

Backlog Técnico y Roadmap

Table of contents

1 Índice	1
2 Tablero de tareas	1
3 Bugs conocidos	5
4 Roadmap	6
4.1 Horizonte inmediato (0–2 semanas)	6
4.2 Próximo ciclo (2–6 semanas)	6
4.3 Mediano plazo (>6 semanas)	6

1 Índice

- [Tablero de tareas](#)
- [Bugs conocidos](#)
- [Roadmap](#)

2 Tablero de tareas

ID	Categoría	Descripción	Estado	Responsable	Metadatos
API-Refactor-201	API	Dividir routers presupuestos/policiales en paquetes con modelos Pydantic y utilidades compartidas.	Completado	Backend	Nuevos módulos <code>api/routes/budgets/*</code> y <code>api/routes/policiales/*</code> , pruebas manuales de endpoints críticos.
OPS-Tooling-202	Tooling	Enforzar límite de 400 líneas mediante <code>scripts/report_large_files.py</code> en pre-commit/CI.	Completado	Backend	Hook local agregado en <code>.pre-commit-config.yaml</code> , falla si un archivo supera el umbral configurable.

ID	Categoría	Descripción	Estado	Responsable	Metadatos
203	ETL	ETL-Refactor Modularizar <code>etl/pipelines/dipres_budget</code> (constantes, modelos, IO) y separar pipeline bronze en paquete dedicado.	Completado	Dattok	BudgetHierarchyBuilder.process delega en builder_processing.process_dataframe; queda pipeline DipresBudgetBronze con helpers desacoplados.
304	ETL	ETL-Refactor Migrar <code>ine_estadisticas_judiciales</code> a paquete modular (<code>settings</code> , <code>workbooks</code> , <code>tidy</code> , <code>pipeline</code>).	Completado	Dattok	Nuevo paquete <code>etl/pipelines/ine_estadisticas_judiciales/</code> con docstrings, helpers reutilizables y QA centralizado.
305	ETL	ETL-Refactor Particionar <code>etl/pipelines/enusc.py</code> en paquete (<code>io</code> , <code>modules</code> , <code>reports</code> , <code>pipeline</code>).	Completado	Dattok	Nuevo paquete <code>etl/pipelines/enusc/</code> con componentes <code>settings</code> , <code>catalog</code> , <code>operations</code> , <code>pipeline</code> y escritor DB.
306	ETL	ETL-Refactor Dividir <code>etl/pipelines/ine_policiales.py</code> y CLI asociada en servicios reutilizables.	Completado	Dattok	Paquete <code>etl/pipelines/ine_policiales/</code> con módulos <code>settings</code> , <code>sheets</code> , <code>quality</code> ; queda pendiente modularizar la CLI.
307	ETL	ETL-Refactor Modularizar <code>etl/pipelines/seguridad_fuentes.py</code> por fuente (Carabineros, PDI, Fiscalía).	Completado	Dattok	Nuevo paquete <code>seguridad_fuentes/</code> separa Fiscalía, Gendarmería, DIPRES e INE; pipeline actualizado con QA y escrituras.
308	ETL	ETL-Refactor Extraer agregaciones de <code>etl/pipelines/enusc_agg.py</code> a submódulos parametrizables.	Completado	Dattok	Nuevo paquete <code>etl/pipelines/enusc_agg/</code> con <code>settings</code> , <code>specs</code> , <code>transforms</code> , <code>preparation</code> , <code>aggregations</code> , <code>builder</code> , <code>pipeline</code> ; wrapper <code>etl/pipelines/enusc_agg.py</code> mantiene compatibilidad y se añadieron pruebas unitarias para <code>transforms/agregaciones</code> clave.
309	ETL	ETL-Refactor Reorganizar pipeline de Fiscalía en etapas <code>extract</code> , <code>transform</code> , <code>load</code> .	Completado	Dattok	Paquete <code>fiscalia_persecucion_penal/</code> con manifiestos, periodos, QA y pipeline modular; pendiente agregar pruebas unitarias específicas.
310	OPS-Tooling	Reescribir <code>scripts/ingest_ine_policiales.py</code> como paquete CLI con <code>commands</code> , <code>services</code> , <code>validators</code> .	Completado	Dattok	CLI con subcomandos (<code>ingest</code> , <code>metadata</code>), filtros región/indicador y logs JSON; módulos <code>ingest_*</code> reutilizables.
311	OPS-Tooling	Modularizar <code>scripts/generate_report_previews.py</code> separando renderers y pipelines de datos.	Completado	Dattok	Se creó <code>scripts/report_previews/</code> con módulos por dominio y utilidades reutilizables; script principal orquesta salida.

ID	Categoría	Descripción	Estado	Responsable	Metadatos
OPS-Tooling 312	Factorizar	<code>scripts/build_cofog_series.py</code> en loaders/adaptadores reutilizables.	Completado	Datos	Se creó paquete <code>scripts/cofog_series/</code> con loaders, mapping y pipeline; script principal invoca <code>build_cofog_series()</code> .
ETL-Refactor 310	ETL	Completar pruebas e integración del pipeline ENUSC modularizado.	En progreso	Datos	Faltan pruebas integrales con SAV anonimizados, documentación de nuevas clases (<code>PersonHouseholdExporter</code> , <code>ModuleExporter</code>) y automatizar QA (ver <code>docs/enusc_todo.md</code>).
OPS-Tooling 320	Unificar CLI/QA para pipelines (preview, ingest, validate, compare).		Pendiente	Datos	Diseñar <code>etl.cli</code> multi-catálogo, move <code>SchemaValidator</code> a módulo común, exponer validación/manifest en CLI & CI.
QA- Observación 401	Automatizar dashboards/QA y comparación de manifests multi-catálogo.		Pendiente	Datos	Configurar job CI/cron que ejecute <code>compare-manifests</code> y genere notebooks/dashboards con datos en <code>data/silver/**/reportes/</code> .
DB- Migración 313	Extraer utilidades comunes de Alembic (<code>202510090002_add_presupuestos</code> , <code>202510090005_create_rrhh_tables</code>).		Pendiente	Datos	Crear <code>alembic_utils/schema_helpers.py</code> , normalizar downgrades y documentar cambios.
FRONT-Entendimiento 204	Centralizar rutas de assets (<code>/frontend/assets/data/*.json</code>) en <code>frontend/src/utils/constants.ts</code> .		Completado	Frontend	Nuevas constantes <code>CATALOG_SUMMARY_PATH</code> , <code>GRAPHS_CATALOG_PATH</code> , <code>COFOG_FUNCTIONS_PATH</code> consumidas por servicios.
UI- Catálogo 201	Implementar pruebas UI Playwright para la nueva vista de gráficos y la ficha de dataset.		Completado	Frontend	Nuevas suites Playwright (<code>tests/ui/test_graph_catalog_playwright.py</code> , <code>tests/ui/test_dataset_playwright.py</code>).
DATAMetadatos 202	Automatizar la generación incremental de <code>graphs_catalog.json</code> desde los registros del registry y <code>jobs_runs</code> .		Completado	Datos	Nueva utilidad <code>update_graph_catalog</code> (<code>analysis/charts/metadata.py</code>) + pruebas (<code>tests/unit/test_graph_catalog_update.py</code>).
DOCDocumentación 203	Actualizar <code>docs/graphs_catalog.md</code> con ejemplos embebidos y guías para uso del nuevo catálogo en dataset específicos.		Completado	Datos	Se añadió sección “Cómo consumir” (TS/Python, links profundos) y se describieron assets auxiliares.
ETLIngesta 001	Descargar Ejecución Total DIPRES 2022/2024 mensual (marzo, junio, septiembre) y Q4; 2025 mensual (marzo-diciembre) + T3/T4.		Completado	Datos	Fuente: <code>dipres_coverage.pending_downloads</code>

ID	Categoría	Descripción	Estado	Responsable	Metadatos
002	ETLIngesta	Obtener Proyecto de Ley de Presupuestos 2026 (CSV/XML + anexos XLSX).	Completado	Datos	Bloquea comparaciones interanuales 2025→2026
003	ETLIngesta	Incorporar dataset <code>ley_presupuesto</code> (ley vigente) con metadatos completos.	Completado	Datos	Dataset YAML + API de descargas listo; resta completar cobertura histórica
004	ETLScraper	Automatizar endpoints AJAX para Ejecución Total <= 2023 y secciones Partida/Capítulo/Programa.	Completado	Datos	Identificar API interna <code>w3-multipropertyvalues-*</code>
005	DWHModelado	Validar que subtítulos 01–35 alimenten correctamente la jerarquía <code>naturaleza</code> (ingreso/gasto).	Completado	Datos	Tests: <code>tests/data/test_dipres_budget_pipeline.py</code>
006	API-API	Ampliar pruebas de <code>presupuestos/stats</code> para cubrir <code>naturaleza</code> y <code>unidad</code> .	Completado	Datos	Añadir fixtures multimonedas
007	UI-UI	Tests de integración (Playwright) para nuevas tarjetas de variaciones/ <code>top_changes</code> .	Completado	Frontend	Escenario happy-path y filtros
008	OPSObservación	Exponer métricas de <code>status_codes</code> y <code>dipres_coverage</code> a Prometheus.	Completado	Ops	Depende de decisión de plataforma
009	DOCDocumentación	Añadir <code>docs/dipres_catalog.md</code> en sincronía con <code>dipres_coverage</code> (script + status page).	Completado	Datos	Revisar tras cada corrida de ingestión
010	ETLIngesta	Recuperar Proyecto/Ley Nacionales 2012 hacia atrás (búsqueda en archivo 594/598, PDFs).	Completado	Datos	Portal actual responde 404 (<code>w3-multipropertyvalues-15168-{pvid}</code>)
011	ETLIngesta	Resolver endpoints legacy/AJAX para Ejecución Total 2010–2021 (y 2023).	Completado	Datos	Identificar API JSON usada por DIPRES para tabs dinámicas
103	DATATasas	Incorporar tipo de cambio de promedio anual previo a 2019 para convertir ejecuciones históricas en dólares.	Completado	Datos	Fuente sugerida: Banco Central (serie Observatorio F073.TCO.PRE.Z.D)
012	ETLIngesta	Descargar y normalizar Ejecución Total 2023 (mensual, pesos/dólares).	Completado	Datos	Script <code>download_dipres_history.py</code> + <code>DipresBudgetBronze</code> .
013	ETLIngesta	Documentar que 2003–2014 sólo cuentan con informes trimestrales (mar/jun/sep/dic) y ajustar cobertura automática.	Completado	Datos	<code>coverage_summary</code> marca <code>coverage=quarterly</code> ; 2003 Q1 sin fuente pública.

ID	Categoría	Descripción	Estado	Responsable	Metadatos
ETL-014	Ingesta	Regenerar jerarquía bronze (totales, nodos) e índice (<code>hierarchy_index.json</code> , COFOG) tras actualización 2023.	Completado	Dipres	BudgetBronze, <code>build_hierarchy_index.py</code> , <code>build_cofog_series.py</code> .
OBS-009	Observación	Integrar Sentry (DSN configurable) para capturar excepciones en API/frontend y documentar variables <code>SENTRY_*</code> .	Completado	Dipres	<code>entry-sdk</code> , middleware ASGI, docs/operations.md, docs/envs.md.
DAT-104	Metadatos	Implementar esquema <code>meta_*</code> , <code>jobs_*</code> , <code>monitor_*</code> en Postgres (lineage, runs, métricas).	Completado	Dipres	Alembic 202510090003 + test <code>test_meta_schema_migration.py</code> .
DAT-105	Metadatos	Instrumentar ETL para poblar <code>meta/files/lineage</code> y <code>jobs.runs</code> en cada corrida.	Completado	Dipres	BudgetBronze actualiza <code>meta/files/lineage</code> y <code>jobs_runs</code> (ver tests).
QA-106	Calidad de datos	Configurar checks automáticos (dbt/SQL) y registrar resultados en <code>meta.data_quality</code> .	Completado	Dipres	BudgetBronze publica checks de duplicados y consistencia (ver <code>tests/data/test_metadata_instrumentation.py</code>).
API-107	API Catálogo	Diseñar endpoints jerárquicos <code>/api/<ver>/data/{schema.table}</code> con paginación, filtros y lineage navegable.	Completado	Dipres	Endpoint <code>/api/{version}/data/{schema.table}</code> + pruebas <code>test_data_endpoint.py</code> .
UX-108	UI Navegación	Implementar layout persistente + rutas anidadas + breadcrumbs/árbol jerárquico (estado global).	Completado	Dipres	Navegación jerárquica con estado persistente + nuevas vistas <code>/mapa/</code> y <code>/reportes/</code> .
MON-109	Monitoreo	Construir vista <code>/monitoreo</code> (integraciones, precargas, ETL, API, latencias) y enlazar con logs/Sentry.	Completado	Dipres	Persistencia en <code>jobs_runs</code> , <code>jobs_integrations</code> , <code>monitor_api_metrics</code> ; dashboard muestra p50/p95/p99, RPS y checks ETL.
PER-110	Reportes	Reconstruir tablas gold + materialized views para métricas publicadas y diseñar refresco/invalidación caché.	Completado	Dipres	Pipelines escriben gold <code>presupuesto_*</code> , refrescan MVs y registran loader metrics / cache HTTP interno.

3 Bugs conocidos

ID	Descripción	Estado	Nota
WEB-101	Landing <code>/docs/<page></code> devuelve 404 si el archivo no existe. Confirmar cobertura de tests y manejo de enlaces.	Completado	Añadir prueba negativa (<code>tests/ui/test_docs_route.py</code>)

ID	Descripción	Estado	Nota
DATA-102	Falta dataset <code>ley_presupuesto</code> en manifest, endpoints retornan vacío.	Completo	Manifest y API <code>raw-assets</code> ya incluyen la ley vigente

4 Roadmap

4.1 Horizonte inmediato (0–2 semanas)

- [Seguimiento] Confirmar ejecución nocturna de `update_graph_catalog` y registrar snapshot resultante en `data/logs/`.

4.2 Próximo ciclo (2–6 semanas)

- Monitorear adopción del publicador de métricas y decidir si se elevan dashboards Prometheus (OPS-008 resuelto).
- Priorizar ampliación de pruebas para `etl/metadata` antes de incorporar nuevos pipelines.

4.3 Mediano plazo (>6 semanas)

- Re-evaluar tareas estratégicas una vez cerradas UI-201 y DATA-202.