**PRD - Módulo Controle de Acesso com Reconhecimento Facial**

**Sistema Integrado FELKA Transportes**

**1. VISÃO GERAL DO PRODUTO**

**1.1 Objetivo**

Desenvolver um módulo de controle de acesso completo que utilize reconhecimento facial como método principal de autenticação para funcionários e visitantes, com fallback para entrada manual em casos de falha. O sistema deve garantir segurança, rastreabilidade e fluidez operacional na portaria.

**1.2 Stakeholders**

* **Usuários Primários**: Porteiros (tablet/mobile)
* **Usuários Secundários**: Administradores RH (web), Gestores de Segurança (web)
* **Sistemas Integrados**: Módulo RH, Módulo Veículos, Sistema de Checklists

**1.3 Métricas de Sucesso**

* Taxa de reconhecimento facial > 95%
* Tempo médio de entrada < 10 segundos
* 100% dos acessos rastreados
* Redução de 80% no tempo de cadastro de visitantes recorrentes

**2. FUNCIONALIDADES PRINCIPAIS**

**2.1 Cadastro de Reconhecimento Facial**

**2.1.1 Cadastro de Funcionário**

**Interface**: PC/Desktop na portaria **Fluxo**:

1. Inserir matrícula ou CPF
2. Busca automática na base de colaboradores ativos (integração RH)
3. Exibição de dados: nome, cargo, setor, foto cadastral
4. Captura facial via câmera (múltiplos ângulos)
5. Processamento e encoding facial
6. Confirmação de cadastro

**Validações**:

* Funcionário deve estar ativo no sistema RH
* Qualidade mínima da captura facial
* Não permitir duplicatas

**2.1.2 Cadastro de Visitante**

**Interface**: PC/Desktop na portaria **Fluxo**:

1. Inserir dados básicos: nome completo, CPF
2. Captura facial ao vivo
3. Processamento e encoding facial
4. Armazenamento para futuras visitas

**Observação**: Dados específicos da visita (empresa, responsável, tipo) serão inseridos a cada acesso.

**2.2 Controle de Entrada/Saída (Interface Mobile/Tablet)**

**2.2.1 Tela Principal - Layout Dividido**

**Estrutura**: Três seções principais

**Seção Superior**: Reconhecimento Facial

* Câmera ativa para captura
* Botões: "Entrada Manual Funcionário" | "Entrada Manual Visitante"
* Indicador visual de status (vermelho/verde)
* Display de resultado (nome + cargo/status)

**Seção Inferior Esquerda**: Saída de Veículos

* Lista de veículos com checklist aprovado
* Dados: placa, motorista, hora do checklist
* Ação: "Dar Saída" → registra porteiro, base, timestamp

**Seção Inferior Direita**: Retorno de Veículos

* Lista de veículos em trânsito
* Ação: "Registrar Retorno" → seleção de base + timestamp

**2.2.2 Reconhecimento Facial - Fluxo Principal**

1. Captura automática quando pessoa se aproxima
2. Processamento em tempo real (< 3 segundos)
3. Resultados possíveis:
   * **Sucesso**: Tela verde + nome + registro automático
   * **Falha**: Tela vermelha + opções manuais
   * **Não reconhecido**: Tela amarela + sugestão de cadastro

**2.2.3 Entrada Manual de Funcionário (Fallback)**

**Trigger**: Falha no reconhecimento facial **Fluxo**:

1. Botão "Entrada Manual de Funcionário"
2. Campo de busca: CPF ou nome (autocomplete)
3. Seleção do funcionário
4. Verificação de status ativo
5. Confirmação de entrada
6. Registro com tag "Entrada Manual"

**2.2.4 Entrada Manual de Visitante (Fallback)**

**Trigger**: Falha no reconhecimento facial ou visitante novo **Fluxo**:

1. Busca por nome/CPF (se já cadastrado)
2. Se encontrado: carrega dados básicos
3. Se não encontrado: formulário de cadastro rápido
4. Campos adicionais obrigatórios:
   * Empresa visitante
   * Tipo de visita
   * Responsável autorizador
   * Placa do veículo (opcional)
5. Geração de crachá temporário
6. Registro com tag específica

**2.3 Geração de Crachás**

**2.3.1 Crachá de Funcionário**

* Nome completo
* Cargo
* Setor
* Foto (se disponível)
* Código identificador
* Data/hora de entrada

**2.3.2 Crachá de Visitante**

* Nome completo
* Empresa visitante
* Tipo de visita
* Responsável autorizador
* Data/hora de entrada
* Identificação: "VISITANTE"

**2.4 Controle de Saída**

**Funcionários**: Reconhecimento facial na saída **Visitantes**: Busca manual por nome + confirmação de saída

**3. REQUISITOS TÉCNICOS**

**3.1 Reconhecimento Facial**

**Algoritmo Recomendado**: Face-api.js ou MediaPipe Face Detection **Justificativa**:

* Leve e eficiente para dispositivos móveis
* Funcionamento offline após carregamento inicial
* Boa precisão para aplicações comerciais
* Suporte nativo em navegadores modernos

**Especificações**:

* Encoding facial em 128 ou 512 dimensões
* Threshold de similaridade: 0.6 (ajustável)
* Captura mínima: 640x480
* Formato de armazenamento: JSON com descriptors

**3.2 Hardware Recomendado**

**PC Portaria (Cadastro)**:

* Câmera HD (1080p) com autofoco
* Processador: i5 ou equivalente
* RAM: 8GB mínimo
* Conexão estável com o servidor

**Tablet/Mobile (Controle)**:

* Câmera frontal HD
* Android 8+ ou iOS 12+
* RAM: 4GB mínimo
* Tela: 10" mínimo para boa usabilidade

**3.3 Integrações**

* **API RH**: Busca de funcionários ativos
* **Módulo Veículos**: Dados de checklist e placas
* **Sistema de Logs**: Rastreabilidade completa
* **Gerador de PDF**: Crachás e relatórios

**4. EXPERIÊNCIA DO USUÁRIO (UX)**

**4.1 Persona Principal: João (Porteiro)**

**Contexto**: 45 anos, ensino médio, familiarizado com tablets **Necessidades**: Interface simples, resposta rápida, mínimo de cliques **Frustações**: Sistemas complexos, muitas telas, falhas técnicas

**4.2 Jornada do Usuário**

**4.2.1 Entrada de Funcionário (Cenário Ideal)**

1. Funcionário se aproxima da câmera
2. Sistema reconhece automaticamente (2-3s)
3. Tela verde + "Bem-vindo, [Nome]"
4. Registro automático no sistema
5. Liberação da catraca/entrada

**Tempo total**: < 10 segundos

**4.2.2 Entrada de Funcionário (Cenário com Falha)**

1. Funcionário se aproxima da câmera
2. Sistema não reconhece (após 5s)
3. Tela vermelha + "Não reconhecido"
4. Porteiro clica "Entrada Manual"
5. Busca por nome/CPF
6. Confirmação e liberação

**Tempo total**: < 30 segundos

**4.2.3 Entrada de Visitante Novo**

1. Visitante se aproxima
2. Sistema não reconhece
3. Porteiro seleciona "Entrada Manual Visitante"
4. Busca rápida (não encontrado)
5. Formulário de dados básicos
6. Geração e impressão de crachá
7. Liberação

**Tempo total**: < 60 segundos

**4.3 Princípios de Design**

* **Simplicidade**: Máximo 3 cliques para qualquer ação
* **Feedback Visual**: Cores claras para status (verde/vermelho/amarelo)
* **Responsividade**: Interface adaptável a tablets 10-12"
* **Acessibilidade**: Botões grandes, contraste adequado
* **Offline-First**: Funcionamento sem conexão por períodos curtos

**5. ARQUITETURA DO SISTEMA**

**5.1 Componentes Principais**

Frontend (React/Next.js)

├── Tela Cadastro (Desktop)

├── Tela Controle (Mobile/Tablet)

├── Interface Reconhecimento Facial

└── Gerador de Crachás

Backend (Node.js/Express)

├── API Reconhecimento Facial

├── Serviços de Integração RH

├── Sistema de Logs

└── Gerador de Relatórios

Banco de Dados (PostgreSQL)

├── Tabela facial\_encodings

├── Tabela access\_logs

├── Tabela visitor\_data

└── Integração com tabelas RH

**5.2 Fluxo de Dados**

1. **Cadastro**: Captura → Encoding → Armazenamento
2. **Reconhecimento**: Captura → Comparação → Resultado
3. **Fallback**: Busca Manual → Validação → Registro
4. **Auditoria**: Logs → Histórico → Relatórios

**6. CRONOGRAMA DE DESENVOLVIMENTO**

**Sprint 1 (Semana 1-2): Fundação**

* Setup do projeto e estrutura básica
* Implementação do schema Prisma
* Interface básica de cadastro (desktop)

**Sprint 2 (Semana 3-4): Reconhecimento Facial**

* Integração com Face-api.js
* Sistema de encoding e comparação
* Testes de precisão e performance

**Sprint 3 (Semana 5-6): Interface Mobile**

* Desenvolvimento da tela de controle (tablet)
* Integração com módulos existentes
* Sistema de fallback manual

**Sprint 4 (Semana 7-8): Recursos Avançados**

* Geração de crachás
* Sistema de logs e auditoria
* Históricos web administrativos

**Sprint 5 (Semana 9-10): Testes e Refinamento**

* Testes de integração
* Ajustes de UX
* Documentação e treinamento

**7. CRITÉRIOS DE ACEITAÇÃO**

**7.1 Funcionais**

* [ ] Cadastro facial com taxa de sucesso > 95%
* [ ] Reconhecimento em tempo real < 3 segundos
* [ ] Fallback manual funcional em < 30 segundos
* [ ] Integração completa com módulo RH
* [ ] Geração automática de crachás
* [ ] Controle de veículos funcional
* [ ] Históricos web acessíveis e exportáveis

**7.2 Não-Funcionais**

* [ ] Interface responsiva em tablets 10-12"
* [ ] Funcionamento offline por até 30 minutos
* [ ] Logs de auditoria completos
* [ ] Backup automático dos encodings faciais
* [ ] Tempo de resposta da interface < 2 segundos

**7.3 Segurança**

* [ ] Encodings faciais criptografados
* [ ] Logs com rastreabilidade completa
* [ ] Acesso baseado em perfis de usuário
* [ ] Proteção contra ataques de replay

**8. RISCOS E MITIGAÇÕES**

**8.1 Riscos Técnicos**

**Risco**: Baixa taxa de reconhecimento facial **Mitigação**: Sistema robusto de fallback + múltiplos ângulos de captura

**Risco**: Performance lenta em dispositivos móveis **Mitigação**: Otimização de algoritmos + processamento assíncrono

**8.2 Riscos Operacionais**

**Risco**: Resistência dos usuários ao novo sistema **Mitigação**: Treinamento adequado + interface intuitiva

**Risco**: Falhas de hardware (câmera) **Mitigação**: Modo manual sempre disponível

**8.3 Riscos de Segurança**

**Risco**: Vazamento de dados biométricos **Mitigação**: Criptografia + armazenamento local de encodings

**9. CONSIDERAÇÕES LEGAIS E PRIVACIDADE**

**9.1 LGPD (Lei Geral de Proteção de Dados)**

* Consentimento explícito para captura facial
* Direito de exclusão dos dados biométricos
* Armazenamento apenas local (não cloud)
* Política de retenção de dados clara

**9.2 Termo de Consentimento**

* Para funcionários: termo no cadastro
* Para visitantes: aviso na entrada
* Opção de recusa com entrada manual alternativa

**10. ROADMAP FUTURO**

**Fase 2 (3-6 meses)**

* Integração com catracas eletrônicas
* App mobile para pré-cadastro de visitantes
* Dashboard analytics avançado

**Fase 3 (6-12 meses)**

* Reconhecimento de placas (OCR)
* Integração com sistemas de segurança externos
* IA para detecção de comportamentos suspeitos

**11. RECURSOS NECESSÁRIOS**

**11.1 Equipe**

* 1 Desenvolvedor Full-Stack (React/Node.js)
* 1 Especialista em Computer Vision
* 1 Designer UX/UI
* 1 Analista de Testes

**11.2 Hardware**

* 2 câmeras HD para portaria
* 2 tablets 10" para porteiros
* 1 PC para cadastros
* Impressora térmica para crachás

**11.3 Software/Licenças**

* Face-api.js (open source)
* Bibliotecas de processamento de imagem
* Certificados SSL para segurança

**CONCLUSÃO**

Este PRD estabelece as bases para um sistema de controle de acesso moderno e eficiente, focado na experiência do usuário e na segurança operacional. A implementação por sprints permite validação contínua e ajustes conforme necessário.

A escolha por Face-api.js como algoritmo principal oferece o melhor custo-benefício entre performance, precisão e facilidade de implementação, sendo ideal para o ambiente proposto.