

Fleets Flow — Documento Definitivo do MVP

Operação de Cabotagem de Granéis Líquidos (Combustíveis) a partir do Porto do Itaqui (São Luís–MA)

0) Contexto operacional do MVP (por que Itaqui + combustíveis)

O Porto do Itaqui é publicamente apresentado como um hub relevante para movimentação e distribuição de derivados de petróleo (gasolina, diesel, QAV, GLP), com papel logístico estratégico e movimentação expressiva de graneis líquidos. [Serviços e Informações do Brasil+2Porto do Itaqui+2](#) Em infraestrutura, o porto possui múltiplos berços com profundidades informadas na faixa de 12 a 19 metros, permitindo atracação de navios de grande porte, o que favorece operações recorrentes de graneis líquidos e entreposto. [Porto do Itaqui+1](#)

Objetivo do MVP: validar ponta a ponta a proposta do Fleets Flow em um cenário realista e frequente: **afretamento → pré-compliance → programação e documentos da operação → carregamento → saída → chegada ao destino → fechamento contratual e evidência auditável.**

1) Proposta de valor do MVP (o que o desenvolvedor precisa ter claro)

1.1 O que o Fleets Flow faz (no MVP)

1. **Garante elegibilidade (Chaves Verdes) para operar e participar do Marketplace:**

- **Chave Verde da Empresa (Org Key)**
- **Chave Verde do Ativo/Embarcação (Asset/Vessel Key)**

2. **Evita “revalidação diária” de documentos:**

- Documentos têm validade longa → o sistema controla vigência e alerta (90/60/30)
- Gate “direito de operar” ocorre em pontos-chave (login e emissão de chave)

3. **Emite Chave Única da Operação (auditável):**

- Hash SHA-256 (64 chars) do snapshot da operação
- Assinatura ICP-Brasil A3 (Certisign SDK)
- Rastreabilidade e auditoria por entidade fiscalizadora/reguladora

4. Executa o fluxo “Uber-like” de contratação spot:

- Demanda criada por contratante (só com Chave Verde da Empresa)
- Oferta feita por contratado (só com Chave Verde da Empresa + Ativo)
- Chat + termos estruturados + assinatura + geração de documentos

1.2 O que o MVP não precisa (para funcionar)

- API para validar todos os documentos diariamente (não existe para a maioria e não é eficiente)
 - Integração total com todos os sistemas governamentais no dia 1 (usar Gate com provedores disponíveis/contratados e política de fallback)
-

2) Escopo do MVP — cenário simulado de cabotagem (combustíveis)

2.1 Participantes (atores)

- **Contratante:** distribuidora/trader/operador logístico de combustíveis ([PREENCHER entidade])
 - **Armador/Operador** (Contratado): empresa que disponibiliza embarcação para cabotagem ([PREENCHER])
 - **Ativo:** navio-tanque / barcaça tanque / unidade compatível ([PREENCHER tipo e IMO/MMSI])
 - **Porto Origem:** Porto do Itaqui (São Luís–MA)
 - **Porto Destino:** [PREENCHER — ex.: porto no NE/N]
 - **Órgãos/sistemas** (quando aplicável): PSP, Mercante, ANTAQ/SAMA etc. (com acesso conforme ACT/TCT e escopo)
-

3) Monetização sugerida (aberta para debate)

Não é “uma” forma. São camadas; o MVP já pode testar 2–3 sem travar produto.

3.1 SaaS (recorrência)

- **Plano por Empresa (contratante/contratado):** usuários, permissões, limites
- **Plano por Ativo (embarcação):** número de embarcações com Asset Key ativa

3.2 Cobrança por uso (alto alinhamento com valor)

- **Fee por emissão de Chave Única** (por operação)
- **Fee por assinatura/contrato gerado** (quando houver geração documental)

3.3 Marketplace (take rate)

- **Taxa por match fechado** (success fee)
- **Taxa por anúncio destacado / prioridade no ranking** (feature premium, cuidado com governança)

3.4 Serviços complementares (opcional)

- “Validação humana assistida” (fila de revisão)
 - Onboarding documental e parametrização por tipo de embarcação/operação
-

4) Guia do Desenvolvedor — Fluxo ponta a ponta (passo a passo)

4.1 Pré-requisitos estruturais (antes da primeira operação)

4.1.1 Cadastro e Chaves

1. Cadastro da Empresa Contratante

- CNPJ, responsáveis, usuários, papel “Demand Creator”
- Emite/atualiza **Org Key** (Chave Verde da Empresa)
 - GREEN exige: cadastro mínimo + requisitos críticos internos OK

2. Cadastro da Empresa Contratada (Armador/Operador)

- CNPJ, responsáveis, usuários, papel “Supplier”
- Emite/atualiza **Org Key**

3. Cadastro do Ativo (embarcação)

- IMO/MMSI/nome/tipo/bandeira
- Vinculado à empresa operadora
- Emite/atualiza **Asset Key** (Chave Verde do Ativo)

4.1.2 Documentos com validade (controle interno)

Para cada documento:

- tipo, emissor, emissão, validade, criticidade (CRÍTICO/RELEVANTE/INFO)
- status (VALID/EXPIRING/EXPIRED/PENDING_REVIEW)
- alertas automáticos 90/60/30

MVP: a “verdade” é o controle interno + auditoria. APIs externas entram como Gate, não como validação contínua.

4.2 Login (Gate operacional de “direito de operar”)

Objetivo: não travar login por indisponibilidade externa, mas condicionar emissão de chave.

Fluxo

1. Usuário entra com **usuário/senha**
 2. Sistema valida usuário e permissões
 3. Se o perfil exige Gate:
 - chama provedores configurados (ex.: Mercante, SAMA, consulta cadastral quando houver)
- [PREENCHER provedores habilitados no MVP]**
4. Resultado do Gate:
 - PASS / WARN / FAIL
 5. Regras:
 - PASS → pode operar e pode emitir chave (se docs internos ok)
 - WARN → pode operar, mas chave tende a AMARELA ou exige revisão humana
 - FAIL → login pode ser restrito; emissão de chave bloqueada
-

4.3 Afretamento e criação da demanda (Marketplace Spot)

Passo 1 — Contratante cria a demanda (só com Org Key GREEN)

Tela/rota: `Criar Demanda`

Campos mínimos do MVP:

- Tipo: **CABOTAGEM — Granéis Líquidos (Combustíveis)**
- Origem: Itaqui (UN/LOCODE **[PREENCHER]**)

- Destino: **[PREENCHER]**
- Janela (início/fim): **[PREENCHER]**
- Tipo de embarcação: navio-tanque / barcaça tanque (conforme catálogo)
- Requisitos hard:
 - capacidade mínima **[PREENCHER]**
 - calado/limite **[PREENCHER]**
 - certificações mínimas (por criticidade) **[PREENCHER]**
- Política de preço:
 - A (preço) ou C (sem preço) para MVP

Ao publicar:

- demanda vira **OPEN**
- dispara **Matching**

Passo 2 — Matching e notificação (somente elegíveis)

O sistema filtra **apenas** embarcações com:

- **Org Key (contratado) = GREEN**
- **Asset Key (embarcação) = GREEN**
- Requisitos hard atendidos
- Disponibilidade (declarada) compatível

Notifica via canais do MVP: e-mail/push/WhatsApp **[PREENCHER]**

Registra trilha: quem recebeu, quando, por qual critério.

Passo 3 — Propostas e shortlist

Contratado envia:

- proposta (preço e/ou janela alternativa)
- termos operacionais básicos

Contratante:

- aceita uma proposta → cria **Negociação** e abre **Chat**

4.4 Negociação (Chat) + Termos estruturados + “Fechar”

Passo 4 — Chat com trilha auditável

Chat deve ter:

- mensagens
- anexos
- botão “Propor Termos” (campos estruturados) para evitar contrato “por conversa”

Passo 5 — Revalidação antes de fechar (obrigatória)

No clique “Fechar e assinar”:

- Revalidar imediatamente:
 - Org Key (contratante)
 - Org Key (contratado)
 - Asset Key (embarcação)
- Se qualquer um não estiver GREEN:
 - bloquear fechamento (ou permitir com chave AMARELA/VERMELHA conforme política)

Passo 6 — Assinatura e geração documental

- Assinatura ICP-Brasil A3 via Certisign SDK
 - Geração de documentos:
 - Contrato / Ordem de Serviço / Carta-frete (templates)
 - Anexos operacionais (SOW)
 - PDF final:
 - hash
 - QR para consulta auditável
-

4.5 Emissão da Chave Única da Operação (o “selo” do MVP)

Quando emitir

- após contrato assinado **ou**
- após “fechamento operacional” (política do MVP) [**PREENCHER**]

Como emitir (algoritmo sem código)

1. Criar **Operation Snapshot** (JSON imutável) contendo:

- IDs das partes
- embarcação (IMO/MMSI)
- rota (Itaqui → destino)
- janela
- requisitos
- resultado do Gate (PASS/WARN/FAIL + evidências)
- hash do conjunto documental interno (docSetHash)
- carimbo de tempo

2. Canonizar snapshot e gerar:

- `unique_key = SHA256(snapshot) → 64 chars`

3. Assinar com A3:

- `signature = Sign_A3(unique_key)`

4. Salvar:

- `unique_key`
- assinatura
- metadados do certificado
- snapshot id
- status tricolor:
 - GREEN: Gate PASS + docs críticos OK
 - YELLOW: warn ou vencimento próximo
 - RED: fail ou doc crítico vencido

4.6 Execução operacional (carregamento, saída, chegada)

No MVP, o foco não é automação de toda autoridade portuária; é **rastreabilidade + checklist + evidências**.

Passo 7 — Preparação para carregamento (Itaqui)

- Checklist operacional “pré-berço”
- Confirmação de janela e berço (informação manual no MVP)

- Documentos operacionais específicos anexados (quando necessário)

Passo 8 — Carregamento

- Registro de evento: `LOADING_START`, `LOADING_END`
- Registro de volumes (mínimo): **[PREENCHER]**
- Anexos: B/L e documentos comerciais (quando existirem) **[PREENCHER]**
- Logs e auditoria

Passo 9 — Saída

- Evento `DEPARTURE_CONFIRMED`
- Atualiza status da operação
- Se houver integração PSP/Mercante no MVP: registrar `externalRefs` **[PREENCHER]**

Passo 10 — Chegada ao destino

- Evento `ARRIVAL_CONFIRMED`
- Checklist de descarga
- Evento `DISCHARGE_START/END`
- Encerramento `OPERATION_COMPLETED`

Passo 11 — Fechamento e evidências finais

- pacote de evidências:
 - contrato assinado
 - chave única (hash + assinatura)
 - timeline de eventos
 - anexos
- export auditável (PDF/JSON) para auditor/fiscalizador/homologador

5) Estados do sistema (para o dev não errar)

5.1 Demanda

OPEN → MATCHING → NEGOTIATING → AWARDED → CONTRACT_SIGNED → COMPLETED
(CANCELLED / EXPIRED)

5.2 Proposta

SENT → COUNTERED → ACCEPTED/REJECTED → EXPIRED

5.3 Negociação

OPEN → SIGNING → SIGNED (ou CANCELLED)

5.4 Operação

PLANNED → LOADING → SAILED → ARRIVED → DISCHARGING → COMPLETED

6) Criticidade documental (padrão para cabotagem combustíveis — MVP)

CRÍTICO (bloqueia **GREEN**) — exemplos típicos:

- Conformidades de segurança/gestão (ISM/ISPS quando aplicável)
- Certificados essenciais do navio/classe (conforme matriz)
- Seguro/condição contratual obrigatória [**PREENCHER**]

RELEVANTE (gera alertas e pode virar **YELLOW**)

- Planos ambientais e operacionais
- Documentos de apoio de fiscalização

INFO

- anexos internos e evidências complementares

A matriz final por tipo de embarcação e operação fica como tabela `document_types` com `rules_json` (parametrizável), mas o MVP já roda com um conjunto mínimo.

7) Integrações no MVP (clareza para engenharia)

Para o MVP, recomendo formalmente classificar integrações em:

- **Obrigatórias (MVP):** nenhuma integração crítica que derrube operação
- **Desejáveis (MVP):** 1 provedor de Gate (ex.: Mercante ou SAMA) [**PREENCHER conforme acesso**]
- **Fase 2:** demais integrações e automações

Isso evita “MVP refém de API”.

8) Entregáveis práticos para o time de desenvolvimento (checklist)

1. Catálogo inicial:
 - vessel types
 - operation types
 - document types + criticidade + prazos 90/60/30
 2. Módulos:
 - Auth + RBAC
 - Org Key engine
 - Asset Key engine
 - Document lifecycle + alerts
 - Marketplace (demand/proposal/negotiation)
 - Chat + termos estruturados
 - Contract generator (templates)
 - Unique Key issuance + A3 signing
 - Audit logs + export
 3. Painéis mínimos:
 - Dashboard do contratante (demandas/propostas/contratos)
 - Dashboard do contratado (oportunidades/negociações)
 - Pannel de compliance (chaves + documentos vencendo)
-

9) Campos que você precisa preencher para fechar a simulação como “real”

- Porto destino do MVP: [**PREENCHER**]

- Tipo exato de embarcação e requisitos técnicos: [PREENCHER]
 - Política de assinatura: quem assina e em quais etapas: [PREENCHER]
 - Provedor(es) de Gate no MVP (ex.: Mercante/SAMA): [PREENCHER]
 - Templates documentais mínimos (contrato/carta-frete/OS): [PREENCHER]
-

Referências factuais usadas para o contexto Itaqui (2024–2025)

- Hub e movimentação de combustíveis/derivados e papel estratégico (governo federal) [Serviços e Informações do Brasil](#)
 - Resultados e relevância do Itaqui em 2024 e combustível como destaque (com base em estatístico aquaviário citado) [Porto do Itaqui+1](#)
 - Indicadores recentes com granéis líquidos e volumes em 2025 (indicadores do Porto do Itaqui) [Porto do Itaqui](#)
 - Infraestrutura e profundidades de berços (12–19m) [Porto do Itaqui](#)
 - Crescimento e recordes com combustíveis e granéis líquidos (comunicados oficiais do Porto do Itaqui) [Porto do Itaqui](#)
-

Se você me enviar [PORTO DESTINO] e o tipo de embarcação (navio-tanque? barcaça tanque?), eu transformo este documento em uma “linha do tempo de operação” com dados preenchidos, incluindo: requisitos hard do match, conjunto documental crítico do MVP e os eventos operacionais exatos (timestamps e estados) para o dev implementar sem ambiguidades.

agora faça uma simulação narrada desta operação, passo a passo, bem detalhada, para que o desenvolvedor possa imaginar melhor como seria na prática.

A seguir vai uma **simulação narrada, passo a passo e bem detalhada** da operação do MVP: **cabotagem de combustíveis (granéis líquidos)** saindo do **Porto do Itaqui (São Luís–MA)**, usando o **Fleets Flow na prática**.

(É uma narrativa operacional para o desenvolvedor “ver o filme” e entender onde cada funcionalidade entra.)

Simulação narrada — MVP Fleets Flow

Cabotagem de Combustíveis (Granel Líquido) — Itaqui → Destino

Personagens (exemplo)

- **Contratante:** *NordFuel Distribuidora* (empresa que precisa redistribuir diesel para outro porto)
- **Contratado/Armador:** *Atlântico Cabotagem* (empresa que oferece embarcação)
- **Embarcação:** *MT LITORAL BRAVO* (navio-tanque de cabotagem)
- **Destino:** *Porto X* (neste MVP, o destino pode ser parametrizável; aqui tratamos como “Porto X”)
- **Usuários:**
 - *Camila* (Contratante / Demand Creator)
 - *Rafael* (Contratado / Fleet Ops)
 - *Ana* (Compliance da Atlântico Cabotagem)

Regra estrutural ativa: Marketplace só funciona quando:

- Contratante: **Chave Verde da Empresa** = GREEN
 - Contratado: **Chave Verde da Empresa** = GREEN e **Chave Verde do Ativo** = GREEN
-

ETAPA 0 — Preparação (dias antes)

0.1 Onboarding da NordFuel (contratante)

Camila entra no Fleets Flow para cadastrar a empresa.

- Ela cadastra CNPJ, razão social, responsáveis e usuários
- O sistema roda o checklist mínimo interno (cadastro completo + permissões)
- A **Chave Verde da Empresa (NordFuel)** é emitida como:
 - **GREEN**
- No painel, a NordFuel vê:
 - “✓ Empresa apta a criar demandas”

Para o dev: aqui o sistema gera **Org Key** e grava status, data, e motivo.

0.2 Onboarding da Atlântico Cabotagem (contratado)

Rafael e Ana cadastram a empresa e a embarcação.

Empresa (Atlântico)

- Cadastro + responsáveis
- Documentos corporativos críticos (ex.: procuração, dados de faturamento, etc. — parametrizável)
- Resultado: **Chave Verde da Empresa = GREEN**

Ativo (MT LITORAL BRAVO)

- IMO, MMSI, tipo “Navio-tanque”, bandeira
- Upload dos documentos do navio (com validade longa)
- O sistema classifica cada documento por criticidade
- Para cada documento o sistema registra:
 - issued_at, expires_at
 - status: VALID
 - alertas: 90/60/30

Se tudo crítico estiver ok:

- **Chave Verde do Ativo (navio) vira GREEN**

Para o dev: isso é o “motor de chave” (Org Key + Asset Key) + “document lifecycle”.

0.3 Alertas automáticos

Ana vê no painel:

- “Nenhum documento crítico vence nos próximos 90 dias”
- (Se existisse algo vencendo em 60 dias, já apareceria amarelo e acionaria alerta)

ETAPA 1 — Login no dia da operação (Gate operacional)

1.1 Camila (NordFuel) faz login

Camila digita usuário e senha.

- O sistema autentica
- Verifica Org Key da NordFuel: **GREEN**

- Como Camila vai **criar demanda**, o sistema roda o “Gate operacional” rápido:
 - se houver provedor habilitado: consulta de status (PASS/WARN/FAIL)
 - se não houver: resultado “WARN” (sem bloquear login)

Neste caso:

- Gate retorna **PASS** (ou **WARN** se MVP sem provedor)
- Camila entra no painel

Para o dev: o Gate não derruba login; ele condiciona “emissão de chave” e “marketplace”.

ETAPA 2 — Afretamento spot no Marketplace (modelo Uber)

2.1 Criar demanda (contratante)

Camila clica em “**Marketplace → Criar Demanda**”.

Ela preenche:

- Tipo de operação: **CABOTAGEM**
- Carga: **Diesel S10 (granel líquido)**
- Origem: **Porto do Itaqui (São Luís–MA)**
- Destino: **Porto X**
- Janela operacional:
 - início: amanhã 08:00
 - fim: amanhã 20:00
- Requisitos técnicos:
 - tipo: navio-tanque
 - capacidade mínima: 8.000 m³
 - calado máximo: 11,5m (por restrições do destino, por exemplo)
 - exigências adicionais: “ISPS ativo”, “seguro vigente” (parametrizável)
- Preço:
 - modo A (oferece preço) ou modo C (sem preço)
 - aqui: **oferece R\$ [PREENCHER]** e permite contra-proposta

Camila clica “**Publicar demanda**”.

O sistema:

- `cria Demand #000231`

- status: **OPEN**
 - TTL: 12h
 - dispara “Matching”
-

2.2 Matching (filtro duro) + notificação

O motor procura embarcações elegíveis:

Filtro duro (hard)

- Empresa contratada Org Key = GREEN
- Embarcação Asset Key = GREEN
- Tipo navio-tanque
- Janela disponível (agenda declarada)
- Requisitos técnicos atendidos

A Atlântico Cabotagem passa no filtro:

- Org Key = GREEN
- Asset Key do MT LITORAL BRAVO = GREEN
- Capacidade atende
- Está na região / pode chegar no Itaqui a tempo

O sistema notifica o Rafael:

- push/email: “Nova demanda compatível: Demand #000231”

Registra no log:

- quem foi notificado
- por qual critério
- timestamp

Para o dev: `market_demand_targets` com trilha de auditoria.

ETAPA 3 — Proposta e contra-proposta

3.1 Rafael abre a demanda e envia proposta

Rafael entra na demanda #000231 e vê:

- requisitos
- janela
- preço oferecido (se modo A)

Ele envia:

- Aceite com condições, ou contra-proposta
- Exemplo:
 - preço: R\$ [PREENCHER] (contra-proposta)
 - janela: mantém
 - observação: “precisa confirmação de berço até 18:00 de hoje”

O sistema cria `Proposal` #P-771:

- status `SENT`
 - TTL da proposta: 4h
-

3.2 Contratante seleciona e abre negociação

Camila recebe a proposta no painel.

Ela:

- aceita a proposta `P-771`
- status demanda muda para **NEGOTIATING**
- sistema cria negociação `N-180`
- cria sala de chat `CHAT-180`

Para o dev: amarração `demand -> negotiation -> chat`.

ETAPA 4 — Chat + termos estruturados

4.1 Conversa no chat

No chat, Camila e Rafael alinham:

- horário de apresentação do navio
- instruções do terminal
- condições de estadia
- forma de pagamento
- penalidades por atraso

O Fleets Flow não deixa isso “solto”:

há um botão “**Propor Termos**” (campos estruturados):

- Preço final
- Condições de pagamento
- Responsável por demurrage (se aplicável)
- Janela definitiva
- Cláusulas operacionais

Ao salvar, o sistema cria uma **versão** dos termos:

- `terms_v1, terms_v2 ...`
 - tudo auditável
-

4.2 “Fechar e assinar” (ponto crítico)

Camila clica em “**Fechar e assinar**”.

Antes de permitir:

o sistema revalida automaticamente:

- Org Key NordFuel: GREEN?
- Org Key Atlântico: GREEN?
- Asset Key MT LITORAL BRAVO: GREEN?
- Documentos críticos: todos válidos?
- Gate operacional (se habilitado): PASS?

Se tudo ok:

- libera assinatura

Se algo mudou (ex.: documento expirou no meio do dia):

- bloqueia ou rebaixa para “amarela/vermelha”, conforme política

ETAPA 5 — Assinatura ICP + geração de documentos

5.1 Assinatura

O sistema solicita assinatura A3:

- Camila assina pelo lado do contratante
- Rafael ou representante legal assina pelo contratado

O Fleets Flow registra:

- certificado (serial/subject/issuer)
- timestamp
- assinatura

5.2 Geração documental

Com as assinaturas:

o sistema gera automaticamente:

- Contrato/Ordem de Serviço
- Anexo operacional (SOW)
- Carta-frete (se aplicável ao modelo)
- PDF final com:
 - hash SHA-256 do documento
 - QR auditável

Status:

- negociação: SIGNED
- contrato: SIGNED

ETAPA 6 — Emissão da Chave Única da Operação

6.1 Geração do snapshot

O sistema monta o **Operation Snapshot**:

- IDs das partes
- navio (IMO/MMSI)
- rota Itaqui → Porto X
- janela
- termos finais
- resultado do Gate
- hash do conjunto documental crítico (docSetHash)
- timestamp

6.2 Chave única

- Canoniza JSON
- SHA-256 → 64 caracteres
- Assina o hash via A3
- Armazena no banco

A plataforma exibe:

- **Chave Única:** **A1B2 . . .**
- Status: **GREEN**
- QR: “Validar operação”

Para o dev: isso é o coração auditável do produto.

ETAPA 7 — Operação no Itaqui (carregamento)

7.1 “Check-in operacional”

No dia seguinte, Rafael abre a operação e marca:

- “Embarcação chegou ao fundeio”
- evento: `ARRIVED_AT_ANCHORAGE`

O sistema solicita evidências mínimas (MVP):

- horário
- posição (manual ou AIS se houver)
- observação

7.2 Atracação e início do carregamento

Quando o navio atraca:

- evento: `BERTHED`

Quando inicia carregamento:

- `LOADING_START`
- anexos opcionais: instrução de carregamento, checklist terminal

Durante carregamento:

- o operador registra volume parcial (opcional no MVP)

Ao finalizar:

- `LOADING_END`
 - volume final: **[PREENCHER]**
 - o sistema tranca o volume como dado de operação (auditável)
-

ETAPA 8 — Saída do Itaqui (zarpe)

Rafael registra:

- `DEPARTURE_CONFIRMED`
- timestamp e evidência

Se houver `externalRefs` (PSP/Mercante etc.):

- grava protocolo/referência (se disponível no MVP)
- se não houver, registra “não aplicável no MVP”

ETAPA 9 — Trânsito e chegada ao destino

Durante a viagem:

- (MVP pode ser manual) status “Em trânsito”
- opcional: integração AIS para posição

Ao chegar:

- `ARRIVAL_CONFIRMED`
 - `DISCHARGE_START`
 - `DISCHARGE_END`
-

ETAPA 10 — Encerramento e pacote de evidências

Rafael marca:

- `OPERATION_COMPLETED`

O Fleets Flow gera automaticamente:

- Pacote de evidências:
 - contrato assinado
 - chave única (hash + assinatura)
 - linha do tempo de eventos
 - anexos
 - QR de validação

Camila (contratante) baixa o pacote para auditoria interna.

O que o dev deve “ver” claramente nessa simulação

Pontos onde a plataforma cria valor (e deve existir no MVP)

1. **Chaves Verdes separadas** (empresa + ativo) como pré-requisito do marketplace

2. **Matching só para elegíveis** (evita notificar irregular)
3. **Chat + termos estruturados** (evita contrato informal e risco jurídico)
4. **Assinatura ICP A3** como ato formal
5. **Chave única auditável** (snapshot + hash + assinatura)
6. **Linha do tempo de eventos** (carregamento/saída/chegada) com trilha auditável
7. **Alertas 90/60/30** para documentos com validade longa (sem “API diária”)