

Portal E.Rastreio

- [Capítulo 01 - Visão de Negócio](#)
- [Capítulo 02 - Módulos e Funcionalidades](#)
- [Capítulo 03 - Telas e Fluxos](#)
- [Capítulo 04 - Regras de Negócio](#)

Capítulo 01 - Visão de Negócio

Visão Geral do Sistema E.Rastreio

Propósito de Negócio

O **E.Rastreio Portal** é um sistema de **gestão operacional de frotas e rastreamento veicular**, concebido para atender cooperativas e operadores de transporte, com integração nativa à plataforma **COOCA+**.

Seu objetivo central é consolidar, em uma única plataforma, a operação de frotas brasileiras — indo além do rastreamento GPS puro para oferecer uma camada de gestão que abrange controle operacional, conformidade trabalhista, gestão financeira e monitoramento de segurança. O sistema atua como a camada de inteligência e gestão sobre uma base de dados de posicionamento GPS, transformando dados brutos de telemetria em informação acionável para gestores e analistas.

Padrão Arquitetural

A aplicação adota o padrão de **Monólito Modular Orientado a Domínios**.

Nesse modelo, a aplicação é executada como uma única unidade implantável (monólito), porém sua lógica de negócio é segmentada em domínios autônomos e coesos. Cada domínio encapsula suas próprias responsabilidades — controllers, serviços, modelos e rotas — mantendo fronteiras bem definidas entre as áreas funcionais do sistema.

Essa abordagem combina a simplicidade operacional de um monólito (implantação e manutenção unificadas) com a organização e a manutenibilidade tipicamente associadas a arquiteturas modulares, facilitando a evolução independente de cada área de negócio.

Tecnologias Principais

O sistema é construído sobre uma base tecnológica moderna e consolidada no ecossistema PHP:

Camada	Tecnologia
Linguagem	PHP 8.2+ (ambiente atual: 8.4)
Framework	Laravel 11
Interface (UI)	Blade (com layout baseado em Metronic)

Camada	Tecnologia
Autenticação	Sessão (guard <code>web</code> + <code>laravel/ui</code>) para usuários da interface; tokens Laravel Sanctum para clientes externos da API
Persistência	MySQL (arquitetura de banco de dados dual)
Ambiente	Docker / Laravel Sail

A plataforma ainda se apoia em recursos complementares para geração de documentos (PDF e Excel), armazenamento de arquivos em nuvem compatível com S3 e execução de rotinas automatizadas agendadas, sustentando as necessidades operacionais do negócio.

Capítulo 02 - Módulos e Funcionalidades

Introdução

Este capítulo descreve os módulos do E.Rastreio Portal conforme eles se apresentam na **interface do usuário**, seguindo a organização do menu lateral (sidebar). Cada grupo de menu reúne funcionalidades relacionadas e é exibido de acordo com as permissões do usuário autenticado.

1. Painéis

Ponto de entrada do sistema após o login.

- **Padrão:** dashboard principal com indicadores consolidados da frota.

2. Produtividade

Gestão de desempenho de analistas e metas operacionais.

- **Meu Dashboard:** indicadores individuais de produtividade.
- **Equipe:** visão consolidada da equipe (requer permissão de gerenciamento).
- **Config. SLA:** definição de metas e parâmetros de acompanhamento (requer permissão específica de SLA).

3. Câmeras

Monitoramento por câmeras embarcadas, integrado à plataforma **Jimi Cloud**.

- **Devices:** listagem de dispositivos, com acesso a visualização ao vivo (individual e em grade), gravações e transmissão HLS.
- **Comandos:** envio de instruções remotas às câmeras.
- **Downloads:** solicitação e gestão de arquivos de vídeo, com armazenamento em nuvem.

4. Rastreamento

Menu central que concentra o acompanhamento da frota, telemetria e compartilhamento. Reúne funcionalidades de três áreas do sistema.

- **Painel:** status dos rastreadores (online/offline, última comunicação).
- **Mapa em Tempo Real:** posição atual dos veículos, com velocidade, ignição e localização.
- **Replay de Rastreamento:** reconstituição do trajeto percorrido em um período.
- **Relatório de Rotas:** consolidação de deslocamentos, com exportação em Excel e PDF.
- **Telemetria:** painel analítico de desempenho por veículo. A partir de filtros de veículo e período, apresenta métricas organizadas em cinco seções — *Visão Geral, Tempo e Eficiência, Paradas, Segurança e Velocidade* — com detalhe em modais e exportação em PDF. Consulte o Capítulo 03 para o detalhamento das telas.
- **Score de Direção** (*requer permissão de telemetria*): ranking de motoristas, score individual, perfis de pontuação e análise por IA (Gemini), com exportação em PDF.
- **Premiação** (*requer permissão de telemetria — gerenciar*): definição e acompanhamento de recompensas por desempenho.
- **Espelhamentos:** compartilhamento externo de mapas via link tokenizado (com senha opcional) e cercas geográficas (geofences) que disparam alertas por e-mail na entrada/saída de veículos.

5. Ponto (EPONTODIG)

Controle da jornada de trabalho dos motoristas, com foco em conformidade trabalhista.

- **Painel:** visão consolidada das jornadas.
- **Automático:** cálculo de jornada a partir dos dados de GPS (ignição, movimento, paradas), com reprocessamento de períodos.
- **Manual:** lançamento e ajuste manual de pontos.

“ As exportações legais (**AFD, AEJ** e espelho de ponto em PDF) e a análise de jornada estão disponíveis a partir dos relatórios do módulo.

6. Cooca+

Integração com a plataforma externa de compras.

- **Portal de Compras:** consulta de limite, reservas e títulos a pagar junto à API da CoocaMais. Quando o usuário possui mais de um grupo com matrícula Cooca+ cadastrada, os dados são exibidos em abas, uma por grupo. Os dados de reservas e títulos são carregados via AJAX sob demanda.

7. Gestão

Gestão financeira ligada à operação da frota e dos motoristas.

- **Painel:** visão consolidada de entradas e saídas.
- **Receitas:** lançamento e categorização de receitas.
- **Despesas:** lançamento e categorização de despesas.
- **Transferências:** movimentações entre contas/entidades.
- **Acertos:** fechamento e liquidação de valores por motorista.

8. Cadastros

Gestão das entidades operacionais fundamentais.

- **Pessoas:** cadastro de motoristas e pessoas (dados pessoais, CNH, vínculos).
- **Veículos:** cadastro completo, com integração à tabela **FIPE**, vínculo com rastreadores e histórico de motoristas. O acompanhamento de CRM do veículo é acessado a partir daqui.
- **Rastreadores:** cadastro dos dispositivos e vínculo com veículos.

“ **Atenção:** o cadastro do dispositivo rastreador fica aqui, enquanto o *painel de acompanhamento* dos rastreadores fica no menu **Rastreamento**.

9. Administrador

Exibido apenas para o administrador máximo (master admin)

Área de configuração global, controle de acesso e gestão de entidades administrativas.

- **Grupos:** organização hierárquica de motoristas e veículos, com dados de empregador e integração CoocaMais.
- **Perfis de Acesso:** definição de perfis de permissão reutilizáveis.
- **Usuários:** gestão de contas e vínculos com veículos, grupos e associações.
- **Clientes de API:** gerenciamento de acesso à API REST do portal para sistemas e empresas externas. Permite cadastrar clientes (com nome e CNPJ/CPF), definir quais

veículos cada cliente pode consultar, e gerar ou revogar tokens de acesso individuais. O token gerado é exibido uma única vez no momento da geração e não pode ser recuperado posteriormente.

- **Associação:** entidades que agregam grupos, com credenciais de integração e regras de pernoite.
- **Alertas Pernoite:** monitoramento e tratamento de alertas de pernoite dos veículos.
- **Alertas Sinal:** monitoramento e resolução de alertas de perda de sinal.
- **Auditoria:** identificação e resolução de pendências operacionais e de cadastro.
- **Permissões em Massa:** atribuição de permissões a múltiplos usuários de uma só vez.
- **Configurações:** parâmetros globais do sistema.

10. Módulos em Preparação

Os módulos a seguir existem na estrutura do sistema, porém **ainda não estão expostos no menu de navegação** e, portanto, não são acessíveis pela interface no estado atual. Estão documentados aqui para registro e planejamento de evolução.

10.1. Zonas de Risco

Módulo destinado ao **monitoramento de áreas de risco** da operação.

- **Painel de zonas de risco:** visualização e acompanhamento das áreas mapeadas, com o objetivo de apoiar a gestão de segurança da frota.
- **Situação atual:** existe uma referência residual à rota do painel no gatilho do menu "Painéis", mas sem link visível na navegação. Aguarda desenvolvimento e exposição na interface.

10.2. Conjuntos

Módulo destinado ao **agrupamento e organização de entidades operacionais**.

- **Gestão de conjuntos:** estruturação de agrupamentos e associação a usuários, conforme as necessidades de gestão da operação.
- **Situação atual:** não possui entrada no menu de navegação. Aguarda desenvolvimento e exposição na interface.

Observações de Navegação

- **Rastreamento** é um menu combinado que agrega três domínios distintos (Rastreadores, Telemetria e Espelhamento).

- **Grupos, Associação, Alertas e Auditoria**, embora pertençam à área de administração no código, são acessados pelo menu **Administrador** na interface.
- Alguns nomes de negócio diferem da nomenclatura técnica interna do sistema.
- **Zonas de Risco** e **Conjuntos** ainda não estão disponíveis na navegação.

Capítulo 03 - Telas e Fluxos

“ Este capítulo descreve **o que o usuário vê e como interage** com o sistema, seguindo a ordem dos grupos do menu lateral. Para a lista de módulos e suas permissões, consulte o **Capítulo 02**. Para a base tecnológica e arquitetura, consulte o **Capítulo 01**.

3.1 Shell Visual da Aplicação

Toda página autenticada segue um layout fixo de três regiões:

- **Aside esquerdo** — menu de navegação colapsável; minimiza para ícones via botão de seta no topo.
- **Header fixo** (65 px) — toggle mobile do aside à esquerda; dropdown do usuário (Perfil / Logout) à direita. A área central do header não contém links de navegação.
- **Área de conteúdo** — ocupa o restante da tela; cada página injeta seu conteúdo dinamicamente.

A aplicação está configurada como **PWA**: possui `manifest.json`, service worker registrado e ícones de instalação — permitindo adicionar o portal à tela inicial de dispositivos móveis.

3.2 Fluxo de Autenticação

O usuário acessa `/login` — tela sem aside nem header autenticado. Após login bem-sucedido, é redirecionado para o dashboard principal. As demais telas (recuperação de senha, redefinição, confirmação) seguem o mesmo layout público.

3.3 Painéis

3.3.1 Dashboard Padrão

Primeira tela após o login. Apresenta uma visão executiva da frota em tempo real.

Filtro de usuário (visível apenas para administradores): seletor + botão "Aplicar" para visualizar os indicadores de outro usuário.

Quatro cards de KPI em linha:

Card	Cor	Valor
Total de Veículos	Azul	Contagem
Patrimônio FIPE	Verde	R\$ formatado
Veículos Parados	Ciano	Contagem
Em Movimento	Amarelo	Contagem

Card: Eventos de Excesso de Velocidade nas últimas 24h.

Tabela: Últimas Posições — colunas: Veículo (placa), Data/Hora, Endereço, Velocidade (km/h).

3.4 Produtividade

3.4.1 Meu Dashboard

Dashboard pessoal do usuário logado. O widget de críticas é carregado via AJAX, sem bloquear o carregamento da página principal.

3.4.2 Equipe

Visão consolidada de produtividade dos membros da equipe gerenciada.

3.4.3 Dashboard Geral

Visão organizacional ampla. **Não possui entrada no menu lateral** — disponível apenas mediante acesso direto.

3.4.4 Configuração de SLA

CRUD padrão (ver convenção na seção 3.10) para definição de metas e parâmetros de SLA utilizados nos cálculos de produtividade.

3.5 Câmeras

3.5.1 Devices (listagem)

Ponto de entrada do módulo. A partir de cada linha da listagem, o usuário acessa todas as sub-telas da câmera. A ação "Atualizar Status" está disponível diretamente na listagem.

3.5.2 Visualização ao Vivo

- **Live individual:** player HLS integrado para visualização de câmera única.
- **Grid de câmeras:** múltiplas câmeras lado a lado. Streams são iniciados e encerrados individualmente pelos botões na grade.

3.5.3 Gravações

Filtro por câmera e período → listagem de clipes → player integrado com controle de reprodução. A mesma tela oferece a ação de solicitar download de trechos específicos.

3.5.4 Downloads

Gerenciamento centralizado de arquivos de vídeo solicitados. O usuário acompanha o status de cada pedido e realiza o download ou streaming quando pronto.

3.5.5 Comandos

Envio de instruções remotas às câmeras. O histórico é exibido em formato de bolhas de conversa, similar a um chat.

3.6 Rastreamento

3.6.1 Painel de Rastreadores

Lista/grade operacional com status de cada rastreador (online/offline, última comunicação). Os dados são atualizados periodicamente via AJAX, sem recarregar a página.

3.6.2 Mapa em Tempo Real

Tela dominada pelo mapa Google Maps. Posições são consumidas via AJAX: lista geral de veículos e posição individual por ID.

3.6.3 Replay de Rastreamento

O usuário seleciona veículo e intervalo de datas/horas; o sistema carrega o histórico via AJAX e reproduz o trajeto animado sobre o mapa.

3.6.4 Relatórios

Tela	Exports
Relatório de Rotas	Excel, PDF
Utilização (Km e Horas)	PDF

Ambas seguem o padrão: formulário de filtro (veículo + período) → botões de exportação.

3.6.5 Telemetria — Painel Analítico

Ao acessar, um **loader overlay** cobre a tela enquanto os dados são processados. A tela é organizada em seções nomeadas.

Filtros: seleção de veículo + data inicial + data final (com seletor de usuário para administradores). O painel não exibe dados sem veículo selecionado.

Seção 1 — Visão Geral:

Card	Cor	Interação
KM Rodados (Período)	Azul	Clique → abre rota no Google Maps
KM Rodados (Hoje)	Verde	Clique → abre rota de hoje no Google Maps
Tempo Ocioso Total	Amarelo	Clique → modal com lista detalhada
Excessos de Velocidade	Vermelho	Clique → modal com ocorrências
Velocidade Média	Ciano	—
Velocidade Máxima	Escuro	—

Botão "**Ver Velocidades por Dia**" → modal com tabela de velocidade média e máxima por dia do período.

Seção 2 — Métricas de Tempo e Eficiência (4 cards):

- Tempo de Condução

- Tempo Parado
- Eficiência Operacional (% tempo em movimento / tempo total)
- KM/Hora Útil

Seção 3 — Paradas (3 cards):

- Número de Paradas → clique abre modal com tabela (colunas: Início, Fim, Duração, Tipo). Os tipos de parada são: **Curta** (< 30 min), **Longa** (30 min – 3h), **Pernoite** (> 3h em período noturno ou cruzando meia-noite) e **DSR** (> 24h).
- Tempo Médio de Parada
- Distância Média Entre Paradas (km)

Seção 4 — Segurança (3 cards):

- Acelerações Bruscas → clique abre modal com tabela (data/hora, aceleração m/s², velocidade, localização)
- Freadas Bruscas → clique abre modal com mesmo padrão
- Horário de Maior Movimento → exibe a hora de pico; clique abre modal com gráfico de KM por hora do dia (0h–23h)

Seção 5 — Velocidade (3 cards):

- Tempo em Baixa Velocidade (< 30 km/h)
- Tempo em Velocidade Média (30–70 km/h)
- Tempo em Alta Velocidade (> 70 km/h)

Exportações em PDF (botões persistentes na tela): Relatório de Velocidades por Dia e Relatório de Excesso de Velocidade.

3.6.6 Score de Direção — Ranking

Dashboard de ranking de motoristas com pontuações. A partir de uma linha do ranking, o usuário navega para a **tela de detalhe individual**, que exibe análise de comportamento, análise via IA, integração com plataforma parceira de análise e exportação em PDF.

Sub-fluxo — Perfis de Score: CRUD para configurar os critérios de pontuação.

3.6.7 Premiação

CRUD de programas de premiação para motoristas com base no score de direção.

3.6.8 Espelhamentos

CRUD de mapas públicos compartilháveis. Ações disponíveis: criar, editar, excluir, ativar/desativar e regenerar senha de geofence. O link gerado abre a área pública descrita na seção 3.12.

3.7 Ponto (EPONTODIG)

3.7.1 Painel de Ponto

Dashboard com indicadores consolidados das jornadas de trabalho dos motoristas.

3.7.2 Ponto Automático

Tela mais interativa do módulo. Fluxo principal:

1. Seleção do motorista e período.
2. Acionamento do processamento ou reprocessamento de um dia específico.
3. **Polling periódico** para exibir progresso do job em andamento.
4. Após processamento, períodos calculados são exibidos para revisão.
5. Cada período pode ser editado em modal inline ou excluído.

3.7.3 Ponto Manual

Formulário simples para lançamento de registro de ponto: motorista, data e hora. Registros existentes são listados na mesma tela e podem ser excluídos.

3.7.4 Relatório de Jornada

Acessada por rota direta (sem item no menu lateral). Oferece:

- Visualização do espelho de ponto.
- Downloads: **AFD**, **AEJ** e espelho em PDF.
- Análise de jornada.
- Modal de ajuda contextual disponível na página.

3.8 Cooca+

3.8.1 Portal de Compras

A tela exibe os dados do(s) cooperado(s) vinculados aos grupos do usuário. Quando o usuário possui **múltiplos grupos** com matrícula Cooca+ cadastrada, os resultados são organizados em **abas** (uma por grupo). Se nenhum grupo possuir matrícula configurada, um alerta de aviso é

exibido.

Os dados das três visões são carregados **sob demanda via AJAX**:

- **Painel principal**: limite, saldo e dados gerais do cooperado.
- **Reservas**: listagem de reservas ativas.
- **Títulos a Pagar**: obrigações financeiras junto à CoocaMais.

3.9 Gestão

3.9.1 Painel Financeiro

Dashboard com visão consolidada do mês selecionado. Possui **filtro de mês/ano** no topo (padrão: mês atual).

Quatro cards de KPI:

Card	Cor	Detalhe
Total Receitas	Verde	Subtítulo: contagem de fretes do mês
Total Despesas	Vermelho	Subtítulo: contagem de abastecimentos
Transferências	Ciano	Detalhado em Adiantamentos (–) e Devoluções (+)
Saldo	(variável)	Receitas – Despesas – Adiantamentos + Devoluções

Ações rápidas no toolbar: botões "Nova Receita", "Nova Despesa", "Nova Transferência" — atalhos para criação direta sem passar pela listagem.

Últimas 5 entradas para cada tipo (Receitas, Despesas, Transferências) em tabelas abaixo dos cards.

Gráfico mensal: evolução comparativa de receitas vs. despesas dos últimos 6 meses.

3.9.2 Movimentações — Receitas, Despesas e Transferências

As três telas seguem o padrão CRUD (ver seção 3.10): listagem com botão "Cadastrar", formulário de criação e formulário de edição.

3.9.3 Acertos

Fluxo com etapa extra na criação: o usuário pode **buscar itens pendentes via AJAX** para compor o acerto antes de salvar. A tela de detalhe exibe o acerto completo após o registro.

3.10 Cadastros

Convenção CRUD

A maioria das telas de cadastro e configuração do sistema segue este padrão visual:

- **Listagem:** tabela com registros, botão "Cadastrar" no topo, ações por linha (Editar, Excluir). A exclusão abre modal de confirmação.
- **Criação:** formulário limpo.
- **Edição:** mesmo formulário, pré-preenchido.

3.10.1 Pessoas

Formulário com duas abas: **Geral** (dados pessoais/empresariais) e **Adicional** (informações complementares).

3.10.2 Veículos (tela mais rica do domínio)

A **edição** de um veículo é organizada em seis abas:

Aba	Conteúdo
Geral	Dados de identificação e características do veículo
Parâmetros	Configurações operacionais (inclui limite de velocidade usado na Telemetria)
Rastreadores	Vinculação e remoção de rastreadores
Motoristas	Motorista atual + histórico de vínculos; ação "Encerrar vínculo atual"
Histórico (CRM)	Linha do tempo de notas e ocorrências
Pernoite	Configuração de alertas de pernoite

O formulário também disponibiliza o botão **Consultar FIPE**, que preenche automaticamente o valor de mercado via integração externa.

3.10.3 Rastreadores

CRUD padrão de dispositivos de rastreamento com vínculo a veículos.

3.10.4 CRM do Veículo

Tela dedicada ao histórico operacional de um veículo, acessada por link contextual. Não aparece no menu lateral — é acessada por links dentro do cadastro do veículo. Possui:

- **Filtros:** origem, tipo de evento, data início e data fim.
 - **Resumo:** indicadores consolidados do histórico.
 - **Linha do tempo:** notas e ocorrências com opção de adicionar nova nota.
-

3.11 Administrador

Todas as telas desta seção seguem a convenção CRUD da seção 3.10. Os destaques de interação:

- **Usuários:** além do CRUD, cada usuário possui duas sub-telas de configuração:
 - **Permissões de Módulos:** matriz de módulos × níveis de acesso.
 - **Vínculos de Veículos e Grupos:** define o escopo de dados visível para o usuário.
 - **Permissões em Massa:** não segue o padrão CRUD. Tela especial para aplicar permissões a múltiplos usuários de uma só vez.
 - **Cientes de API:** o fluxo de geração de token exibe o token **uma única vez** na tela — após fechar, não pode ser recuperado.
 - **Alertas Sinal:** a tela de detalhe permite marcar como resolvido e adicionar contatos responsáveis.
-

3.12 Área Pública — Espelhamento

Acessível sem autenticação via link tokenizado, esta tela usa um layout simplificado — sem menu de navegação, header autenticado ou footer. Exibe apenas o mapa ao vivo com os veículos do link compartilhado. Quando o token é inválido ou expirado, o sistema exibe uma tela de erro específica.

Capítulo 04 - Regras de Negócio

4.1 Visão Geral

Este capítulo documenta as decisões algorítmicas e regras de validação que governam o comportamento interno do sistema E.Rastreo Portal. As regras estão organizadas por domínio e foram extraídas diretamente do código-fonte (`app/Domains/*/` e `app/Models/`).

4.2 Módulo: Jornada de Trabalho

4.2.1 Detecção de Movimento e Parada

O sistema determina se um veículo está em movimento ou parado com base em dados brutos de posição GPS (tabela `tc_positions` do Traccar). O serviço central é o `VehicleMovementService`, que implementa a seguinte lógica:

- **Limiar de velocidade padrão:** velocidade < 5 km/h classifica o veículo como parado. A velocidade é convertida de nós para km/h (fator: $\times 1.852$).
- **Parada mínima válida:** um intervalo de parada só é reconhecido se tiver duração ≥ 5 minutos contínuos.
- **Paradas dentro de raio geográfico (carga/descarga):** quando configurado um raio (`radiusKm`), mesmo que o veículo supere o limiar de velocidade, ele ainda é considerado "parado" enquanto permanece dentro da distância do ponto de referência. Isso modela cenários de carga e descarga.
- **Intervalo de posição inválido:** posições com intervalo superior a 600 segundos entre si são descartadas e reiniciam o acumulador de parada.

Esses parâmetros (`speed_threshold_kmh`, `min_stop_minutes`, `min_driving_seconds`) podem ser sobrescritos por veículo por meio da tabela `veiculo_parametros`, permitindo calibração individual por frota.

4.2.2 Derivação dos Períodos de Direção (Ponto Automático)

Os períodos de direção são derivados como os **gaps entre as paradas válidas**, delimitados pela primeira e última posição GPS do dia. O algoritmo (

`PontoAutomaticoService::getDrivingPeriodsFromPositions`) funciona da seguinte forma:

1. Busca todas as posições do device no dia (00:00–23:59:59).
2. Chama o `VehicleMovementService` para obter os intervalos de parada.
3. A janela entre o fim de uma parada e o início da próxima é classificada como **período de direção**, desde que sua duração seja $\geq$ `min_driving_seconds` (padrão: 60 s).
4. Um período de direção final é adicionado se ainda houver tempo entre a última parada e a última posição do dia.

Se não houver posições para o dia, o registro de ponto é criado com todos os tempos zerados.

4.2.3 Classificação dos Tempos da Jornada

O `DriverTimesheetService` possui uma lógica alternativa (modo legado) para segmentação de posições brutas, com as seguintes constantes baseadas na **Lei 13.103/2015 (Lei do Motorista)**:

Parâmetro	Valor
Pausa mínima (intervalo)	15 minutos (900 s)
Refeição mínima	1 hora (3.600 s)
Descanso mínimo	8 horas (28.800 s)
Tempo máximo de direção	8 horas (28.800 s)
Limiar de velocidade (movimento)	> 1 km/h
Viagem mínima	5 minutos / 500 m

As paradas são classificadas hierarquicamente:

- Duração $\geq$ 28.800 s $\rightarrow$ **Descanso**
- Duração $\geq$ 3.600 s $\rightarrow$ **Refeição/Almoço**
- Duração $\geq$ 900 s $\rightarrow$ **Pausa**
- Abaixo disso $\rightarrow$ descartada

4.2.4 Cálculo do Tempo de Descanso (Ponto Automático)

No fluxo principal (automático), o tempo de descanso é calculado por subtração:

descanso = 24 horas – tempo_total_de_direção – 1 hora (almoço fixo)

O resultado nunca é negativo (`max(0, descanso)`). O almoço de 1 hora é tratado como fixo e não deduzido do cálculo de horas extras.

4.2.5 Cálculo de Horas Noturnas

Horas noturnas são contabilizadas para os períodos de **direção** que ocorram dentro das faixas:

- Madrugada: **00:00 - 05:00**
- Noite: **22:00 - 23:59:59**

O sistema calcula a interseção exata (em segundos) entre cada período de direção e as faixas noturnas.

4.2.6 Cálculo de Horas Extras

As regras seguem a legislação trabalhista brasileira para motoristas profissionais:

- **Jornada regular:** configurável por veículo (`horas_regulares`); padrão = **8 horas**.
- **Hora extra 50%:** aplicável apenas em dias **úteis**. Todo tempo de direção que exceda a jornada regular é computado como extra 50%.
- **Hora extra 100%:** aplicável quando o dia for **domingo** ou **feriado nacional**. Nesse caso, todo tempo de direção acima da jornada regular é computado como extra 100%.
- Em dias úteis, **não há extra 100%** pelo simples fato de ultrapassar certo limite — todo o excesso é 50%.
- Se o veículo trabalhou menos que a jornada regular, os campos de hora extra são zerados.

4.2.7 Detecção de Feriados

O `HolidayService` consulta a **BrasilAPI** (`brasilapi.com.br/api/feriados/v1/{ano}`) para obter os feriados nacionais do ano em questão. O resultado é cacheado por **24 horas** para evitar chamadas repetidas. Em caso de falha da API, o sistema usa um fallback com os 9 feriados nacionais fixos (ex.: Ano Novo, Tiradentes, Dia do Trabalho etc.).

4.2.8 Vínculo Motorista–Veículo por Data

A identidade do motorista em um registro de ponto é resolvida consultando a tabela `veiculo_motorista_historico`, que registra períodos de vigência do vínculo (`data_inicio`, `data_fim`). O sistema localiza o motorista cuja janela de vigência contém a data do ponto. Se houver vários vínculos, o mais recente é selecionado. O vínculo pode estar aberto (`data_fim IS NULL`), indicando que ainda está em vigor.

4.2.9 Vínculo Device–Veículo por Data

Para processamento retroativo ou por motorista, o sistema resolve qual rastreador (`id_rastreador`) estava associado ao veículo em cada data específica, consultando `rastreador_historico` com o filtro de período de vigência (`is_principal = true`).

4.2.10 Edição Manual de Registros de Ponto

Registros de ponto podem ser editados manualmente. As regras aplicadas no momento da edição são:

- **Validação de formato:** horário de início e fim devem estar no formato `H:i:s`.
 - **Período noturno:** se o horário de fim for anterior ao de início, adiciona-se 1 dia ao fim (ex.: turno 23:00–02:00).
 - **Tipo de dia:** o dia pode ser marcado como **DSR** ou **Férias**, zerando todos os períodos de direção e atribuindo 24 horas de descanso.
 - Após qualquer edição, o sistema **recalcula automaticamente** todos os totais: tempo de direção, descanso, horas noturnas, horas extras 50% e 100%, com respeito ao flag `is_holiday` que pode ser marcado manualmente para dias de feriado não reconhecido pela API.
 - O campo `notes.manually_edited = true` é gravado para rastreabilidade.
-

4.3 Módulo: Telemetria e Score de Condução

4.3.1 Detecção de CAN Bus

Antes de calcular qualquer score, o sistema detecta se o rastreador do veículo transmite dados de CAN Bus (barramento do veículo). Isso é feito amostrando até 50 posições recentes: se ao menos **10%** possuírem os atributos `rpm`, `throttle` ou `fuelConsumption`, o veículo é marcado como `tem_can = true`. A presença de CAN desbloqueia indicadores adicionais de pontuação.

4.3.2 Resolução do Device para Telemetria

Para veículos com múltiplos rastreadores ativos, o sistema prioriza a seguinte hierarquia ao selecionar o device para telemetria:

1. Rastreador marcado com `is_telemetria = true`
2. Rastreador com `is_principal = true`
3. Qualquer rastreador ativo

4.3.3 Cálculo de Métricas Brutas (por Posição)

O `TelemetriaScoreService` processa cada par de posições consecutivas e acumula métricas. As regras de filtragem são:

- Pares com intervalo ≤ 0 s ou > 600 s são descartados (interrupção de sinal).
- **Motor ligado:** definido como `velocidade  $\geq 2$  km/h` (movimento estrito), OU para veículos com CAN: `rpm > 0`, OU sem CAN: flag `ignition = true`.
- **Idle (motor ocioso):** motor ligado, mas parado por **mais de 120 segundos** consecutivos. Paradas curtas (≤ 120 s, ex.: semáforo) **não contam como idle** — esse comportamento replica o cálculo do GOBRAX.
- **Excesso de velocidade:** posição com velocidade acima do `limite_velocidade` configurado no veículo (padrão: 90 km/h).
- **Aceleração/frenagem brusca:** calculada pela variação de velocidade entre duas posições dividida pelo intervalo. Limiar: $> 3,0 \text{ m/s}^2$ (positivo = aceleração, negativo = frenagem).
- **Contagem de viagens:** incrementada a cada transição `ignition: true  $\rightarrow$  false`.

4.3.4 Métricas CAN-Exclusivas

Quando `tem_can = true`, são calculadas métricas adicionais por faixa:

Faixas de RPM (modelo GOBRAX):

Faixa	Limiar (configurável no <code>ScoreProfile</code>)	Classificação
Início (ideal)	$\leq$ <code>rpm_inicio_max</code> (padrão: 1.300 RPM)	Positivo
Final (bom)	1.301 - <code>rpm_final_max</code> (padrão: 1.500 RPM)	Positivo parcial
Amarelo	1.501 - <code>rpm_freio_motor_max</code> (2.300), acelerando (<code>throttle $\geq 5\%$</code>)	Negativo
Freio motor (positivo)	1.501 - 2.300, pé fora (<code>throttle < 5%</code>) + combustível $< 0,1$	Positivo (corte de injeção real)
Vermelho	> 2.300 RPM	Negativo forte

Faixas de Pressão do Acelerador (throttle):

Faixa	Limiar	Classificação
Ideal	$\leq$ <code>throttle_ideal_max</code> (padrão: 60%)	Positivo
Atenção	61% - <code>throttle_atencao_max</code> (padrão: 70%)	Penalidade leve
Crítico	$> 70\%$	Penalidade forte

O sistema usa o **pico** (máximo entre as duas amostras do intervalo) para o throttle, a fim de capturar picos que caem nas bordas da amostragem.

Embalô (aproveitamento de inércia): veículo em movimento (velocidade ≥ 2 km/h), pé fora do acelerador (throttle $< 5\%$) e combustível < 3 L/h e RPM > 300 . Sinaliza aproveitamento da inércia do veículo.

Piloto automático (estimado): pé fora do pedal (throttle $< 5\%$) + motor injetando (combustível ≥ 3 L/h) + velocidade ≥ 20 km/h. Proxy para uso de cruise control, que o rastreador não reporta diretamente.

4.3.5 Cálculo de Distância (km/dia)

Para veículos **com CAN**, o sistema prefere o odômetro do veículo (`obd0dometer` ou `odometer`, em metros) ao invés do cálculo GPS Haversine. Regras de sanidade aplicadas ao delta do odômetro:

- Delta > 2.000 km/dia $\rightarrow$ descartado (considerado dado inválido).
- Delta $> 3 \times$ o km calculado por GPS (quando GPS > 5 km) $\rightarrow$ descartado (possível reset ou spike).

Para veículos **sem CAN**, ou quando o odômetro não passar na sanidade, o km é calculado pela fórmula **Haversine** entre posições consecutivas. Segmentos com velocidade implícita > 150 km/h são descartados como saltos de GPS.

4.3.6 Cálculo de km/L (Eficiência de Combustível)

A eficiência é calculada usando o campo `fuelUsed` (acumulado pelo CAN). O sistema captura o **delta** entre o primeiro e o último valor não-zero do dia. Valores acima de **1.000 L/dia** são descartados como inválidos (unidade errada ou reset do dispositivo).

4.3.7 Normalização das Notas por Indicador (0–100)

Cada indicador é normalizado para uma nota de 0 a 100:

Indicador	Fórmula resumida
Idle	100 até ~3% de idle/ignição, caindo linearmente a 0 próximo de 40%
Excesso	100 sem excesso; cada ponto percentual de tempo em excesso desconta 10 pts
Aceleração brusca	$100 - (\text{eventos}/100 \text{ km}) \times 5$
Freada brusca	$100 - (\text{eventos}/100 \text{ km}) \times 5$

Indicador	Fórmula resumida
RPM	Ponderado: $\text{RPM_início} \times 1,0 + \text{RPM_final} \times 0,7 + \text{Freio_motor} \times 1,0$, dividido pelo tempo em movimento
Throttle	$100 - (\text{crítico\%} \times 300) - (\text{atenção\%} \times 100)$
Embalo	Linear; 100 pts a partir de ~20% do tempo em movimento em embalo
Piloto automático	0 abaixo de 50% do tempo em movimento, 100 pts a 85% (linear)
km/L	$(\text{km_L_real} / \text{meta_km_L}) \times 100$, máximo 100

4.3.8 Score Final

O score final é a **média ponderada** das notas pelos pesos configurados no `ScoreProfile` do grupo do veículo. Quando não há CAN, apenas os indicadores GPS (idle, excesso, aceleração, freada) são usados; seus pesos são **renormalizados** para somar 100%, descartando os indicadores CAN.

O score **do período** (ex.: mensal) é calculado de forma **agregada** — somando os tempos/eventos e pontuando uma vez sobre os totais, não pela média das notas diárias. Isso replica o comportamento do GOBRAX e evita distorções em dias com poucos km.

4.3.9 Cálculo de Desperdício e Economia de Combustível

Baseados na meta de eficiência (`meta_kmL`) do `ScoreProfile` e no preço do diesel configurado:

```
desperdício_litros = max(0, litros_consumidos - (km_total / meta_kmL))
desperdício_R$ = desperdício_litros × preco_diesel

economia_litros = max(0, (km_total / meta_kmL) - litros_consumidos)
economia_R$ = economia_litros × preco_diesel
```

4.3.10 Premiação por Desempenho

O `PremiacaoService` calcula o valor de premiação de um motorista com o seguinte algoritmo:

- Gate de km mínimo:** se o motorista rodou menos que `km_minimo_mes`, recebe R\$ 0,00.
- Fator de nota:** proporcional ao intervalo entre `nota_minima` e 100. Abaixo da nota mínima, o valor base é zero.
- Fator de km:** `min(1,0, km_rodado / meta_km_mes)`. Penaliza quem não atingiu a meta de quilometragem.

4. **Base** = `premiacao_max` × `fator_nota` × `fator_km`

5. **Premiação extra (opcional):**

- Ativada pelo flag `extra_ativo`.
- Calculada sobre os km acima de `extra_km_minimo`: `(km_elegivel / extra_km_considerado) × extra_valor_por_km`.
- Se `extra_atrelar_nota = true`, o valor extra é multiplicado por `nota/100`.
- Se `extra_zerar_abaixo_range = true` e a nota está abaixo do mínimo, o valor total (base + extra) é zerado.

4.4 Módulo: Rastreadores e Monitoramento em Tempo Real

4.4.1 Classificação de Status do Veículo

O `VehicleTrackingService` classifica cada veículo em três estados, usando a última posição GPS recebida:

Status	Condição
em_movimento	Velocidade > 5 km/h E última posição há ≤ 5 horas
parado	Velocidade ≤ 5 km/h E última posição há ≤ 5 horas
sem_sinal	Última posição há > 5 horas (independente da velocidade)

4.4.2 Relatório Diário de Utilização

O `VehicleUsageReportService` produz um resumo diário com km rodado, segundos de direção e velocidade média. A lógica de cálculo de tempo de direção é **idêntica** à do Ponto Automático (`getDrivingPeriodsFromPositions`), garantindo que os valores sejam consistentes entre os dois módulos.

A velocidade média é calculada como `km / (driving_seconds / 3600)`, sem contar os períodos de parada.

4.4.3 Conversão de Placa (Padrão Mercosul)

O modelo `Veiculo` implementa a conversão bidirecional entre o padrão antigo (ex.: `ABC1234`) e o Mercosul (ex.: `ABC1D34`), mapeando a 5ª posição: `A=0, B=1, C=2, ..., J=9`. As buscas por placa sempre consideram **ambas as variantes** para não perder registros.

4.5 Módulo: Acessos e Permissões

4.5.1 Hierarquia de Usuários

O sistema tem três perfis de acesso principais:

- **Master Admin** (`is_master_admin = true`): acesso irrestrito a todos os módulos, veículos e grupos. As verificações de permissão retornam `true` imediatamente para esse perfil, sem consultar o banco.
- **Analista** (`is_analista = true`): usuário operacional, associado a um gestor (`gestor_id`).
- **Usuário padrão**: acesso determinado pelas permissões de módulo vinculadas.

4.5.2 Controle de Acesso a Veículos

Um usuário não-admin enxerga apenas os veículos que pertençam a:

1. **Grupos** aos quais ele está vinculado (`user_grupo`), ou
2. **Veículos** diretamente vinculados a ele (`user_veiculo`).

A union dos dois conjuntos (sem duplicatas) forma o escopo de visibilidade do usuário.

4.5.3 Permissões por Módulo e Ação

Permissões são armazenadas em JSON na tabela pivot `user_module_permissions`. Cada registro contém um mapa de `{action_slug: true/false}`. A verificação de permissão (`hasPermission`) exige que o módulo esteja ativo (`is_active = true`) e que a ação específica esteja marcada como `true` no JSON.

4.5.4 Operação em Massa de Permissões

O `BulkPermissionService` aplica permissões a múltiplos usuários (até **500 por operação**, em lotes de 100). Regras de segurança:

- **Master admins são ignorados** automaticamente (nunca são alterados por operações em massa).
- **Auto-revogação bloqueada**: o sistema impede que um usuário revogue a própria permissão `admin.ver` de si mesmo.
- A operação é **não-destrutiva**: ao revogar, se após a remoção de uma ação ainda houver outras `true` no JSON, o registro é atualizado; caso contrário, é deletado integralmente.
- Toda a operação é executada dentro de uma **transação de banco de dados**.

As ações válidas são validadas contra o método `actionsAvailable()` do módulo. Slugs inválidos causam rejeição de toda a requisição.

4.5.5 Aplicação de Perfil de Acesso

Ao aplicar um `AccessProfile` a um usuário, o serviço `AccessProfileApplicationService`:

1. Verifica se o usuário é Master Admin; se for, **não faz nada** (proteção de integridade).
2. Propaga o flag `is_analista` do perfil para o usuário.
3. Vincula os módulos do perfil ao usuário com as permissões definidas no perfil (via `attach`).

4.6 Módulo: Produtividade e SLA de Analistas

4.6.1 Métricas de Output

O output de um analista é calculado contando os registros de `veiculo_contato` criados por ele no período, classificados por origem:

Tipo	Critério
Sinal	Registro vinculado a um <code>alerta_sinal_id</code>
Pernoite	Registro vinculado a um <code>alerta_pernoite_id</code>
Auditoria (veículo)	Registro vinculado a <code>auditoria_sistema_id</code>
Auditorias extras	Auditorias de categoria <code>grupo</code> ou <code>usuario</code> resolvidas pelo analista no período
Notas	Contatos sem vínculo com nenhum alerta ou auditoria

4.6.2 Apuração de SLA

O SLA é apurado por origem de alerta, usando as configurações da tabela `sla_configs` (resolvida via `SlaConfig::resolver(origem, subtipo)`):

- **Alerta de Sinal:** dois subtipos — `<24h` (alerta recente) e `>=24h` (crítico). O SLA é medido em minutos entre `created_at` e `resolvido_em`.
- **Alerta de Pernoite:** SLA único (`pernoite`), medido entre `created_at` e `resolvido_at`.
- **Auditoria:** SLA por `tipo_pendencia`. O prazo começa na `data_execucao` da auditoria (início do dia).

Um atendimento é classificado como "**no prazo**" se o tempo de resolução (em minutos) for $\leq$ `sla_minutos`. A **aderência** (%) é $\text{no_prazo} / (\text{no_prazo} + \text{estourado}) \times 100$.

4.6.3 Alertas Críticos em Tempo Real

O sistema identifica atendimentos com risco de estouro de SLA combinando a configuração `sla_minutos` com o campo `warning_pct` (percentual de decurso do prazo a partir do qual o alerta entra na lista crítica). Um alerta é considerado crítico quando:

$$\text{pct_decorrida} = (\text{minutos_desde_abertura} / \text{sla_minutos}) \times 100 \geq \text{warning_pct}$$

4.6.4 Score de Qualidade

O score de qualidade de um analista é composto por três fatores ponderados:

Fator	Peso	Critério
Observações ricas	60%	% de resoluções com descrição $\geq$ 30 caracteres
Taxa sem reabertura	30%	$1 - (\text{reaberturas} / \text{total de resoluções})$
Taxa de resolução	10%	% de eventos do tipo <code>resolucao</code> entre todos os eventos

O score máximo é 100. **Reabertura** é contada quando uma auditoria que o analista havia resolvido voltou a aparecer como pendente para a mesma entidade + tipo.

4.7 Módulo: Cadastros

4.7.1 Auditoria Automática do Sistema

O `SystemAuditService` executa verificações diárias (`runFullAudit`) e registra pendências na tabela `auditoria_sistema`. As verificações são:

Grupos:

- Grupo sem `id_grupo_rastreadores` (ID eRastreio não configurado).
- Grupo sem `matricula_coocamais` (Matrícula Cooca+ não configurada).

Usuários (excluindo Master Admins):

- Usuário sem nenhum grupo vinculado.

- Usuário sem grupo e sem veículo vinculado.

Veículos (apenas status "Ativo"):

- Veículo sem grupo vinculado (`id_grupo` nulo).
- Veículo sem rastreador principal ativo (`is_principal = true` na tabela `rastreador_historico`).
- Veículo que passou pela **Região Metropolitana de São Paulo** nas últimas 24 horas **sem motorista vinculado**. A detecção é feita por busca textual no campo `address` das posições Traccar, cobrindo 15 municípios da Grande SP.

Idempotência: antes de apagar os registros do dia atual e regenerá-los, o serviço preserva quais itens já foram marcados como `resolvido = true` e reaplicará esse status após a nova execução, evitando perda de trabalho do analista.

4.7.2 Consulta de Valor FIPE

O `FipeService` consulta o valor de mercado de veículos (categoria: caminhão) na tabela FIPE via API oficial. O resultado é cacheado por **24 horas** por código FIPE + ano. A resposta textual no formato `"R$ 99.332,90"` é convertida para `float` removendo símbolo, pontos de milhar e substituindo a vírgula decimal por ponto.

4.8 Módulo: Câmeras Embarcadas

4.8.1 Gestão de Gravações (Jimi IoT)

O `RecordingService` gerencia gravações de câmeras Jimi IoT. As regras de operação são:

- **Todos os horários** são enviados ao dispositivo em **UTC**, independente do fuso horário do servidor.
 - **Modo demo:** se a variável `JIMI_DEMO_STREAM_URL` estiver configurada, o sistema retorna dados fictícios sem enviar comandos reais ao dispositivo. Isso permite demonstrações sem hardware.
 - A listagem de gravações tolera variações no formato da resposta da API do dispositivo, buscando a lista nos campos `list`, `items`, `records`, `fileList`, `_list` ou `data`.
-

4.9 Módulo: Alertas de Pernoite

4.9.1 Ciclo de Estados do Alerta de Pernoite

Um alerta de pernoite percorre os seguintes estados:

```
avaliar → aprovado
        → reprovado
        → sem_sinal (veículo sem sinal no momento do pernoite)
```

O estado inicial é `avaliar`. A transição para `aprovado` ou `reprovado` registra `resolvido_por_user_id` e `resolvido_at`. Alertas no estado `avaliar` ou `sem_sinal` são listados na fila de trabalho do analista e entram no cálculo de SLA.

4.10 Módulo: Alertas de Sinal

4.10.1 Ciclo de Estados do Alerta de Sinal

```
aberto → resolvido
```

Alertas no estado `aberto` entram na fila de trabalho. A resolução registra `resolvido_por_user_id` e `resolvido_em`. O campo `horas_sem_sinal` determina o **subtipo de SLA** aplicável: `<24h` ou `>=24h`.

4.11 Módulo: Financeiro

4.11.1 Tipos de Receita e Validação Condicional

As receitas possuem dois tipos com conjuntos de campos distintos:

Tipo	Campos obrigatórios	Campos exclusivos
<code>frete</code>	<code>origem</code> , <code>destino</code>	<code>odometro_inicial</code> , <code>odometro_final</code> , <code>peso</code> , <code>tipo_carga</code>
<code>outro</code>	<code>descricao</code>	—

A validação é aplicada condicionalmente com base no `tipo`. Para o tipo `frete`, os campos de `origem` e `destino` são obrigatórios, enquanto para `outro` a `descricao` passa a ser obrigatória. Campos que não se aplicam ao tipo selecionado são **explicitamente zerados** antes de persistir.

4.11.2 Resolução Automática do Motorista

Tanto nas receitas quanto nas despesas, o motorista responsável é **determinado automaticamente** a partir do veículo selecionado. O sistema consulta a relação `motoristaAtual` do veículo (via `VeiculoMotoristaHistorico`) e atribui o `pessoa_id` resultante. Se nenhum veículo for selecionado, `motorista_id` é nulo. O usuário não informa o motorista manualmente.

4.11.3 Tratamento de Máscara Monetária

Todos os campos de valor monetário chegam no formato de exibição brasileiro (ex.: `R$ 1.234,56`). O sistema remove o símbolo `R$`, pontos de milhar e espaços, depois converte a vírgula decimal em ponto antes de persistir como decimal: `"R$ 1.234,56" → 1234.56`.

4.11.4 Despesas de Abastecimento

Despesas categorizadas como "**Abastecida**" desbloqueiam campos extras: `produto` (tipo de combustível), `odometro` (leitura do hodômetro) e `litros` (volume abastecido, mínimo 0,01 L). Para qualquer outra categoria, esses três campos são **zerados automaticamente** antes de salvar, independente do que foi enviado no formulário.

4.11.5 Imutabilidade após Acerto

Receitas, despesas e transferências que já foram incluídas em um **Acerto** (`acerto_id IS NOT NULL`) tornam-se imutáveis. Qualquer tentativa de editar ou excluir esses registros é bloqueada antes de qualquer processamento de dados, com redirecionamento e mensagem de aviso. Esse é o mecanismo de integridade contábil do módulo.

4.11.6 Cálculo do Saldo do Acerto

O Acerto é a operação de fechamento financeiro de um motorista para um período. Ele agrega itens de três tabelas distintas e calcula o saldo pela seguinte fórmula:

```
saldo = total_receitas - total_despesas - total_adiantamentos + total_devolucoes
```

As **transferências de adiantamento** são tratadas como saída de caixa (equivalente a despesa), enquanto as **devoluções** são tratadas como entrada (equivalente a receita). Toda a criação do acerto ocorre dentro de uma **transação de banco de dados**, incluindo o vínculo dos itens (`acerto_id`) e o cálculo dos totais.

4.11.7 Vínculo de Transferências ao Motorista

As transferências são vinculadas a **veículos**, não diretamente a motoristas. Ao buscar as transferências elegíveis para um acerto, o sistema identifica todos os veículos que o motorista conduziu no período (via `veiculo_motorista_historico`) e então busca as transferências desses

veículos que ainda não possuem `acerto_id`.

4.11.8 Exclusão de Acerto (Liberação dos Itens)

A exclusão de um acerto utiliza soft-delete, mas antes de deletar, todos os itens vinculados (receitas, despesas e transferências) têm seu `acerto_id` zerado, ficando disponíveis para um novo fechamento. Todo o processo ocorre dentro de uma transação.

4.11.9 Dashboard Financeiro

O dashboard agrega os dados do mês selecionado (padrão: mês atual). Para o gráfico dos últimos 6 meses, as **devoluções são somadas às receitas** e os **adiantamentos são somados às despesas**, tratando transferências como equivalentes de seus respectivos fluxos financeiros.

4.12 Módulo: Portal de Compras (Cooca+ / SigaMC)

4.12.1 Visão Geral

Este módulo é um **painel de consulta** a dois sistemas externos de cooperativa. Não há lógica de negócio local além de filtragem e orquestração de chamadas às APIs. Os dados exibidos são inteiramente provenientes das APIs de terceiros e não são persistidos no banco do portal.

4.12.2 Escopo de Consulta por Usuário

A consulta à Cooca+ é feita pelos **grupos** do usuário logado. O sistema filtra apenas os grupos que possuem o campo `matricula_coocamais` preenchido e realiza uma consulta por matrícula para cada um deles. Grupos sem matrícula configurada são silenciosamente ignorados.

Exceção Master Admin: quando o usuário é Master Admin, é possível selecionar qualquer usuário do sistema e visualizar os dados dos grupos desse usuário-alvo, não do admin logado.

4.12.3 Operações Disponíveis por API

Operação	API	Parâmetro-chave	Finalidade
Limites do cooperado	Cooca+	<code>userId</code> (código do cooperado)	Consulta saldo e limites disponíveis

Operação	API	Parâmetro-chave	Finalidade
Reservas / Pedidos	Cooca+	<code>userId</code>	Lista pedidos em aberto (tipo 2, status 0)
Títulos a pagar	SigaMC	<code>matricula</code>	Consulta financeiro do cooperado na SigaMC

Ambas as APIs usam autenticação configurada via variáveis de ambiente (`services.coocamais.*` e `services.sigamc.*`). Falhas de comunicação são logadas e retornam um array com a chave `error` — o sistema não lança exceções para o usuário.

4.13 Módulo: Espelhamento (Compartilhamento Público de Frota)

4.13.1 Ciclo de Vida de um Espelhamento

Um espelhamento é um link público e temporário que expõe a posição em tempo real de um subconjunto de veículos sem exigir autenticação do visualizador. Seu ciclo de estados é:

```
Ativo (is_active=true + expires_at no futuro)
  → Expirado (is_active=true + expires_at no passado)
  → Desativado (is_active=false)
```

- A **exclusão** via interface é uma desativação lógica (`is_active = false`), não uma deleção física.
- Um espelhamento **expirado não pode ser reativado** diretamente; é necessário editar a data de validade antes de reativar.
- Cada acesso público incrementa o contador `access_count` e atualiza `last_accessed_at`.

4.13.2 Geração e Segurança do Token

O token de acesso público é uma string aleatória de **48 caracteres** (`Str::random(48)`), gerada no momento da criação. Não há rotação automática de token — ele é permanente para o ciclo de vida do espelhamento.

4.13.3 Resolução do Token

Em todos os endpoints públicos, o token é validado contra três condições simultâneas:

1. `token` existe na tabela.

2. `is_active = true`.
3. `expires_at > now()`.

Se qualquer condição falhar, a resposta é HTTP 410 (Gone) com mensagem de link inválido ou expirado.

4.13.4 Dados Expostos Publicamente

O endpoint público de veículos **sanitiza** os dados antes de retorná-los, expondo apenas: placa, marca, modelo, cor, ano, status de movimento, velocidade, latitude, longitude, nome do motorista, endereço e tecnologia do rastreador. Dados sensíveis de negócio (ID do rastreador, IMEI, ESN, linha telefônica, grupo, associação) são omitidos da resposta pública.

Veículos sem coordenadas GPS válidas são **filtrados** e não aparecem no mapa.

4.13.5 Funcionalidade de Geofence no Espelhamento

O criador do espelhamento pode habilitar geofences (áreas de alerta geográfico) para o link público. As regras são:

Autenticação de geofence:

- Protegida por uma senha numérica de **6 dígitos**, gerada aleatoriamente e armazenada com hash bcrypt. A senha em texto plano também é persistida para exibição ao criador.
- A sessão de autenticação tem **TTL de 30 minutos**. Após esse prazo, o visitante precisa autenticar novamente para editar geofences.
- Visitantes não autenticados podem **visualizar** as geofences, mas o e-mail de notificação é **maskado** (ex.: `j***@empresa.com.br`).

Limites e validações:

- Máximo de **20 geofences ativos** por espelhamento. Tentativas de criação além desse limite são rejeitadas com HTTP 422.
- Um geofence só pode ser criado para veículos que façam parte do mesmo espelhamento. Tentativas com veículos externos são rejeitadas.
- Ao editar um geofence, o estado `currently_inside` é **resetado para** `false`, evitando estados inconsistentes após alteração de coordenadas ou raio.

Cálculo de contenção: O sistema utiliza a fórmula **Haversine** (raio da Terra = 6.371.000 m) para calcular a distância entre a última posição do veículo e o centro da geofence. Um ponto é considerado "dentro" se a distância calculada for $\leq$ `raio_metros` configurado.

Resumo dos Domínios

Domínio	Status de Implementação	Nível de Lógica Interna
Jornada de Trabalho	Implementado e maduro	Alto — cálculos legislativos complexos
Telemetria / Score	Implementado e maduro	Alto — engine de scoring configurável
Rastreadores	Implementado	Médio — detecção de movimento e status
Acessos	Implementado	Médio — permissões granulares
Produtividade / SLA	Implementado	Médio — KPIs e SLA de analistas
Cadastros / Auditoria	Implementado	Médio — auditoria automática diária
Financeiro	Implementado (sem service layer)	Médio — ciclo de acerto bem definido
Portal Cooca+ / SigaMC	Implementado (gateway puro)	Baixo — proxy de APIs externas, sem persistência
Espelhamento	Implementado	Médio — ciclo de vida de link, geofence com auth própria
Câmeras	Implementado	Baixo-Médio — orquestração de hardware Jimi IoT