

Análisis Técnico del Repositorio: Carpeta Ciudadana Backend

Información General

Nombre del Proyecto: Carpeta Ciudadana Backend

Repositorio: github.com/ogticrd/carpeta-ciudadana-backend

El backend centraliza la lógica de notificaciones y cuenta con un componente llamado "wrapper" que agrupa a las instituciones que funcionan de forma similar, como los ayuntamientos, las empresas distribuidoras de electricidad (EDES), y las compañías de agua. Esto permite manejar todas estas entidades de manera unificada, evitando realizar múltiples peticiones separadas para cada una y optimizando así la comunicación y el rendimiento del sistema.

Elemento	Descripción
Arquitectura	Microservicios
Framework principal	NestJS
Lenguaje	TypeScript
Comunicación entre servicios	gRPC
Contenedores	Docker, <code>docker-compose</code> , múltiples Dockerfiles
Gestión de paquetes	npm/yarn
Infraestructura como código	<code>docker-compose.yml</code> , múltiples archivos de entorno (<code>.env.example</code>)

Análisis de Microservicios

Microservicio	Funcionalidad
gateway	Punto de entrada, autenticación, health check, orquestación general

Microservicio	Funcionalidad
metadata	Gestión de usuarios, sesiones, OTPs, suscripciones de notificación
notification	Envío y manejo de notificaciones, workflows
wrapper-institutions	Integración con instituciones (ej. JCE, Edesur, pasaportes)

DevOps / Automatización

Elemento	Descripción
CI/CD	Workflows por microservicio (<code>gateway-deployment.yml</code> , etc.)
Limpieza de caché	<code>cleanup-cache-branch.yml</code> para mantener el entorno limpio
Dependabot	Actualización automática de dependencias (<code>dependabot.yml</code>)
Entornos definidos	<code>.env.example</code> , <code>development.Dockerfile</code> , múltiples <code>Dockerfile</code>
Linting y Formateo	<code>.eslintrc.js</code> , <code>.prettierrc</code> , uso de <u>Biome</u>
Debugging	Archivos <code>.vscode/launch.json</code> en cada microservicio

Análisis de Workflows CI/CD (`.github/workflows/`)

Archivo	Descripción
<code>cleanup-cache-branch.yml</code>	Limpia las cachés asociadas a una Pull Request (PR) cuando esta es cerrada.
<code>gateway-deployment.yml</code>	Este archivo parece estar relacionado con el despliegue del Gateway .
<code>metadata-deployment.yml</code>	Este archivo probablemente gestiona el despliegue del microservicio Metadata . La última actualización sugiere que se corrigió alguna cuestión relacionada con el entorno de producción.
<code>notification-deployment.yml</code>	Relacionado con el despliegue del microservicio Notification . La última actualización menciona la integración o actualización con Novu (probablemente una plataforma de notificaciones).
<code>wrapper-institutions-deployment.yml</code>	Este archivo gestiona el despliegue de Wrapper Institutions , que probablemente interactúa con servicios externos o APIs de instituciones gubernamentales.

Control de Calidad

Elemento	Detalles
Framework de testing	jest
Tipo de pruebas	e2e (end-to-end), ejemplos: app.e2e-spec.ts
Cobertura	Se encuentran *.spec.ts para pruebas unitarias (ej: *.service.spec.ts)

Seguridad

Elemento	Descripción
API Key Middleware	auth/api-key.middleware.ts (wrapper-institutions)
Manejo de tokens OTP	Varios microservicios trabajan con OTPs y suscripciones seguras
Entornos seguros	.env.example usado en todos los servicios
Dependabot	Protege contra vulnerabilidades de paquetes
Sin SonarQube	No se encontró integración con SonarQube

Herramientas y Librerías Detectadas

Tipo	Herramientas / Librerías
Backend Framework	NestJS
Contenedores	Docker, Docker Compose
Proto/gRPC	Archivos .proto para cada servicio
Lint/Formato	ESLint, Prettier, Biome
Tests	Jest
Gestión de paquetes	Yarn o npm
VSCoDe Dev Env	.vscode en todos los servicios

Documentación

Sección	Contenido Detectado
Introducción	Describe brevemente el propósito del backend: "simplificar la gestión de documentos y trámites ciudadanos".

Sección	Contenido Detectado
Arquitectura	Enumera los microservicios (gateway, metadata, notification, wrapper-institutions) y explica sus responsabilidades.
Tecnologías	Menciona NestJS, TypeScript y gRPC como stack principal.

Revisión #13

Creado 4 junio 2025 18:23:02

Actualizado 4 junio 2025 20:10:29