# PegaTrouxa - Escopo Técnico do Projeto SaaS

## Objetivo

Desenvolver uma aplicação SaaS hospedada em uma VPS que permite ao usuário criar arquivos .pdf falsos de comprovantes PIX. Ao serem abertos pela vítima, os PDFs redirecionam para uma página HTML maliciosa que simula validações bancárias e coleta dados sensíveis do navegador da vítima.

## Funcionalidades Principais

### 1. Autenticação

- Cadastro e login com autenticação por token (JWT)
- Painel individual por usuário SaaS

### 2. Geração de PDF

- Formulário para criação de iscas
- Escolha entre múltiplos templates HTML
- Geração de PDF com link de redirecionamento embutido
- PDF pronto para download com nome personalizado

### 3. Templates HTML

- Templates prontos para ataques simulados
- Sistema modular para adicionar, editar ou desativar templates
- Templates configuráveis com variáveis como valor do PIX, banco simulado, nome da vítima

### 4. Coleta de Dados

- Imagem da câmera frontal (getUserMedia)
- Geolocalização precisa (navigator.geolocation)
- Áudio ambiente (se autorizado)
- Dados do navegador (User-Agent, IP, data/hora)
- Dados enviados para API e vinculados ao usuário SaaS

5. Dashboard do Usuário SaaS

- Preview da imagem da webcam
- Player de áudio (se aplicável)
- Latitude e longitude com botão “Abrir Rota” (Google Maps)
- Data/hora de acesso
- Sistema de filtro por data, link, IP
- Notificação opcional (Telegram, Discord, etc)

Arquitetura Técnica

Frontend:

- HTML5 + TailwindCSS ou Bootstrap
- JavaScript puro para os templates maliciosos
- DataTables.js para o dashboard

Backend:

- Python 3 + Flask
- SQLite ou PostgreSQL
- Geração de PDF via WeasyPrint
- Autenticação com JWT
- Endpoints protegidos por token

Infraestrutura:

- VPS Linux (Ubuntu 22.04)
- Nginx com HTTPS (Let's Encrypt)
- Separação de dados por usuário
- Armazenamento local ou S3-like (MinIO)

Fluxo Resumido

1. Usuário SaaS cria conta e faz login
2. Escolhe template e gera PDF malicioso
3. PDF é entregue ao alvo
4. Alvo abre → redirecionado para HTML com JS malicioso
5. Navegador coleta dados (imagem, áudio, localização)
6. Dados armazenados e vinculados ao criador
7. Criador visualiza os dados no painel

Tarefas Técnicas Imediatas

- Estrutura modular Flask
- Endpoints: /api/register, /api/login, /api/pdf, /api/templates, /api/collect
- Tabelas: users, templates, pdf_links, capturas
- Telas: Login, Registro, Dashboard, Geração de PDF
- Página maliciosa com JS para coleta
- PDF com JS de redirecionamento
- API para coleta e armazenamento dos dados
- Tabela no dashboard com DataTables.js
- Player de mídia e botão Google Maps
- VPS com Nginx + HTTPS, Gunicorn, MinIO