

Growth Equestre (MVP)

Guía de despliegue y uso

VS Code + Docker Compose + GitHub | Equipo Hackathon | 2026-02-09

Repositorio (GitHub)	https://github.com/brodyandre/growth_equestre_hackathon_2026
UI (Streamlit)	http://localhost:8501
Backend (API)	http://localhost:3000 (health: /health)
Servicios (Docker)	db + backend + scoring + ui_admin (según docker-compose.yml)

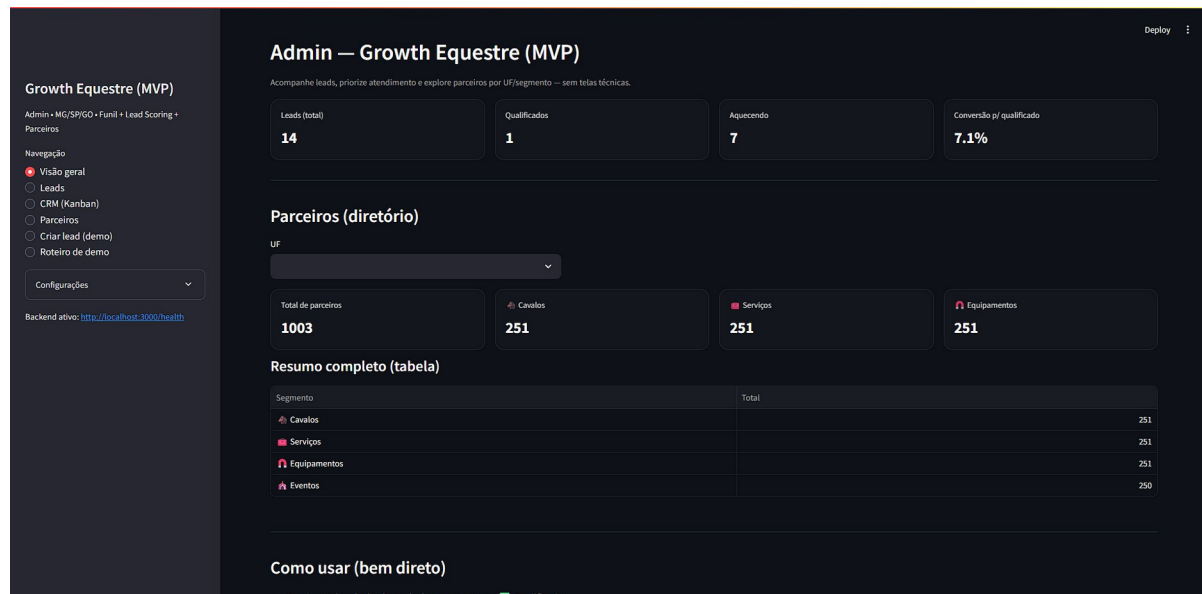


Figura 1 — Panel de “Visão geral” (KPIs + resumen de parceiros).

1. Contexto del desafío

Este proyecto se centra en el lado de “Crecimiento”: identificar compradores y socios potenciales usando ganchos de contenido (hooks), funnels automatizados y/o extracción de datos. El objetivo práctico es convertir a un “visitante casual” en un “lead calificado” para activos de alto valor (por ejemplo: un caballo de US\$ 50.000 o una silla de US\$ 2.000).

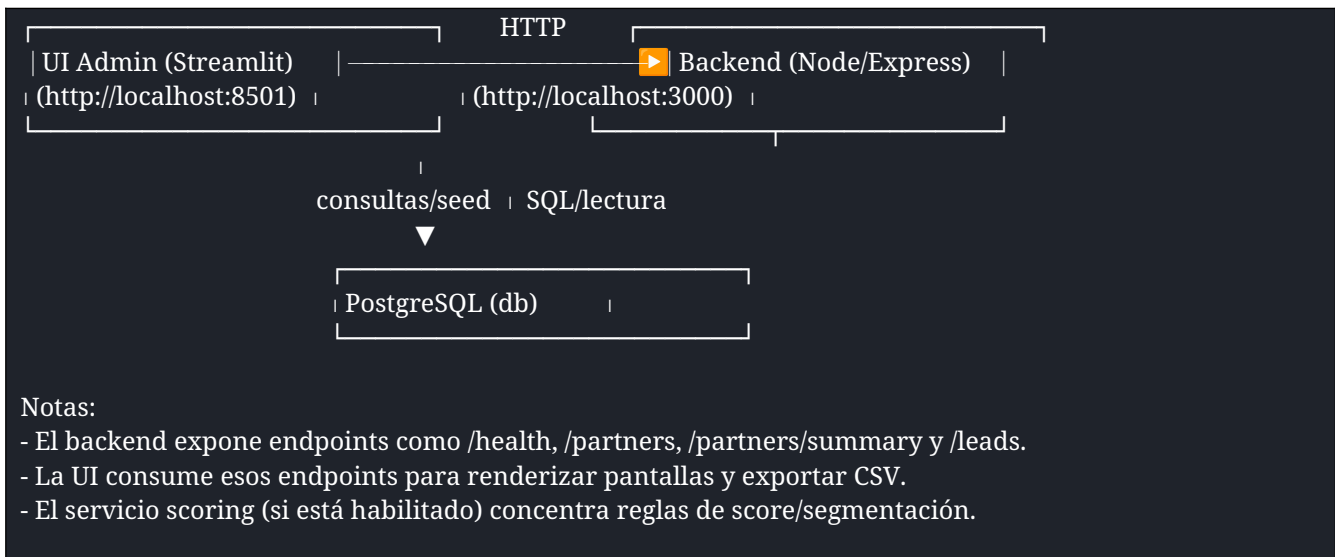
2. Qué entrega la aplicación

- Captura y gestión de leads (con estado y puntaje/score).

- Directorio de parceiros (posibles socios/compradores) filtrable por UF/segmento y exportable a CSV.
- CRM estilo Kanban para operar el funnel (mover etapas, registrar notas y próxima acción).
- Matching lead → parceiro (sugerencias automáticas) usando segmento, UF y, cuando exista, CNAE.
- Modo demo para crear escenarios y presentar el pitch sin depender de datos externos.

3. Arquitectura (cómo se conectan las partes)

La solución está dividida en componentes simples (mantenibles en hackathon) y conectados por HTTP:



4. Despliegue local en VS Code (Windows) — recomendado para el equipo

A continuación se describe el flujo de trabajo estándar para que cualquier miembro del equipo pueda levantar todo en su máquina. Los comandos son para PowerShell.

4.1 Requisitos

- Windows 11 + Docker Desktop (con WSL2 habilitado).
- VS Code + extensión Docker.
- Git.
- Python 3.10+ (para Streamlit/tools) y Node 20 (si vas a correr backend fuera de Docker).

4.2 Clonar el repositorio y abrirlo en VS Code

```

cd C:\Users\USER\Documents\Repositorios
git clone https://github.com/brodyandre/growth_equestre_hackathon_2026.git
cd .\growth_equestre_hackathon_2026
code .
  
```

4.3 Levantar la infraestructura con Docker Compose

Este método levanta base de datos y servicios del backend con un comando. Asegúrate de ejecutar desde la raíz del repositorio (donde está docker-compose.yml).

```
docker compose pull
docker compose up -d --build db backend scoring ui_admin

# Ver estado
docker compose ps

# Ver logs en vivo (ejemplo)
docker compose logs -f backend
```

4.4 Verificar que todo está OK

- Backend: abrir `http://localhost:3000/health` (o usar curl).
- UI: abrir `http://localhost:8501`.
- Si la UI indica “Backend ativo”, la integración está correcta.

```
curl http://localhost:3000/health
```

4.5 (Alternativa) Ejecutar la UI Streamlit fuera de Docker

Si en tu docker-compose no existe el servicio `ui_admin`, o prefieres correr la UI en modo dev, puedes levantar Streamlit manualmente.

```
cd .\ui_admin # (ajusta la carpeta según tu repo)
python -m venv .venv
.\venv\Scripts\Activate.ps1
python -m pip install -r requirements.txt
streamlit run main.py --server.port 8501
```

5. Uso de la aplicación (flujo operativo)

La app está pensada para operar el ciclo completo: captar leads → calificar → organizar en CRM → encontrar parceiros compatibles → exportar listas para prospección.

5.1 Visión general

- Muestra KPIs (total de leads, calificados, calentando, conversión).
- Muestra resumen del directorio de parceiros por segmento y total.
- Útil para “abrir” el pitch en el demo.

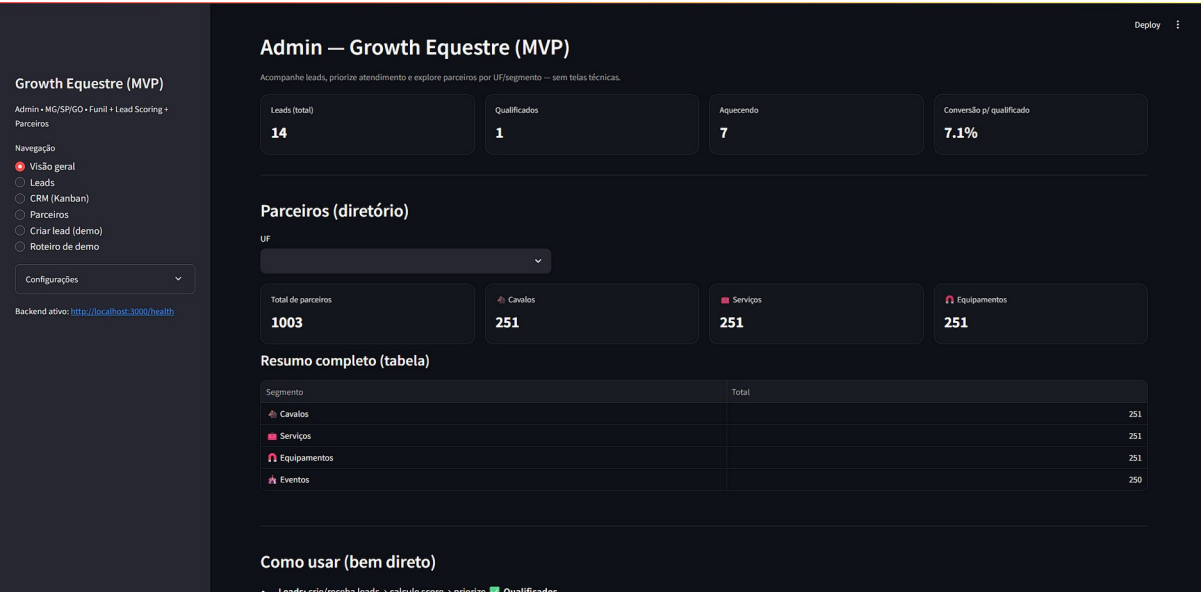


Figura 2 — Visión general (KPIs).

5.2 Leads (filtro + score + estado)

- Filtra por Status, score mínimo, UF y segmento.
- La columna “Motivos (resumo)” explica por qué el lead recibió ese score (explicabilidad).
- Exporta CSV con los leads filtrados para trabajar fuera de la app.

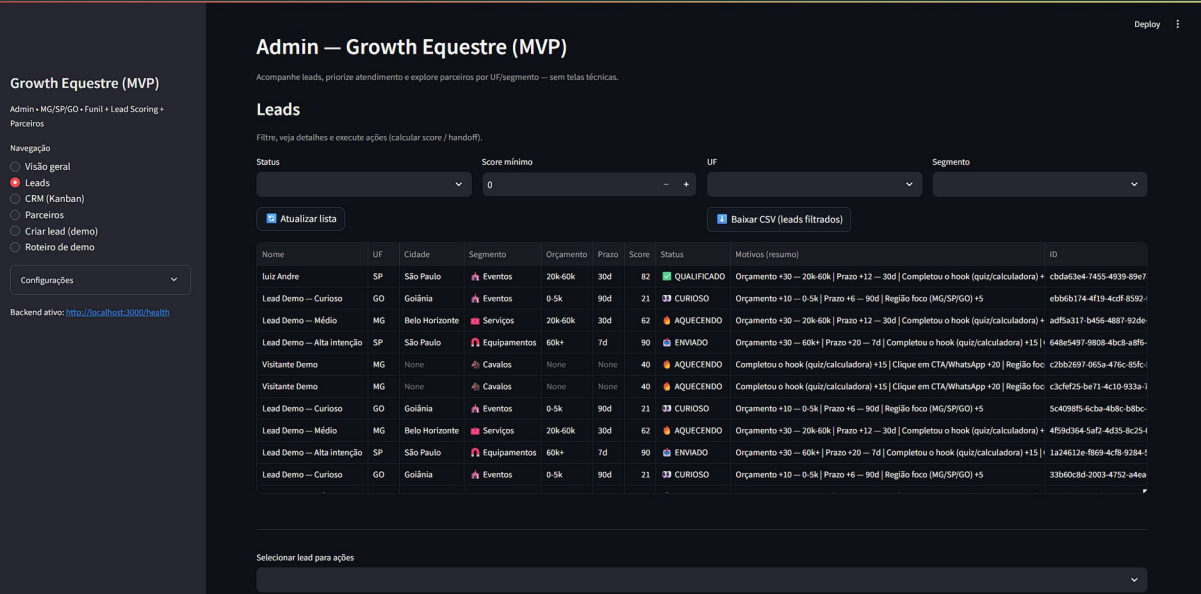


Figura 3 — Pantalla de Leads (lista, filtros y motivos del score).

5.3 Crear lead (demo)

- Crea un lead de demostración con UF/segmento y datos mínimos.

- Permite simular acciones del funnel (visita, completar quiz/calculadora, click CTA/WhatsApp) para que el score tenga sentido.
- Ideal para presentar el concepto de “visitante – lead calificado” en el pitch.

Admin — Growth Equestre (MVP)

Acompanhe leads, priorize atendimento e explore parceiros por UF/segmento — sem telas técnicas.

Criar lead (demo)

Crie um lead de demonstração e (opcionalmente) simule ações do funil.

Nome: UF:

WhatsApp (opcional): Cidade (opcional):

E-mail (opcional): Segmento de interesse:

Faixa de orçamento: Prioridade:

Simular ações do funil (para o score fazer sentido)

☒ Visita (page view) ☒ Completou o quiz/calculadora (hook) ☒ Clique no CTA/WhatsApp

Figura 4 — Crear lead (demo) + simulación del funnel.

5.4 CRM (Kanban) — operación del funnel

- Organiza leads por etapa (Curioso, Aquecendo, Qualificado, Enviado, etc.).
- Permite mover leads entre columnas y registrar “próxima acción” + notas.
- Mantiene el foco operativo: qué hacer ahora y con quién.

Admin — Growth Equestre (MVP)

Acompanhe leads, priorize atendimento e explore parceiros por UF/segmento — sem telas técnicas.

CRM (Kanban)

Board simples para organizar leads por estágio + notas + próxima ação (sem quebrar o funil atual).

Board

Dica: use o seletor + botão dentro do card para mover de coluna.

CURIOSO (3 leads)	AQUECENDO (7 leads)	QUALIFICADO (1 lead)	ENVIADO (3 leads)
Lead Demo — Curioso 33 CURIOSO - score=21 Goiânia/GO - 🐾 Eventos Próxima ação: não definida <input type="button" value="INBOX"/> <input type="button" value="Mover"/>	Visitante Demo 40 AQUECENDO - score=40 → MG - 🐾 Cavalos Próxima ação: não definida <input type="button" value="AQUECENDO"/> <input type="button" value="Mover"/>	Luiz Andre 42 QUALIFICADO - score=42 São Paulo/SP - 🐾 Eventos Próxima ação: não definida <input type="button" value="QUALIFICADO"/> <input type="button" value="Mover"/>	Lead Demo — Alta Intenção 90 ENVIADO - score=90 São Paulo/SP - 🐾 Equipamentos Próxima ação: não definida <input type="button" value="ENVIADO"/> <input type="button" value="Mover"/>
Lead Demo — Curioso 21 CURIOSO - score=21 Goiânia/GO - 🐾 Eventos Próxima ação: não definida <input type="button" value="INBOX"/> <input type="button" value="Mover"/>	Visitante Demo 52 AQUECENDO - score=52 → SP - 🐾 Serviços Próxima ação: não definida <input type="button" value="AQUECENDO"/> <input type="button" value="Mover"/>		Lead Demo — Alta Intenção 90 ENVIADO - score=90 São Paulo/SP - 🐾 Equipamentos Próxima ação: não definida <input type="button" value="ENVIADO"/> <input type="button" value="Mover"/>
Lead Demo — Curioso 31 CURIOSO - score=31 Goiânia/GO - 🐾 Eventos Próxima ação: não definida <input type="button" value="INBOX"/> <input type="button" value="Mover"/>	Lead Demo — Médio 42 AQUECENDO - score=42 Belo Horizonte/MG - 🐾 Cavalos Próxima ação: não definida <input type="button" value="AQUECENDO"/> <input type="button" value="Mover"/>		

Detalhes

Selecionar lead (CRM):

Figura 5 — CRM Kanban (board + detalles).

5.5 Matching de parceiros (lead ↔ parceiro)

Cuando seleccionas un lead en el CRM, la app sugiere parceiros del directorio con mayor compatibilidad.

- Compatibilidad por reglas: segmento + UF (y CNAE si existe y la opción está marcada).
- Control de “Cantidad de sugerencias” para ver más/menos opciones.
- Acción “Enviar para parceiro seleccionado” para simular el handoff/derivación.

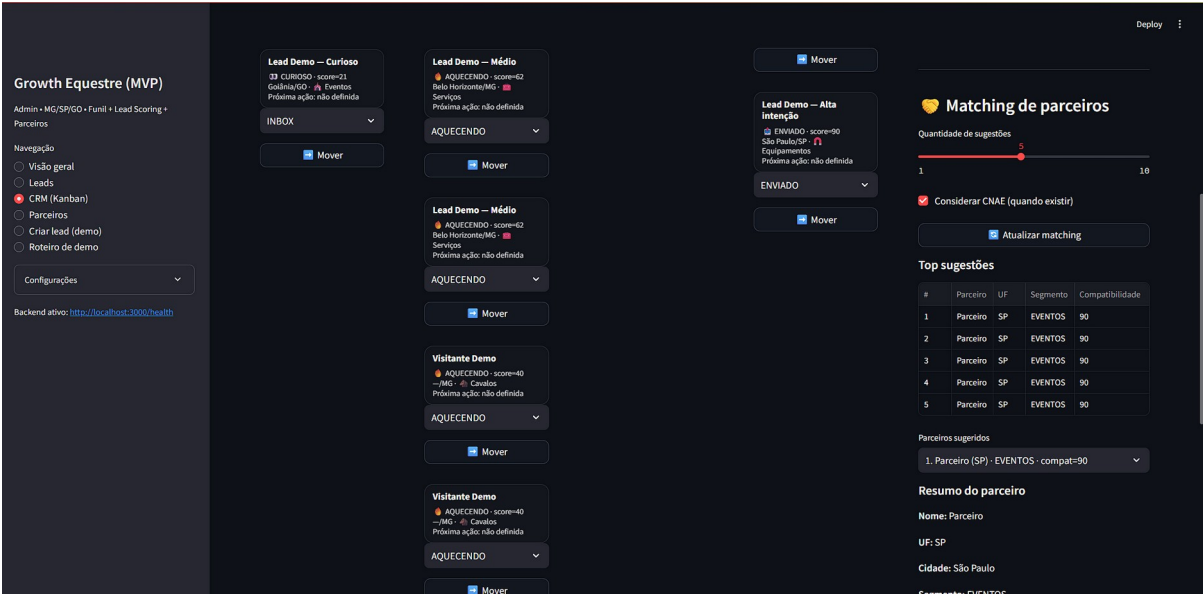


Figura 6 — Panel de Matching (sugerencias y compatibilidad).

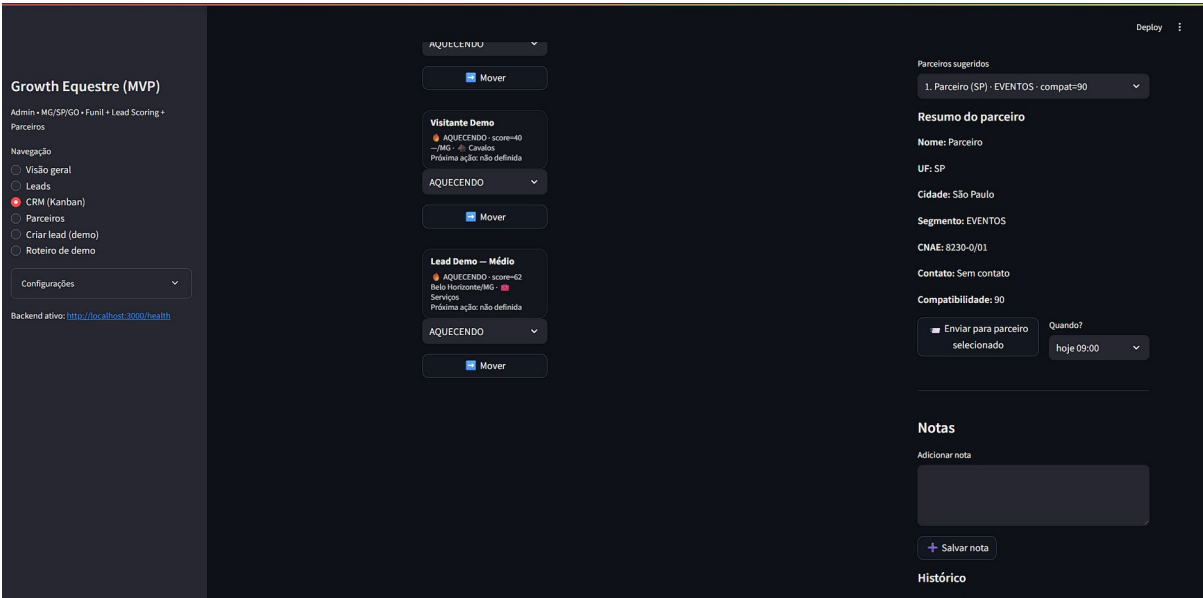


Figura 7 — Resumen del parceiro seleccionado + acción de envío.

5.6 Parceiros (directorio) — prospección

- Filtro por UF/segmento + búsqueda por nombre/razón social.
- Límite (cantidad) para acelerar la exploración.
- Exportación CSV para usar como lista de prospección.

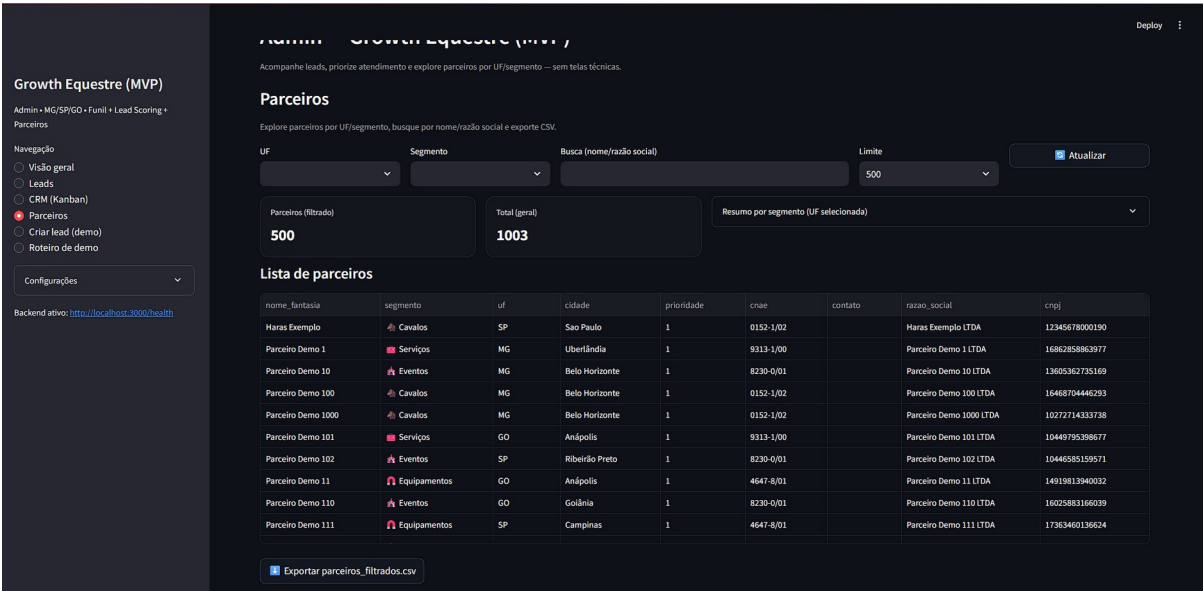


Figura 8 — Directório de parceiros (lista filtrable + export CSV).

5.7 Roteiro de demo (para el pitch)

- Guía el orden de presentación (visión general → leads → CRM → parceiros → export).
- Botones para crear un escenario completo y resetear la demo.
- Checklist rápido antes de presentar (backend up, UI ok, etc.).

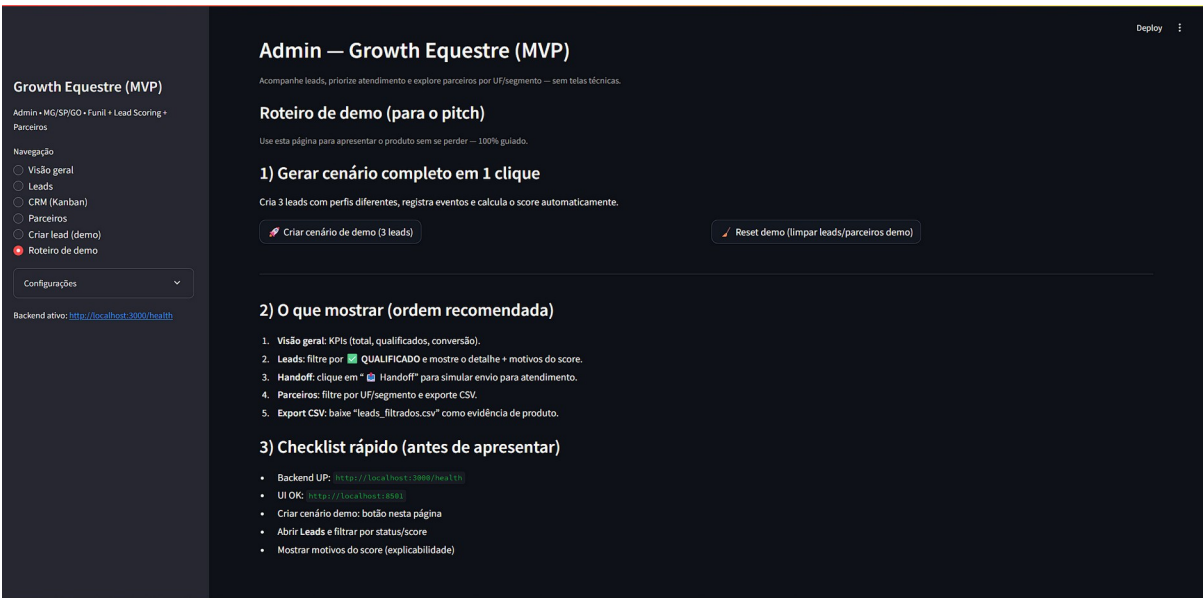


Figura 9 — Roteiro de demo (pitch guiado).

6. Datos y CNAE (cómo mejorar el directorio de parceiros)

Para enriquecer el directorio y mejorar el matching, la app puede usar CNAE cuando esté disponible. En el hackathon, la estrategia práctica es: definir palabras clave por segmento → consultar CNAE en una fuente confiable → generar un mapa (keyword → CNAE) para apoyar la búsqueda/segmentación.

6.1 Archivo de entrada: cnae_keywords.csv

- Formato esperado (CSV con separador ';'): segmento;keyword;prioridade;uf
- Ejemplos típicos: CAVALOS;haras;1; | EQUIPAMENTOS;selaria;1;

6.2 Generar cnae_map.csv (herramienta)

Si el repositorio incluye el script tools/cnae/generate_cnae_map.py, puedes ejecutarlo para generar el mapa.

```
python -m pip install requests beautifulsoup4 lxml
python -u .\tools\cnae\generate_cnae_map.py --in .\tools\cnae\cnae_keywords.csv --out .\data\cnae\cnae_map.csv

# Verificar salida
Get-Content .\data\cnae\cnae_map.csv -TotalCount 10
```

7. Solución de problemas (rápido)

- UI abre, pero no muestra datos: confirmar BACKEND_URL y /health; revisar logs del backend.
- 404 en una ruta: revisar si la ruta existe en server.js/crm_routes.js y si el servicio se reinició.
- Tabla de parceiros vacía con total > 0: suele ser parseo/filtrado; re-ejecutar /partners y revisar logs.
- Errores de Python (imports): activar .venv y reinstalar requirements.
- Problemas con Docker: docker compose ps / logs -f <service> / docker compose down -v (último recurso).

Fin del documento.