



SUMÁRIO

SUMÁRIO.....	2
OBJETIVO.....	3
CARACTERÍSTICAS.....	3
CONTEXTO TÉCNICO.....	3
ESCOPO TÉCNICO.....	4
Back-End.....	4
Front-End.....	4
REQUISITOS.....	5
Expectativas.....	5
Diferenciais.....	5
FORMATOS.....	6
Formato de entrega.....	6
Observações.....	6

OBJETIVO

Desenvolver um **SDK (Software Development Kit)** de avaliação de uso do usuário para antifraudes que ajude aplicações de e-commerce a validar a identidade de usuários no momento do **login, checkout ou em ações sensíveis**, com o mínimo de fricção.

O SDK deve ser capaz de **coletar dados úteis** para avaliar o risco de fraude, **gerar um score de confiança** e permitir ao sistema hospedeiro **agir com base na avaliação do SDK**.

CARACTERÍSTICAS

O SDK deve ser:

- Modular e reutilizável, possível de integrar em qualquer aplicação web moderna;
- Plugável tanto no frontend quanto no backend;
- Com documentação de uso simples;
- Fácil de instalar.

CONTEXTO TÉCNICO

A **NexShop** é uma plataforma de e-commerce moderna que utiliza **Node.js** no backend (**com ExpressJS**) e **React + TailwindCSS** no frontend. Ela opera em um ambiente escalável e modular, com foco em performance, UX e personalização.

Com o crescimento acelerado de suas vendas e usuários, a NexShop passou a enfrentar tentativas de fraude digital em diversas etapas da jornada de compra, principalmente no login e no checkout. Por isso, a empresa busca uma solução leve, integrável e inteligente para aumentar a confiabilidade da identidade do usuário, reduzindo fraudes sem gerar atrito na experiência de compra.

ESCOPO TÉCNICO

Lado da Aplicação (Back-End)

Responsável por:

- Receber os dados enviados pelo front via endpoint HTTP
- Executar lógica de análise de risco:
 - Análise por regras (ex: novo dispositivo + geolocalização suspeita = alto risco)
 - Pontuação ou classificação do nível de risco
- Retornar uma resposta padronizada à aplicação cliente, indicando:
 - allow (prossiga com a ação)
 - review (ação requer validação adicional)
 - deny (ação deve ser bloqueada)

O backend deve ser desacoplado, permitindo integração com qualquer aplicação Node.js, Python, Ruby ou outras linguagens. **(Importante: O NexShop possui o backend em NodeJS, e é com ele que a solução será validada).**

Lado do Usuário (Front-End)

Responsável por:

- Coletar características do ambiente do usuário:
 - Device fingerprint (navegador, IP, timezone, resolução, idioma, etc.)
 - Comportamento do usuário (tempo na página, movimentos do mouse, interações, foco da aba, etc.)
 - Metadados da sessão
- Encaminhar de forma segura os dados ao backend via HTTP
- **Opcional:** módulo de captura facial com consentimento do usuário

Deve ser escrito de forma desacoplada de qualquer framework, podendo ser consumido por aplicações React, Vue, Angular ou Vanilla JS.

REQUISITOS

CATEGORIA	REQUISITOS
Front-End	<ul style="list-style-type: none">• Leve e não invasivo• Coleta passiva de informações (sem interação extra)• Compatível com browsers modernos• Comunicação segura com o backend
Back-End	<ul style="list-style-type: none">• Expor uma rota REST para ingestão de dados (/identity/verify)• Processamento assíncrono (se necessário)• Módulo configurável com regras de risco• Retorno estruturado com status e score
Documentação	<ul style="list-style-type: none">• Manual de integração claro e objetivo• Exemplos de uso no frontend e backend• Configurações básicas (ex: nível de sensibilidade, whitelists, callbacks)
Instalação	<ul style="list-style-type: none">• Integração em uma linha no front (<code>initSDK()</code>)• Middleware simples no backend (<code>app.use(sdk.middleware())</code>)

Expectativas

- SDK funcional com capacidade de:
 - Capturar dados relevantes no cliente
 - Enviar e processar os dados no backend
 - Retornar recomendação de ação para login ou checkout
- Exemplo de integração com um ambiente simulado da NexShop (login ou pagamento)
- Documentação básica de uso e integração

Diferenciais

- Integração com API de verificação facial ou biométrica
- Aprendizado automático com base no comportamento do usuário
- Painel de monitoramento (dashboard de eventos)
- Geração de alertas e logs para sistemas SIEM

FORMATOS

Formato de entrega

- Código-fonte do SDK (frontend e backend) em repositório no GitHub;
- Instruções de instalação e uso detalhadas (documentação detalhada);
- Apresentação final para banca com demonstração da solução;

Observações

- Os times devem priorizar desacoplamento, reusabilidade e segurança por padrão.
- Toda coleta de dados sensíveis deve respeitar boas práticas de privacidade (ex: consentimento no uso de câmera).
- O uso de bibliotecas externas é permitido desde que bem justificado e referenciado.