Documentação Final do Sistema de Controle de Acesso de Estacionamento

# 📘 Introdução

Este documento apresenta a documentação técnica e visual do sistema de controle de acesso de estacionamento desenvolvido por João Vitor. O sistema integra backend com Node.js e Sequelize, frontend com React, e banco de dados PostgreSQL hospedado no Supabase.

# 📋 Visão Geral do Projeto

O objetivo do projeto é permitir que usuários (alunos e colaboradores) cadastrem seus veículos e acessos ao estacionamento sejam monitorados por um operador (guarita). O sistema permite o cadastro de visitantes, registro de entrada/saída por placa e acompanhamento de logs administrativos.

# 🛠️ Tecnologias Utilizadas

- Node.js + Express  
- Sequelize ORM  
- PostgreSQL (Supabase)  
- React + Vite  
- Axios  
- JWT (JSON Web Token)  
- CSS puro

# 🧠 Funcionalidades Backend

- Cadastro e login de usuários com autenticação JWT  
- Controle de acesso por tipo (guarita, aluno, colaborador)  
- CRUD de veículos  
- Cadastro e listagem de visitantes  
- Registro de entrada e saída de veículos por placa  
- Logs administrativos com nome do operador e ação

# 🖥️ Funcionalidades Frontend

- Login com salvamento de token e tipo de usuário  
- Dashboard com navegação dinâmica por tipo  
- Telas de veículos, visitantes, acessos e logs  
- Menu lateral fixo com navegação e estilo profissional  
- Layout responsivo e limpo utilizando CSS modular

# 🔐 Segurança

- JWT armazenado com expiração  
- Middleware de autenticação nas rotas  
- Proteção de rotas por tipo de usuário  
- Variáveis sensíveis protegidas via .env

# 📁 Organização do Projeto

backend/  
├── controllers/  
├── models/  
├── routes/  
├── config/  
├── server.js  
└── .env  
  
frontend/  
├── components/  
├── pages/  
├── styles/  
├── App.jsx  
└── main.jsx

# 🚀 Build e Deploy

O frontend é empacotado com `npm run build` e pode ser hospedado em serviços como Vercel ou servido pelo próprio backend com Express. O backend pode ser hospedado no Render, Railway ou outros serviços de Node.js.

# ✅ Conclusão

O sistema foi finalizado com todas as funcionalidades operacionais, código limpo, layout agradável e documentação adequada. Está pronto para testes em produção ou deploy imediato.