

# **Manual Técnico y de Arquitectura Definitivo**

**PWA VecinosExpress**

Versión 1.0 – Agosto 2025

# 1. Visión General del Sistema Integrado

El proyecto consiste en una robusta Aplicación Web Progresiva (PWA) diseñada para ofrecer los servicios de "VecinosExpress" y un módulo completo de Notarización Remota en Línea (RON). La arquitectura está concebida para ser segura, modular y altamente escalable.

Arquitectura General:

- Frontend: React + TypeScript con TailwindCSS y componentes de Shadcn UI. - Backend: Node.js con Express y estructura modular. - Base de Datos: PostgreSQL gestionada con Drizzle ORM. - Integraciones: • Pasarelas de pago: MercadoPago, Stripe, PayPal. • Videollamada: Agora para RON. • Verificación: GetAPI.cl (biometría, NFC). • Comunicación: SendGrid (correos y notificaciones).

## 2. Flujo Detallado de la PWA

Fase 1: Inicio y Sesión del Operador

- Splash Screen inicial. - Login con credenciales. - Validación con bcrypt + JWT (20h).

Fase 2: Selección de Servicio

- Pantalla principal con tarjetas de servicios. - Trámite con Plantilla Estándar: • Selección de plantilla. • Relleno y vista previa. • Verificación NFC y biométrica. • Pago por POS. • Firma con eToken. • Envío automático del PDF final.

Flujo B: Notarización Remota (RON)

- Selección de servicio RON y pago inicial. - Generación de ID y acceso a sala con Agora. - Cliente y certificador acceden a videollamada grabada. - Firma y sellado automático. - Envío del documento certificado por correo.

### 3. Estructura Completa de Carpetas y Archivos

/client/ ■■■■ public/ ■■■■ src/ ■ ■■■■ assets/ ■ ■■■■ components/ ■ ■■■■ hooks/ ■ ■■■■ lib/ ■ ■■■■ pages/ ■ ■ ■■■■ Login.tsx, Dashboard.tsx, DocumentWorkflow.tsx, IdentityVerification.tsx ■ ■ ■■■■ ron/ → RonPayment.tsx, RonWaitingRoom.tsx, RonSession.tsx ■ ■■■■ App.tsx, main.tsx ■■■■ package.json /server/ ■■■■ migrations/ ■■■■ src/ ■ ■■■■ controllers/, middleware/, services/ ■ ■■■■ notarypro/ (módulo exclusivo RON) ■ ■■■■ routes/ → auth.routes.ts, ron.routes.ts, payment.routes.ts, index.ts ■■■■ index.ts /shared/ ■■■■ schema.ts .env.example, tsconfig.json, package.json