

# Documento de Entregáveis

**Projeto:** Plataforma de Auditoria e Mensuração de Transporte por Telemetria – Atvos

## Sprint 1 – Levantamento de Requisitos e Modelagem Inicial

**Período:** Semanas 1–2

### Entregáveis

#### 1. Requisitos Funcionais e Não Funcionais

Documento descrevendo as funcionalidades mínimas do MVP, incluindo:

- Ingestão de dados de telemetria via API (com fallback para upload de CSV em caso de indisponibilidade).
- Identificação automática de viagens.
- Cálculo automático do km variável.
- Consolidação de custos fixos e variáveis.
- Geração de relatórios operacionais D-1 (diários) e mensais.

Inclui requisitos não funcionais:

- Usabilidade da interface web.
- Governança e rastreabilidade dos dados (logs e trilha de auditoria).
- Segurança de acesso com perfis (Operação / Financeiro / Admin).
- Desempenho mínimo: processamento D-1 disponível até D+1.

#### 2. Mapeamento de Processos e Jornada do Usuário

Identificação dos atores principais:

- **Gestor de Transporte:** acompanhamento de viagens e validação de divergências.
- **Financeiro:** fechamento mensal e auditoria.
- **Operacional:** monitoramento diário.
- **Transportadora:** fornecimento de dados.

Definição das jornadas dentro da plataforma, como:

- Visualização de viagens D-1.
- Filtro de divergências >10%.
- Exportação de relatórios CSV/Excel.
- Geração de consolidados mensais.

### 3. Dicionário de Dados e Tabelas de Apoio

Inventário das fontes de dados (API de telemetria, boletins, arquivos da transportadora) e definição das tabelas auxiliares:

- Veículos (placa, tipo, unidade).
- Tarifas por km (rota / área / tipo).
- Custos fixos (mensal por veículo ou rota).
- Pols com geofencing (usina, frente, garagem, alojamento).
- Turnos (janelas de horário).
- Dicionário de rotas (palavras-chave → rota canônica).

### 4. Arquitetura Inicial da Solução

Documento técnico descrevendo a arquitetura:

- **Frontend:** Streamlit.
- **Backend:** Python (FastAPI ou Django).
- **Banco de dados:** PostgreSQL (MVP), com SQLite para protótipos locais.
- **Camadas de dados:** raw, stg, fact, agg.
- Estratégia de ingestão via API / CSV.
- Estratégia de deploy inicial e logging.

**Tecnologias:** Google Docs, Miro/Figma, Python, PostgreSQL/SQLite.

## Sprint 2 – Tratamento de Dados e Arquitetura Técnica

**Período:** Semanas 3–4

### Entregáveis

#### 1. Pipeline de API e Tratamento de Dados (*Marcos e João*)

Implementação de rotinas em Python para ingestão e pré-processamento:

- Leitura via CSV (simulação de API) e estrutura preparada para API real.
- Validação de schema.
- Normalização de campos (odômetro, timestamps, velocidade).
- Integração com banco de dados.

#### 2. Base de Dados Estruturada (*Murilo e Ícaro*)

Implementação do banco PostgreSQL (Supabase):

- **raw:** dados crus.
- **stg:** dados normalizados.
- **fact:** viagens, auditorias, consolidados.

- **agg:** agregações diárias e mensais.

Inclui:

- Tabelas auxiliares.
- Views consolidadas (D-1 e mensal).
- Rotinas de carga incremental.
- Políticas de retenção.
- Perfis de acesso (Operação / Financeiro / Admin).

### 3. Wireframe da Plataforma (*Isabelle*)

Protótipo navegável no Figma com:

- Login.
- Lista de viagens D-1.
- Filtros (data, unidade, placa).
- Detalhe da viagem.
- Exportação de relatórios.

**Tecnologias:** Python (pandas, geopandas), PostgreSQL, SQLAlchemy, APIs de telemetria.

## Sprint 3 – Plataforma Web (MVP Operacional)

**Período:** Semanas 5–6

### Entregáveis

#### 1. Interface Web Inicial (*Murilo e Isabelle*)

Aplicação em **Streamlit** com:

- Login e perfis.
- Lista de viagens D-1.
- Km por telemetria vs declarado.
- Flags de divergência.
- Filtros por unidade, placa, área e data.

#### 2. Mapa Interativo (*João*)

Visualização geográfica das rotas:

- Origem e destino por Pols.
- Diferenciação AGR x IND.
- Destaque visual para rotas divergentes.

#### 3. Cadastro e Upload de Tabelas de Apoio (*Ícaro*)

Interface para:

- Cadastro de veículos, tarifas, custos fixos, Pols e turnos.
- Upload de CSV para atualização em lote.

#### **4. Exportação CSV D-1 (Marcos)**

Relatório diário consolidado com:

- Km telemetria.
- Km declarado.
- Divergências (%).
- Flags.
- Custo variável.

**Tecnologias:** Streamlit, Python (FastAPI/Django), Plotly/Leaflet, PostgreSQL.

## **Sprint 4 – Relatórios e Outputs de Fechamento**

**Período:** Semanas 7–8

### **Entregáveis**

#### **1. Relatório Mensal Consolidado**

Agregação por:

- Veículo.
- Rota.
- Unidade.

Inclui:

- Custos fixos e variáveis.
- Divergências.
- Score de auditoria por viagem e frota.

#### **2. Pacote de Medição – Layout Atvos**

Exportação em Excel/CSV no layout oficial da Atvos para fechamento mensal.

#### **3. Painel Financeiro Mensal**

Dashboard em **Streamlit** com KPIs:

- Custo por km.
- Proporção fixo x variável.
- Ranking de transportadoras.

- Top divergências por unidade e rota.

#### 4. Alertas de Divergência (Opcional MVP)

Alertas automáticos para viagens com divergência >10%.

**Tecnologias:** Streamlit, Python (pandas, openpyxl/xlsxwriter), PostgreSQL.

## Sprint 5 – Testes, Validação e Refinamento

**Período:** Semanas 9–10

### Entregáveis

#### 1. Testes de Usabilidade e Validação Funcional

Testes com gestores de transporte e financeiro, com ajustes baseados em feedback.

#### 2. Refinamento da Interface e Relatórios

Melhorias de UX, filtros e padronização de relatórios exportados.

#### 3. Treinamento e Documentação

- Manual detalhado (PDF).
- Guia rápido (1 página).
- Treinamento remoto/presencial.

#### 4. Roadmap Pós-MVP

Documento com propostas futuras:

- Integração com ERP.
- App mobile para motoristas.
- Auditoria avançada com machine learning.

**Tecnologias:** Python (pytest/unittest), Streamlit, Exportação PDF/Excel.

## Resumo Geral

Sprint	Foco Principal	Entregáveis-chave	Duração
1	Requisitos + Modelagem	Jornadas, arquitetura, dicionário de dados	2 semanas
2	Dados + Pipeline	ETL, banco estruturado, wireframes	2 semanas

3	Plataforma Web	Streamlit MVP, mapas, exportações	2 semanas
4	Fechamento	Relatórios mensais, KPIs, pacotes Atvos	2 semanas
5	Validação	Testes, treinamento e roadmap	2 semanas