah, penyediaan dukungan pelayanan, koordinasi, pengembangan sumber daya manusia, pemantauan, pengawasan, evaluasi, dan pelaporan di bidang perhubungan. Dokumen perencanaan dinas menempatkan lalu lintas jalan, angkutan jalan, kepelabuhanan, dan pelayaran sebagai kelompok layanan utama

yang harus dikelola secara terarah (Dinas Perhubungan Kabupaten Bengkalis 2021).

Pembagian empat bidang teknis pada struktur tahun 2024 memberikan batas organisasi bagi pelaksanaan layanan. Bidang Lalu Lintas Jalan menangani lingkup manajemen, rekayasa, dampak, pengawasan, pengendalian, dan operasional lalu lintas jalan. Bidang Angkutan Jalan dan Pemadu Moda berhubungan dengan angkutan orang, terminal, serta angkutan barang. Bidang Kepelabuhanan ditempatkan pada lingkup pembangunan, pengoperasian, dan pengelolaan pelabuhan. Sementara itu, Bidang Pelayaran mencakup keselamatan pelayaran dan angkutan sungai, danau, serta penyeberangan. Uraian ini mengikuti nama unit pada peraturan dan tidak menambahkan kewenangan yang tidak tercantum dalam sumber resmi (Pemerintah Kabupaten Bengkalis 2024).

Dalam konteks Bidang Pelayaran, Perjanjian Kinerja Tahun 2024 memperlihatkan sasaran peningkatan kualitas layanan pelabuhan. Dokumen tersebut juga menempatkan kegiatan pengawasan pengoperasian pelabuhan sungai, danau, dan penyeberangan pada Seksi Keselamatan Pelayaran, sedangkan Seksi Angkutan Sungai, Danau dan Penyeberangan berkaitan dengan pemenuhan persyaratan perizinan pembangunan dan pengoperasian pelabuhan sungai dan danau. Informasi ini menunjukkan bahwa lingkup bidang tidak hanya bersifat administratif, tetapi juga mencakup pengawasan dan fasilitasi dokumen sesuai pembagian kerja yang ditetapkan (Dinas Perhubungan Kabupaten Bengkalis 2024). Indikator pada dokumen tersebut memakai jumlah laporan pengawasan dan jumlah dokumen pemenuhan persyaratan sebagai ukuran kerja. Laporan ini tidak menyalin target anggaran atau jumlah target tahun 2024 sebagai kondisi tahun 2026, tetapi menggunakan jenis kegiatannya untuk menjelaskan lingkup yang telah terverifikasi.

Pelayanan kapal Ro-Ro berada dalam konteks angkutan penyeberangan yang menghubungkan Pulau Bengkalis dengan daratan Sumatra melalui lintasan Air Putih–Sungai Selari. Publikasi resmi Pemerintah Kabupaten Bengkalis pada Maret 2025 memuat peran Kepala Bidang Pelayaran dalam penyampaian informasi pemesanan tiket dan pengaturan layanan arus balik. Sumber tersebut memperkuat

hubungan Bidang Pelayaran dengan pelayanan penyeberangan Ro-Ro, tetapi tidak digunakan untuk menyimpulkan bahwa seluruh kegiatan operasional pelabuhan berada pada satu bidang, karena struktur resmi juga memisahkan Bidang Kepelabuhanan dan UPTD (Dinas Komunikasi, Informatika dan Statistik Kabupaten Bengkalis 2025). Dalam praktik pelayanan, penyampaian informasi jadwal, pemesanan tiket, dan ketentuan perjalanan perlu dikoordinasikan dengan petugas yang berhadapan langsung dengan pengguna. Hubungan koordinatif inilah yang menjadi konteks pengamatan penulis selama berada di lingkungan pelayanan penyeberangan.

Kegiatan magang penulis dimulai dengan penyampaian surat dan koordinasi penempatan di Dinas Perhubungan pada 10 Maret 2026. Berdasarkan *logbook*, penulis kemudian ditempatkan pada pos pelabuhan Ro-Ro untuk mengikuti orientasi awal dan mendukung pengamatan layanan TRIPS. Penempatan lapangan tersebut memberi kesempatan kepada penulis untuk memahami alur kedatangan pengguna, pemeriksaan tiket, pengaturan kendaraan, dan koordinasi petugas. Catatan tersebut digunakan sebagai pengalaman penulis pada hari kegiatan, bukan sebagai dasar untuk menilai kinerja keseluruhan layanan.

Setelah penugasan lapangan, penulis kembali ke kantor Dinas Perhubungan pada 1 April 2026 untuk merangkum temuan operasional. Orientasi internal pada 2 April 2026 membantu penulis mengenali hubungan antara administrasi pelayanan dan koordinasi operasional pelabuhan. Pada 6 April 2026, penulis mengikuti pembahasan awal Ro-Ro Alert bersama Kepala Bidang Pelayaran. Rangkaian kegiatan tersebut menjadi dasar untuk memahami kebutuhan informasi dari sudut pandang unit kerja sebelum masuk ke proses perancangan sistem. Catatan 24 April 2026 juga memuat keikutsertaan penulis dalam pertemuan Dinas Perhubungan dengan BPTD Riau yang membahas TRIPS serta pelayanan Pelabuhan Air Putih dan Sungai Selari. Kegiatan ini digunakan sebagai konteks koordinasi antarpihak, tanpa menambahkan keputusan rapat yang tidak tercatat dalam sumber.

Hubungan kegiatan magang dengan Bidang Pelayaran terutama terletak pada pemahaman proses, pengamatan kondisi lapangan, dan identifikasi kebutuhan pencatatan operasional. Penulis mempelajari informasi yang dibutuhkan petugas

ketika memantau kedatangan dan keberangkatan kapal, mencatat kendala, serta berkoordinasi selama pelayanan berlangsung. Pengembangan Ro-Ro Alert kemudian diarahkan sebagai proyek perangkat lunak yang mendukung kebutuhan pencatatan dan pemantauan tersebut. Pada BAB II, hubungan ini dijelaskan sebatas konteks organisasi dan pekerjaan; arsitektur, fitur, basis data, integrasi, dan pengujian akan dibahas pada bab berikutnya.

Ruang lingkup Bidang Pelayanan memberikan landasan bagi pelaksanaan magang karena kegiatan penulis berhubungan langsung dengan layanan angkutan penyeberangan dan kebutuhan informasi operasional kapal Ro-Ro. Penempatan pada bidang ini juga membantu penulis memahami bahwa pengembangan sistem harus memperhatikan pembagian kewenangan, koordinasi antara kantor dan lapangan, serta prosedur pelayanan yang berlaku. Proyek Ro-Ro Alert masih ditempatkan sebagai hasil pengembangan dan bahan evaluasi; laporan ini tidak menyatakan bahwa sistem telah diterapkan secara penuh atau menghasilkan dampak operasional yang belum dibuktikan.



Gambar 2.3 Lokasi Kantor Dinas Perhubungan Kabupaten Bengkalis

BAB III

BIDANG PEKERJAAN SELAMA MAGANG

Kegiatan magang pada Dinas Perhubungan Kabupaten Bengkalis dikelompokkan berdasarkan jenis pekerjaan agar pembahasannya menunjukkan hubungan antara target, proses pelaksanaan, perangkat atau data pendukung, hasil, dan kendala. Pengelompokan ini tidak mengikuti urutan harian *logbook* secara langsung, melainkan merangkum kegiatan yang memiliki tujuan serupa ke dalam enam bidang pekerjaan. Bidang tersebut mencakup orientasi dan dukungan operasional TRIPS, analisis serta perancangan Ro-Ro Alert, pengembangan tiga komponen sistem, dan kegiatan integrasi sampai dokumentasi.

Uraian disusun dari catatan kegiatan penulis selama 10 Maret sampai 28 Juni 2026 dan dikonfirmasi dengan dokumentasi proyek yang relevan. Ro-Ro Alert terdiri atas aplikasi Flutter, PC Lokal/HMI, dan *Cloud* Laravel. Pada tahap awal, Kepala Bidang Pelayaran menjelaskan alur sistem yang diharapkan. Berdasarkan penjelasan tersebut, penulis menyusun *flowchart* dan mengembangkan seluruh komponen Ro-Ro Alert secara mandiri. Penjelasan teknis rinci mengenai rancangan dan implementasi ditempatkan pada BAB IV.

Tabel 3.1 Rekapitulasi Bidang Pekerjaan Selama Magang

No.	Bidang Pekerjaan	Periode	Kegiatan Utama	Hasil
1	Orientasi dan Dukungan Operasional TRIPS	10 Maret-2 April 2026	Orientasi instansi, pendampingan posko, pengaturan arus, pembayaran, validasi tiket, dan rekap temuan.	Pemahaman alur pelayanan serta catatan kebutuhan operasional.
2	Analisis Kebutuhan dan Perancangan Ro-Ro Alert	6-13 April 2026	Identifikasi aktor dan data, penyusunan arsitektur, alur, <i>use case</i> , <i>activity diagram</i> , serta skema data.	Rancangan tiga komponen sebagai acuan pengembangan.
3	Pengembangan <i>Cloud</i> Laravel	14 April-11 Mei 2026	Pengembangan data, autentikasi, <i>master</i> data, dasbor, laporan, API, sinkronisasi, dan komunikasi waktu nyata.	Komponen <i>Cloud</i> untuk administrasi, pemantauan, penyimpanan, dan integrasi.

No.	Bidang Pekerjaan	Periode	Kegiatan Utama	Hasil
4	Pengembangan PC Lokal dan HMI	12 Mei-1 Juni 2026	Penyimpanan lokal, HMI, ATA/ATD, TRT, kendala, audio, sinkronisasi, sinyal status, dan <i>fallback</i> .	Komponen lokal untuk pencatatan, peringatan, dan penghubung <i>Cloud</i> .
5	Pengembangan Aplikasi Flutter	2-26 Juni 2026	Fondasi aplikasi, autentikasi, <i>cache</i> , status dermaga, ATA/ATD, kendala, foto/lokasi, profil, dan mode jaringan.	Aplikasi lapangan dengan jalur LAN dan <i>fallback</i> 4G.
6	Integrasi, Pengujian, Dokumentasi, dan Presentasi	Bertahap sampai 26 Juni 2026	Integrasi antarkomponen, pemeriksaan fungsi, pencatatan temuan, perbaikan, dokumentasi, dan presentasi progres.	Temuan integrasi, perbaikan bertahap, dan dokumentasi pra-demonstrasi.

3.1. Orientasi dan Dukungan Operasional TRIPS

Tahap awal magang berlangsung pada 10 Maret sampai 2 April 2026 dengan target memahami lingkungan kerja Dinas Perhubungan Kabupaten Bengkalis dan proses pelayanan penyeberangan. Penulis menyerahkan dokumen kerja praktik, berkoordinasi mengenai ruang lingkup penugasan, dan ditempatkan pada Bidang Pelayaran. Kegiatan orientasi dan dukungan operasional TRIPS dilaksanakan oleh penulis bersama tiga rekan magang, sehingga jumlah peserta magang yang terlibat pada kegiatan ini adalah empat orang. Penempatan awal diarahkan ke Posko Pelabuhan Ro-Ro untuk mendukung kegiatan lapangan selama uji coba TRIPS. Melalui kegiatan tersebut, penulis mengenali pembagian peran antara petugas kantor, petugas posko, aparat pendukung, dan pengguna jasa.

Pada kegiatan posko, penulis membantu pemilahan arus kendaraan sesuai mekanisme pelayanan yang berlaku. Kendaraan roda empat diarahkan ke jalur pemesanan TRIPS, sedangkan kendaraan roda dua tetap menggunakan layanan loket. Penulis juga membantu pengguna memindai kode QR, memahami tahapan pemesanan, dan menyelesaikan pembayaran nontunai melalui QRIS. Dukungan tersebut dilakukan bersama petugas Dinas Perhubungan dan petugas lain di lapangan, sehingga peran penulis bersifat membantu operasional dan tidak menggantikan kewenangan petugas resmi.

Pengamatan berikutnya mencakup validasi tiket digital dan penerbitan *boarding pass*. Tiket pada ponsel atau cetakan pengguna dipindai menggunakan perangkat validasi, kemudian hasil pemeriksaan digunakan untuk melanjutkan pelayanan kendaraan. Pada beberapa kesempatan, perangkat EDC dibawa mendekati kendaraan agar proses validasi tidak mengharuskan pengemudi keluar dari antrian. Penulis juga membantu menjaga urutan kendaraan pada Posko 1 dan Posko 2 serta mengarahkan kendaraan ke jalur yang sesuai. Kegiatan tersebut memperlihatkan bahwa alur digital tetap memerlukan koordinasi fisik di gerbang pelayanan.

Berdasarkan pengamatan selama kegiatan, kondisi lapangan tidak selalu mendukung proses digital secara seragam. Pada periode peningkatan arus kendaraan, mekanisme manual digunakan sebagai alternatif untuk menjaga pelayanan tetap berjalan. Penulis menemukan pengguna yang mengalami kesulitan memahami informasi ketersediaan tiket dan pengguna yang proses pemesanannya tertunda karena koneksi seluler tidak stabil atau area dengan jangkauan terbatas. Penanganan dilakukan dengan memeriksa kembali koneksi, mengulangi tahapan pemesanan, memberi penjelasan singkat, dan mengarahkan pengguna ke mekanisme pelayanan yang ditetapkan petugas. Temuan ini tidak digunakan untuk menyatakan bahwa TRIPS gagal, melainkan untuk menggambarkan kebutuhan dukungan operasional pada kondisi tertentu.

Setelah kegiatan posko berakhir, penulis kembali ke kantor pada awal April untuk merekap temuan mengenai pembayaran, konektivitas, antarmuka, pemisahan jalur, dan kebutuhan petugas. Orientasi proses internal juga membantu penulis memahami koordinasi antara kegiatan pelayanan di lapangan dan pencatatan yang dilakukan oleh bidang terkait. Data yang digunakan pada tahap ini berupa catatan pengamatan, alur tiket dan *boarding pass*, serta informasi operasional yang dapat dilihat langsung. Tidak ada data pribadi pengguna, nilai transaksi, atau informasi sensitif yang dipindahkan ke dalam laporan.

Hasil tahap orientasi adalah pemahaman mengenai alur pelayanan penyeberangan, interaksi antara proses digital dan manual, serta pentingnya kesinambungan informasi ketika koneksi berubah. Kendala utama yang diamati

berkaitan dengan jaringan, perbedaan pemahaman pengguna, dan kebutuhan koordinasi antrean. Pengalaman tersebut menjadi konteks awal untuk mengidentifikasi kebutuhan Ro-Ro Alert, khususnya pencatatan peristiwa operasional, penyimpanan data ketika koneksi terbatas, dan penyelarasan informasi antara petugas lapangan, perangkat lokal, serta pusat pemantauan.

Selain memperoleh gambaran proses pelayanan, penulis belajar menyesuaikan komunikasi dengan pengguna yang mempunyai tingkat pemahaman digital berbeda. Arahan perlu disampaikan secara singkat tanpa menghambat kendaraan lain, sedangkan masalah teknis harus segera diteruskan kepada petugas yang berwenang apabila tidak dapat diselesaikan di posko. Pengalaman ini menegaskan bahwa kebutuhan perangkat lunak tidak hanya berkaitan dengan fungsi pencatatan, tetapi juga dengan kejelasan informasi, kemudahan penggunaan, dan tersedianya prosedur alternatif ketika sistem atau jaringan tidak dapat dipakai secara normal.



Gambar 3.1 Kegiatan Orientasi di Dinas Perhubungan Kabupaten Bengkalis

3.2. Analisis Kebutuhan dan Perancangan Ro-Ro Alert

Analisis kebutuhan dan perancangan Ro-Ro Alert dilaksanakan pada 6 sampai 13 April 2026. Tahap ini dimulai melalui diskusi mengenai kebutuhan *monitoring* operasional kapal Ro-Ro dengan menggunakan pengalaman lapangan dan tata kerja pelabuhan sebagai konteks. Target pekerjaan adalah menyusun batas sistem yang dapat membantu pencatatan waktu sandar dan berangkat, kendala

selama kapal berada di dermaga, serta penyampaian informasi antarpelaksana. Pembahasan tidak diarahkan untuk menggantikan TRIPS karena TRIPS berfokus pada layanan tiket, sedangkan Ro-Ro Alert ditempatkan pada kebutuhan *monitoring* operasional.

Kebutuhan disusun dari proses yang dapat diidentifikasi melalui observasi dan diskusi. Aktor yang dipetakan mencakup petugas lapangan, PC Lokal/HMI, *admin* pada *Cloud* Laravel, dan layanan sistem yang menghubungkan ketiganya. Data utama yang dicatat meliputi waktu sandar atau *actual time of arrival* (ATA), waktu berangkat atau *actual time of departure* (ATD), target waktu pelayanan, kondisi kendala, durasi jeda, serta status operasi. Foto dan lokasi dipertimbangkan sebagai bukti kegiatan lapangan karena kebutuhan tersebut tercatat pada perancangan kendala dan pencatatan operasional.

Hasil analisis membagi Ro-Ro Alert menjadi tiga komponen. Aplikasi Flutter ditujukan untuk petugas lapangan saat mencatat ATA, ATD, dan kendala. PC Lokal/HMI berperan mempertahankan proses operasional pada jaringan pelabuhan, menyimpan data lokal, menjalankan penghitungan waktu, dan meneruskan data ke *Cloud*. *Cloud* Laravel menyediakan pengelolaan data, pemantauan, laporan, dan komunikasi antarkomponen. Pembagian tersebut digunakan untuk memisahkan tanggung jawab data sekaligus mengurangi ketergantungan pada satu jalur koneksi.

Kebutuhan konektivitas dirancang dengan jaringan LAN sebagai jalur utama antara aplikasi Flutter dan PC Lokal. Apabila PC Lokal tidak dapat dijangkau tetapi perangkat masih memiliki koneksi internet, jalur *fallback* 4G diarahkan ke *Cloud* Laravel. Data yang lebih dahulu diterima *Cloud* kemudian perlu direkonsiliasi kembali ke PC Lokal agar status operasional tidak terpisah. Oleh karena itu, rancangan juga mencakup sinkronisasi, identitas transaksi yang konsisten, penyimpanan sementara, dan penanganan data yang tertinggal ketika komunikasi waktu nyata terputus.

Proses perancangan menghasilkan arsitektur tiga komponen, alur sistem, *use case*, *activity diagram*, dan rancangan basis data. Kepala Bidang Pelayaran terlebih dahulu menjelaskan alur sistem yang diharapkan. Penulis kemudian menyusun *flowchart*, arsitektur, *use case*, *activity diagram*, dan rancangan basis

data secara mandiri. Seluruh UML tersebut telah diverifikasi dan disetujui oleh Kepala Bidang Pelayaran. Masukan pada presentasi progres digunakan penulis untuk merapikan hubungan antarproses sebelum pengembangan dilanjutkan.

Perangkat dan data yang mendukung tahap ini meliputi catatan operasional, dokumentasi Ro-Ro Alert, serta berkas draw.io untuk arsitektur, alur, *use case*, *activity diagram*, dan skema basis data. Hasil akhirnya adalah rancangan kerja yang menjadi acuan pengembangan *Cloud* Laravel, PC Lokal/HMI, dan aplikasi Flutter. Kendala tahap perancangan terutama berupa perubahan rincian kebutuhan dan perlunya menyelaraskan kontrak data antarkomponen. Detail setiap diagram, hubungan basis data, dan alur teknis akan dibahas pada BAB IV agar BAB III tetap berfokus pada bidang pekerjaan yang dilakukan.

Batas kebutuhan juga ditetapkan agar rancangan tidak memuat kegiatan yang tidak memiliki bukti. Analisis tidak menggunakan hasil wawancara formal atau angka dampak karena sumber tersebut belum tersedia. Kebutuhan diperlakukan sebagai hasil observasi, penjelasan alur oleh Kepala Bidang Pelayaran, diskusi progres, *logbook*, dan dokumentasi proyek. Beberapa istilah teknis pada diagram berubah seiring penyelarasan antarkomponen, sehingga penulis memeriksa kembali nama data, arah komunikasi, serta tanggung jawab penyimpanan sebelum pengembangan dimulai. Pendekatan tersebut membantu penulis menjaga konsistensi fungsi setiap komponen dalam pengembangan mandiri. Rancangan juga membatasi pemakaian data sensitif dan tidak memasukkan kredensial, identitas pengguna jasa, atau konfigurasi privat ke dalam dokumen kerja.

3.3. Pengembangan *Cloud* Laravel

Pengembangan *Cloud* Laravel berlangsung pada 14 April sampai 11 Mei 2026. Komponen ini ditujukan sebagai pusat pengelolaan data, pemantauan, laporan, dan komunikasi dengan PC Lokal serta aplikasi Flutter. Seluruh pekerjaan pada komponen *Cloud* Laravel dilakukan secara mandiri oleh penulis dan dirangkum berdasarkan *logbook* serta dokumentasi proyek. Teknologi yang

digunakan mencakup Laravel, PHP, basis data MySQL atau MariaDB, autentikasi Sanctum untuk aplikasi, dan Reverb untuk komunikasi waktu nyata.

Tahap awal difokuskan pada penyusunan struktur data yang diperlukan oleh sistem. Model dan relasi disiapkan untuk pengguna, pelabuhan, kapal, dermaga, operasional, kendala, pengaturan, dan data pendukung lain. Data awal juga disusun agar lingkungan pengembangan dapat digunakan untuk demonstrasi dan pengujian. Sasaran pekerjaan ini bukan sekadar membuat tabel penyimpanan, tetapi memastikan informasi ATA, ATD, foto, koordinat, status sinkronisasi, dan kendala mempunyai hubungan yang konsisten ketika digunakan oleh tiga komponen sistem.

Setelah fondasi data tersedia, pengembangan dilanjutkan pada autentikasi *admin* dan pengelolaan *master* data. Fungsi yang dikerjakan mencakup masuk dan keluar sistem, pembatasan akses, pengelolaan akun petugas, kapal, pelabuhan, dan dermaga, serta pengaturan target waktu pelayanan. Validasi digunakan untuk menjaga keterkaitan data dan mencegah perubahan yang dapat menghilangkan riwayat operasional. Pengembangan dilakukan melalui penyesuaian bertahap terhadap kebutuhan yang muncul dari perancangan dan komunikasi antarkomponen.

Bagian pemantauan dikembangkan untuk menampilkan kondisi pelabuhan dan dermaga, kapal aktif, status berjalan atau tertunda, target ATD, serta status PC Lokal. Dasbor juga mengolah ringkasan kegiatan harian dan indikator kinerja yang berasal dari data operasional. Modul laporan disiapkan untuk menyaring data berdasarkan tanggal, kapal, dermaga, dan pelabuhan, disertai informasi kendala serta petugas yang terkait. Hasil pekerjaan pada tahap ini menyediakan tampilan administratif untuk membaca data yang telah diterima *Cloud* tanpa memindahkan proses penghitungan lokal secara tidak perlu.

Integrasi *Cloud* dikembangkan melalui layanan API untuk registrasi perangkat, sinyal status berkala, pengambilan *master* data, sinkronisasi operasional, unggahan foto, dan siklus kendala. Jalur *fallback* 4G juga disiapkan agar aplikasi Flutter dapat mengirim peristiwa sandar, berangkat, atau kendala ketika PC Lokal tidak dapat dijangkau. Komunikasi Reverb digunakan untuk menyampaikan perubahan pengaturan, sinyal keadaan darurat, dan kendali jarak jauh. *Cloud*

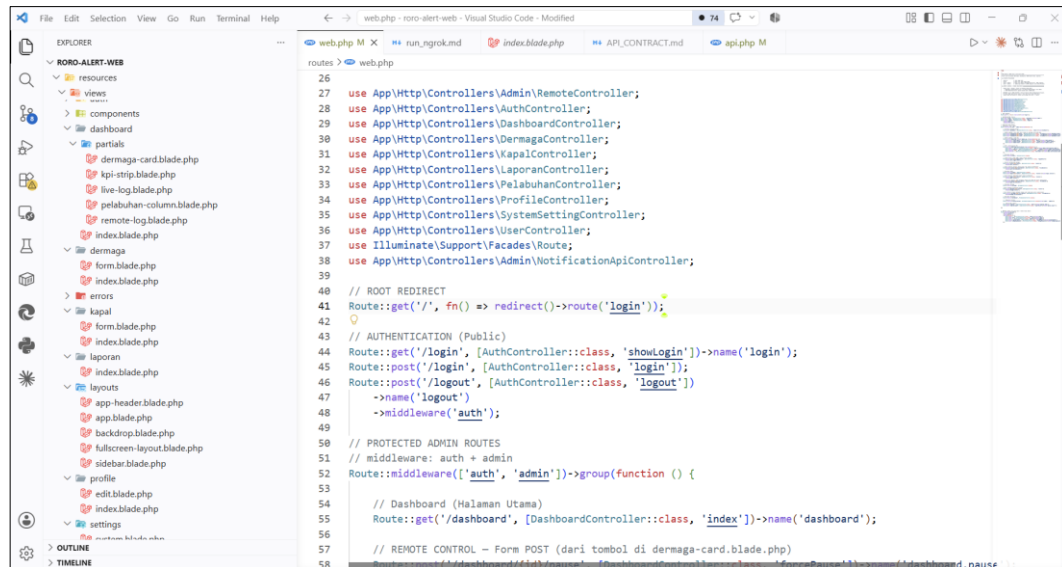
mencatat status perintah, sedangkan perubahan operasional tetap dijalankan oleh PC Lokal sesuai pembagian tanggung jawab sistem.

Selama pengembangan, *payload* dan respons API beberapa kali disesuaikan agar identitas petugas, status konflik, foto, waktu, serta kode kesalahan dapat dipahami secara sama oleh komponen lain. Pengujian dilakukan terhadap rute *web* dan API, aktivitas pada dasbor, alur *fallback* kendala, unggahan foto, dan komunikasi waktu nyata. Pada 11 Mei, progres *Cloud* Laravel didemonstrasikan bersama Kepala Bidang untuk memperlihatkan pemantauan, *master* data, laporan, sinkronisasi, dan kendali jarak jauh. Demonstrasi tersebut digunakan sebagai evaluasi progres dan tidak diperlakukan sebagai bukti penerapan penuh.

Hasil bidang pekerjaan ini adalah komponen *Cloud* yang dapat menampung data operasional, menyediakan fungsi administrasi, menampilkan pemantauan dan laporan, serta berkomunikasi dengan dua komponen lainnya. Kendala yang ditemukan meliputi perubahan kontrak API, kebutuhan konsistensi waktu dan identitas, serta penyelarasan status ketika koneksi tidak tersedia. Penanganan dilakukan melalui validasi, penggunaan identitas transaksi yang tidak berubah, pencatatan status sinkronisasi, dan pembaruan dokumentasi kontrak. Pembahasan struktur program, daftar *endpoint*, dan model basis data secara rinci ditempatkan pada BAB IV.

Perangkat pendukung pada fase *Cloud* meliputi lingkungan pengembangan Laravel, basis data relasional, *browser* untuk memeriksa dasbor, serta dokumentasi kontrak untuk menyelaraskan permintaan dari PC Lokal dan Flutter. Data yang digunakan berupa data contoh kapal, dermaga, petugas, pengaturan waktu, dan catatan operasi untuk kebutuhan pengembangan; data tersebut tidak diperlakukan sebagai data operasional resmi instansi. Penulis menyusun dan menyesuaikan fungsi *Cloud* sekaligus melakukan integrasi lintas komponen secara mandiri. Kredensial, *token*, alamat *server* privat, dan konfigurasi rahasia tidak dicantumkan dalam laporan. Status setiap fungsi diperiksa berdasarkan catatan periode pengembangan, sehingga fitur yang diperbarui setelah periode tersebut dibedakan dari hasil yang telah tersedia selama magang. Evaluasi akhir fase berfokus pada

kesiapan untuk integrasi berikutnya dan catatan temuan dipakai untuk menentukan bagian yang perlu diselaraskan pada PC Lokal.



Gambar 3.2 Kegiatan Pengembangan *Cloud* Laravel

3.4. Pengembangan PC Lokal dan HMI

Pengembangan PC Lokal dan HMI dilaksanakan pada 12 Mei sampai 1 Juni 2026. Komponen ini dirancang sebagai perangkat tepi di pelabuhan yang tetap dapat melayani pencatatan operasional pada jaringan lokal dan meneruskan data ke *Cloud* ketika koneksi tersedia. Target pekerjaan mencakup penyimpanan data lokal, tampilan status dermaga, penghitungan waktu pelayanan, peringatan audio, dan mekanisme sinkronisasi. Dokumentasi proyek mengonfirmasi penggunaan PHP dan JavaScript untuk layanan serta HMI, dengan MySQL atau MariaDB sebagai basis data lokal.

Fondasi komponen disusun melalui skema data untuk *buffer* operasional, *master* kapal, *master* dermaga, konfigurasi sistem, riwayat kendala, dan status sinkronisasi. HMI kemudian dikembangkan dalam bentuk kartu dermaga yang menampilkan kapal aktif, target waktu, status berjalan atau *pause*, indikator keterlambatan, serta kondisi *Cloud* dan sinkronisasi. Pemutakhiran status dilakukan secara berkala dari layanan lokal sehingga petugas dapat melihat keadaan dermaga tanpa bergantung pada tampilan *Cloud*.

Layanan operasional lokal mendukung pencatatan sandar atau ATA dan berangkat atau ATD. Data yang diproses meliputi identitas operasi, kapal, dermaga, petugas, waktu, foto, dan koordinat. PC Lokal menghitung target waktu serta *turn round time* (TRT) berdasarkan data operasi dan konfigurasi yang berlaku. Ketika kendala terjadi, fungsi *pause* dan *resume* digunakan untuk mencatat awal serta akhir kendala dan menyesuaikan akumulasi waktu jeda. Penjelasan rumus dan algoritma penghitungan tidak diuraikan pada bab ini.

Peringatan audio disiapkan untuk membantu pemantauan waktu pada HMI. Sistem memuat tahapan peringatan sebelum target dan sirine ketika kondisi melewati target yang ditetapkan. Pengembangan juga memperhatikan risiko suara terpicu lebih dari satu kali ketika beberapa *tab* HMI terbuka atau halaman dimuat ulang. Penanganan dilakukan melalui pengunci, penanda audio yang telah dijalankan, dan mekanisme aktivasi suara oleh pengguna. Pengujian audio dicatat secara umum karena angka skenario pada *logbook* belum didukung laporan uji terpisah yang telah disahkan.

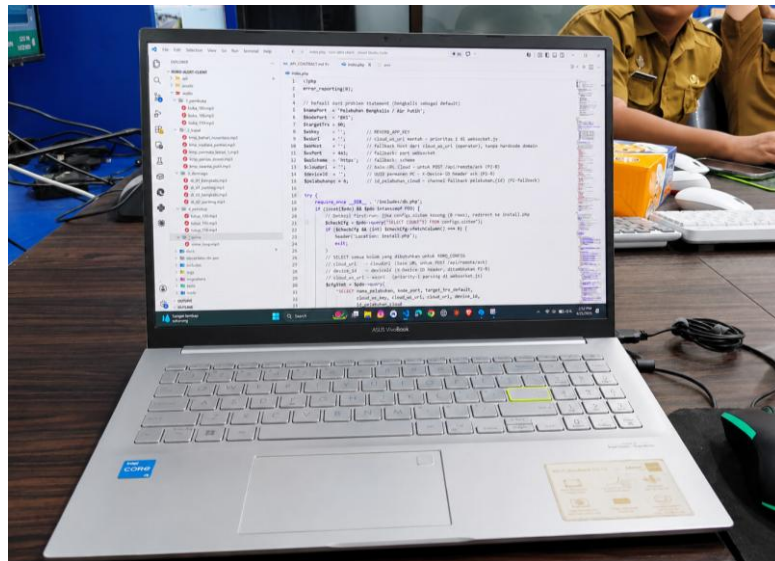
Sinkronisasi PC Lokal ke *Cloud* mencakup pengiriman data operasional, foto ATA/ATD, data kendala, dan foto kendala. Proses tarik data digunakan untuk memperbarui *master* kapal, dermaga, serta konfigurasi TRT. Sinyal status berkala membantu *Cloud* mengetahui kondisi perangkat, sedangkan Reverb digunakan untuk menerima pengaturan, kendali jarak jauh, dan data *fallback* yang lebih dahulu masuk ke *Cloud*. Setiap perintah kendali dilaksanakan secara lokal dan hasilnya dikirim kembali sebagai catatan status.

Untuk menghadapi gangguan koneksi waktu nyata, komponen lokal dilengkapi rekonsiliasi data yang tertinggal. Data *fallback* dapat diterima melalui sinyal Reverb atau diambil kembali ketika koneksi pulih. Pengujian *endpoint* lokal didokumentasikan menggunakan Postman dan diselaraskan dengan kontrak API. Pengujian *helper* sinkronisasi, audio, dan demonstrasi progres dilakukan pada akhir fase, tetapi laporan ini tidak mencantumkan jumlah pengujian karena bukti pengujian formal belum tersedia di luar catatan *logbook*.

Hasil pekerjaan berupa PC Lokal/HMI yang mendukung pencatatan lokal, tampilan keadaan dermaga, penghitungan waktu, peringatan, serta sinkronisasi ke

Cloud. Kendala utamanya adalah menjaga konsistensi data pada perubahan jaringan, menghindari pengulangan pengiriman, dan memastikan peringatan audio tidak berganda. Penanganan dilakukan dengan *buffer*, penanda status sinkronisasi, pengiriman ulang terkontrol, rekonsiliasi, dan pengunci audio. Pengembangan ini tetap memerlukan validasi lebih lanjut pada konfigurasi perangkat dan lingkungan lapangan yang sebenarnya.

Data uji pada fase ini menggunakan data contoh operasi, konfigurasi dermaga, pengaturan TRT, foto uji, dan keadaan sinkronisasi. Pemeriksaan dilakukan dari HMI, layanan lokal, Postman, dan respons *Cloud* untuk menelusuri apakah perubahan keadaan dicatat dengan urutan yang benar. Ketika ditemukan perbedaan, perbaikan dilakukan pada validasi, pemetaan data, atau mekanisme pemulihan, kemudian fungsi terkait diuji kembali. Walaupun *logbook* mencatat pengujian *helper* dan audio, hasil tersebut belum dipakai sebagai angka keberhasilan resmi. Konfigurasi *speaker*, jaringan, dan perangkat PC yang akan digunakan di pelabuhan juga masih perlu disesuaikan sebelum evaluasi lapangan. Selain itu, proses sinkronisasi harus membedakan data yang menunggu, gagal sementara, atau tidak boleh diulang karena konflik. Pemisahan status tersebut membantu pemeriksaan temuan tanpa menghapus data lokal dan menjadi dasar bagi perbaikan mekanisme rekonsiliasi berikutnya. Pengamatan terhadap indikator HMI juga digunakan untuk memastikan petugas dapat mengenali keadaan lokal, kondisi koneksi, serta data yang belum terkirim tanpa membuka basis data secara langsung. Setiap perbaikan dicatat agar pengujian ulang dapat diarahkan pada fungsi yang berubah. Hal ini mempermudah penelusuran kesalahan selama integrasi berikutnya.



Gambar 3.3 Kegiatan Pengembangan PC Lokal dan HMI

3.5. Pengembangan Aplikasi Flutter

Pengembangan aplikasi Flutter berlangsung pada 2 sampai 26 Juni 2026 dengan sasaran menyediakan sarana pencatatan bagi petugas lapangan. Aplikasi dirancang untuk berkomunikasi dengan PC Lokal melalui LAN sebagai jalur utama dan menggunakan *Cloud* melalui jaringan 4G ketika PC tidak dapat dijangkau. Penulis merancang dan mengembangkan komponen Flutter secara mandiri dengan mengacu pada *flowchart* dan arsitektur yang telah disetujui Kepala Bidang Pelayaran. Teknologi yang digunakan mencakup Flutter, Dart, Drift/SQLite, penyimpanan *token* yang aman, kamera, dan GPS.

Tahap fondasi mencakup penyiapan struktur fitur, tema, navigasi, manajemen keadaan, serta basis data lokal. Drift digunakan untuk menyimpan sesi, *master* kapal dan dermaga, konfigurasi, riwayat operasional, dan keadaan pencatat jeda. Penyimpanan lokal dibutuhkan agar data *master* dan konteks operasi tetap tersedia ketika jaringan berubah. Autentikasi terhubung ke *Cloud*, sedangkan *token* disimpan melalui mekanisme penyimpanan aman. Jika sinkronisasi *master* gagal tetapi data lokal tersedia, aplikasi dapat mempertahankan informasi yang diperlukan untuk melanjutkan pemeriksaan.

Pengelolaan koneksi membedakan mode LAN, *fallback* 4G, dan tidak ada koneksi. Pada mode LAN, aplikasi membaca status dermaga dan mengirim operasi ke PC Lokal. Ketika PC tidak merespons tetapi perangkat memiliki internet, permintaan dialihkan ke *Cloud*. Pemilihan jalur dilakukan pada saat operasi agar keputusan tidak hanya bergantung pada kondisi ketika halaman dibuka. Data yang lebih dahulu masuk melalui *Cloud* kemudian diselaraskan kembali dengan PC Lokal melalui mekanisme komunikasi dan rekonsiliasi yang telah disiapkan.

Halaman utama dikembangkan untuk menampilkan status dermaga, kapal aktif, ringkasan kegiatan, riwayat hari berjalan, dan mode jaringan. Alur ATA meminta pemilihan kapal serta dermaga, foto, lokasi, waktu, dan identitas transaksi. Alur ATD menggunakan identitas operasi yang sama untuk menutup kegiatan dan mencatat foto serta lokasi keberangkatan. Foto dan GPS digunakan sebagai bagian dari data operasional karena kebutuhan tersebut tercatat dalam *logbook* dan dokumentasi, tetapi laporan tidak menampilkan foto pengguna atau koordinat yang bersifat sensitif.

Aplikasi juga menyediakan pelaporan dan penyelesaian kendala. Petugas dapat memilih kategori, menambahkan catatan, mengambil foto, dan menjalankan fungsi *pause* atau *resume* pada operasi aktif. Durasi jeda dipertahankan secara lokal agar perubahan jaringan tidak menghilangkan konteks kendala. Fitur riwayat membantu membaca status pengiriman, sedangkan halaman profil digunakan untuk informasi petugas, pengaturan alamat PC Lokal, pengujian koneksi, dan preferensi tampilan. Cakupan fitur tetap disesuaikan dengan catatan pengembangan, bukan seluruh kemungkinan yang terdapat pada dokumentasi setelah masa magang.

Pengujian dilakukan pada *parser* status, autentikasi, sinkronisasi *master*, identitas transaksi, GPS dan *geofence*, basis data lokal, pencatat jeda, serta penanganan konflik kapal dan dermaga. Penolakan dari *server* dipisahkan dari gangguan yang masih dapat dicoba kembali agar aplikasi tidak menggandakan operasi. Penguatan lain mencakup pembatasan tangkapan layar pada halaman tertentu, pemeriksaan perangkat, pengurangan *log* sensitif, dan pencatatan galat yang tidak menyimpan *token* atau koordinat. Angka keberhasilan dan jumlah

skenario tidak dicantumkan karena bab ini tidak menyajikan laporan pengujian formal.

Hasil bidang pekerjaan ini adalah aplikasi lapangan yang mempunyai fondasi autentikasi, *cache* lokal, status dermaga, pencatatan ATA/ATD, kendala, riwayat, profil, serta pemilihan jalur LAN dan 4G. Kendala yang dihadapi meliputi perubahan kontrak API, konsistensi status antara perangkat, data lokal yang belum tersinkron, dan perbedaan kondisi perangkat uji. Perbaikan dilakukan melalui validasi, penyimpanan lokal, identitas transaksi yang konsisten, penanganan konflik, serta pengujian bertahap. Status *build*, penandatanganan APK, dan validasi pada perangkat lapangan masih memerlukan bukti tambahan.

Data pengembangan aplikasi meliputi *master* kapal dan dermaga, identitas petugas, status operasi, foto uji, lokasi uji, serta respons dari PC Lokal dan *Cloud*. Penyimpanan lokal membantu memisahkan data yang sudah terkirim, gagal, atau perlu diperiksa kembali. Dalam prosesnya, penulis juga menyesuaikan tampilan dan pesan galat agar petugas dapat membedakan masalah koneksi dari penolakan aturan operasi. Dokumentasi sampai 26 Juni merangkum cara menyiapkan aplikasi, batasan jaringan, dan pemeriksaan pra-demonstrasi. Versi perangkat lunak setelah periode magang tidak digunakan sebagai dasar narasi fase ini. Aplikasi juga belum dinyatakan sebagai rilis operasional karena bukti penandatanganan, distribusi resmi, dan penerimaan pengguna belum tersedia. Pengujian lapangan dengan petugas pada dua lokasi pelabuhan juga belum dapat dinyatakan selesai berdasarkan bukti yang ada. Temuan aplikasi diteruskan sebagai bahan penyelarasan integrasi akhir. Status tersebut membantu membedakan data lokal dan data *Cloud*.



Gambar 3.4 Kegiatan Pengembangan Aplikasi Flutter

3.6. Integrasi, Pengujian, Dokumentasi, dan Presentasi

Integrasi, pengujian, dokumentasi, dan presentasi dilakukan secara bertahap pada akhir setiap fase, bukan sebagai satu pekerjaan yang hanya berlangsung pada akhir magang. Targetnya adalah memastikan aplikasi Flutter, PC Lokal/HMI, dan *Cloud* Laravel menggunakan struktur data serta urutan operasi yang selaras. Pengujian pada bab ini diperlakukan sebagai pemeriksaan selama pengembangan dan bukan bukti kelulusan uji penerimaan, penggunaan resmi oleh instansi, atau penyelesaian administratif proyek.

Integrasi PC Lokal dengan *Cloud* mencakup registrasi perangkat, pembaruan *master* data, pengiriman data operasional dan foto, sinkronisasi kendala, sinyal status berkala, serta kendali jarak jauh. PC Lokal menyimpan data terlebih dahulu ketika koneksi tidak tersedia dan mengirimkannya setelah jalur kembali dapat digunakan. *Cloud* mencatat data terpusat dan mengirim sinyal melalui Reverb, sedangkan PC tetap menjalankan perubahan keadaan operasi sesuai pembagian tanggung jawab. Temuan mengenai identitas perangkat, urutan waktu, dan status sinkronisasi diperbaiki melalui validasi serta pembaruan kontrak data.

Integrasi Flutter dengan PC Lokal diperiksa melalui jalur LAN untuk membaca status dermaga dan mencatat ATA, kendala, serta ATD. Jalur *fallback* 4G diperiksa untuk kondisi ketika PC tidak dapat dijangkau tetapi perangkat masih

memiliki internet. Setelah data masuk melalui *Cloud*, PC Lokal menerima sinyal atau mengambil data yang tertinggal ketika koneksi pulih. Mekanisme tersebut membutuhkan identitas operasi yang sama, penanganan pengiriman ulang, dan rekonsiliasi agar satu kegiatan tidak tercatat sebagai dua operasi berbeda.

Pengujian fungsi utama dilakukan selama setiap fase dan mencakup rute *web/API*, perubahan status, unggahan foto, sinkronisasi, peringatan audio, *parser* data, autentikasi, basis data lokal, GPS/*geofence*, dan penanganan konflik. Temuan dicatat dan diperbaiki sebelum pengujian berikutnya. Karena sumber yang tersedia berupa *logbook* dan dokumentasi pengembangan, jumlah skenario serta tingkat keberhasilan tidak dilaporkan sebagai hasil formal. Beberapa bagian, khususnya perilaku pada perangkat dan jaringan lapangan, masih membutuhkan validasi lanjutan.

Dokumentasi disusun untuk menjaga kesesuaian tiga komponen. Bahan yang dihasilkan meliputi penjelasan arsitektur, kontrak API, alur LAN dan 4G, strategi sinkronisasi, pengaturan lingkungan, serta daftar pemeriksaan pra-demonstrasi. Dokumentasi GitBook yang bertanggal sampai 26 Juni digunakan untuk mengonfirmasi pekerjaan dalam periode magang. Perubahan GitBook pada 6 sampai 8 Juli 2026 diperlakukan sebagai pembaruan lanjutan setelah periode magang dan tidak dimasukkan sebagai kegiatan resmi dalam rentang tersebut.

Presentasi progres dilakukan pada tahap perancangan, setelah penyusunan basis data *Cloud*, pada akhir pengembangan *Cloud*, dan pada akhir fase PC Lokal/HMI. *Logbook* juga memuat entri demonstrasi APK serta presentasi akhir pada 29 Juni 2026. Karena tanggal tersebut berada satu hari setelah akhir periode magang 28 Juni dan seluruh *logbook* masih berstatus *draft*, kegiatan itu dicatat sebagai kegiatan lanjutan yang memerlukan konfirmasi administratif. Hasil yang dapat dinyatakan sampai akhir periode adalah integrasi bertahap, perbaikan berdasarkan temuan, dokumentasi pra-demonstrasi, dan tersedianya komponen untuk evaluasi lebih lanjut.

Kendala lintas fase yang paling sering memerlukan perhatian adalah perubahan kontrak data, keterlambatan sinkronisasi, perbedaan status antara jalur LAN dan 4G, serta keterbatasan bukti pengujian formal. Penanganannya dirangkum

pada Tabel 3.3 berdasarkan catatan yang benar-benar tersedia. Tabel 3.2 juga membedakan perangkat dan data yang digunakan pada setiap kelompok pekerjaan agar teknologi tidak dicantumkan hanya berdasarkan kemungkinan. Dengan batas tersebut, hasil BAB III merupakan catatan bidang pekerjaan dan progres pengembangan, sedangkan keputusan implementasi, uji penerimaan, dan dampak sistem tidak disimpulkan tanpa bukti tambahan. Foto kegiatan, peserta presentasi, umpan balik tertulis, dan status pengesahan *logbook* tetap dicatat sebagai data yang perlu dilengkapi. Batasan tersebut menjaga agar laporan tidak melampaui bukti kegiatan yang tersedia. Dengan demikian, hasil tahap ini tetap dapat ditelusuri kembali ke sumbernya.



Gambar 3.5 Demonstrasi dan Presentasi Hasil Proyek

Tabel 3.2 Perangkat Lunak dan Data Pendukung

No.	Perangkat Lunak/Data	Penggunaan	Bidang Pekerjaan	Keterangan
1	TRIPS, QRIS, EDC, tiket dan <i>boarding pass</i>	Mendukung pelayanan tiket, pembayaran, validasi, dan pengamatan alur lapangan.	3.1	Digunakan pada kegiatan posko; tidak memuat data pribadi pengguna.
2	draw.io dan berkas diagram	Menyusun arsitektur, alur sistem, <i>use case</i> , <i>activity diagram</i> , dan skema data.	3.2	Diagram disimpan untuk pembahasan rinci pada BAB IV.
3	Laravel, PHP, MySQL/MariaDB, dan Reverb	Mengembangkan <i>Cloud</i> , penyimpanan terpusat, dasbor, laporan, API, dan komunikasi waktu nyata.	3.3 dan 3.6	Versi teknologi tidak dicantumkan karena tidak diperlukan pada BAB III.
4	PHP, JavaScript, MySQL/MariaDB, dan audio lokal	Mengembangkan layanan PC Lokal, HMI, penyimpanan, peringatan, serta sinkronisasi.	3.4 dan 3.6	Konfigurasi rahasia dan alamat privat tidak dicantumkan.
5	Flutter, Dart, Drift/SQLite, kamera, dan GPS	Mengembangkan aplikasi petugas, penyimpanan lokal, foto, lokasi, dan pencatatan operasi.	3.5	Data sensitif, <i>token</i> , foto pengguna, dan koordinat tidak dicantumkan.
6	Postman dan kontrak API	Memeriksa <i>endpoint</i> , <i>payload</i> , respons, kode kesalahan, dan kesesuaian antarkomponen.	3.3-3.6	Hasil dijelaskan secara umum tanpa angka uji yang belum disahkan.
7	GitBook dan dokumentasi proyek	Mencatat arsitektur, alur jaringan, sinkronisasi, API, batasan, dan pemeriksaan pra-demonstrasi.	3.2-3.6	Pembaruan Juli dipisahkan dari periode magang.
8	Jaringan LAN/4G dan data operasional	Menguji jalur utama, jalur <i>fallback</i> , sinkronisasi, dan rekonsiliasi data.	3.2-3.6	Tidak mencantumkan kredensial atau alamat <i>server</i> privat.

Tabel 3.3 Kendala dan Penyelesaian Selama Magang

No.	Kendala	Dampak	Penanganan	Status
1	Koneksi seluler tidak stabil pada kegiatan TRIPS	Pemesanan atau validasi pengguna dapat tertunda.	Memeriksa koneksi, mengulangi proses, memberi pendampingan, dan mengikuti alternatif pelayanan petugas.	Selesai dengan penyesuaian

No.	Kendala	Dampak	Penanganan	Status
2	Perubahan kebutuhan dan kontrak data	<i>Payload</i> atau respons antarkomponen dapat tidak seragam.	Menyelaraskan dokumentasi, identitas petugas, waktu, foto, dan kode konflik.	Selesai dengan penyesuaian
3	Sinkronisasi terputus atau data tertinggal	Status PC Lokal dan <i>Cloud</i> dapat berbeda sementara.	Menggunakan <i>buffer</i> , status sinkronisasi, pengiriman ulang terkontrol, sinyal, dan rekonsiliasi.	Masih memerlukan pengembangan
4	Peringatan audio berpotensi terpicu ganda	Peringatan pada HMI dapat berulang.	Menambahkan pengunci, penanda audio, dan aktivasi suara oleh pengguna.	Selesai dengan penyesuaian
5	Konflik kapal atau dermaga	Operasi berisiko tercatat ganda atau memakai sumber daya yang sedang aktif.	Menggunakan identitas transaksi, validasi keadaan, dan respons konflik tanpa pengiriman ulang otomatis.	Selesai dengan penyesuaian
6	Perbedaan kondisi perangkat dan mode jaringan	Perilaku aplikasi dapat berbeda antara LAN, 4G, dan tanpa koneksi.	Menerapkan <i>cache</i> lokal, pemilihan jalur per operasi, dan pengujian bertahap.	Perlu validasi lebih lanjut
7	Tanggal demo akhir tercatat 29 Juni 2026	Status kegiatan terhadap akhir periode 28 Juni belum pasti.	Memisahkan entri sebagai kegiatan lanjutan dan menunggu konfirmasi administratif.	Perlu validasi lebih lanjut

BAB IV

RANCANG BANGUN SISTEM RO-RO ALERT BERBASIS FLUTTER *MOBILE*, PC LOKAL, DAN *CLOUD* LARAVEL UNTUK *MONITORING* OPERASIONAL KAPAL RO-RO PADA DINAS PERHUBUNGAN KABUPATEN BENGKALIS

4.1. Metodologi

Metodologi pada pembangunan Ro-Ro Alert disusun berdasarkan kegiatan yang tercatat selama magang dan artefak rancangan yang dihasilkan. Pembahasan tidak menetapkan nama metode formal yang tidak tercantum dalam *logbook*. Rangkaian kegiatan menunjukkan prosedur pengembangan bertahap yang bergerak dari pengamatan kondisi operasional, analisis kebutuhan, perancangan, pengembangan tiga komponen, integrasi, pengujian bertahap, hingga evaluasi dan dokumentasi.

Bagian metodologi mencakup empat unsur yang saling berhubungan. Prosedur pembuatan sistem menjelaskan urutan pembangunan solusi, metodologi pengumpulan data menerangkan sumber informasi yang digunakan, proses perancangan menunjukkan perubahan kebutuhan menjadi artefak teknis, sedangkan tahapan dan jadwal pelaksanaan merangkum realisasi pekerjaan berdasarkan *logbook*. Susunan tersebut digunakan agar hubungan antara kondisi lapangan, keputusan rancangan, dan hasil pengembangan dapat ditelusuri secara sistematis.

4.1.1. Prosedur Pembuatan Sistem/Solusi

Pembuatan Ro-Ro Alert diawali dengan identifikasi kondisi pelayanan penyeberangan dan kebutuhan pencatatan operasional. Pengamatan selama penugasan di lingkungan pelabuhan memberikan gambaran mengenai alur kendaraan, validasi layanan, koordinasi petugas, serta pengaruh konektivitas terhadap proses digital. Catatan tersebut tidak langsung diterjemahkan menjadi fitur. Penulis terlebih dahulu mengelompokkan masalah yang relevan dengan

pemantauan waktu sandar dan berangkat, pencatatan kendala, penyediaan bukti kegiatan, dan kebutuhan pemantauan oleh pihak pengelola.

Tahap berikutnya berupa analisis kebutuhan melalui diskusi dan penyelarasan dengan proses kerja Bidang Pelayaran. Kebutuhan yang muncul dipisahkan menurut aktor, data, dan kewenangan. Dari tahap ini ditetapkan rancangan tiga komponen: aplikasi Flutter untuk petugas lapangan, PC Lokal/HMI sebagai pengendali operasi normal di area pelabuhan, serta *Cloud* Laravel sebagai pusat data, administrasi, pemantauan, dan laporan. Pembagian tersebut mencegah satu komponen mengambil seluruh tanggung jawab serta menyediakan jalur kerja ketika kualitas jaringan berubah.

Setelah batas komponen ditentukan, penulis menyusun arsitektur, alur umum, *use case*, *activity diagram*, dan rancangan basis data. Artefak ini dipakai untuk memeriksa apakah setiap fungsi memiliki aktor, data masukan, keputusan, hasil, dan tempat penyimpanan yang sesuai. Jalur normal dirancang menggunakan jaringan lokal antara aplikasi Flutter dan PC Lokal, kemudian PC Lokal menyinkronkan data ke *Cloud*. Jalur alternatif melalui jaringan 4G dirancang untuk kondisi ketika perangkat tidak dapat menjangkau PC Lokal.

Perancangan diikuti pengembangan secara bertahap. *Cloud* Laravel dibangun lebih dahulu untuk menyediakan data *master*, autentikasi, penyimpanan terpusat, pemantauan, laporan, serta kontrak integrasi. PC Lokal/HMI kemudian dikembangkan sebagai penyimpan data lokal, pengelola waktu operasional, peringatan audio, dan penghubung ke *Cloud*. Aplikasi Flutter dikembangkan sebagai sarana pencatatan lapangan yang dapat memilih jalur LAN atau 4G. Urutan tersebut mengikuti ketersediaan layanan yang diperlukan oleh komponen berikutnya, bukan menunjukkan metode linier yang tidak memungkinkan perubahan.

Setiap komponen diperiksa bersama kontrak data dan perilaku komponen lain. Temuan pada satu bagian dapat menghasilkan penyesuaian pada bagian sebelumnya, misalnya penyelarasan identitas operasi, status kendala, format waktu, hubungan foto dengan data utama, serta cara menangani konflik kapal atau dermaga. Dengan demikian, prosedur pengembangan memiliki sifat berulang pada

tingkat artefak dan integrasi. Perubahan dilakukan untuk menjaga arti data tetap sama ketika berpindah dari perangkat lapangan ke PC Lokal dan *Cloud*.

Tahap integrasi dan pengujian dilakukan sepanjang pengembangan, bukan hanya setelah seluruh komponen selesai. Pemeriksaan mencakup fungsi utama, respons kegagalan, penyimpanan lokal, sinkronisasi, jalur *fallback*, dan konsistensi status. Temuan dicatat untuk perbaikan berikutnya, sedangkan dokumentasi digunakan untuk menjaga kesesuaian antarkomponen. Karena *logbook* belum disahkan dan tidak mencantumkan penetapan metode formal, laporan ini menyebut pendekatan tersebut sebagai prosedur pengembangan bertahap melalui analisis, perancangan, pengembangan, integrasi, pemeriksaan, dan evaluasi.

Pada setiap perpindahan tahap, hasil sebelumnya dipakai sebagai dasar pemeriksaan tahap berikutnya. Temuan observasi menjadi masukan analisis, hasil analisis ditelusurkan ke diagram, dan diagram dibandingkan dengan kebutuhan penyimpanan serta integrasi. Apabila ditemukan perbedaan, artefak yang terkait diselaraskan sebelum fungsi dilanjutkan. Pemeriksaan keterlacakan ini penting karena satu peristiwa operasional diproses oleh lebih dari satu komponen dan dapat melalui jalur komunikasi yang berbeda.

Dokumentasi disusun bersamaan dengan perkembangan rancangan untuk mencatat batas tanggung jawab, aturan pertukaran data, dan keputusan yang perlu dikonfirmasi. Hasil tiap tahap diperlakukan sebagai dasar kerja yang dapat diperbarui, bukan sebagai persetujuan akhir. Dengan prosedur tersebut, pembangunan tetap mempunyai urutan yang dapat dijelaskan tanpa mengabaikan penyesuaian berulang yang muncul selama integrasi *Cloud*, PC Lokal, dan aplikasi Flutter.

4.1.2. Metodologi Pengumpulan Data

Data untuk perancangan Ro-Ro Alert diperoleh melalui observasi, diskusi, dokumentasi, studi dokumen, dan pencatatan *logbook*. Kelima cara tersebut digunakan sesuai kebutuhan fase pekerjaan. Kegiatan yang berlangsung tidak didokumentasikan sebagai wawancara formal karena tidak tersedia daftar

narasumber, instrumen pertanyaan, atau berita acara wawancara. Pertukaran informasi dengan pembimbing lapangan dan pihak bidang karena itu dinyatakan sebagai diskusi kerja yang digunakan untuk memahami kebutuhan dan memeriksa kesesuaian rancangan.

Observasi dilakukan terutama pada fase awal magang ketika penulis mengikuti orientasi dan mendukung kegiatan operasional di pelabuhan. Objek yang diamati meliputi alur pelayanan kendaraan, penggunaan tiket digital, validasi, *boarding pass*, pembayaran nontunai, pemisahan proses digital dan manual, serta kendala konektivitas. Informasi dari observasi dipakai untuk mengenali kebutuhan pencatatan yang tetap dapat berjalan pada kondisi jaringan berbeda, kebutuhan bukti foto dan lokasi, serta pentingnya informasi status yang dapat dilihat oleh petugas dan pengelola.

Diskusi digunakan pada fase analisis dan perancangan untuk merumuskan aktor, data operasional, pembagian komponen, serta alur normal dan alternatif. Kepala Bidang Pelayaran menjelaskan alur sistem yang diharapkan, kemudian penulis menyusun seluruh *flowchart* dan UML secara mandiri. Rancangan tersebut dipresentasikan, diverifikasi, dan disetujui oleh Kepala Bidang Pelayaran sebelum dijadikan acuan pengembangan.

Dokumentasi dan studi dokumen mencakup catatan kegiatan, diagram, struktur kebutuhan, dokumentasi arsitektur, serta kontrak integrasi proyek. Diagram membantu memeriksa urutan proses dan tanggung jawab aktor, sedangkan dokumentasi arsitektur digunakan untuk mengonfirmasi batas aplikasi Flutter, PC Lokal/HMI, dan *Cloud* Laravel. Dokumentasi yang diperbarui setelah masa magang hanya dipakai untuk mengonfirmasi struktur teknis yang juga didukung catatan dan diagram, bukan sebagai bukti bahwa kegiatan tersebut dilakukan pada Juli 2026.

Logbook berfungsi sebagai sumber kronologi untuk menentukan periode dan urutan pekerjaan. Entri yang relevan dikelompokkan berdasarkan analisis, perancangan, pengembangan komponen, integrasi, dan dokumentasi. Informasi tersebut dibandingkan dengan diagram serta dokumentasi agar kebutuhan yang ditulis mempunyai lebih dari satu dasar. Seluruh entri masih berstatus *draft* dan

belum disahkan, sehingga *logbook* digunakan sebagai catatan kegiatan penulis, bukan sebagai bukti persetujuan resmi atau penerapan sistem secara penuh.

Keabsahan informasi dijaga melalui pembandingan antarsumber. Kronologi dari *logbook* diperiksa terhadap diagram yang dibuat pada fase perancangan, sedangkan batas teknis pada dokumentasi dibandingkan dengan kebutuhan yang tercatat. Jika suatu rincian hanya ditemukan pada satu sumber atau belum mempunyai persetujuan, uraian diberi batas sebagai rancangan atau hal yang masih memerlukan konfirmasi. Cara ini juga diterapkan pada tanggal kegiatan, nilai konfigurasi, radius lokasi, dan status penerapan sistem.

Pengumpulan data tidak mencakup survei kuantitatif, pengukuran performa produksi, atau pengujian penerimaan pengguna karena bukti untuk kegiatan tersebut tidak tersedia. Oleh sebab itu, laporan tidak menyajikan persentase keberhasilan, waktu respons, ataupun kesimpulan penerimaan. Data yang digunakan terbatas pada informasi yang diperlukan untuk menjelaskan proses pembangunan dan rancangan Ro-Ro Alert secara dapat ditelusuri.

4.1.3. Proses Perancangan

Proses perancangan dimulai dengan mengubah temuan operasional menjadi pernyataan kebutuhan yang dapat ditangani sistem. Alur kerja pelabuhan diuraikan menjadi peristiwa sandar, pemantauan waktu, pelaporan kendala, penyelesaian kendala, dan keberangkatan. Setiap peristiwa kemudian dipasangkan dengan aktor, data masukan, validasi, perubahan status, dan hasil pencatatan. Cara tersebut membantu memisahkan kebutuhan pelayanan dari rincian teknologi yang dipilih untuk mengimplementasikannya.

Identifikasi aktor menghasilkan pembagian utama antara petugas lapangan, *admin*, dan PC Lokal/HMI sebagai komponen pendukung operasi. Petugas lapangan berhubungan dengan pencatatan kegiatan melalui aplikasi Flutter. *Admin* mengelola data, memantau operasi, dan meninjau laporan melalui *Cloud* Laravel. PC Lokal/HMI menerima tindakan pada jalur normal, menjaga *timer* serta

peringatan, menyimpan data sementara, dan menyinkronkan hasil. Pembagian ini menjadi dasar *use case* serta pembatasan akses.

Arsitektur sistem kemudian disusun dengan tiga komponen yang mempunyai penyimpanan dan kewenangan berbeda. Aplikasi Flutter menangkap tindakan petugas, identitas operasi, foto, dan lokasi. PC Lokal menjadi otoritas operasi normal pada jaringan lokal dan melakukan perhitungan *Turn Round Time*. *Cloud* Laravel menyimpan data *master* dan data terpusat, menyediakan *dashboard*, laporan, autentikasi, serta komunikasi antarkomponen. Batas tersebut digunakan untuk menghindari perubahan status yang saling bertentangan.

Alur integrasi dirancang dalam dua jalur. Pada jalur normal, aplikasi Flutter mengirim tindakan melalui LAN ke PC Lokal, kemudian PC Lokal menyimpan dan menyinkronkan data ke *Cloud*. Pada jalur *fallback*, aplikasi Flutter dapat mengirim melalui 4G ke *Cloud* ketika PC Lokal tidak terjangkau. Perubahan yang masuk ke *Cloud* harus dicerminkan kembali ke PC Lokal melalui komunikasi waktu nyata atau proses *catch-up* agar status lokal dan terpusat dapat direkonsiliasi.

Setelah arsitektur dan alur umum tersedia, *use case* dipakai untuk menguji cakupan fungsi berdasarkan aktor. *Activity diagram* kemudian merinci keputusan dan alur kegagalan untuk autentikasi, profil, administrasi, pemantauan, pencatatan operasional, audio, sinkronisasi, dan rekonsiliasi. Tiga skema basis data disusun untuk *Cloud*, PC Lokal, dan Flutter agar setiap langkah mempunyai tempat penyimpanan yang sesuai serta hubungan data yang dapat ditelusuri.

Konsistensi antarartefak diperiksa melalui identitas operasi yang sama, status yang sepadan, relasi antara data utama dan foto, pembatasan lingkup pelabuhan, serta urutan perubahan ATA, kendala, dan ATD. Kontrak integrasi dipakai untuk menyelaraskan makna data pada LAN dan *Cloud* tanpa mencantumkan kredensial atau alamat privat. Hasil perancangan ini menjadi acuan pembangunan, tetapi parameter operasional dan penerimaan pengguna tetap memerlukan validasi sebelum dapat dinyatakan sebagai ketentuan resmi.

Perancangan fungsi dan data dilakukan secara berpasangan. Setiap fungsi diperiksa terhadap data yang dibaca, dibuat, diperbarui, serta dikirim ke komponen lain. Sebaliknya, setiap kelompok data diperiksa apakah mempunyai sumber,

pemilik kewenangan, hubungan dengan operasi, dan status sinkronisasi. Pemeriksaan dua arah ini digunakan untuk mengurangi fungsi tanpa tempat penyimpanan dan entitas yang tidak mendukung proses operasional.

Rancangan juga memisahkan alur berhasil dari kondisi pengecualian. Konflik kapal atau dermaga, data bukti yang belum lengkap, jaringan LAN yang tidak tersedia, kegagalan 4G, pengiriman ulang, serta perubahan status yang lebih lama ditunjukkan sebagai cabang proses. Penyimpanan lokal dan penanda status dirancang agar kegagalan komunikasi tidak langsung dianggap sebagai keberhasilan. Pada saat koneksi kembali tersedia, komponen melakukan sinkronisasi atau rekonsiliasi dengan identitas operasi yang sama.

Tahap akhir perancangan berupa peninjauan kesesuaian antara arsitektur, alur sistem, *use case*, *activity diagram*, tabel kebutuhan, dan skema basis data. Artefak-artefak tersebut tidak berdiri sendiri: perubahan aktor harus tercermin pada pembatasan akses, perubahan alur harus mempunyai aktivitas dan data, sedangkan perubahan penyimpanan harus tetap mengikuti arah kewenangan. Hasil peninjauan menjadi dasar implementasi pada bagian berikutnya dan tetap membuka ruang validasi bersama pengguna.

4.1.4. Tahapan dan Jadwal Pelaksanaan

Tahapan pelaksanaan disusun sebagai realisasi berdasarkan kegiatan yang tercatat pada *logbook*, bukan perbandingan antara jadwal rencana dan realisasi. Pengelompokan dilakukan menurut hasil utama setiap fase agar hubungan antara kebutuhan, rancangan, pengembangan komponen, dan integrasi dapat terlihat. Beberapa kegiatan pemeriksaan dan dokumentasi berlangsung bersamaan dengan pengembangan karena penyesuaian satu komponen memerlukan penyelarasan pada komponen lain.

Rentang fase menunjukkan urutan kerja dari pengamatan lapangan sampai dokumentasi pra-demonstrasi. Periode integrasi dibuat melintasi fase *Cloud*, PC Lokal, dan Flutter karena pemeriksaan kontrak serta data dilakukan secara bertahap.

Tanggal 29 Juni 2026 tidak dimasukkan sebagai periode resmi pada tabel karena berada setelah akhir masa magang 28 Juni dan masih memerlukan konfirmasi.

Tabel 4.1 Tahapan dan Jadwal Pelaksanaan Proyek Ro-Ro Alert

No.	Tahap	Periode Pelaksanaan	Kegiatan Utama	Hasil/Status
1	Observasi dan identifikasi kebutuhan	10 Maret-2 April 2026	Orientasi, pendampingan operasional, dan pencatatan kondisi pelayanan serta konektivitas.	Temuan operasional sebagai dasar kebutuhan.
2	Analisis dan perancangan	6-13 April 2026	Diskusi kebutuhan, arsitektur, alur sistem, <i>use case</i> , <i>activity diagram</i> , dan rancangan basis data.	Artefak rancangan tiga komponen.
3	Pengembangan <i>Cloud</i> Laravel	14 April-11 Mei 2026	Penyusunan data pusat, autentikasi, <i>master data</i> , <i>dashboard</i> , laporan, API, dan komunikasi waktu nyata.	Komponen pusat administrasi dan pemantauan.
4	Pengembangan PC Lokal/HMI	12 Mei-1 Juni 2026	Pencatatan lokal, <i>timer</i> TRT, kendala, audio, sinkronisasi, <i>heartbeat</i> , dan rekonsiliasi.	Komponen lokal dan penghubung <i>Cloud</i> .
5	Pengembangan aplikasi Flutter	2-26 Juni 2026	Autentikasi, penyimpanan lokal, ATA/ATD, kendala, foto/lokasi, profil, serta mode LAN dan 4G.	Aplikasi pencatatan petugas lapangan.
6	Integrasi dan pengujian bertahap	14 April-26 Juni 2026	Pemeriksaan fungsi dan kesesuaian kontrak antarkomponen serta perbaikan temuan.	Alur lintas komponen yang terus diselaraskan.
7	Dokumentasi dan evaluasi	6 April-26 Juni 2026	Pembaruan diagram, catatan kontrak, dokumentasi teknis, dan bahan demonstrasi progres.	Dokumentasi pra-demonstrasi; <i>logbook</i> masih <i>draft</i> .

Tabel 4.1 memperlihatkan bahwa pembangunan berlangsung secara bertahap dengan keluaran antara berupa temuan operasional, artefak rancangan, komponen sistem, serta dokumentasi. Status pada tabel tidak menunjukkan serah terima atau penerapan penuh. Seluruh entri *logbook* yang menjadi dasar jadwal masih berstatus *draft* sehingga periode dan hasilnya tetap perlu diperiksa bersama dokumen resmi magang.

4.2. Perancangan dan Implementasi

Bagian ini membahas analisis data dan rancangan Ro-Ro Alert sebagai dasar sebelum implementasi dijelaskan pada bagian berikutnya. Pembahasan dibatasi pada kebutuhan, arsitektur, alur, *use case*, *activity diagram*, dan perancangan basis data. Uraian antarmuka, kode, pengujian, dampak, dan penerapan sistem tidak dibahas pada tahap ini.

4.2.1. Analisis Data dan Kebutuhan

Analisis kebutuhan menggunakan temuan observasi, diskusi kerja, *logbook*, diagram, dan dokumentasi proyek. Data yang dibutuhkan dikelompokkan menjadi data *master*, data operasional, data bukti, data status, dan data audit. Pengelompokan ini membantu menentukan informasi yang harus tersedia sebelum operasi dimulai, informasi yang dihasilkan selama proses kapal, serta informasi yang diperlukan untuk meninjau kembali tindakan dan perpindahan data.

Aktor utama adalah petugas lapangan dan *admin*. Petugas lapangan memerlukan akses untuk *login*, melihat status dermaga, mencatat peristiwa operasional, melaporkan kendala, menyelesaikan kendala, dan memperbarui profil. *Admin* memerlukan akses terpisah untuk mengelola data *master*, memantau status operasi, meninjau laporan, mengatur konfigurasi, dan mengirim perintah jarak jauh. PC Lokal/HMI dimodelkan sebagai komponen pendukung yang menerima tindakan, menjaga status lokal, serta menghubungkan proses lapangan dengan *Cloud*.

Data *master* meliputi informasi pelabuhan, dermaga, kapal, akun, dan konfigurasi target. Data tersebut dikelola secara terpusat agar aplikasi Flutter dan PC Lokal memakai acuan yang sama. Salinan data *master* dapat disimpan secara lokal untuk mendukung ketersediaan layanan, tetapi perubahan resminya tetap mengikuti kewenangan *Cloud*. Nilai target *Turn Round Time* dan parameter lain tidak dinyatakan sebagai kebijakan tetap sebelum ada konfirmasi pengguna.

Data operasional berpusat pada *Actual Time of Arrival* (ATA), yaitu waktu aktual kapal sandar, dan *Actual Time of Departure* (ATD), yaitu waktu aktual kapal

berangkat. *Turn Round Time* (TRT) menggambarkan rentang proses antara ATA dan ATD. Sistem membedakan *Gross* TRT sebagai keseluruhan rentang waktu dan *Net* TRT sebagai rentang setelah durasi kendala yang tercatat diperhitungkan. Pencatatan kendala memerlukan kategori, deskripsi, waktu mulai dan selesai, durasi, status, serta hubungan dengan operasi aktif.

Setiap tindakan lapangan memerlukan bukti dan konteks yang memadai. Rancangan memasukkan foto dan lokasi untuk ATA, ATD, serta kendala, disertai identitas petugas dan waktu pencatatan. Validasi lokasi digunakan untuk memeriksa kesesuaian konteks dermaga, tetapi radius atau ambang ketelitian tidak diperlakukan sebagai ketentuan resmi. Foto dan koordinat merupakan data terbatas yang hanya boleh diproses untuk kebutuhan operasi dan audit yang berwenang.

Kebutuhan ketersediaan menghasilkan rancangan penyimpanan lokal dan dua jalur koneksi. Pada kondisi normal, aplikasi Flutter menggunakan LAN untuk berkomunikasi dengan PC Lokal. PC Lokal menyimpan *buffer* operasi dan menjaga fungsi HMI ketika internet tidak stabil. Jika PC Lokal tidak dapat dijangkau tetapi perangkat mempunyai akses internet, aplikasi menggunakan jalur *fallback* 4G ke *Cloud*. Jika kedua jalur tidak tersedia, data tidak boleh dianggap berhasil terkirim dan harus mempertahankan status yang dapat ditinjau atau dicoba kembali.

Sinkronisasi harus menjaga identitas dan urutan perubahan data. Satu operasi menggunakan identitas unik yang sama saat dikirim ulang agar pencatatan tidak berganda. Metadata dan foto dapat dikirim terpisah, tetapi keduanya harus dapat dihubungkan kembali. Rekonsiliasi diperlukan ketika data lebih dahulu masuk ke *Cloud* melalui 4G atau ketika komunikasi waktu nyata terputus. PC Lokal memerlukan mekanisme *catch-up* untuk mengambil perubahan yang terlewat dan memperbarui status lokal secara idempoten.

Cloud membutuhkan fungsi pemantauan, laporan, notifikasi, dan audit perintah jarak jauh. Namun, perintah *pause*, *resume*, atau *reset* dari *admin* tidak boleh langsung mengubah operasi normal tanpa eksekusi dan pemeriksaan oleh PC Lokal. Hasil eksekusi perlu dikirim kembali sebagai *acknowledgement* agar status perintah dapat diaudit. Keamanan mencakup autentikasi, pembatasan peran,

lingkup pelabuhan, perlindungan kredensial, dan penghindaran pencantuman data privat pada laporan.

Kebutuhan dibedakan menurut tingkat kepastian. Fungsi yang muncul pada *logbook* dan diagram dinyatakan terverifikasi, sedangkan rincian yang hanya terdapat pada dokumentasi dinyatakan terdokumentasi. Kemudahan penggunaan, angka target, radius lokasi, interval komunikasi, dan kriteria penerimaan masih memerlukan validasi pengguna. Perbedaan ini mencegah rancangan teknis dianggap sebagai kebijakan operasional yang telah disahkan.

Tabel 4.2 Kebutuhan Fungsional Sistem Ro-Ro Alert

No.	Aktor/Komponen	Kebutuhan Fungsional	Deskripsi	Status/Sumber
1	<i>Admin Cloud</i>	Autentikasi dan pembatasan akses	Memverifikasi identitas <i>admin</i> dan membatasi halaman administrasi sesuai peran dan lingkup pelabuhan.	Terverifikasi - <i>use case</i> dan dokumentasi <i>Cloud</i>
2	<i>Admin Cloud</i>	Pengelolaan data <i>master</i>	Mengelola akun, kapal, dermaga, pelabuhan, serta konfigurasi yang diperlukan komponen lain.	Terverifikasi - <i>logbook</i> dan dokumentasi <i>Cloud</i>
3	<i>Admin Cloud</i>	Pemantauan dan notifikasi	Menampilkan status operasional, status PC Lokal, aktivitas, dan pemberitahuan perubahan data.	Terdokumentasi - diagram dan GitBook
4	<i>Admin Cloud</i>	Laporan, target TRT, dan perintah jarak jauh	Menyediakan laporan KPI, pengaturan target, serta pengiriman <i>pause</i> , <i>resume</i> , atau <i>reset</i> yang tetap dieksekusi PC Lokal.	Terdokumentasi; parameter perlu validasi pengguna
5	Petugas lapangan	<i>Login</i> , profil, dan status layanan	Mengakses aplikasi sesuai identitas petugas, memperbarui profil, serta melihat status dermaga dan aktivitas hari berjalan.	Terverifikasi - diagram dan <i>logbook</i>
6	Petugas lapangan	Pencatatan ATA dan ATD	Mencatat waktu aktual sandar dan berangkat beserta kapal, dermaga, foto, lokasi, dan identitas petugas.	Terverifikasi - <i>logbook</i> dan <i>activity diagram</i>

No.	Aktor/Komponen	Kebutuhan Fungsional	Deskripsi	Status/Sumber
7	Petugas lapangan	Pelaporan dan penyelesaian kendala	Mencatat <i>pause</i> dan <i>resume</i> pada operasi aktif beserta kategori, deskripsi, foto, lokasi, dan durasi.	Terverifikasi - <i>logbook</i> dan <i>activity diagram</i>
8	PC Lokal/HMI	Operasi lokal, <i>timer</i> , audio, dan penyimpanan	Menjaga operasi normal pada LAN, menghitung TRT, mengaktifkan peringatan, dan menyimpan <i>buffer</i> sebelum sinkronisasi.	Terverifikasi - dokumentasi PC Lokal
9	Sistem sinkronisasi	Sinkronisasi dan rekonsiliasi LAN/4G	Mengirim metadata dan foto secara idempoten, menerima pembaruan, serta memulihkan peristiwa <i>fallback</i> yang terlewat.	Terverifikasi - arsitektur dan <i>activity diagram</i>

Tabel 4.2 merangkum fungsi pada tingkat kelompok agar kebutuhan tidak berubah menjadi daftar tombol atau halaman. Fungsi yang berkaitan dengan parameter operasional dan pengalaman penggunaan tetap perlu dikonfirmasi bersama pengguna sebelum dijadikan acuan penerimaan.

Tabel 4.3 Kebutuhan Nonfungsional Sistem Ro-Ro Alert

No.	Kategori	Kebutuhan	Kriteria atau Penjelasan	Status
1	Ketersediaan	Operasi lokal tetap dapat dicatat saat internet tidak stabil.	PC Lokal menyimpan data operasional untuk dikirim ketika koneksi tersedia.	Terdokumentasi
2	Konektivitas	Sistem mendukung jalur LAN dan alternatif 4G.	Pemilihan jalur dilakukan berdasarkan keterjangkauan PC Lokal dan koneksi perangkat.	Terverifikasi
3	Keandalan	Pengiriman ulang tidak boleh menggandakan operasi.	UUID yang sama dipertahankan untuk <i>retry</i> dan rekonsiliasi lintas komponen.	Terverifikasi
4	Keamanan	Akses dibatasi berdasarkan identitas, peran, dan lingkup pelabuhan.	Kredensial serta konfigurasi rahasia tidak ditampilkan dalam dokumentasi laporan.	Terdokumentasi
5	Kemudahan penggunaan	Alur pencatatan mengikuti urutan kerja petugas.	<i>Form</i> , status, dan pesan kegagalan perlu mudah dipahami di lingkungan lapangan.	Perlu validasi pengguna

No.	Kategori	Kebutuhan	Kriteria atau Penjelasan	Status
6	Kompatibilitas	Aplikasi <i>mobile</i> , <i>browser</i> HMI, dan layanan <i>Cloud</i> dapat bertukar data.	Kontrak data membedakan format LAN dan <i>Cloud</i> tanpa mengubah makna operasi.	Terdokumentasi
7	Integritas data	Status akhir dan hubungan ATA-ATD-kendala harus konsisten.	Validasi mencegah dermaga atau kapal aktif dicatat ganda dan menjaga relasi UUID.	Terverifikasi
8	Auditabilitas	Perubahan penting memiliki waktu, pelaku, status, dan riwayat.	Riwayat operasional, kendala, sinkronisasi, serta perintah jarak jauh disimpan sesuai kewenangan komponen.	Terdokumentasi
9	Pemeliharaan	Konfigurasi dan dokumentasi dapat diperbarui tanpa membocorkan rahasia.	Pemisahan tanggung jawab dan dokumentasi kontrak mendukung penyelarasan antarkomponen.	Terdokumentasi

Kebutuhan nonfungsional pada Tabel 4.3 dirumuskan tanpa angka target karena dokumen persetujuan belum tersedia. Kriteria tersebut berfungsi sebagai arah rancangan untuk menjaga ketersediaan, integritas, keamanan, dan keterlacakan, bukan sebagai hasil pengujian atau jaminan tingkat layanan.

Keterkaitan kebutuhan dengan komponen diperiksa melalui arah pengelolaan data. *Cloud* bertanggung jawab atas autentikasi, *master*, *dashboard*, laporan, dan audit terpusat. PC Lokal bertanggung jawab atas operasi normal di jaringan pelabuhan, *timer*, audio, *buffer*, serta sinkronisasi. Flutter bertanggung jawab atas interaksi petugas dan pengambilan bukti. Pembagian ini menjadi dasar untuk menentukan fungsi yang tetap tersedia ketika internet terputus dan fungsi yang menunggu hubungan dengan pusat.

Analisis juga menetapkan aturan konsistensi yang perlu dipertahankan saat implementasi. ATA harus mendahului ATD, satu operasi aktif merujuk pada kapal dan dermaga yang sah, kendala aktif harus berhubungan dengan operasi yang berjalan, serta *resume* hanya menutup kendala yang masih terbuka. Perhitungan *Gross TRT* dan *Net TRT* menggunakan waktu operasi dan akumulasi kendala yang terhubung, sehingga perubahan status tidak boleh diproses sebagai catatan terpisah tanpa identitas yang sama.

Pada sisi integrasi, kebutuhan tidak berhenti pada keberhasilan pengiriman pertama. Setiap komponen perlu membedakan data baru, pengiriman ulang, kegagalan sementara, dan penolakan final. UUID digunakan untuk menjaga idempotensi, sedangkan status sinkronisasi membantu menentukan tindakan berikutnya. Data yang telah mencapai status lebih final tidak boleh ditimpa oleh peristiwa lama yang tiba terlambat. Prinsip ini berlaku pada pengiriman dari PC Lokal maupun rekonsiliasi data *fallback* dari *Cloud*.

Analisis keamanan difokuskan pada batas yang didukung rancangan. Pengguna harus melalui autentikasi dan memperoleh fungsi berdasarkan peran serta lingkup pelabuhan. Kredensial, *token*, alamat privat, foto, dan koordinat tidak ditampilkan sebagai isi laporan. Tindakan administratif yang memengaruhi operasi memerlukan pencatatan perintah dan hasil eksekusi. Rincian teknis seperti masa berlaku *token*, retensi foto, serta prosedur pemulihan masih memerlukan ketetapan pengelola sebelum menjadi persyaratan penerapan.

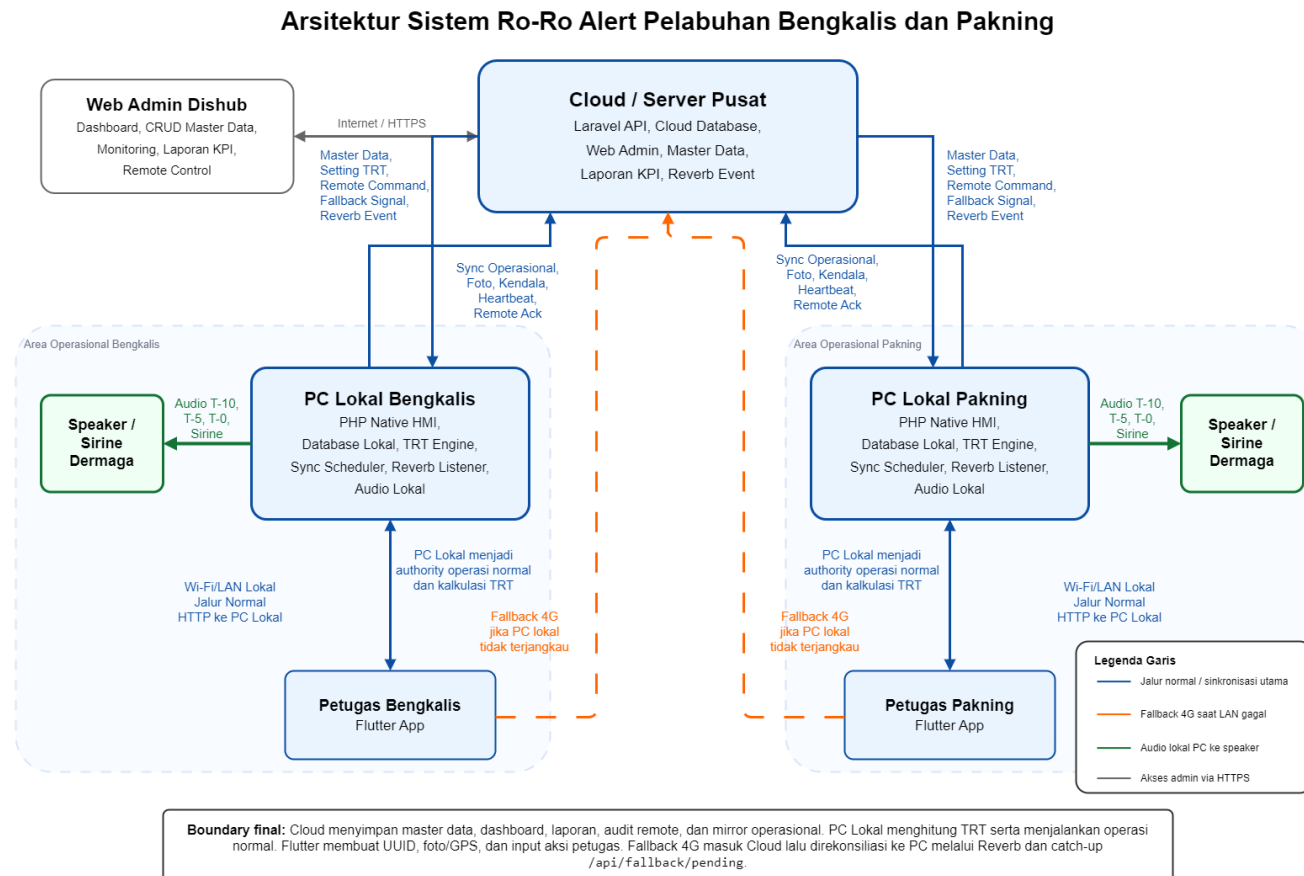
4.2.2. Rancangan Sistem/Solusi

Rancangan Ro-Ro Alert disajikan melalui artefak yang saling melengkapi. Arsitektur menjelaskan komponen dan jalur komunikasi, alur sistem menunjukkan urutan operasional, *use case* memetakan kewenangan aktor, *activity diagram* merinci keputusan pada setiap proses, sedangkan skema basis data menunjukkan tanggung jawab penyimpanan. Penjelasan berikut membahas fungsi artefak tersebut tanpa masuk ke rincian kode implementasi.

1. Arsitektur Sistem

Arsitektur Ro-Ro Alert menggunakan tiga komponen yang ditempatkan sesuai kebutuhan lapangan dan pengelolaan terpusat. Pemisahan ini memungkinkan pencatatan tetap dekat dengan sumber kegiatan, sementara data *master* dan pemantauan dapat dikelola melalui *Cloud*. Gambar 4.1 menunjukkan hubungan

aplikasi Flutter, PC Lokal/HMI, *Cloud* Laravel, *web admin*, serta perangkat audio pada dua area operasional.



Gambar 4.1 Arsitektur Sistem Ro-Ro Alert

Aplikasi Flutter berada pada sisi petugas lapangan. Komponen ini menangkap identitas petugas, pilihan kapal dan dermaga, waktu tindakan, foto, serta lokasi. Aplikasi juga membuat identitas unik operasi sebelum pengiriman sehingga percobaan ulang dapat merujuk pada pencatatan yang sama. Pada jalur normal, Flutter berkomunikasi dengan PC Lokal melalui Wi-Fi atau LAN. Pada kondisi ketika PC Lokal tidak terjangkau, aplikasi dapat menggunakan koneksi 4G menuju *Cloud*.

PC Lokal/HMI berfungsi sebagai *edge node* pada tiap pelabuhan. Komponen ini menerima tindakan operasional pada jalur normal, menyimpan *buffer* lokal, menjaga status *running* atau *paused*, menghitung target dan TRT, mengaktifkan peringatan audio, serta menyinkronkan data ke *Cloud*. Kedekatan PC Lokal dengan jaringan pelabuhan membantu proses tetap berjalan ketika internet menuju pusat tidak stabil. PC Lokal juga menjadi pelaksana perubahan status yang berasal dari perintah *admin*.

Cloud Laravel menjadi pusat pengelolaan data *master*, autentikasi, pemantauan, laporan, dan audit. Data operasional yang dikirim PC Lokal disimpan sebagai data terpusat untuk ditampilkan pada *dashboard*. *Cloud* juga menerima tindakan *fallback* dari aplikasi Flutter dan menyediakan komunikasi waktu nyata untuk menyampaikan perubahan kepada PC Lokal atau *web admin*. Peran tersebut tidak menjadikan *Cloud* sebagai pengendali langsung seluruh status pada jalur normal.

Jalur biru pada diagram menggambarkan komunikasi normal dan sinkronisasi utama. Tindakan lapangan mengalir dari Flutter ke PC Lokal, lalu metadata, foto, kendala, dan status perangkat dikirim ke *Cloud*. Jalur jingga menunjukkan *fallback* 4G ketika LAN gagal. Data yang lebih dahulu diterima *Cloud* harus direkonsiliasi ke PC Lokal melalui sinyal waktu nyata dan proses *catch-up* agar kedua penyimpanan mempunyai status yang sepadan.

Pembagian tanggung jawab juga berlaku pada penyimpanan. Flutter menyimpan sesi, data *master* tertentu, dan riwayat lokal yang diperlukan perangkat. PC Lokal menyimpan data operasi dan konfigurasi yang dibutuhkan HMI. *Cloud* menyimpan data pusat serta audit lintas perangkat. Arsitektur ini merupakan

rancangan yang didukung diagram dan dokumentasi proyek; keberhasilan operasional, kapasitas, serta penerapan penuh tetap memerlukan validasi dan pengujian pada lingkungan penggunaan.

Batas kewenangan menjadi pengikat antarkomponen. Flutter dapat memulai tindakan petugas dan membentuk UUID, tetapi penerimaan tindakan tetap melalui pemeriksaan komponen tujuan. PC Lokal menjaga keadaan operasional normal dan tidak mengubah data *master* pusat. *Cloud* menerima sinkronisasi, menyediakan tampilan lintas komponen, dan menyimpan audit, tetapi tindakan *fallback* yang melewati PC Lokal tetap harus direkonsiliasi. Batas ini mengurangi kemungkinan dua komponen menetapkan keadaan akhir yang berbeda untuk operasi yang sama.

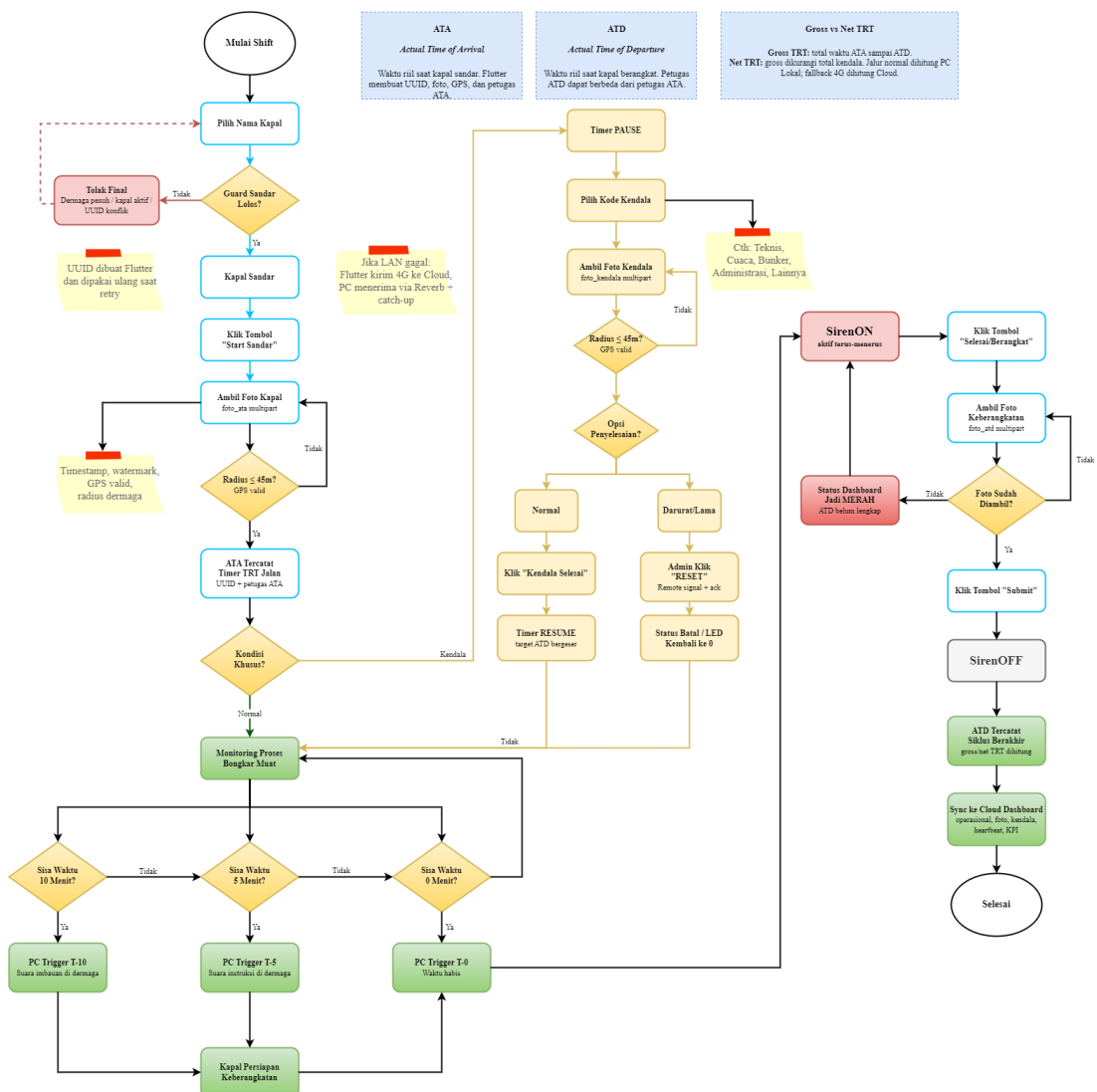
Komunikasi waktu nyata dipakai sebagai pemberitahuan perubahan, sedangkan basis data tetap menjadi sumber pemulihan apabila pemberitahuan tidak diterima. Ketika PC Lokal tersambung kembali, proses *catch-up* membaca perubahan setelah posisi terakhir yang telah diproses. Setiap peristiwa diperiksa berdasarkan UUID, lingkup pelabuhan, dan urutan status sebelum diterapkan. Dengan demikian, sambungan yang terputus tidak mengharuskan seluruh data dibuat ulang dan duplikasi dapat dikenali.

Arsitektur juga memisahkan data persisten, *cache*, dan data *runtime*. Data pusat yang diperlukan perangkat disalin sebagai *cache*, data operasi aktif disimpan dekat dengan HMI, dan informasi sesi disimpan terbatas pada aplikasi. Foto dapat dikirim terpisah dari metadata dengan hubungan yang tetap dijaga. Pemisahan tersebut mendukung pengiriman bertahap, tetapi aturan retensi, kapasitas, enkripsi penyimpanan, dan pemulihan bencana belum dinyatakan final karena belum tersedia bukti ketetapan operasional.

Kondisi perangkat juga dipantau sebagai bagian dari arsitektur. PC Lokal mengirim *heartbeat* dan status sinkronisasi agar *Cloud* dapat membedakan perangkat yang terhubung, terlambat memperbarui, atau memerlukan pemeriksaan. Informasi tersebut mendukung pemantauan, tetapi tidak menggantikan data operasi. Ketika status koneksi berubah, data persisten dan penanda sinkronisasi dipakai untuk memulihkan hubungan secara bertahap tanpa mengandalkan pesan waktu nyata yang mungkin terlewat.

2. Alur Sistem

Alur sistem menggambarkan perjalanan satu kegiatan kapal sejak petugas memulai shift sampai operasi diselesaikan dan disinkronkan. Diagram menghubungkan pencatatan waktu, kendala, peringatan, penyimpanan, dan jalur komunikasi. Gambar 4.2 menempatkan keputusan penting pada titik yang dapat menghasilkan proses normal, pengulangan *input*, penolakan konflik, atau perpindahan ke jalur *fallback*.



Gambar 4.2 Alur Sistem *Monitoring* Operasional Kapal Ro-Ro

Proses diawali dengan autentikasi petugas dan ketersediaan data *master*. Petugas memilih kapal serta dermaga yang sesuai dengan lingkup pelabuhan. Sebelum ATA dicatat, sistem memeriksa apakah dermaga masih tersedia dan kapal tidak sedang aktif pada operasi lain. Pemeriksaan ini mencegah dua operasi menggunakan sumber daya yang sama. Jika kondisi tidak sesuai, petugas menerima informasi penolakan dan dapat memperbaiki pilihan tanpa membuat identitas operasi baru yang tidak diperlukan.

Actual Time of Arrival dicatat ketika kapal benar-benar sandar. Aplikasi menyiapkan identitas unik, waktu, foto, lokasi, dan identitas petugas. Data divalidasi sebelum dikirim melalui LAN atau 4G. Setelah diterima, status operasi menjadi berjalan dan target ATD dihitung dari konfigurasi TRT yang berlaku pada rancangan. Identitas yang sama dipertahankan untuk percobaan ulang agar sistem dapat membedakan pengiriman ulang dari operasi baru.

Selama operasi berjalan, PC Lokal memantau sisa waktu menuju target. Diagram menampilkan tahapan peringatan yang dapat dikonfigurasi dan sirine ketika waktu terlampaui. Nilai seperti T-10, T-5, dan T-0 diperlakukan sebagai *slot* konfigurasi rancangan, bukan kebijakan resmi Dinas Perhubungan. Mesin audio juga memerlukan pengunci agar satu tahapan tidak diputar berulang atau dipicu oleh lebih dari satu *tab* HMI.

Apabila terjadi kendala, petugas memilih kategori, menambahkan deskripsi, serta menyiapkan foto dan lokasi. Pencatatan *pause* menghentikan perhitungan efektif dan membuka *log* kendala yang terhubung ke operasi. Setelah kondisi selesai, *resume* menutup *log* aktif, menghitung durasi jeda, dan menggeser target ATD sesuai rancangan. Jalur kegagalan mempertahankan informasi yang belum berhasil dikirim agar petugas dapat memperbaiki data atau mencoba kembali.

Actual Time of Departure dicatat ketika kapal berangkat. Aplikasi menggunakan identitas operasi ATA yang sama, merekam waktu aktual, foto, lokasi, dan petugas ATD, kemudian mengirimnya melalui jalur yang tersedia. Penyimpanan menutup kendala yang masih tertunda, menghitung *Gross* TRT, total kendala, dan *Net* TRT, serta mengubah status operasi menjadi selesai. Peringatan dan *timer* dihentikan setelah hasil pencatatan diterima.

Pada jalur LAN, PC Lokal menjadi titik pencatatan dan perhitungan normal sebelum data disinkronkan ke *Cloud*. Pada *fallback* 4G, *Cloud* menerima tindakan ketika PC Lokal terlewat, kemudian perubahan disampaikan kembali melalui komunikasi waktu nyata dan rekonsiliasi. Diagram juga memuat nilai radius serta tahapan audio sebagai konfigurasi rancangan. Nilai tersebut tidak dinyatakan sebagai ketentuan resmi karena bukti persetujuan pengguna belum tersedia.

Setiap pencatatan memakai pola pemeriksaan yang sama: aplikasi memvalidasi masukan, memilih jalur komunikasi, komponen tujuan memeriksa status terkini, lalu hasil penerimaan dikembalikan kepada petugas. Pesan berhasil hanya ditampilkan setelah ada hasil yang dapat dipertanggungjawabkan. Apabila respons tidak diterima, data mempertahankan status *draft* atau *pending* sehingga petugas dapat meninjau dan mencoba kembali tanpa membuat catatan baru yang terpisah.

Perubahan jaringan tidak mengubah makna tindakan. ATA melalui LAN dan ATA melalui 4G tetap memakai atribut inti serta UUID yang sama. Perbedaannya terletak pada komponen pertama yang menerima data dan kebutuhan rekonsiliasi setelahnya. *Cloud* meneruskan perubahan *fallback*, sedangkan PC Lokal memeriksa apakah peristiwa sudah pernah diterapkan. Status yang sama atau lebih lama diabaikan secara idempoten, sementara data baru memperbarui keadaan lokal dan posisi sinkronisasi.

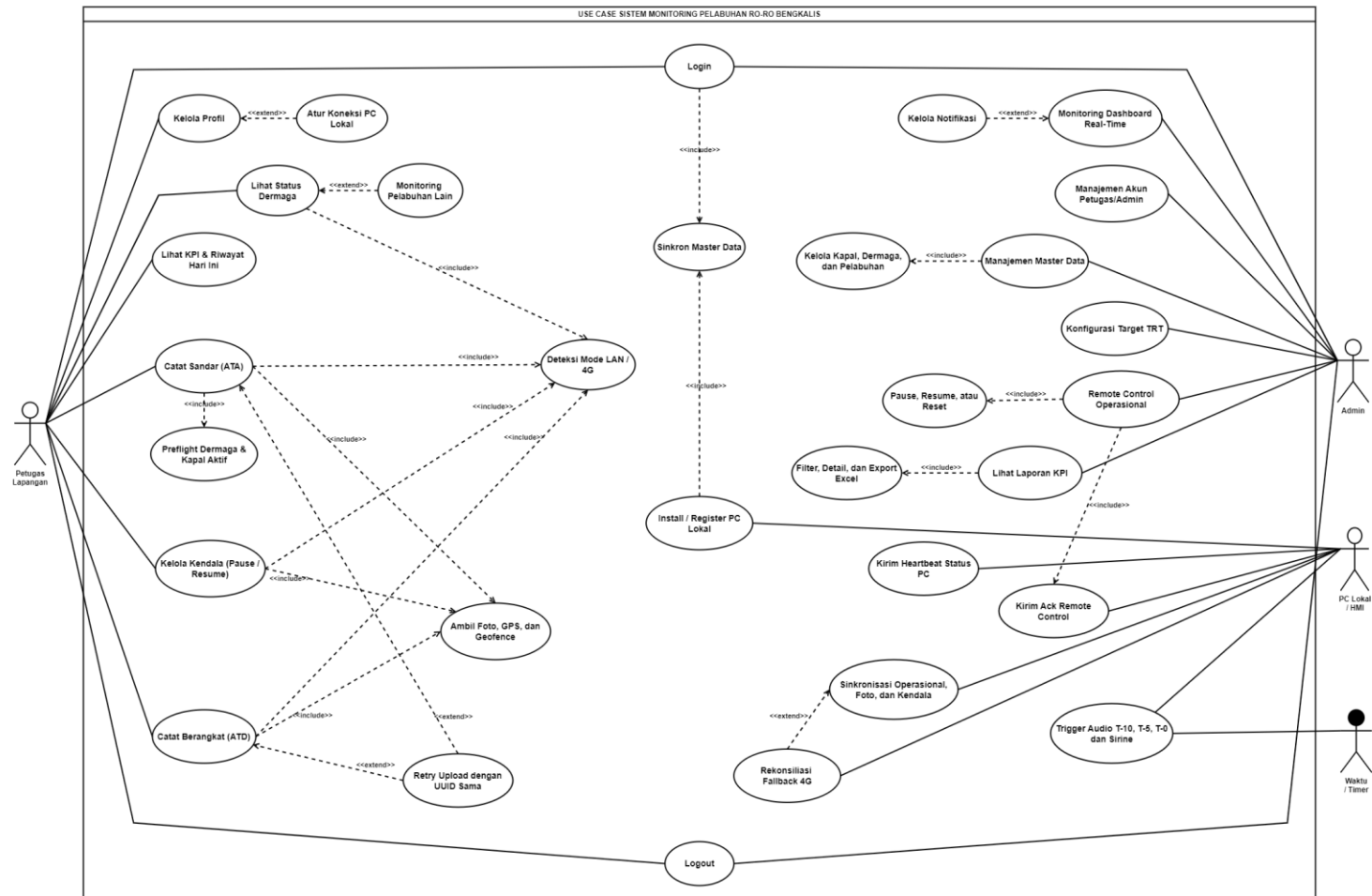
Alur *pause* dan *resume* dirancang sebagai pasangan agar durasi kendala dapat dihitung dari *log* yang sama. *Pause* hanya diterima ketika operasi masih berjalan dan tidak mempunyai kendala aktif yang bertentangan. *Resume* menutup *log* tersebut dan mengembalikan operasi ke keadaan berjalan. Jika ATD diterima ketika masih ada catatan yang perlu diselesaikan, komponen melakukan pemeriksaan penutupan sesuai rancangan sebelum hasil TRT dibentuk. Urutan ini menjaga keterlacakan perhitungan.

Setelah operasi selesai, data tidak langsung dianggap konsisten pada seluruh komponen. PC Lokal masih dapat mempunyai metadata atau foto berstatus *pending*, sedangkan *Cloud* dapat mempunyai peristiwa *fallback* yang belum diterapkan lokal. Proses sinkronisasi dan rekonsiliasi berjalan sampai status

hubungan data dapat ditinjau. Alur tersebut menjelaskan rancangan perilaku saat koneksi berubah, tetapi tidak menyatakan interval pengulangan atau batas waktu layanan tertentu.

3. *Use Case*

Use case digunakan untuk menunjukkan hubungan antara aktor dan fungsi utama Ro-Ro Alert. Diagram tidak menjelaskan urutan teknis setiap permintaan, tetapi membantu memastikan bahwa kebutuhan petugas, *admin*, dan komponen pendukung berada dalam batas kewenangan yang jelas. Gambar 4.3 memuat Petugas Lapangan, *Admin*, PC Lokal/HMI, serta waktu atau *timer* sebagai pemicu proses tertentu.



Petugas Lapangan berinteraksi dengan fungsi *login*, pengelolaan profil, pengaturan koneksi PC Lokal, pemantauan status, serta pencatatan ATA, ATD, dan kendala. Fungsi operasional melibatkan pemeriksaan ketersediaan dermaga dan kapal, pengambilan foto, lokasi, serta deteksi mode LAN atau 4G. Petugas juga dapat melihat KPI dan riwayat hari berjalan sesuai lingkup aksesnya. Hubungan tersebut menempatkan aplikasi Flutter sebagai alat bantu tindakan, bukan pengganti kewenangan petugas.

Admin berinteraksi dengan *dashboard* waktu nyata, notifikasi, akun pengguna, data *master*, konfigurasi target TRT, laporan KPI, dan *remote control*. Akses administratif dipisahkan dari petugas lapangan karena perubahan data *master* dan perintah jarak jauh mempunyai dampak lintas komponen. *Remote control* dimodelkan sebagai pengiriman *pause*, *resume*, atau *reset*, sedangkan eksekusi status tetap melibatkan PC Lokal dan menghasilkan catatan *acknowledgement*.

PC Lokal/HMI dimodelkan sebagai komponen pendukung untuk instalasi, pengiriman *heartbeat*, sinkronisasi operasional, pengiriman *acknowledgement*, rekonsiliasi *fallback*, dan pemicu audio. Aktor waktu atau *timer* digunakan untuk menunjukkan bahwa peringatan dapat berlangsung otomatis berdasarkan status dan konfigurasi, bukan hanya akibat tindakan langsung pengguna. Pemodelan tersebut membedakan fungsi manusia dari proses sistem yang berjalan berkala.

Relasi *include* menunjukkan proses yang selalu menjadi bagian fungsi lain, seperti validasi data, deteksi jalur, atau pengambilan bukti. Relasi *extend* menunjukkan perilaku yang hanya muncul pada kondisi tertentu, misalnya pemantauan pelabuhan lain atau percobaan ulang. Relasi ini membantu memeriksa cakupan, tetapi tidak digunakan untuk menyatakan bahwa seluruh skenario telah diterima atau diuji pengguna.

Pembatasan akses mengikuti identitas, peran, dan lingkup pelabuhan. Petugas tidak memperoleh fungsi administrasi, sedangkan *admin* tidak menjalankan pencatatan lapangan melalui aplikasi petugas. PC Lokal hanya menangani fungsi sistem sesuai identitas instalasinya. *Use case* menjadi dasar *activity diagram* dan tabel kebutuhan fungsional. Seluruh UML telah diverifikasi dan disetujui oleh Kepala Bidang Pelayaran.

Batas sistem pada *use case* mencakup aplikasi petugas, layanan *Cloud*, dan fungsi PC Lokal yang saling berhubungan. Aktor manusia hanya ditautkan ke tujuan yang benar-benar mereka mulai, sedangkan pekerjaan otomatis ditautkan ke PC Lokal/HMI atau *timer*. Pemisahan ini mencegah proses sinkronisasi, *heartbeat*, rekonsiliasi, dan pemicu audio disalahartikan sebagai tindakan manual yang harus selalu dijalankan oleh petugas.

Keterlacakan *use case* diperiksa terhadap tabel kebutuhan fungsional dan *activity diagram*. Fungsi autentikasi, profil, administrasi, *monitoring*, laporan, pencatatan ATA, kendala, ATD, audio, sinkronisasi, dan rekonsiliasi mempunyai uraian aktivitas yang sesuai. Fungsi yang bergantung pada konfigurasi atau lingkup akses tetap melalui pemeriksaan kewenangan. Dengan demikian, diagram menjadi peta cakupan rancangan, bukan daftar bukti bahwa setiap fungsi telah diterapkan atau diterima secara resmi.

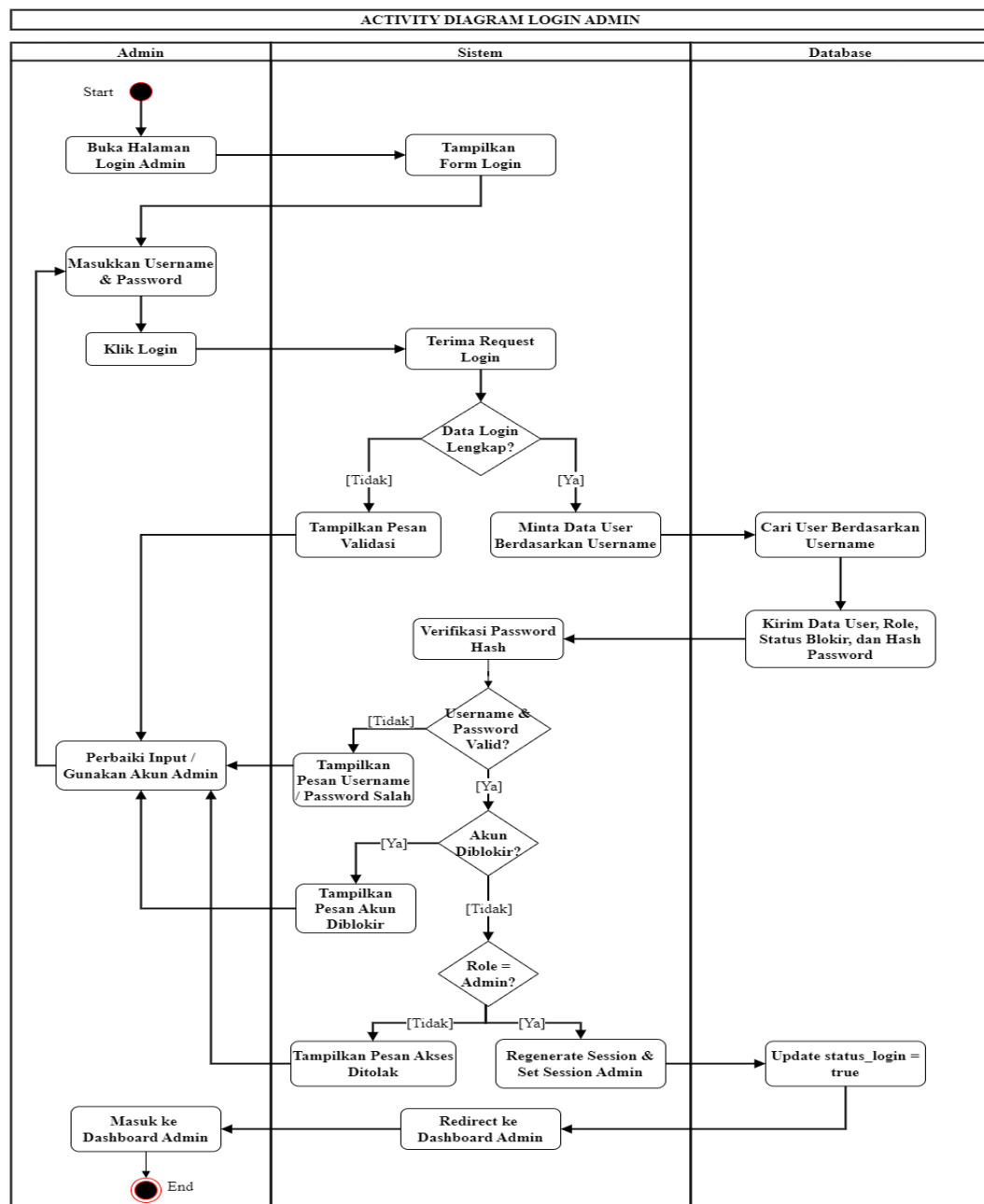
Beberapa fungsi menggunakan data yang sama dari sudut berbeda. Petugas melihat status untuk mengambil tindakan lapangan, *admin* melihatnya untuk pemantauan dan laporan, sedangkan PC Lokal menggunakannya untuk *timer* serta audio. *Use case* mempertahankan tujuan masing-masing aktor tanpa menggandakan sumber kebenaran operasi. Rincian keberhasilan, kegagalan, dan pengulangan kemudian dijelaskan pada *activity diagram* agar batas kewenangan tetap konsisten ketika proses berpindah komponen.

4. Activity Diagram

Activity diagram merinci urutan proses, keputusan, jalur keberhasilan, dan penanganan kegagalan untuk fungsi yang terdapat pada *use case*. Delapan belas diagram dikelompokkan menjadi autentikasi, pengelolaan akun, administrasi dan pemantauan, pencatatan operasional, serta integrasi PC Lokal. Setiap diagram memakai *swimlane* untuk menunjukkan perpindahan tanggung jawab antara aktor, aplikasi, layanan, dan penyimpanan.

Login admin dimulai ketika *admin* membuka halaman autentikasi dan mengisi kredensial. Sistem memeriksa kelengkapan masukan, mencari akun,

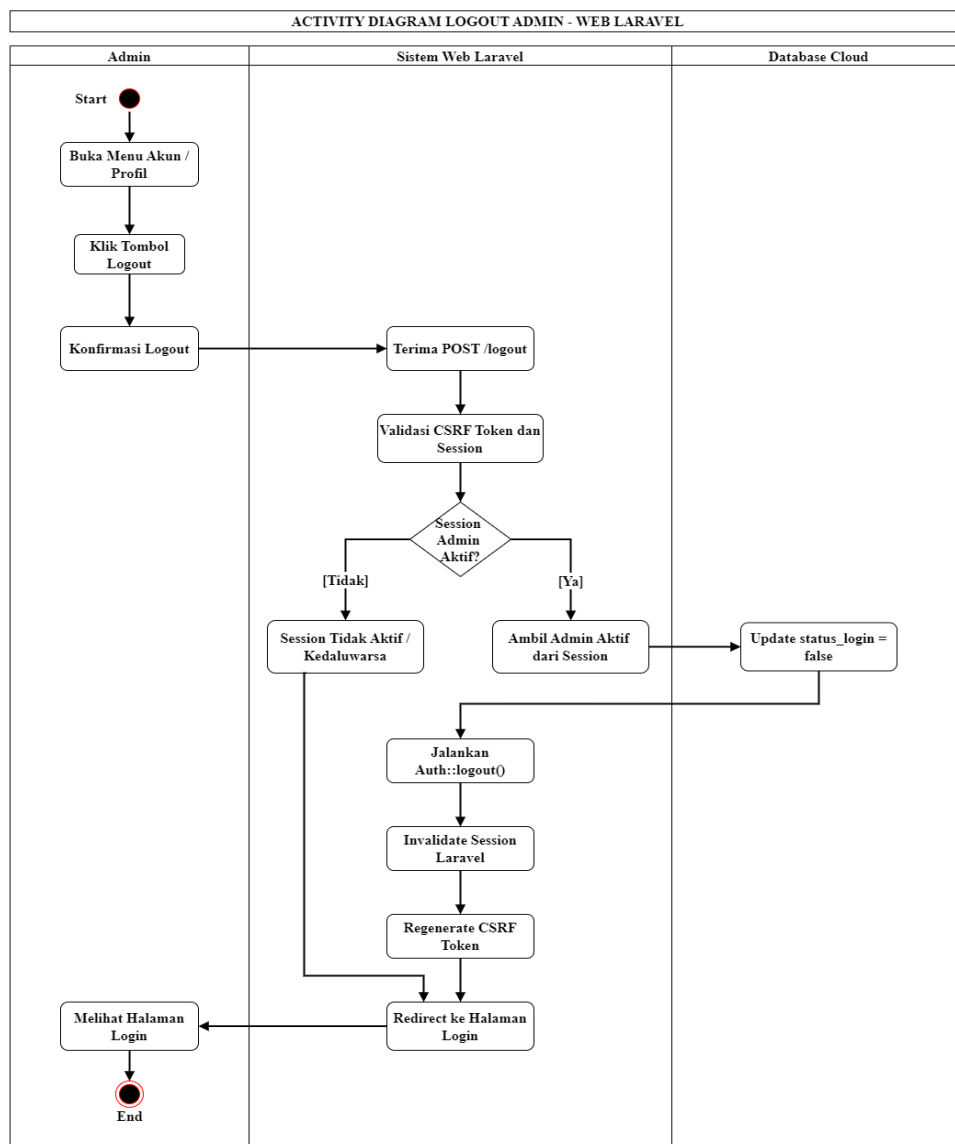
memverifikasi *hash* kata sandi, memastikan akun tidak diblokir, dan memeriksa peran *admin*. Jalur kegagalan mengembalikan pesan validasi, kredensial salah, akun diblokir, atau akses ditolak agar masukan dapat diperbaiki. Jika seluruh pemeriksaan berhasil, sistem membentuk *session* baru, memperbarui status *login*, lalu mengarahkan *admin* ke *dashboard*. Proses berakhir setelah halaman utama administrasi dapat diakses sesuai kewenangan. Catatan autentikasi mendukung penelusuran sesi tanpa membuka isi kata sandi.



Gambar 4.4 Activity Diagram Login Admin

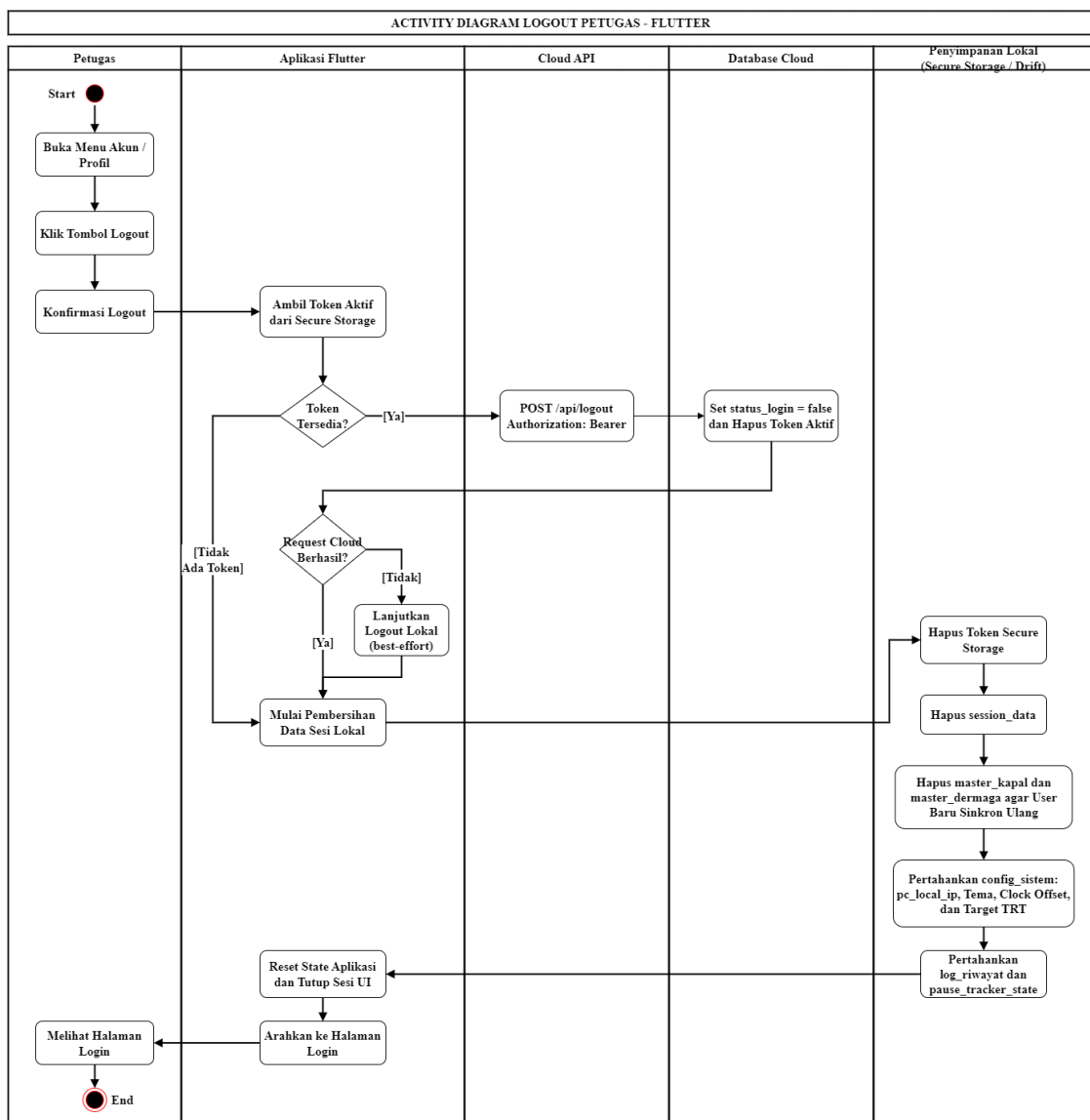
Login petugas menghubungkan aplikasi Flutter, *Cloud* Laravel, dan penyimpanan lokal. Setelah kredensial dinilai lengkap, aplikasi meminta autentikasi ke *Cloud*. Respons berhasil menghasilkan *token*, profil, peran, lingkup pelabuhan, serta waktu *server* yang disimpan secara terbatas pada perangkat. Aplikasi kemudian menyinkronkan data *master*. Jika sinkronisasi gagal tetapi *cache* tersedia, petugas dapat memakai data lokal; jika tidak tersedia, proses menawarkan percobaan ulang. Jalur kegagalan autentikasi menampilkan alasan tanpa menyimpan sesi baru. Proses selesai ketika petugas diarahkan ke halaman utama dengan data yang dapat digunakan.

Logout admin diawali dari menu akun dan memerlukan konfirmasi. Sistem memeriksa *session* serta perlindungan permintaan sebelum mengambil akun aktif. Jika *session* sudah tidak berlaku, pengguna tetap diarahkan menuju halaman *login*. Pada jalur berhasil, status *login* diperbarui, autentikasi diakhiri, *session* dibatalkan, dan *token* perlindungan dibentuk ulang. Jalur ini memastikan halaman administrasi tidak tetap menggunakan *session* lama. Proses berakhir setelah *admin* melihat halaman *login*, sedangkan kegagalan *session* tidak mengaktifkan akses baru. Konfirmasi membantu mencegah penghentian sesi akibat pilihan yang tidak disengaja pengguna.



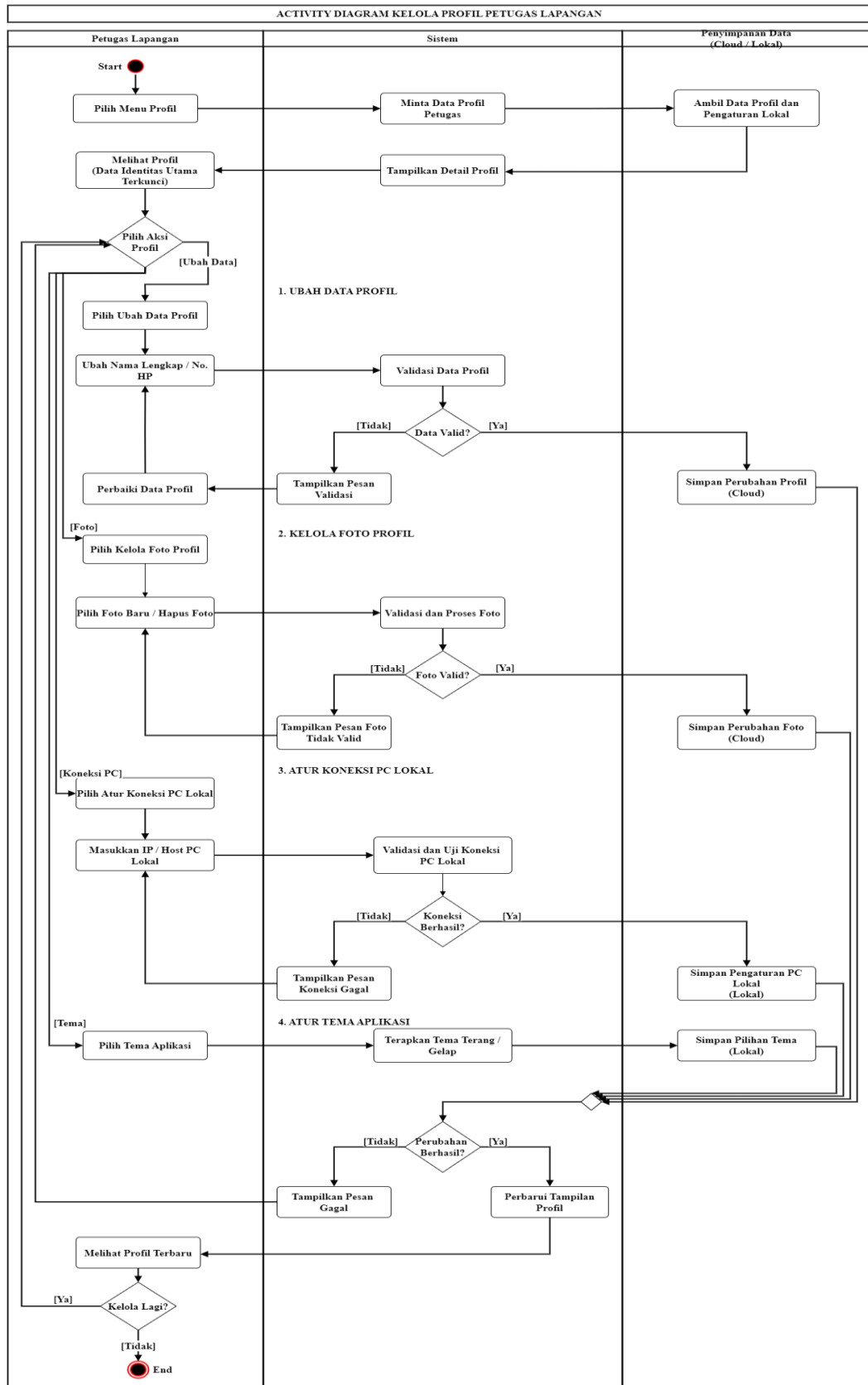
Gambar 4.6 Activity Diagram Logout Admin

Logout petugas mempertimbangkan kondisi jaringan dan data lokal. Aplikasi mengambil *token* yang tersimpan, kemudian mencoba memberi tahu *Cloud* apabila *token* dan koneksi tersedia. Kegagalan permintaan tidak menahan pembersihan lokal. Aplikasi menghapus *token* serta data sesi dan *master* yang perlu disinkronkan ulang, tetapi mempertahankan konfigurasi perangkat dan riwayat yang masih diperlukan untuk kesinambungan operasi. Tampilan dan *state* aplikasi kemudian direset. Proses berakhir pada halaman *login*, baik *logout Cloud* berhasil maupun hanya dapat diselesaikan secara lokal. Data operasi yang belum selesai tidak diperlakukan sebagai sesi autentikasi baru.



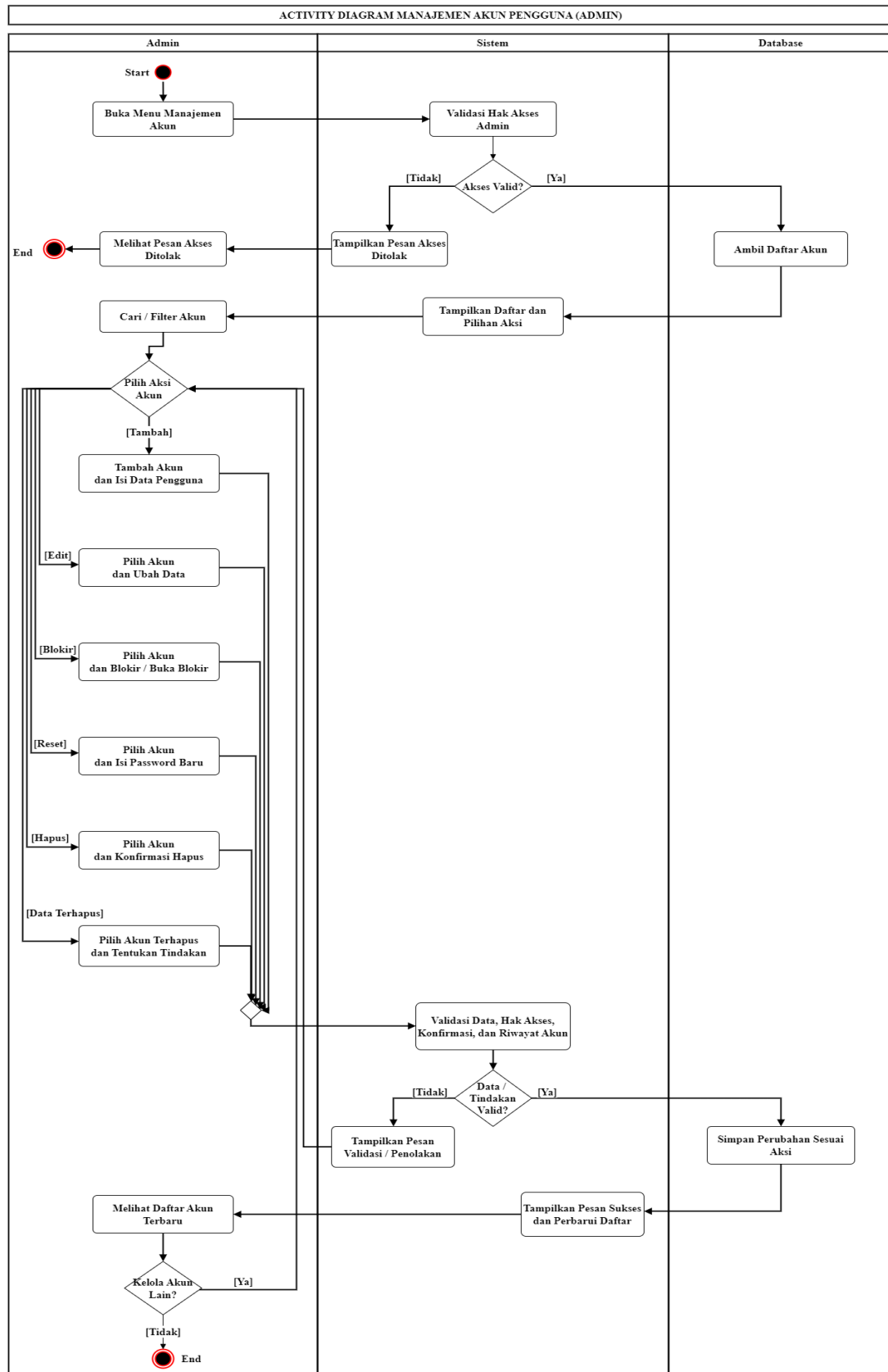
Gambar 4.7 Activity Diagram Logout Petugas

Pengelolaan profil petugas dimulai dengan menampilkan data identitas yang diizinkan. Petugas dapat mengubah nama atau nomor kontak, mengelola foto profil, mengatur koneksi PC Lokal, dan memilih tema aplikasi. Setiap perubahan memiliki validasi tersendiri dan hanya disimpan jika data atau *file* memenuhi ketentuan. Pengujian koneksi membedakan format alamat yang salah, PC yang tidak dapat dijangkau, dan koneksi yang berhasil. Jalur kegagalan mengembalikan pesan perbaikan tanpa mengganti data lama. Proses selesai ketika profil terbaru ditampilkan atau petugas memilih keluar dari pengelolaan. Perubahan tersimpan ditampilkan kembali agar petugas dapat memeriksa hasilnya.



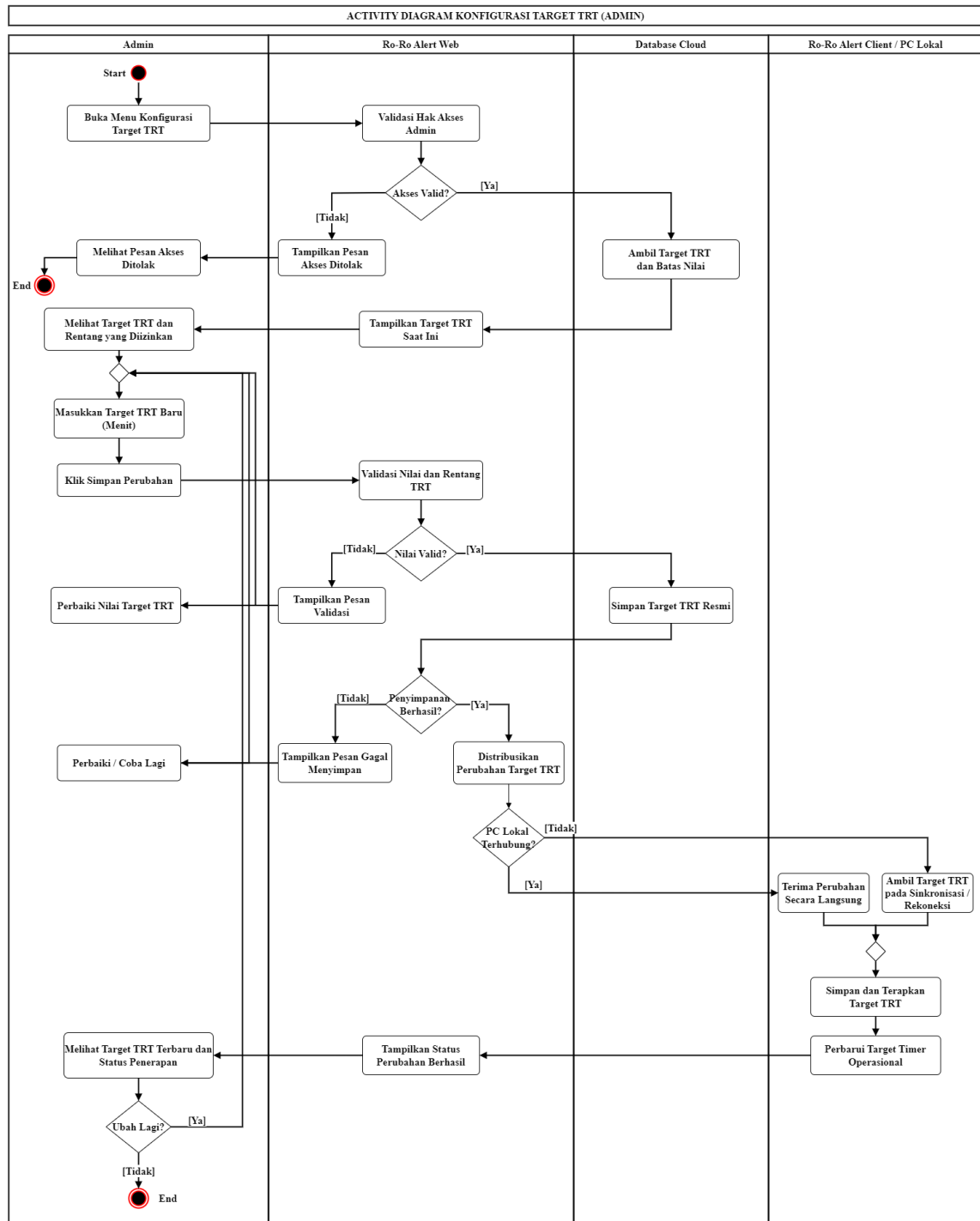
Gambar 4.8 Activity Diagram Kelola Profil Petugas Lapangan

Manajemen akun hanya dapat dibuka oleh *admin* yang mempunyai hak akses. Setelah daftar akun diambil, *admin* dapat mencari dan memilih tindakan tambah, ubah, blokir, *reset*, hapus, atau penanganan data terhapus. Sistem menggabungkan pilihan tersebut pada validasi data, kewenangan, konfirmasi, dan riwayat sebelum perubahan disimpan. Jalur tidak valid menampilkan penolakan tanpa mengubah basis data. Jika tindakan berhasil, daftar akun diperbarui dan *admin* dapat melanjutkan pengelolaan atau mengakhiri proses. Pemusnahan permanen tetap dibatasi oleh hubungan data dan kebijakan yang berlaku. Tindakan penting dipertahankan pada riwayat agar perubahan akun dapat ditelusuri.



Gambar 4.9 Activity Diagram Manajemen Akun Pengguna

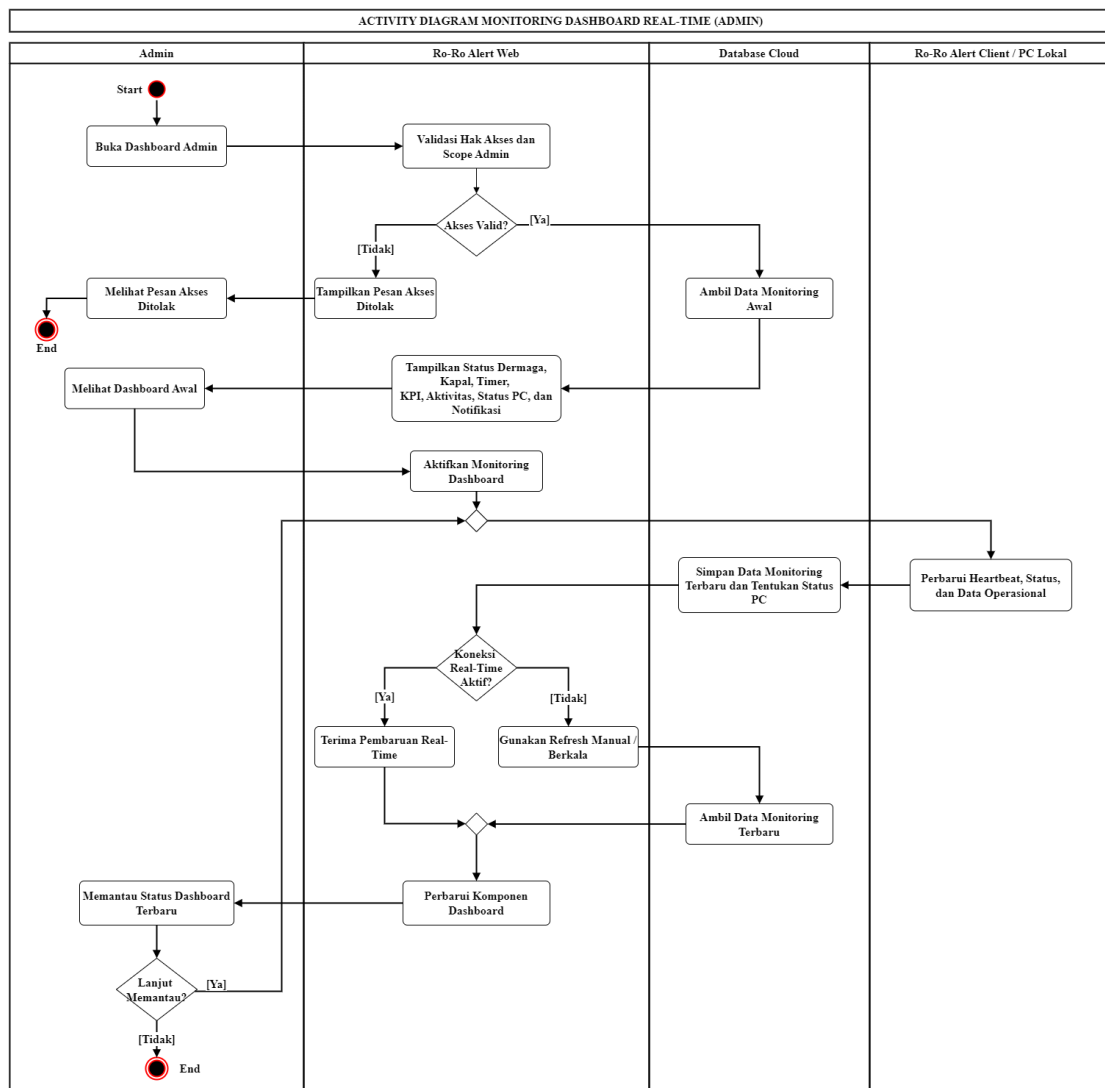
Konfigurasi target TRT dimulai ketika *admin* membuka nilai yang sedang digunakan. Sistem memeriksa hak akses, mengambil nilai serta batas konfigurasi, kemudian menampilkan formulir perubahan. Nilai baru divalidasi sebelum disimpan. Jika penyimpanan gagal, nilai lama dipertahankan dan *admin* memperoleh pesan perbaikan. Jika berhasil, perubahan didistribusikan kepada PC Lokal yang terhubung atau diambil saat rekonsiliasi berikutnya. PC Lokal menyimpan konfigurasi dan menyesuaikan target operasi sesuai aturan rancangan. Nilai angka pada diagram merupakan konfigurasi sistem yang masih memerlukan validasi sebagai kebijakan resmi. Riwayat perubahan diperlukan untuk membedakan konfigurasi lama dan konfigurasi aktif.



Gambar 4.10 Activity Diagram Konfigurasi Target TRT

Monitoring dashboard dimulai setelah *admin* melewati pemeriksaan akses dan lingkup pelabuhan. *Cloud* menyediakan status dermaga, kapal, *timer*, KPI, aktivitas, status PC, dan notifikasi. *Dashboard* kemudian mengaktifkan pemantauan perubahan. Jika komunikasi waktu nyata tersedia, pembaruan diterima

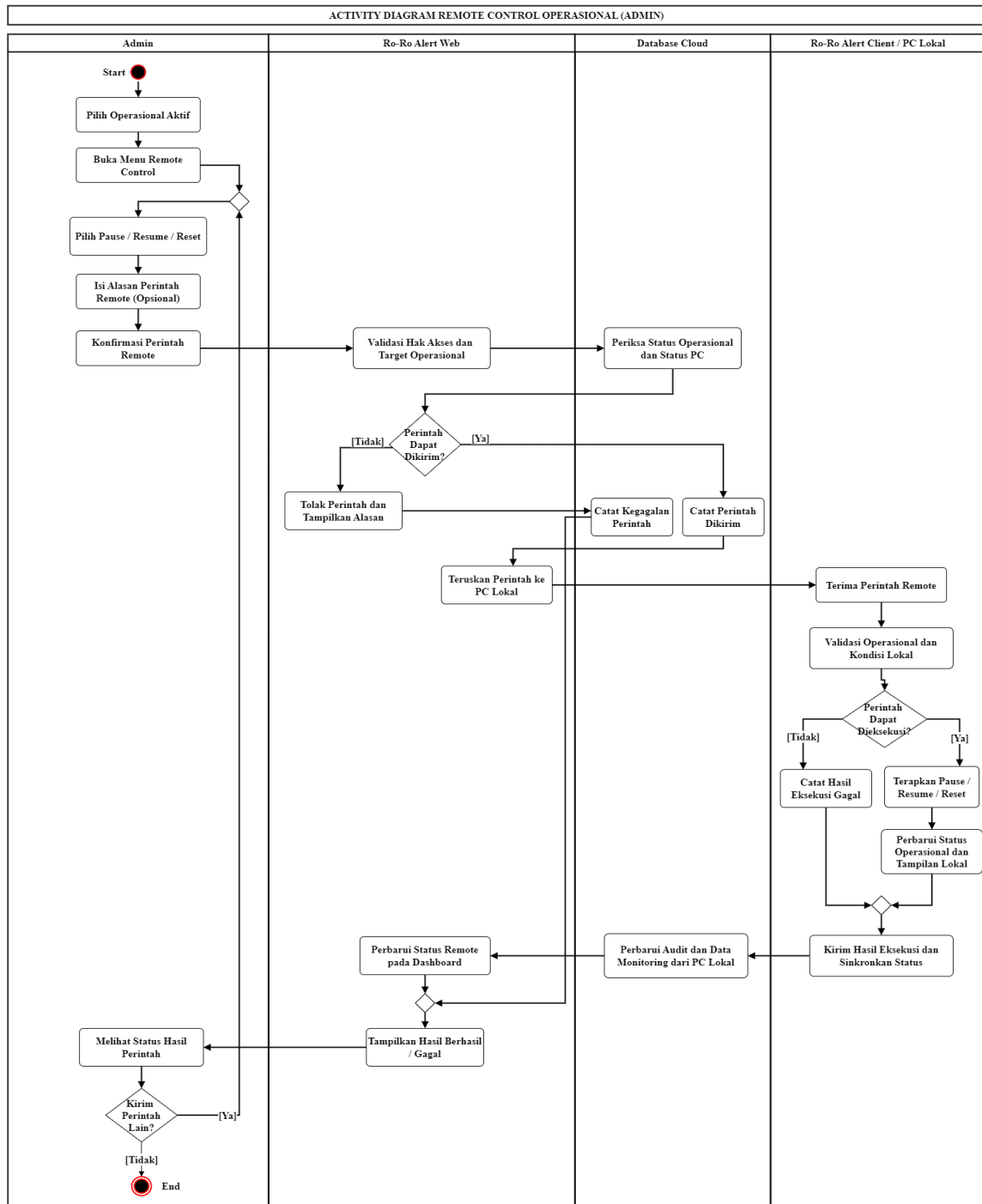
segera; jika tidak, sistem menggunakan penyegaran manual atau berkala. Kedua jalur bergabung pada pembaruan komponen tampilan. Jalur akses tidak sah berakhir dengan penolakan, sedangkan jalur normal berulang selama *admin* memilih melanjutkan pemantauan. Diagram tidak menyatakan latensi atau tingkat ketersediaan tertentu. Waktu pembaruan terakhir membantu *admin* menilai kebaruan informasi yang ditampilkan.



Gambar 4.11 Activity Diagram Monitoring Dashboard Real-Time

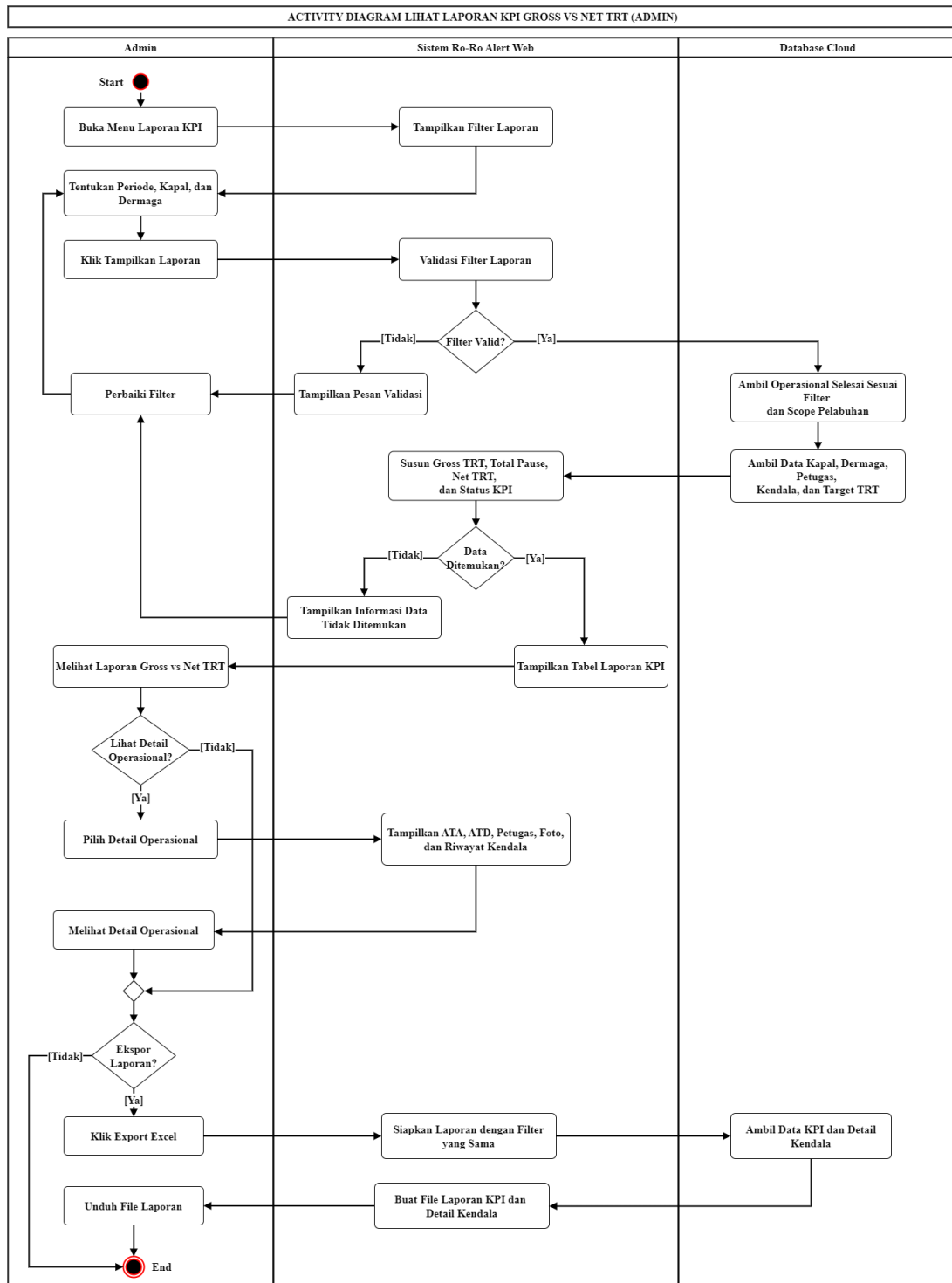
Remote control dimulai ketika *admin* memilih operasi aktif dan menentukan *pause*, *resume*, atau *reset* beserta alasan bila diperlukan. *Cloud* memvalidasi akses, lingkup, status operasi, dan kesiapan PC sebelum mencatat serta meneruskan perintah. PC Lokal memeriksa kondisi lokal dan hanya menjalankan tindakan yang

sah. Hasil eksekusi dikirim kembali sebagai *acknowledgement* untuk memperbarui audit dan *dashboard*. Jalur kegagalan dapat terjadi sebelum pengiriman atau saat eksekusi lokal; keduanya menampilkan alasan tanpa memaksakan perubahan status. Proses berakhir ketika *admin* melihat hasil dan memilih mengirim perintah lain atau berhenti.



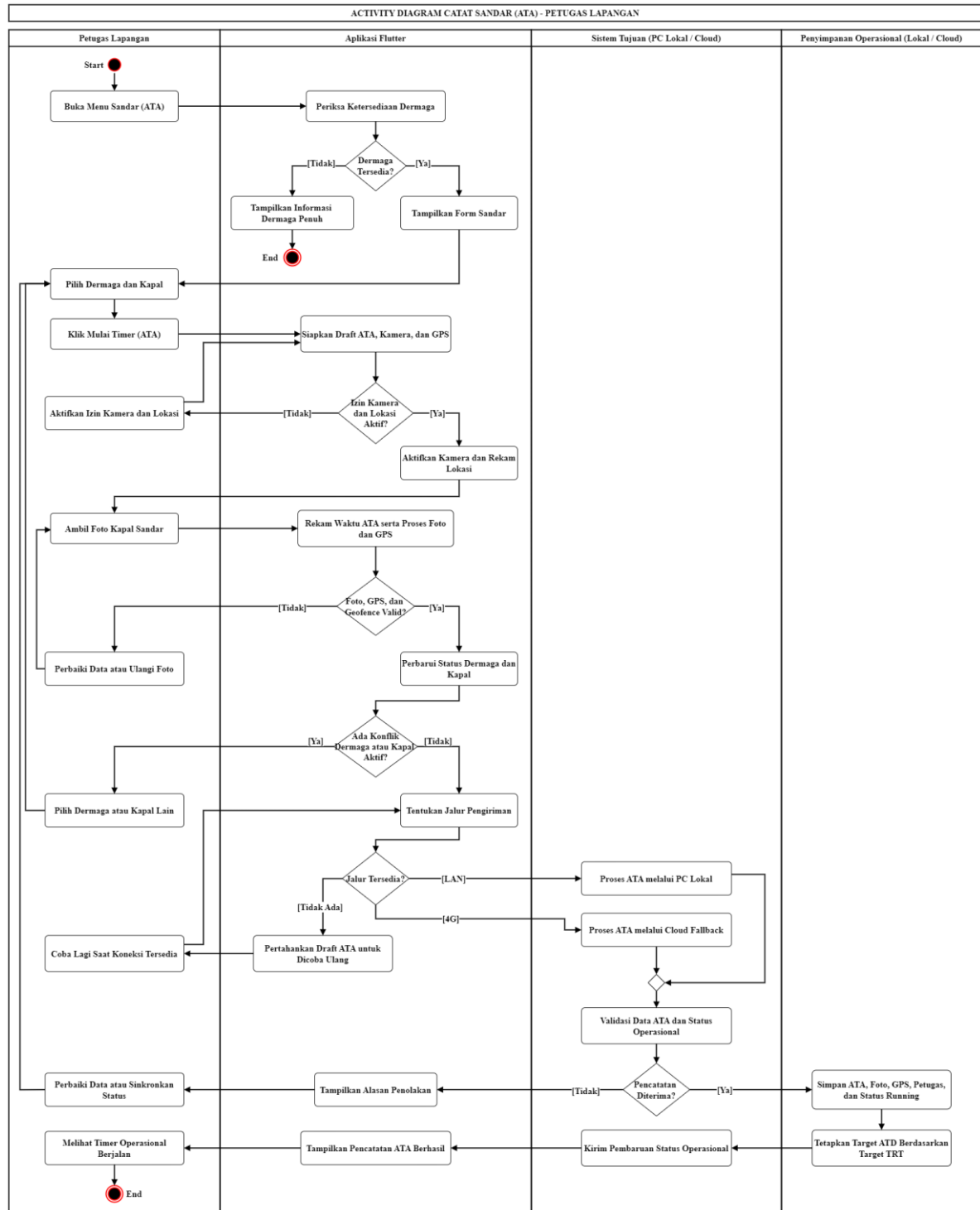
Gambar 4.12 Activity Diagram Remote Control Operasional

Laporan KPI diawali dengan pemilihan periode, kapal, dan dermaga. Sistem memvalidasi filter dan mengambil operasi yang telah selesai sesuai lingkup *admin*. Data ATA, ATD, target, kendala, *Gross TRT*, *Net TRT*, serta status KPI kemudian disusun. Jika data tidak ditemukan, sistem menampilkan informasi yang sesuai dan memberi kesempatan memperbaiki filter. Jika tersedia, *admin* dapat meninjau ringkasan dan detail operasional. Pilihan ekspor memakai filter yang sama untuk membentuk berkas laporan. Proses selesai setelah *admin* mengunduh laporan atau memilih tidak melanjutkan ekspor. Filter yang digunakan dipertahankan agar tampilan dan hasil ekspor tetap konsisten.



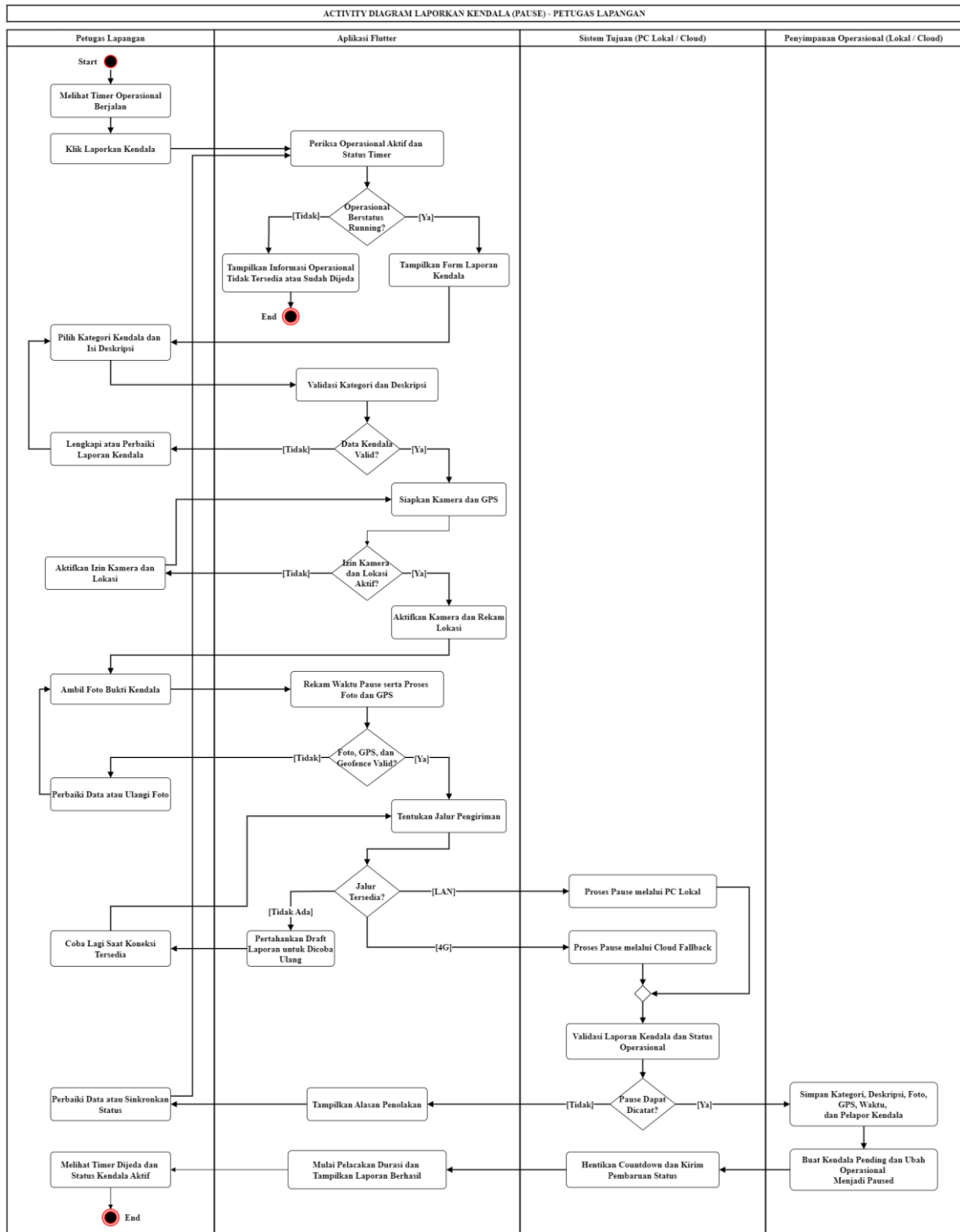
Gambar 4.13 Activity Diagram Laporan KPI Gross TRT dan Net TRT

Pencatatan sandar dimulai ketika petugas membuka menu ATA dan sistem memeriksa ketersediaan dermaga. Petugas memilih kapal dan dermaga, mengaktifkan kamera serta lokasi, lalu mengambil foto. Aplikasi memvalidasi foto, waktu, lokasi, dan kondisi kapal atau dermaga sebelum menentukan jalur LAN atau 4G. Ketika pencatatan diterima, penyimpanan menghubungkan UUID, petugas, bukti, dan status *running* serta menetapkan target ATD. Konflik atau data tidak lengkap dikembalikan sebagai penolakan untuk diperbaiki, sedangkan ketiadaan koneksi mempertahankan *draft* agar tidak dianggap terkirim. Proses berakhir ketika *timer* operasi mulai berjalan.



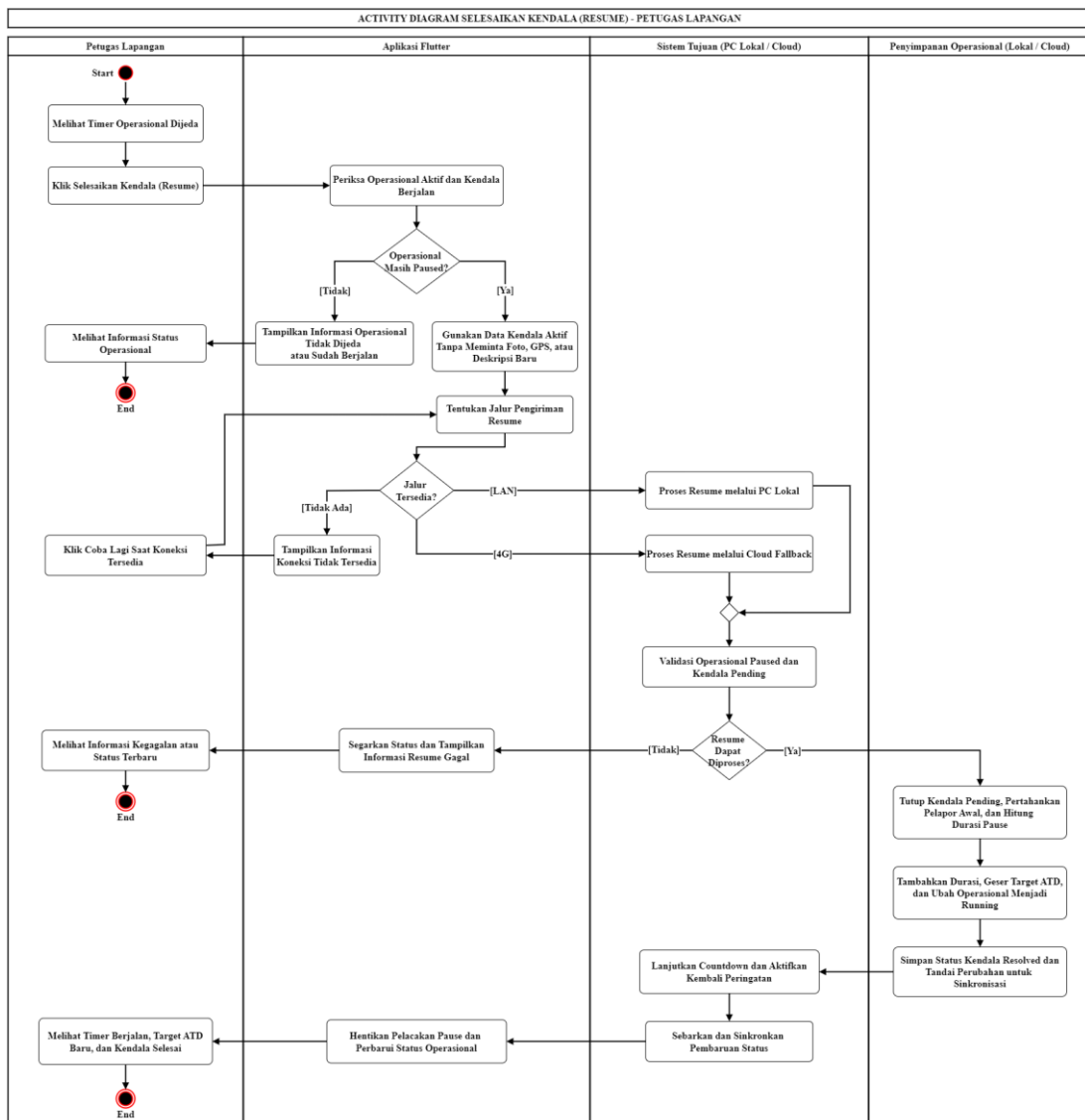
Gambar 4.14 Activity Diagram Pencatatan Sandar (ATA)

Pelaporan kendala hanya dapat dilakukan pada operasi yang sedang berjalan. Petugas memilih kategori, mengisi deskripsi, serta mengambil foto dan lokasi. Setelah data divalidasi, aplikasi memilih jalur pengiriman yang tersedia. Sistem tujuan memeriksa operasi dan memastikan tidak ada kendala *pending* yang bertentangan. Jika diterima, *log* kendala disimpan, *countdown* dihentikan, dan status operasi berubah menjadi *paused*. Jalur kegagalan mempertahankan data untuk koreksi atau percobaan ulang tanpa membuka *pause* ganda. Proses selesai ketika petugas melihat *timer* dijeda dan kendala aktif tercatat. Identitas *log* dipertahankan sebagai acuan ketika kendala diselesaikan.



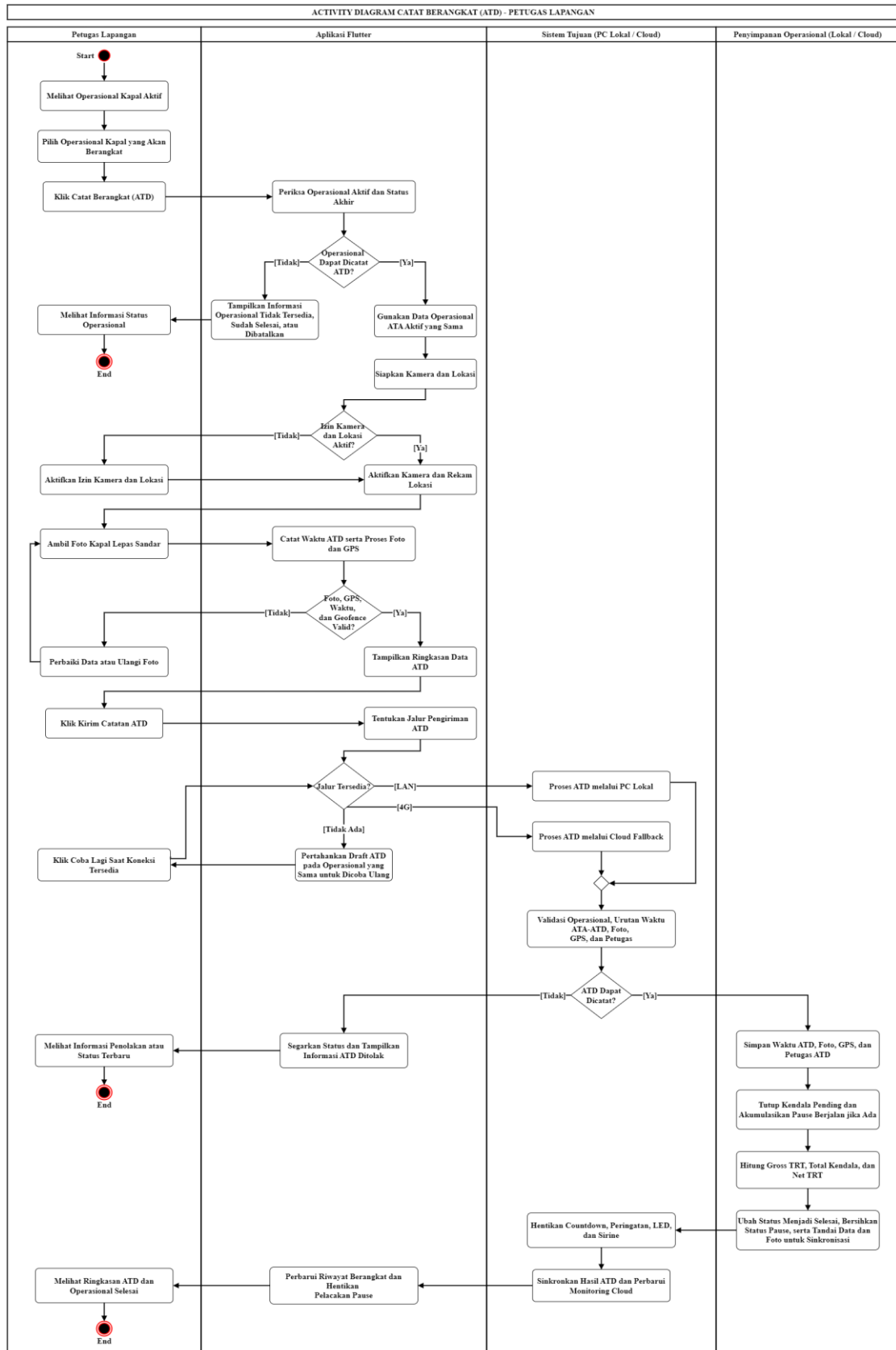
Gambar 4.15 Activity Diagram Pelaporan Kendala (Pause)

Penyelesaian kendala dimulai dari operasi yang berstatus *paused* dan memiliki *log* kendala aktif. Aplikasi menggunakan identitas kendala yang sudah ada sehingga petugas tidak diminta membuat bukti baru yang tidak diperlukan. Setelah jalur LAN atau 4G dipilih, sistem memvalidasi bahwa *resume* masih dapat diproses. Keberhasilan menutup kendala, menghitung durasi *pause*, menggeser target ATD, mengubah operasi menjadi *running*, dan mengaktifkan kembali pemantauan waktu. Kegagalan mempertahankan status sebelumnya serta menampilkan informasi terbaru. Proses berakhir ketika petugas melihat *timer* berjalan dan kendala dinyatakan selesai.



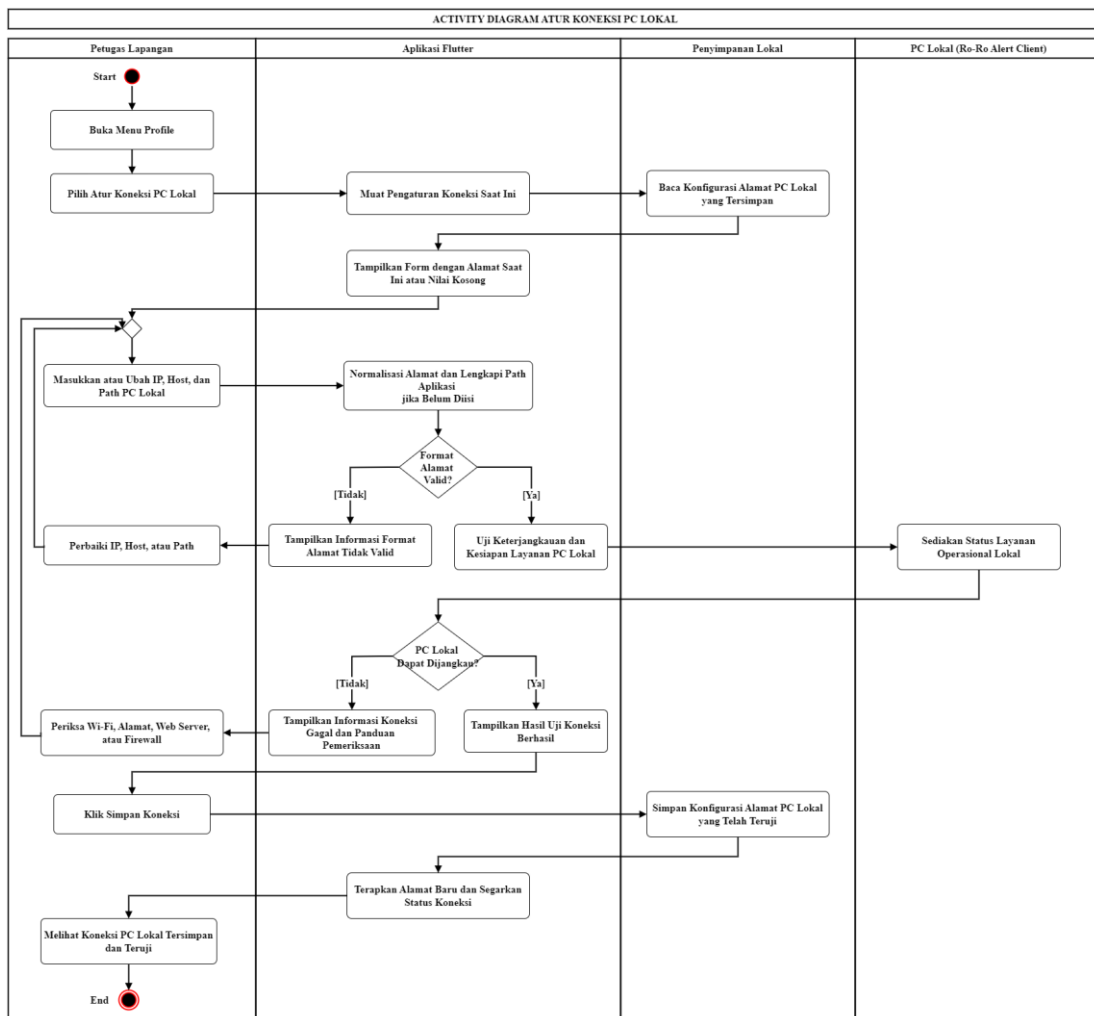
Gambar 4.16 Activity Diagram Penyelesaian Kendala (Resume)

Pencatatan berangkat menggunakan operasi ATA yang masih aktif. Petugas memilih kapal, memulai ATD, mengaktifkan kamera dan lokasi, lalu mengambil foto keberangkatan. Sistem memvalidasi hubungan operasi, bukti, petugas, dan jalur komunikasi. Setelah data diterima, kendala *pending* ditutup, durasi *pause* diakumulasi, *Gross* TRT dan *Net* TRT dihitung, serta status diubah menjadi selesai. *Timer*, peringatan, dan sirine dihentikan. Jalur kegagalan tidak menutup operasi dan memberi alasan koreksi atau penyesuaian status. Proses selesai ketika ringkasan ATD dan hasil operasi dapat ditinjau. UUID memastikan ATD terhubung dengan pencatatan ATA yang sama.



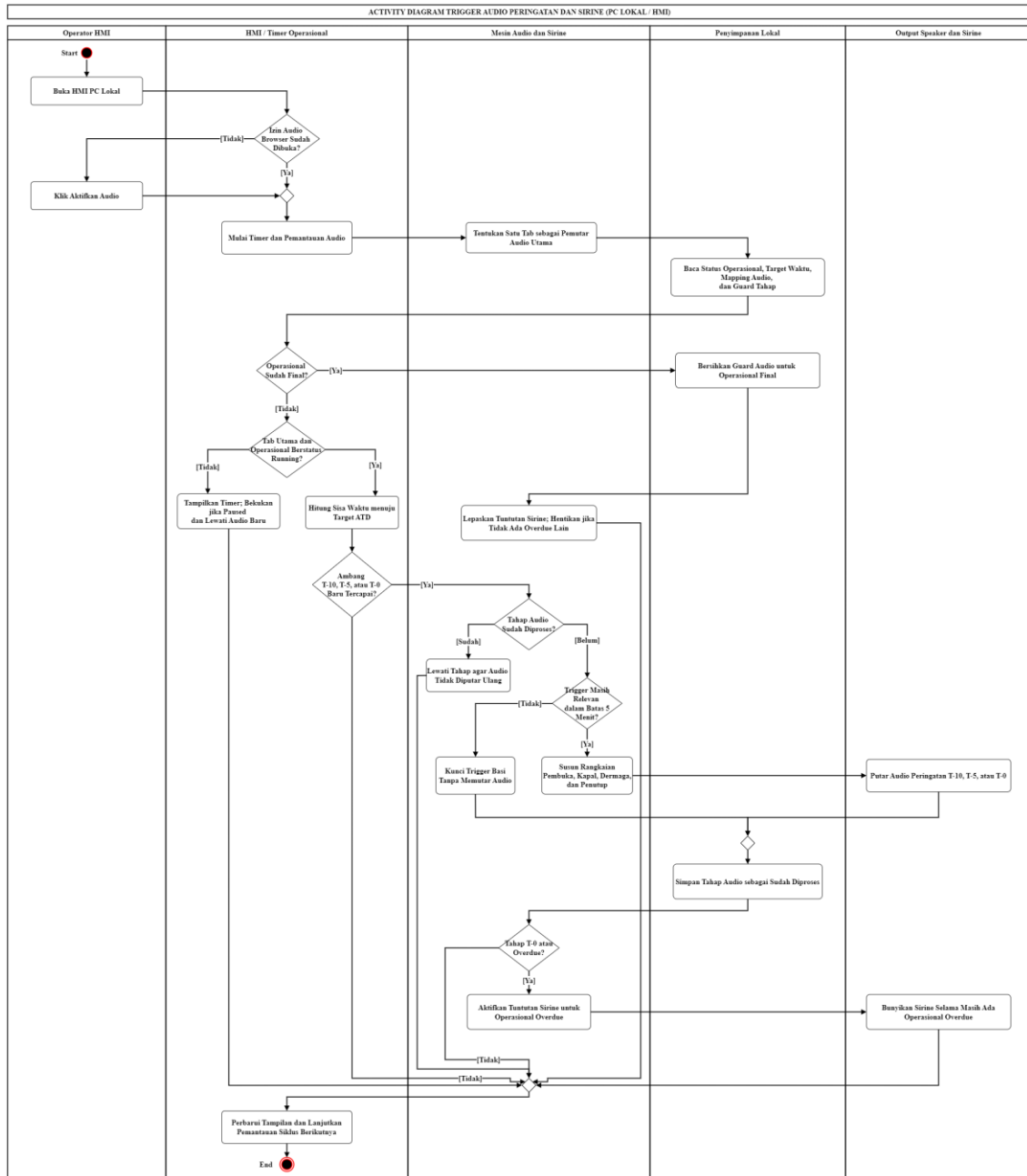
Gambar 4.17 Activity Diagram Pencatatan Berangkat (ATD)

Pengaturan koneksi PC Lokal memungkinkan aplikasi Flutter menyimpan alamat layanan LAN tanpa menampilkan konfigurasi privat pada laporan. Petugas membuka menu koneksi, memasukkan atau memperbarui alamat, lalu aplikasi menormalisasi format dan menguji keterjangkauan layanan. Format yang salah atau layanan yang tidak merespons menghasilkan petunjuk pemeriksaan jaringan, alamat, layanan *web*, atau *firewall*. Hanya hasil uji yang berhasil yang dapat disimpan sebagai koneksi teruji. Setelah nilai diterapkan, status koneksi disegarkan dan petugas melihat konfigurasi tersimpan. Proses berakhir tanpa mengganti nilai valid apabila pengujian gagal.



Gambar 4.18 Activity Diagram Pengaturan Koneksi PC Lokal

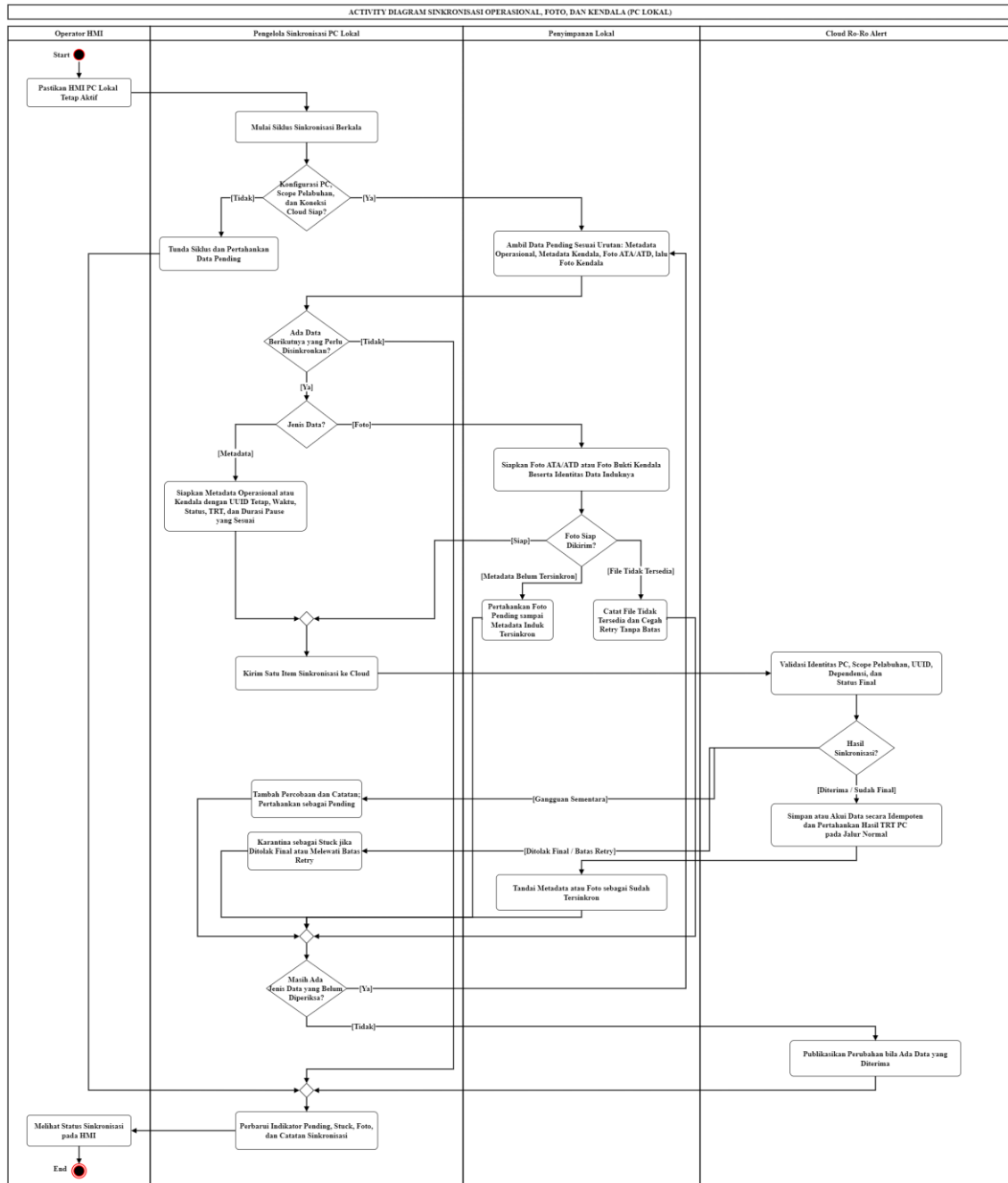
Audio peringatan berjalan pada HMI PC Lokal setelah pengguna mengaktifkan audio sesuai ketentuan *browser*. *Timer* membaca status operasi, target waktu, konfigurasi audio, dan penanda tahapan yang sudah diproses. Jika operasi *paused* atau selesai, pemutaran tidak dilanjutkan. Pada tahap yang sesuai, mesin audio memeriksa pengunci agar bunyi tidak berulang, menyusun rangkaian, lalu memutar peringatan melalui *speaker*. Ketika operasi melewati target, sirine dapat diaktifkan sampai kondisi berubah. Status pemutaran disimpan lokal dan siklus pemantauan berulang. *Slot* waktu pada diagram adalah konfigurasi rancangan, bukan kebijakan operasional resmi.



Gambar 4.19 Activity Diagram Audio Peringatan dan Sirine

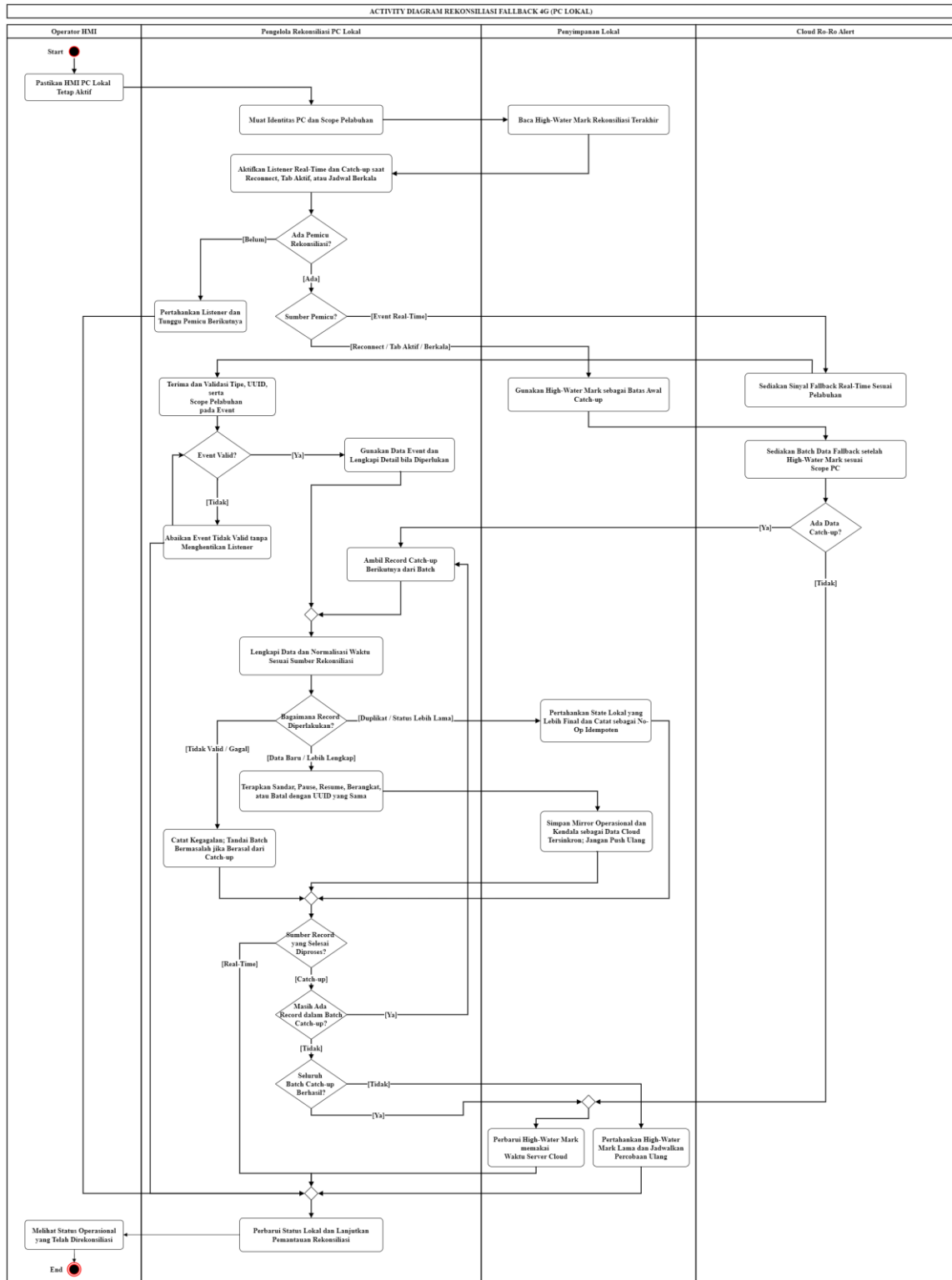
Sinkronisasi PC Lokal dimulai dengan memeriksa identitas instalasi, lingkup pelabuhan, serta koneksi. Data *pending* diproses berurutan dengan membedakan metadata dan foto. *Cloud* memvalidasi identitas, UUID, hubungan data, serta status akhir. Hasil sementara menambah percobaan dan mempertahankan data untuk dikirim ulang, sedangkan penolakan final dapat menempatkan data pada status karantina agar tidak diulang tanpa batas. Data yang diterima ditandai

tersinkron, kemudian perubahan dipublikasikan untuk pemantauan. Siklus selesai setelah indikator *pending*, foto, dan catatan diperbarui, lalu dapat berjalan kembali pada periode berikutnya.



Gambar 4.20 Activity Diagram Sinkronisasi Operasional, Foto, dan Kendala

Rekonsiliasi *fallback* menangani data yang masuk ke *Cloud* ketika PC Lokal tidak menerima tindakan langsung. Pemicu dapat berasal dari peristiwa waktu nyata, sambungan kembali, *tab* aktif, atau pemeriksaan berkala. PC memvalidasi lingkup dan UUID, lalu menggunakan *event* atau mengambil *batch* setelah *high-water mark* terakhir. Data baru diterapkan secara idempoten, sedangkan duplikat atau status yang lebih lama tidak menimpa keadaan lokal yang lebih final. Kegagalan dicatat agar *batch* dapat diulang. Setelah seluruh data berhasil diproses, *high-water mark* diperbarui dan status lokal disegarkan. Proses ini menutup kemungkinan peristiwa terlewat tanpa menganggap kanal waktu nyata sebagai penyimpanan permanen.



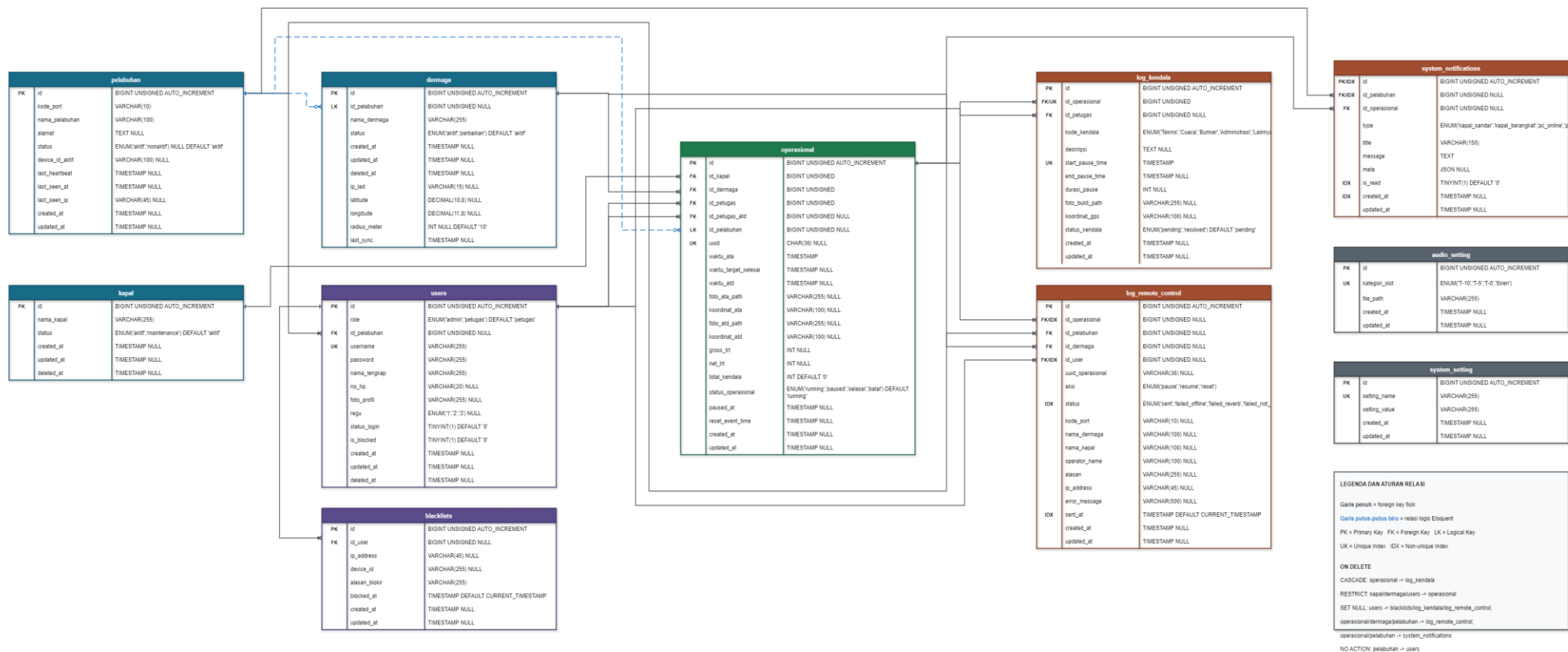
Gambar 4.21 Activity Diagram Rekonsiliasi Fallback 4G

5. Perancangan Basis Data

Ro-Ro Alert menggunakan tiga penyimpanan karena setiap komponen mempunyai kebutuhan ketersediaan dan kewenangan berbeda. *Cloud* Laravel menyimpan data terpusat dan audit, PC Lokal menyimpan operasi yang harus tetap berjalan di lingkungan pelabuhan, sedangkan aplikasi Flutter menyimpan sesi, *cache*, dan riwayat yang diperlukan perangkat. Pemisahan tersebut tidak berarti data berdiri sendiri; identitas operasi, data *master*, status, dan penanda sinkronisasi digunakan untuk menjaga hubungan antarpilar.

Basis data *Cloud* Laravel berfungsi sebagai penyimpanan pusat bagi data pelabuhan, dermaga, kapal, pengguna, operasi, kendala, notifikasi, konfigurasi, serta audit perintah jarak jauh. Hubungan data memungkinkan *dashboard* dan laporan menelusuri satu operasi ke kapal, dermaga, pelabuhan, petugas ATA, petugas ATD, dan catatan kendala. Gambar 4.22 menunjukkan entitas inti dan relasinya tanpa memuat data pengguna nyata.

Skema Database Ro-Ro Alert - Core Application Tables



Gambar 4.22 Skema Basis Data *Cloud* Laravel

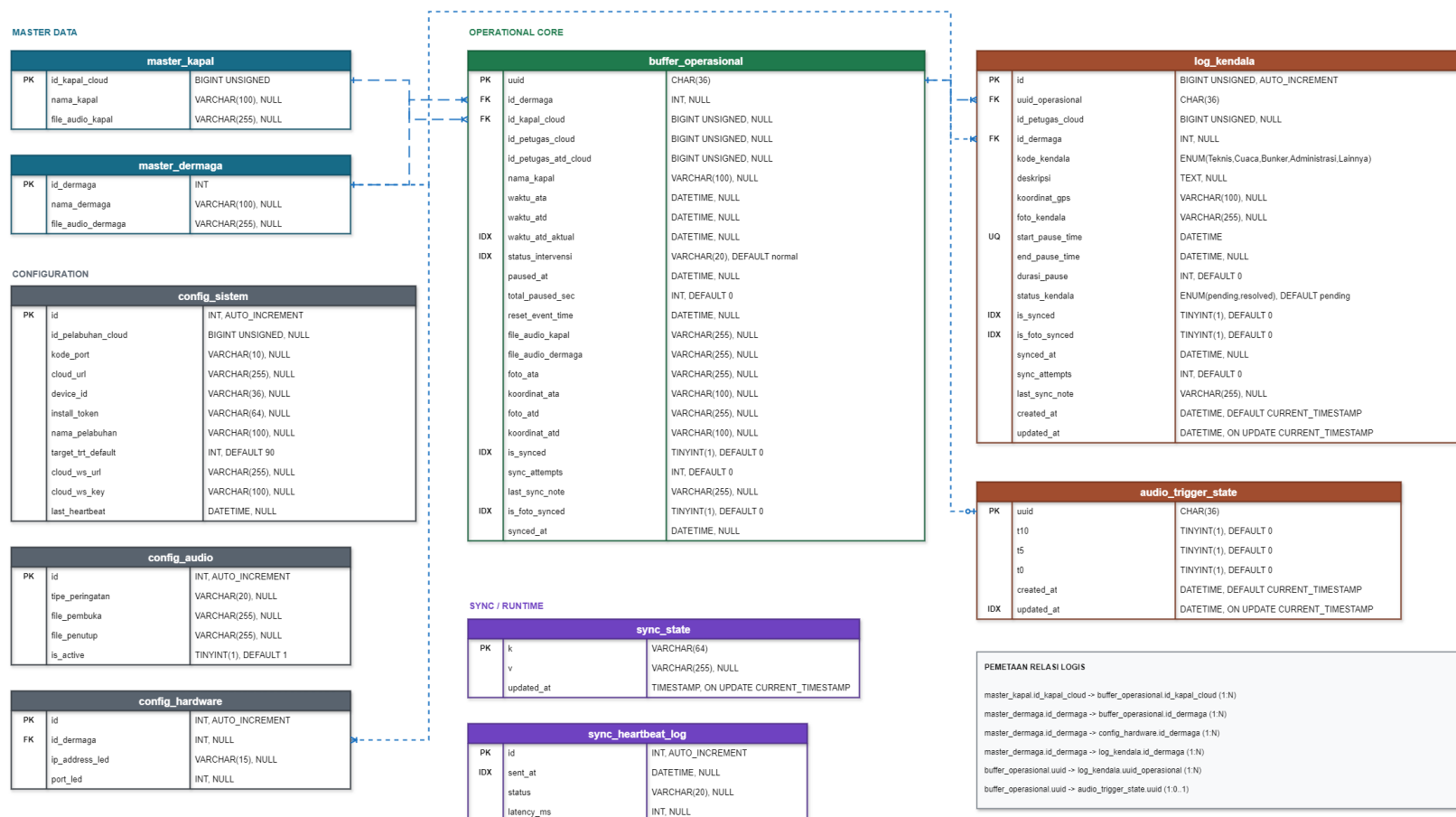
Entitas operasional menjadi pusat hubungan basis data *Cloud*. Satu catatan menyimpan identitas unik, waktu ATA, target, ATD, jalur bukti foto dan lokasi, hasil *Gross TRT* dan *Net TRT*, total kendala, serta status. *Log* kendala berhubungan dengan operasi dan petugas pelapor, sedangkan audit *remote control* menyimpan perintah, status pengiriman, dan hasil *acknowledgement*. Struktur tersebut memisahkan perubahan status operasi dari catatan perintah *admin* sehingga keduanya dapat ditinjau secara berbeda.

Data *master* dan konfigurasi dikelola pada entitas terpisah agar dapat digunakan oleh lebih dari satu operasi. Pelabuhan menjadi batas lingkup, dermaga dan kapal menjadi sumber daya operasional, serta akun menentukan peran pengguna. Notifikasi dan pengaturan sistem mendukung pemantauan dan distribusi konfigurasi. Kata sandi hanya dipahami sebagai nilai yang dilindungi pada sistem; laporan tidak menampilkan isi kredensial, *token*, alamat *server*, atau data pribadi.

PC Lokal menggunakan basis data MySQL atau MariaDB untuk mendukung operasi normal ketika akses internet tidak stabil. Data *master* disalin secukupnya dari *Cloud*, sedangkan *buffer* operasional, *log* kendala, konfigurasi audio, status sinkronisasi, dan *heartbeat* disimpan lokal. Gambar 4.23 memperlihatkan pemisahan antara kelompok data *master*, konfigurasi, inti operasional, dan *runtime* sinkronisasi.

Skema Database Ro-Ro Alert - PC Lokal (roro-alert-client)

Notasi: PK = Primary Key, FK = referensi logis, IDX = indeks, UQ = bagian unique key. Seluruh relasi digambar biru putus-putus karena database lokal belum memasang constraint FOREIGN KEY fisik.



Gambar 4.23 Skema Basis Data PC Lokal

Buffer operasional memakai UUID sebagai kunci hubungan dengan data *Cloud*. Catatan ini menyimpan waktu, status, petugas, jalur foto, akumulasi jeda, dan penanda sinkronisasi. *Log* kendala merujuk pada UUID operasi agar *pause* dan *resume* tidak terpisah dari kapal yang sedang diproses. Penanda sinkronisasi membedakan data yang belum dikirim, telah diterima, atau memerlukan penanganan karena penolakan final. Pemisahan metadata dan foto memungkinkan keduanya dikirim secara bertahap dengan hubungan yang tetap terjaga.

Konfigurasi lokal menyimpan identitas instalasi dan pengaturan yang diperlukan HMI, tetapi nilainya tidak dicantumkan dalam laporan. Tabel status audio mencegah tahapan peringatan diputar berulang. *Sync state* menyimpan posisi rekonsiliasi terakhir, sedangkan *heartbeat log* membantu meninjau komunikasi perangkat. Relasi pada basis data lokal sebagian bersifat logis karena konsistensi juga dijaga oleh aplikasi dan kontrak integrasi, sehingga pemeriksaan data tetap diperlukan saat sinkronisasi.

Aplikasi Flutter menggunakan Drift/SQLite untuk menyimpan sesi terpilih, *cache* kapal dan dermaga, konfigurasi, *log* riwayat, serta status *pause*. Penyimpanan ini mendukung tampilan dan pemulihan *state* ketika jalur LAN atau 4G berubah. Gambar 4.24 menunjukkan data yang diperlukan perangkat tanpa menempatkan aplikasi sebagai sumber *master* data pusat.

Skema Database Ro-Ro Alert - Flutter (roro-alert-flutter)

Notasi: PK = Primary Key dan REF = referensi logis/denormalisasi. Seluruh relasi digambar biru putus-putus karena skema Drift/SQLite belum memasang constraint FOREIGN KEY fisik.

AUTENTIKASI / SESI

session_data		
PK	id_user	INTEGER
	token_sanctum	TEXT
	username	TEXT
	nama_lengkap	TEXT
	role	TEXT, DEFAULT 'petugas'
REF	id_pelabuhan	INTEGER
	regu	TEXT
	pelabuhan	TEXT
	foto_profil	TEXT, NULL
	no_hp	TEXT, NULL
	is_blocked	INTEGER, DEFAULT 0
	remember_me	INTEGER, DEFAULT 0
	login_time	INTEGER, epoch seconds

KONFIGURASI

config_sistem		
PK	config_key	TEXT
	config_value	TEXT

Seeded config_sistem keys on fresh install:
target_trt_menit=45, min_trt_menit=12, max_trt_menit=480, pc_local_ip="", app_theme=dark,
clock_offset_seconds=0, last_app_version="", remember_me=false.

MASTER DATA

master_kapal		
PK	id_kapal_cloud	INTEGER
	nama_kapal	TEXT
	updated_at	DATETIME

master_dermaga		
PK	id_dermaga_cloud	INTEGER
	nama_dermaga	TEXT
REF	id_pelabuhan	INTEGER
	latitude	REAL
	longitude	REAL
	radius_meter	REAL, DEFAULT 10.0
	updated_at	DATETIME

CATATAN IMPLEMENTASI

Local storage outside SQLite:
FlutterSecureStorage is used for sanctum_token and remember_me. It is intentionally not modeled as a Drift table, but session_data also stores token_sanctum for the active local session.

Database file: roro_alert.db
Drift schemaVersion: 3
Migration v2 added session profile fields. Migration v3 added log_rivayat.waktu_atd for cross-day ATA/ATD KPI.

OPERASIONAL LOKAL

log_rivayat		
PK	uuid	TEXT
	jenis_aksi	TEXT, Sandar/Berangkat/Kendala
	upload_status	TEXT, DEFAULT 'draft'
	waktu_eksekusi	DATETIME
	waktu_atd	DATETIME, NULL
REF	nama_kapal	TEXT, NULL, denormalized
REF	nama_dermaga	TEXT, NULL, denormalized
	jalur_kirim	TEXT, NULL, LAN/4G
	gps_unavailable	BOOLEAN, DEFAULT false
	foto_temp_path	TEXT, NULL

STATUS RUNTIME

pause_tracker_state		
PK	uuid	TEXT, REF log_rivayat.uuid
	total_paused_seconds	INTEGER, DEFAULT 0
	pause_start_time	DATETIME, NULL
	is_paused	BOOLEAN, DEFAULT false

PEMETAAN RELASI LOGIS

session_data.id_pelabuhan -> master_dermaga.id_pelabuhan (scope 1:N)
master_kapal.nama_kapal -> log_rivayat.nama_kapal (denormalisasi 1:N)
master_dermaga.nama_dermaga -> log_rivayat.nama_dermaga (denormalisasi 1:N)
log_rivayat.uuid -> pause_tracker_state.uuid (1:0..1)

log_rivayat.upload_status FSM:
draft -> processing -> uploading -> Terkirim_PC / Terkirim_Cloud / Gagal / Ditolak_409 / Ditolak_422 / Ditolak_GPS_Palsu / Ditolak_Konflik: Kendala child rows use uuid prefix: [ataUuid]-kendala-[micros].

Gambar 4.24 Skema Basis Data Flutter

Log riwayat memakai UUID untuk menghubungkan tindakan sandar, berangkat, atau kendala dengan operasi yang sama. Status pengiriman menunjukkan apakah data masih berupa *draft*, sedang diproses, telah dikirim melalui PC atau *Cloud*, gagal, atau ditolak. *Pause tracker* mempertahankan informasi operasi yang sedang dijeda agar aplikasi dapat menampilkan durasi dan melanjutkan *state* dengan benar. Data *master* lokal bersifat *cache* dan diperbarui dari sumber pusat ketika koneksi tersedia.

Sinkronisasi antarpemilihan mengikuti arah kewenangan data. *Cloud* mendistribusikan *master* dan konfigurasi, PC Lokal mengirim hasil operasi normal, sedangkan Flutter membuat identitas dan bukti tindakan petugas. Pada *fallback*, *Cloud* menerima operasi lebih dahulu dan PC Lokal melakukan rekonsiliasi. UUID, status akhir, waktu, lingkup pelabuhan, dan penanda sinkronisasi digunakan untuk mencegah duplikasi atau penimpaan data yang lebih final. Skema ini merupakan rancangan pengembangan dan tidak dinyatakan sebagai basis data produksi yang telah diterapkan penuh.

Relasi antarpemilihan dipertahankan melalui identitas yang sama, bukan melalui hubungan basis data langsung. Ketika satu operasi dibuat pada Flutter, UUID dibawa ke PC Lokal atau *Cloud* bersama metadata tindakan. Foto dapat menyusul dengan referensi UUID dan jenis bukti. Kendala memakai identitas operasi serta identitas lognya sendiri, sehingga penutupan kendala tidak bergantung pada urutan unggah foto. Pola ini membantu setiap komponen memeriksa hubungan data walaupun pengiriman terjadi bertahap.

Perubahan data *master* mengikuti arah dari *Cloud* menuju *cache* lokal. Dermaga, kapal, akun, dan konfigurasi yang dibutuhkan operasi dapat digunakan PC Lokal atau Flutter setelah sinkronisasi, tetapi salinan tersebut tidak menjadi sumber perubahan resmi. Sebaliknya, data operasional normal mengalir dari PC Lokal ke *Cloud*. Arah yang berbeda ini digunakan untuk menentukan konflik mana yang dapat diselesaikan otomatis dan mana yang perlu ditahan untuk pemeriksaan.

Status sinkronisasi merupakan bagian penting dari rancangan basis data. Catatan *pending* menunjukkan data masih perlu dikirim, status berhasil menunjukkan penerimaan komponen tujuan, sedangkan penolakan final

memerlukan penanganan tanpa pengulangan tanpa batas. Percobaan, waktu pembaruan, dan posisi rekonsiliasi membantu proses dilanjutkan setelah gangguan. Nilai teknis untuk jumlah percobaan, interval, retensi, dan kapasitas belum ditetapkan sebagai persyaratan final.

Ketiga skema juga mendukung kebutuhan audit tanpa menyalin data sensitif ke dalam laporan. Identitas pengguna disimpan sebagai hubungan data, sementara kredensial dan *token* tidak dicantumkan. Foto serta koordinat terhubung dengan operasi dan aksesnya dibatasi oleh peran. Sebelum penerapan, rancangan masih memerlukan peninjauan migrasi, pencadangan, pemulihan, retensi, perlindungan data, dan pengujian konsistensi dengan data contoh yang telah disetujui.

Integritas relasi perlu diperiksa pada batas transaksi dan sinkronisasi. Pembentukan operasi, perubahan status, serta pencatatan kendala sebaiknya tidak meninggalkan hubungan setengah selesai ketika salah satu penyimpanan gagal. Karena basis data berada pada tiga komponen, konsistensi lintas komponen dicapai melalui validasi, idempotensi, status pengiriman, dan rekonsiliasi, bukan melalui satu transaksi basis data bersama. Rancangan pengujian untuk kondisi tersebut dibahas pada bagian pengujian setelah implementasi tersedia.

4.2.3 Implementasi Sistem/Solusi

Tahap implementasi menerjemahkan rancangan pada BAB III ke dalam tiga komponen utama, yaitu aplikasi *cloud* berbasis Laravel, PC lokal beserta *Human-Machine Interface* (HMI), dan aplikasi petugas berbasis Flutter. Ketiganya dihubungkan melalui API dan mekanisme sinkronisasi yang mempertahankan pencatatan ketika kualitas jaringan berubah. Pembahasan pada subbab ini berfokus pada tujuan setiap komponen, proses implementasi, fitur yang telah dibangun, hubungan dengan rancangan, kontribusi penulis, dan keterbatasan yang diketahui.

Berdasarkan dokumentasi pengembangan dan catatan kegiatan, penulis secara mandiri menyusun basis data, API, autentikasi, data *master*, *dashboard*, perhitungan TRT, pencatatan operasional, penyimpanan lokal, integrasi kamera dan GPS, serta mekanisme sinkronisasi pada seluruh komponen Ro-Ro Alert. Uraian

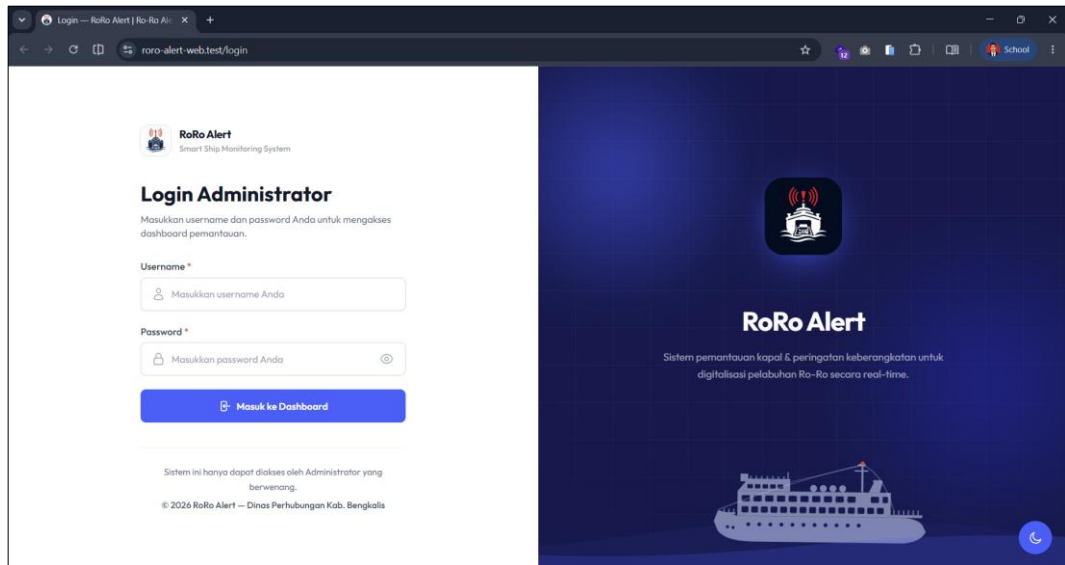
berikut membedakan hasil implementasi dari pembuktian fungsional dan ketahanan jaringan yang ditempatkan pada subbab pengujian sistem.

Dokumentasi visual pada Gambar 4.25 sampai Gambar 4.46 menggunakan tangkapan layar yang telah dimasukkan ke dalam laporan. Setiap gambar diberi nama sesuai fungsi yang ditampilkan agar keterkaitannya dengan uraian implementasi dapat ditelusuri dengan jelas.

1. Implementasi *Cloud* Laravel

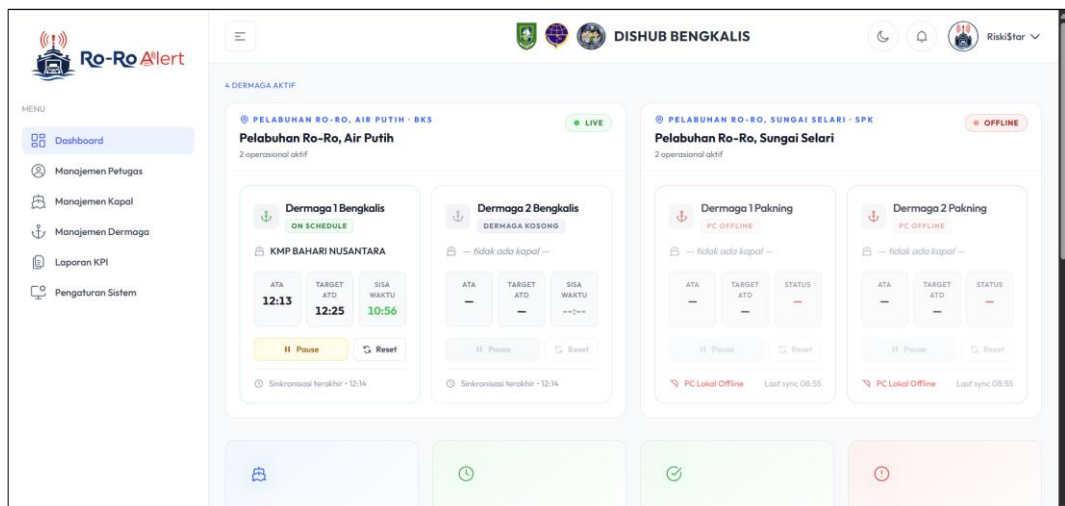
Komponen *cloud* dibangun dengan Laravel sebagai pusat penyimpanan, administrasi, pelaporan, API, dan distribusi pembaruan. Basis data *cloud* menyimpan akun, data *master*, catatan perjalanan, kendala, foto, *status* perangkat, *log* sinkronisasi, serta *log remote control*. Dalam alur normal, PC lokal tetap menjadi otoritas perhitungan TRT, sedangkan *cloud* menerima data secara idempoten dan menyediakan tampilan lintas perangkat.

Implementasi *cloud* meliputi autentikasi, manajemen akun dan data *master*, *dashboard*, laporan KPI, *endpoint* sinkronisasi, *fallback* dari Flutter, *remote control*, *audit trail*, serta Laravel Reverb untuk pembaruan *real-time*. Setiap permintaan operasional menggunakan identitas data agar *retry* dapat ditangani melalui *upsert*. Reverb digunakan untuk mempercepat penyampaian perubahan, namun rekonsiliasi tetap dilakukan melalui API karena *event* tidak bersifat permanen.



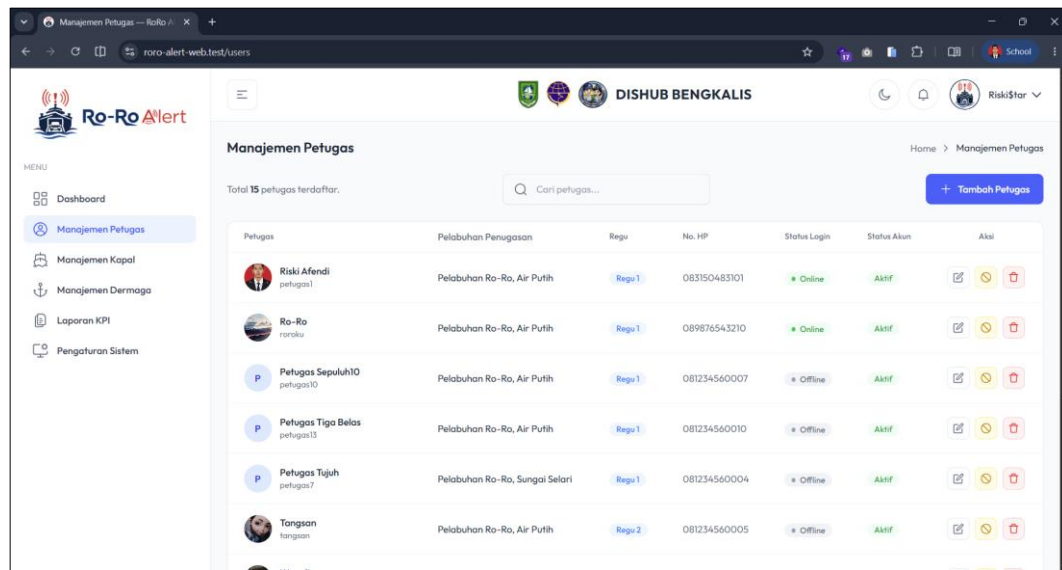
Gambar 4.25 Halaman *Login Admin Ro-Ro Alert Web*

Gambar 4.25 memperlihatkan pintu masuk administrator pada aplikasi *cloud*. Implementasi autentikasi memisahkan akses administrator berbasis sesi *web* dari akses aplikasi bergerak yang menggunakan Laravel Sanctum. Setelah isian dikirim, aplikasi memeriksa identitas pengguna dan hanya mengarahkan akun yang sah ke halaman sesuai hak aksesnya. Penulis secara mandiri mengembangkan alur autentikasi dan pengamanan rute agar pengguna yang belum masuk tidak dapat membuka modul administrasi secara langsung.

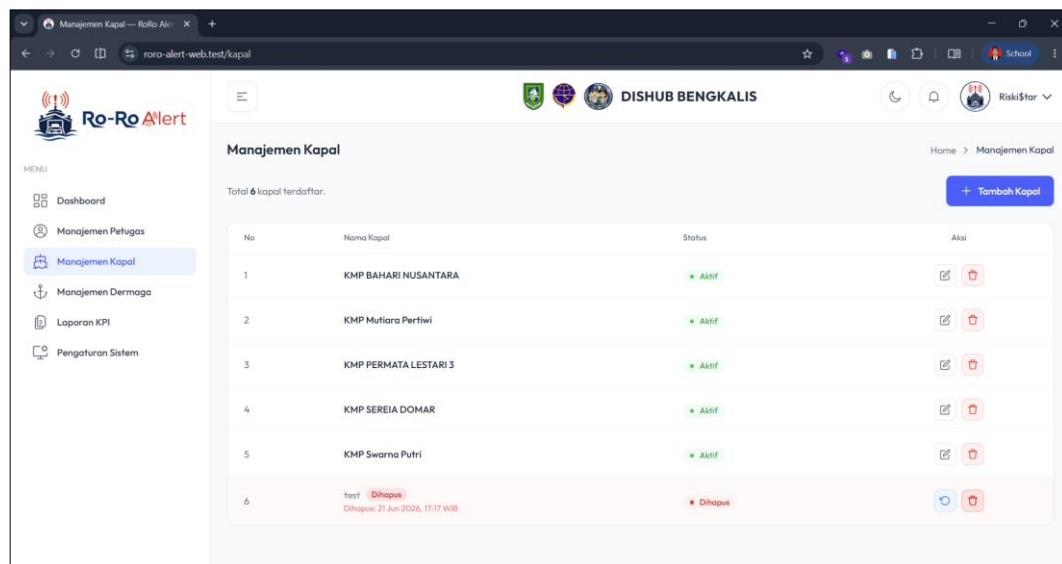


Gambar 4.26 *Dashboard Monitoring Operasional Ro-Ro Alert Web*

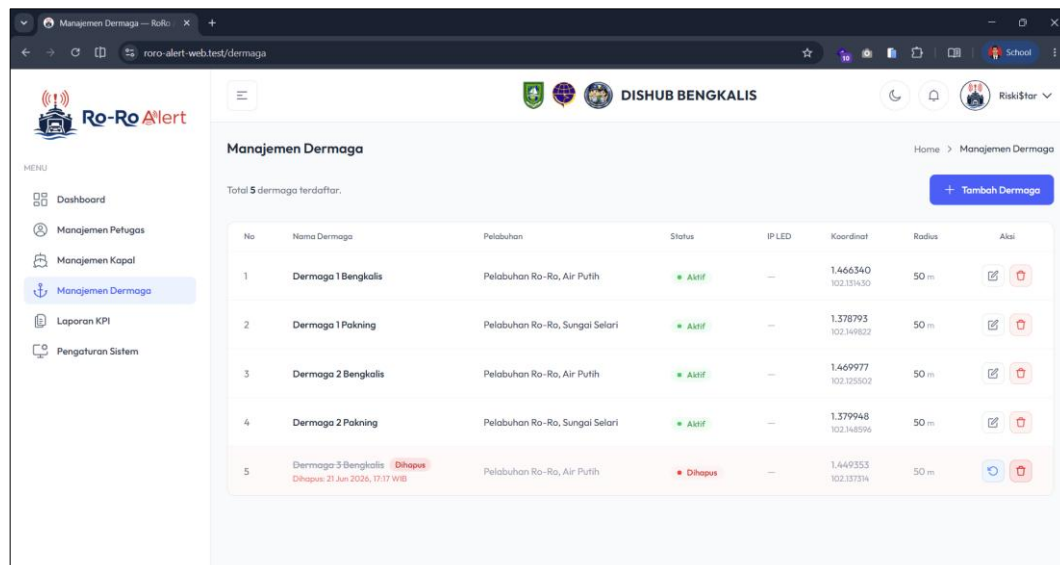
Gambar 4.26 menunjukkan *dashboard* sebagai pusat pemantauan operasional pada sisi *cloud*. Informasi diringkas dari data kapal, dermaga, perjalanan, waktu ATA dan ATD, durasi kendala, *Gross TRT*, *Net TRT*, serta target yang berlaku. Pembaruan dari PC Lokal diterima melalui API dan dapat disebarakan melalui Laravel Reverb. Penulis secara mandiri menyusun data ringkasan, indikator operasional, dan penghubung *dashboard* dengan data hasil sinkronisasi.



Gambar 4.27 Halaman Manajemen Petugas Ro-Ro Alert Web

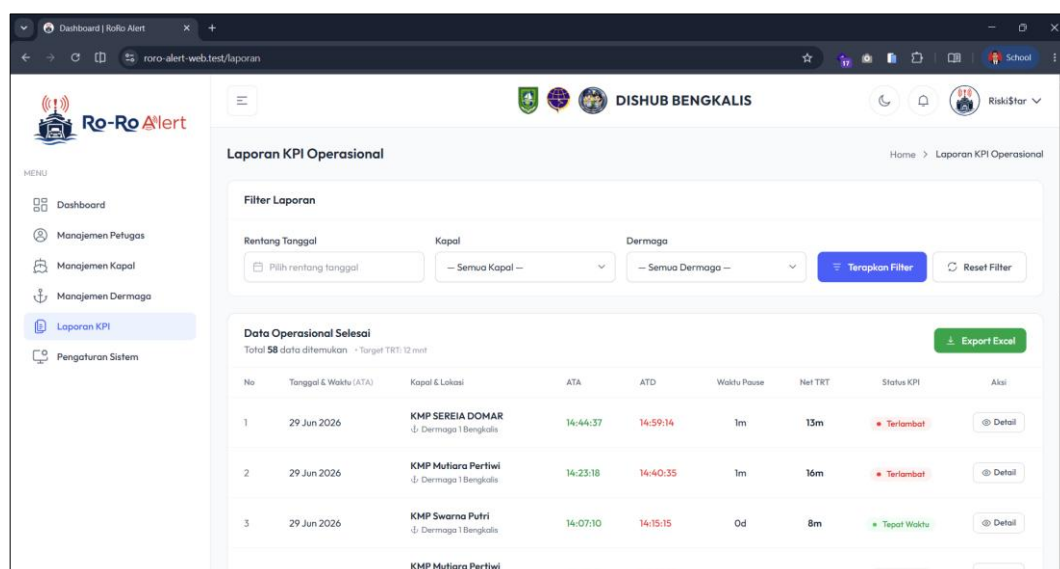


Gambar 4.28 Halaman Manajemen Kapal Ro-Ro Alert Web

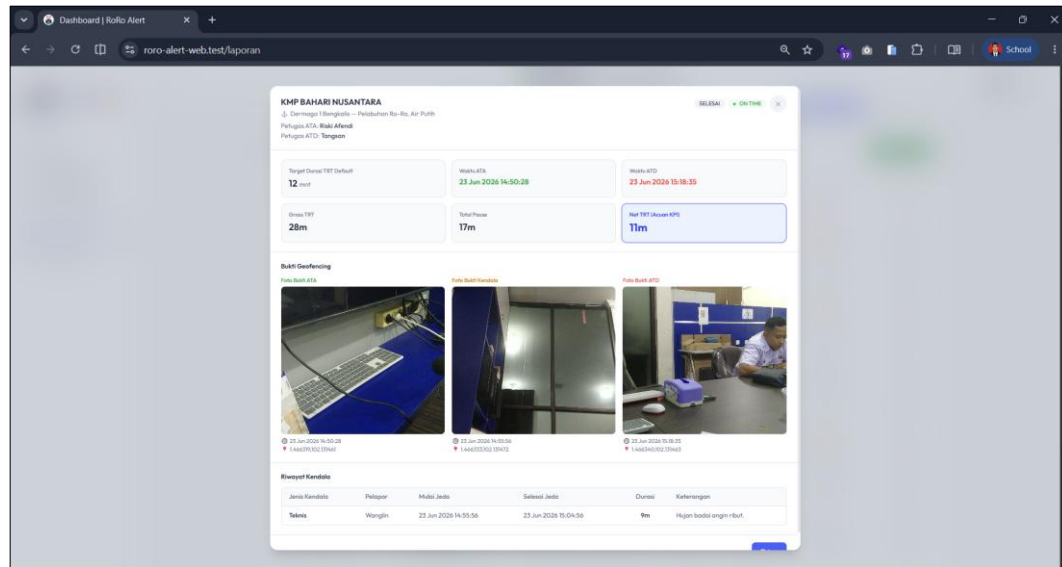


Gambar 4.29 Halaman Manajemen Dermaga Ro-Ro Alert Web

Gambar 4.27 sampai Gambar 4.29 memperlihatkan pengelolaan data *master* petugas, kapal, dan dermaga pada Ro-Ro Alert Web. Ketiga gambar dipertahankan karena setiap halaman menunjukkan entitas dan fungsi administrasi yang berbeda. Data *master* ditempatkan di *cloud* agar konfigurasi pusat dapat digunakan secara konsisten oleh PC Lokal dan aplikasi Flutter. Penulis secara mandiri mengembangkan struktur basis data, relasi model, validasi, serta antarmuka pengelolaan untuk ketiga jenis data tersebut.

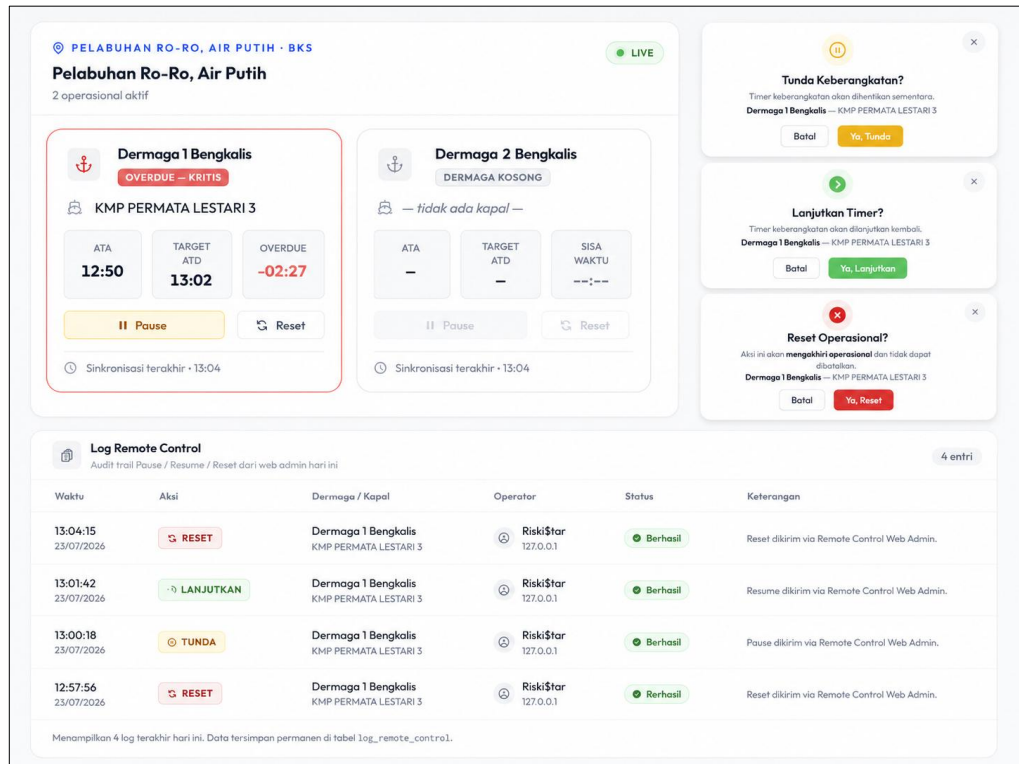


Gambar 4.30 Halaman Laporan KPI Gross TRT dan Net TRT



Gambar 4.31 Detail Operasional pada Laporan KPI *Gross TRT* dan *Net TRT*

Gambar 4.30 menunjukkan halaman laporan KPI yang memuat filter dan daftar operasional selesai, sedangkan Gambar 4.31 menampilkan detail satu operasional beserta target TRT, waktu ATA dan ATD, *Gross TRT*, total jeda, *Net TRT*, bukti foto, serta riwayat kendala. *Gross TRT* dihitung dari rentang ATA sampai ATD, sedangkan *Net TRT* memperhitungkan pengurangan durasi kendala. Penulis secara mandiri mengembangkan kueri agregasi, penyajian ringkasan, filter, detail operasional, dan fasilitas ekspor laporan.



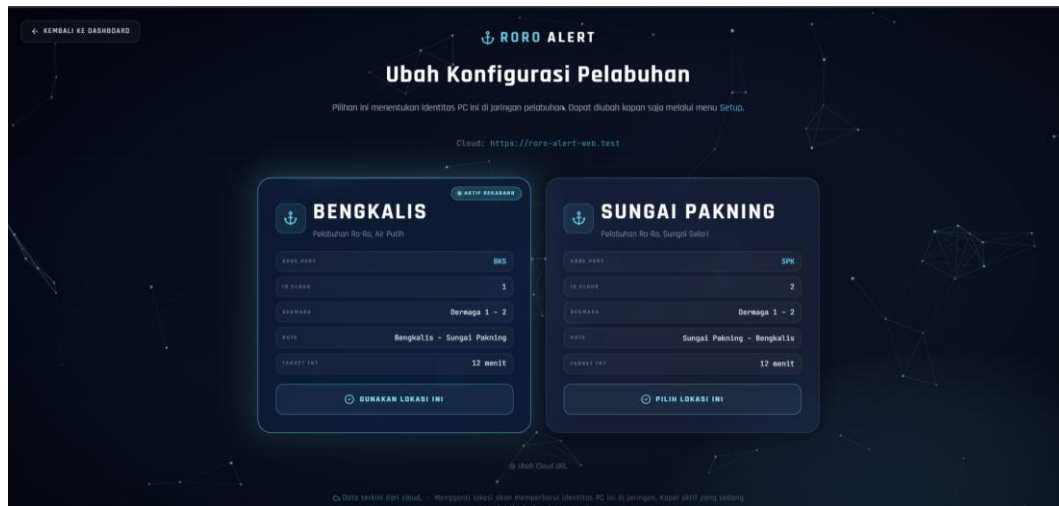
Gambar 4.32 Halaman *Remote Control* dan *Log* Perintah Operasional

Gambar 4.32 memperlihatkan modul *remote control* yang digunakan administrator untuk mengirim perintah operasional tertentu kepada PC Lokal. *Cloud* memvalidasi tujuan dan jenis perintah, menyimpan catatan audit, lalu menyiarkan *event* melalui Reverb. PC Lokal mengeksekusi perintah terhadap kondisi setempat dan mengirim *acknowledgement* agar status pada *log cloud* diperbarui. Penulis secara mandiri mengembangkan pencatatan perintah, penyiaran, penerimaan *acknowledgement*, dan tampilan riwayat.

2. Implementasi PC Lokal dan HMI

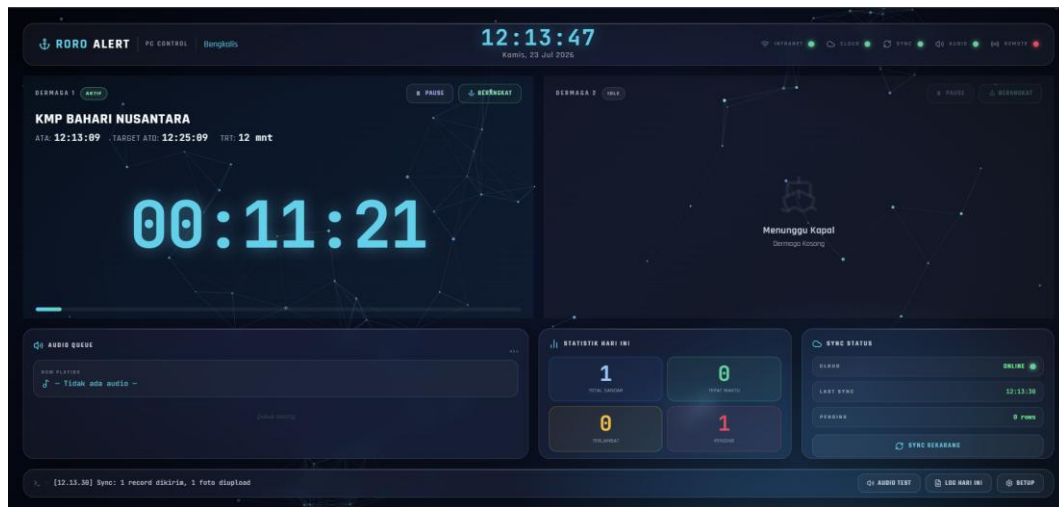
PC lokal dibangun sebagai pusat kendali lapangan dan penyangga data pada kondisi koneksi internet yang tidak stabil. Komponen ini menggunakan basis data lokal, menyediakan API untuk aplikasi Flutter pada jaringan LAN, menjalankan perhitungan TRT, menampilkan HMI, memicu audio dan sirine, serta menyinkronkan catatan ke *cloud*.

Alur operasional pada PC mencakup ATA, ATD, *pause*, *resume*, *reset* sesuai kewenangan, foto dan lokasi, *heartbeat*, *remote command*, serta *fallback catch-up*. Data disimpan lebih dahulu sebelum dikirim, sehingga gangguan internet tidak langsung menghentikan pencatatan. Implementasi ini mengikuti rancangan *offline-first* dan pembagian tanggung jawab antara perangkat lokal dan *cloud*.



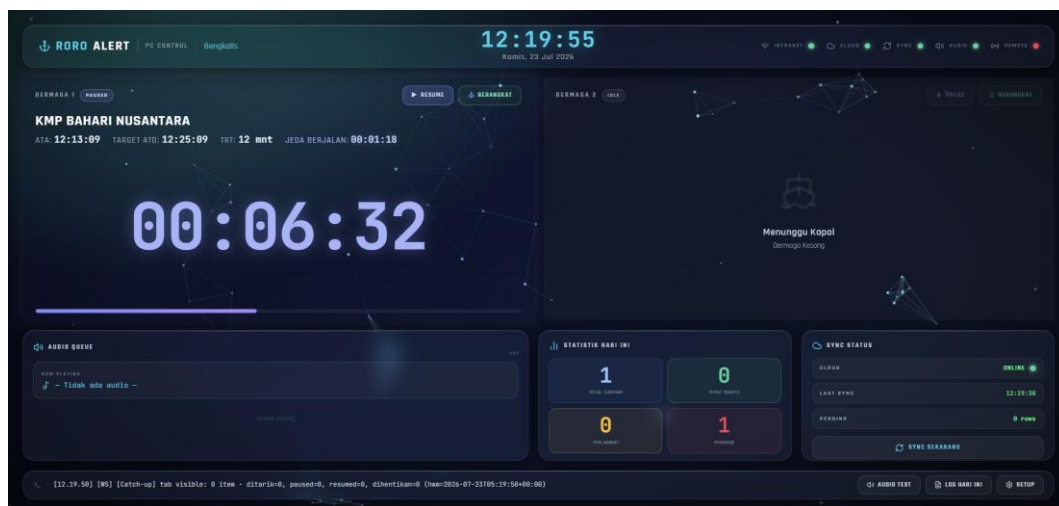
Gambar 4.33 Halaman Instalasi dan Registrasi PC Lokal

Gambar 4.33 memperlihatkan proses awal pemasangan PC Lokal sebelum digunakan sebagai pusat kendali di lapangan. Perangkat menyimpan konfigurasi cakupan, mendaftarkan identitasnya ke *cloud*, kemudian menarik data *master* yang dibutuhkan. Basis data lokal digunakan sebagai penyangga ketika koneksi internet tidak stabil. Penulis secara mandiri mengembangkan alur instalasi, registrasi perangkat, penyimpanan konfigurasi, dan penarikan awal data *master*.



Gambar 4.34 *Dashboard* HMI PC Lokal

Gambar 4.34 menunjukkan HMI pada PC Lokal sebagai tampilan pemantauan terdekat dengan operasi pelabuhan. HMI membaca kondisi dari basis data lokal dan memperbarui status secara berkala, sehingga petugas dapat melihat perjalanan aktif, waktu ATA, TRT berjalan, target, serta keadaan koneksi. Penulis secara mandiri membangun penyajian status, API lokal, dan penghubung antara data operasional dengan HMI.



Gambar 4.35 Status Operasional Kapal dalam Kondisi Dijeda pada HMI PC Lokal

Gambar 4.35 memperlihatkan kondisi satu kapal pada Dermaga 1 ketika operasional sedang dijeda. HMI menampilkan status *paused*, waktu ATA, target ATD, target TRT, durasi jeda berjalan, dan tombol *Resume* serta *Berangkat*. *Timer*

utama tidak bertambah selama status *paused*, sedangkan durasi jeda tetap dicatat sebagai dasar pengurangan pada perhitungan *Net TRT* setelah operasional dilanjutkan. Dermaga 2 ditampilkan dalam keadaan *idle*. Tampilan ini menunjukkan implementasi pemantauan status dan pengendalian operasional pada PC Lokal tanpa menyatakan bahwa siklus *resume* dan ATD telah terjadi pada gambar yang sama.



Gambar 4.36 Status Sinkronisasi, *Heartbeat*, dan Audio Peringatan PC Lokal

Gambar 4.36 menunjukkan komponen pendukung pada PC lokal, yaitu sinkronisasi, *heartbeat*, dan audio peringatan. Catatan lokal diberi status *pending*, *synced*, atau *quarantine* agar proses *retry* dapat dilakukan tanpa menghilangkan data bermasalah. Foto dikirim terpisah dari data utama untuk mengurangi risiko seluruh proses gagal karena berkas berukuran besar. *Heartbeat* menyampaikan keadaan perangkat, sedangkan audio T-10, T-5, T-0 dan sirine memakai status pemicu yang disimpan agar tidak diputarkan berulang. *Scheduler* pada *browser* menjalankan *push* dan *heartbeat* sekitar 30 detik serta *pull* berkala ketika HMI terbuka; kondisi ini menjadi keterbatasan implementasi saat ini.

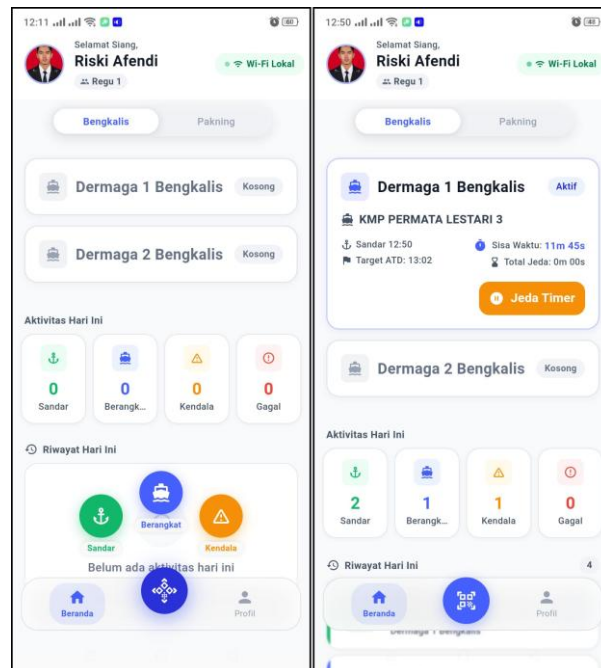
3. Implementasi Aplikasi Flutter

Aplikasi Flutter digunakan petugas lapangan untuk mencatat kejadian dari lokasi operasional. Struktur aplikasi menerapkan BLoC untuk pengelolaan *state*, Drift/SQLite untuk penyimpanan lokal, *secure storage* untuk sesi, dan lapisan jaringan yang dapat memilih jalur LAN atau 4G. Data operasional memakai UUID versi 4 agar pengiriman ulang tetap merujuk pada catatan yang sama.

Fitur yang diimplementasikan meliputi *login*, beranda, pemuatan data *master*, ATA, pelaporan kendala, *resume*, ATD, pengambilan foto dan GPS, riwayat, profil, serta pengaturan koneksi PC lokal. Jalur utama mengarah ke PC melalui LAN, sedangkan *fallback* mengirim ke *cloud* ketika PC tidak terjangkau. Implementasi saat ini masih memiliki keterbatasan berupa *login* yang membutuhkan *cloud*, belum tersedianya *background queue* penuh, dan konfigurasi *production signing* yang belum final.

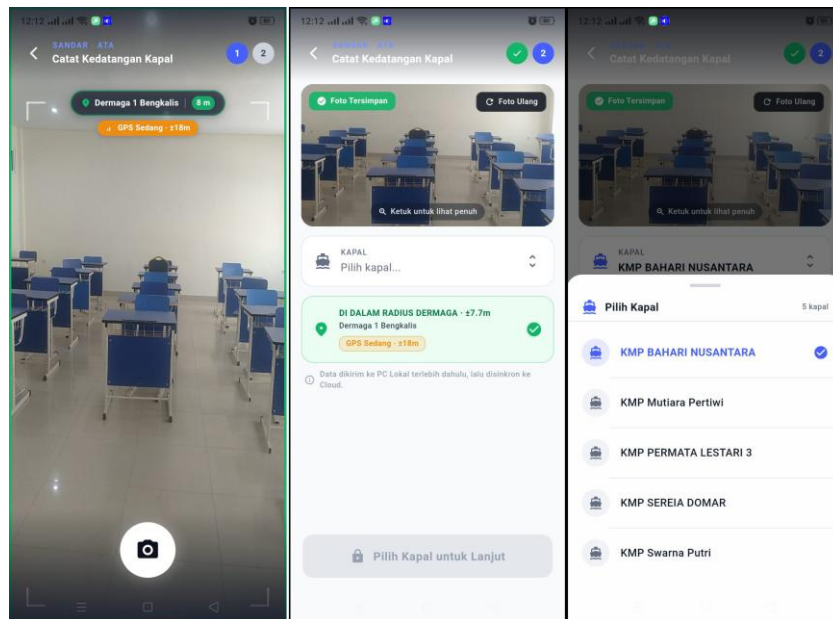
Gambar 4.37 Halaman *Login* Aplikasi Ro-Ro Alert

Gambar 4.37 memperlihatkan autentikasi petugas pada aplikasi Flutter. Proses *login* selalu menghubungi *cloud* melalui *Laravel Sanctum*, lalu *token* yang diterima disimpan pada *secure storage*. Aplikasi tidak menggunakan *refresh token*; apabila layanan mengembalikan status tidak sah, pengguna diarahkan untuk masuk kembali. Penulis mengembangkan integrasi *autentikasi*, penyimpanan sesi, pengamanan tangkapan layar, serta peringatan dasar untuk perangkat *root* atau emulator. Karena autentikasi membutuhkan *cloud*, *login* belum dapat dilakukan sepenuhnya dalam keadaan tanpa internet.



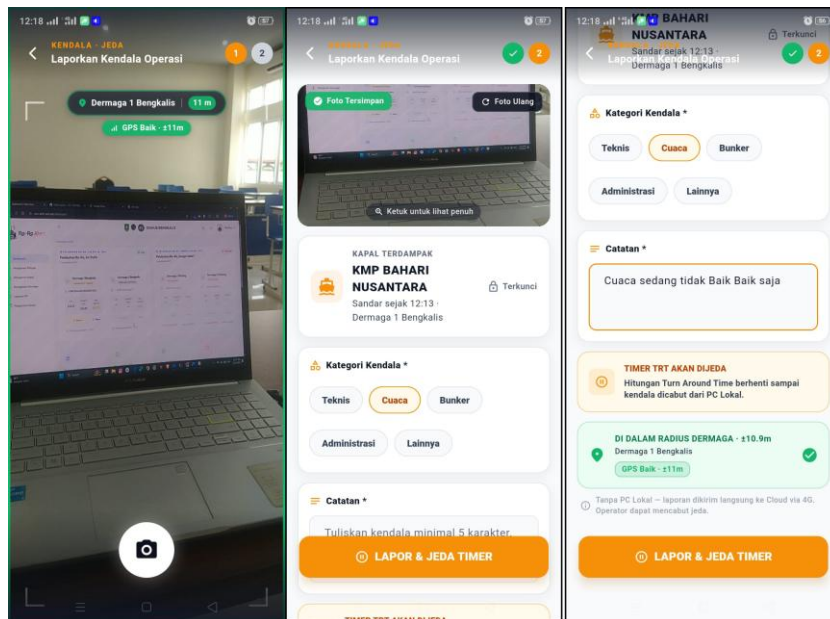
Gambar 4.38 Halaman Beranda Aplikasi Ro-Ro Alert

Gambar 4.38 menunjukkan beranda yang merangkum keadaan operasional petugas. Aplikasi mengambil *snapshot* dari *cloud* dan menggabungkannya dengan catatan lokal pada Drift agar informasi yang baru dibuat di perangkat tetap terlihat. Indikator koneksi membantu petugas mengenali penggunaan jalur LAN menuju PC Lokal atau jalur 4G menuju *cloud*. Penulis secara mandiri mengembangkan pengelolaan *state* dengan pola BLoC, *cache data master*, ringkasan perjalanan, dan pemilihan jalur jaringan otomatis.



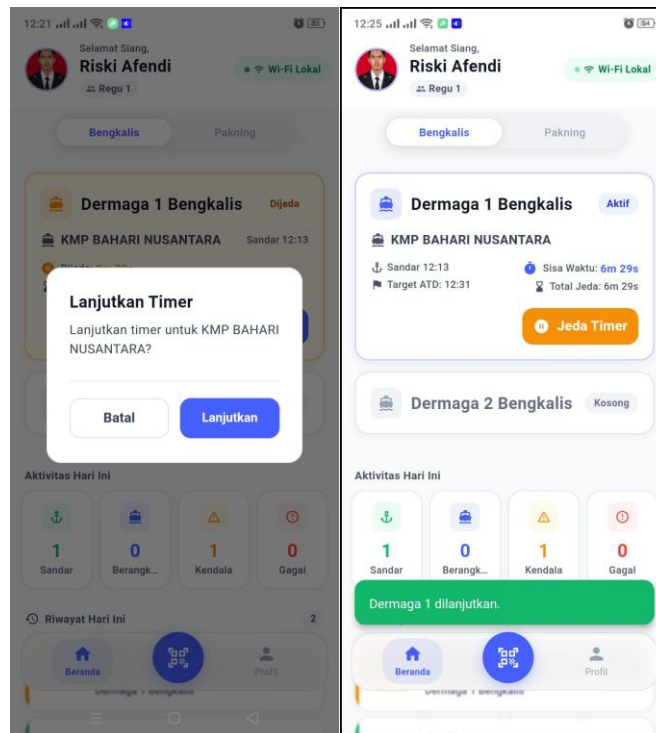
Gambar 4.39 Halaman Pencatatan Sandar (ATA)

Gambar 4.39 memperlihatkan pencatatan waktu kapal sandar atau ATA sebagai awal perjalanan operasional. Aplikasi menyiapkan UUID versi 4 pada catatan sehingga permintaan yang sama dapat dikirim ulang tanpa membuat data ganda. Formulir dapat menyertakan waktu, koordinat GPS, pemeriksaan *geofence*, foto, dan *watermark* sesuai konfigurasi. Pada kondisi LAN, data dikirim ke PC lokal; ketika jalur tersebut tidak tersedia, aplikasi dapat memakai jalur 4G ke *cloud*. Penulis mengembangkan formulir, validasi, penyimpanan lokal, pengambilan foto/GPS, dan integrasi *multipart* untuk proses ini.



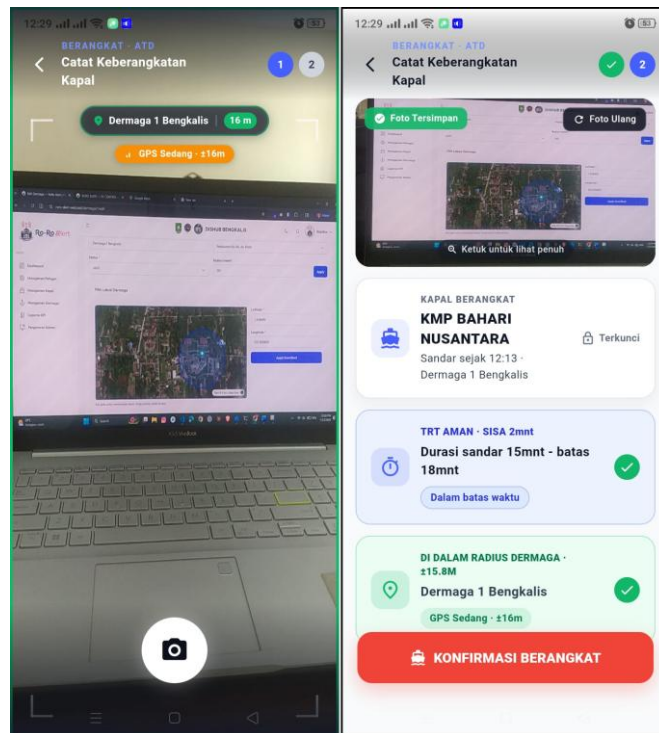
Gambar 4.40 Halaman Pelaporan Kendala Operasional

Gambar 4.40 menunjukkan formulir pelaporan kendala yang menghentikan perhitungan *Net TRT* untuk sementara. Petugas memilih jenis kendala dan dapat menambahkan catatan, posisi, serta foto pendukung. Catatan kendala disimpan dengan UUID yang tetap dipakai pada saat *retry*, lalu dikirim melalui jalur jaringan yang tersedia. Pada PC lokal, perubahan tersebut memperbarui status perjalanan dan menjadi dasar pemicu informasi pada HMI. Penulis mengimplementasikan alur *pause*, validasi data, penyimpanan lokal, serta pengiriman bukti pendukung tanpa memasukkan data sensitif ke *log* aplikasi.



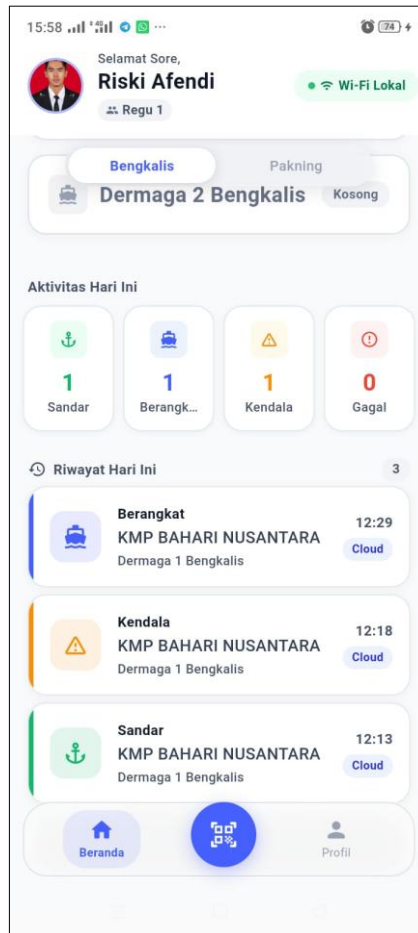
Gambar 4.41 Halaman Penyelesaian Kendala (*Resume*)

Gambar 4.41 memperlihatkan penyelesaian kendala atau *resume*. Aplikasi menampilkan kendala aktif agar petugas tidak menutup catatan yang keliru, kemudian menyimpan waktu selesai sebagai pasangan waktu *pause*. Selisih kedua waktu dipakai untuk menghitung total durasi kendala dan *Net TRT*. Penulis secara mandiri mengembangkan pelacak *pause* lokal, pemilihan kendala aktif, pencegahan penyelesaian berulang, dan sinkronisasi *resume* melalui LAN maupun *fallback* 4G.



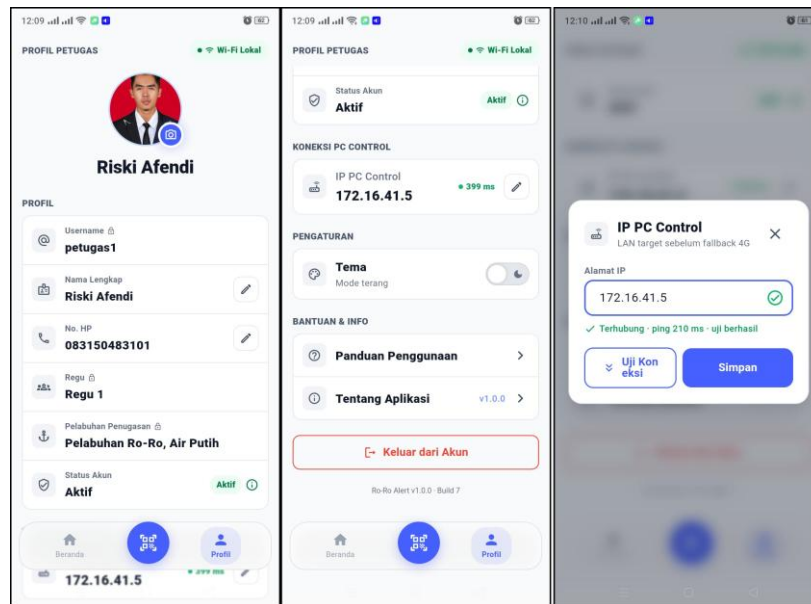
Gambar 4.42 Halaman Pencatatan Berangkat (ATD)

Gambar 4.42 menunjukkan pencatatan ATD sebagai penutup perjalanan. Sebelum data dikirim, aplikasi memeriksa perjalanan aktif dan membentuk catatan beridentitas UUID. PC lokal atau *cloud* pada jalur *fallback* kemudian menerapkan aturan finalisasi agar pengiriman ulang tidak menghasilkan penutupan ganda. Waktu ATD bersama ATA dan durasi kendala menjadi dasar nilai akhir *Gross TRT* dan *Net TRT*. Penulis mengimplementasikan formulir ATD, pemeriksaan keadaan perjalanan, penyertaan bukti, penyimpanan lokal, dan pengiriman ke jalur jaringan yang tersedia.



Gambar 4.43 Halaman Riwayat Operasional

Gambar 4.43 memperlihatkan riwayat operasional yang tersimpan pada perangkat. Catatan lokal membantu petugas meninjau tindakan yang baru dilakukan meskipun jaringan berubah, sekaligus menunjukkan status pengiriman setiap data. Penyimpanan *Drift* memuat catatan operasional, *cache* data *master*, sesi, dan konfigurasi yang dibutuhkan aplikasi. Implementasi saat ini belum berupa *background queue* penuh; karena itu proses pengiriman ulang tetap bergantung pada alur aplikasi yang tersedia. Penulis mengembangkan penyimpanan lokal, pemetaan status, dan tampilan riwayat sebagai dukungan keterlacakan di sisi petugas.



Gambar 4.44 Halaman Profil dan Pengaturan Koneksi

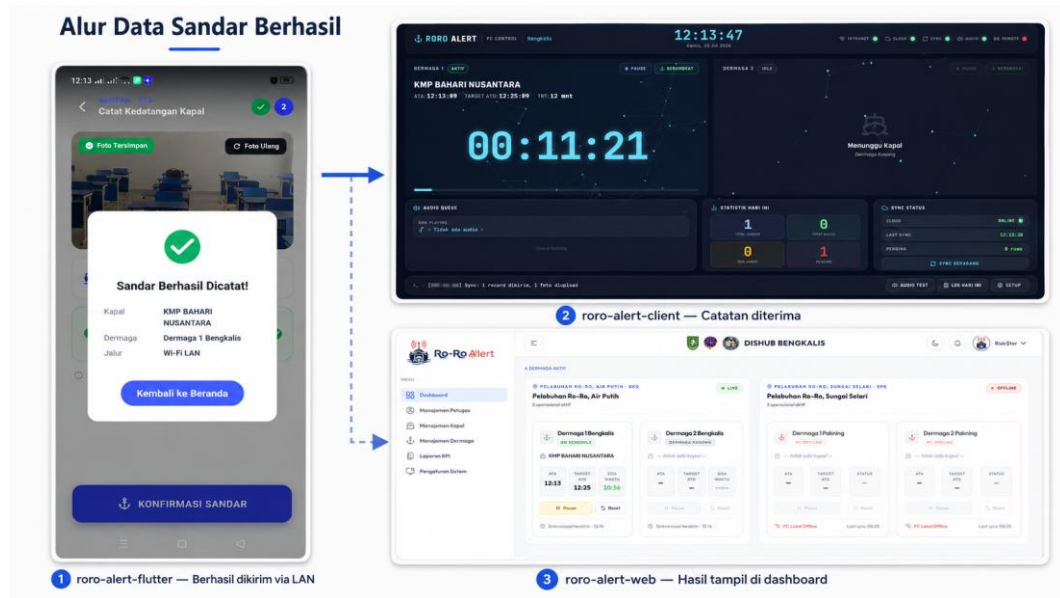
Gambar 4.44 menunjukkan halaman profil dan pengaturan koneksi yang digunakan petugas untuk memeriksa sesi serta tujuan PC Lokal. Konfigurasi alamat PC diperlukan agar aplikasi dapat memakai LAN sebagai jalur utama, sedangkan pemeriksaan jaringan menentukan perpindahan ke 4G ketika PC tidak dapat dijangkau. Penulis secara mandiri mengembangkan pembacaan profil, penyimpanan konfigurasi, pemilihan mode jaringan, dan *logout*.

4. Implementasi Integrasi dan Sinkronisasi

Integrasi menghubungkan ketiga komponen melalui kontrak API yang menetapkan struktur *payload*, identitas UUID, *status* respons, dan aturan perubahan data. Jalur normal mengalir dari Flutter ke PC lokal melalui LAN, lalu dari PC ke *cloud*. Jalur *fallback* mengalir dari Flutter ke *cloud* melalui 4G dan didistribusikan kembali kepada PC melalui Reverb serta proses *catch-up*.

Idempotensi diterapkan agar *retry* tidak membuat catatan ganda. PC menyimpan *status* sinkronisasi, *cloud* melakukan *upsert* berdasarkan UUID, dan rekonsiliasi memeriksa *high-water mark* serta keadaan final perjalanan. *Acknowledgement* digunakan pada perintah jarak jauh dan perubahan tertentu agar *cloud* dapat mencatat penerimaan oleh PC. Mekanisme tersebut mengurangi risiko

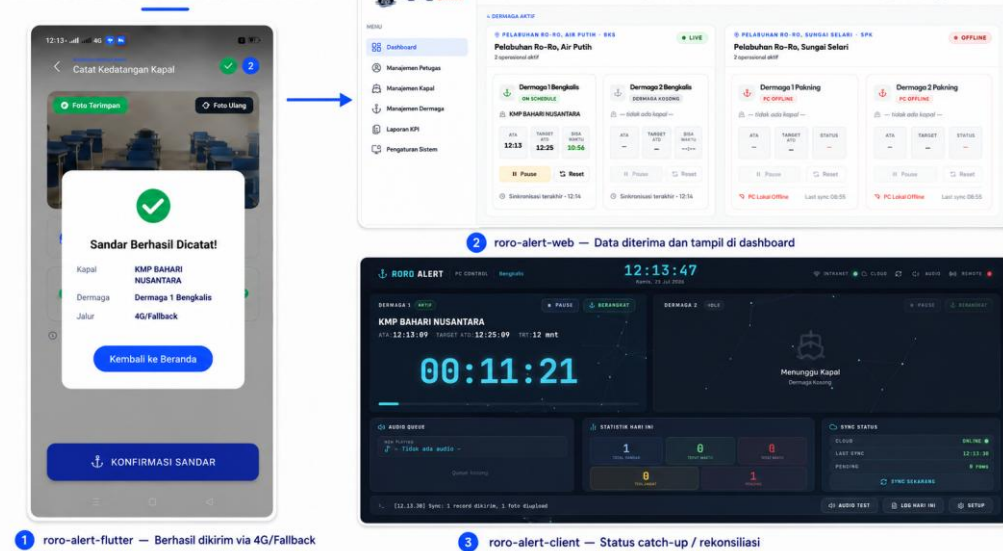
selisih data, tetapi hasilnya pada gangguan jaringan nyata tetap menjadi ruang lingkup pengujian.



Gambar 4.45 Sinkronisasi Normal Flutter-PC Lokal-Cloud

Gambar 4.45 memperlihatkan jalur sinkronisasi normal dari Flutter ke PC Lokal dan selanjutnya ke *cloud*. Aplikasi mengirim catatan melalui LAN dengan UUID yang tetap. PC Lokal menyimpan data, memperbarui TRT, lalu mendorong data *pending* ke *cloud*. *Cloud* melakukan *upsert* berdasarkan UUID sehingga *retry* tidak membentuk catatan baru. Penulis secara mandiri mengembangkan kontrak API, pemetaan *payload*, status sinkronisasi, dan penanganan duplikasi pada ketiga komponen.

Alur Data Sandar 4G/Fallback



Gambar 4.46 Fallback 4G dan Rekonsiliasi Data ke PC Lokal

Gambar 4.46 menunjukkan jalur *fallback* ketika aplikasi Flutter tidak dapat menjangkau PC lokal melalui LAN. Catatan dikirim ke *cloud* menggunakan UUID yang sama, kemudian *cloud* menyimpan data dan dapat menghitung TRT sementara pada jalur *fallback*. Perubahan disiarkan melalui Reverb agar PC lokal dapat segera memperbarui keadaan, tetapi *rekonsiliasi* tidak bergantung pada *event* tersebut. PC melakukan *catch-up* dengan *high-water mark* untuk mengambil data yang terlewat, memeriksa UUID dan keadaan final, serta mengirim *acknowledgement* bila diperlukan. Mekanisme ini dirancang untuk mengurangi duplikasi dan perbedaan keadaan setelah koneksi pulih. Penulis mengimplementasikan pemilihan jalur jaringan, *endpoint fallback*, *event real-time*, *catch-up*, serta aturan *idempotensi*. Keterbatasannya, *overlay* status pada Flutter bersifat lokal pada perangkat dan ketahanan seluruh skenario masih perlu dinilai melalui pengujian terstruktur.

5. Pengujian Sistem

Pengujian sistem dilakukan dengan membedakan pengujian yang dapat dijalankan ulang, bukti integrasi pada *changelog*, dan catatan *runtime* aplikasi. Pengujian ulang pada 19 Juli 2026 menggunakan laptop ASUS VivoBook dengan prosesor 11th Gen Intel Core i3-1125G4 @ 2.00 GHz, RAM 8 GB, sistem operasi

Windows 11 Home Single Language 64-bit, dan prosesor berbasis x64. Perangkat lunak pendukung meliputi Laragon, Visual Studio Code, serta Google Chrome dengan panel/ekstensi Google Lighthouse. Komponen PC Lokal diuji menggunakan PHP 8.5.5 dan Node.js 24.0.2. Komponen *Cloud* diuji menggunakan PHP 8.5.5, Laravel 12.56.0, dan basis data SQLite *in-memory*.

Status Lulus hanya digunakan ketika hasil yang dapat diperiksa memenuhi keluaran yang diharapkan. Status Lulus dengan catatan digunakan ketika fungsi berhasil tetapi bukti terbatas pada pengujian internal, *changelog*, atau *log*. Status Gagal menunjukkan hasil aktual yang tidak memenuhi harapan. Status Belum diuji dan Perlu validasi lebih lanjut dipakai ketika rencana uji tersedia tetapi artefak hasil, kondisi perangkat, atau persetujuan belum memadai.

Perangkat Android yang digunakan adalah OPPO A5 2020 dengan Android 9 dan RAM 3 GB. Dokumentasi pengujian lanjutan masih perlu mencatat nomor *build* APK, tanggal dan jam uji, jenis jaringan atau SIM, lokasi, data masukan, serta bukti tangkapan layar atau *log* yang disetujui.

Tabel 4. 4 Skenario Pengujian Fungsional

No.	Komponen/Fitur	Skenario	Hasil yang Diharapkan	Hasil Aktual	Status
1	PC Lokal - fungsi sinkronisasi	Menjalankan pengujian fungsi konversi waktu, pemetaan status, perhitungan TRT, dan klasifikasi data sinkronisasi.	Seluruh keluaran sesuai dengan data uji.	17 skenario lulus dan 0 gagal pada PHP 8.3.23.	Lulus
2	PC Lokal - mesin audio	Menjalankan pengujian <i>state machine</i> audio dengan objek audio tiruan.	Pemicu, sirine, deduplikasi, <i>pause/resume</i> , dan pemulihan <i>state</i> sesuai aturan.	53 skenario lulus dan 0 gagal pada Node.js 24.0.2.	Lulus dengan catatan
3	<i>Cloud - dashboard</i> dan aktivitas	Menjalankan lima pengujian fitur filter, KPI, status, dan aktivitas <i>dashboard</i> .	Seluruh respons dan perhitungan sesuai data uji.	Lima pengujian lulus pada Laravel 12.56.0 dan SQLite <i>in-memory</i> .	Lulus
4	<i>Cloud - konfigurasi</i> foto	Menguji nilai konfigurasi foto yang hilang atau tidak sah.	Sistem memakai nilai <i>fallback</i> yang aman.	Pengujian unit lulus.	Lulus
5	<i>Cloud - akses awal</i>	Menguji pengalihan dari halaman awal menuju halaman administrasi.	Pengalihan menghasilkan tujuan yang benar.	Pengujian fitur lulus.	Lulus
6	<i>Cloud - autentikasi fallback</i>	Mengirim permintaan <i>fallback</i> tanpa autentikasi Sanctum.	Permintaan ditolak.	Pengujian fitur lulus.	Lulus

No.	Komponen/Fitur	Skenario	Hasil yang Diharapkan	Hasil Aktual	Status
7	Cloud - <i>fallback</i> kendala	Menjalankan enam skenario <i>pause, resume</i> , dan ATD pada basis data uji.	Migrasi dan <i>fixture</i> menyediakan seluruh kolom yang diperlukan.	Enam pengujian gagal karena tabel <i>dermaga</i> pada SQLite uji tidak memiliki kolom lokasi <i>_pelabuhan</i> . Karena sudah diganti Latitude & Longitude.	Gagal
8	PC Lokal - ATA dan ATD	Meninjau catatan <i>endpoint</i> lokal untuk sandar dan berangkat dengan foto serta koordinat.	Tindakan tercatat dan menghasilkan respons berhasil.	<i>Log</i> memuat entri ATA dan ATD berhasil sampai 1 Juli 2026.	Lulus dengan catatan
9	Flutter - unit dan <i>state</i>	Memeriksa bukti hasil pengujian aplikasi Flutter.	Tersedia laporan hasil formal dan artefak yang dapat diperiksa.	Hanya ditemukan catatan <i>logbook</i> berstatus <i>draft</i> ; keluaran formal pengujian tidak tersedia.	Perlu validasi lebih lanjut

Tabel 4.5 Hasil Pengujian Integrasi Flutter, PC Lokal, dan Cloud

No.	Skenario Integrasi	Komponen	Kondisi	Hasil Aktual	Status/Bukti
1	ATA/ATD melalui LAN	Flutter-PC Lokal	Wi-Fi LAN	<i>Changelog</i> v2.6.1 mencatat alur end-to-end telah diverifikasi.	Lulus dengan catatan; laporan formal perangkat tidak tersedia
2	ATA/ATD melalui <i>fallback</i>	Flutter-Cloud	Jaringan 4G	<i>Changelog</i> v2.6.1 dan <i>log fallback</i> mencatat alur sandar/berangkat.	Lulus dengan catatan; detail perangkat dan jaringan belum tersedia

No.	Skenario Integrasi	Komponen	Kondisi	Hasil Aktual	Status/Bukti
3	Perintah jarak jauh	Cloud-PC Lokal	<i>Pause, resume, reset, dan acknowledgement</i>	<i>Changelog</i> v2.5.1 mencatat perintah dan <i>acknowledgement</i> telah diverifikasi.	Lulus dengan catatan; tidak ada laporan uji bertanda tangan
4	Kendala melalui <i>fallback</i>	Flutter-Cloud-PC Lokal	<i>Pause dan resume</i>	<i>Log</i> mencatat <i>pause</i> dan <i>resume</i> ; satu permintaan <i>fallback</i> menghasilkan HTTP 404.	Lulus dengan catatan
5	<i>Heartbeat</i>	PC Lokal-Cloud	Koneksi pengembangan/lokal	<i>Log</i> memuat <i>timeout</i> dan HTTP 500 berulang.	Gagal; konektivitas perlu diperiksa
6	Sinkronisasi foto	PC Lokal-Cloud	Unggah foto ATA, ATD, dan kendala	<i>Log</i> memuat kegagalan unggah HTTP 500 pada beberapa tanggal.	Gagal; <i>endpoint</i> /penyimpanan perlu diperiksa
7	<i>Catch-up</i> rekonsiliasi	Cloud-PC Lokal	Siklus pemeriksaan berkala	Siklus berjalan, tetapi seluruh catatan yang ditemukan berisi 0 <i>item</i> .	Perlu validasi lebih lanjut
8	Demonstrasi tiga komponen	Flutter-PC Lokal-Cloud	Demonstrasi 29 Juni 2026	Tercantum pada <i>logbook</i> berstatus <i>draft</i> tanpa bukti perangkat dan hasil yang disetujui.	Perlu validasi lebih lanjut

Tabel 4.6 Hasil Pengujian Audio, Sinkronisasi, dan Fallback

No.	Skenario	Kondisi Uji	Hasil yang Diharapkan	Hasil Aktual	Status
1	Pemetaan waktu, status, dan TRT	Data sintetis pada <i>helper</i> sinkronisasi	Hasil konversi dan perhitungan sesuai nilai uji.	Bagian terkait dalam 17 skenario lulus.	Lulus
2	Klasifikasi data sinkronisasi	<i>already_final</i> , karantina, <i>retry</i> , dan <i>markSynced</i>	Setiap respons masuk ke tindak lanjut yang tepat.	Bagian terkait dalam 17 skenario lulus; <i>log</i> juga memuat penanganan <i>already_final</i> .	Lulus
3	Pemicu audio dan sirine	T-10, T-5, T-0, deduplikasi, <i>pause/resume</i> , dan reload	<i>State machine</i> mencegah pengulangan serta mengikuti status operasi.	Bagian terkait dalam 53 skenario lulus memakai MockAudio.	Lulus dengan catatan
4	Keluaran audio perangkat fisik	<i>Speaker/amplifier</i> , beberapa <i>tab</i> , dan mute sistem operasi	Suara terdengar sesuai urutan tanpa tumpang tindih.	Tersedia rencana uji, tetapi tidak ada hasil lapangan.	Belum diuji
5	Pemulihan audio	<i>Restart</i> , penggunaan semalam, mode privat, dan pergantian keluaran audio	<i>State</i> pulih dan audio tetap dapat diputar.	Tersedia skenario manual, tetapi tidak ada hasil formal.	Belum diuji
6	<i>Fallback</i> operasional	Sandar, <i>pause</i> , <i>resume</i> , dan berangkat	Tindakan tersimpan dan dapat ditelusuri pada <i>log</i> .	<i>Log</i> memuat keempat tindakan; terdapat satu respons 404.	Lulus dengan catatan
7	<i>Heartbeat</i>	Koneksi PC Lokal menuju <i>Cloud</i>	Status koneksi terkirim tanpa galat berulang.	Terjadi <i>timeout</i> dan HTTP 500.	Gagal
8	Sinkronisasi foto	Unggah bukti ATA, ATD, dan kendala	Foto diterima <i>Cloud</i> dan ditandai tersinkron.	<i>Log</i> memuat HTTP 500 pada beberapa unggahan.	Gagal
9	<i>Catch-up</i> berisi data	Koneksi kembali setelah ada data <i>fallback</i>	<i>Batch</i> non-kosong diterapkan secara idempoten.	<i>Log</i> yang tersedia hanya memuat siklus dengan 0 <i>item</i> .	Perlu validasi lebih lanjut

Hasil pengujian ulang menunjukkan 17 skenario *helper* sinkronisasi dan 53 skenario mesin audio lulus. Namun, keberhasilan mesin audio masih terbatas pada MockAudio dan belum membuktikan keluaran pada *speaker*, *amplifier*, peramban, atau perangkat fisik. Pada *Cloud*, 9 pengujian lulus dan 6 pengujian gagal. Enam kegagalan berasal dari ketidaksesuaian skema pengujian karena kolom lokasi_pelabuhan belum tersedia pada tabel dermaga di SQLite uji. Oleh karena itu, kegagalan tersebut perlu diperbaiki pada *migration* atau *fixture* sebelum alur *fallback* dapat dinilai kembali.

Log aplikasi juga menunjukkan kondisi yang belum tertutup oleh pengujian unit. Pencatatan ATA, ATD, *pause*, *resume*, dan *fallback* telah terekam, tetapi *heartbeat* serta unggah foto masih memuat *timeout* atau HTTP 500. *Log catch-up* hanya berisi siklus dengan nol *item* sehingga belum membuktikan pemulihan *batch* non-kosong. Entri pengujian dan demonstrasi pada *logbook* masih berstatus *draft*, sehingga tidak digunakan sebagai persetujuan pengguna atau laporan pengujian formal.

4.2.4 Dampak Implementasi Sistem/Solusi

Dampak implementasi pada tahap ini dibatasi pada perubahan yang terlihat pada prototipe dan hasil pengujian internal. Belum tersedia pengukuran sebelum-sesudah, data waktu pelayanan, data antrean, atau validasi pengguna yang dapat digunakan untuk menyatakan peningkatan kuantitatif. Oleh karena itu, uraian berikut memakai istilah potensi manfaat, perubahan pada prototipe, dan masih memerlukan validasi pengguna atau lapangan.

Tabel 4.7 Perbandingan Proses Sebelum dan Sesudah Implementasi

No.	Aspek	Kondisi Sebelumnya	Kondisi pada Prototipe/Rancangan	Bukti	Keterangan
1	Pencatatan	Catatan operasional berasal dari beberapa tahap dan media kerja.	Prototipe menghubungkan ATA, ATD, kendala, foto, lokasi, dan petugas melalui identitas operasi.	<i>Log</i> ATA/ATD/ <i>fallback</i> dan pengujian internal.	Potensi manfaat; masih memerlukan validasi pengguna.
2	Ketersediaan data	Ketergantungan pada jalur koneksi dapat menghambat pengiriman ke pusat.	Prototipe menyediakan jalur LAN, <i>fallback</i> 4G, penyimpanan lokal, dan pengiriman ulang.	<i>Changelog</i> v2.6.1 dan <i>log fallback</i> .	Perubahan pada prototipe; ketahanan lapangan belum diukur.
3	<i>Monitoring</i>	Informasi operasional tidak berada dalam satu tampilan pemantauan prototipe.	<i>Dashboard</i> menggabungkan status operasi, aktivitas, KPI, dan status koneksi.	Lima pengujian <i>dashboard</i> lulus.	Berdasarkan hasil pengujian internal; belum mengukur dampak layanan.
4	Sinkronisasi	Perubahan lintas perangkat berisiko terlambat atau berulang.	UUID, klasifikasi respons, <i>retry</i> , karantina, <i>acknowledgement</i> , dan rekonsiliasi disiapkan.	17 pengujian <i>helper</i> lulus; <i>log</i> menampilkan error aktual.	Potensi manfaat dengan catatan masalah <i>heartbeat</i> dan foto belum selesai.
5	Dokumentasi	Jejak teknis perubahan dan gangguan belum tersusun dalam satu rangkaian prototipe.	<i>Log</i> , <i>changelog</i> , status sinkronisasi, dan riwayat tindakan menyediakan jejak pemeriksaan.	GitBook <i>changelog</i> dan <i>log</i> aplikasi.	Perubahan pada prototipe; kebijakan retensi perlu ditetapkan.
6	Kesiapan pengembangan berikutnya	Pengembangan antarkomponen memerlukan acuan fungsi dan integrasi.	Pengujian internal, kontrak integrasi, dan daftar temuan memberi dasar perbaikan berikutnya.	Hasil uji 19 Juli 2026 dan daftar kendala.	Potensi manfaat; masih memerlukan validasi lapangan dan pengguna.

Berdasarkan hasil pengujian internal, prototipe telah menyediakan satu rangkaian pencatatan, pemantauan, sinkronisasi, dan dokumentasi teknis. Perubahan tersebut dapat menjadi dasar pengembangan berikutnya, tetapi belum dapat dinyatakan sebagai peningkatan efisiensi, pengurangan antrean, atau peningkatan pelayanan. Pernyataan dampak operasional memerlukan data pembandingan dan validasi pengguna pada kondisi kerja yang sebenarnya.

4.2.5 Kendala Implementasi

Kendala implementasi disusun dari hasil pengujian yang dijalankan ulang, *log* aplikasi, *changelog*, rencana pengujian manual, dan keterbatasan bukti yang tersedia. Kendala dibedakan antara masalah yang telah memiliki penanganan awal, masalah yang masih perlu diperbaiki, dan bagian yang hanya dapat dipastikan melalui validasi lapangan.

Tabel 4.8 Kendala Implementasi dan Tindak Lanjut

No.	Kendala	Penyebab	Dampak	Penanganan	Status/Tindak Lanjut
1	Konektivitas <i>heartbeat</i>	<i>Timeout</i> dan HTTP 500 pada komunikasi PC Lokal- <i>Cloud</i> .	Status koneksi pada pusat dapat tidak mutakhir.	<i>Retry</i> dan pencatatan galat sudah tersedia; layanan tujuan dan jaringan perlu diperiksa.	Belum selesai
2	Lingkungan uji integrasi <i>Cloud</i>	Skema SQLite uji tidak memiliki kolom <i>dermaga.lokasi_pelabuhan</i> .	Enam skenario <i>fallback</i> tidak dapat mencapai langkah utama.	Selaraskan <i>migration</i> , <i>factory</i> , dan <i>fixture</i> , lalu jalankan kembali seluruh suite.	Belum selesai
3	Duplikasi/status akhir sinkronisasi	Respons <i>already_final</i> dapat terus diulang jika tidak diklasifikasikan.	Antrean sinkronisasi berisiko tertahan oleh data yang tidak perlu dikirim ulang.	<i>Helper</i> memisahkan <i>markSynced</i> , karantina, dan <i>retry</i> ; pengujian terkait lulus.	Selesai dengan catatan
4	Sinkronisasi foto	<i>Endpoint</i> atau penyimpanan <i>Cloud</i> menghasilkan HTTP 500.	Bukti foto belum selalu tersinkron walaupun metadata tersedia.	Pisahkan proses unggah, simpan status <i>retry</i> , dan telusuri respons <i>server</i> .	Memerlukan pengembangan lanjutan
5	Kompatibilitas perangkat	Matriks model Android, versi OS, dan kondisi jaringan belum terdokumentasi.	Kesesuaian antargawai belum dapat dinyatakan.	Lakukan uji perangkat nyata dan isi identitas perangkat serta <i>build</i> APK.	Memerlukan validasi lapangan
6	Audio perangkat fisik	Pengujian saat ini memakai MockAudio dan belum mencakup <i>speaker</i> /peramban nyata.	Keterdengaran, tumpang tindih, dan pemulihan keluaran belum terbukti.	Jalankan rencana uji <i>speaker</i> , mute, beberapa <i>tab</i> , <i>restart</i> , dan penggunaan panjang.	Memerlukan validasi lapangan

No.	Kendala	Penyebab	Dampak	Penanganan	Status/Tindak Lanjut
7	Perubahan kebutuhan dan kontrak	Antarmuka serta aturan lintas komponen berubah selama pengembangan.	Dokumentasi dan pengujian perlu diselaraskan berulang kali.	Perubahan dicatat pada <i>changelog</i> dan skenario pengujian diperbarui.	Selesai dengan catatan
8	Penerapan lapangan	<i>Scheduler</i> HMI memerlukan aplikasi/peramban aktif dan prototipe belum diterapkan penuh.	Operasi tanpa HMI aktif dan kondisi lapangan belum tervalidasi.	Kembangkan mekanisme layanan latar dan lakukan uji operasional terkontrol.	Memerlukan pengembangan lanjutan
9	Validasi pengguna	Demonstrasi hanya tercatat pada <i>logbook draft</i> tanpa persetujuan formal.	Kesesuaian alur dan manfaat bagi pengguna belum dapat disimpulkan.	Lakukan sesi validasi dengan skenario, komentar, keputusan, dan bukti persetujuan.	Memerlukan validasi lapangan
10	Waktu dan kelengkapan pengujian	Artefak hasil Flutter, APK final, dan penandatanganan produksi belum tersedia.	Cakupan hasil yang dapat dinyatakan masih terbatas.	Selesaikan <i>build</i> final, arsipkan keluaran uji, dan ulangi pengujian end-to-end.	Memerlukan pengembangan lanjutan

Prioritas tindak lanjut adalah menyelesaikan *hosting web admin*, melakukan *setup* dan konfigurasi *PC Control* pada dua pelabuhan, serta menerapkan Ro-Ro Alert di lapangan. Pengujian integrasi perlu diulang pada laptop dan perangkat Android yang telah didokumentasikan, jaringan LAN dan 4G, *speaker*/peramban nyata, serta skenario putus-sambung. Integrasi *LED timer display* masih ditunda karena modul penghubung antara *website* dan perangkat LED belum tersedia.

Dokumentasi visual pada bagian implementasi telah menggunakan gambar yang dimasukkan ke dalam laporan. Nama gambar dan keterangannya disesuaikan dengan fungsi yang terlihat pada setiap tangkapan layar.

BAB V

PENUTUP

5.1 Kesimpulan

1. Pelaksanaan magang di Dinas Perhubungan Kabupaten Bengkalis, khususnya pada Bidang Pelayaran, memberikan pengalaman kerja langsung kepada penulis dalam memahami lingkungan kerja instansi, mengamati kegiatan operasional pelabuhan, dan mendukung kegiatan operasional TRIPS. Kegiatan tersebut membantu penulis mengenali hubungan antara kebutuhan petugas, alur pencatatan, ketersediaan informasi, dan kondisi jaringan. Pengalaman tersebut juga menjadi ruang penerapan kompetensi Rekayasa Perangkat Lunak melalui penyusunan kebutuhan, perancangan, pengembangan, pengujian, dan dokumentasi Ro-Ro Alert.
2. Pemahaman terhadap alur kerja dan kebutuhan informasi diperoleh melalui pengamatan kegiatan serta penjelasan awal mengenai alur sistem yang dibutuhkan. Dari proses tersebut, penulis mengidentifikasi kebutuhan pencatatan ATA, ATD, penghitungan TRT, jeda operasi, kendala, foto, lokasi, status koneksi, serta pemantauan data operasional. Hasil analisis digunakan untuk menyusun alur yang lebih terstruktur dan menjadi dasar pengembangan prototipe. Dengan demikian, tujuan untuk memahami proses operasional dan menganalisis kebutuhan pencatatan serta pemantauan dapat dinyatakan tercapai dalam ruang lingkup kegiatan magang.
3. Proses analisis menghasilkan rancangan Ro-Ro Alert yang menghubungkan aplikasi Flutter, PC Lokal/HMI, dan Cloud Laravel. Rancangan yang tersedia mencakup arsitektur sistem, alur sistem, use case, activity diagram, rancangan basis data, serta mekanisme integrasi dan sinkronisasi. Pembagian fungsi ketiga komponen disusun agar pencatatan lapangan, pengelolaan operasi lokal, dan pemantauan terpusat berada dalam satu rangkaian proses. Hasil ini menunjukkan bahwa kebutuhan yang telah

diidentifikasi dapat diterjemahkan menjadi rancangan teknis yang dapat digunakan sebagai dasar implementasi dan evaluasi lanjutan.

4. Pada tahap implementasi, penulis secara mandiri mengembangkan prototipe ketiga komponen utama sesuai ruang lingkup yang direncanakan. Cloud Laravel menyediakan autentikasi, pengelolaan data master, dashboard, laporan KPI, penyimpanan data terpusat, kendali jarak jauh, dan layanan integrasi. PC Lokal/HMI mendukung pengelolaan operasi lokal, pencatatan ATA dan ATD, penghitungan TRT, pause dan resume, pencatatan kendala, audio peringatan, serta penyimpanan dan sinkronisasi lokal. Aplikasi Flutter mendukung autentikasi petugas, pencatatan operasional, ATA dan ATD, pelaporan kendala, pengambilan foto dan lokasi, penyimpanan lokal, komunikasi melalui LAN, serta jalur fallback 4G. Hasil tersebut menunjukkan tercapainya tujuan perancangan dan pengembangan, tetapi belum menjadi bukti kesiapan seluruh fungsi untuk lingkungan produksi.
5. Integrasi dikembangkan dengan jalur utama dari aplikasi Flutter menuju PC Lokal/HMI melalui LAN, dilanjutkan sinkronisasi dari PC Lokal/HMI ke Cloud Laravel, serta jalur fallback dari aplikasi Flutter ke Cloud Laravel melalui 4G ketika PC Lokal/HMI tidak dapat dijangkau. Mekanisme rekonsiliasi, UUID, idempotensi, acknowledgement, retry, dan klasifikasi status disiapkan untuk membantu menjaga ketertelusuran data dan mengurangi pengiriman ulang yang tidak diperlukan. Alur tersebut didukung oleh log, changelog, dan pengujian internal, tetapi belum dapat dinyatakan selalu berhasil karena masih terdapat kegagalan heartbeat, unggah foto, serta belum adanya bukti catch-up dengan batch berisi data.
6. Pengujian internal memberikan bukti bahwa 17 skenario helper sinkronisasi PC Lokal/HMI dan 53 skenario mesin audio lulus. Pengujian mesin audio pada tahap internal menggunakan MockAudio. Berdasarkan informasi tambahan dari pelaksanaan akhir, audio juga telah diuji dan diterapkan pada perangkat yang digunakan tanpa bug yang ditemukan selama pengujian tersebut. Hasil pengujian fisik itu tetap perlu dilengkapi dengan catatan perangkat, peramban, konfigurasi audio, tanggal, skenario, dan bukti formal

agar dapat ditelusuri. Pada Cloud Laravel, 9 pengujian lulus dan 6 pengujian gagal karena ketidaksesuaian skema SQLite uji pada kolom lokasi_pelabuhan. Log juga mencatat alur ATA, ATD, pause, resume, dan fallback, sekaligus menunjukkan respons HTTP 404, timeout, HTTP 500 pada heartbeat dan unggah foto, serta siklus catch-up yang masih berisi 0 item. Oleh karena itu, tujuan pengujian internal tercapai sebagian dan masih memerlukan penyelesaian temuan yang tercatat.

7. Dokumentasi teknis dan laporan magang telah disusun sebagai dasar evaluasi serta pengembangan berikutnya. Pada tingkat prototipe, Ro-Ro Alert menyediakan mekanisme untuk menyatukan pencatatan operasional, menggunakan jalur LAN dan fallback, memantau data secara terpusat, mencatat kendala, menjalankan sinkronisasi, dan menyediakan jejak teknis melalui log serta changelog. Presentasi akhir telah dilakukan bersama Kabid dan pegawai Bidang Pelayaran, dan aplikasi berjalan normal pada alur yang diperagakan dalam sesi tersebut. Hasil ini mendukung fungsi prototipe pada demonstrasi, tetapi belum dapat disetarakan dengan validasi pengguna formal, penerapan penuh, atau serah terima. Potensi sistem juga belum dapat dinyatakan sebagai peningkatan efisiensi atau dampak layanan karena belum ada pengukuran sebelum dan sesudah implementasi. Serah terima belum dilakukan karena web admin belum di-hosting, Ro-Ro Alert belum digunakan langsung di lapangan, dan PC Control belum selesai disiapkan pada dua pelabuhan. Serah terima direncanakan setelah ketiga tahap tersebut selesai dan hasil penerapannya diperiksa. Integrasi LED timer display juga masih menunggu modul penghubung. Selain itu, laporan formal pengujian Flutter dan APK final, tangkapan layar versi final, pengesahan logbook, serta bukti serah terima belum tersedia. Dengan keterbatasan tersebut, hasil magang tetap membentuk rancangan dan prototipe terintegrasi yang memerlukan validasi lapangan serta pengembangan lanjutan.

5.2 Saran

Berdasarkan pelaksanaan magang dan evaluasi hasil pengembangan Ro-Ro Alert, saran disusun untuk mahasiswa yang akan melaksanakan magang di Dinas Perhubungan Kabupaten Bengkalis, pihak instansi, dan Politeknik Negeri Bengkalis. Bagian ini juga memuat pencapaian selama magang serta pekerjaan yang masih perlu diselesaikan agar batas hasil pengembangan dapat dipahami secara jelas.

1. Rekomendasi bagi Mahasiswa yang Akan Melaksanakan Magang di Dinas Perhubungan Kabupaten Bengkalis

Dinas Perhubungan Kabupaten Bengkalis dapat dipertimbangkan sebagai tempat magang bagi mahasiswa Program Studi Rekayasa Perangkat Lunak apabila pada periode tersebut terdapat kebutuhan pengembangan sistem informasi atau proyek perangkat lunak yang jelas. Sebelum memulai magang, mahasiswa disarankan mengonfirmasi ruang lingkup proyek, calon pengguna, data yang dapat digunakan, perangkat yang tersedia, target penyelesaian, dan jadwal validasi bersama pembimbing lapangan. Karena tugas utama instansi berfokus pada pelayanan perhubungan, ketersediaan proyek yang sesuai dengan kompetensi Rekayasa Perangkat Lunak dapat berbeda pada setiap periode. Apabila tidak terdapat proyek yang relevan, mahasiswa sebaiknya mempertimbangkan lokasi magang lain agar kegiatan yang diperoleh tetap sesuai dengan bidang keilmuan. Mahasiswa yang memilih Dinas Perhubungan juga perlu siap bekerja mandiri, aktif berkomunikasi dengan pembimbing lapangan dan calon pengguna, serta mencatat perubahan kebutuhan sejak awal. Logbook, hasil pengujian, versi aplikasi, tangkapan layar final, dan bukti kegiatan sebaiknya diarsipkan secara berkala agar penyusunan laporan dan proses serah terima tidak terkendala pada akhir kegiatan.

2. Pencapaian Selama Magang

Selama magang, penulis memperoleh pengalaman dalam kegiatan operasional TRIPS dan memahami kebutuhan pencatatan serta pemantauan operasional kapal Ro-Ro. Berdasarkan alur awal yang dijelaskan oleh Kepala Bidang Pelayaran, penulis menyusun flowchart dan secara mandiri menghasilkan arsitektur sistem, use case, activity diagram, serta rancangan basis data yang telah

diverifikasi dan disetujui oleh Kepala Bidang Pelayaran. Penulis juga secara mandiri mengembangkan prototipe roro-alert-web berbasis Laravel, roro-alert-client sebagai PC Lokal/HMI, dan roro-alert-flutter sebagai aplikasi petugas lapangan. Fungsi yang telah tersedia meliputi autentikasi, pengelolaan data master, pencatatan ATA dan ATD, penghitungan TRT, pause dan resume, pelaporan kendala dengan foto dan lokasi, dashboard pemantauan, laporan, sinkronisasi melalui LAN, jalur fallback 4G, serta peringatan audio. Pengujian internal mencatat 17 skenario helper sinkronisasi dan 53 skenario mesin audio lulus. Audio juga telah diuji pada perangkat pelaksanaan akhir dan dapat digunakan tanpa bug yang ditemukan selama pengujian tersebut. Presentasi akhir telah dilaksanakan bersama Kepala Bidang dan pegawai Bidang Pelayaran, dan aplikasi berjalan normal pada alur yang diperagakan dalam sesi tersebut.

3. Bagian yang Belum Tercapai dan Perlu Ditindaklanjuti

Ro-Ro Alert belum mencapai tahap penerapan penuh dan serah terima. Web admin roro-alert-web belum di-hosting, PC Control belum disiapkan pada dua pelabuhan, dan sistem belum digunakan secara langsung dalam operasional lapangan. Oleh karena itu, serah terima belum dilakukan dan perlu menunggu penyelesaian ketiga tahap tersebut beserta pemeriksaan hasil penerapannya. Integrasi LED timer display juga masih ditunda karena modul penghubung antara website dan perangkat LED belum tersedia. Selain itu, pengujian penerimaan pengguna dan pengujian lapangan yang terdokumentasi masih perlu dilakukan, termasuk pengujian ulang terhadap skenario Cloud yang gagal akibat ketidaksesuaian skema pengujian, heartbeat, unggah foto, catch-up dengan batch berisi data, perpindahan LAN dan 4G, serta kompatibilitas pada perangkat Android yang digunakan. Bagian tersebut perlu diselesaikan sebelum sistem dinyatakan siap digunakan secara operasional.

4. Saran bagi Dinas Perhubungan Kabupaten Bengkalis

Dinas Perhubungan Kabupaten Bengkalis disarankan melanjutkan penerapan Ro-Ro Alert secara bertahap melalui penyelesaian hosting web admin, konfigurasi PC Control pada dua pelabuhan, simulasi operasional, dan uji penerimaan pengguna yang terdokumentasi. Instansi juga perlu menetapkan pihak

yang bertanggung jawab terhadap konfigurasi, pencadangan data, keamanan akses, pemeliharaan, dan pencatatan perubahan sistem. Hasil uji terbatas sebaiknya menjadi dasar perbaikan sebelum sistem digunakan lebih luas, sedangkan integrasi LED timer display dilakukan setelah modul penghubung tersedia dan telah diuji.

5. Saran bagi Politeknik Negeri Bengkalis

Politeknik Negeri Bengkalis disarankan terus menjalin kerja sama dengan instansi pemerintah dan dunia industri sebagai tempat magang, serta membantu memastikan kesesuaian antara lokasi penempatan, proyek yang tersedia, dan kompetensi program studi. Pemantauan berkala terhadap ruang lingkup pekerjaan, bukti kegiatan, hasil pengujian, dan artefak akhir dapat membantu mahasiswa menyelesaikan proyek dan laporan dengan data yang lebih lengkap serta dapat diverifikasi.

DAFTAR PUSTAKA

- Dinas Komunikasi, Informatika dan Statistik Kabupaten Bengkalis. 2025. “Hingga Senin Pagi, Sekitar 28,42 Persen atau Sebanyak 682 Tiket Sudah Dipesan.” 24 Maret 2025. [Portal Diskominfo Kabupaten Bengkalis](#). Diakses 18 Juli 2026.
- Dinas Perhubungan Kabupaten Bengkalis. 2021. *Indikator Kinerja Utama Dinas Perhubungan Kabupaten Bengkalis Tahun 2021*. Bengkalis: Dinas Perhubungan Kabupaten Bengkalis. [Dokumen IKU Dinas Perhubungan \(PDF\)](#). Diakses 18 Juli 2026.
- Dinas Perhubungan Kabupaten Bengkalis. 2024. *Perjanjian Kinerja Tahun 2024*. Bengkalis: Dinas Perhubungan Kabupaten Bengkalis. [Dokumen Perjanjian Kinerja \(PDF\)](#). Diakses 18 Juli 2026.
- Pemerintah Kabupaten Bengkalis. 2021. *Rencana Pembangunan Jangka Menengah Daerah Kabupaten Bengkalis Tahun 2021–2026*. Bengkalis: Badan Perencanaan Pembangunan Daerah Kabupaten Bengkalis. [Dokumen RPJMD \(PDF\)](#). Diakses 18 Juli 2026.
- Pemerintah Kabupaten Bengkalis. 2024. Peraturan Bupati Bengkalis Nomor 44 Tahun 2024 tentang Perubahan atas Peraturan Bupati Bengkalis Nomor 99 Tahun 2021 tentang Kedudukan, Susunan Organisasi, Tugas dan Fungsi serta Tata Kerja Dinas Perhubungan Kabupaten Bengkalis. Bengkalis: Pemerintah Kabupaten Bengkalis. Tersedia pada: [JDIH Kabupaten Bengkalis](#). Diakses 18 Juli 2026.
- Pemerintah Kabupaten Bengkalis. 2026. “Sebelum Mudik Lebaran, Kendaraan Roda Empat Harus Siapkan Tiket Lewat Aplikasi TRIPS Bengkalis.” 9 Maret 2026. [Portal Pemerintah Kabupaten Bengkalis](#). Diakses 18 Juli 2026.
- PPID Kabupaten Bengkalis. n.d. “Dinas/Badan.” [Portal PPID Kabupaten Bengkalis](#). Diakses 18 Juli 2026.
- Republik Indonesia. 2008. *Undang-Undang Nomor 17 Tahun 2008 tentang Pelayaran*, sebagaimana diubah terakhir dengan Undang-Undang Nomor 66 Tahun 2024. Lembaran Negara Republik Indonesia Tahun 2008 Nomor 64. [Database Peraturan BPK](#). Diakses 18 Juli 2026.
- Republik Indonesia. 2018. *Peraturan Presiden Nomor 95 Tahun 2018 tentang Sistem Pemerintahan Berbasis Elektronik*. Lembaran Negara Republik Indonesia Tahun 2018 Nomor 182. [Database Peraturan BPK](#). Diakses 18 Juli 2026.

LAMPIRAN

Lampiran 1. Surat Keterangan Selesai Magang

SURAT KETERANGAN

Nomer: 500.11.18.01 / Dishub - P1g / 11 / 2026

Yang bertanda tangan di bawah ini menerangkan bahwa :

Nama : Riski Afendi
Tempat/ Tgl. Lahir : Teluk Ketapang / 27 Maret 2003
Alamat : JL. Jendral Sudirman

Telah melakukan Magang pada perusahaan kami, Dinas Perhubungan Kabupaten Bengkalis sejak tanggal 10 maret sampai dengan 28 juni 2026 sebagai tenaga Magang.

Selama bekerja di perusahaan kami, yang bersangkutan telah menunjukkan ketekunan dan kesungguhan bekerja dengan baik.

Surat keterangan ini diberikan untuk dipergunakan sebagaimana mestinya.

Demikian agar yang berkepentingan maklum.

Bengkalis, 29 Juni 2026
Kepala Bidang Pelayaran
Dishub Kabupaten Bengkalis



Edi Kurniawan, ST. M.T
NIP: 198206182015031001

Lampiran 2. Surat Penerimaan Magang



PEMERINTAH KABUPATEN BENGKALIS **DINAS PERHUBUNGAN**

Jalan Pramuka Nomor 050, Desa Senggoro, Bengkalis, Riau 28714, Telepon (0766) 21865
Laman dishub.bengkalis.go.id, Pos-el dishub@bengkalis.go.id, dishub.bengkalis@gmail.com

Bengkalis, 9 Maret 2026

Nomor : 000.1/Dishub-Set/III/2026
Sifat : Biasa
Lampiran : 1 Berkas
Hal : **Persetujuan Kerja Praktik (KP) Pada Dinas Dinas Perhubungan
Kabupaten Bengkalis**

Kepada Yth. Direktur Politeknik Negeri Bengkalis Kabupaten Bengkalis
di-
Bengkalis

Menindaklanjuti Surat Kementerian Pendidikan Tinggi, Sains Dan Teknologi Politeknik Negeri Bengkalis Kabupaten Bengkalis Nomor : 1409/PL31/TU/2026 tanggal 9 Maret 2026, Hal Permohonan Kerja Praktik (KP), dengan ini disampaikan sebagai berikut.

1. Pada perinsipnya dapat disetujui untuk melaksanakan Kerja Praktik (KP) bagi peserta Kerja Praktik (KP) pada Dinas Perhubungan Kabupaten Bengkalis, adapun Peserta Kerja Praktik (KP) sebagai berikut :

N0	NAMA	NIM	PRODI
1.	M. FIKRI SETIAWAN	6304221487	D-IV Rekayasa Perangkat Lunak
2.	RISKI AFENDI	6304221510	D-IV Rekayasa Perangkat Lunak
3.	RAMA OTARI	6304221462	D-IV Rekayasa Perangkat Lunak

2. Selama pelaksanaan kegiatan Kerja Praktik (KP) diharapkan yang bersangkutan mematuhi ketentuan/tertib yang berlaku di Dinas Perhubungan Kabupaten Bengkalis selama 4 (empat) bulan.

Demikian disampaikan, untuk dapat dimaklumi dan dipergunakan sebagaimana mestinya.


Kepala Dinas Perhubungan
Kabupaten Bengkalis
Sekretaris
AL HAMIDI, SH.,M.IP
Pembina
NIP. 19770210 201102 1 001

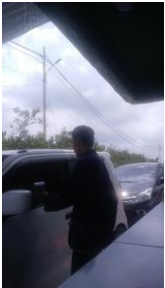
Lampiran 3. Logbook yang Telah Disahkan

Lampiran ini memuat rekap kegiatan harian selama empat bulan yang terdiri atas 81 entri pada periode 10 Maret sampai 29 Juni 2026. Struktur tabel menggunakan kolom No., Hari/Tanggal, Kegiatan, Dokumentasi Kegiatan, dan Keterangan dengan mengacu pada format lampiran logbook laporan kerja praktek terdahulu.

Rekap Kegiatan Maret 2026 (18 entri)

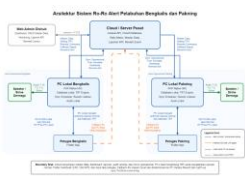
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
1	Selasa, 10 Maret 2026	Pra-Posko Penyerahan dokumen dan orientasi penempatan	-	Menyerahkan surat permohonan kerja praktik kepada Dinas Perhubungan Kabupaten Bengkalis dan berkoordinasi dengan Kepala Bidang mengenai ruang lingkup penugasan. Berdasarkan arahan tersebut, penempatan awal dipusatkan di Posko Pelabuhan Ro-Ro untuk mendukung uji coba operasional platform TRIPS secara langsung di lapangan.
2	Rabu, 11 Maret 2026	Operasional Posko Pengaturan jalur kendaraan pengguna TRIPS		Mengatur pemilahan arus kendaraan di area masuk pelabuhan dengan mengarahkan kendaraan roda empat menuju jalur pemesanan online TRIPS, sementara kendaraan roda dua tetap menggunakan loket offline. Pemisahan tersebut menjaga pergerakan kendaraan tetap teratur dan mengurangi potensi penumpukan pada gerbang masuk.
3	Kamis, 12 Maret 2026	Operasional Posko Kolaborasi pengendalian antrean		Berkolaborasi dengan petugas Dinas Perhubungan dan Satpol PP untuk mengendalikan antrean mobil, sepeda motor, becak, serta gerobak di sekitar posko. Pada saat yang sama, memberikan panduan pemindaian QR Code dan akses tiket digital TRIPS agar pengguna dapat mengikuti jalur pelayanan yang sesuai.

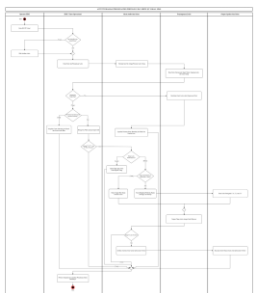
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
4	Jumat, 13 Maret 2026	Operasional Posko Pendampingan pembayaran QRIS		Memberikan pendampingan langsung kepada pengguna yang mengalami kendala pada proses pembayaran non-tunai melalui QRIS. Penjelasan singkat mengenai tahapan transaksi diberikan dari kendaraan ke kendaraan sehingga penyelesaian pembayaran dapat dilakukan tanpa menahan antrean di belakangnya.
5	Sabtu, 14 Maret 2026	Operasional Posko Pelayanan boarding pass melalui EDC BRKS		Mempercepat pelayanan boarding pass dengan membawa perangkat Electronic Data Capture (EDC) BRKS mendekati kendaraan sebagai mobile point of service. Barcode pembayaran dipindai langsung dari sisi pengemudi, kemudian hasil validasi digunakan untuk melanjutkan penerbitan tiket tanpa memindahkan kendaraan dari antrean.
6	Minggu, 15 Maret 2026	Operasional Posko Pengendalian antrean Posko 1 dan Posko 2		Mengawasi urutan kendaraan pada Posko 1 dan Posko 2 untuk mencegah penyerobotan antrean serta menjaga pemisahan layanan antara sepeda motor dan mobil. Pengendalian dilakukan berdasarkan urutan kedatangan, sementara layanan mobile tetap diarahkan kepada kendaraan roda empat yang telah menggunakan TRIPS.
7	Senin, 16 Maret 2026	Operasional Posko Validasi tiket digital dan distribusi boarding pass		Mengoperasikan proses validasi keberangkatan di loket kedua dengan memindai tiket dari ponsel atau cetakan pengguna melalui kamera sensor. Setelah validasi berhasil, menyerahkan boarding pass, tiket fisik, dan karcis antrean berbasis pembagian jam, sedangkan kendaraan roda dua tetap diarahkan ke layanan offline.
8	Selasa, 17 Maret 2026	Operasional Posko Pengamanan arus mudik dan pemisahan jalur		Menjaga pemisahan antara kendaraan yang telah melakukan reservasi melalui TRIPS dan kendaraan dengan transaksi konvensional pada periode peningkatan arus mudik. Pergerakan di Posko 1 dan Posko 2 diatur agar antrean tidak meluas ke jalan utama serta tidak menghambat akses menuju area pelabuhan.

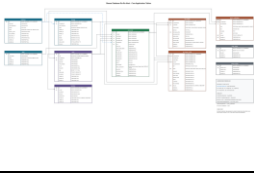
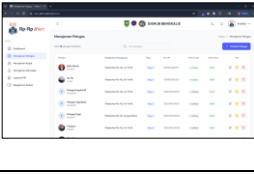
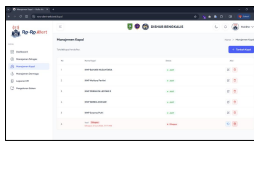
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
9	Rabu, 18 Maret 2026	Operasional Posko Pelayanan puncak arus mudik menjelang Idulfitri		Mengelola pelayanan lapangan ketika volume kendaraan memasuki fase puncak menjelang Idulfitri 1447 H. Pada kondisi tersebut, penggunaan TRIPS dihentikan sementara sehingga pemeriksaan tiket, pengaturan antrean, dan transaksi kendaraan dilaksanakan melalui mekanisme fisik serta pengawasan manual.
10	Senin, 23 Maret 2026	Izin Izin perbaikan kendaraan pribadi	-	Mengajukan izin tidak mengikuti kegiatan lapangan karena kerusakan pada bagian rangka sepeda motor yang digunakan sebagai transportasi menuju lokasi magang. Perbaikan diselesaikan pada hari yang sama agar penugasan operasional dapat dilanjutkan kembali pada jadwal berikutnya.
11	Selasa, 24 Maret 2026	Operasional Posko Penataan arus balik pasca-Idulfitri		Kembali melaksanakan pengawasan di posko pada periode arus balik dengan menata pergerakan kendaraan setelah libur Idulfitri. Selain menjaga urutan antrean, memastikan kendaraan roda empat tetap mengikuti mekanisme tiket digital TRIPS yang masih diterapkan sampai akhir Maret.
12	Rabu, 25 Maret 2026	Operasional Posko Pendampingan kendala antarmuka TRIPS		Mendampingi pengguna yang mengalami kebingungan saat memeriksa ketersediaan tiket karena antarmuka TRIPS belum menampilkan informasi sisa kuota secara langsung. Panduan diberikan pada tahap pemilihan jadwal dan pemesanan agar keterbatasan antarmuka tidak memperlambat layanan di posko.
13	Kamis, 26 Maret 2026	Operasional Posko Pendampingan kendala konektivitas		Membantu pengguna yang proses pemesanan tiketnya tertunda akibat koneksi seluler tidak stabil dan area blank spot di sekitar pelabuhan. Pendampingan difokuskan pada pemeriksaan ulang koneksi serta pengulangan proses pemesanan sehingga kendala jaringan tidak berkembang menjadi antrean baru.

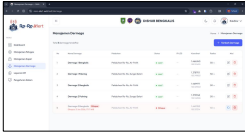
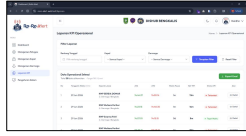
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
14	Jumat, 27 Maret 2026	Operasional Posko Optimalisasi layanan Posko 1 dan Posko 2		Mengoptimalkan pemeriksaan tiket pada Posko 1 dan Posko 2 melalui kombinasi layanan loket untuk sepeda motor dan layanan mobile bagi kendaraan roda empat. Pembagian tugas tersebut menjaga antrean tetap bergerak sekaligus memastikan setiap kendaraan memasuki jalur pemeriksaan yang sesuai.
15	Sabtu, 28 Maret 2026	Operasional Posko Antisipasi peningkatan volume akhir pekan		Mengantisipasi peningkatan volume kendaraan akhir pekan dengan mempercepat validasi aplikasi bagi kendaraan roda empat dan mempertahankan pemeriksaan manual untuk roda dua. Arahan jalur diberikan sejak area masuk agar perbedaan mekanisme pelayanan tidak menimbulkan persilangan antrean.
16	Minggu, 29 Maret 2026	Operasional Posko Pengawasan akhir uji coba TRIPS		Mengawal operasional posko pada tahap akhir uji coba TRIPS dengan memantau kelancaran pemesanan, pembayaran, validasi tiket, dan pergerakan kendaraan. Temuan lapangan mengenai kendala antarmuka, jaringan, serta pola antrean dicatat sebagai bahan evaluasi untuk penyempurnaan sistem berikutnya.
17	Senin, 30 Maret 2026	Transisi Sistem Peralihan operasional setelah penghentian sementara TRIPS		Mendukung peralihan pelayanan setelah TRIPS dihentikan sementara untuk pembenahan fungsi server. Pengaturan kendaraan dikembalikan secara bertahap pada mekanisme manual, sementara alur pemeriksaan tetap dijaga agar perubahan sistem tidak mengganggu keteraturan posko.
18	Selasa, 31 Maret 2026	Operasional Posko Penutupan operasional lapangan		Menyelesaikan penugasan operasional di pelabuhan seiring berakhirnya kegiatan posko dan mengembalikan pelayanan kendaraan pada transaksi di tempat. Penutupan ini sekaligus menandai selesainya pengumpulan pengalaman lapangan mengenai penggunaan TRIPS, pola antrean, dan kendala pelayanan pengguna.

Rekap Kegiatan April 2026 (21 entri)

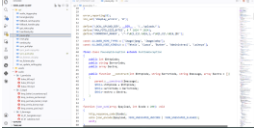
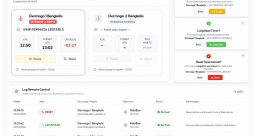
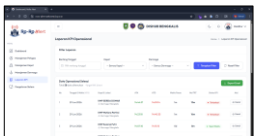
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
19	Rabu, 1 April 2026	Pasca-Posko Pelaporan kembali dan rekap temuan lapangan	-	Melapor kembali ke kantor Dinas Perhubungan setelah penugasan lapangan selesai dan mulai menyusun rekap temuan operasional. Catatan difokuskan pada kendala pembayaran, jaringan, antarmuka, pemisahan jalur, serta kebutuhan petugas selama puncak arus mudik dan arus balik.
20	Kamis, 2 April 2026	Orientasi Divisi Orientasi proses layanan internal Dishub	-	Mengikuti orientasi proses kerja internal pada bidang pelayanan untuk memetakan alur administrasi dan koordinasi operasional pelabuhan. Hasil pengamatan digunakan untuk mengenali bagian layanan yang masih bergantung pada pencatatan manual dan berpotensi didukung melalui sistem monitoring.
21	Senin, 6 April 2026	Inisiasi Proyek Kick-off Ro-Ro Alert dan pengumpulan kebutuhan		Memulai proyek Ro-Ro Alert melalui diskusi awal bersama Kepala Bidang untuk merumuskan kebutuhan sistem berdasarkan pengalaman operasional TRIPS dan tata kerja pelabuhan. Pembahasan mencakup kebutuhan monitoring ATA/ATD, kendala, target waktu, peringatan, sinkronisasi data, serta pembagian peran antara aplikasi lapangan, PC lokal, dan web admin.
22	Selasa, 7 April 2026	Pra-code Penyusunan flowchart dan arsitektur sistem tiga pilar Ro-Ro Alert		Menggabungkan flowchart proses operasional dan arsitektur sistem tiga pilar untuk memetakan alur ATA/ATD, Kendala/Pause-Resume, sinkronisasi Cloud, fallback 4G, dashboard monitoring, serta batas tanggung jawab data antara roro-alert-flutter, roro-alert-client, dan roro-alert-web.

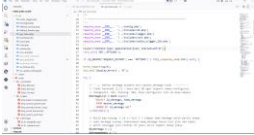
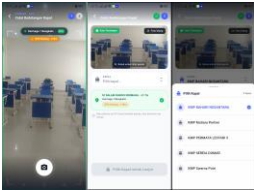
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
23	Rabu, 8 April 2026	Pra-code Presentasi flowchart dan arsitektur sistem Ro-Ro Alert		Mempresentasikan flowchart dan arsitektur tiga pilar Ro-Ro Alert bersama Kepala Bidang untuk menjelaskan alur ATA/ATD, Kendala/Pause-Resume, sinkronisasi Cloud, fallback 4G, dashboard monitoring, serta pembagian tanggung jawab antara aplikasi Flutter, PC lokal/HMI, dan web admin. Masukan dari presentasi diterapkan untuk merapikan hubungan antarproses sebelum penyusunan use case.
24	Kamis, 9 April 2026	Pra-code Penyusunan presentasi use case diagram		Menyusun use case diagram berdasarkan aktor petugas lapangan, PC lokal/HMI, admin dashboard, dan sistem Cloud, kemudian mempresentasikannya bersama Kepala Bidang. Pembahasan difokuskan pada batas akses, skenario ATA/ATD, Kendala/Pause-Resume, sinkronisasi, fallback 4G, serta keterkaitan setiap aktor dengan proses operasional pelabuhan.
25	Jumat, 10 April 2026	Pra-code Penyempurnaan activity diagram Kendala/Pause-Resume dan status operasional		Menyempurnakan activity diagram Kendala/Pause-Resume dengan pemetaan validasi foto dan GPS, perubahan status paused/running, pergeseran target ATD, pencatatan durasi kendala, serta pengaruhnya terhadap net TRT dan status operasional.
26	Senin, 13 April 2026	Pra-code Presentasi final alur sistem dan activity diagram		Menyelesaikan activity diagram sinkronisasi PC lokal, fallback 4G, rekonsiliasi Reverb/catch-up, monitoring dashboard, dan remote control, kemudian melakukan presentasi final alur sistem bersama Kepala Bidang. Validasi akhir tersebut menyatukan flowchart, arsitektur, use case, dan activity diagram sebagai acuan implementasi tiga pilar Ro-Ro Alert.
27	Selasa, 14 April 2026	Database Cloud Fondasi users dan pelabuhan		Membuat migration awal Cloud Laravel untuk tabel 'users' dan 'pelabuhan', termasuk role admin/petugas, 'id pelabuhan', 'kode_port', 'device_id aktif', heartbeat, dan data koneksi PC lokal.

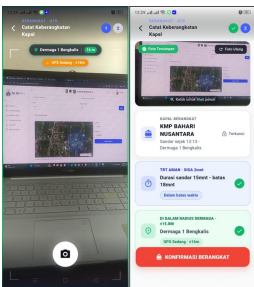
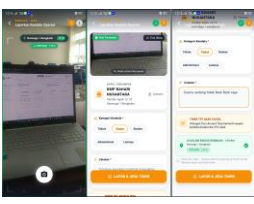
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
28	Rabu, 15 April 2026	Database Cloud Schema operasional inti		Mengembangkan schema 'kapal', 'dermaga', 'operasional', 'log_kendala', 'audio_setting', 'system_setting', 'blacklists', dan token Sanctum agar 3 pilar memiliki tabel pusat untuk ATA, ATD, foto, koordinat, sync, serta kendala.
29	Kamis, 16 April 2026	Database Cloud Model dan relasi data		Membuat model Eloquent 'Operasional', 'LogKendala', 'Dermaga', 'Kapal', 'Pelabuhan', 'User', dan setting pendukung dengan relasi 'belongsTo', 'hasMany', serta accessor foto profil untuk kebutuhan dashboard dan API.
30	Jumat, 17 April 2026	Database Cloud Seeder dan presentasi schema database		Menyiapkan seeder admin, petugas, kapal, dermaga, target TRT, dan slot audio T-10/T-5/T-0/Siren agar environment demo Cloud dapat langsung digunakan. Setelah schema, model, relasi, dan data awal tersusun, mempresentasikan rancangan database kepada Kepala Bidang untuk memastikan struktur data mendukung kebutuhan monitoring serta audit operasional.
31	Senin, 20 April 2026	Auth Admin Login, logout, dan keamanan		Mengonfigurasi autentikasi web admin melalui 'AuthController', logout session, update 'status_login', 'AdminMiddleware', dan 'AntiXssMiddleware' agar dashboard hanya diakses admin serta input berbahaya tercatat ke blacklist.
32	Selasa, 21 April 2026	Master Data CRUD petugas		Mengembangkan modul CRUD petugas dengan validasi username, role, regu, pelabuhan, foto profil, soft delete, restore, force delete, dan toggle blokir akun untuk kontrol operator lapangan.
33	Rabu, 22 April 2026	Master Data CRUD kapal dan pelabuhan		Mengembangkan CRUD kapal dan pelabuhan dengan status aktif/maintenance, 'kode_port', relasi jumlah dermaga/petugas, soft delete, serta guard agar histori operasional tidak terhapus permanen.

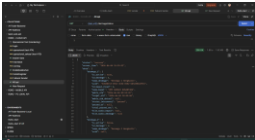
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
34	Kamis, 23 April 2026	Master Data CRUD dermaga dan geofence		Membuat CRUD dermaga dengan `id_pelabuhan`, koordinat latitude/longitude, radius meter, IP LED, status perbaikan, dan `last_sync` sebagai dasar geofence serta monitoring PC lokal.
35	Jumat, 24 April 2026	API Contract & Rapat Target TRT, master data, dan rapat studi tiru BPTD Riau	 	Menyesuaikan pengaturan target_trt_default melalui SystemSettingController, TrtParser, cache master-data, dan event SettingUpdated agar nilai TRT konsisten pada dashboard, PC lokal, dan Flutter. Pada hari yang sama, mengikuti rapat serta studi tiru BPTD Riau di Dinas Perhubungan Kabupaten Bengkalis mengenai penerapan e-ticketing TRIPS selama arus mudik dan arus balik, termasuk pembahasan alur layanan serta temuan operasional di Pelabuhan Air Putih dan Sei Selari.
36	Senin, 27 April 2026	Dashboard Monitoring Dashboard pelabuhan dan dermaga		Mengembangkan dashboard monitoring berbasis pelabuhan dengan card dermaga, status running/paused, kapal aktif, target ATD, indikator PC online, dan ringkasan KPI harian.
37	Selasa, 28 April 2026	Dashboard Monitoring KPI dan live activity		Mengintegrasikan KPI `total_sandar`, on-time, overdue, rata-rata Net TRT, live activity kendala/ATD, serta konversi waktu WIB agar data operasional tampil objektif pada web admin.
38	Rabu, 29 April 2026	Sync Install, heartbeat, dan master-data API		Mengembangkan endpoint `GET /api/install/pelabuhan`, `POST /api/install/register`, `POST /api/heartbeat`, dan `GET /api/master-data` untuk registrasi PC lokal, status online, dan distribusi data kapal/dermaga/config.
39	Kamis, 30 April 2026	Sync Sinkronisasi operasional dan foto API		Mengembangkan `POST /api/operasional/sync` dan `POST /api/operasional/sync-foto` dengan UUID idempoten, guard dermaga/kapal aktif, update `last_sync`, serta upload foto ATA/ATD ke storage Cloud.

Rekap Kegiatan Mei 2026 (21 entri)

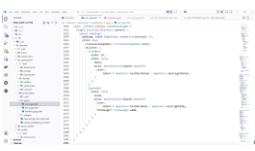
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
40	Jumat, 1 Mei 2026	Sync API Fallback 4G ATA/ATD		Mengembangkan endpoint fallback <code>'POST /api/fallback/sandar'</code> , <code>'POST /api/fallback/berangkat'</code> , <code>'GET /api/fallback/detail/{uuid}'</code> , dan <code>'GET /api/fallback/pending'</code> agar Flutter tetap dapat mengirim data saat jalur Wi-Fi PC lokal gagal.
41	Senin, 4 Mei 2026	Sync API Siklus kendala		Mengembangkan siklus kendala melalui <code>'POST /api/kendala/pause'</code> , <code>'POST /api/kendala/resume'</code> , <code>'POST /api/fallback/kendala'</code> , <code>'sync-kendala'</code> , dan <code>'sync-kendala-foto'</code> dengan <code>'durasi_pause'</code> detik serta foto bukti.
42	Selasa, 5 Mei 2026	API Penyesuaian dan response Contract payload		Menyesuaikan <code>'docs/API_CONTRACT.md'</code> untuk perubahan payload, status code, header, <code>'id_petugas_atd'</code> , pelapor kendala, <code>'DERMAGA_OCCUPIED'</code> , <code>'KAPAL_ALREADY_ACTIVE'</code> , dan pola retry final tanpa deploy production permanen.
43	Rabu, 6 Mei 2026	Remote Broadcast acknowledgement Control dan		Mengintegrasikan remote pause/resume/reset dari web admin melalui <code>'RemoteController'</code> , <code>'RemoteControlSignal'</code> , <code>'log_remote_control'</code> , dan <code>'POST /api/remote/ack'</code> agar Cloud mencatat audit trail tanpa langsung mengubah state PC lokal.
44	Kamis, 7 Mei 2026	Laporan KPI operasional dan export		Mengembangkan laporan KPI dengan filter tanggal, kapal, dermaga, scope pelabuhan, detail kendala, Petugas ATA/ATD, Pelapor, status on-time, dan export Excel untuk rekap performa operasional.
45	Jumat, 8 Mei 2026	Dashboard Monitoring Notifikasi dan Reverb		Mengintegrasikan <code>'NotificationService'</code> , <code>'system_notifications'</code> , dropdown notifikasi, toast, <code>'PcOnlineStatusChanged'</code> , <code>'OperasionalUpdated'</code> , dan status Reverb agar web admin menerima perubahan operasional secara real-time.

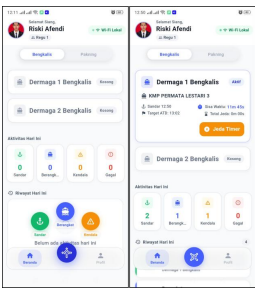
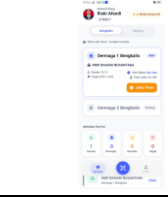
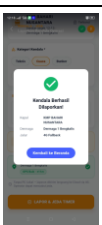
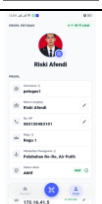
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
46	Senin, 11 Mei 2026	Testing Validasi endpoint dan demo progres web admin		Menguji route web/API, dashboard live activity, fallback kendala, konfigurasi upload foto, dan integrasi Cloud Laravel, kemudian mendemonstrasikan progres web admin bersama Kepala Bidang. Hasil demo digunakan untuk mengonfirmasi alur monitoring, master data, KPI, sinkronisasi, dan remote control sebelum integrasi dilanjutkan ke PC lokal/HMI.
47	Selasa, 12 Mei 2026	Database Lokal Schema operasional PC lokal		Menyusun schema lokal untuk 'buffer_operasional', 'master_kapal', 'master_dermaga', 'config_sistem', 'sync_heartbeat_log', dan 'log_kendala' agar PC lokal/HMI menjadi source of truth dermaga dengan UUID, ATA/ATD, pause, foto, dan status sync.
48	Rabu, 13 Mei 2026	HMI Dashboard Card dermaga dan indikator status		Membuat dashboard HMI di 'index.php' dengan card dermaga, status kapal, countdown target ATD, badge 'PAUSED'/'OVERDUE', indikator Cloud/Sync/Audio/Remote, dan jumlah pending sync.
49	Kamis, 14 Mei 2026	Endpoint Operasional Polling status real-time		Mengembangkan 'api/get_status.php' sebagai polling 1 detik yang mengirim shape dermaga aktif/kosong, statistik harian, pending sync/foto, dan flag 'audio_t10'/'audio_t5'/'audio_t0' untuk HMI.
50	Jumat, 15 Mei 2026	Endpoint Operasional Pencatatan Sandar/ATA		Mengembangkan 'api/sandar.php' untuk mencatat ATA kapal dengan validasi UUID, dermaga, kapal, petugas, GPS, foto 'foto_ata', target ATD, dan guard 'DERMAGA_OCCUPIED' serta 'KAPAL_ALREADY_ACTIVE'.

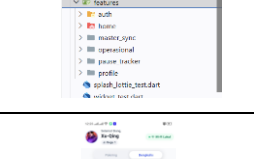
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
51	Senin, 18 Mei 2026	Upload Foto Pencatatan Berangkat/ATD		Mengembangkan `api/berangkat.php` untuk mencatat ATD aktual dengan foto `foto_atd`, koordinat, `id_petugas_atd_cloud`, perhitungan gross/net TRT, auto-resolve kendala pending, dan pembersihan guard audio.
52	Selasa, 19 Mei 2026	Kendala/Pause Pause, resume, dan reset operasi		Membuat alur `api/pause.php` dan `api/reset.php` dengan `action=pause resume`, log `log_kendala`, `total_paused_sec`, pergeseran target ATD, serta status batal agar operasi lokal tetap auditabel saat terjadi kendala.
53	Rabu, 20 Mei 2026	Audio Warning T-10/T-5/T-0 dan sirine		Mengintegrasikan `AudioEngine` untuk playlist T-10/T-5/T-0, sirine overdue, audio gate, guard `audio_trigger_state`, dan lock multi-tab agar peringatan PC lokal tidak berbunyi ganda.
54	Kamis, 21 Mei 2026	Sync Cloud Push operasional dan foto		Mengembangkan `sync.js` dan `api/sync_push.php` untuk mengirim batch operasional, foto ATA/ATD, data kendala, dan foto kendala ke Cloud dengan indikator pending, stuck, retry, serta quarantine row.
55	Jumat, 22 Mei 2026	Sync Cloud Pull master data dan TRT		Mengembangkan `api/sync_pull.php` untuk menarik master kapal/dermaga, memakai `X-Device-ID` dan `X-Install-Token`, menjaga path audio lokal, serta recalculation target ATD saat TRT Cloud berubah.
56	Senin, 25 Mei 2026	Heartbeat Identitas PC dan status online		Mengonfigurasi identitas PC melalui `config_sistem` berisi `cloud_url`, `device_id`, `kode_port`, `id_pelabuhan_cloud`, dan mengirim heartbeat 30 detik dengan latensi ke `sync_heartbeat_log`.

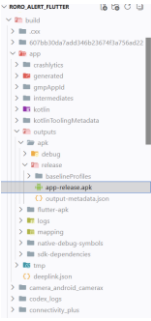
No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
57	Selasa, 26 Mei 2026	WebSocket Listener outbound Remote Reverb		Mengintegrasikan <code>`assets/js/websocket.js`</code> dengan channel <code>`roro-settings`</code> , <code>`roro-remote-control.{kode_port}`</code> , dan <code>`pelabuhan.{id_pelabuhan}`</code> agar browser HMI menerima remote pause/resume/reset serta sinyal fallback 4G.
58	Rabu, 27 Mei 2026	WebSocket Remote Eksekusi dan ack remote control		Mengembangkan <code>`api/ws_listener.php`</code> untuk mengeksekusi remote pause/resume/reset secara lokal lalu mengirim ack <code>`executed`</code> atau <code>`failed`</code> ke Cloud melalui <code>`/api/remote/ack`</code> dengan header <code>`X-Device-ID`</code> .
59	Kamis, 28 Mei 2026	Fallback Catch-up data dari Cloud		Membuat <code>`api/fallback_handler.php`</code> dan <code>`api/fallback_catchup.php`</code> untuk menarik <code>`/api/fallback/pending`</code> , menyimpan high-water mark di <code>`sync_state`</code> , dan memulihkan sandar/berangkat/kendala saat Reverb sempat putus.
60	Jumat, 29 Mei 2026	Postman Dokumentasi endpoint lokal Testing uji		Mendokumentasikan skenario uji Postman untuk <code>`sandar.php`</code> , <code>`berangkat.php`</code> , <code>`pause.php`</code> , <code>`sync_push.php`</code> , <code>`sync_pull.php`</code> , foto, GPS, dan kontrak <code>`X-Device-ID`</code> agar validasi PC lokal konsisten dengan <code>`docs/API_CONTRACT.md`</code> .

Rekap Kegiatan Juni 2026 (21 entri)

No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
61	Senin, 1 Juni 2026	Testing Validasi helper dan demo progres PC lokal		Menguji helper sinkronisasi dan audio engine melalui test_sync_helpers.php serta test_audio_engine.js, dengan hasil 17 pengujian konversi/sync dan 53 pengujian trigger T-10/T-5/T-0, sirine, recovery, serta anti-replay. Setelah pengujian, mendemonstrasikan progres PC lokal/HMI bersama Kepala Bidang untuk memvalidasi dashboard, audio warning, sinkronisasi Cloud, dan fallback catch-up.
62	Selasa, 2 Juni 2026	Foundation Flutter Dependency, theme, routing, dan struktur feature		Mengonfigurasi fondasi Flutter dengan dependency 'dio', 'flutter_bloc', Drift, secure storage, kamera, GPS, theme terang/gelap, routing awal, serta struktur 'core' dan 'features' untuk aplikasi mobile petugas.
63	Rabu, 3 Juni 2026	Foundation Flutter Database lokal dan utility inti		Membuat skema Drift untuk session, master kapal, master dermaga, config, log riwayat, dan 'pause_tracker_state', lalu menghubungkannya dengan DAO agar cache operasional tetap tersedia saat LAN atau 4G tidak stabil.
64	Kamis, 4 Juni 2026	Fallback Network LAN/Cloud 4G client		Mengembangkan 'DioClient' dan 'NetworkInfo' untuk membedakan mode 'lan', 'fallback4G', dan 'noConnection', termasuk timeout LAN 2 detik, Cloud timeout 30 detik, serta header ngrok untuk demo Cloud.
65	Jumat, 5 Juni 2026	Auth Login Cloud dan session petugas		Mengintegrasikan login Cloud berbasis Sanctum dengan penyimpanan token di secure storage, session petugas di Drift, 'server_time' ke 'TimeSync', remember me, logout, dan pemetaan error 401/403.
66	Senin, 8 Juni 2026	Master Sync Kapal, dermaga, config TRT, dan server time		Menghubungkan endpoint '/api/master-data' untuk menyimpan kapal, dermaga, radius geofence, konfigurasi TRT, dan 'server_time' ke cache lokal sehingga form ATA/ATD dapat memakai data master yang konsisten.

No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
67	Selasa, 9 Juni 2026	Home Status Status dermaga LAN dan Cloud		Mengembangkan Home Page yang mengambil status dermaga dari `GET /api/get_status.php` saat LAN aktif dan `GET /api/dermaga/status` saat fallback 4G, dengan parser status kapal, `paused_at`, dan `total_paused_sec`.
68	Rabu, 10 Juni 2026	Home Status Tab pelabuhan, KPI, dan polling		Menyesuaikan tab Bengkulu/Pakning, card dermaga, mode jaringan, KPI harian, riwayat hari ini, serta polling 2/10 detik agar petugas melihat port sendiri dan port lawan secara ter-scope.
69	Kamis, 11 Juni 2026	ATA/ATD Alur Sandar/ATA		Mengembangkan alur Sandar/ATA dengan kamera, foto wajib, GPS, geofence, UUID v4 idempotent, watermark petugas-koordinat-waktu, serta pengiriman multipart ke `sandar.php` atau `api/fallback/sandar`.
70	Jumat, 12 Juni 2026	ATA/ATD Alur Berangkat/ATD dan UI operasional		Mengembangkan alur Berangkat/ATD memakai UUID operasi aktif yang sama, foto ATD, koordinat, perhitungan status, guard operasi aktif, dan penyelarasan header serta tombol mengambang pada ATA/ATD/Kendala.
71	Senin, 15 Juni 2026	Kendala Pause/resume dan PauseTracker		Mengembangkan Kendala Page dengan pre-fill operasi aktif, kategori kendala, catatan, foto wajib, validasi GPS/geofence, action `pause/resume`, dan integrasi `PauseTracker` untuk durasi jeda dalam detik.
72	Selasa, 16 Juni 2026	Fallback Endpoint operasional 4G fallback		Menyesuaikan jalur fallback 4G untuk Sandar, Berangkat, dan Kendala melalui `api/fallback/sandar`, `api/fallback/berangkat`, dan `api/fallback/kendala`, sambil menjaga payload LAN tetap kompatibel dengan PC lokal.
73	Rabu, 17 Juni 2026	Profile Profil petugas dan IP PC lokal		Mengembangkan Profile Page untuk restore `GET /api/profile`, update nama/no HP/foto melalui multipart `_method=PATCH`, hapus foto profil, status login, pengaturan IP PC lokal, uji ping, dan preferensi tema.

No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
74	Kamis, 18 Juni 2026	Security Screenshot guard, device check, dan token warning		Menguatkan lapisan keamanan dengan `SecurePageMixin`, deteksi root/emulator, peringatan umur token lebih dari 25 hari, penghapusan log sensitif, serta lifecycle yang mematikan GPS saat halaman tidak membutuhkan lokasi.
75	Jumat, 19 Juni 2026	Crash Reporting Crashlytics dan logging aman		Mengintegrasikan `CrashReporter` berbasis Firebase Crashlytics release-only dengan custom key terbatas pada `network_mode`, `upload_status`, `upload_jenis`, dan `gps_accuracy` agar error upload dapat dilacak tanpa token, UUID, atau koordinat.
76	Senin, 22 Juni 2026	API Contract Atribusi petugas per aksi		Menyesuaikan payload LAN dengan `id_petugas_cloud` untuk Sandar, Berangkat, dan pelapor Kendala, sementara jalur 4G tetap memakai identitas petugas dari token Sanctum sesuai kontrak API terbaru.
77	Selasa, 23 Juni 2026	API Contract Handling konflik kapal dan dermaga		Memperbaiki penanganan HTTP 409 `DERMAGA_OCCUPIED`, `KAPAL_ALREADY_ACTIVE`, `ALREADY_PAUSED`, dan `NOT_PAUSED` sebagai penolakan final atau sinkron ulang tanpa retry otomatis yang berisiko menggandakan operasi.
78	Rabu, 24 Juni 2026	Testing Unit test dan state test operasional		Menguji parser status, tab pelabuhan, master sync, auth cubit, UUID, GPS/geofence, database, PauseTracker, conflict handling, dan guard Sandar agar perubahan API contract tidak merusak flow mobile petugas.
79	Kamis, 25 Juni 2026	Home Status KPI dan riwayat lintas device		Mengintegrasikan snapshot `/api/home/activity/today` agar KPI dan Riwayat Hari Ini tetap tampil pada HP baru atau akun berbeda, lalu menggabungkannya dengan log lokal same-port yang belum tersinkron.
80	Jumat, 26 Juni 2026	Dokumentasi README, status implementasi, dan checklist pra-demo		Mendokumentasikan arsitektur tiga pilar, endpoint LAN/Cloud, storage lokal, auth, build Android, batasan ngrok/signing, serta checklist preflight agar APK demo dapat diverifikasi secara berulang.

No.	Hari/Tanggal	Kegiatan	Dokumentasi Kegiatan	Keterangan
81	Senin, 29 Juni 2026	Demo APK dan presentasi akhir Ro-Ro Alert		Menyiapkan APK demo versi 1.0.0+7, menjalankan preflight Cloud/ngrok, serta memvalidasi tab Bengkulu/Pakning, mode 4G, KPI, riwayat Cloud, dan alur operasional pada perangkat Android. Seluruh progres web admin, PC lokal/HMI, dan Flutter kemudian dipresentasikan bersama Kepala Bidang sebagai demo akhir proyek Ro-Ro Alert.


 Mengetahui,
 Pembimbing Lapangan,
Edi Kurniawan, ST., M.T
 NIP. 198206162015031001

Bengkulu, 29 Juni 2026
 Mahasiswa,


Riski Afendi
 6304221510

Lampiran 4. Lembar Penilaian dari Instansi

PENILAIAN DARI INSTANSI KERJA PRAKTEK DINAS PERHUBUNGAN KABUPATEN BENGKALIS

Nama : Riski Afendi
NIM : 6304221510
Program Studi : Rekayasa Perangkat Lunak, Politeknik Negeri Bengkalis

No	Aspek Penilaian	Bobot	Nilai
1	Disiplin	20%	95
2	Tanggung-Jawab	25%	95
3	Penyesuaian diri	10%	90
4	Hasil Kerja	30%	95
5	Perilaku secara umum	15%	95
	Total jumlah (1+2+3+4+5) 100%		94,5

Keterangan:

Nilai : Kriteria

81 – 100 : Istimewa

71 – 80 : Baik sekali

66 – 70 : Baik

61 – 65 : Cukup baik

56 – 60 : Baik

Catatan:

Riski sudah cukup memahami alur dan layout
kerja sistem sehingga sudah mengimplementasikan
bagian yang di berikan.

Bengkalis, 29 Juni 2026
Kepala Bidang Pelayaran
Dishub Kabupaten Bengkalis



Edi Kurniawan, ST., M.T
NIP: 198206162015031001

Lampiran 5. Sertifikat Magang



Lampiran 6. Absensi Kehadiran

DAFTAR KEHADIRAN MAGANG
DINAS PERHUBUNGAN KABUPATEN BENGKALIS
TAHUN 2026

NAMA LENGKAP : RISKI EFENDI
JURUSAN : REKAYASA PERANGKAT LUNAK

BULAN	KEHADIRAN																															JUMLAH ABSENSI			KET
																																S	I	A	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
Maret									✓	✓	✓	✓	✓		✓	✓	✓	✓						✓	✓	✓	✓	✓	✓	✓	✓	✓			
April	✓						✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Mei				✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Juni		✓	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			

KEPALA SUBPAGKAWIJUM DAN KEPENGAWAJAN
STANISIA SUDARSONA
Pejabat Muda TL I (II/II)
NIP. 19931002 20100 1 004

Lampiran 7. Dokumentasi Kegiatan Kerja Praktek

