Barafani, Facundo

Dubowez, Juan Cruz

Solución tecnológica para el aprovechamiento de datos del Instituto Nacional del Agua

Tesis para la obtención del título de grado de

Ingeniero Civil

Directores:

Carreño, Ignacio Luciano

Porrini, Federico Eduardo

Cuaderno de Trabajo Final

Solución tecnológica para el aprovechamiento de datos del Instituto Nacional del Agua

Ingeniería de Sistemas

Integrantes:

- BARAFANI, Facundo (2011665)
- DUBOWEZ, Juan Cruz (2011632)

Córdoba, 2023

Índice

Agradecimientos

El desarrollo de este trabajo final marca una de las etapas más importantes en nuestras vidas. Es la representación de todos los años vividos durante el transcurso de la carrera, en donde tuvimos el placer de aprender no solo temas relacionados a la carrera sino también otros temas que tienen que ver con el desarrollo íntegro de nuestra persona. Esta etapa no hubiese sido posible sin el apoyo de muchas personas, es por esto que queremos darle nuestro agradecimiento a:

Nuestras familias, por ser el pilar fundamental y estar presentes en todo momento, además de confiar en nosotros y alentarnos a seguir adelante.

Nuestros compañeros, que hicieron que estos tiempos sean de disfrute y que con el correr de los años se transformaron en amigos y futuros colegas.

La Facultad de Ingeniería de la Universidad Católica de Córdoba y todos los profesores, por darnos la oportunidad de crecer tanto personal como profesionalmente.

El equipo del INA-CIRSA, Ana Laura Ruibal Conti, Marcia Ruiz, María Inés Rodríguez, Martin Ludueña y Pablo Andreoni por abrirnos las puertas y brindarnos toda la información necesaria para el desarrollo de este Trabajo Final.

A nuestros tutores Ignacio Luciano Carreño y Federico Eduardo Porrini por su colaboración, tiempo y conocimiento brindado para lograr un buen desempeño en el Trabajo Final.

Abstract

The importance that bodies of water represent for human life entails a growing interest in analyzing water quality and understanding the nature of its changes. The San Roque Reservoir in the province of Córdoba presents a eutrophic condition that has been sustained over time. For almost 40 years, water quality, hydrological, and meteorological data from various sources have been collected, whose volume and complexity require the application of data mining techniques for their exploitation. The objective of this final project is to collect all the gathered data, automating the logic of preventing inconsistent data, with technologies that also favor the performance of subsequent analytical and predictive processes and establish a common structure so that future data is already collected with such technology. These subsequent processes will allow for efficient and adaptable analysis of information to business needs. It also provides data visibility, without requiring technical knowledge. Therefore, a visual interface is provided where analytics can be performed, with charts, for example, besides seeing the data updated in real time which also allows for quick decision-making in cases of alarming values.

The main objective of this graduation project is to provide the National Water Institute with a technological solution that processes in real-time the collections of hydrometeorological and limnological data. Its main processing manager is an open-source ETL that collects, processes, cleans the data, and stores it, the other strength lies in the structure and logic of the database and the modern visual interface that meets all the needs to represent the data and generate highly relevant information. The application of the same is very valuable for the management of the water body, giving visibility to the actual situation, both current and past, and especially making use of such a vast amount of data that is stored but is completely obsolete.

Resumen

La importancia que los cuerpos de agua representan para la vida humana, conlleva un interés creciente en analizar la calidad del agua y comprender la naturaleza de sus cambios. El Embalse San Roque, de ahora en adelante ESR, en la provincia de Córdoba presenta una condición eutrófica que se ha sostenido en el tiempo. A lo largo de casi 40 años, se han recolectado datos de calidad del agua, hidrológicos y meteorológicos de distintas fuentes, cuyo volumen y complejidad requieren la aplicación de técnicas de minería de datos para su explotación.

El objetivo de éste trabajo final es coleccionar todos los datos recopilados, automatizando la lógica de prevención de datos inconsistentes, con tecnologías que favorezcan también la realización de procesos analíticos y predictivos posteriores y establecer una estructura común para que los futuros datos ya sean recopilados con dicha tecnología. Dichos procesos posteriores permitirán hacer análisis de información de manera eficiente y adaptable a las necesidades de negocio. Así también dar visibilidad de los datos, sin requerir de conocimientos técnicos. Por eso se proporciona una interfaz visual donde se puede hacer analítica, con gráficos, por ejemplo, además ver los datos actualizados en tiempo real lo que también permite tomar decisiones rápidas en casos de valores alarmantes.

El presente proyecto de titulación tiene como objetivo principal, proveer al Instituto Nacional del Agua (INA) una solución tecnológica que procese en tiempo real las recopilaciones de datos hidrometeorológicos y de limnología. Su principal gestor de procesamiento es un ETL de código abierto que recolecta, procesa y limpia los datos y los almacena, el otro fuerte se encuentra en la estructura y lógica de la base de datos y la moderna interfaz visual que contempla todas las necesidades para representar los datos y generar información de alta relevancia.

La aplicación de los mismos resulta muy valiosa para la gestión del cuerpo de agua, dando visibilidad a la situación real, tanto actual como pasada, y especialmente dar uso a tan vasta cantidad de datos que se almacenan pero se encuentran completamente obsoletos.

Presentación del tema

Durante cuatro décadas, desde 1984, se han recopilado diversos datos hidrológicos, meteorológicos y de calidad de agua (tanto químicos como biológicos) de múltiples fuentes. Sin embargo, la abundancia y la complejidad de esta información presentan un desafío considerable para su adecuado aprovechamiento y análisis.

Uno de los principales desafíos radica en la ausencia de un sistema centralizado por parte del Instituto Nacional del Agua (INA) para el almacenamiento y la disponibilidad de estos datos recopilados. Esto impide la realización de visualizaciones, operaciones estadísticas y/o predictivas sobre los mismos. Además, el manejo de grandes volúmenes de datos requiere procesos de corrección y almacenamiento automatizados, actualmente realizados manualmente por operarios, lo que aumenta la probabilidad de errores y resulta en una carga de trabajo tediosa.

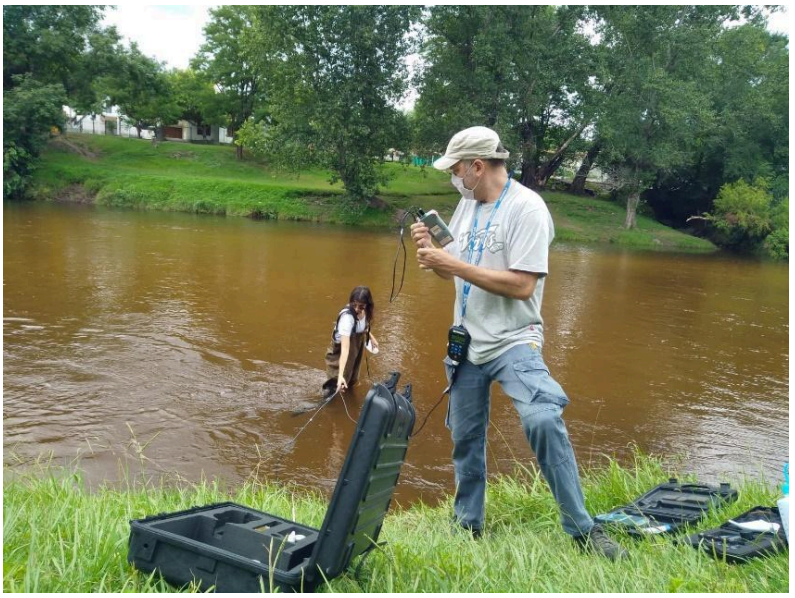
El proyecto consiste en la creación de un Sistema de Gestión de Datos, desarrollando un software de alta calidad que pueda procesar los datos brutos provenientes de los sensores, ubicados en las cuencas de los rios tributarios del ESR, también unifique planillas con resultados de análisis en laboratorio, hasta representarlos en tiempo real a través de gráficos interactivos.

Este sistema, entre otras cosas, permitirá a las entidades públicas y privadas correspondientes, así como al público en general, tener acceso a información completa y fiable, en tiempo real, sobre el estado del agua en el embalse, lo que contribuirá a su gestión y conservación efectiva.

La aplicación de técnicas de minería de datos[1] como ETL[2] para el procesamiento automático y en tiempo real de los datos provenientes de diversas fuentes, el uso de bases de datos relacionales normalizadas para un almacenamiento eficiente y la garantía de integridad de la información, junto con análisis avanzados, interfaces robustas y amigables, y alertas automatizadas, nos permitirá generar información valiosa para la toma de decisiones y mantener la calidad del agua en óptimas condiciones.

[1] Minería de datos: Es el proceso de explorar y analizar grandes conjuntos de datos para descubrir patrones, tendencias y relaciones ocultas que pueden ser útiles para la toma de decisiones y la predicción de futuros eventos o comportamientos.

[2] ETL: Proviene de las palabras en inglés, extract, transform y load. Que significa el proceso que contiene la extracción de datos, las transformaciones que se le aplican y su carga en un destino.



Imágenes de las dos grandes fuentes de datos (sensores hidrometeorológicos primero y muestras de agua para laboratorio segundo)

Glosario

LAyCA: Área de Limnología Aplicada y Calidad de Agua.

ESR: Embalse San Roque.

SGA: Sistema de Gestión de Alertas.

Limnología: Es una ciencia que se encarga del estudio ecológico de los ambientes acuáticos continentales (lagos, lagunas, embalses, ríos, arroyos, quebradas) tanto en aspectos físicos, químicos y biológicos.

Eutrófico: Acerca de un cuerpo de agua, que es rico en nutrientes y minerales y, por tanto, que presenta un exceso de crecimiento de algas, lo que ocasiona una disminución del contenido de oxígeno en detrimento de otros organismos.

ETL: Proviene de las palabras en inglés, extract, transform y load. Que significa el proceso que contiene la extracción de datos, las transformaciones que se le aplican y su carga en un destino.

Minería de datos: Es el proceso de explorar y analizar grandes conjuntos de datos para descubrir patrones, tendencias y relaciones ocultas que pueden ser útiles para la toma de decisiones y la predicción de futuros eventos o comportamientos.

Diagnóstico

Antes de iniciar cualquier tarea de solución, realizamos un análisis de la situación previa.

El flujo de datos:

El flujo de datos se origina en dos fuentes principales: en primer lugar, aproximadamente 113 sensores funcionales, distribuidos en 22 estaciones de medición a lo largo de las cuencas de los ríos que desembocan en el ESR. Estos sensores recopilan datos de variables hidrometeorológicas, los cuales son transmitidos mediante señal VHF a un decodificador en la sede del INA-CIRSA en Villa Carlos Paz. Allí, un software enlatado[3] llamado Datawise procesa y almacena estos datos en una base de datos Postgres denominada SGA (Sistema de Gestión de Alertas), alojada en un servidor local. Además, este software tiene la capacidad de exportar información en archivos de texto plano, los cuales se utilizan para rastrear datos o para compartirlos con empleados que los soliciten. Sin embargo, estos archivos suelen estar dispersos y carecen de un lugar concreto de almacenamiento.

La segunda fuente de datos consiste en hojas de cálculo que registran mediciones mensuales de calidad del agua en el ESR a distintas profundidades y en los ríos que desembocan en él. Estas hojas incluyen tanto los datos ingresados manualmente por el INA como las hojas de cálculo proporcionadas por Aguas Cordobesas (ACSA), que detallan los resultados de sus propias investigaciones. Para integrar y gestionar los datos de ambas entidades, el INA realiza una consolidación manual en una única hoja de cálculo de Excel. Posteriormente, esta hoja se almacena en Google Drive para facilitar su acceso y referencia futuros.

[3] Software enlatado se le llama a un software comercializado o prefabricado que ha sido desarrollado y empaquetado para cumplir con ciertas funciones específicas. Esto implica que el software ya está configurado y listo para su uso inmediato, sin necesidad de modificaciones significativas por parte del usuario final.

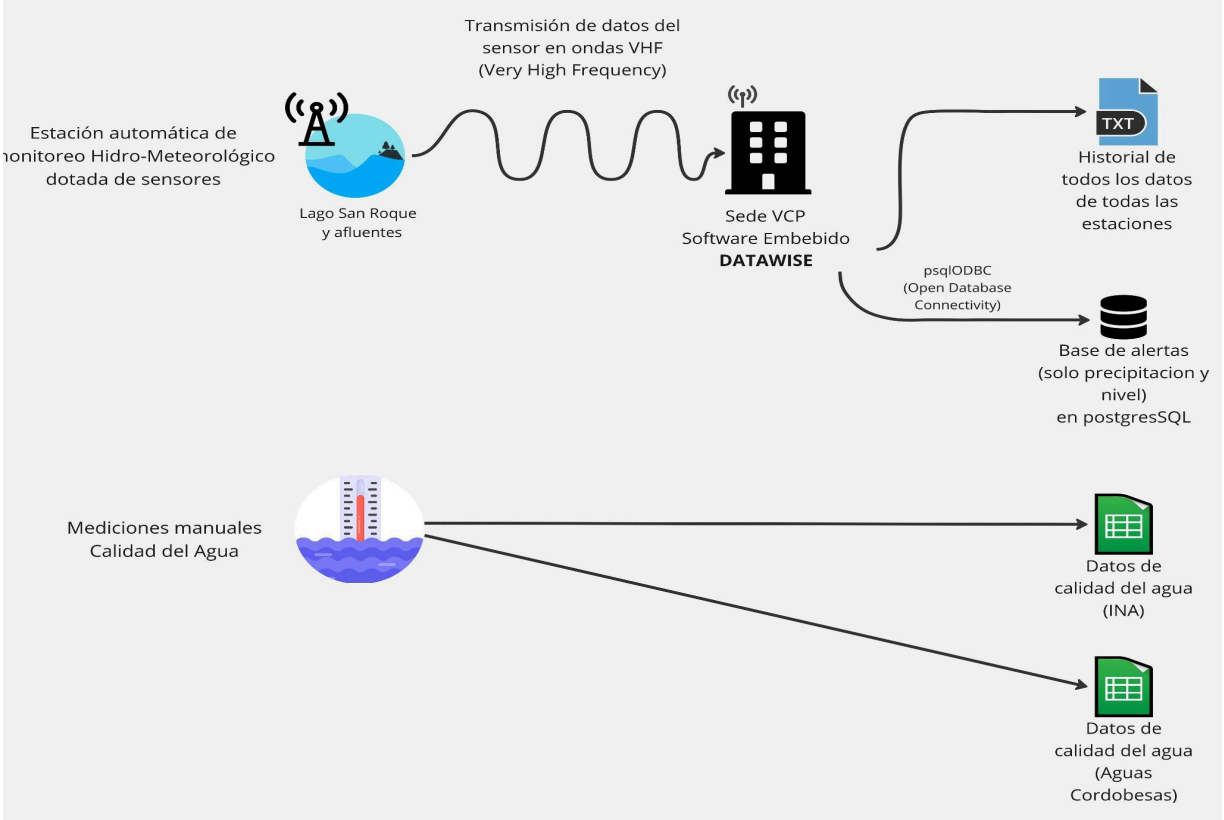
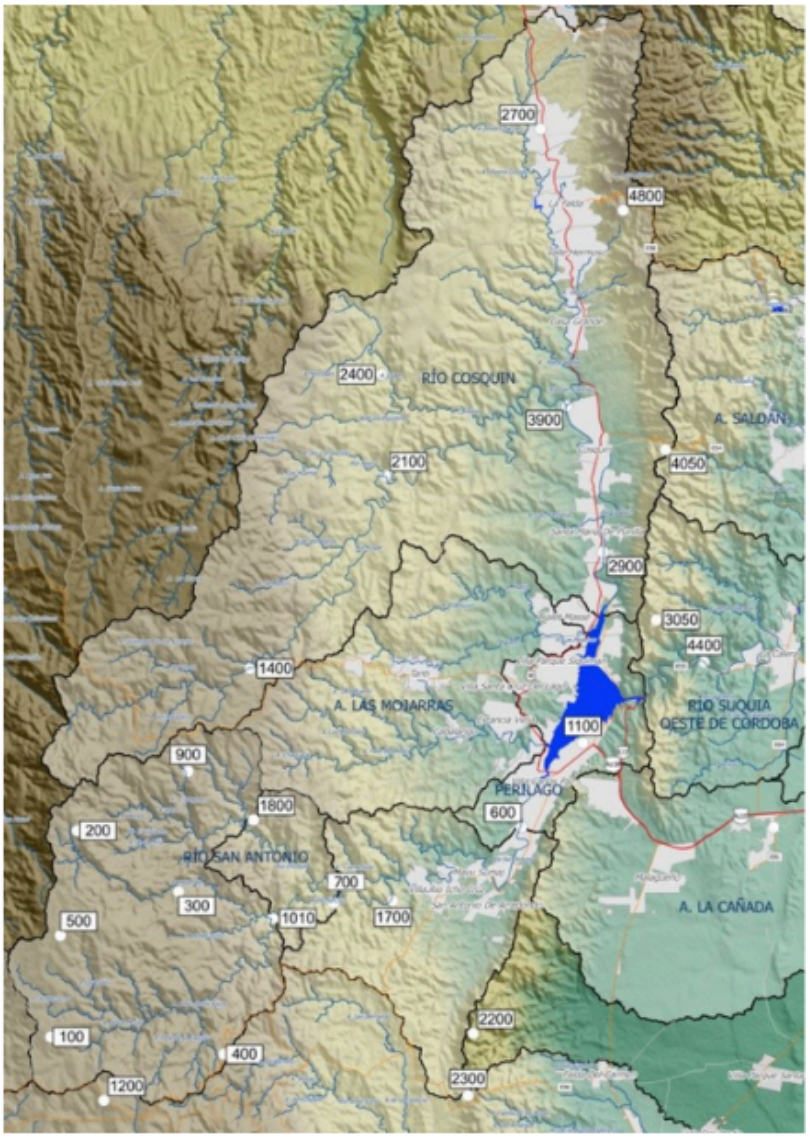


Gráfico de arquitectura de flujo de datos existente. Año 2023



Distintas estaciones hidrometeorológicas que reportan datos a la central en VCP(1100). Diciembre del año 2023.

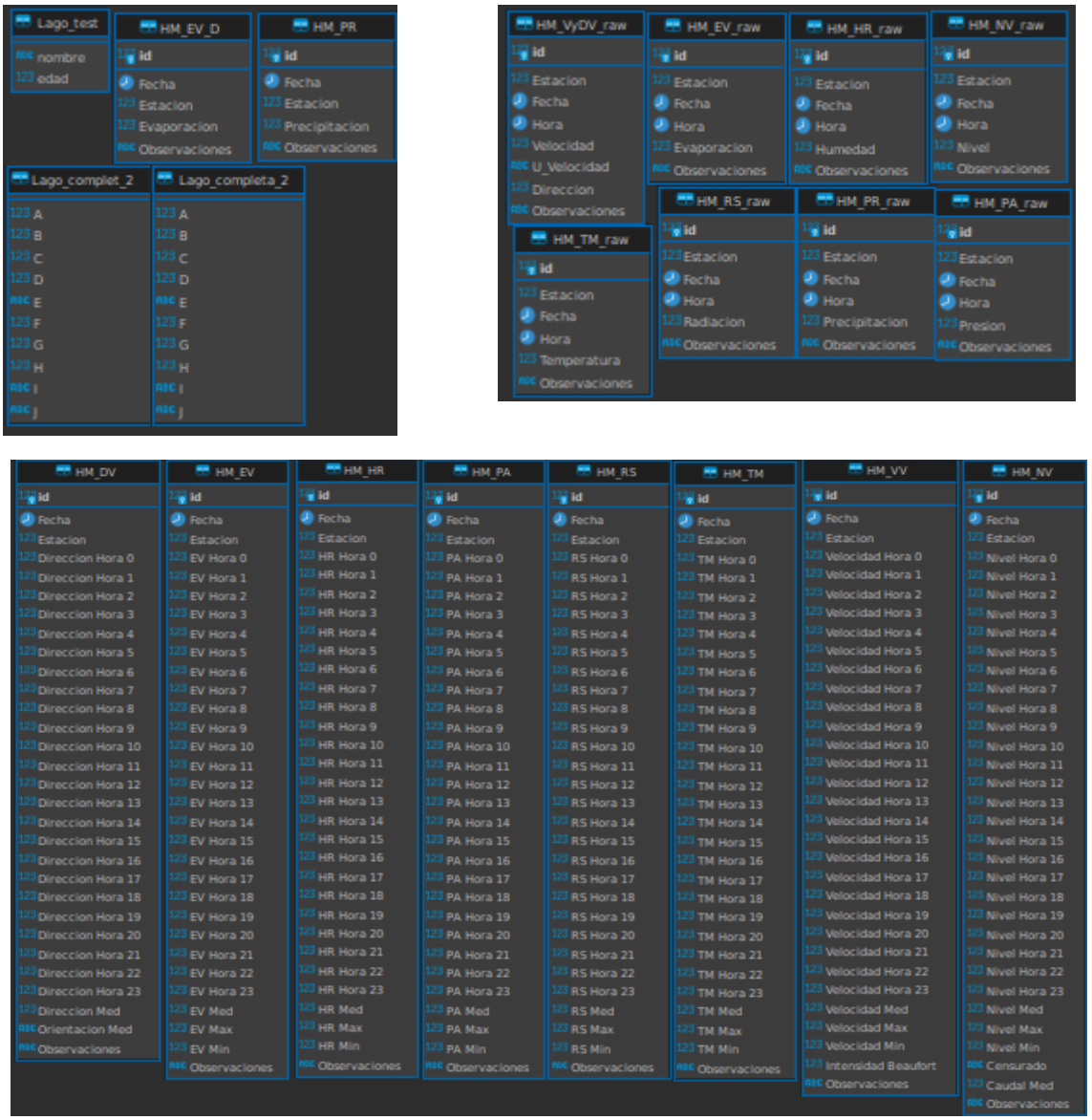
La situación a nivel hidrometeorológico:

El sistema enlatado en Villa Carlos Paz recibe por señal VHF la medición y la almacena en una base propia, la cual si se desea acceder, este exporta los datos en formato texto plano. Así también a través de ODBC envía los datos en tiempo real a una base de datos postgres llamada SGA pero que almacena solo los sensores que miden precipitación y nivel. Por otro lado, hay una base de datos estática creada por el Ing. Pablo Andreoni para su tesis, la cual contiene todas las mediciones históricas de todos los sensores existentes desde el año 1984 a 2017. En total se encuentran millones de datos recopilados por sensores alrededor del lago San Roque.

El inconveniente radica en que, si bien la base SGA registra mediciones en tiempo real, se limita a almacenar únicamente dos variables. En relación con la base de datos histórica, presenta desafíos significativos: su estructura no es la más adecuada, carece de

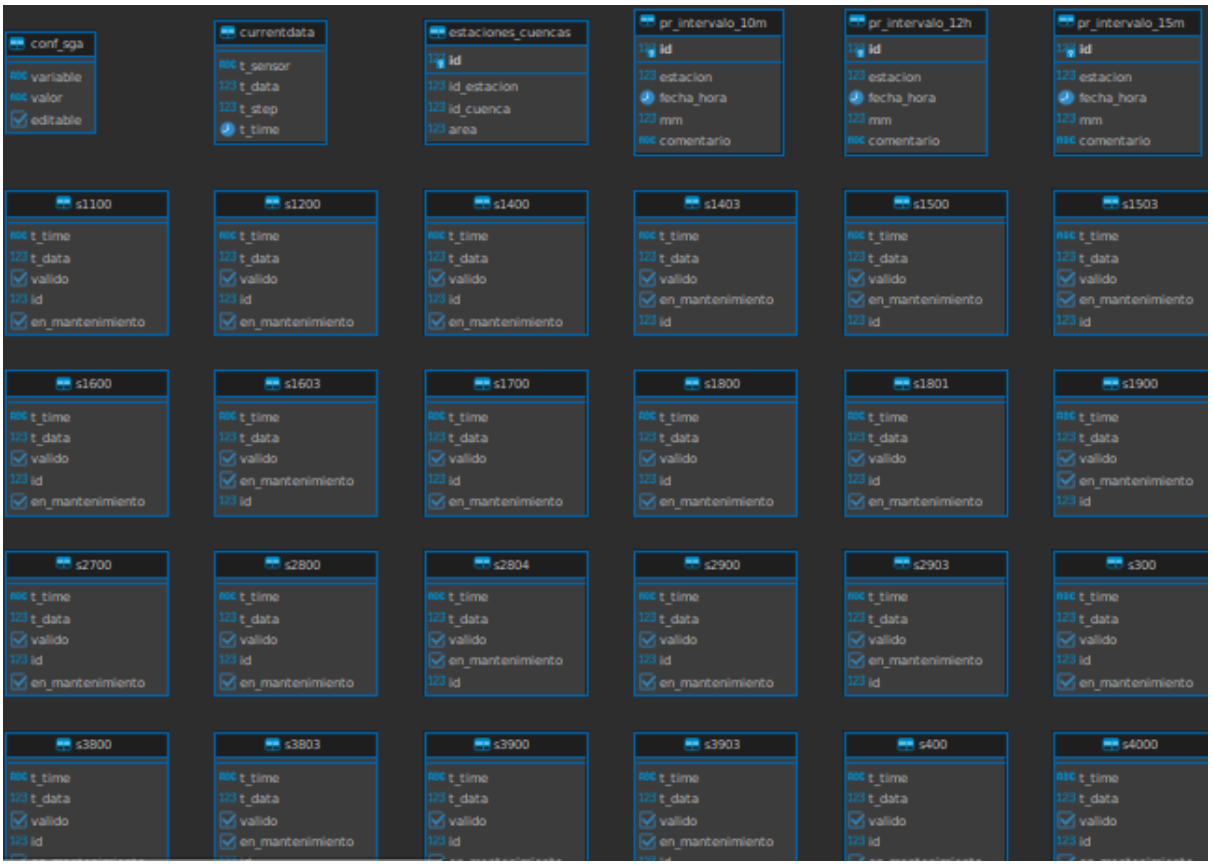
escalabilidad, y requiere de normalización y optimización en varias tablas. Además, hay campos que necesitan ser reevaluados. Un aspecto crítico es su naturaleza estática, sin actualizaciones ni inclusión de nuevos datos desde el año 2017. Asimismo, existe una discrepancia en la estructura de los archivos txt comparada con la estructura empleada en PostgreSQL.

Se muestran a continuación algunas tablas de la base de datos histórica:



Imágenes: estructura de la base de datos histórica existente.

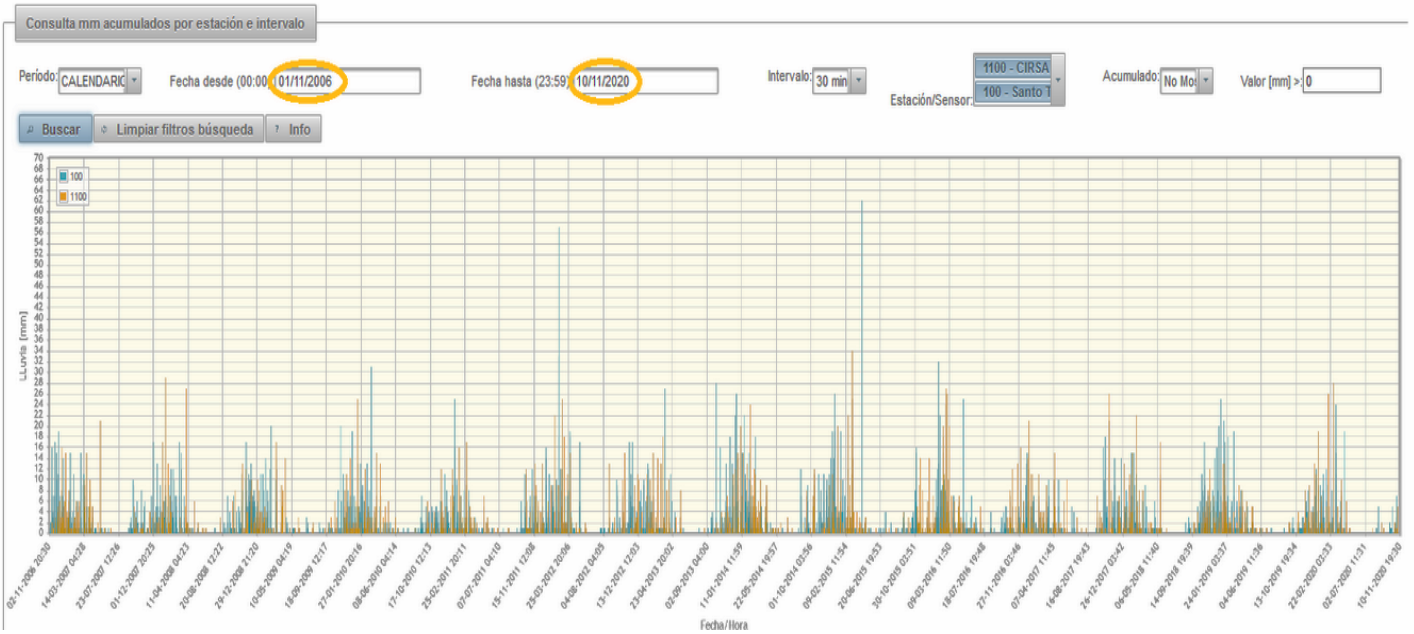
Se muestran a continuación algunas tablas de la base de datos SGA:



Porción de la estructura de la base de datos SGA existente.

Medio para visualizar los datos y obtener información:

- No disponible para todos: No todos los usuarios tienen acceso al sistema.
- Sin capacidad de personalización: El sistema no permite ajustes o adaptaciones según las necesidades individuales.
- Limitado: El sistema tiene restricciones en cuanto a funcionalidades o alcance.
- Engorroso para descubrir datos importantes: El proceso de búsqueda y análisis de datos relevantes es complicado y poco eficiente.
- Visualización para datos de hidrometeorología limitada: Solo se permite visualizar datos para las variables de precipitación y nivel.
- Falta de un medio de visualización para datos de calidad del agua: No existe una herramienta o plataforma dedicada para visualizar los datos de calidad del agua de manera efectiva.

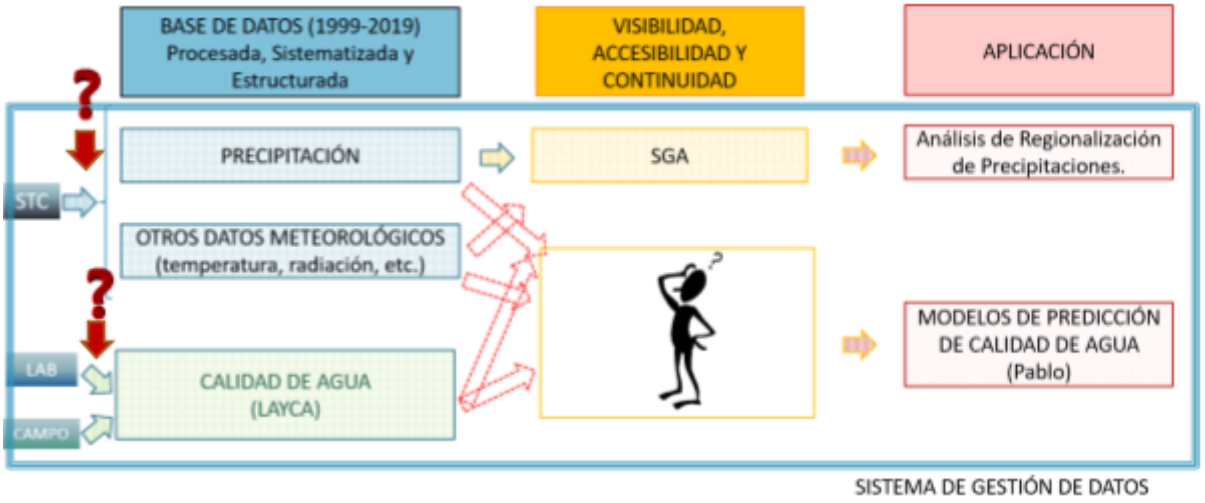


Medio de visualización de información propio del INA, para datos hidrometeorológicos.

Soporte y gobierno de datos:

En hidrometeorología, solo existe soporte para datos de precipitación y nivel de agua. El resto de las variables han sido sobreseídas, lo que implica cada vez menos sensores y periodos donde faltan datos.

Calidad de agua no tiene un soporte sistematizado ni automatizado.



Representación de la necesidad brindada por el INA.

La situación a nivel calidad de agua:

Los datos recopilados abarcan una amplia gama de parámetros, incluyendo aspectos químicos, físicos y biológicos, como la presencia de algas. Además, se incluyen detalles descriptivos que ayudan a contextualizar la extracción de las muestras.

El INA lleva a cabo la recopilación y análisis de muestras una vez al mes. Este proceso implica la extracción manual de muestras en el lago, ríos y la usina, realizadas a diferentes profundidades. Posteriormente, se completan hojas de cálculo con los resultados obtenidos.

Por otro lado, ACSA realiza estudios similares y envía sus hojas de cálculo al INA. Sin embargo, el proceso de integración de estas hojas de cálculo en la base de datos del INA es completamente manual. Esta tarea se ve dificultada por la variabilidad en el formato de los datos, la inclusión o exclusión de variables entre períodos y cambios en las técnicas de medición. Esta falta de estandarización dificulta la unificación de los datos y resulta en un proceso tedioso.

Además, el uso de hojas de cálculo para almacenar y gestionar esta información presenta limitaciones, especialmente considerando la gran cantidad de datos y la complejidad de la información cruzada. Se hace evidente la necesidad de implementar un sistema más eficiente y adecuado para el manejo de esta información.

Nº de Monitoreo	Registro	Muestra	Fecha	Día	Mes	Año	Año Hidrológico	Época	Área	Sitio	Código	Descripción	HORA	Z (m)	pH	pH Lab	OD (mg/l)	% sat OD	Cond. (µS/cm)	Cond (µS/
315	16333	138111	31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca	C1	SUBSUPERFICI	9:44	0,2	8,57	8,4	7,61	103,4	288	
315	16334		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				1,0	8,61		7,54	102,7	308	
315	16335		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				2,0	8,62		7,44	101,3	319	
315	16336		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				3,0	8,63		7,15	97,5	321	
315	16337		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				4,0	8,57		5,79	78,3	326	
315	16338	138112	31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca	C2	FOTICA	9:50	5,0	8,48	7,9	5,65	76,0	328	
315	16339		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				6,0	8,37		5,70	76,5	323	
315	16340		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				7,0	8,28		5,56	74,8	322	
315	16341		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				8,0	8,15		5,22	70,0	320	
315	16342		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				9,0	7,85		2,20	29,4	317	
315	16343		31-ene-23	31	1	2023	2022-2023	VERANO	C	Ca				10,0	7,57		1,78	23,8	315	
315	16344		31-ene-23	31	1	2023	2022-2023	VERANO	0	Ca				11,0	7,82		0,75	9,4	319	

Porción de la hoja de cálculo de calidad de agua. Almacena los resultados de todos los muestreos del 2023 en el ESR. También están integradas las muestras que envió ACSA.

Monitoreo	Registro	Muestra	Fechas	Dia	Mes	Año	Año Hidrologico	Epoca	Rio	Sitio	Código	Hora	Temp (H2O °C)	pH LAB	pH	Conduct (µS/cm)	Conduct. Lab. (µS/cm)	TDS (g/L)	OD (mg/L)	Sat. de OD %	Turb. (NTU)	Turb. Lab (NTU)	TOC (mg/l)
270	1187	138117	31-ene-23	31	1	2023	2022-2023	VERANO	RÍO COSQUÍN	VILLA CAEIRO	RCQ	8:57	23,8	7,8	5,63	224	146	0,144	8,42	108,2		6,9	8,4
270	1188	138119	31-ene-23	31	1	2023	2022-2023	VERANO	ARROYO LAS MOJARRAS	BALNEARIO FANTASIO	RLM	10:55	24,6	7,8	5,64	255	162	0,163	8,56	111,3		4,2	13,8
270	1189	138118	31-ene-23	31	1	2023	2022-2023	VERANO	ARROYO LOS CHORRILLOS	BALNEARIO EL DIQUECITO	RLC	12:30	24,5	7,2	5,75	121	70	0,080	8,43	109,3		5,1	11,2
270	1190	138120	31-ene-23	31	1	2023	2022-2023	VERANO	RÍO SAN ANTONIO	AZUD	RSA	13:35	22,4	6,9	5,39	49	96	0,033	8,43	106,4		23,0	10,2
271	1191	139137	28-feb-23	28	2	2023	2022-2023	VERANO	RÍO COSQUÍN	VILLA CAEIRO	RCQ	9:03	23,1	8,2		474	411	0,308				15,0	4,9
271	1192	139139	28-feb-23	28	2	2023	2022-2023	VERANO	ARROYO LAS MOJARRAS	BALNEARIO FANTASIO	RLM	10:20	23,1	7,7		319	267	0,207	7,30	92,1		1,2	7,1
271	1193	139138	28-feb-23	28	2	2023	2022-2023	VERANO	ARROYO LOS CHORRILLOS	BALNEARIO EL DIQUECITO	RLC	11:45	24,7	7,9		256	205	0,167	9,05	117,0		2,8	5,2
									RÍO SAN														

Porción de la hoja de cálculo de calidad de agua ya que posee 180 columnas. Almacena los resultados de todos los muestreos en ríos del año 2023, también hay una para usina.

FECHA-HORA-EXTRAC.	Nro de material	Dirección	Nro lote insp	2,4 D (µg/l	Alcalinidad (mg/l	Aldrin (µg/l	Alfa Endosulfán (µg/l	Alfa HCH (µg/l	Amo
8/22/23 9:05	000000000LSRC10000	LAGO SAN ROQUE C1	900000144380		72				
8/22/23 9:15	000000000LSRC20000	LAGO SAN ROQUE C2	900000144381						
8/22/23 9:20	000000000LSRC40000	LAGO SAN ROQUE C4	900000144382						
8/22/23 9:40	000000000LSRC50000	LAGO SAN ROQUE C5	900000144383						
8/22/23 10:00	000000000LSRTAC100	LAGO SAN ROQUE TAC1	900000144384		72				
8/22/23 10:10	000000000LSRTAC200	LAGO SAN ROQUE TAC2	900000144385						
8/22/23 10:20	000000000LSRTAC400	LAGO SAN ROQUE TAC4	900000144386						
8/22/23 10:30	000000000LSRTAC500	LAGO SAN ROQUE TAC5	900000144387						
8/22/23 14:15	000000000LSR U0000	LAGO SAN ROQUE USINA	900000144388		73				
8/22/23 11:11	000000000LSRDCQ100	LAGO SAN ROQUE DCQ1	900000144389	15	101	0,015	10	0,5	
8/22/23 12:15	000000000LSRDSA100	LAGO SAN ROQUE DSA1	900000144390	15	107	0,015	10	0,5	
8/22/23 9:10	000000000LSRCQ0000	LAGO SAN ROQUE RIO COSQUIN	900000144391						
8/22/23 12:00	000000000LSRLC0000	LAGO SAN ROQUE LOS CHORRILLOS	900000144392	15	116	0,015	10	0,5	
8/22/23 10:20	000000000LSRLM0000	LAGO SAN ROQUE LAS MOJARRAS	900000144393	15	161	0,015	10	0,5	
8/22/23 13:15	000000000LSRSA0000	LAGO SAN ROQUE RIO SAN ANTONIO	900000144394						
8/22/23 10:42	000000000LSR000000	Estacion Metereológica LSR	900000144426						

Porción de la hoja de cálculo que envía ACSA. Almacena todos los valores de las muestras realizadas en el mes. Incluye lago, usina y ríos.

Objetivos

Globales

- Desarrollar un sistema integral de monitoreo y gestión de datos para el Instituto Nacional del Agua (INA) que permita recopilar, procesar, almacenar y visualizar información meteorológica y de calidad del agua en tiempo real.
- Recopilar toda la información previa y almacenarla en la nueva base de datos que usará este sistema, para tener todo en un mismo lugar y que la base a utilizar posea la información completa.
- Reducir los costos asociados a los análisis periódicos de calidad del agua al implementar un monitoreo continuo y automatizado de los datos.
- Aumentar la eficiencia y la productividad al reducir la necesidad de monitoreo manual y al automatizar procesos tediosos.
- Reducir los riesgos relacionados con la calidad del agua mediante la implementación de un preprocesamiento en tiempo real ahorrando así tiempo en el análisis otorgando garantía en la seguridad y la salud de los consumidores.
- Mejorar la toma de decisiones al proporcionar datos precisos y en tiempo real para la toma de decisiones rápidas y eficientes.
- Automatizar operaciones tediosas como la entrada de datos manuales y la unificación de hojas de cálculo otorgando así integridad al desvincular el error humano de la carga de datos de calidad del agua.
- Garantizar la robustez, disponibilidad y confiabilidad de la información sin datos incoherentes ni fallos relacionados con la entrada manual.

Específicos

- Construcción de una base de datos que contuviera en forma coherente, estructurada, validada y accesible, los datos de Calidad de Agua e Hidrometerorológicos monitoreados desde su comienzo hasta la actualidad.
- Desarrollar un módulo de adquisición de datos que recopile información de sensores y la envíe al sistema central en tiempo real, para su procesamiento y almacenamiento.
- Implementar un módulo de procesamiento de datos que realice la transformación y limpieza de los datos hidrometeorológicos crudos y que incluya también la unificación y normalización de los datos de ACSA y el INA (calidad de agua).
- Establecer un módulo de almacenamiento de datos que almacene los datos procesados en la base de datos de alta disponibilidad.
- Desarrollar un módulo de visualización de datos que presente la información en una interfaz gráfica, utilizando tecnologías y herramientas que faciliten la comprensión de los datos, como gráficos, tablas y consultas amigables.
- Automatizar la recopilación de datos en cuerpos de agua a partir de hojas de cálculo, garantizando la actualización constante de la información.
- Realizar pruebas exhaustivas y garantizar la calidad y confiabilidad del sistema en su conjunto.
- Implementar el sistema en una infraestructura que sea práctica, efectiva, confiable, segura, y fácilmente comprensible y manipulable por futuros operadores.
- Generar la documentación suficiente para explicar el sistema en su completitud, información previa al trabajo que se fue requiriendo y no estaba asentada. Esto permitirá no depender de personas concretas, ni de su memoria o de documentos que pueden ser extraviados.
- Preparar las bases para futuras aplicaciones de modelos de predicción basados en los datos recopilados y procesados.

Marco teórico

Se separa primero en contenido teórico necesario del dominio del negocio y luego el contenido teórico propio de la ingeniería aplicada.

A nivel dominio del negocio:

Los lagos son cuerpos de agua muy importantes por sus variadas funciones y múltiples usos tales como albergar vida acuática, provisión de agua para consumo humano y regulación de flujos hídricos, entre otros (Jorgensen & Vollenweider, 1989). La calidad del agua de los lagos depende de múltiples factores tanto intrínsecos (procesos físico-químicos y biológicos dentro del ecosistema acuático), como extrínsecos (condiciones meteorológicas, climáticas, tipo de suelo, usos del cuerpo de agua, entre otros). Los procesos relacionados a la calidad del agua pueden variar en períodos cortos (horarios, diarios o estacionales) y en períodos o ciclos más largos (quinquenios, décadas, entre otros).

Los procesos que ocurren en períodos más largos suelen estar asociados a la variabilidad climática de la región. La evaluación de los mismos requiere de mediciones por largos períodos de tiempo generando extensas bases de datos que comprenden múltiples variables de calidad del agua, hidrológicas y meteorológicas. Dada la magnitud de estos repositorios, la manipulación, integración y análisis de los datos resulta complejo. Surge entonces la necesidad de utilizar nuevas herramientas que permitan explotar la información en todo su potencial y en tiempos razonables.

El ESR, ubicado en la Provincia de Córdoba es un lago artificial construido en 1888 con el fin de proveer agua para consumo humano, riego y energía hidroeléctrica.

Actualmente es uno de los centros turísticos más importantes del país con un extenso uso recreativo, recibiendo la localidad de Villa Carlos Paz aproximadamente unos 370.000 visitantes por quincena en verano según estimación oficial (Secretaría de Turismo de Villa Carlos Paz, 2015). La calidad de su agua se ha deteriorado con el tiempo y actualmente presenta un estado eutrófico avanzado (Instituto Nacional del Agua, 2019) generando serios problemas ambientales (mortandad de peces y malos olores, entre otros), económicos (inversión en infraestructura sofisticada para potabilización del agua, remediación del lago y disminución del turismo, entre otros) y sociales (efectos sobre la salud pública y cierre de escuelas, entre otros).

Se sabe que el enriquecimiento por nutrientes inorgánicos (en particular nitrógeno y fósforo) favorece la eutrofización de los lagos (Sigee, 2005); aunque al mismo tiempo, se

sabe que las condiciones ambientales generan tanto un efecto directo sobre los procesos biológicos, tal es el caso de la fotosíntesis, sensible a la radiación solar y la temperatura (Amé et al., 2017), como indirecto, por ejemplo la dinámica y magnitud de la liberación de fósforo de los sedimentos, afectada también por la temperatura y los vientos (Rodríguez et al., 2000). Esos procesos, sumado a los fenómenos estacionales asociados, la dinámica hidrológica del embalse y el cambio climático a largo plazo, dan indicio de que estas condiciones ambientales inducen un efecto sobre el estado trófico.

Tras la introducción teórica, se ofrecen al lector los conceptos básicos de meteorología y limnología, así como los detalles sobre los materiales y métodos utilizados. Se describen las características del objeto de estudio, el ESR, junto con las fuentes de datos disponibles para la calidad del agua y las mediciones hidrológicas y meteorológicas. Además, se explican los procedimientos empleados para depurar, transformar, seleccionar e integrar los datos en cada caso.

Conceptos de meteorología

La meteorología es el estudio de la atmósfera y sus fenómenos. El clima, en tanto, es la condición de la atmósfera en un momento y lugar particular (Ahrens & Henson, 2017).

Algunas de las variables más importantes que caracterizan el clima y que son las que se utilizarán, se detallan a continuación, siguiendo los conceptos de Ahrens y Henson (2017)

Temperatura: “es una medida de la velocidad promedio (movimiento promedio) de los átomos y moléculas, donde altas temperaturas se corresponden con velocidades promedio más altas y viceversa”. En particular se analizará la temperatura del aire, que remite al grado de calor o frío de los gases que lo componen.

Radiación solar: “es la energía transferida desde el sol a un objeto. Se puede pensar la radiación como un flujo continuo de partículas o fotones, que son paquetes discretos de energía”.

Humedad: “es una medida de la cantidad de vapor de agua en el aire”. En particular, la humedad relativa, variable de estudio en este trabajo es “el cociente entre la cantidad real de vapor de agua en el aire y la máxima cantidad de vapor de agua requerida para la saturación a una temperatura (y presión) particular”.

Precipitación: “es cualquier cantidad de agua (líquida o sólida) que cae desde una nube y alcanza el suelo”.

Presión atmosférica: “es la presión ejercida por una masa de aire sobre una región”. Es igual a la temperatura por la densidad del aire por una constante. A mayor altitud la presión atmosférica disminuye.

Viento: “es el movimiento en masa del aire por diferencia de presión atmosférica, “la compensación de las diferencias de presión atmosférica entre dos puntos” (Roth, 2003). En particular, el viento está definido por una velocidad, o distancia recorrida por unidad de tiempo; y también una dirección, determinada por el lugar desde y hacia donde sopla”. Ambas variables se analizarán en este trabajo.

Estas variables meteorológicas se encuentran íntimamente relacionadas. A continuación se muestra un diagrama simplificado donde se establecen algunas de las principales relaciones entre las variables mencionadas anteriormente.

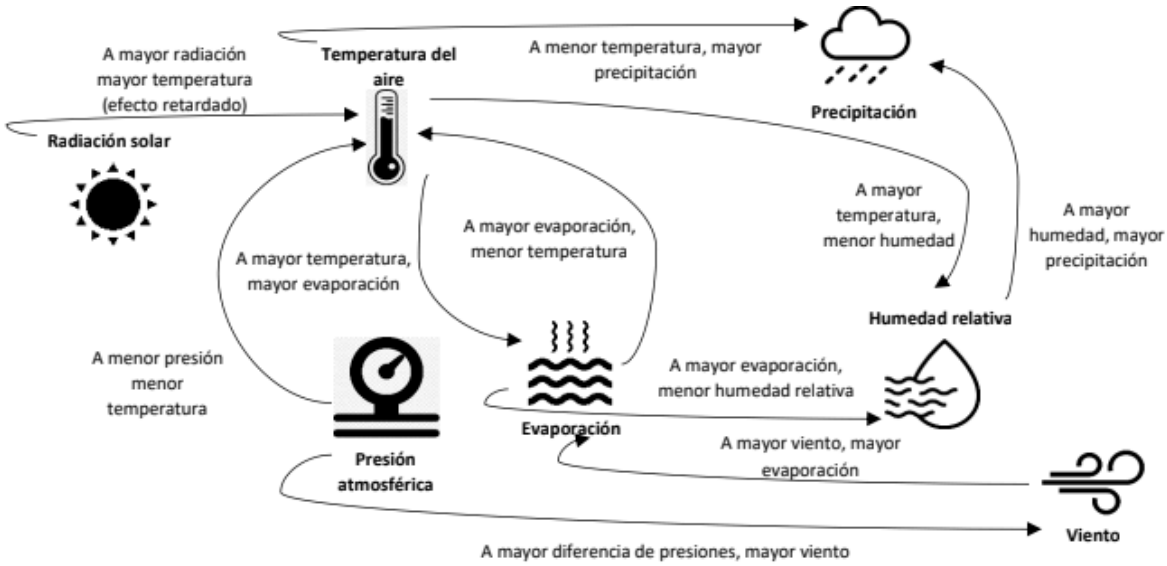


Diagrama de algunas de las principales relaciones entre las variables meteorológicas.

Conceptos de limnología

El estudio de las particularidades de la calidad del agua del ESR y sus ríos tributarios, requiere un abordaje profundo desde la limnología. Esta ciencia es una rama de la ecología que se dedica al estudio de los ecosistemas acuáticos continentales, tal es el caso del agua dulce (en el embalse y en los ríos). En términos más formales, es el estudio de las relaciones funcionales y de productividad de las comunidades de agua dulce y la manera en cómo las afecta el ambiente químico, físico y biológico (Roldán Pérez & Ramírez Restrepo, 2008).

El objetivo de esta sección es dar un marco a la posterior interpretación de los resultados en términos de las variables más comunes que definen el estado de un cuerpo

de agua, desde el punto de vista de la biología y las principales interacciones que se producen entre ellas. Algunas de estas variables se detallan a continuación, siguiendo distintos autores:

pH: “es una abreviatura de potencial de hidrogeniones (H+), e indica la concentración de estos en el agua”. Esta variable indica la intensidad de la condición ácida o básica del agua (Roldán Pérez & Ramírez Restrepo, 2008).

Oxígeno disuelto: se refiere a la cantidad de O2 disuelto en el agua.

Porcentaje de saturación de oxígeno disuelto: “es el porcentaje máximo de oxígeno que puede disolverse en el agua a una presión y temperaturas determinadas” (Roldán Pérez & Ramírez Restrepo, 2008).

Conductividad eléctrica: se refiere a la capacidad de una solución para conducir una corriente eléctrica en función de la concentración de iones presentes en ella (Roldán Pérez & Ramírez Restrepo, 2008).

Sólidos disueltos totales: se refiere a la concentración total de sustancias o minerales disueltos (Roldán Pérez & Ramírez Restrepo, 2008).

Temperatura: es la temperatura (tal y como se definió en la sección meteorología) del agua.

Potencial óxido-reducción: químicamente hablando, se refiere a las reacciones rédox, “aquellas en que cambia el estado o grado de oxidación de las especies reaccionantes, se produce un intercambio de electrones entre los reactivos” (Burriel Martí et al., 1985).

Turbiedad: “es el grado en que el agua interfiere con la transmisión de luz a través de ella”, por este motivo la luz se reemite y no se transmite a través de la suspensión. La turbidez originada en el agua por el aporte de materiales externos (por ejemplo, de creciente en el embalse) se denomina alóctona y la producida dentro del mismo cuerpo de agua, autóctona. Incide directamente en la producción primaria y en el flujo de energía en el ecosistema, asociado con la luz (Roldán Pérez & Ramírez Restrepo, 2008).

Color del agua: “está constituido por la luz no absorbida” (Roldán Pérez & Ramírez Restrepo, 2008).

Condición de estratificación: se refiere a la formación de capas en el agua con propiedades distintas según algún criterio. La estratificación térmica se refiere a la formación de dos capas bien definidas: en regiones de clima templado, se tiene una caliente

superficial y otra fría profunda (en regiones frías es a la inversa), divididas por una zona de descenso brusco llamada termoclina. La zona superficial recibe el nombre de epilimnio y la profunda hipolimnio (Roldán Pérez & Ramírez Restrepo, 2008). Otro tipo de estratificación es la lumínica, que se da por las características de turbiedad del agua, donde asimismo se tienen capas diferenciadas en términos de la disponibilidad de luz. En ausencia de estratificación el cuerpo de agua se dice que está en condición de mezcla.

Nitrógeno y sus variantes: La importancia de este elemento en el agua radica en que es el componente fundamental de las proteínas, las cuales constituyen la base estructural de los seres vivos. El nitrógeno puede ser tomado por algas y plantas en tres formas: como nitratos (NO3-), como ion amonio (NH4+) y como nitrógeno molecular (N2) (Roldán Pérez & Ramírez Restrepo, 2008).

Fósforo y sus variantes: Se trata del elemento biológico que desempeña el papel más importante en el metabolismo biológico. Es el menos abundante de los elementos y el que desempeña un rol limitante en la productividad primaria (Roldán Pérez & Ramírez Restrepo, 2008). Existen distintas formas de medir el fósforo en el agua: fósforo reactivo soluble y fósforo total, que se obtiene por distintos pretratamientos de la muestra, entre otros.

Clorofila- a: “Es uno de los pigmentos que caracterizan a algas y cianobacterias. Hay varios tipos de clorofila, siendo la clorofila-a la que domina como pigmento fotosintético primario y es a menudo la utilizada para estimar biomasa” (Wetzel & Likens, 2000).

Carbono orgánico total (COT): “Se refiere a todo el material orgánico contenido en el agua” (Roldán Pérez & Ramírez Restrepo, 2008).

Fitoplancton: “Son los productores primarios de plancton” (Roldán Pérez & Ramírez Restrepo, 2008). “El fitoplancton de agua dulce se compone de bacterias fotosintéticas y algas, que varían en tamaño y forma de los organismos” (Sigee, 2005).

Estas variables limnológicas (físicas, químicas y biológicas) se encuentran relacionadas y el diagrama que se observa en la siguiente figura resume algunas de las principales asociaciones.

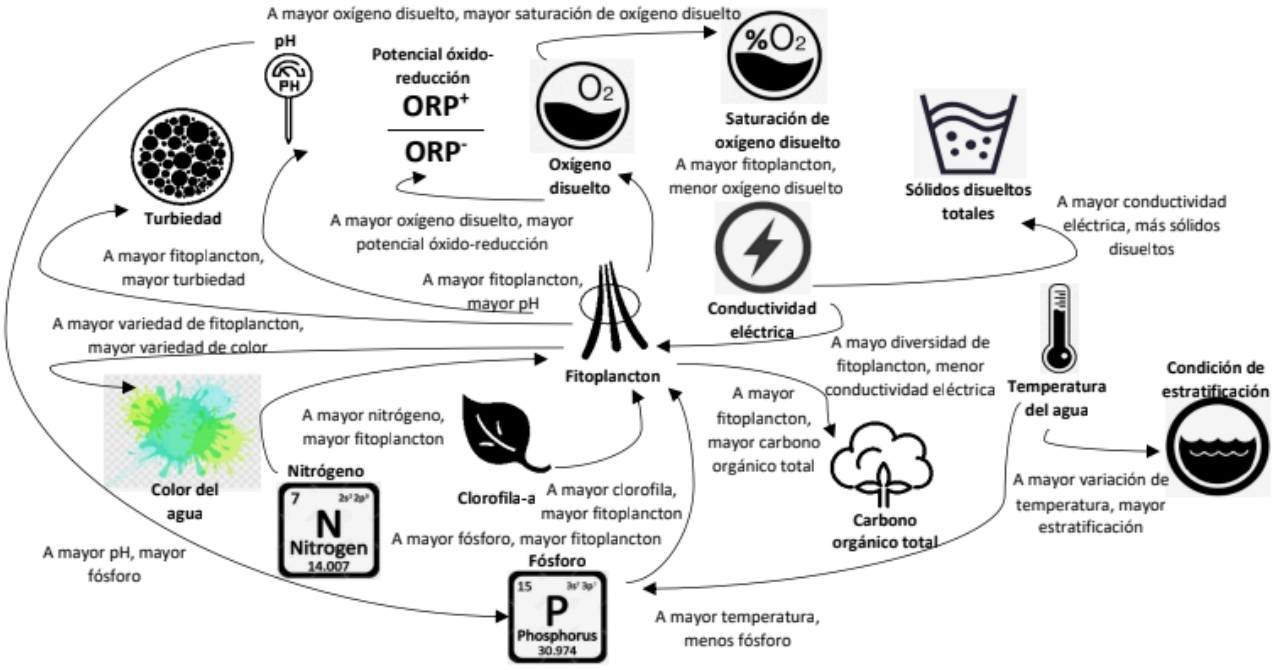


Diagrama de algunas de las principales relaciones entre las variables limnológicas.

Sitio de estudio: Embalse San Roque

El ESR se encuentra ubicado en la provincia de Córdoba, dentro del departamento de Punilla. Su ubicación exacta es 31° 22' 36" (S) - 64° 27' 54,6" (O) y se encuentra a una altitud de 608 msnm (metros sobre el nivel del mar). Se denomina embalse, y no lago en tanto se trata de un cuerpo de agua artificial, resultante del embalsamiento de sus actuales ríos tributarios. El dique o represa que lo forma, es una estructura de hormigón que embalsa 210 Hm3 de agua, ocupando una superficie de 1.630 Ha, siendo su profundidad media de 13 m. El tiempo de residencia, que es el tiempo promedio que toma rellenar la cuenca del embalse con agua si se vaciara, proporciona una medida de la circulación de agua dentro de los ecosistemas acuáticos (Sigee, 2005), es de 0,6 años. La cota del vertedero alcanza una altura de 35,5 m(Instituto Nacional del Agua, 2014).

El dique se construyó dos veces, la primera en 1888 (referencias posteriores a paredón viejo, se refieren a parte de la estructura que contenía al embalse en esta primera construcción) y la segunda en 1944. El embalse abastece de agua potable a un total de 940.000 personas en la Ciudad de Córdoba, representando más del 90% del caudal total (el restante es aportado por el dique Los Molinos). El caudal remanente permite el riego por medio de canales. Existe una central de energía eléctrica que aprovecha el flujo de agua a través del dique, denominada central San Roque, ubicada a 11 km aguas abajo sobre el curso del río Suquía (construida en 1959), genera 26MW/h.

La red hidrográfica está conformada por los ríos Cosquín y San Antonio con módulos de aproximadamente 5,5 m3/s y 3,5 m3/s respectivamente, por un lado, y los arroyos Las Mojarras y Los Chorrillos con cerca de 0,3 m3/s cada uno (Rodríguez et al., 2000).

Imagen aérea del ESR, se aprecia el dique y el embudo.

Monitoreo de variables de calidad del agua

Desde el año 1998, el INA-CIRSA realiza el monitoreo de la calidad del agua del ESR y de sus ríos tributarios. El mismo se realiza mensualmente y se registran variables físicas, químicas y biológicas. Los objetivos de este programa de monitoreo son realizar estudios limnológicos y hacer el seguimiento del proceso de eutrofización del lago (Instituto Nacional del Agua, 2014). En las secciones siguientes se describen los detalles al respecto.

Sitios de muestreo, instrumental y técnicas de análisis

En lo que respecta al monitoreo del lago, se realiza en los cuatro puntos de desembocadura de los ríos afluentes: San Antonio (DSA), Cosquín (DCQ), Las Mojarras (DLM) y Los Chorrillos (DLC), otros dos puntos ubicados dentro del lago: el centro (C) y la presa (TAC). Cabe mencionar además que otros dos puntos se muestrearon en el pasado:

Paredón Viejo (PV) y Club Náutico (CN). En la siguiente imagen se muestran los puntos de muestreo mencionados, mientras que en la tabla muestra el detalle de las distintas ubicaciones físicas asociadas a los puntos de muestreo en el lago y sus respectivas denominaciones (a lo largo del tiempo han variado, de ahí que se tenga más de una entrada por cada caso en la tabla).

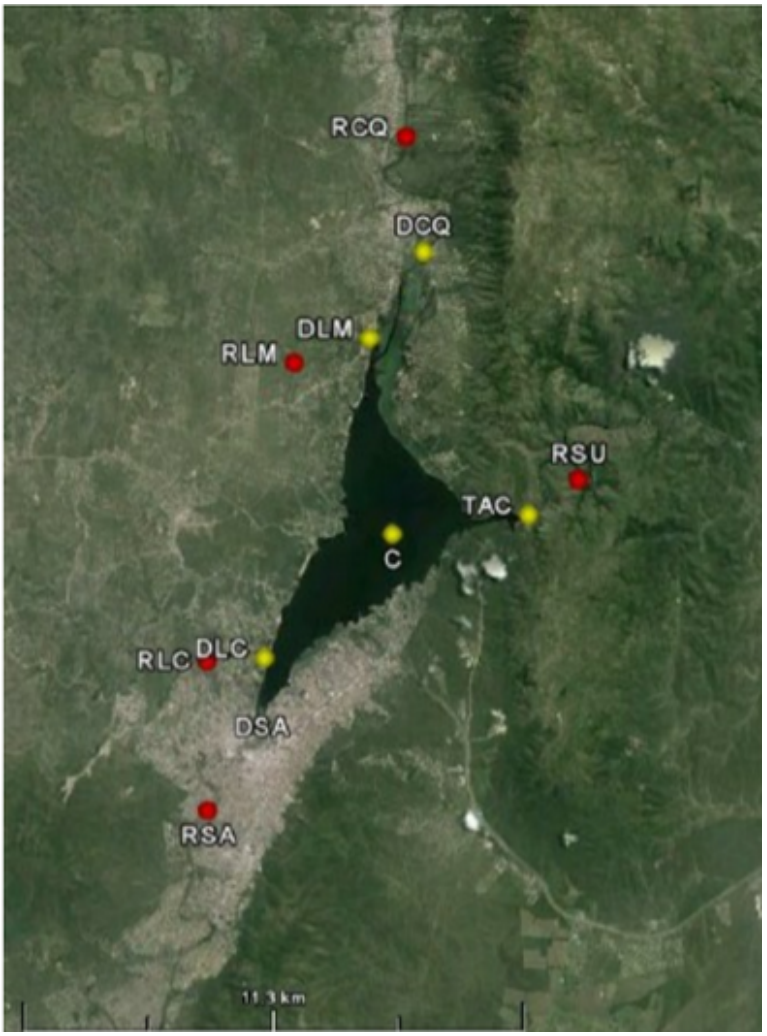


Imagen de los puntos de muestreo del ESR.

Código	Nombre	Latitud (S)	Longitud (O)
C	Centro del lago	31º 22' 36''	64º 27' 54.6''
TAC	Área de presa	31º 22' 24.27''	64º 25' 58.25''
DLM	Desembocadura Arroyo las Mojarras.	31º 20' 13.77''	64º 28' 8.38''
DCQ	Desembocadura del Río Cosquín próxima al puente	31º 19' 11.34''	64º 27' 21.88''
	Desembocadura del Río Cosquín en Plaza Federal	31º 20' 37.61''	64º 28' 9.48''
	Desembocadura del Río Cosquín conjunta al Arrollo las Mojarras	31º 20' 23.30''	64º 28' 78''
DLC	Desembocadura del Arroyo Los Chorrillos.	31º 24' 1''	64º 29' 52.34''
DSA	Desembocadura del Río San Antonio próxima al puente	31º 24' 56.06''	64º 29' 49.03''
	Desembocadura del Río San Antonio próxima al hotel Hippocampus	31º 24' 35.24''	64º 29'' 48.36''

Tabla detalle de la denominación y ubicación de los puntos de muestreo en el lago usados en la actualidad.

En lo que se refiere al monitoreo de los ríos tributarios, el mismo se realiza en puntos específicos del recorrido de cada uno, tal y como se detalla en la siguiente tabla. Existe un punto de muestreo particular, que corresponde al Río Suquía, efluente del agua del lago, que atraviesa la presa, el cual no es tenido en cuenta en el presente estudio.

Código	Nombre	Latitud (S)	Longitud (O)
RLM	Arrollo Las Mojarras a la altura de Azud	31º 20' 29.87''	64º 29'14.84''
RLC	Arrollo Los Chorrillos a la altura del balneario El Diquecito	31º 24' 5.39''	64º 30'34.89''
RSA	Rio San Antonio a la altura del balneario Fantasio	31º 25' 53.88''	64º 30' 37.26''
RCQ	Río Cosquín a la altura de Villa Caeiro	31° 17' 47.32"	64° 27' 35.58"

Tabla detalle de la denominación y ubicación de los puntos de muestreo en los ríos.

A efectos de llevar a cabo el monitoreo de calidad de agua, se cuenta con varios instrumentos. Algunas variables físicas y químicas de interés son relevadas in-situ, mientras que otras variables químicas y biológicas son determinadas en laboratorio, tanto en lo que respecta al agua del lago, como de los ríos.

En cuanto a las variables que se determinan in-situ, el procedimiento consiste en realizar mediciones en el agua por medio de una sonda multiparamétrica, que consiste en un equipo que consta de diversos sensores (parte sumergible), unido por un cable a una pantalla que permite realizar la lectura de los parámetros. La sonda se sumerge a distintas profundidades en los diferentes puntos de muestreo mencionados previamente. La variable principal en este sentido, en cuanto a que da contexto a todas las demás es la profundidad. El instrumento de medición indica la profundidad en metros (m) a la que se encuentra la sonda, calculada en función de la presión de la columna de agua.

Imagen de una sonda multiparamétrica.

Al mismo tiempo se mide también la transparencia del agua en el lago, utilizando un disco de Secchi. La transparencia, es una característica del agua que varía con los efectos del color y la turbiedad. Su medición es de forma muy simple, con un disco de 25 cm de diámetro, con fondo blanco y pintado de modo tal que un cuadrante blanco alterne con uno negro (disco de Secchi).

El mismo cuenta con una cuerda graduada en cm con el disco ubicado en cero. El procedimiento consiste en introducir el disco desde la superficie y observar a la profundidad a la cual deja de verse. La medición se realiza en un área de la superficie del agua a la sombra.

Por último, se releva también la cota del lago, a partir de una escala en metros que se encuentra sujeta a una de las paredes de la presa.

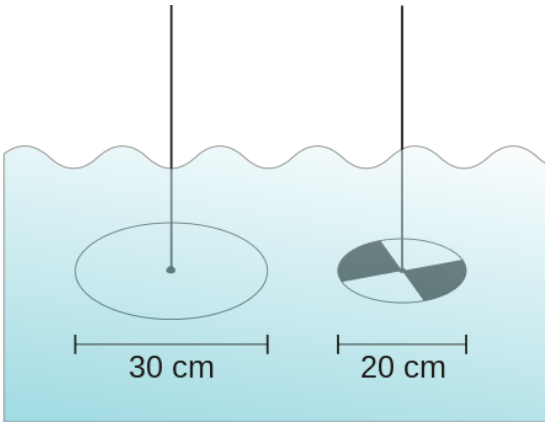


Ilustración de un disco de Secchi.

Por otra parte, el equipo que realiza la campaña de monitoreo, toma muestras de agua en los distintos puntos de muestreo mencionados y a profundidades determinadas, para ser luego analizadas en el laboratorio. A tal efecto se utiliza un muestreador vertical u horizontal de tipo Niskin, que permite, mediante el sistema de un mensajero, tomar la muestra de agua a las distintas profundidades de muestreo cuando se encuentra sumergido.

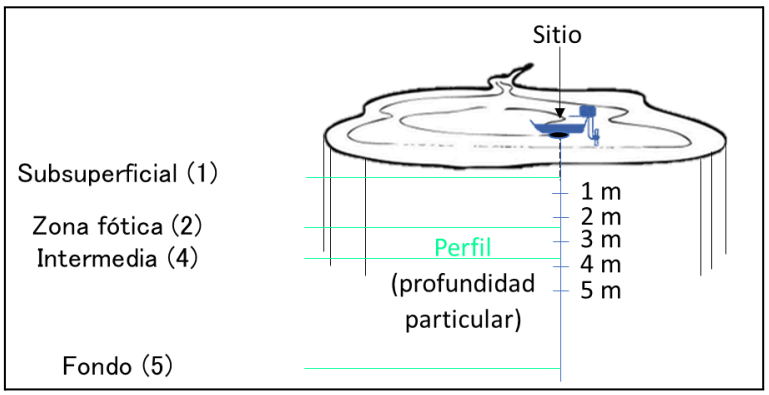
Imagen de un muestreador de tipo Niskin.

Datos de calidad de agua del lago

Los datos de calidad de agua que se relevan a partir del programa de monitoreo en el ESR son, en parte, registrados de manera manual en una planilla papel que posteriormente se transcribe en formato digital y se revisa (datos relevados in-situ) y en parte informados por el laboratorio que realiza las determinaciones restantes (datos de laboratorio), en forma digital.

Posteriormente, se conforma una nueva versión de una planilla MS Excel®, actualizada con los nuevos datos revisados. Esto se realiza una vez por año y está a cargo del equipo de calidad de agua de INA-CIRSA en la sede de Córdoba capital.

Los datos del lago se estructuran a partir de los siguientes variables de referencia: número de monitoreo (secuencial), registro (identificador del registro, secuencial), muestra (identificador que asigna el laboratorio a cada muestra que se analiza), fecha, sitio, código (es un indicador del perfil de agua que se detalla enseguida) y Z (profundidad de la medición). Por cada combinación de estos valores, se tienen una serie de variables medidas. La profundidad, se mide en general metro a metro, y 1m por encima del fondo. El siguiente diagrama ilustra el alcance del muestreo, en términos de los principales conceptos involucrados.



Perfiles. Medición datos calidad de agua a distintas profundidades del ESR.

Las variables se pueden clasificar según su alcance sea: por profundidad, donde se tiene una medición por cada fecha, sitio, código y Z (nivel de profundidad); por perfil, donde se tiene una única medición por fecha, sitio y código; por sitio, donde la medición es única por fecha y sitio; o general, donde se tiene una única medición por fecha. A su vez, las mediciones de ciertas variables, de acuerdo al método de medición pueden presentar datos censurados si se alcanza el límite de sensibilidad. Tal es el caso de las variables químicas que se miden en laboratorio (donde el instrumento no permite determinar concentraciones menores a un cierto límite, hay una pérdida por defecto, y se presume menor) y la variable de transparencia o Secchi (donde el instrumento no permite determinar transparencia a mayor profundidad de un cierto límite, hay una pérdida por exceso, se presume mayor). Por

último, vale mencionar que los monitoreos que contemplan la medición de todas las variables mencionadas con anterioridad se han efectuado desde septiembre de 1999 (anteriormente algunas de las variables de laboratorio aún no se medían), empleando distintas sondas y laboratorios (en ocasiones de manera alternada) según estuvieron disponibles.

En relación al perfil de agua, en que se miden las variables de laboratorio, se trata de determinadas profundidades (en que también están medidas las variables in-situ) que se seleccionan siguiendo un criterio conjunto de luz y temperatura, determinado según la época (remitirse a la definición de condición de estratificación). Así en condición de estratificación, se tienen los siguientes: subsuperficial, fótica, epilimnio, hipolimnio y fondo; mientras que, en condición de mezcla, se tienen: subsuperficial, fótica, intermedia y fondo. La siguiente tabla muestra el detalle de los perfiles de agua. Los códigos de perfil marcados con gris son por la escasa cantidad de datos existentes.

Sitio	Código del perfil	Mezcla		Estratificación	
		Criterio	Descripción del perfil	Criterio	Descripción del perfil
TAC	TAC1	0,20m	**Subsuperficial**	0,20m	**Subsuperficial**
	TAC2	2,5m*Secchi(m) (zona fótica)	**Fótica**	2,5m* Secchi(m) (zona fótica)	**Fótica**
	TAC3	NA		1m por encima de termoclina	Epilimnio
	TAC4	Altura de toma conductos turbina	**Toma**	Altura de toma conductos turbina	**Toma**
	TAC5	1m del fondo	**Fondo**	1 metro del fondo	**Fondo**
C	C1	0,20m	**Subsuperficial**	0,20m	**Subsuperficial**
	C2	2,5m* Secchi(m) (zona fótica)	**Fótica**	2,5m* Secchi(m) (zona fótica)	**Fótica**
	C3	NA		1m por encima de termoclina	Epilimnio
	C4	Entre 2 y 4m	**Intermedia**	1m por debajo de termoclina	**Hipolimnio**
	C5	1m del fondo	**Fondo**	1m del fondo	**Fondo**
DSA	DSA1	0,20m	**Subsuperficial**	0,20m	**Subsuperficial**
	DSA5	1m del fondo	Fondo	1m del fondo	Fondo
DCQ	DCQ1	0,20m	**Subsuperficial**	0,20m	**Subsuperficial**
	DCQ5	1m del fondo	Fondo	1m del fondo	Fondo
DLM	DLM1	0,20m	**Subsuperficial**	0,20m	**Subsuperficial**
DLC	DLC1	0,20m	**Subsuperficial**	0,20m	**Subsuperficial**

Detalle de los perfiles de agua.

De manera análoga al registro de los datos de calidad de agua del lago, se produce el de datos de ríos, obtenidos de los distintos puntos de muestreo.

Los datos de ríos se estructuran a partir de las siguientes variables de referencia: muestra (identificador que asigna el laboratorio a cada muestra que se analiza), fecha, código (se corresponde a uno de los códigos de los puntos de muestreo). Por cada combinación de estos valores, se tienen una serie de variables medidas. En este caso, a diferencia de los datos del lago, las mediciones se realizan a un solo nivel de profundidad.

Monitoreo de variables hidrológicas y meteorológicas

En esta sección se describen las características del procedimiento de relevamiento de datos hidrológicos y meteorológicos.

Estaciones meteorológicas y sensores

Los datos de las variables hidrológicas y meteorológicas son obtenidos a través del Sistema Telemétrico de transmisión de datos hidrológicos y meteorológicos a tiempo real (STC) del Centro de la Región Semiárida (CIRSA) perteneciente al Instituto Nacional del Agua (INA). El mismo se encuentra funcionando desde 1984. Este sistema se encuentra operado desde la estación central de recepción ubicada en la sede Villa Carlos Paz del INA-CIRSA (Colladón & Vélez) y consta de una red de estaciones automáticas que están distribuidas a lo largo de las subcuencas que conforman la cuenca del ESR. Entre las subcuencas más importantes se encuentran la del río San Antonio y la del río Cosquín, principales ríos afluentes. Además, existen cuencas menores como la cuenca de los arroyos Las Mojarras y Los Chorrillos, y una estación ubicada en la cuenca del río Suquía, único efluente del embalse.

Se consideraron un total de 25 estaciones ubicadas en las subcuencas de los ríos afluentes, más 3 estaciones especiales, localizadas: en la sede Villa Carlos Paz del INA-CIRSA, en la zona del perilago, y en la cuenca del río Suquía. Cada estación consta de un nombre y se encuentra identificada por un código numérico. La siguiente tabla indica el código y nombre de la estación, cuenca a la que pertenece, coordenadas y altitud sobre el nivel del mar.

Cuenca	id	Nombre	latitud	longitud	altitud
San Antonio	100	El Galpón	-31.5083	-64.82	2380
	200	Puesto Pereyra (Los Gigantes)	-31.4489	-64.8089	2292
	300	Las Ensenadas (El Cóndor)	-31.6011	-64.7911	2286
	400	El Cajón	-31.4428	-64.6925	1280
	500	Cañada Larga (Copina)	-31.575	-64.7117	1740
	600	Puesto Garay	-31.4153	-64.7353	1625
	700	La Casita	-31.4667	-64.7417	1555
	900	Confluencia Malambo	-31.4986	-64.6817	1340
	1000	Observatorio Bosque Alegre	-31.495	-64.6475	2180
	1100	San Bernardo	-31.5633	-64.7131	965
	1200	La Quebrada	-31.4883	-64.5997	918
	1400	Confluencia El Cajón	-31.4897	-64.6389	760
	1600	Bo. El Canal	-31.4467	-64.5144	675
	1700	CIRSA - Villa Carlos Paz	-31.3992	-64.4742	660
	1800	Santo Tomás	-31.5653	-64.8264	2250
Cosquin	2100	La Hoyada	-31.3572	-64.6942	1393
	2200	Pampa de Olaen	-31.3575	-64.6108	1256
	2300	Villa Giridino	-31.05	-64.5042	1045
	2400	San José de los Ríos	-31.2483	-64.6042	967
	2700	Las Junturas Cosquin	-31.2086	-64.4919	716
	2800	Villa Caeiro	-31.2914	-64.4603	681
Anisacate	2900	Camino El Cuadrado	-31.0978	-64.4469	1378
	3050	Pan de Azúcar	-31.2333	-64.4833	1100
	3400	La Suela	-31.4667	-64.5817	536
Suquia	3900	El Diquecito	-31.3536	-64.3933	536
Perilago	4050	Rep. Cerro Minero San Roque	-31.3297	-64.425	1100
Rio de Los Sauces/Traslasieras	4400	La Posta	0	0	0
Cordoba	4800	INTA Manfredi	0	0	0

Detalle de las estaciones existentes hasta el momento del trabajo (2023).

Las estaciones se encuentran dotadas de instrumentos (sensores) que permiten realizar la medición de las principales variables meteorológicas y el nivel de los ríos. Cada puesto transmite en tiempo real, por banda radial Very High Frequency (VHF), en forma directa o a través de repetidoras, a la central de recepción y procesamiento ubicada en Villa Carlos Paz.

A continuación, se detallan los sensores instalados y su tipo:

Temperatura: el sensor mide la temperatura del aire en grados centígrados (°C). En particular, el tipo de instrumento se denomina termistor, es un termómetro eléctrico.

Humedad Relativa: el sensor mide la humedad relativa del aire (representa una proporción, por ende, no tiene unidades), por medio de un dispositivo de tecnología propietaria denominado HUMICAP®(Vaisala, 2012).

Precipitación: el instrumento registra la lluvia caída en milímetros(mm). Para medir esta variable se utiliza un pluviómetro del tipo doble cubeta basculante. El dispositivo acumula el valor precipitado en un contador totalizador (con precisión de 1mm).

Velocidad del viento: el dispositivo mide la velocidad en kilómetros por hora (km/h), y se denomina anemómetro con tazas. El dispositivo acumula la distancia recorrida por los giros, y al completar los 5 km (step) emite una señal, que, junto con el diferencial de tiempo transcurrido desde la última señal, permite calcular la velocidad.

Dirección del viento: en este caso se utiliza una veleta, que apunta en la dirección de destino del viento, entre 0° y 360°. Como este instrumento funciona en conjunto con el anterior, cada vez que se emite una señal para registrar la velocidad, a la vez se indica la dirección del viento en el momento del envío de la señal.

Presión atmosférica: el instrumento utilizado registra la presión en hectopascales(hPa), y se denomina barómetro del tipo aneroide.

Radiación solar: este sensor registra la radiación medida en langleys (Ly) se denomina piranómetro y es del tipo bimetálico fotovoltaico.

Evaporación: mide la cantidad de agua en milímetros(mm) que se transforma en vapor.

Nivel: se mide en ríos. Este tipo de instrumento se denomina hidrómetro, y es capaz de registrar el nivel medido en metros. En cada estación el sensor tiene un valor de altura que se denomina nivel de referencia del hidrómetro, que es el menor valor que es capaz de registrar. Este valor, es distinto del cero, que se refiere al punto físico en que se encuentra ubicado. Los sensores registran valores dentro de un rango predeterminado, pudiéndose establecer la sensibilidad del mismo para detectar la variación en el nivel del río (step) (Colladón, 2018). Análogamente, se tiene un sensor que mide el nivel del lago.

Datos meteorológicos e hidrológicos relevados por sensores

Los datos relevados por los sensores ubicados en las estaciones automáticas de monitoreo, una vez transmitidos a la sede central de procesamiento son almacenados en los servidores en base PostgreSQL y en archivos de texto plano, mediante la intervención de software. Posteriormente, los archivos son revisados y comentados por un experto para

garantizar la calidad de los datos. Se genera un archivo por año, identificado por el año y el código del sensor. La estructura del archivo, consta de: encabezado con el código del sensor, nombre de la estación y tipo de sensor (variable que mide); y datos organizados en columna, donde por cada medición se registra fecha, hora y valor. En el caso particular del viento, en un mismo archivo se registra velocidad y dirección del viento (se tienen dos columnas de valor).

```
Sensor #  302  La Casita T. Temperature Sensor
  DATE      TIME    degrees C
12/31/2009 23:30:29   12.8
12/31/2009 22:28:35   14.5
12/31/2009 21:26:40   14.5
12/31/2009 20:24:46   14.5
12/31/2009 19:22:52   15.1
12/31/2009 18:20:58   15.6
```

Extracto de un archivo típico que registra información de un sensor(temperatura en este caso).

Dadas las características geomorfológicas de cada una de las cuencas, y teniendo en cuenta las recomendaciones de la Organización Meteorológica Mundial, las estaciones automáticas se hallan estratégicamente ubicadas para censar la realidad que las rodea. La mayor parte de las mismas está dotada de sensores de precipitación y nivel de los ríos, puesto que el fenómeno de crecidas producto de precipitación intensa es quizás el fenómeno más importante que se busca monitorear en términos de la seguridad de los habitantes de estas regiones. En cuanto a los demás tipos de sensores, los más comunes son los correspondientes a temperatura y humedad relativa en la cuenca del río San Antonio; el resto se ubican en muy pocas estaciones, ya que son de más largo alcance.

Debido a diferentes factores, entre ellos la gradual instalación de las estaciones, dificultades técnicas en los procesos de medición y transmisión, rotura del instrumental o imposibilidad de mantenimiento o calibración, los datos no han estado siempre disponibles durante el período completo de medición, pudiendo faltar un año completo, meses de determinados años, fechas específicas o determinadas horas para una misma fecha.

En lo que respecta a la frecuencia de medición, varía entre estaciones, sensores y fecha porque depende de la calibración del instrumento, pero considerando las medianas por estación, podría decirse en general, que es de entre 18 y 62 minutos para todas las variables que se consideran de medición programada: temperatura, nivel, presión atmosférica, radiación velocidad y dirección del viento. Este término hace referencia a que cada registro es el resultado de una medición instantánea que se realiza a plazos de tiempo preestablecidos, y dicho registro es suficiente para determinar el valor de la variable en el momento en que ocurre la medición.

En el caso de la precipitación, la medición es no programada, ya que se realiza cuando se produce 1 milímetro de precipitación y la naturaleza del registro es acumulativa. Esto quiere decir, que la determinación del valor de la variable es diferencial, así, por ejemplo, a efectos de establecer la lluvia precipitada en un día, debe calcularse la diferencia entre el primer y el último registro para esa fecha. El dispositivo de medición en este caso emite una señal con el valor actual del acumulador dos veces al día (cada doce horas) en caso que no ocurra precipitación, como señal de latido, indicando que se encuentra operativo.

Otro aspecto importante respecto de la lógica de funcionamiento del sensor de precipitación, que afecta la interpretación de los datos, es que al producirse el volcado de uno de los recipientes por alcanzar el peso suficiente, el otro recipiente entra en acción generando un reinicio del acumulador a cero. Esto significa, que, a partir de ese momento, el primer registro es cero, y el registro siguiente marca uno, para indicar el primer milímetro medido desde el volcado.

Preprocesamiento de datos

Se describe la teoría de la etapa de preprocesamiento consistente en la depuración, cálculo y estructuración de los datos crudos tendiente a generar el conjunto de datos apto para la aplicación de técnicas y modelos de Minería de Datos. Este preprocesamiento queda funcional para los datos que entran en tiempo real.

Datos meteorológicos e hidrológicos

En lo que respecta al preprocesamiento de los datos meteorológicos e hidrológicos, los objetivos son depurar los errores propios del funcionamiento de los sensores y/o la transmisión de los datos (en adelante denominados inconsistencias).

Como primer paso, los datos deben almacenarse en una base de datos con la estructura deseada, por lo tanto estructurar los datos de los distintos sensores implica un primer procesamiento.

Luego se hacen los procesamientos para el tratamiento de inconsistencias:

Tratamiento de inconsistencias

Esta etapa consiste en identificar diferentes tipos de anomalías en el registro de los datos. Es una etapa fundamental para poder realizar luego las fases de análisis y predicción, ya que cualquier irregularidad o error en los datos puede eventualmente distorsionar las métricas de cálculo que se aplicarán posteriormente.

Detección de perturbaciones:

Es preciso identificar irregularidades que pudieran eventualmente distorsionar las métricas de cálculo; las alteraciones que se producen en el desarrollo normal de un proceso en adelante son denominadas perturbaciones. Las mismas pueden agruparse en dos clases:

- Perturbaciones en sentido estricto: entendidas como uno o más valores consecutivos

que en el contexto de la serie temporal resultan atípicos y deben ignorarse para evitar sesgar los promedios. Estos valores, una vez identificados se marcan en el campo de observaciones de la tabla, para permitir su filtrado con posterioridad. Las hay de tres tipos:

i. Perturbaciones puntuales: las que implican un incremento o un descenso de un único valor en el contexto de la serie como lo ilustra el siguiente gráfico:

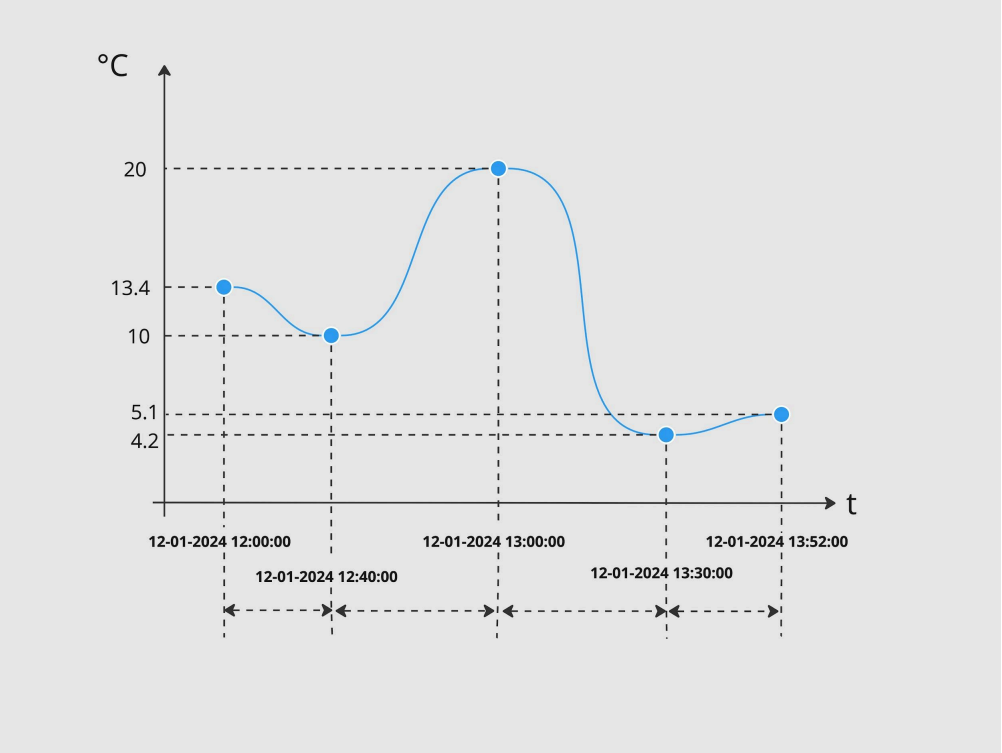


Gráfico ilustrativo de cómo luce una perturbación puntual.

Este es el caso más frecuente, y para su corrección se considera el cumplimiento de tres condiciones, a saber:

a. Perturbación: Sean tres registros A, B y C, tal que A se produce primero, B después que A y C después que B, tal que son consecutivos en ese orden; entonces la medición A es menor que B y C es menor que B (B es un pico) o A es mayor que B y C es mayor que B (B es una depresión). Por ejemplo, considerando la variable temperatura, la siguiente secuencia representa la primera condición (pico): A=10°C, B=20°C, C=11°C; mientras que la siguiente secuencia cumpliría con la segunda (depresión): A=30°C, B=24°C, C=29°C.

b. Límite temporal: Responde a la pregunta: ¿hasta cuándo se considera una perturbación? Vale decir la tolerancia en términos de tiempo transcurrido entre dos registros consecutivos, uno de los cuales es candidato a ser perturbación puntual. En este sentido la estrategia utilizada consiste en calcular el valor absoluto de la diferencia de tiempo entre dos registros consecutivos, y posteriormente un determinado percentil de la distribución de las diferencias de tiempo según sus características, sin por ello seleccionar valores demasiado altos que pudieran hacer ver como perturbaciones fenómenos de más largo alcance por falta de datos (crecidas de los ríos, descenso brusco de temperatura por avance de un frente frío, entre otros).

El siguiente gráfico ejemplifica una perturbación puntual dentro de la serie temporal para la variable de temperatura. Según los criterios definidos, se admite una variabilidad tolerable en la temperatura de hasta 4.4°C. Además, se observa que las mediciones se toman a intervalos regulares, con un límite máximo de 62 minutos entre observaciones consecutivas. Este valor atípico, que excede la variación tolerada, se destaca en el gráfico para ilustrar su impacto en el análisis y para demostrar la necesidad de un procesamiento cuidadoso de los datos a fin de mantener la integridad de la serie temporal.

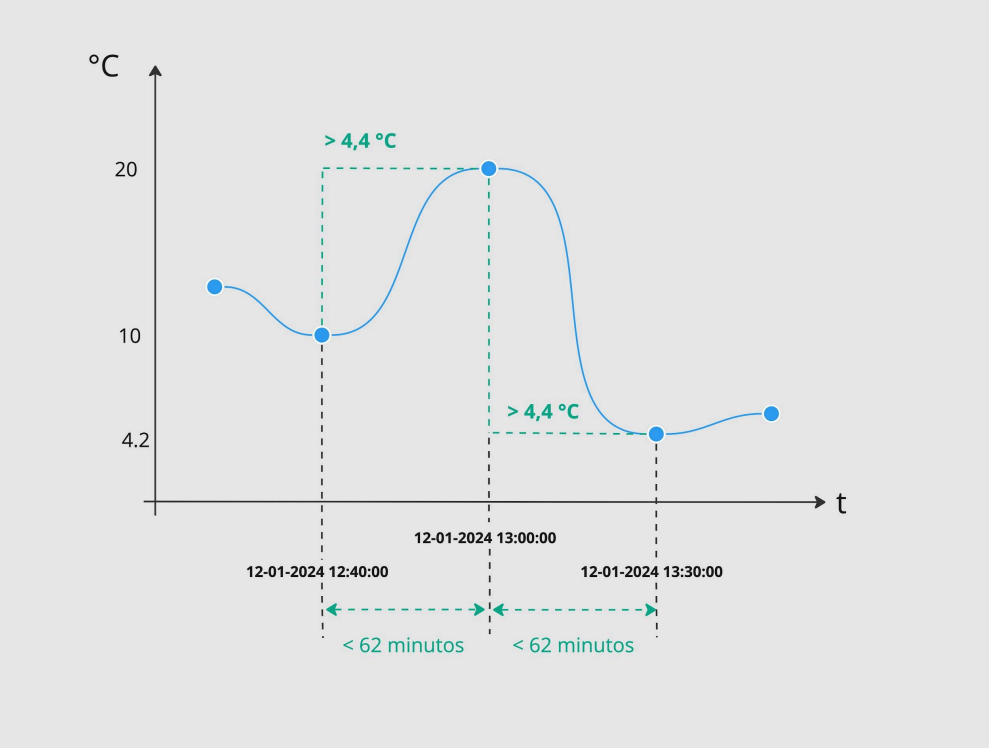


Gráfico ilustrativo donde se cumplen con las condiciones de tiempo y variación para cumplir con la definición de perturbación puntual.

En la siguiente serie temporal, se observan casos en los que las fluctuaciones de temperatura no cumplen con los criterios previamente establecidos para ser consideradas como perturbaciones significativas. Específicamente, aunque se identifican variaciones en la temperatura, estas no se sostienen por el intervalo de tiempo mínimo de 62 minutos entre mediciones consecutivas, que es esencial para validar una perturbación según nuestro protocolo. En el gráfico adjunto, se muestra un pico de temperatura que inicialmente podría interpretarse como una perturbación puntual. Sin embargo, al no persistir más allá del límite temporal estipulado, esta variación se clasifica como irrelevante para el análisis y, por tanto, se excluye del modelo predictivo. Este enfoque asegura que solo las perturbaciones que reflejan cambios genuinos y sostenidos sean consideradas, manteniendo así la precisión y fiabilidad de nuestras evaluaciones hidrometeorológicas.

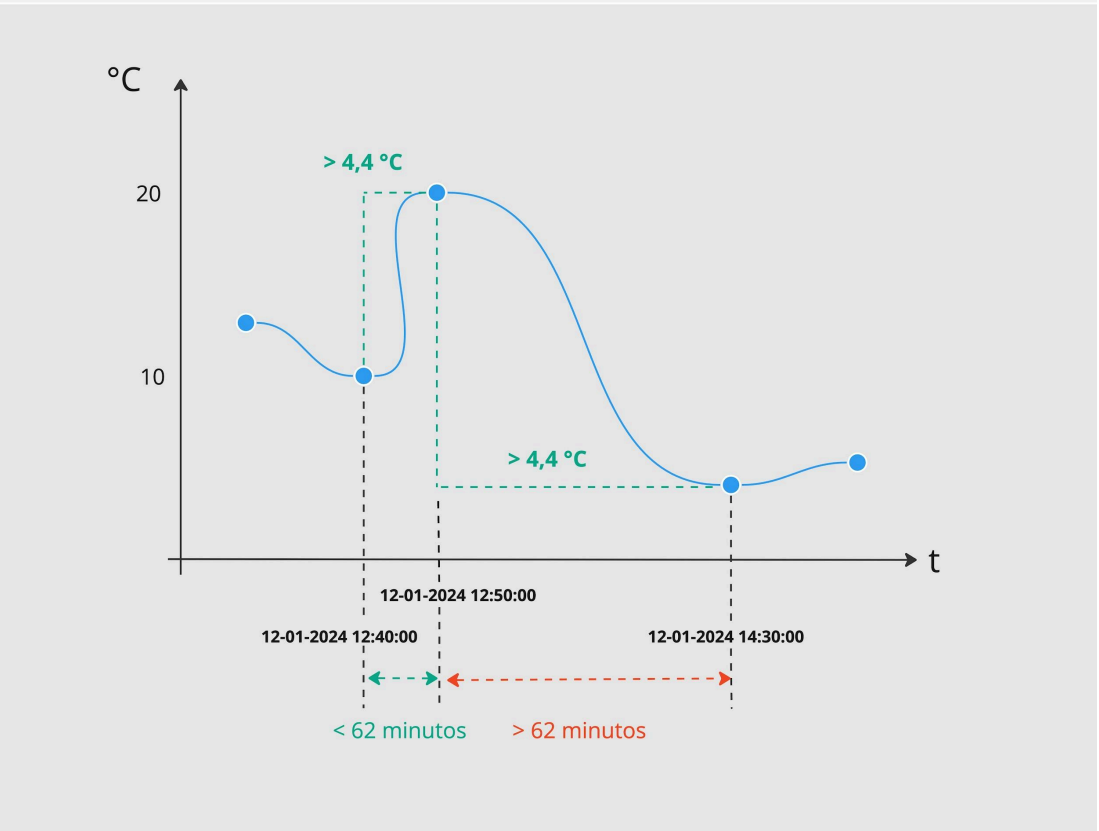


Gráfico ilustrativo donde **no** se cumplen con las condiciones de tiempo, quedando así descartada la perturbación puntual.

Un caso especial en cuanto al límite temporal constituye la variable nivel, donde, cuando se mide para el lago, las estaciones 2003 y 2004 lo hacen de manera instantánea, mientras que la estación 2020 lo hace de manera diaria. Dicho esto, es preciso establecer límites temporales por separado: para las dos estaciones que miden nivel varias veces al día y para la estación restante y los datos de recolección manual donde se tiene la información una vez al día. En la siguiente tabla se pueden observar dos filas para el nivel del lago, una para cada caso de los expuestos.

c. Límite de medición: Responde a la pregunta: ¿hasta cuánta variación comprenderá una perturbación? Se refiere a la tolerancia en términos de la diferencia en la magnitud de dos mediciones consecutivas, una de ellas candidata a ser considerada perturbación puntal. En este caso se recurre a una estrategia similar a la del punto anterior, calculando el valor absoluto de las variaciones de magnitud entre registros consecutivos, y posteriormente estableciendo un umbral superior (tolerancia) por medio de la definición de valor extremo severo: 3 veces la diferencia intercuartílica más el cuartil 75. Un caso particular constituye la dirección del viento, que sigue una distribución circular (0° es igual a 360°), por ende, se calculan cuartiles circulares.

Otro caso especial resulta la variable nivel que, al medirse para ríos y lagos, la dinámica y características propias de cada uno hace que las distribuciones sean completamente distintas y por ende deban realizarse por separado las determinaciones: por un lado, estaciones que miden nivel de ríos y por otro las que hacen lo propio con el nivel del lago. Con respecto al caso del río, para la diferencia en tiempo establecida en el punto anterior, la variación en magnitud es 0, lo cual da indicio de no existencia de perturbaciones puntuales en este subgrupo.

ii. Perturbaciones no puntuales o extendidas: las que implican un alza o una baja de al menos dos valores, en el contexto de la secuencia. A tal efecto se identificarán como perturbaciones no puntuales los casos que cumplan las siguientes condiciones:

a. Perturbación: Sean cuatro registros: A, B, C y D, donde A se produce primero, B después que A, C después que B y D después que C. A es el primer registro anterior a B sin perturbación, mientras que D es el registro inmediato siguiente a C sin perturbación. Entre B y C puede haber cero o más registros, y estos dos son los que marcarán el comienzo y el final de la secuencia de valores atípicos, respectivamente. Por ejemplo, considerando la variable temperatura, la siguiente secuencia cumpliría con la condición: A=10°C, B=20°C, C=20°C, D=11°C (se trata de un pico extendido).

b. Límite temporal: Entre cada uno de los registros consecutivos que intervienen debe haber una diferencia no mayor en el tiempo que corresponde, siguiendo la estrategia indicada en el punto anterior, al percentil 75 como valor límite. Esto apunta a reducir aún más las posibilidades de falsos positivos en términos de perturbaciones.

c. Límite de medición: Entre los registros externos (A y D) y los registros límite internos (B y C) debe haber una diferencia de magnitud mayor a la identificada en i.3 (marcando el inicio y el final de la secuencia). En cambio, entre los registros límite internos la diferencia debe ser menor a ese mismo valor (se espera que la perturbación se mantenga en el tiempo).

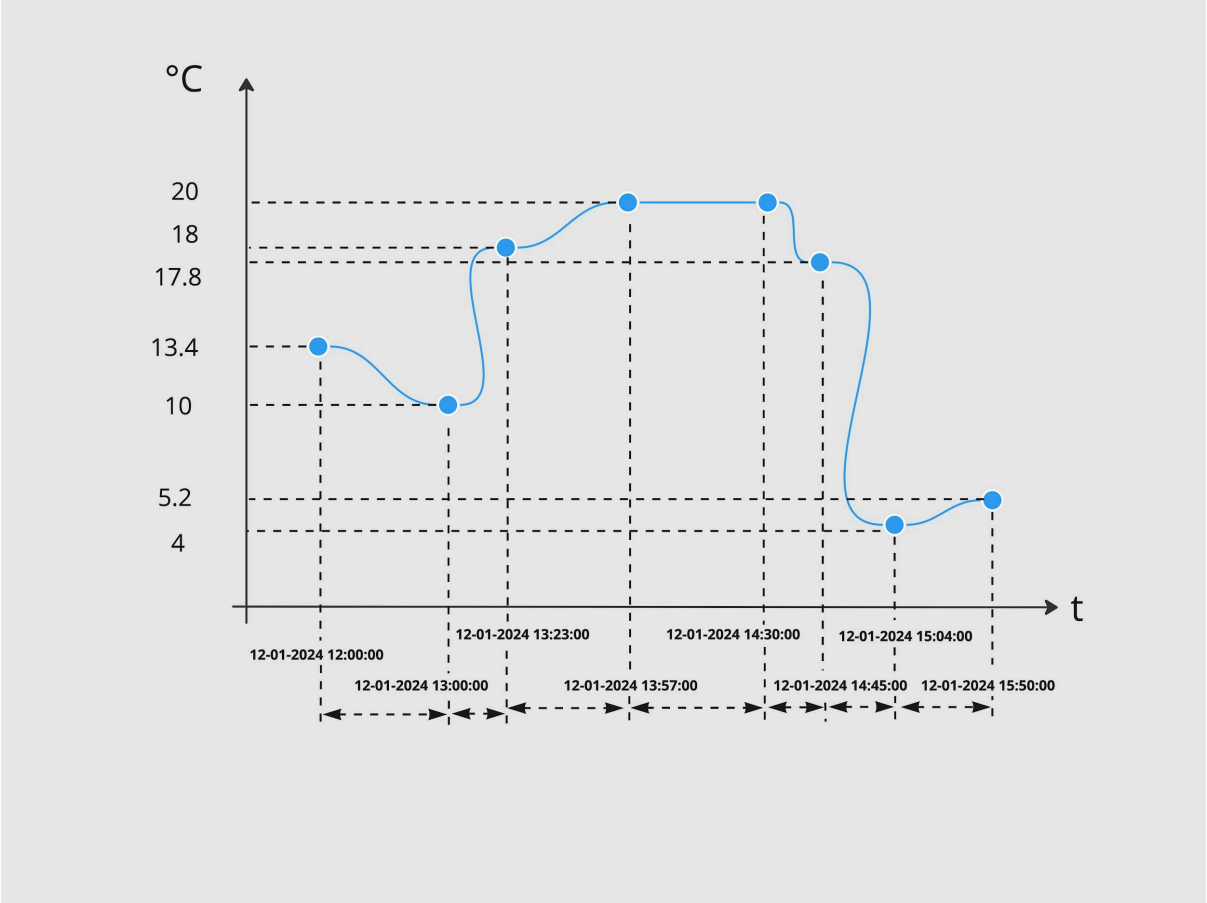


Gráfico ilustrativo de cómo luce una perturbación extendida.

A continuación, se ilustra el análisis de cómo se interpreta el siguiente hecho para la variable temperatura a la cual, teniendo en cuenta los mismos criterios que en las perturbaciones puntuales se le añade un nuevo criterio temporal, el cual define que para que no se descarte una posible perturbación extendida el intervalo de tiempo entre medidas consecutivas debe ser menor o igual a los 33 minutos, ilustrado luce de la siguiente manera:

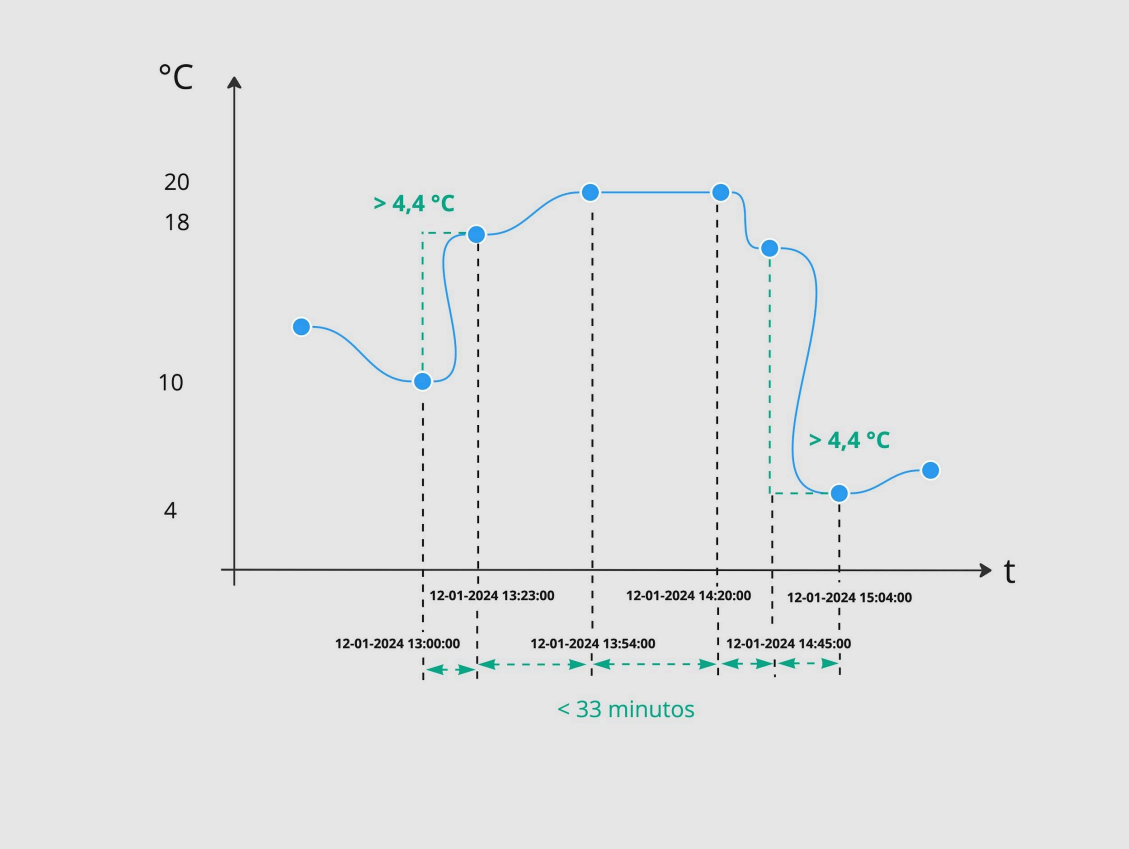


Gráfico ilustrativo donde se cumple con todos los criterios temporales y de variación en la medición.

Quedando demostrado el caso donde se confirma una perturbación extendida, a continuación se ilustra cómo sería en el caso de que sea una posible perturbación extendida que fue descartada:

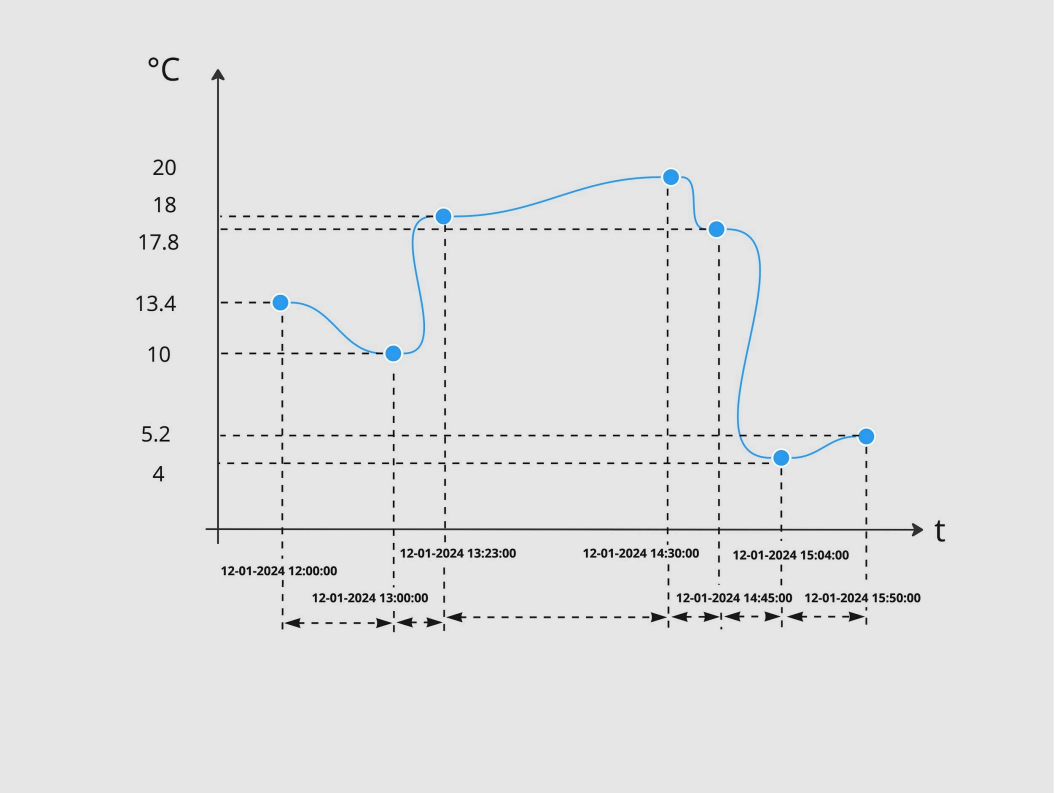


Gráfico ilustrativo de cómo luce una perturbación extendida descartada.

Este descarte surge de un análisis de ambos criterios tanto temporales como de variación del valor, el cual se puede interpretar más fácilmente con el siguiente gráfico:

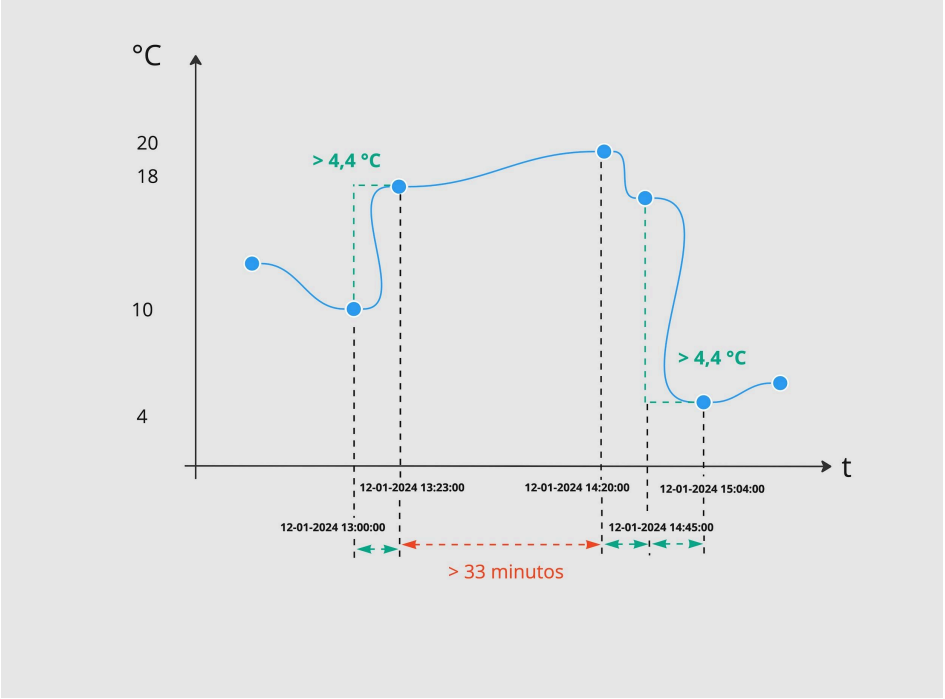


Gráfico ilustrativo donde **no** se cumplen con las condiciones de tiempo, quedando así descartada la perturbación extendida.

iii. Perturbaciones excepcionales o valores extremos: constituyen mediciones minoritarias extremadamente altas o bajas en el contexto de todas las series, que escapan a las definiciones en i y ii (alguna de las condiciones antes explicitadas no se cumple), pero que sin embargo no hacen sentido en términos de la naturaleza de la variable o bien no se tiene evidencia histórica disponible de que tales valores se hubieran producido alguna vez. A efectos de realizar esta corrección, se considera la definición de valor extremo severo para determinar los umbrales de tolerancia (ver tabla), utilizando diferentes percentiles según las características de la distribución.

En el caso de las variables dirección del viento y humedad relativa, que se encuentran acotadas a un rango, se refiere a los valores fuera de dicho rango. En particular, la humedad relativa tiene un límite inferior de 0 y un límite superior que se definió en 1.01 (101%) para preservar la condición excepcional de sobresaturación que podría presentarse.

Un caso particular constituye en este sentido el nivel de los ríos, donde el mínimo nivel viene dado por la altura de instalación del sensor (variable de acuerdo a la estación y algunos casos el período de tiempo) determinando el menor valor que es capaz de registrar. En este caso, se siguió la documentación técnica referida al respecto para filtrar valores por debajo del límite. Para mayor detalle al respecto referirse a Colladón (2018).

La siguiente tabla define cuáles serán los valores a considerar para hacer el tratamiento de inconsistencias y determinar si hay o no y el tipo de perturbación.

Variable	Perturbación puntual			Perturbación continua	Valores extremos		
	Ventana de tiempo (minutos)	*Criterio (percentil)*	*Variación tolerada*	*Ventana de tiempo (minutos)*	*Límite inferior*	*Límite superior*	*Criterio (percentiles)*
Temperatura	62	90	4,4ºC	33	-19,9ºC	47,3ºC	25-75
Nivel de lago	182	90	0,04m	NA	(*)	55,16m	5-95
Nivel de lago (diario)	1440	90	0,13m	NA	(*)	55,16m	5-95
Nivel de rio	22	90	0m	NA	(*)	5m	5-95
Humedad relativa	62	90	0,148	62	0	1,01	NA
Velocidad del viento	73	90	10,4km/h	NA	0km/h(**)	147,70 km/h	10-90
Dirección del viento	37	75	0,173682º	NA	0º	360º	NA
Radiación solar	62	90	0,448ly	NA	0ly	2,43ly	25-75
Presión atmosférica	62	90	1,6hPa	62	669.6hPa	1092,40hPa	25-75

Nota: (*) El límite inferior o mínimo de nivel depende del punto de muestreo. NA: No Aplica.
(**) Cabe aclarar que, en el caso de la velocidad del viento, los valores con velocidad 0 corresponden a los casos en que justo se completó el step de 5km y se decide considerarlos para el cálculo (ver 2.3.1).

Tabla detección de perturbaciones.

Valores duplicados:

Se trata de dos registros que, o bien tienen la misma fecha y hora, o bien tienen la misma fecha y una diferencia muy pequeña en la hora, y no constituyen perturbaciones. Se establece un orden de frecuencia, donde esa diferencia pequeña entre las horas puede ser de hasta 5 segundos siendo este el tiempo máximo menor a un minuto (entendiendo que el sensor pasado este umbral es capaz de emitir una nueva señal que no constituirá un duplicado sino una nueva medición, con mayor frecuencia de lo normal).

De esta manera, los registros que cumplan dicha condición y además el valor de la medición sea el mismo, entonces arbitrariamente todos excepto uno, no se los almacenará en la base. En el caso en que la medición no coincida (caso minoritario), todos los registros intervienen del cálculo posterior (se almacenan todos, no hay razón para invalidar alguno, se interpreta que son registros independientes, que se han generado con una frecuencia más alta por alguna razón de configuración momentánea del dispositivo de medición).

Valores nulos:

Muchas veces los sensores fallan al medir, o hay problemas en la comunicación de las señales y eso provoca que al Datawise le llegue una medición nula. Estas también se desea no almacenar en la base de datos para que no interfiera en cálculos estadísticos.

Se hace un conteo de la cantidad de valores nulos también para poder tener mayor visibilidad del funcionamiento del sensor.

A nivel Técnico de solución:

Ingeniería de Software

Es el área de estudio que se ocupa de la creación de software confiable y de calidad mediante el uso de métodos y técnicas de ingeniería. Dentro de dicha área, uno de los conceptos claves que existe es el del ciclo de vida del software, este consiste de una serie de pasos o etapas que son esenciales para garantizar que los programas desarrollados cumplan con estándares de eficiencia, confiabilidad, seguridad y satisfagan las necesidades de los usuarios finales.

Estos pasos incluyen actividades como la definición de requisitos, el diseño de la arquitectura, el desarrollo, la implementación, las pruebas, el mantenimiento, entre otros, dependiendo de la metodología y enfoque específicos.

Las etapas de dicho ciclo son:

Planificación

Antes de empezar un proyecto de desarrollo de un sistema de información, es necesario hacer ciertas tareas que influirá decisivamente en el éxito del mismo. Dichas tareas son conocidas como el *fuzzy front-end (difuso)* del proyecto, puesto que no están sujetas a plazos y la información puede ser incompleta y/o no completamente clara.

Algunas de las tareas de esta fase incluyen actividades como la determinación del ámbito del proyecto, la realización de un estudio de viabilidad, el análisis de los riesgos asociados, la estimación del coste del proyecto, su planificación temporal y la asignación de recursos a las diferentes etapas del proyecto.

Análisis

Por supuesto, hay que averiguar qué es exactamente lo que tiene que hacer el software. Por eso, la etapa de análisis en el ciclo de vida del software corresponde al proceso a través del cual se intenta descubrir qué es lo que realmente se necesita y se llega a una comprensión adecuada de los requerimientos del sistema (las características que el sistema debe poseer). Esta etapa también se la conoce como ingeniería de requerimientos.

Durante esta etapa se definen detalladamente los requisitos del software, abarcando funcionalidades, restricciones y características esenciales para satisfacer tanto a los usuarios como a los objetivos del negocio. La comunicación activa con stakeholders (interesados), que pueden incluir clientes y usuarios finales, desempeña un papel fundamental para comprender sus expectativas. Se emplean técnicas de modelado, como diagramas de casos de uso, para visualizar la funcionalidad del sistema. Los requisitos se validan y verifican para garantizar su coherencia y claridad, y cualquier conflicto o dificultad se aborda mediante procesos de negociación y priorización. La documentación detallada de los requisitos, junto con un enfoque sólido en la gestión de cambios, establece las bases necesarias para las fases subsiguientes del desarrollo del software, asegurando que lo construido se alinee de manera precisa con las expectativas y necesidades del cliente.

Diseño

En esta fase se utiliza la información recopilada para estudiar opciones e implementar la estructura del proyecto. Esto abarca la definición de la base de datos, la lógica del flujo de datos y la interfaz de usuario. Se analizan requisitos detalladamente, considerando la integración de módulos existentes y seleccionando tecnologías adecuadas. Se modela cómo funcionará el software, definiendo aspectos como la interfaz, el lenguaje de programación, la seguridad y la arquitectura. Se crean prototipos para obtener retroalimentación del cliente y realizar ajustes iterativos. La fase destaca la importancia de catálogos de patrones de diseño para evitar errores comunes y garantizar una solución

robusta. Se establece la estructura técnica del software, incorporando la retroalimentación del cliente para asegurar un producto final satisfactorio.

Implementación

Dependerá tanto de las decisiones de diseño tomadas como del entorno en el que el software deba funcionar.

Es donde propiamente se desarrolla el sistema, donde se programa. Haciendo entregas iterativas e incrementales (si es que se aplica una metodología ágil). También lleva consigo preparar el entorno donde se desarrollará, probara y funcionará el sistema. Se crea la infraestructura.

Pruebas

Una vez terminado el proceso de desarrollo empieza el testeo y la fase de pruebas de la aplicación. En esta etapa ponemos a prueba los errores que hayan podido aparecer en las etapas anteriores. Es una fase de corrección, eliminación y perfeccionamiento de posibles fallos, no previsto en los pasos previos. Se busca sobre todo que el software cumpla con las necesidades del cliente.

Una de sus formas son las pruebas unitarias. Estas validan la funcionalidad de una aplicación, desde niveles bajos hasta niveles altos. Las pruebas unitarias a menudo se integran en todo el proceso de construcción de software y se ejecutan automáticamente como parte de sus procesos de CI/CD, de lo cual hablaremos en un momento. La fase de pruebas se ejecuta con frecuencia en paralelo a la fase de desarrollo.

Despliegue

Cuando los equipos desarrollan software, lo codifican y prueban en una copia diferente que no es a la que acceden los usuarios. El software que los clientes usan se llama producción, mientras que las otras copias están en el entorno de compilación o entorno de pruebas.

Disponer de un entorno de compilación y de un entorno de producción diferenciados garantiza que los clientes puedan seguir usando el software incluso cuando se modifica o actualiza. La fase de despliegue incluye varias tareas para llevar la última copia compilada al entorno de producción, como empaquetado, configuración del entorno e instalación

Fase de Mantenimiento

En este periodo el software ya está en funcionamiento. Con el tiempo alguna función puede quedar obsoleta, pueden detectarse algunas limitaciones o que aparezcan propuestas que mejoren la estabilidad del proyecto. Generalmente se vuelve a repetir todo el ciclo, ante la aparición de nuevas propuestas.

Herramientas, técnicas y métodos a aplicar:

Extracción, transformación y carga de datos (ETL)

Un problema habitual al que se enfrentan las organizaciones es cómo recopilar datos de varios orígenes, en varios formatos. A continuación, tendrá que moverlos a uno o varios almacenes de datos. Es posible que el destino no sea el mismo tipo de almacén de datos que el origen. A menudo el formato es diferente, o bien es necesario dar forma a los datos o limpiarlos antes de cargarlos en el destino final.

Con los años se han desarrollado varias herramientas, servicios y procesos para ayudar a afrontar estos desafíos. Independientemente del proceso que se utilice, hay una necesidad común de coordinar el trabajo y aplicar cierto nivel de transformación de datos en la canalización de datos.

Proceso de extracción, transformación y carga (ETL)

Extracción, transformación y carga (ETL) es una canalización de datos que se usa para recopilar datos de varios orígenes. Luego, transforma los datos según las reglas de negocio y los carga en un almacén de datos de destino. El trabajo de transformación en ETL tiene lugar en un motor especializado y, a menudo, implica el uso de tablas de almacenamiento provisional para conservar los datos temporalmente a medida que estos se transforman y, finalmente, se cargan en su destino.

La transformación de datos que tiene lugar a menudo conlleva varias operaciones como filtrado, ordenación, agregación, combinación de datos, limpieza de datos, deduplicación y validación de datos, hasta lógicas más complejas.

Frecuentemente, las tres fases del proceso ETL se ejecutan en paralelo para ahorrar tiempo. Por ejemplo, mientras se extraen datos, puede que esté funcionando un proceso de transformación sobre los datos ya recibidos y de preparación para la carga, y puede que empiece a funcionar un proceso de carga sobre los datos preparados, en lugar de tener que esperar a que termine todo el proceso de extracción.

Python

Es un lenguaje de alto nivel de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código, multiplataforma y dinamismo. Es uno de los lenguajes de programación más populares. Python viene con una amplia biblioteca estándar que proporciona una amplia gama de módulos y funciones para realizar diversas tareas, desde manipulación de archivos hasta acceso a bases de datos y creación de interfaces gráficas de usuario.

Pandas

Pandas es una biblioteca de código abierto de Python que proporciona estructuras de datos y herramientas de análisis de datos de alto rendimiento y fácil de usar que permiten a los usuarios trabajar de manera eficiente con datos tabulares y de series temporales. Además ofrece una amplia gama de herramientas para limpiar y preparar datos.

Tkinter

Tkinter es una biblioteca estándar de Python para crear interfaces gráficas de usuario. Permite a los desarrolladores crear ventanas, widgets[4] y otras herramientas interactivas para sus programas de manera sencilla

DataWise

DataWise es un software avanzado para la adquisición y control de datos ambientales. Se especializa en recolectar, analizar y manipular datos de diversas fuentes de telemetría. Incluye capacidades de gestión de bases de datos, visualización de datos, controles en tiempo real, análisis hidrológico, pronóstico, manejo de alarmas y herramientas web.

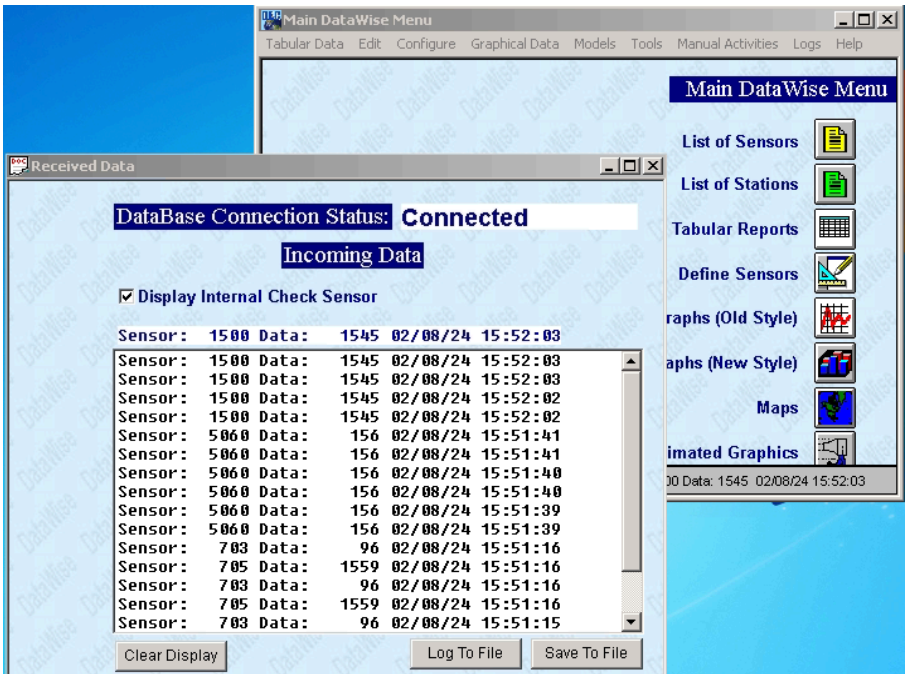


Imagen del DataWise utilizado en el INA-CIRSA

ODBC

ODBC (Open Database Connectivity) es un estándar de programación que permite a las aplicaciones acceder a datos en sistemas de gestión de bases de datos (DBMS)

[4] Widget: son componentes básicos de una interfaz gráfica de usuario y pueden incluir botones, cajas de texto, etiquetas, barras de desplazamiento, menús desplegables, casillas de verificación, botones de radio, entre otros, que permiten a los usuarios interactuar con la aplicación.

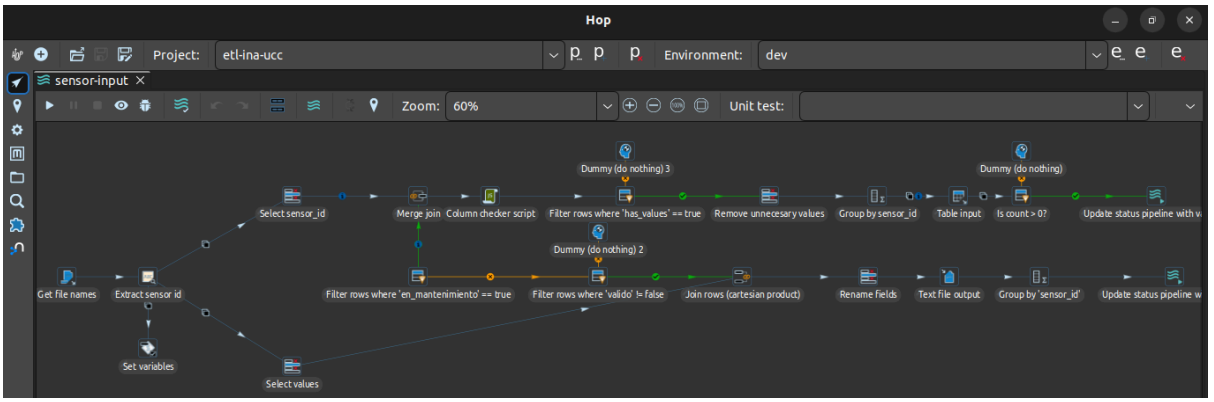
independientemente de la base de datos y del sistema operativo. Proporciona una interfaz común para que las aplicaciones puedan realizar consultas y actualizar datos en diferentes tipos de bases de datos sin necesidad de escribir código específico para cada sistema de base de datos.

Apache Hop

Es una de las herramientas centrales que usamos para el desarrollo del sistema. Contiene la lógica de extracción, transformación y guardado (ETL). Es de código abierto, gratuito. Permite el desarrollo con interfaz gráfica lo que facilita el entendimiento y el desarrollo mismo. También permite el CI-CD (lo definiremos luego). Y no tiene problemas con los tipos de entrada de datos y guardado.

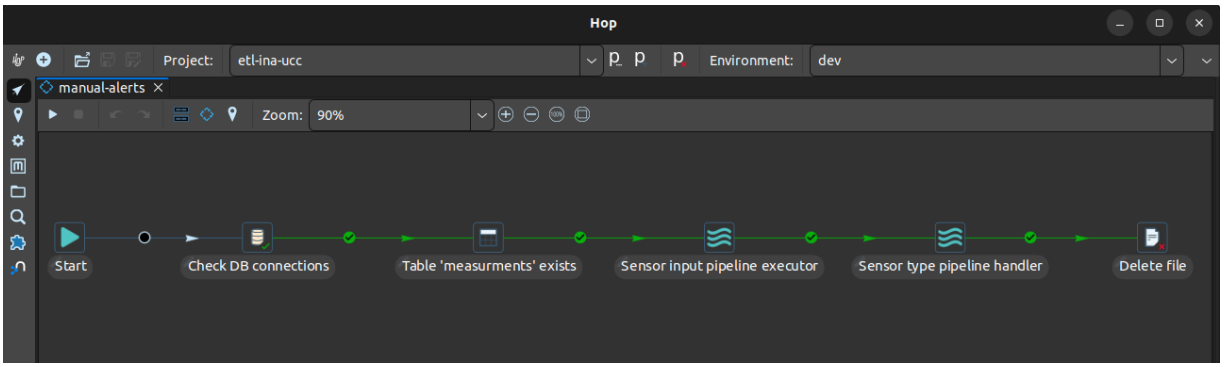
Los pipelines de Hop manipulan datos provenientes de diferentes fuentes, leyendo el input, aplicando sus transformaciones (operaciones donde se aplica una serie de reglas de negocio o funciones sobre los datos extraídos para convertirlos en datos que serán cargados) y generando finalmente el output, en distintas fuentes de carga.

- Pipeline: son colecciones de transformaciones, conectadas como tuberías por donde pasan los datos. Todas las transformaciones en un pipeline se ejecutan en paralelo.



Ejemplo de un pipeline. En este caso es el que usamos para obtener los distintos estados por los que pasa el sensor en el conjunto de datos que se ingresa, luego se invoca a otro pipeline que es el que hará la actualización en la base de datos.

- Workflow: son colecciones de acciones (entre ellas pueden ser pipelines), conectadas también como tuberías. Todas las acciones de un flujo de trabajo se ejecutan de forma secuencial. Por lo que cada acción debe esperar a la respuesta de la acción anterior para ser ejecutada.



Ejemplo de un Workflow. En este caso es el que usamos para cargar datos históricos.

Apache Kafka

Es una popular plataforma de streaming de eventos que sirve para recoger, procesar y almacenar datos de eventos de streaming o datos sin principio ni final concretos. Entonces, permite recibir los datos en tiempo real y procesarlos. Es de alto rendimiento, baja latencia y masivamente escalable.

Justamente esta es nuestra necesidad, ya que los datos van ingresando automáticamente y necesitamos procesarlos en tiempo real ya que la información se debe visualizar rápidamente.

Cuando ocurre un evento, en nuestro caso, cuando llega un nuevo dato, dispara una acción. Pensando en ETL, Apache Kafka sería la tecnología que permita la extracción.

Esta tecnología se puede utilizar nativamente en Apache hop.

Zookeeper

ZooKeeper es un servicio centralizado para mantener la configuración de información, nombrar, proporcionar sincronización distribuida y proporcionar servicios de grupo en sistemas distribuidos. Es necesario implementarlo para usar Kafka ya que Kafka usa ZooKeeper para gestionar y coordinar sus nodos de servidor Kafka (brokers). ZooKeeper actúa como una autoridad de consenso y configuración centralizada para el estado del clúster de Kafka, lo que incluye mantener la lista de temas y particiones, y la información sobre cuáles brokers están vivos o muertos. Además, ayuda en el balanceo de líderes de partición y en la elección de líderes en caso de falla de un broker. Esto es crucial para la operación confiable y eficiente de un clúster de Kafka. En nuestro caso se utiliza para orquestar un único cluster que se encarga de transmitir todas las alertas emitidas por los sensores de hidrometeorología, redireccionando todos estos por un solo topic para hacerle la tarea al workflow de Apache Hop más amena y permitir dinamismo en el manejo de los sensores disponibles, cabe destacar que la generación de mensajes es posible gracias al uso del conector que provee Debezium para bases de datos PostgreSQL, el cual se menciona y se describe a continuación.

Debezium

Debezium es una plataforma de código abierto para captura de datos modificados (CDC). CDC es un patrón que identifica y captura cambios en los datos de una base de datos, y luego los entrega en tiempo real a otros sistemas. Debezium se utiliza para monitorear las bases de datos, como es en nuestro caso PostgreSQL, y emitir un evento cada vez que se realiza un cambio en los datos. Esto es útil para actualizar sistemas externos de manera instantánea, lo que hace posible transmitir mediante un broker de mensajes de Kafka, las alertas de hidrometeorología de manera instantánea y aplicarle el preprocesamiento de manera automática y ser posteriormente almacenada en la base de datos definitiva.

Schema Registry

Schema Registry es un sistema que administra y controla los esquemas de los mensajes que se pasan entre servicios, particularmente en arquitecturas basadas en eventos o en microservicios. Cuando se utilizan servicios como Kafka para transmitir mensajes, es crucial que el formato de los mensajes (esquema) sea comprensible para todos los servicios que los reciben. Schema Registry ayuda a asegurar que todos los mensajes sigan el mismo formato, y también permite la evolución de los esquemas sin interrumpir los sistemas que dependen de ellos. Esto es esencial para mantener la integridad y la compatibilidad de los datos a lo largo del tiempo.

Base de datos relacional

Una base de datos relacional es un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí.

Las bases de datos relacionales se basan en el modelo relacional, una forma intuitiva y directa de representar datos en tablas. En una base de datos relacional, cada fila en una tabla es un registro con una ID única, llamada clave. Las columnas de la tabla contienen los atributos de los datos y cada registro suele tener un valor para cada atributo, lo que simplifica la creación de relaciones entre los puntos de datos.

En los primeros años de las bases de datos, cada aplicación almacenaba datos en su propia estructura única. Cuando los desarrolladores querían crear aplicaciones para usar esos datos, tenían que conocer muy bien esa estructura de datos concreta a fin de encontrar los datos que necesitaban. Esas estructuras de datos eran poco eficaces, el mantenimiento era complicado y era difícil optimizarlas para ofrecer un buen rendimiento en las aplicaciones. El modelo de base de datos relacional se diseñó para resolver el problema causado por estructuras de datos múltiples y arbitrarias.

El modelo de datos relacional proporcionó una forma estándar de representar y consultar datos que podría utilizar cualquier aplicación. Desde el principio, los desarrolladores se dieron cuenta de que la virtud principal del modelo de base de datos relacional era el uso de tablas, ya que era una forma intuitiva, eficiente y flexible de almacenar y acceder a información estructurada y represente la realidad que se quiere almacenar de la mejor manera posible.

El modelo relacional es el ideal para mantener la uniformidad de los datos en todas las aplicaciones y copias de la base de datos (llamadas instancias). Por ejemplo, cuando un cliente deposita dinero en un cajero automático y, a continuación, mira el saldo en un teléfono móvil, el cliente espera ver ese depósito reflejado inmediatamente. Las bases de datos relacionales son perfectas para este tipo de uniformidad, y garantizan que todas las instancias de una base de datos tengan los mismos datos en todo momento.

Normalización de la base de datos: consiste en la creación de tablas y el establecimiento de relaciones entre ellas según reglas diseñadas tanto para proteger los datos como para hacer que la base de datos sea más flexible al eliminar la redundancia y las dependencias incoherentes. Hay algunas reglas en la normalización de una base de datos. Cada regla se llama "forma normal". Si se cumple la primera regla, se dice que la base de datos está en "primera forma normal ". Si se observan las tres primeras reglas, se considera que la base de datos está en "tercera forma normal". Aunque son posibles otros niveles de normalización, la tercera forma normal se considera el nivel más alto necesario para la mayoría de las aplicaciones.

SQL

Con el tiempo, los desarrolladores comenzaron a usar el lenguaje de consulta estructurado (SQL) para escribir y hacer consultas en una base de datos: esto sería otra de las grandes virtudes del modelo relacional.

Se basa en el álgebra relacional y proporciona un lenguaje matemático de uniformidad interna que facilita la mejora del rendimiento de todas las consultas en bases de datos.

Metabase

Es una plataforma de inteligencia empresarial de código abierto diseñada para permitir a los usuarios crear, visualizar y compartir análisis de datos de manera fácil y rápida. Ofrece una interfaz intuitiva y basada en la web que permite a los usuarios explorar conjuntos de datos, realizar consultas, crear visualizaciones y paneles de control, todo sin necesidad de tener experiencia en SQL u otras herramientas de análisis de datos.

Algunas características importantes de Metabase incluyen:

1. Interfaz fácil de usar: Metabase proporciona una interfaz intuitiva y fácil de usar que permite a los usuarios realizar análisis de datos sin necesidad de conocimientos técnicos avanzados.
2. Exploración de datos: Los usuarios pueden explorar y filtrar conjuntos de datos utilizando una variedad de herramientas interactivas, como filtros, segmentaciones y agrupaciones.
3. Consultas SQL: Metabase permite a los usuarios ejecutar consultas SQL personalizadas para realizar análisis más avanzados o trabajar con conjuntos de datos más complejos.
4. Visualizaciones personalizadas: Ofrece una amplia gama de opciones de visualización, incluyendo gráficos de barras, líneas, pastel, mapas y tablas, que pueden personalizarse según las necesidades del usuario.
5. Paneles de control y programación de informes: Los usuarios pueden crear paneles de control personalizados para monitorizar métricas clave y programar informes automatizados para recibir actualizaciones periódicas por correo electrónico.
6. Integraciones: Metabase se integra con una variedad de bases de datos y servicios de almacenamiento de datos, como PostgreSQL, MySQL, SQLite, MongoDB, Amazon Redshift y Google BigQuery, lo que facilita la conexión y el análisis de datos almacenados en diferentes fuentes.

Repositorio

Un repositorio, en términos técnicos, es una ubicación distribuida donde se almacena y gestiona el código fuente de un proyecto de software y así también su configuración. GitHub es una plataforma ampliamente reconocida que permite a los desarrolladores cargar, versionar y colaborar en el código de manera eficiente. Así como también, permite el CI-CD.

Sus ventajas son:

Control de versiones: permite mantener un historial completo de todos los cambios realizados en el código a lo largo del tiempo. Esto es crucial para rastrear y comprender cómo ha evolucionado el software durante el desarrollo.

Colaboración: facilita la colaboración en equipo. Varios desarrolladores pueden trabajar simultáneamente en el proyecto, lo que es esencial para proyectos complejos. Así como la fácil integración de un nuevo desarrollador.

Transparencia: se promueve la transparencia en el proceso de desarrollo. Cualquier persona interesada (con control de acceso permitido) puede acceder al repositorio y examinar el código y los cambios realizados, lo que fomenta la revisión y mejora continua.

Seguridad: ofrece medidas de seguridad avanzadas y la capacidad de restringir el acceso a ciertos colaboradores, lo que garantiza que el código esté protegido.

GitHub

Es una plataforma en línea que facilita a los desarrolladores el almacenamiento, la gestión, el seguimiento y el control de cambios en su código. Actúa como un repositorio de código fuente que emplea Git, un sistema de control de versiones, para mejorar la colaboración entre desarrolladores dentro de un mismo proyecto, optimizando así el trabajo conjunto de forma eficiente.

Respecto a los conceptos de Integración Continua (Continuous Integration, CI) y Despliegue Continuo (Continuous Deployment, CD), que se explicarán con mayor detalle más adelante, GitHub desempeña un papel crucial. Ofrece las herramientas y la infraestructura necesarias para automatizar el proceso de pruebas y despliegue del software, componentes fundamentales de las prácticas de CI/CD.

Docker

Es una plataforma de código abierto diseñada para facilitar la creación, implementación y ejecución de aplicaciones en entornos livianos y portátiles llamados contenedores. Los contenedores son entornos virtualizados que encapsulan una aplicación junto con sus dependencias y configuraciones, garantizando una ejecución coherente en diferentes entornos, ya sea en un entorno de desarrollo local, en la nube o en un centro de datos.

Utilizando contenedorización, Docker aprovecha el kernel del sistema operativo para compartir eficientemente recursos y aislar aplicaciones. Esto garantiza que las aplicaciones empaquetadas en contenedores se ejecuten de manera coherente y predecible, independientemente del entorno. En el contexto de CI/CD, Docker proporciona un entorno reproducible, eliminando discrepancias entre entornos de desarrollo, prueba y producción.

Simplifica la creación de imágenes de contenedores, versiones ejecutables de aplicaciones, facilitando su distribución y ejecución. Esta capacidad hace de Docker una herramienta esencial para lograr consistencia y eficiencia en el despliegue de software en entornos modernos de CI/CD.

IaC o Infrastructure as Code (Infraestructura como código)

Es un enfoque en la administración y aprovisionamiento de infraestructura de tecnología de la información (TI) utilizando código y herramientas de desarrollo de software. En lugar de configurar manualmente servidores y recursos de infraestructura, los profesionales de TI pueden definir y gestionar la infraestructura mediante archivos de código que describen los recursos necesarios y sus configuraciones.

Algunas características y aspectos importantes de la IAC incluyen:

1. Automatización: La IAC permite la automatización del aprovisionamiento y la configuración de la infraestructura, lo que reduce errores humanos y mejora la consistencia en todo el entorno.
2. Repetibilidad: Al definir la infraestructura como código, los procedimientos para aprovisionar y configurar los recursos pueden ser replicados fácilmente en diferentes entornos, como desarrollo, pruebas y producción.
3. Control de versiones: Los archivos de código que describen la infraestructura pueden ser gestionados utilizando sistemas de control de versiones como Git, lo que permite un seguimiento de cambios, revisiones y colaboración entre equipos.
4. Documentación viva: La infraestructura definida como código sirve como documentación viva de la configuración y los requisitos de la infraestructura, lo que facilita la comprensión y la colaboración entre los equipos de desarrollo y operaciones.
5. Flexibilidad y agilidad: La IAC permite a las organizaciones adaptarse rápidamente a cambios en los requisitos y demandas del negocio al permitir la creación y modificación de infraestructura de manera rápida y controlada.

PaaS o Platform as a Service (Plataforma como Servicio)

Es un modelo de servicio en la nube que proporciona a los desarrolladores un entorno completo de desarrollo y despliegue de aplicaciones, eliminando la necesidad de gestionar la infraestructura subyacente. En lugar de preocuparse por la configuración y administración de servidores y sistemas operativos, los desarrolladores pueden centrarse en escribir código y crear aplicaciones utilizando las herramientas y servicios proporcionados por la plataforma. Facilita el despliegue automatizado.

Integración Continua (CI)

Es una práctica por la cual los desarrolladores integran o combinan el código en un repositorio común, facilitando la realización de test o pruebas para detectar y resolver posibles errores ya que se ejecutan automatizados. Con la CI se impide que se desarrollen distintas divisiones de una aplicación que luego puedan tener conflictos entre sí.

Los desarrolladores integran periódicamente su código en el repositorio central (como GitHub) en lugar de realizarlo de forma aislada al final del ciclo de producción (como se realizaba de forma tradicional). De esta forma se descubren antes los conflictos entre los nuevos códigos y los existentes, haciendo que la resolución de los mismos sea más sencilla y acarree un menor coste. Así también se verifica que el software funciona con cada

cambio, y si se rompe se arregla inmediatamente, el objetivo es tener la aplicación en un estado funcional constantemente, menos bugs, releases más rápidos.

Despliegue Continuo (CD)

Está relacionado estrechamente con la entrega continua. Con el despliegue continuo se va un paso más allá de la entrega continua, automatizando todo el proceso de entrega de software al usuario, eliminando la acción manual

Todo el proceso de despliegue sigue una serie de pasos que deben ejecutarse en orden y de forma correcta. Si alguno de estos pasos no se concluye de forma satisfactoria, el despliegue no se llevará a cabo. Por eso es fundamental que el diseño de la automatización de pruebas se realice de forma correcta, pues al no producirse ninguna entrada o acción manual, dependerá en gran medida de cómo sea ese diseño.

El despliegue continuo libera de carga a los equipos de operaciones de procesos manuales, que son una de las principales causas de retrasos en la distribución de aplicaciones.

Esta práctica está estrechamente ligada a la de integración continua, generalmente se aplican juntas. El despliegue depende de un desarrollo integrado y que no falle, y el concepto de integración continua de que el producto esté en estado funcional siempre, depende de que se despliegue también correctamente y en la misma velocidad, por lo tanto dicho despliegue también requiere la automatización.

También ayuda a reducir las tareas repetitivas y replicables y además teniendo mayor control de la operación.

Propuesta de solución

En el transcurso de la investigación realizada en el marco de este trabajo, se ha llevado a cabo un análisis exhaustivo del problema abordado, el cual ha llevado un aproximado de 3 meses completos al comienzo, pero que de igual manera fue acompañando al desarrollo hasta el final, lo cual hizo que los tiempos para las etapas más duras de desarrollo y entregables sean más acotadas. Sin embargo, este proceso ha permitido identificar las áreas críticas y las necesidades claves que demandan atención y resolución.

Basándonos en las conclusiones obtenidas durante la fase de investigación, en este apartado se propone avanzar con una solución integral que abarque desde la definición de los requisitos hasta la implementación y validación del sistema propuesto.

Requerimientos

Respecto a los requerimientos inicialmente establecidos, se observó una evolución significativa a lo largo del desarrollo del proyecto. Emergieron requerimientos no previstos, desconocidos incluso para el dueño del producto, que modificaron la concepción original de las necesidades del proyecto. Fue necesario desarrollar nuevos requerimientos no anticipados, que resultaron ser fundamentales para el avance del trabajo, ya que su ausencia hubiera representado un obstáculo para la implementación de los requerimientos propuestos inicialmente. Consecuentemente, se procedió a la eliminación de algunos requerimientos que dejaron de ser relevantes y a la incorporación de otros nuevos, con el fin de alinear el proyecto con los objetivos redefinidos.

En síntesis, estos son los requerimientos finales:

1. El sistema debe ser capaz de adquirir y procesar datos de todos los sensores que miden diferentes parámetros hidrometeorológicos.
2. El sistema debe automatizar la carga de datos de calidad de agua desde los dos formatos de hojas de cálculos existentes (ACSA e INA).
3. El sistema debe ser capaz de procesar los datos en tiempo real y detectar valores anormales(tratamiento de inconsistencias) y etiquetarlos para su posterior análisis.
4. El sistema debe permitir la integración correcta de las mediciones de ACSA a la estructura de datos del INA.
5. El sistema debe ser capaz de realizar el tratamiento y almacenamiento de datos hidrometeorológicos desde fuentes de entrada como texto plano o csv. (Inserción manual)

6. El sistema debe ser capaz de generar visualizaciones de los datos tanto hidrometeorológicos como de calidad de agua en tiempo real en una interfaz gráfica de usuario personalizable.
7. La base de datos debe reducir el tiempo de consulta con respecto a la existente.
8. El módulo de hidrometeorología debe estar desplegado en el servidor local del INA en la sede de Villa Carlos Paz y el módulo de calidad de agua debe estar desplegado en el servidor local del INA en la sede de Córdoba Capital.

Diseño

En esta etapa del proyecto, se extendieron las responsabilidades más allá del diseño de las mejoras planificadas. Fue esencial emprender un meticuloso proceso de identificación y documentación de las dependencias críticas que eran requeridas por el INA y que aún no se habían establecido. Este trabajo colaborativo, que implicó un esfuerzo conjunto y constructivo con el cliente, demandó una inversión de tiempo adicional que no había sido contemplada en la etapa de planificación inicial.

La tarea detallada consistió en realizar un inventario completo de los sensores, evaluando su estado actual e histórico, así como las variables que cada uno medía en diferentes momentos. A pesar de los desafíos presentados por la falta de una nomenclatura consistente para los sensores, se logró un reconocimiento eficaz de su ubicación y estado, gracias a la colaboración activa con el INA. Se hizo uso de la herramienta DataWise, la cual se adaptó con éxito para satisfacer las necesidades del proyecto, y se recopilaron meticulosamente los datos históricos de aquellos sensores cuya información no se encontraba en la base de datos del SGA.

Además, se estableció un diálogo constructivo con los responsables para seleccionar la solución de alojamiento web más adecuada, asegurando así el éxito y la sostenibilidad del proyecto. Cada una de estas acciones se realizó con un espíritu de cooperación y con el objetivo común de alcanzar los más altos estándares de calidad en el trabajo entregado.

Arquitectura del flujo de datos

El ETL fue pensado para procesar los datos hidrometeorológicos que ingresen en tiempo real, sin embargo, durante el desarrollo, tuvimos la necesidad de cargar los datos históricos (txt y csv) para tener una base de datos completa, por lo cual, se adaptó el ETL para que también se pudieran procesar, estructurar y almacenar dichos archivos con las mediciones históricas en la nueva base, a esto lo llamamos “ETL histórico”.

Es importante señalar que el diagrama de la arquitectura del flujo de datos se enfoca únicamente en el proceso ETL en tiempo real. Esto se debe a que la carga histórica de datos se realizó manualmente anteriormente y ya no se lleva a cabo como parte del proceso actual.

El diseño final se representa en el siguiente esquema:

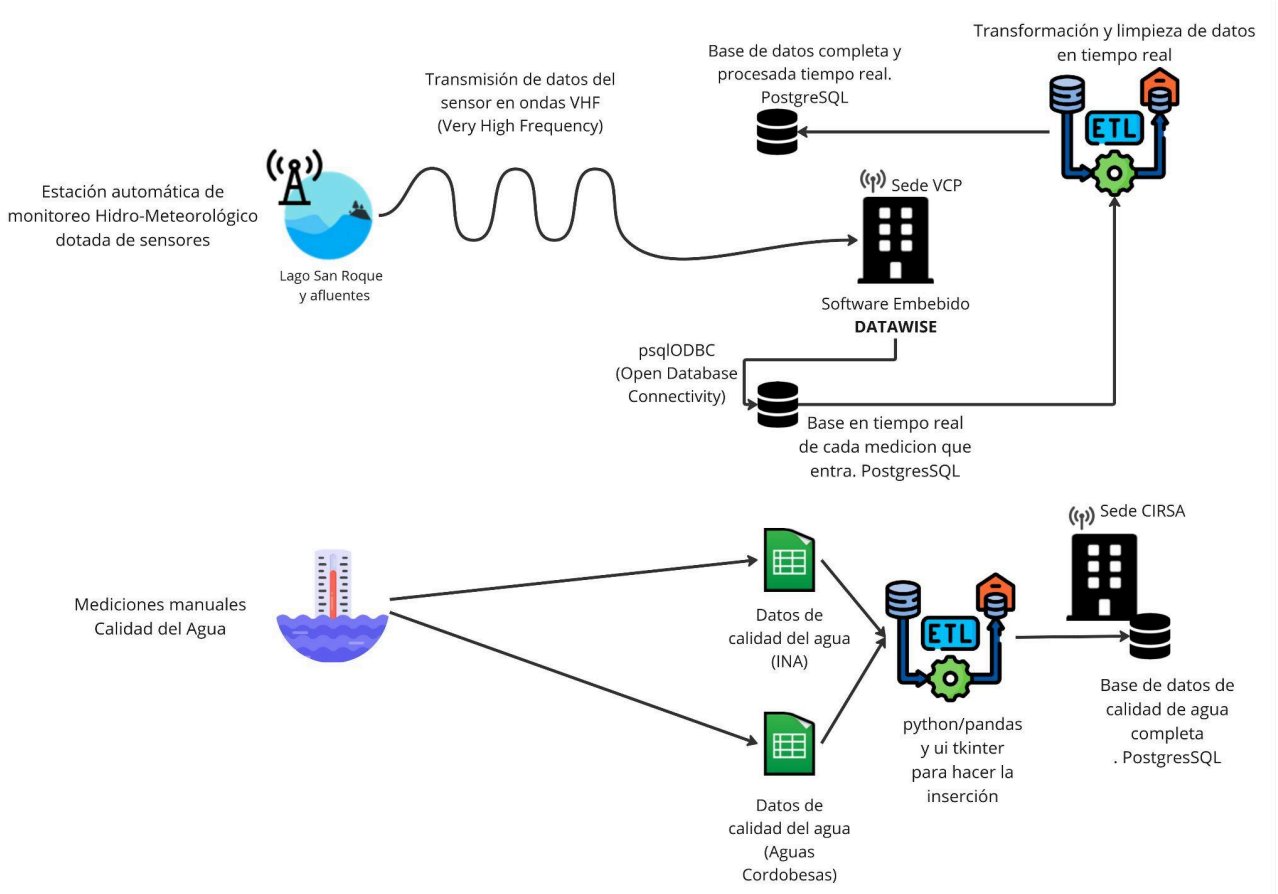


Diagrama arquitectura del flujo de datos luego del trabajo. Febrero del 2024.

Describiendo el diagrama, se aprecian los dos grandes módulos de datos:

Los hidrometeorológicos que su fuente es desde los sensores en las distintas cuencas del lago, que miden y envían la información por telemetría al decodificador en la sede en Villa Carlos Paz, donde DataWise recibe los datos, por ODBC los carga en una base PostgreSQL llamada “postgres”, alojada en el servidor local del INA. A través de Docker, en dicho servidor, están corriendon Debezium, SQLAlchemy, Zookeeper y Kafka, que su función es “escuchar” cuando un nuevo valor se carga en la base “postgres”, capturar la información y el ETL de Apache Hop, también en el mismo servidor, tiene como entrada, un consumidor de Kafka, entonces cuando Kafka reporta esta nueva entrada de dato, el ETL la procesa y la almacena en otra base de datos PostgreSQL, levantada con Docker, llamada “alerts” que es nuestra base final. Con Docker también se levanta una

imagen de Metabase, que es el que brinda la visualización e interfaz de usuario para visualizar los datos de la base alerts.

Los de calidad de agua que tienen como fuente las muestras de laboratorio y mediciones in-situ, una vez al mes, que hace tanto el INA como Aguas Cordobesas.

Dichos registros quedan, por parte de las dos instituciones, en archivos Excel. Desde el servidor local del INA-CIRSA en Córdoba Capital, a través de un programa de escritorio creado en este trabajo que por detrás ejecuta un script python con pandas, se selecciona el archivo Excel y se ejecuta para que este se almacene en la base de datos PostgreSQL llamada “water_quality”, que está corriendo a través de Docker en dicho servidor, donde también se levanta una imagen de Metabase para la visualización de los datos de calidad de agua.

Base de datos

Hidrometeorología:

Para hidrometeorología existía una base de datos histórica, su estructura no invitó a re-utilizarla, por lo cual se ha creado una base de datos nueva, normalizada hasta la 3er forma normal. Que ha reducido hasta en 100 veces el tiempo de espera en cada consulta.

Esta estructura permite manipular cada sensor, sin importar la variable que mida, de una misma manera. Así también permite dentro de la lógica del ETL no utilizar valores fijos, sino que se obtengan de la base de datos, lo que permite una gran mantenibilidad y una misma lógica genérica.

El diagrama entidad relación de la base de datos “alerts” es el siguiente:

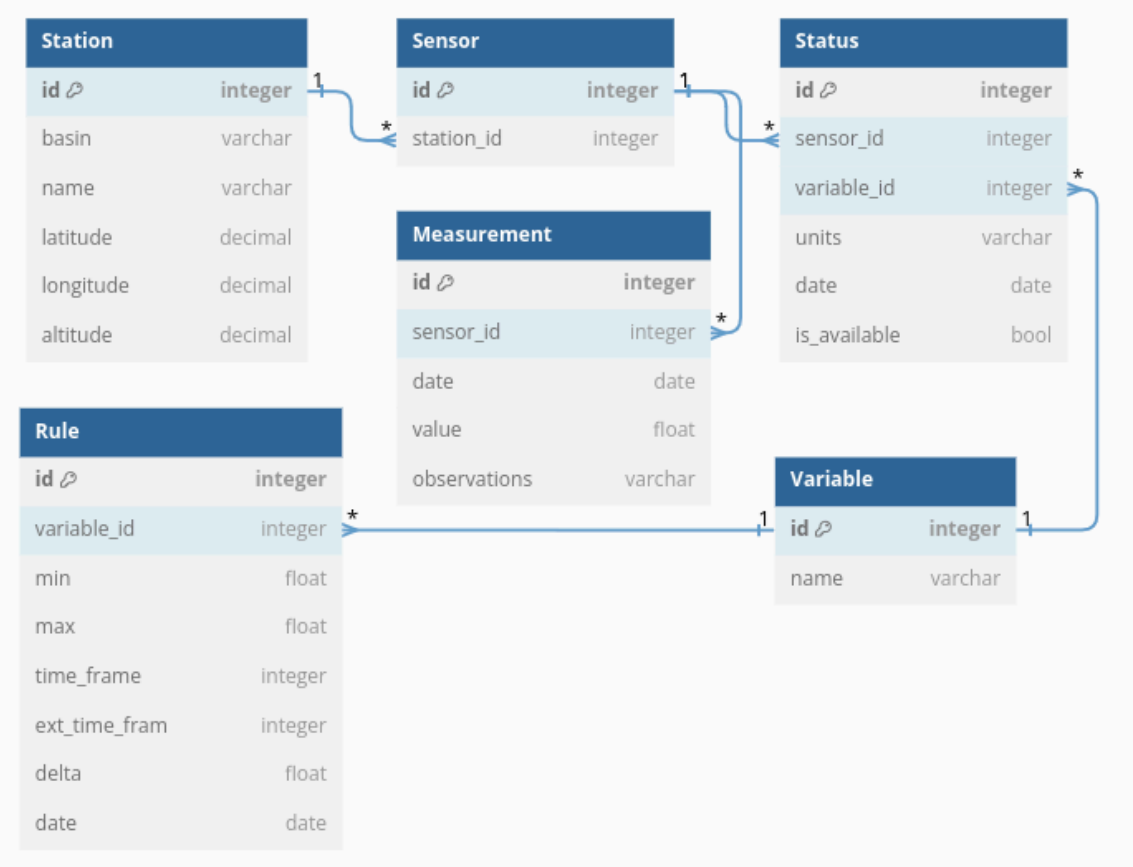


Diagrama entidad relación de la solución final a implementar para el almacenamiento de los datos de hidrometeorología.

Calidad de agua:

No existía una base de datos relacional para almacenar los datos de calidad de agua, son planillas que se actualizan cada mes y cada planilla almacena un año de monitoreos. Esto implica que la base de datos sean archivos sueltos, lo cual no permite la trazabilidad, ni consultas de datos de diferentes años.

Con esta base de datos relacional, cuidadosamente normalizada hasta alcanzar la tercera forma normal, se ha conseguido una representación completa de todos los encabezados presentes en la base de datos Excel. Consolidando toda la información en una única base de datos y eliminando la necesidad de archivos independientes, facilitando consultas rápidas y complejas.

Además, se destaca la capacidad de contar con tablas que almacenan información referencial, como el listado de perfiles, lo que facilita su actualización o expansión sin afectar al resto de la base de datos. Esta estructura también posibilita el almacenamiento tanto de las mediciones del INA como las de ACSA.

El diagrama entidad relación de la base de datos “water_quality” es el siguiente:

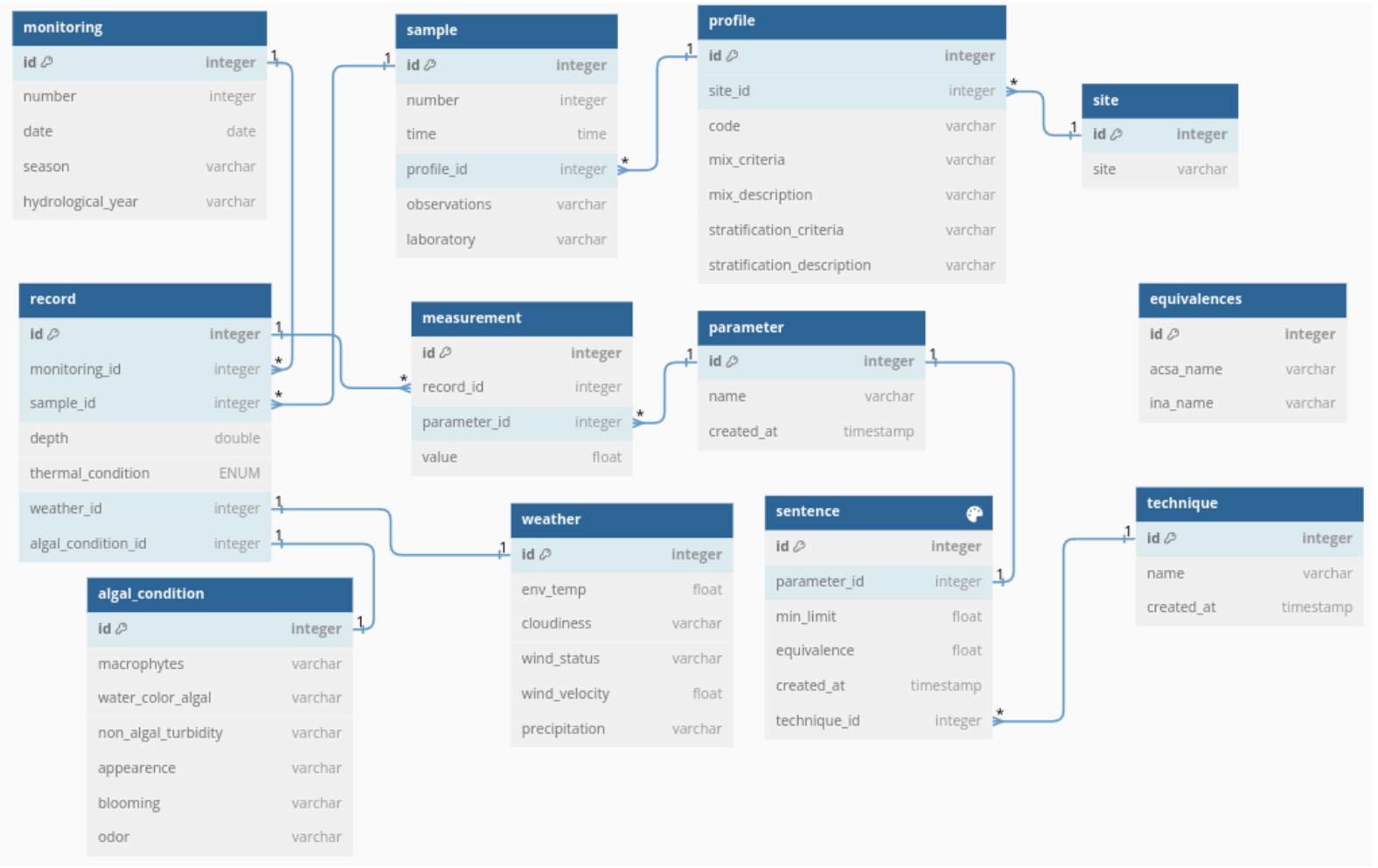


Diagrama entidad relación de la solución final a implementar para el almacenamiento de los datos de calidad de agua.

Stack tecnológico

Se analizó la arquitectura del sistema en tres grandes módulos: ETL para hidrometeorología, ETL para calidad de agua y visualización de información.

ETL para Hidrometeorología

Para llevar a cabo el ETL de hidrometeorología, se optó utilizar Apache Hop. Esta elección se basó en una serie de criterios fundamentales. En primer lugar, debido a su naturaleza de software libre que era un requerimiento, así también, ya se contaba con experiencia previa en el uso de un entorno de desarrollo de ETL visual similar, lo que facilitó la transición y la familiarización con la nueva plataforma.

Otro factor determinante fue la capacidad de Apache Hop para manejar de manera eficiente las complejas transformaciones y bifurcaciones requeridas en nuestro proceso de ETL. Desde el tratamiento de inconsistencias hasta la estructuración de columnas de diferentes fuentes de datos para que coincidan con el esquema de la base de datos,

Apache Hop nos brinda las herramientas necesarias para realizar estas tareas de manera efectiva y escalable.

Además, la posibilidad de realizar pruebas unitarias dentro del entorno de Apache Hop nos permite garantizar la calidad y la integridad de nuestros procesos de ETL. Esto es especialmente importante dada la variedad de inputs con los que trabajamos y la necesidad de manipularlos de manera consistente.

Finalmente por la capacidad de integración de Apache Hop con otros sistemas, especialmente aquellos que permiten la escucha en tiempo real de los datos entrantes, fue un factor decisivo en nuestra elección. Esto nos permite mantener nuestros datos actualizados y procesarlos de manera inmediata tan pronto como llegan, lo que es crucial para nuestras operaciones en tiempo real.

Para este último caso se hizo elección de las siguientes tecnologías: Apache Kafka que nos brinda la capacidad de gestionar flujos de datos a gran escala y en tiempo real, lo que se alinea perfectamente con nuestras necesidades de procesamiento de datos continuo y en tiempo real.

Zookeeper, que proporciona una gestión robusta de clústeres y coordinación de servicios distribuidos es fundamental para garantizar la confiabilidad y la escalabilidad de nuestra infraestructura, ya que coordina la comunicación entre los nodos del clúster Kafka y garantiza la coherencia y la integridad de los datos distribuidos.

Para la captura de cambios en bases de datos en tiempo real, elegimos Debezium, Esta integración es esencial para poder escuchar cuando y qué dato ingresa al PostgreSQL de sensores. Para interactuar con nuestra base de datos relacional y realizar operaciones de consulta y manipulación de datos, utilizamos SQLAlchemy.

ETL para Calidad de Agua

Se utilizó Python con Pandas para el programa de lectura, procesamiento y almacenamiento y la biblioteca Tkinter para desarrollar la interfaz visual para mantener la consistencia y hacer la interfaz también en Python. Esto permitió aprovechar la familiaridad del equipo con el lenguaje y garantizar una integración fluida entre la interfaz de usuario y el proceso de procesamiento de datos.

Al utilizar Python para todo el flujo de trabajo, garantizamos una integración fluida entre la interfaz de usuario y el proceso de procesamiento de datos, lo que simplifica el desarrollo y la mantenibilidad del sistema en su conjunto.

Visualización de información

Se optó por utilizar Metabase para la visualización de la información debido a varias razones que se mencionan en el marco teórico, en particular para esta solución, la

capacidad de exportar reportes, botones para realizar inserciones, paneles de control y consultas sql.

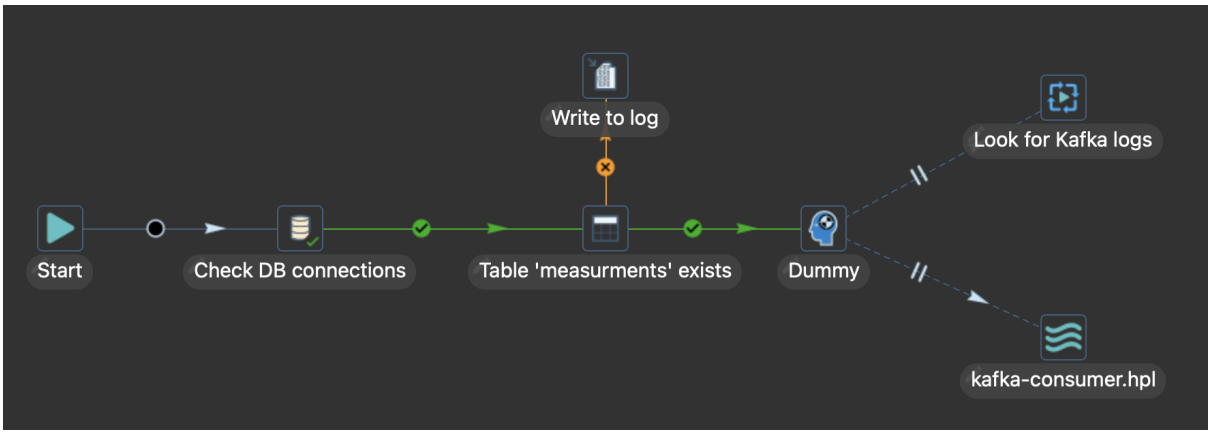
Además, Metabase ofrece una amplia gama de opciones de visualización, incluidos gráficos, tablas dinámicas y mapas, lo que permite representar los datos de manera clara, completa y comprensible.

Implementación

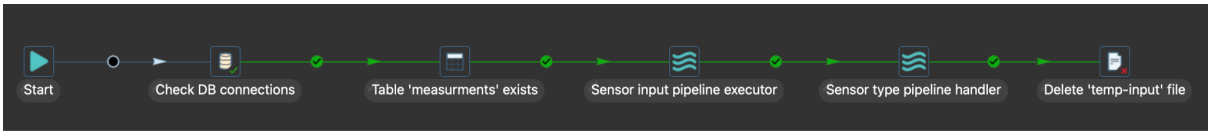
Módulo Hidrometeorología

En este trabajo se llevó a cabo un proceso de Extracción, Transformación y Carga (ETL) utilizando la herramienta Apache Hop. Dicho proceso se divide en dos funcionalidades clave: el procesamiento de datos históricos y el procesamiento de datos en tiempo real. Estas funcionalidades han sido diseñadas para compartir la mayor parte de su lógica operativa, diferenciándose exclusivamente en la fuente de los datos. Esta estrategia permite minimizar las posibilidades de error. Además, en caso de surgir algún inconveniente, facilita la implementación de una solución unificada que no solo reduce el esfuerzo requerido para futuras intervenciones, sino que también simplifica el mantenimiento del sistema.

A continuación, se muestran capturas de pantalla que ilustran los flujos de trabajo (‘workflows’), un término previamente definido en el marco teórico, tanto para el procesamiento de datos en tiempo real como para el de datos históricos:



Captura de pantalla del workflow encargado del procesamiento en tiempo real **‘realtime-alerts’**.



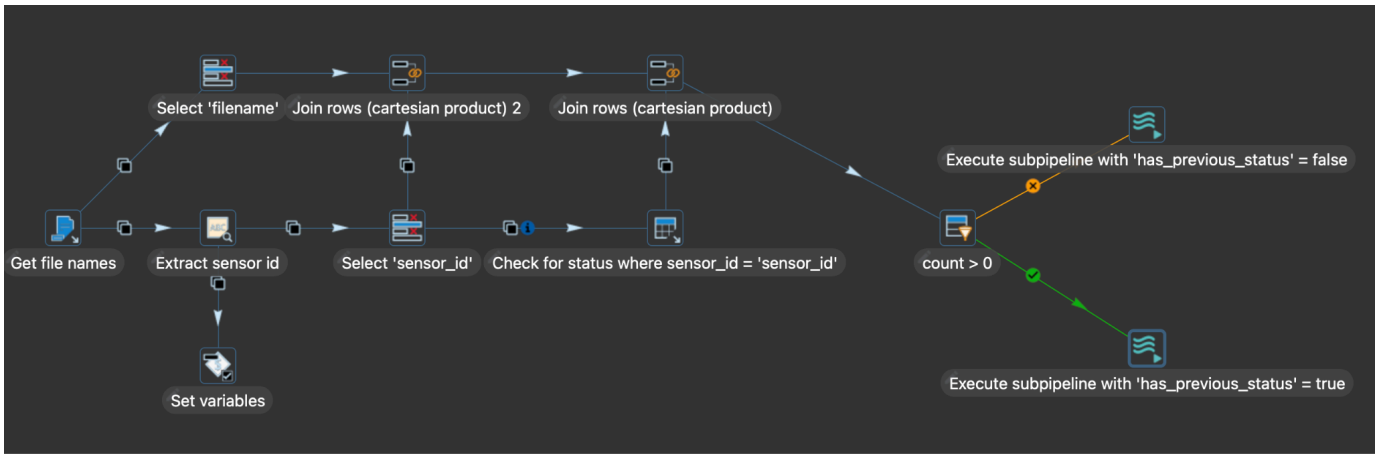
Captura de pantalla del workflow encargado del procesamiento de alertas **‘manual-alerts’**.

Con el fin de facilitar la comprensión y ante el desafío que representa visualizar los pipelines en su totalidad debido a su extenso tamaño, se optó por representarlos a través de una estructura modular. Esta aproximación permite detallar de manera más efectiva el propósito y funcionamiento de cada módulo implicado en el procesamiento de los datos de alertas. A continuación, se describirán los módulos principales y se explicará el papel que cada uno desempeña dentro del proceso global de tratamiento de datos.

Módulo de detección y actualización de estado de un sensor

Este módulo se encuentra comprendido por los siguiente pipelines:

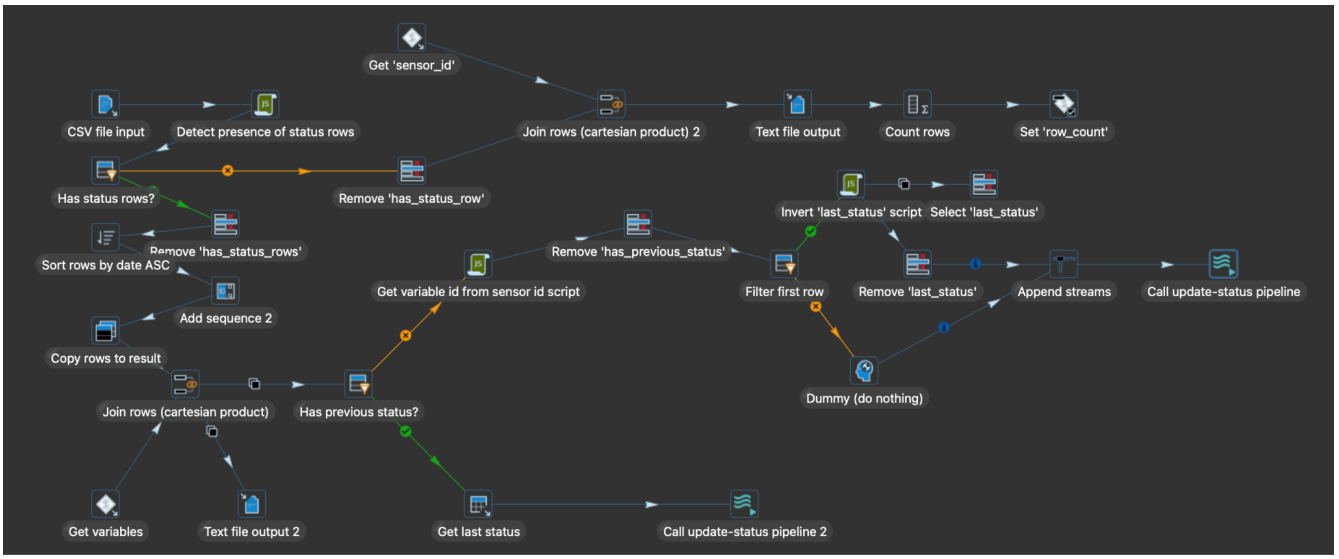
1. "sensor-input": En el contexto de la nomenclatura establecida por "DataWise", software mencionado anteriormente, los datos provenientes de los sensores se organizan siguiendo el formato "s{numero_sensor}". Esta convención permite la identificación del sensor a través del nombre del archivo, donde el número del sensor brinda información preliminar sobre el tipo de variable que mide, basándose en su terminación. Sin embargo, dado que esta correlación no se mantiene de manera consistente en todos los casos, se decidió asignar la variable medida específica en el campo "variable_id" de la tabla "statuses". Esta medida asegura una identificación precisa de la naturaleza de los datos recogidos por cada sensor.



Captura de pantalla del pipeline **'sensor-input'**.

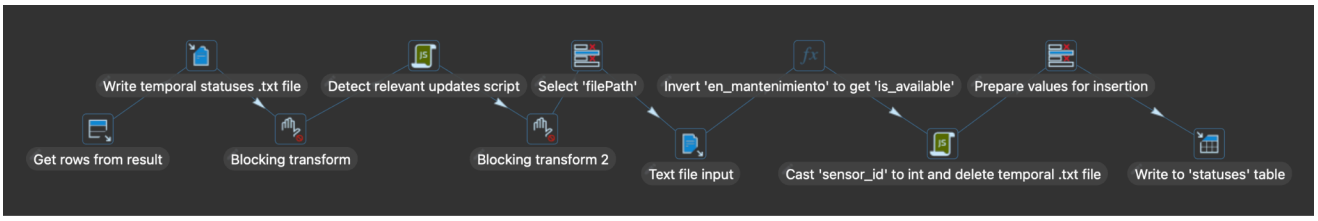
2. "sensor-input-subpipeline": Tras haber identificado el número del sensor y la variable que este mide a partir de la fuente de información, el siguiente paso consiste en actualizar el estado del sensor, si es necesario. Esta actualización se realiza mediante el uso de dos campos proporcionados por DataWise: "valido" y "en_mantenimiento". Estos campos son cruciales para registrar el historial de estados por los que transita un sensor a lo largo del

tiempo, facilitando así un seguimiento detallado de su funcionamiento y estado de mantenimiento.



Captura de pantalla del pipeline **'sensor-input-subpipeline'**.

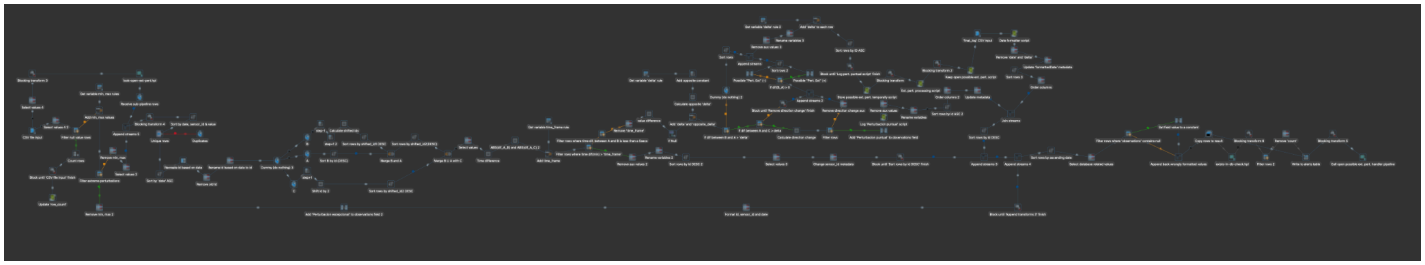
3. "update-status": Este pipeline supervisa el estado más reciente de cada sensor y, al detectar un cambio relevante y más actual que el registrado, procede a actualizar la base de datos con este nuevo estado. Además, este proceso habilita el seguimiento de las modificaciones en las variables medidas por los distintos sensores, una funcionalidad que aporta un valor significativo al sistema, ya que previamente no existía un registro de tales cambios.



Captura de pantalla del pipeline **'update-status'**.

Módulo de procesamiento de alertas

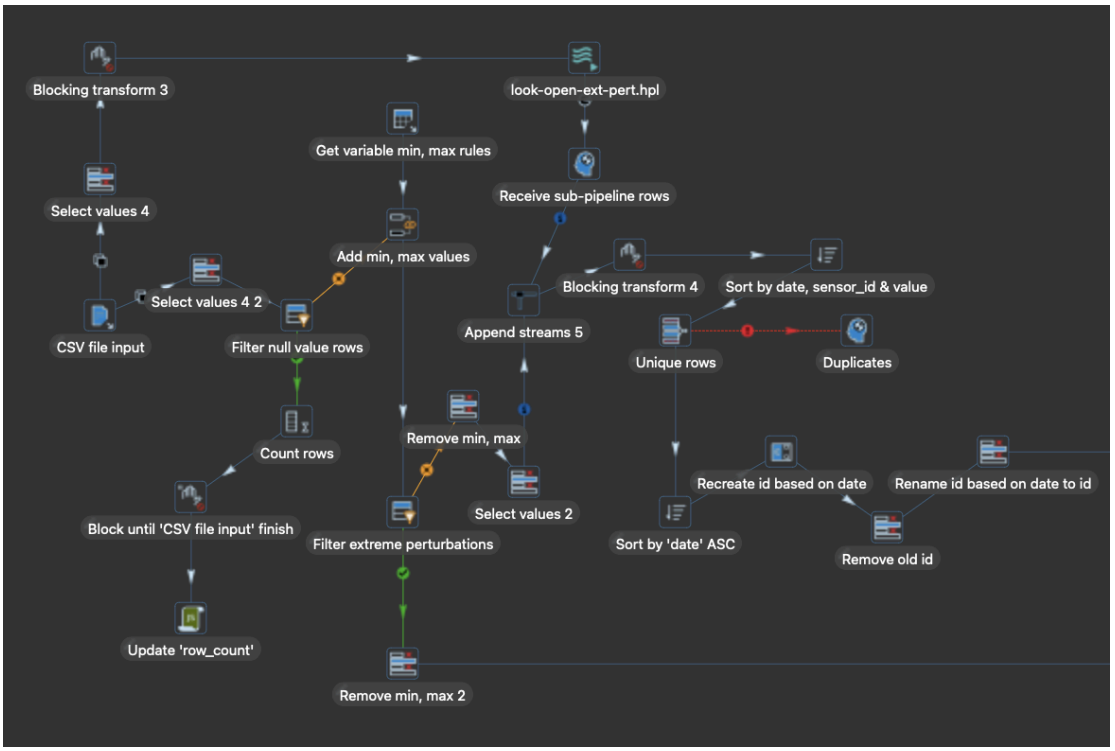
Con la información de la fuente de datos ya establecida, incluyendo el identificador del sensor (id del sensor) y la variable que mide, avanzamos hacia el procesamiento de estos datos. Este proceso, que constituye una etapa esencial de nuestro ETL, destaca por su complejidad. Para ilustrar de manera más detallada esta complejidad y los pasos involucrados en el procesamiento, se adjunta a continuación una imagen que muestra el flujo completo del proceso:



Captura de pantalla del pipeline **'alerts-handler'**.

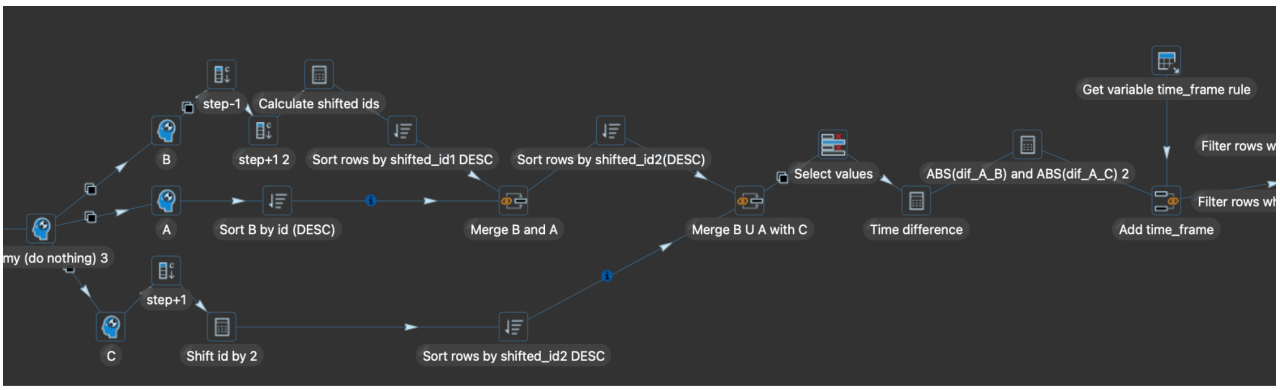
En el mismo se encuentran involucrados los siguientes pipelines:

1. "alert-handler": Es el pipeline principal, cuya estructura compleja se ilustró previamente. Para facilitar su manejo y comprensión, se ha desglosado en varios submódulos detallados a continuación::
    a. "Submódulo de preparación de datos": Este segmento del pipeline tiene la tarea de procesar los datos entrantes de las fuentes. Incluye la inserción de datos nuevos y, si es necesario, la inyección de datos previamente cargados que podrían estar sujetos a una "perturbación extendida", un concepto ya introducido en nuestro marco teórico. La metodología para detectar estas perturbaciones se explorará más adelante, en el sub-módulo dedicado a la clasificación de perturbaciones extendidas. En esta etapa inicial, también se establecen los parámetros para los límites mínimos y máximos de las perturbaciones excepcionales. Estos parámetros han sido definidos y almacenados en la tabla "rules", lo que permite su futura modificación sin necesidad de alterar el código. Este enfoque permite filtrar desde el inicio las perturbaciones excepcionales, optimizando el tiempo de procesamiento al evitar que estos datos pasen innecesariamente por las etapas subsiguientes del pipeline.



Captura de pantalla de fragmento de pipeline **'alert-handler'**.

b. "Submódulo de preparación de series temporales": Tal como su nombre sugiere, esta etapa se centra en la organización y preparación de las series temporales. Esto implica alinear en una sola fila los valores previo (B), actual (A), y futuro (C), facilitando así el cálculo de la variación de los valores y la diferencia de tiempo entre puntos secuenciales. Estos pasos son fundamentales para la detección de perturbaciones, un proceso que ha sido detalladamente fundamentado en el marco teórico. Con las series temporales ya estructuradas, se procede a ejecutar diversas operaciones analíticas, preparando el terreno para avanzar hacia la etapa de detección de perturbaciones.



Captura de pantalla de fragmento de pipeline **'alert-handler'**.

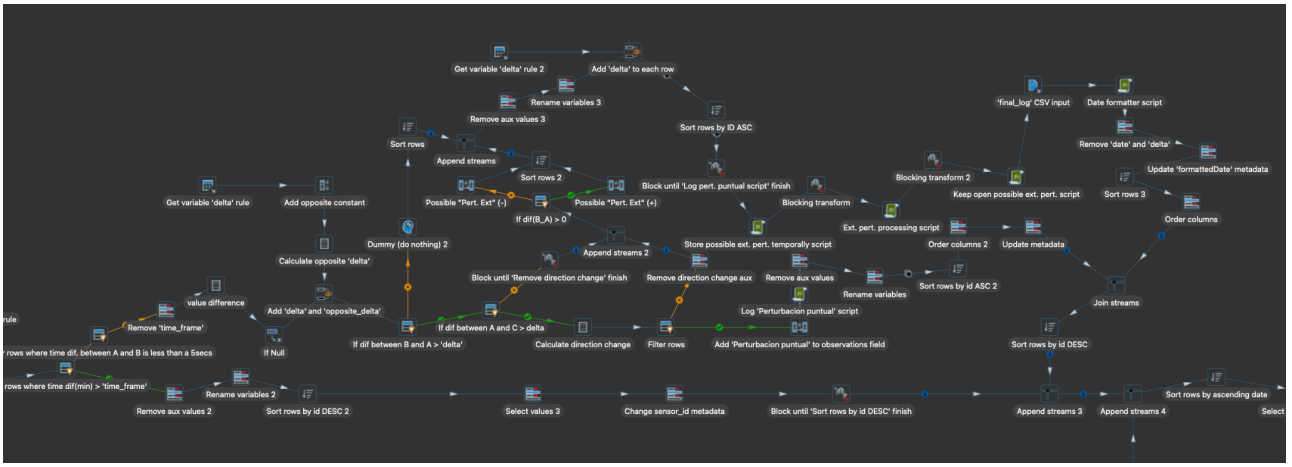
c. "Submódulo de detección de perturbaciones": Tras la preparación de las

series temporales y la obtención de las reglas específicas para cada variable desde la tabla "rules", se establece un punto de partida que consiste en analizar la ventana de tiempo entre los datos. Si los datos no se ajustan a esta ventana de tiempo establecida, se descartan para evitar el procesamiento innecesario. A partir de aquí, el análisis se desarrolla de acuerdo a los siguientes escenarios basados en las variaciones de valor:

i. Diferencia entre A y B menor al delta establecido: En este escenario, al no cumplirse la "condición de apertura" para el análisis, el dato se excluye del mismo. Sin embargo, se conserva para una evaluación futura en caso de que indique una perturbación extendida.

ii. Diferencia entre A y B mayor al delta establecido: Aquí se presenta una "condición de apertura" para el análisis, que puede derivar en dos posibles escenarios:

   1. Diferencia entre A y C con un delta mayor al establecido: En esta situación, se identifica una "Perturbación puntual". Se procede a etiquetar el dato como tal, concluyendo su procesamiento.
   2. Diferencia entre A y C con un delta menor al establecido: Este escenario indica la posibilidad de una "Perturbación extendida", concepto detallado en el marco teórico. Se etiqueta el dato como "Posible perturbación extendida (+)" o "Posible perturbación extendida (-)", según corresponda al signo del delta de variación. El fundamento detrás de esta clasificación se explicará detalladamente en la sección dedicada al script de procesamiento de posibles perturbaciones extendidas.perturbación extendida (+)", dependiendo del signo del delta de variación. La razón de esta etiqueta se desarrollará más adelante cuando se ingrese en detalle en el script de procesamiento de posibles perturbaciones extendidas.



Captura de pantalla de fragmento de pipeline **'alert-handler'**.

d. "Submódulo de procesamiento de perturbaciones extendidas": La detección y

análisis de perturbaciones extendidas representan la fase más compleja del procesamiento de datos. Este tipo de perturbación, por definición, puede abarcar un periodo que comprende *n* mediciones, lo que incrementa significativamente la complejidad de su análisis. Para abordar esta dificultad de manera efectiva, se ha recurrido a la implementación de varios scripts, diseñados para analizar de forma exhaustiva este fenómeno. Los scripts implementados se detallan a continuación:

i. "Store possible ext. temporary script": El script descripto actúa como una solución intermedia en el proceso de manejo y análisis de perturbaciones extendidas dentro de un sistema de procesamiento de datos. Su función principal es optimizar el tratamiento de grandes volúmenes de datos al almacenar temporalmente las observaciones relacionadas con posibles perturbaciones extendidas. Este enfoque reduce significativamente los tiempos de procesamiento que se experimentaban con métodos secuenciales anteriores, permitiendo manejar eficientemente grandes conjuntos de datos, como 300 mil entradas, en un tiempo reducido, potencialmente hasta un minuto. El script prepara el terreno para las etapas subsiguientes del procesamiento de datos, específicamente para un script de seguimiento que se encarga del análisis detallado y el etiquetado final de los datos.

   Al segmentar el proceso en dos etapas distintas: preparación inicial y procesamiento final, se mejora la eficiencia general del sistema y se facilita la gestión de datos a gran escala. Esta estrategia permite una manipulación más rápida y eficaz de los datos, crucial para el análisis de perturbaciones extendidas, que requieren una revisión detallada y pueden ser particularmente desafiantes de identificar y catalogar adecuadamente. El enfoque modular, caracterizado por el uso de scripts especializados para diferentes partes del proceso de análisis, no solo optimiza el rendimiento sino que también mejora la precisión del sistema de detección y etiquetado de perturbaciones, asegurando que los datos finales sean de alta calidad y confiables para su uso en aplicaciones posteriores.

ii. "Ext. pert. processing script": Este script desempeña un papel crucial en la fase final del análisis de perturbaciones en los datos, específicamente en la confirmación o descarte de posibles perturbaciones extendidas. Su funcionamiento se basa en la lectura y evaluación de los registros temporales generados por el script anterior. Aquí, el enfoque se centra en el uso del etiquetado preliminar, esencial para identificar variaciones en los datos que, aunque superan el umbral del delta, deben analizarse en el contexto de su persistencia temporal y su encuadre dentro de un intervalo de tiempo específico, que difiere del criterio de la ventana de tiempo usada para determinar si un dato es relevante para el análisis inicial.

   La importancia de almacenar el signo del delta radica en su papel para definir la existencia de una perturbación extendida, según se detalla en el marco teórico. Para que se considere una

perturbación de este tipo, debe haber un intervalo de datos caracterizado por deltas de signo inverso, lo que indicaría una estabilización o meseta en la variación de los datos, ya sea en aumento o disminución, que se sostiene a lo largo de un número *n* de mediciones. Este patrón sugiere una desviación significativa y sostenida del comportamiento normal esperado, lo que potencialmente señala una perturbación extendida. El script, por tanto, no solo verifica la magnitud de las variaciones (a través del delta) sino también su consistencia y duración en el tiempo, empleando para ello el concepto de deltas inversos como criterio de detección. Esta metodología permite una identificación más precisa de perturbaciones extendidas, diferenciándolas de las fluctuaciones puntuales o transitorias, y facilita una comprensión más profunda de las dinámicas subyacentes en los datos analizados. Este proceso es fundamental para garantizar la fiabilidad y precisión del análisis de datos, especialmente en aplicaciones donde la gestión y el monitoreo de sensores desempeñan roles críticos.

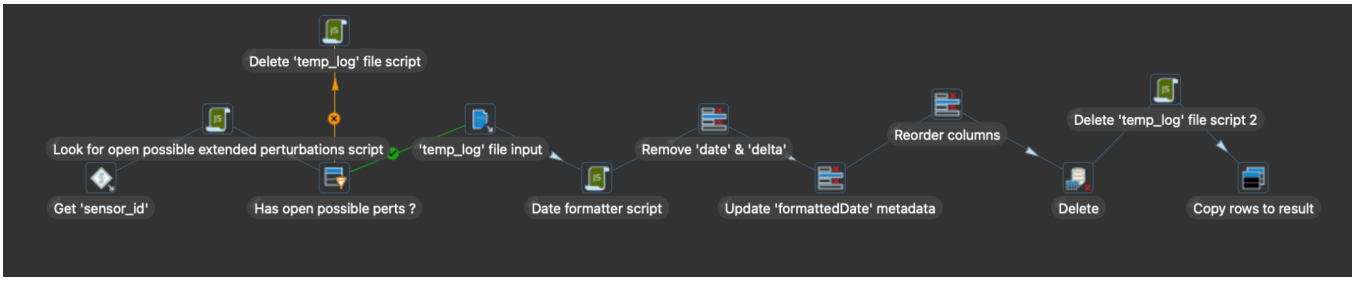
iii. "Keep open ext. pert. script": Este script final cumple una función esencial dentro del proceso de análisis de datos, especialmente en el contexto de perturbaciones. Su objetivo principal es verificar si existe alguna perturbación que haya quedado "abierta" o sin resolver al final del ciclo de procesamiento. Esta revisión es crítica para asegurar que el sistema esté preparado para integrar y analizar nuevas entradas de datos de manera continua, una característica fundamental para sistemas que operan en tiempo real. La lógica detrás de este script se basa en la capacidad de identificar y marcar aquellas perturbaciones que no han sido completamente cerradas o clasificadas durante el proceso de análisis previo. Al hacerlo, el script asegura que cualquier dato futuro que pueda estar relacionado con estas perturbaciones abiertas sea correctamente asociado y procesado en el contexto adecuado. Esta aproximación no solo mejora la precisión del sistema al tratar con datos dinámicos y en constante cambio, sino que también facilita la adaptabilidad del sistema para operar eficientemente en un entorno de tiempo real.

Funcionalmente, el script realiza una última pasada por los datos procesados para identificar cualquier indicio de perturbaciones que no se hayan resuelto plenamente. Esto incluye verificar marcadores, etiquetas, o cualquier otro indicativo utilizado durante el proceso de análisis para señalar una perturbación en curso. Si se encuentra alguna, el script toma las medidas necesarias para dejar estos casos en un estado que permita su fácil identificación y posterior análisis con la llegada de nuevos datos.

Implementar un script de este tipo es crucial para sistemas de monitoreo y análisis en tiempo real, donde la capacidad de responder de manera fluida y continua a nuevos datos es esencial para mantener la integridad y relevancia del análisis. Este enfoque garantiza que el sistema no solo sea capaz de manejar datos

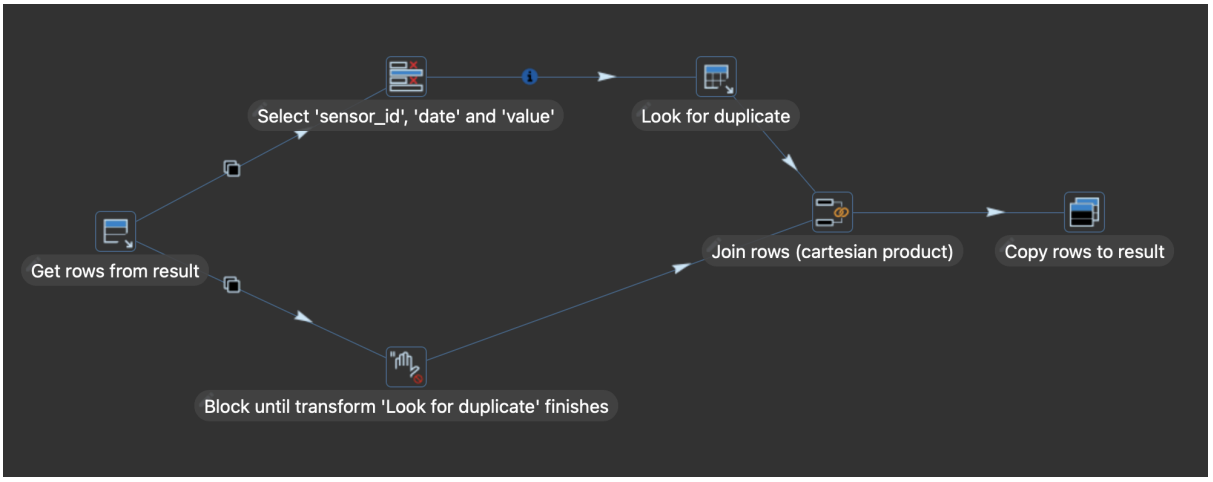
históricos con precisión, sino que también esté preparado para adaptarse y responder a condiciones cambiantes en tiempo real, manteniendo así un alto nivel de precisión y relevancia en sus operaciones de análisis.

2. "look-open-ext-pert": Este pipeline actúa como un elemento inicial crítico dentro del flujo de procesamiento de datos, ejerciendo una función clave en la gestión de perturbaciones extendidas. Su principal responsabilidad es encadenar aquellas perturbaciones extendidas que, según el análisis realizado por el script descrito anteriormente, han quedado "abiertas" o no completamente resueltas. Al hacerlo, el pipeline asegura una transición fluida y coherente de los datos a lo largo del tiempo, evitando cualquier tipo de discontinuidad en el historial de análisis.



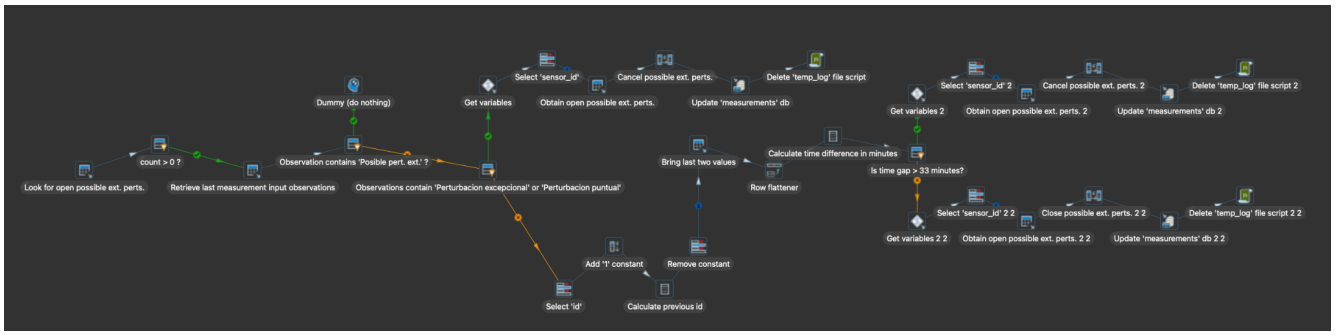
Captura de pantalla de pipeline **'look-open-ext-pert'**.

3. "exists-in-db-check": Este pipeline se encarga de revisar que no se inserten datos duplicados, pensado principalmente para el procesamiento de tiempo real.



Captura de pantalla de pipeline **'exists-in-db-check'**.

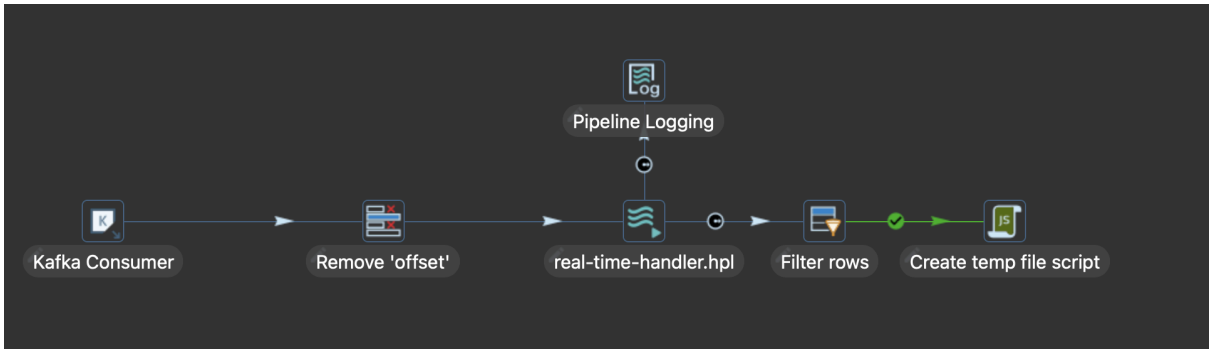
4. "open-possible-ext-pert-handler": Finalmente, se realiza una revisión final para asegurarse que si han quedado perturbaciones extendidas sin cerrar, estas sean válidas, y no hayan quedado errores de procesamiento en los datos ingresados en la base de datos.



Captura de pantalla de fragmento de pipeline **'open-possible-ext-pert-handler'**.

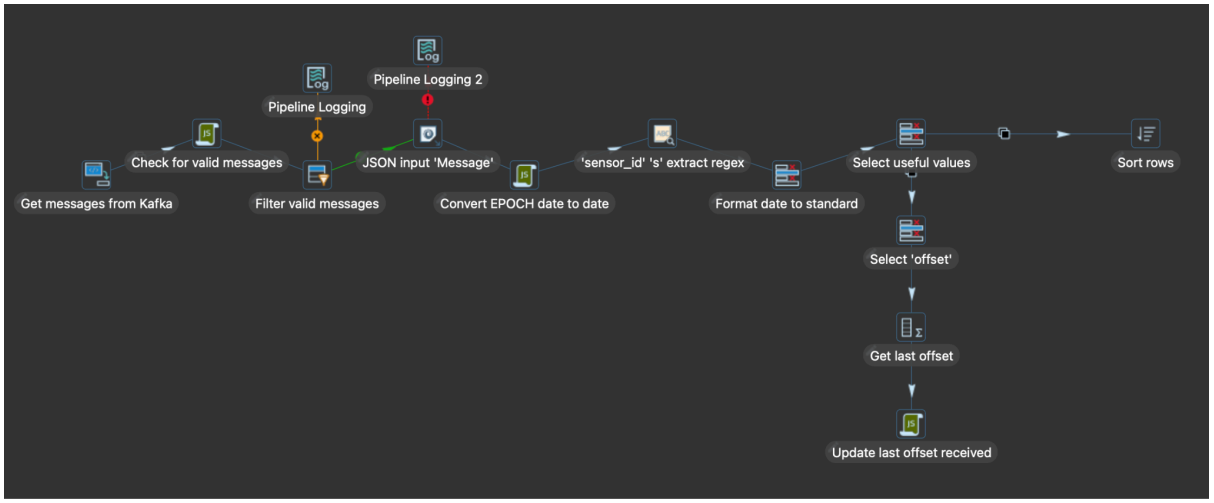
Módulo de procesamiento de alertas en tiempo real

El módulo de alertas en tiempo real, especialmente diseñado para manejar y responder a eventos de datos de manera inmediata. Este módulo se basa en la integración de un consumidor de mensajes de Apache Kafka, lo que le permite recibir y procesar flujos de datos en tiempo real de manera eficiente y escalable.



Captura de pantalla de de pipeline **'kafka-consumer'**.

El cual contiene un pipeline interno para el procesamiento y formateo de los mensajes consumidos por la transformación:

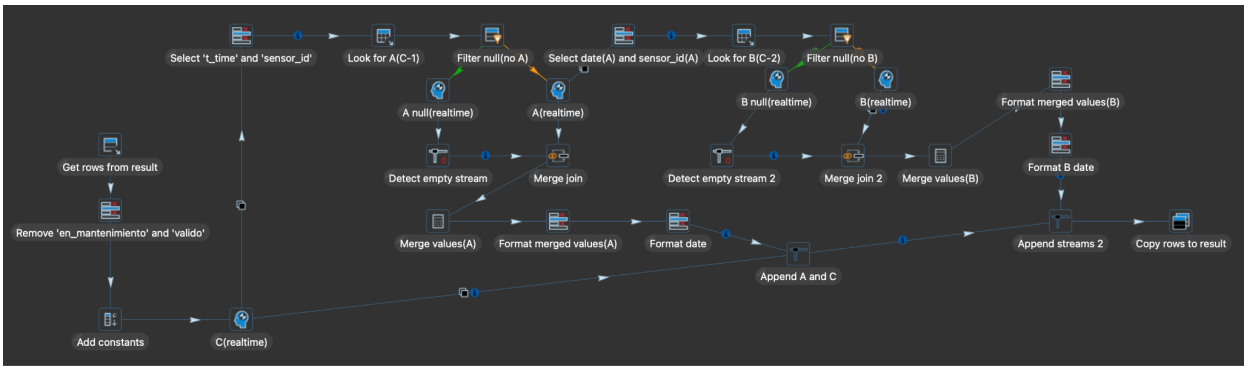


Captura de pantalla de de pipeline **'kafka-message-processor'**

Los mensajes recibidos vienen en formato JSON, el cual se decodifica para ser procesado por el Hop, como lógica adicional, se almacena el “offset” del mensaje, que sería algo similar al índice, para así evitar leer un mensaje en más de una ocasión.

Este pipeline ya retorna un mensaje con el mismo formato que espera el workflow de alertas que se explicó con anterioridad.

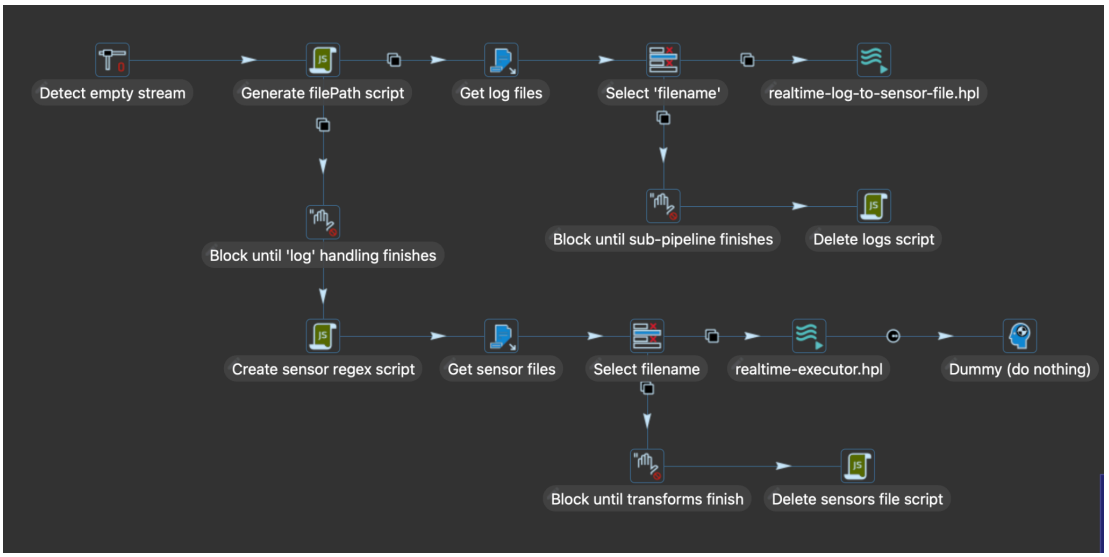
Otro aspecto importante a detallar de este pipeline es el subpipeline que se observa en la imagen “real-time-handler”, el cual podemos ver a continuación:



Captura de pantalla de de pipeline **‘kafka-consumer’**

El mismo es el encargado de una función crucial en el procesamiento real, ya que el procesamiento de perturbaciones consta en un análisis de series temporales, es imposible contar con el dato siguiente en este caso, por lo cual se considera a el valor entrante como nuestro valor futuro (C) y es este pipeline, el encargado de que, en caso de que existan, devuelva los valores previos (A) y (B) para completar la serie temporal.

Una vez ya leídos estos mensajes se escribe en un registro temporal, para que luego, el otro pipeline que es ejecutado en paralelo, realice un “barrido” en intervalos de 60 segundos, y tome los registros que fueron escritos y haga una división de los mismos por sensores, ya que nuestro ‘workflow’ de alertas está preparado para recibir valores de un único valor en simultáneo, y en este caso, los mensajes receptados provienen de todos los sensores que se encuentren disponibles en el momento. Una vez que llama al mismo ‘workflow’ de alertas que es utilizado en el procesamiento manual. Se asegura de eliminar los registros temporales que el otro pipeline en paralelo escribió, evitando así la lectura de un mensaje en más de una ocasión.



Captura de pantalla de de pipeline **'real-time-looker'**

Módulo Calidad de Agua

Se ha realizado un script python con pandas, que permite realizar la lectura de archivos tipo hoja de cálculo (xlsx). El método es propiamente el de un ETL.

Son dos los scripts, uno para archivos hoja de cálculos del INA y otro para hoja de cálculos de ACSA, ya que difieren un poco en cuanto a los encabezados.

Previamente, definimos lo que es el método de Levenshtein que cumple un rol clave en ambos scripts. Levenshtein es un algoritmo utilizado para medir la similitud entre dos cadenas de texto. Se basa en el número mínimo de operaciones necesarias para transformar una cadena en otra, el algoritmo calcula la distancia de edición entre dos cadenas contando el número de operaciones requeridas para convertir una cadena en la otra. Esta distancia se puede utilizar para determinar cuán similares son dos cadenas. Nosotros lo utilizamos para reconocer valores y encabezados similares a los existentes en la base de datos. Ya que por ejemplo ACSA manda parámetros con nombres sutilmente diferentes a los que quiere el INA y esto evita tener que hacer una relación de equivalencia manual para cada parámetro siendo que son más de 130.

El script va a revisar por cada parámetro a cual se asemeja más, lo mismo con los códigos de perfiles. En caso de que todas las comparaciones sean malas. Se le muestra un mensaje al usuario de si desea agregar alguna equivalencia para el parámetro entrante no reconocido y esta equivalencia queda almacenada, esto no se dará muy seguido pero para cuando vengan parámetros nuevos o muy distintos, sean contemplados.

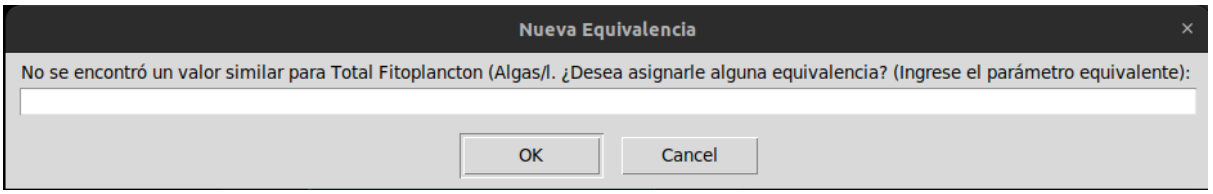


Imagen de la ui, donde solicita una nueva equivalencia.

En el caso de que el usuario no reconozca ese parámetro, o no quiera almacenarlo, para preservar la integridad de los datos, la operación de carga se cancela y no se inserta ningún dato en la base.

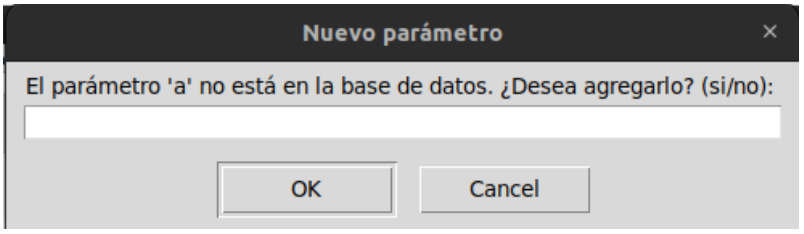


Imagen de la ui, donde solicita agregar nuevo parámetro.

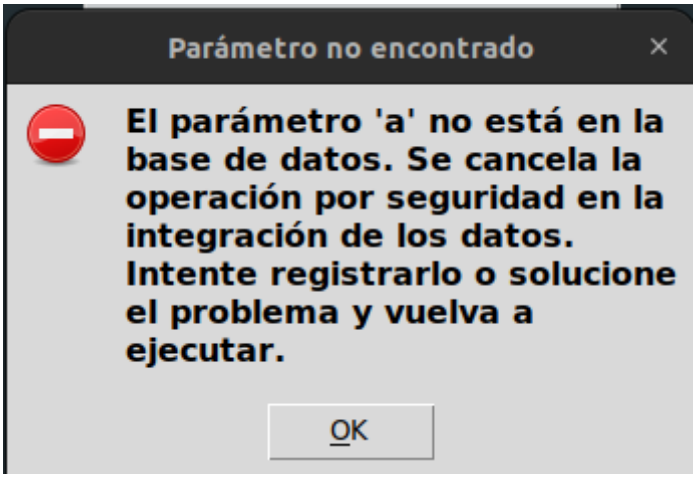


Imagen de la ui, prevención de inconsistencias.

El script para ACSA realiza algunas operaciones para obtener los valores deseados, adaptándose a la forma en que los reportan, por ejemplo, para el número de muestra, ellos lo envían en un encabezado llamado *'Nro lote insp'* y el dato viene por ejemplo: *900000144380.* Por lo que se hace una transformación que busque los valores de ese encabezado, los limpie para que quede 144380 que es el valor de muestra a almacenar.

Así también para reconocer el perfil en el que se tomó la muestra, ACSA lo manda de la siguiente manera, *'Dirección*' y valor: *LAGO SAN ROQUE C1.* Entonces se aplica una transformación para quedarse con C1, de ahí, que consiga el profile_id haciendo una búsqueda a la tabla para ese código (C1). Pero también mandan Direcciones que no están igual en la tabla de perfiles, como *LAGO SAN ROQUE LOS CHORRILLOS* que en la tabla

el código es RLC. Para lo cual se utiliza el modelo de Levenshtein. Y si no, se crea una equivalencia para dicho caso.

Luego realiza un bucle para recorrer cada fila, armando tuplas (encabezado, valor) ya con los encabezados que representan parámetros. Así también aplica Levenshtein para encontrar equivalencia entre el encabezado y un parámetro existente y realiza la inserción.

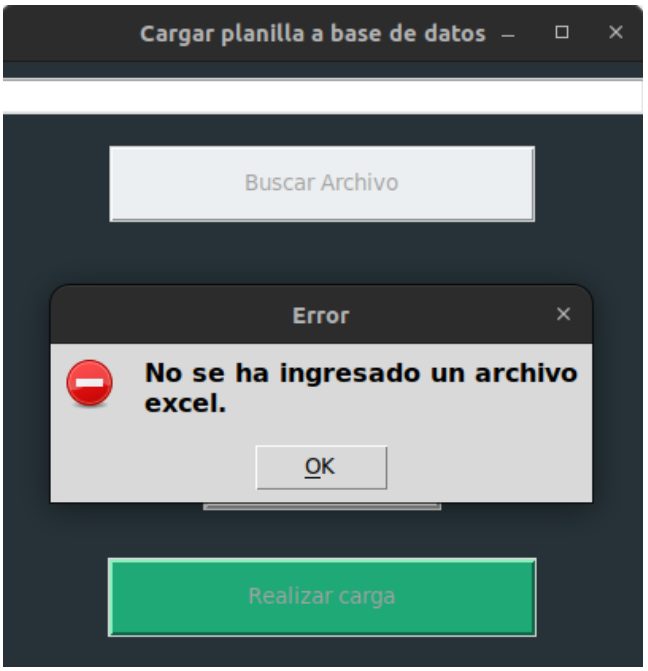
Para el caso de cargar un archivo de ACSA, se hace una transformación en los valores ya que como fue detallado en el marco teórico, muchas técnicas de medición tienen un límite de medición, entonces cuando sucede esto, se imputa un valor por convención. Ocurre que ACSA imputa un valor pero INA tiene una convención distinta, entonces para los valores, en este script se analiza si está dicho parámetro en la tabla de sentencias, si está, revisa si el valor es el imputado por límite de medición, y guarda la medición con el valor equivalente del INA.

Para cada una de las inserciones que hace el script, primero revisa que esta no se encuentre en la base para evitar duplicados.

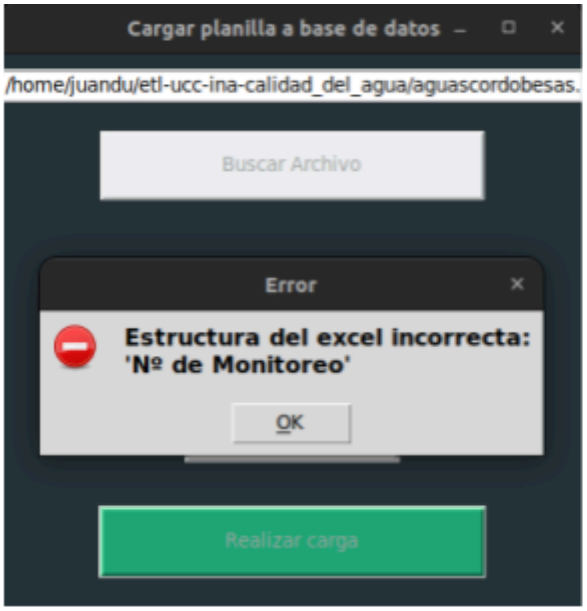
El script para INA es un poco más directo e implica menos transformaciones, primero se obtienen los encabezados que no representan parámetros de medición sino más bien de registro o característicos, como son información de monitoreo, de muestra y de registro, así como características del clima y características de la floración. Para estos datos, se hacen inserciones directas ya que los encabezados son reconocidos y no cambian. Luego para poder armar tuplas (encabezado, valor), se itera sobre los encabezados que son parámetros, se aplica también Levenshtein y se realiza la inserción.

Obviamente este script también tiene prevención de inconsistencias, agregar equivalencias o nuevos parámetros y revisar que la información a insertar no se encuentre en la base para evitar duplicados.

Algunas capturas de la aplicación:

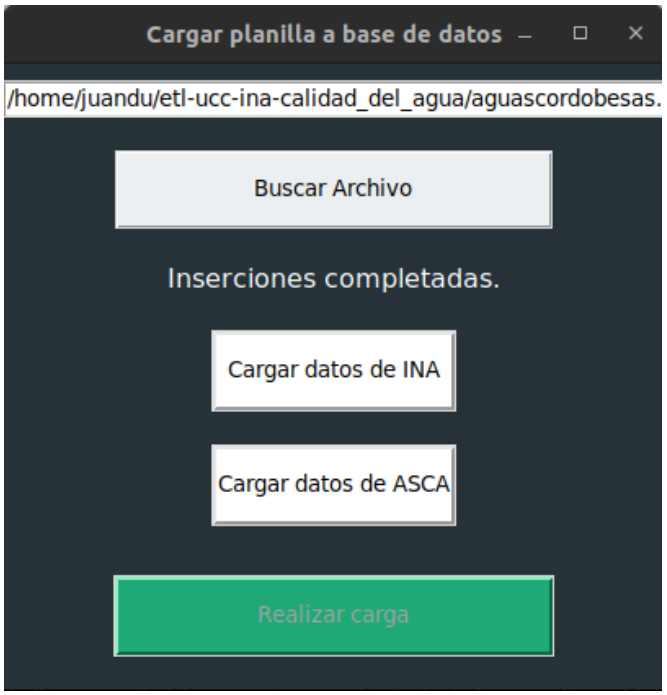


No corre ETL si no se ha ingresado un archivo o si no es formato hoja de cálculo



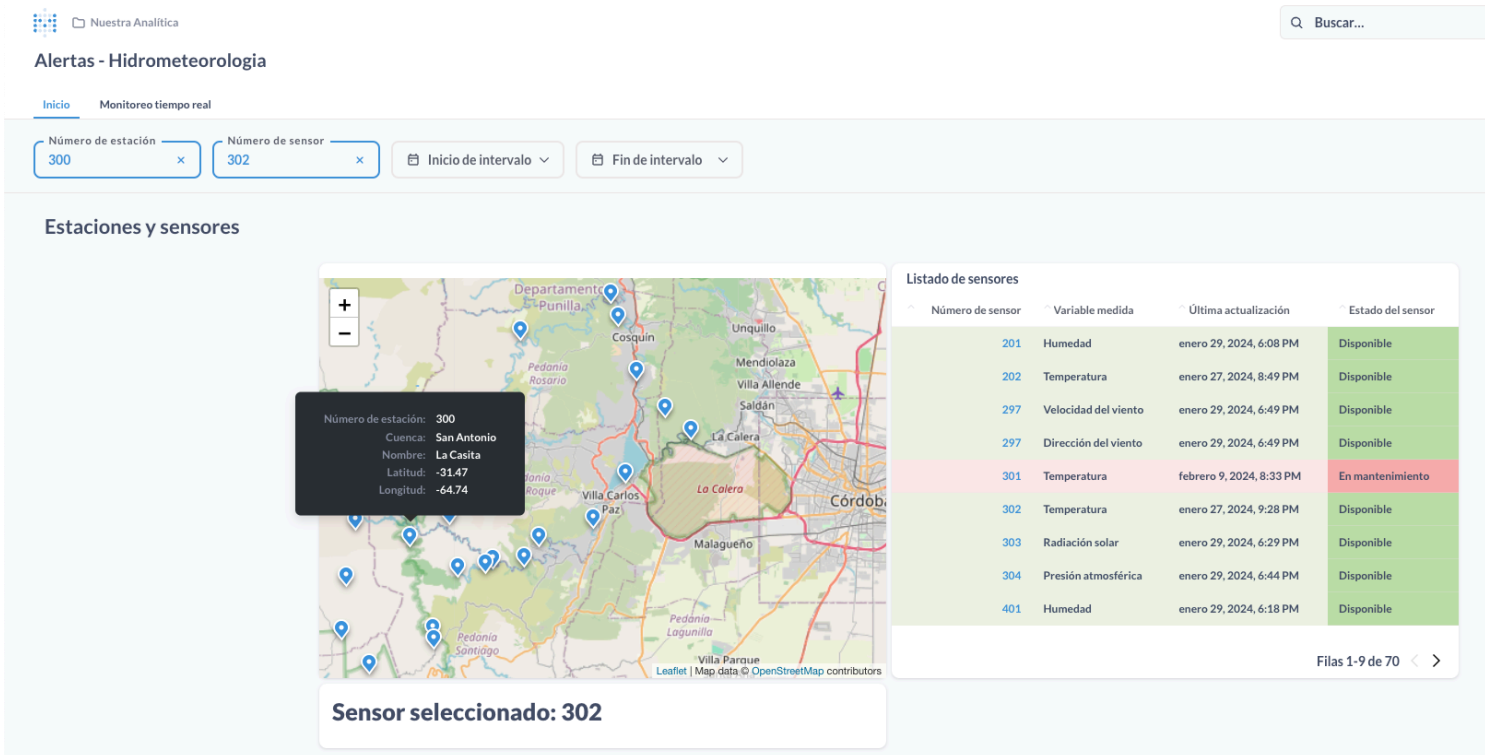
Manejo de errores para prevenir inconsistencias, ayudando al usuario a encontrar el problema.

El botón para ejecutar el script permanece desactivado si no se seleccionó que script utilizar.

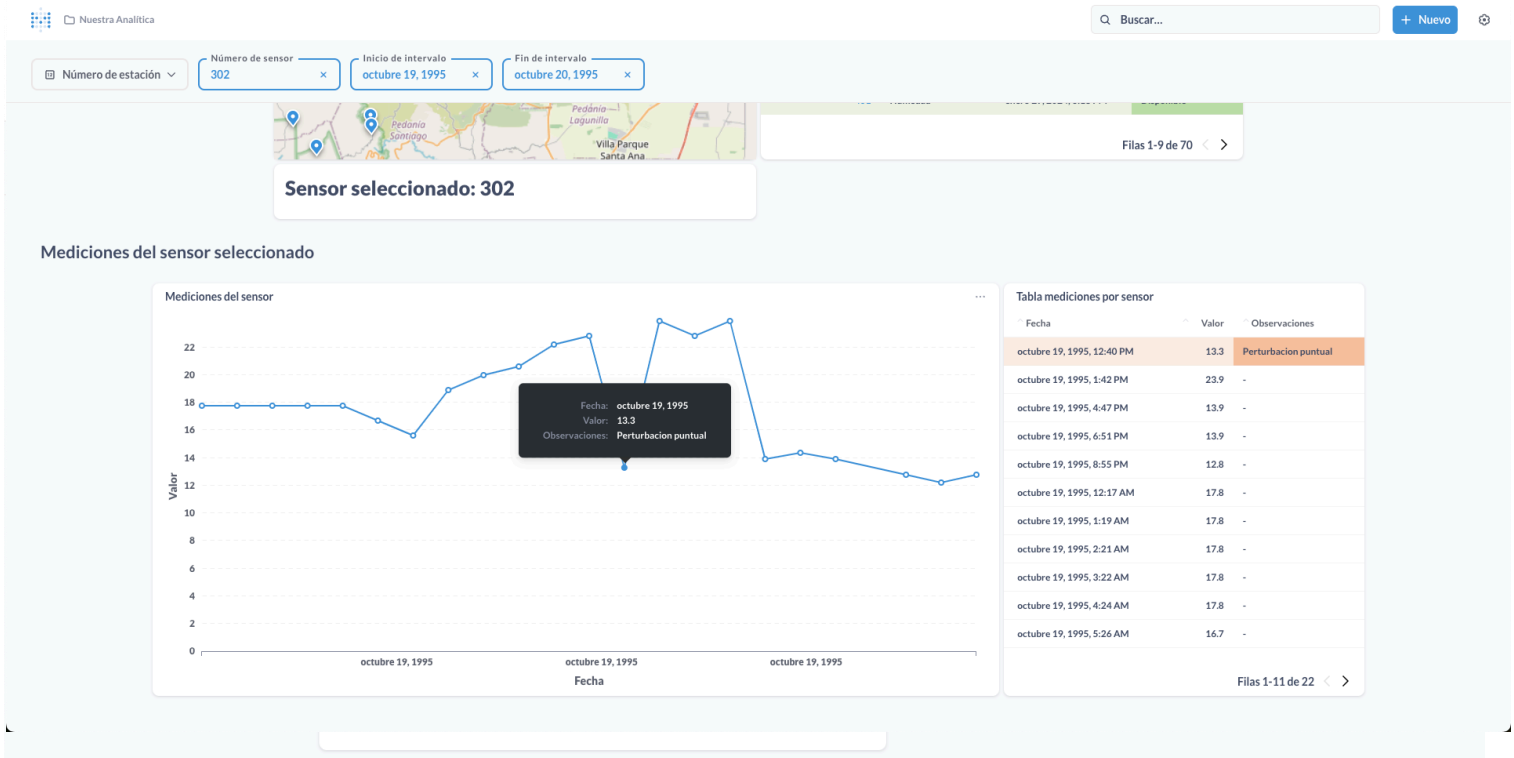


Mensaje cuando las inserciones han sido completadas con éxito.

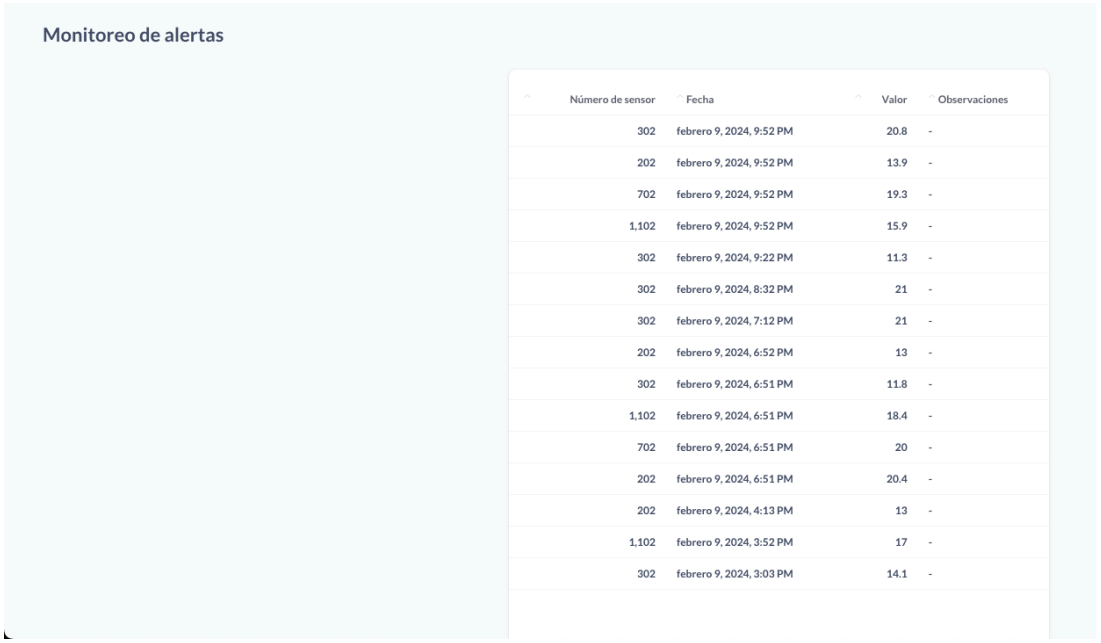
Interfaz de Usuario Visual



Vista de la Interfaz visual de hidrometeorología. Se mapean las estaciones, se ve el estado de todos los sensores.



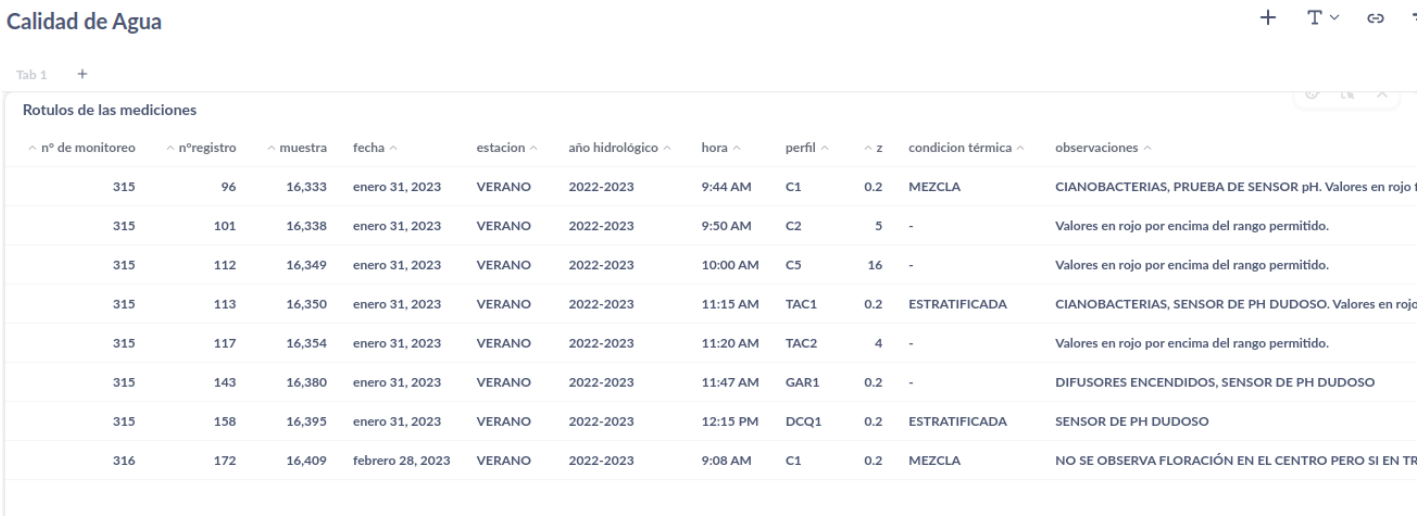
Vista de la Interfaz visual de hidrometeorología. Se elige un sensor y fechas y trae la serie temporal, se muestran los datos y destaca con color cuando hay una perturbación.



Monitoreo de alertas

Número de sensor	Fecha	Valor	Observaciones
302	febrero 9, 2024, 9:52 PM	20.8	-
202	febrero 9, 2024, 9:52 PM	13.9	-
702	febrero 9, 2024, 9:52 PM	19.3	-
1,102	febrero 9, 2024, 9:52 PM	15.9	-
302	febrero 9, 2024, 9:22 PM	11.3	-
302	febrero 9, 2024, 8:32 PM	21	-
302	febrero 9, 2024, 7:12 PM	21	-
202	febrero 9, 2024, 6:52 PM	13	-
302	febrero 9, 2024, 6:51 PM	11.8	-
1,102	febrero 9, 2024, 6:51 PM	18.4	-
702	febrero 9, 2024, 6:51 PM	20	-
202	febrero 9, 2024, 6:51 PM	20.4	-
202	febrero 9, 2024, 4:13 PM	13	-
1,102	febrero 9, 2024, 3:52 PM	17	-
302	febrero 9, 2024, 3:03 PM	14.1	-

Vista de la Interfaz visual de hidrometeorología. Es el monitor en tiempo real donde se ven las mediciones que van entrando, para todos los sensores y si existiesen, se visualizan alertas o perturbaciones rápidamente.



Calidad de Agua

Tab 1

Rotulos de las mediciones

nº de monitoreo	nºregistro	muestra	fecha	estacion	año hidrológico	hora	perfil	z	condicion térmica	observaciones
315	96	16,333	enero 31, 2023	VERANO	2022-2023	9:44 AM	C1	0.2	MEZCLA	CIANOBACTERIAS, PRUEBA DE SENSOR pH. Valores en rojo
315	101	16,338	enero 31, 2023	VERANO	2022-2023	9:50 AM	C2	5	-	Valores en rojo por encima del rango permitido.
315	112	16,349	enero 31, 2023	VERANO	2022-2023	10:00 AM	C5	16	-	Valores en rojo por encima del rango permitido.
315	113	16,350	enero 31, 2023	VERANO	2022-2023	11:15 AM	TAC1	0.2	ESTRATIFICADA	CIANOBACTERIAS, SENSOR DE PH DUDOSO. Valores en rojo
315	117	16,354	enero 31, 2023	VERANO	2022-2023	11:20 AM	TAC2	4	-	Valores en rojo por encima del rango permitido.
315	143	16,380	enero 31, 2023	VERANO	2022-2023	11:47 AM	GAR1	0.2	-	DIFUSORES ENCENDIDOS, SENSOR DE PH DUDOSO
315	158	16,395	enero 31, 2023	VERANO	2022-2023	12:15 PM	DCQ1	0.2	ESTRATIFICADA	SENSOR DE PH DUDOSO
316	172	16,409	febrero 28, 2023	VERANO	2022-2023	9:08 AM	C1	0.2	MEZCLA	NO SE OBSERVA FLORACIÓN EN EL CENTRO PERO SI EN TR

Vista de la interfaz visual de calidad de agua.

Mediciones por registro

parámetro	valor	Nº de Registro
pH	8.57	96
pH Lab	8.4	96
OD (mg/l)	7.61	96
% sat OD	103.4	96
Cond. (µS/cm)	288	96

Filas 1-5 del primer 2000

Vista interfaz visual calidad de agua. Mediciones de cada parámetro con su valor y el número de registro.

Valor maximo alcanzado y promedio por cada parámetro

Parámetro	Maximo alcanzado	fecha	valor promedio
% sat OD	168.9	septiembre 26, 2023	81.34
Alcalin. (mg/l)	128	septiembre 26, 2023	73.74
Amphipleura	280	mayo 30, 2023	3.68
Ankistrodesmus	96,000	agosto 22, 2023	1,263.16
Aphanocapsa	31,000	abril 25, 2023	644.74
Aulacoseira	650,000	septiembre 26, 2023	13,651.43
Ca (mg/l)	31.2	septiembre 26, 2023	18.58
Ceratium	4,800,000	septiembre 26, 2023	187,574.03
Chlamydomonas	400,000	septiembre 26, 2023	22,115.32
Chroomonas	3,300,000	septiembre 26, 2023	246,624.29

Filas 1-10 de 65

Vista interfaz visual de calidad de agua. Se visualizan para cada parámetro, el valor máximo alcanzado y en qué fecha, y el promedio.

Pruebas

El ciclo de pruebas es un elemento crítico en el desarrollo de software; y en este proyecto, dada la importancia de la precisión y veracidad de los datos, se ha enfatizado su ejecución con un alto grado de profesionalismo. La fase de pruebas fue meticulosamente planificada y ejecutada bajo la supervisión y mentoría del profesor e ingeniero Lucas Argañaraz (UCC). Gracias a su experta orientación, el plan de pruebas ha sido calificado como excepcionalmente profesional.

Con el objetivo de documentar y preservar la metodología y el rigor aplicados durante esta fase, se han incluido en este cuaderno las secciones más importantes del documento del Plan de Pruebas. Esto asegura que el conjunto de pruebas implementadas y los resultados obtenidos permanezcan accesibles para consulta y revisión futura.

El plan de pruebas con sus entregables, forma parte de los artefactos entregables del sistema. Muchos casos se hacen a través de código por lo que esto se entrega a través del Github en el repositorio.

Estrategia de ejecución de pruebas

Introducción

La presente estrategia de pruebas está específicamente diseñada para el pipeline ETL encargado del tratamiento y procesamiento de datos de calidad de agua. El enfoque está en garantizar la precisión y validez de los datos procesados a través de pruebas automatizadas y manuales.

Estrategia General

La estrategia se basa en pruebas unitarias automatizadas, pruebas manuales y pruebas de regresión continuas para validar las reglas de negocio y asegurar el correcto funcionamiento del pipeline.

Ciclos de prueba

Ciclo 1: Pruebas Unitarias Automatizadas

Las pruebas unitarias se ejecutan para cada componente del pipeline ETL, validando las reglas de negocio aplicadas en cada paso.

- **Casos de Prueba**: Verifica la transformación de datos, el manejo de excepciones, y que los datos de salida coincidan con las expectativas dadas los datos de entrada.
- **Automatización**: Implementada a través de herramientas de testing y disparadas por eventos de commit en el repositorio de GitHub.

CASOS DE PRUEBA RELACIONADOS

Todos los casos de prueba relacionados a las siguientes historias de usuario:
US1 - Procesamiento de datos de temperatura
US2 - Procesamiento de datos de humedad relativa
US3 - Procesamiento de datos de presión atmosférica

US4 - Procesamiento de datos de radiación solar
US5 - Procesamiento de datos de dirección del viento
US6 - Procesamiento de datos de velocidad del viento

Ciclo 2: Pruebas Manuales

Se realizan pruebas manuales para validar el flujo de datos a través del pipeline y la interacción con interfaces externas o internas del sistema.

- **Casos de Prueba:** Incluyen la verificación de la carga y extracción de datos, la ejecución de transformaciones complejas y la inspección visual de los datos resultantes.

CASOS DE PRUEBA RELACIONADOS

Todos los casos de prueba relacionados a la siguiente historia de usuario:
US1 - Procesamiento de datos de temperatura
US2 - Procesamiento de datos de humedad relativa
US3 - Procesamiento de datos de presión atmosférica
US4 - Procesamiento de datos de radiación solar
US5 - Procesamiento de datos de dirección del viento
US6 - Procesamiento de datos de velocidad del viento
US10 - Obtención de datos de sensores al sistema de base de datos
US11 - Procesamiento de datos de calidad de agua del INA
US12 - Procesamiento de datos de calidad de agua del ACSA

1. Lookup:
    a. almacenar información respectiva a estaciones (ubicación, nombre, sensores).
    b. almacenar información respectiva a parámetros, sentencias, equivalencias, perfiles.
2. Sensores: tener seguimiento histórico y estado actual de los sensores.
3. Parámetros: tener seguimiento histórico y actual de los parámetros y sus sentencias.
4. Datos procesados:
    a. Tener todas las mediciones limpias de errores y con etiquetas en caso de perturbación.
    b. Tener lo mismo de la hoja de cálculo en la base de datos.

Ciclo 3: Pruebas de Regresión Automáticas

Se ejecutan para asegurar que los nuevos cambios no afecten negativamente las funcionalidades existentes.

- **Automatización**: A través del CI/CD integrado en GitHub, estas pruebas se disparan automáticamente después de cada push al repositorio.
- **Casos de Prueba**: Revisan las funcionalidades clave del pipeline para detectar posibles regresiones.

CASOS DE PRUEBA RELACIONADOS

Al ejecutarse toda la suite de tests unitarios para asegurar el funcionamiento correcto en cada cambio se ven involucrado todos los casos de prueba relacionados a las siguientes historias de usuario:
US1 - Procesamiento de datos de temperatura
US2 - Procesamiento de datos de humedad relativa
US3 - Procesamiento de datos de presión atmosférica
US4 - Procesamiento de datos de radiación solar
US5 - Procesamiento de datos de dirección del viento

US6 - Procesamiento de datos de velocidad del viento

Ciclo 4: Pruebas de Caja Negra y Mono

Estas pruebas se centran en verificar la funcionalidad del sistema sin conocer los detalles internos de las operaciones del software.

- **Pruebas de Caja Negra**: Evalúan el sistema solo desde el exterior, asegurando que el proceso completo, desde la entrada hasta la salida de datos, funcione como se espera.
- **Pruebas de Mono**: Se realizan para asegurar que cada componente del pipeline interactúa adecuadamente con los demás componentes y con el entorno de ejecución.

CASOS DE PRUEBA RELACIONADOS

Todos los casos de prueba relacionados a la siguiente historia de usuario:
US1 - Procesamiento de datos de temperatura
US2 - Procesamiento de datos de humedad relativa
US3 - Procesamiento de datos de presión atmosférica
US4 - Procesamiento de datos de radiación solar
US5 - Procesamiento de datos de dirección del viento
US6 - Procesamiento de datos de velocidad del viento
US10 - Obtención de datos de sensores al sistema de base de datos
US11 - Procesamiento de datos de calidad de agua del INA
US12 - Procesamiento de datos de calidad de agua del ACSA

1. Lookup:
    a. almacenar información respectiva a estaciones (ubicación, nombre, sensores).
    b. almacenar información respectiva a parámetros, sentencias, equivalencias, perfiles.
2. Sensores: tener seguimiento histórico y estado actual de los sensores.
3. Parámetros: tener seguimiento histórico y actual de los parámetros y sus sentencias.
4. Datos procesados:
    a. Tener todas las mediciones limpias de errores y con etiquetas en caso de perturbación.
    b. Tener lo mismo de la hoja de cálculo en la base de datos.

Tipos de pruebas a realizar

- **Pruebas Unitarias Automatizadas:** Para las reglas de negocio individuales dentro de cada paso del ETL.
- **Pruebas Manuales:** Para la validación del flujo de datos y la interacción con otros sistemas.
- **Pruebas de Regresión Automáticas:** Para garantizar la estabilidad del pipeline tras cambios en el código.
- **Pruebas de Caja Negra y Mono:** Para comprobar la integridad y la interacción correcta del pipeline como un sistema.

Orden de ejecución

1. **Pruebas Unitarias Automatizadas**: Se ejecutan inmediatamente después de cualquier cambio en el código.
2. **Pruebas Manuales**: Se realizan tras la ejecución exitosa de las pruebas unitarias.
3. **Pruebas de Regresión Automáticas**: Automáticamente disparadas después de las

pruebas unitarias y manuales.

4. **Pruebas de Caja Negra y Mono**: Se llevan a cabo como una verificación final antes de la puesta en producción.

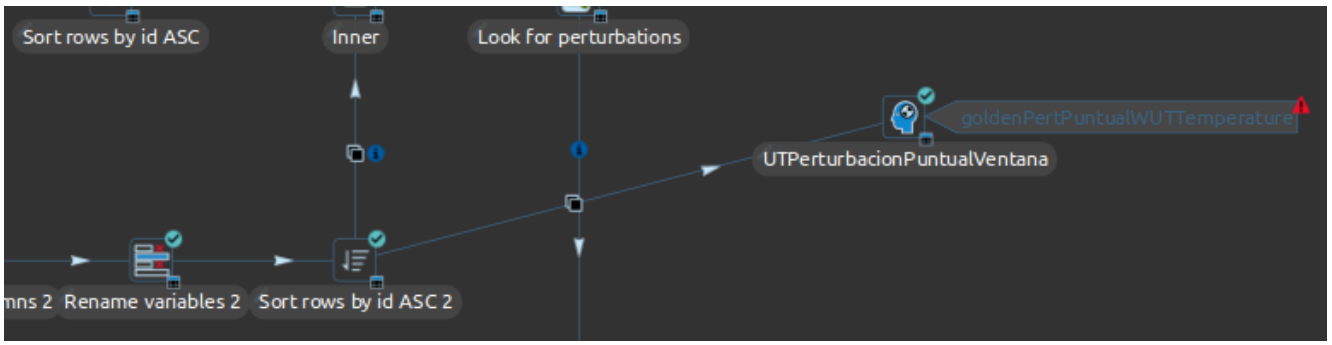
Tras completar cada ciclo de pruebas, se llevará a cabo una fase de revisión y corrección de defectos. Las pruebas de regresión se ejecutarán después de cada ciclo de corrección para validar que los cambios no han introducido nuevos errores.

Automatización y monitorización

Se utilizó la automatización para reducir el tiempo de feedback y mejorar la detección temprana de errores. El CI/CD integrado en GitHub permite monitorizar continuamente la salud del pipeline.

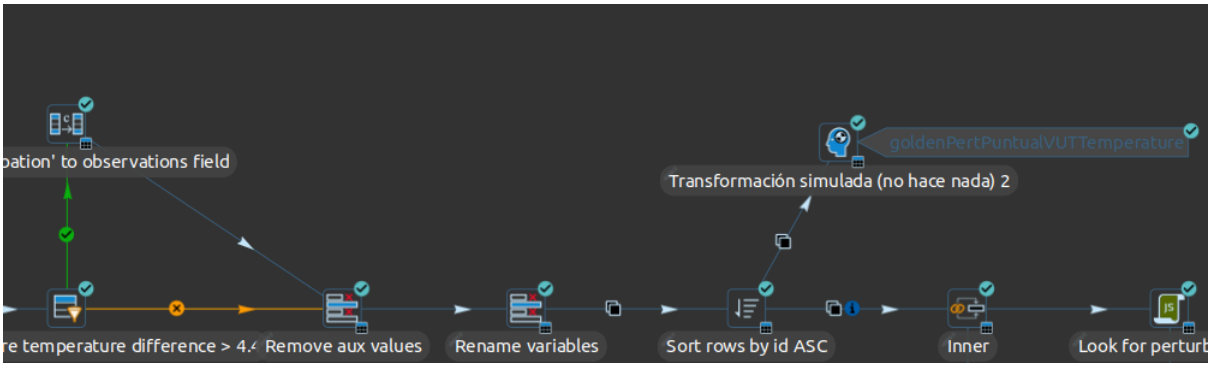
Evidencias

Se adjuntan algunas evidencias de pruebas, algunas son de falla, lo que permitió resolver problemas antes de que estén en producción.

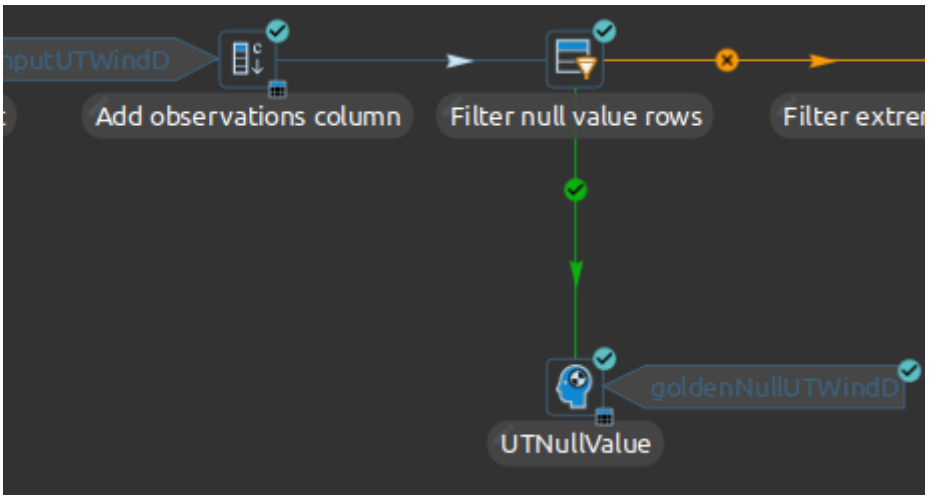


erature	UTPerturbacionPuntualVentana	Y	Validation against golden data failed for row number 2, field observations : transform value [] does not correspond to data set value [Perturbation]
erature	UTPerturbacionPuntualVentana	Y	Validation against golden data failed for row number 4, field observations : transform value [] does not correspond to data set value [Perturbation]
erature	UTPerturbacionPuntualVentana	Y	Validation against golden data failed for row number 6, field observations : transform value [Perturbation] does not correspond to data set value []
erature	UTPerturbacionPuntualVentana	Y	Validation against golden data failed for row number 8, field observations : transform value [] does not correspond to data set value [Perturbation]
erature	UTPerturbacionPuntualVentana	Y	Validation against golden data failed for row number 9, field observations : transform value [] does not correspond to data set value [Perturbation]
erature	UTPerturbacionPuntualVentana	Y	Validation against golden data failed for row number 11, field observations : transform value [] does not correspond to data set value [Perturbation]
erature	UTPerturbacionPuntualVentana	Y	Validation against golden data failed for row number 12, field observations : transform value [] does not correspond to data set value [Perturbation]

Unit test aplicado en un pipeline de Apache Hop que verifica que no se den perturbaciones si la ventana de tiempo es mayor. Falla



Unit test aplicado en un pipeline de Apache Hop que verifica que se etiqueten las perturbaciones puntuales. Aprobado.



Unit test aplicado en un pipeline de Apache Hop que verifica que las mediciones con valor null no continúen el procesamiento y no se almacenen. Aprobado.

C	D
Dirección	Nro lote insp
LAGO SAN ROQUE C1	900000144380
LAGO SAN ROQUE C2	900000144381
LAGO SAN ROQUE C4	900000144382
LAGO SAN ROQUE C5	900000144383
LAGO SAN ROQUE TAC1	900000144384
LAGO SAN ROQUE TAC2	900000144385
LAGO SAN ROQUE TAC4	900000144386
LAGO SAN ROQUE TAC5	900000144387
LAGO SAN ROQUE USINA	900000144388
LAGO SAN ROQUE DCQ1	900000144389
LAGO SAN ROQUE DSA1	900000144390
LAGO SAN ROQUE RIO COSQUIN	900000144391
LAGO SAN ROQUE LOS CHORRILLOS	900000144392
LAGO SAN ROQUE LAS MOJARRAS	900000144393
LAGO SAN ROQUE RIO SAN ANTONIO	900000144394
Estacion Metereológica LSR	900000144426

123 number	ABC code
144,380	C1
144,381	C2
144,382	C4
144,383	C5
144,384	TAC1
144,385	TAC2
144,386	TAC4
144,387	TAC5
144,388	USINA
144,389	DCQ1
144,390	DSA1
144,391	RCQ
144,392	RLC
144,393	RSA
144,394	RYU
144,426	EML1

Prueba Manual para calidad de agua donde se verifica que las muestras junto con el perfil enviado por ACSA(primer imagen), se almacenen correctamente en la base de datos (segunda imagen).

Despliegue

Para el despliegue se han realizado dos propuestas, una con servidor externo y otra con servidor local.

Servidor Externo (Hop Server)

Un servidor en la nube, utilizando Railway como proveedor de hosting.

Pros

- **Escalabilidad**: Opciones para escalar recursos fácilmente según las necesidades.
- **Mantenimiento**: Menor responsabilidad sobre el mantenimiento del hardware.
- **Disponibilidad**: Redundancias y políticas de respaldo que garantizan una alta disponibilidad. 99.95% SLA.
- **Actualizacione**s: Las actualizaciones y parches de seguridad suelen ser manejadas por el proveedor del servicio.
- **Acceso remoto**: Facilidad para acceder al servidor desde cualquier lugar.
- **Seguridad:** Las configuraciones de seguridad están manejadas por el proveedor, lo cual ahorra tener que desarrollarlo uno.

Contras

- **Costo:** Dependiendo del uso, los costos pueden incrementarse.
- **Latencia**: Puede haber latencia si los datos se encuentran en una ubicación geográfica distante al servidor en la nube.
- **Seguridad**: Dependencia del proveedor en términos de medidas de seguridad.
- **Personalización**: Puede haber limitaciones en cuanto a la personalización o configuración del servidor.

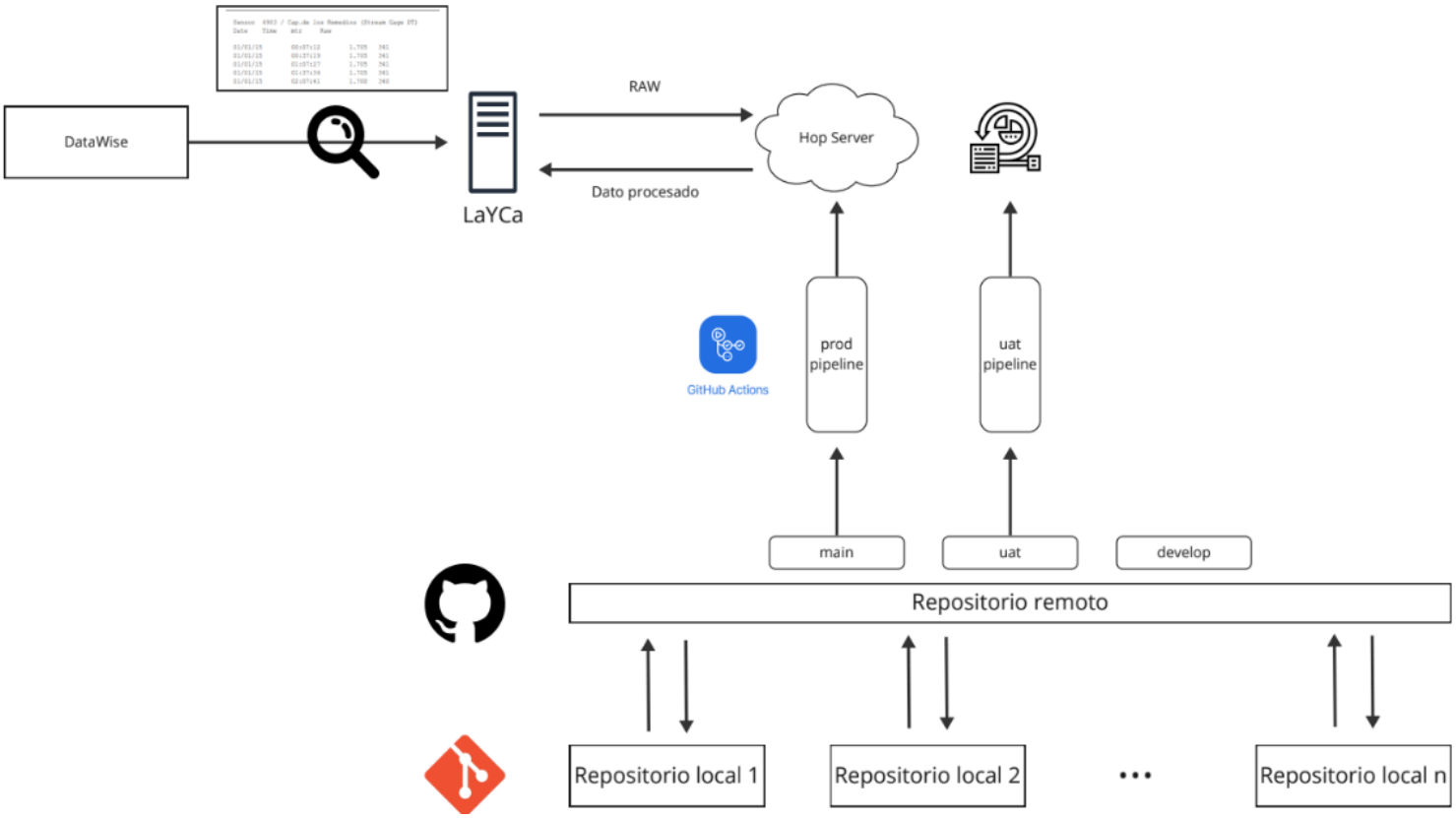


Diagrama de arquitectura de despliegue en servidor externo(railway).

Flujo paso a paso:

1. **DataWise**: Fuente de ingreso de datos reportados por los sensores. Los almacena en LaYCa por odbc.
2. **LaYCa**: Base de datos postgres.
3. **RAW**: Los datos crudos se envían al "Hop Server".
4. **Hop Server**: El ETL Apache Hop, tiene una versión en la nube, donde se encuentra script y se ejecuta. Y lo vuelve a almacenar en la base.
5. **GitHub Actions**: Es una herramienta de automatización de CI/CD integrada en GitHub. Ejecuta flujos de trabajo definidos por eventos en el repositorio, como push, pull requests, etc.
6. **Prod Pipeline y UAT Pipeline**: Son flujos de trabajo automatizados configurados en GitHub Actions. 'Prod pipeline' es para el despliegue en producción, mientras que 'UAT pipeline' se utiliza para el entorno de pruebas de aceptación del usuario. Estos pipelines se activan según los cambios realizados en las ramas correspondientes.

7. **Ramas del Repositorio Remoto**: Las ramas `main`, `uat` y `develop` representan diferentes etapas de desarrollo y despliegue. `main` se utiliza para producción, `uat` para pruebas de aceptación por los usuarios, y `develop` para desarrollo continuo y pruebas.
8. **Repositorios Locales**: Los desarrolladores tienen su propio clon del repositorio en su entorno local. Aquí hacen cambios y pruebas antes de sincronizar su trabajo (haciendo 'push') al repositorio remoto.

Este sistema opera completamente en la nube, con la sección de GitHub residiendo en la propia plataforma de GitHub. Dentro de este entorno, hay acciones específicas diseñadas para Railway, las cuales se activan durante la ejecución de los pipelines de integración y despliegue continuo (CI/CD). Estas acciones se encargan de actualizar tanto el hop server como la base de datos, las cuales son mantenidas como imágenes desplegadas en la plataforma de Railway.

Docker en Servidor On-Premise

Opción de servidor local, en la sede del INA en Villa Carlos Paz. Donde ya reside DataWise.

Pros

- **Control completo**: Control total sobre la configuración, seguridad y administración del servidor.
- **Rendimiento**: Puede ser más rápido si los datos están localizados en la misma red o infraestructura.
- **Sin costos ocultos**: Una vez adquirido el hardware, no hay costos adicionales recurrentes, salvo el mantenimiento.
- **Personalización**: Mayor flexibilidad para personalizar el entorno según las necesidades.
- **Seguridad**: Control total sobre las políticas y medidas de seguridad.

Contras

- **Mantenimiento**: Responsabilidad total sobre el mantenimiento del hardware y software.
- **Escalabilidad**: Escalar recursos puede requerir inversiones adicionales en hardware.
- **Actualizaciones**: Responsabilidad sobre la aplicación de actualizaciones y parches de seguridad.
- **Disponibilidad**: Se requiere una inversión adicional para garantizar alta disponibilidad y redundancia.

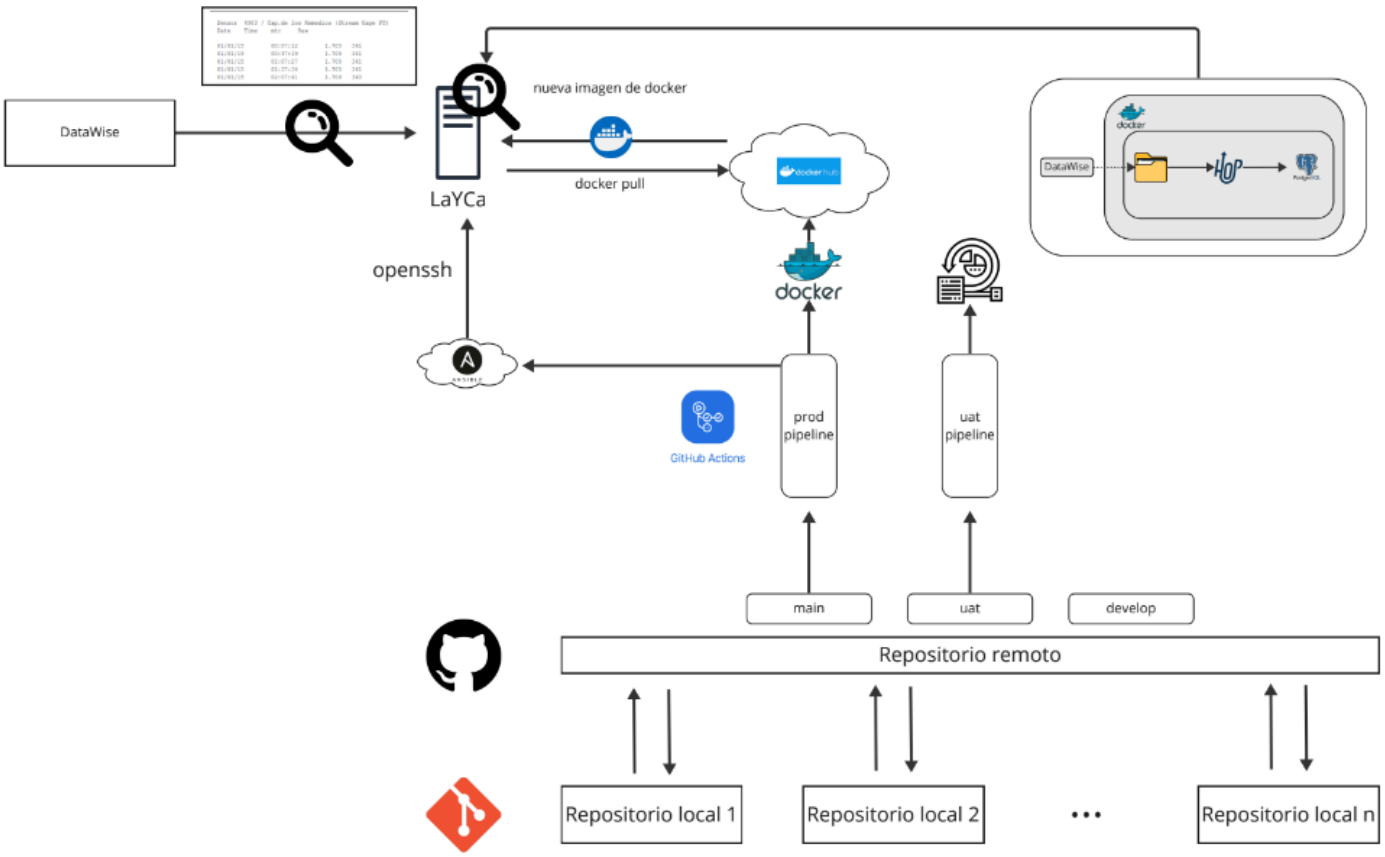


Diagrama de arquitectura de despliegue en servidor local.

Flujo paso a paso:

1. **DataWise**: Fuente de ingreso de datos reportados por los sensores. Los almacena en LaYCa por odbc.
2. **LaYCa**: Base de datos postgres en servidor local.
3. **OpenSSH**: Es una suite de software que proporciona comunicación segura a través de redes inseguras. Se usa para acceder de forma remota a máquinas y transferir archivos de forma segura.
4. **Nueva imagen de Docker**: Se crea una nueva imagen de Docker a partir de un archivo llamado Dockerfile donde se configura y especifica la imagen de la aplicación a crear y se almacena en un registro. Docker es una plataforma de contenedores que permite empaquetar una aplicación con todas sus dependencias en un contenedor estándar.
5. **Docker Pull**: Los desarrolladores o el sistema de integración continua pueden recuperar la nueva imagen de Docker del registro para uso local o en servidores de prueba/producción.

6. **GitHub Actions**: Es una herramienta de automatización que permite definir flujos de trabajo directamente en un repositorio GitHub. Estos flujos de trabajo pueden incluir acciones como construir, testear y desplegar aplicaciones.
7. **Prod Pipeline y UAT Pipeline**: Estos son flujos de trabajo (pipelines) de GitHub Actions para ambientes de producción y UAT (User Acceptance Testing). El código fuente se despliega a través de estos pipelines dependiendo de la rama en la que se trabaje: `main` para producción y `uat` para pruebas de aceptación del usuario.
8. **Ramas del Repositorio Remoto**: `main`, `uat`, y `develop` son ramas típicas de un repositorio de control de versiones como Git. `main` suele contener el código de producción, `uat` es para pruebas de aceptación, y `develop` es para desarrollo y pruebas preliminares.
9. **Repositorios Locales**: Cada desarrollador tiene una copia local del repositorio donde pueden trabajar de manera independiente. Una vez que están listos, pueden subir sus cambios (push) al repositorio remoto en la rama correspondiente.

El diagrama representa un flujo de trabajo para un sistema que utiliza un conjunto diverso de tecnologías, principalmente alojado en un servidor local en LaYCa. Mediante el uso de docker-compose, se instalan y gestionan múltiples imágenes de Docker necesarias para el funcionamiento del sistema, que incluyen Apache Hop, Postgres, Metabase, Kafka, Debezium, Zookeeper, SQLAlchemy, entre otros servicios y aplicaciones requeridas. Mientras que las aplicaciones y servicios se ejecutan en un entorno local o en un entorno privado de nube a través de contenedores Docker, el código fuente reside en los servidores de GitHub. GitHub también facilita el proceso de integración y despliegue continuo (CI/CD) mediante GitHub Actions. Esto permite configurar y mantener diferentes entornos de trabajo, como desarrollo, prueba y producción. GitHub Actions automatiza la actualización de imágenes de Docker, asegurando que las aplicaciones se mantengan al día con las últimas modificaciones del código fuente. Además, permite a los desarrolladores recuperar las imágenes de Docker existentes según sea necesario.

El INA ha decidido que el despliegue sea la segunda opción, es decir, en un servidor local y esa es la que se ha implementado.

Beneficios post-implementación

El sistema de monitoreo de calidad del agua y variables hidrometeorológicas implementado en el ESR generará una serie de beneficios clave que incluyen:

Mejora en la Gestión del Recurso Hídrico: La capacidad para monitorear en tiempo real la calidad del agua y las condiciones hidrometeorológicas permitirá una gestión más eficiente del embalse, optimizando el uso del agua para diversos propósitos y mejorando la sostenibilidad ambiental.

Impacto Económico Positivo: Se anticipa una reducción de los costos asociados a los análisis periódicos de calidad del agua gracias a la automatización y continuidad del monitoreo, beneficiando económicamente a la región al disminuir los costos operativos y de mantenimiento, y mitigando los impactos económicos negativos derivados de la deterioración de la calidad del agua en actividades como el turismo y la pesca.

Beneficios Sociales: La mejora en la calidad del agua impactará positivamente en la salud pública y el bienestar de las comunidades locales, al asegurar un suministro de agua potable más seguro y contribuir al desarrollo sostenible de la región.

Impacto Ambiental: La gestión eficaz de la eutrofización y otros problemas de calidad del agua contribuirá a la conservación de la biodiversidad acuática y terrestre, mejorando el equilibrio ecológico del embalse y sus alrededores.

Fomento de la Responsabilidad Social Universitaria (RSU): El proyecto demuestra un compromiso con la RSU al abordar necesidades sociales y ambientales a través de la innovación y la educación, promoviendo la justicia, la equidad social y el desarrollo humano sostenible.

En conjunto, estos beneficios reflejan el potencial transformador del sistema para la gestión del agua, el medio ambiente, y la comunidad, subrayando la importancia de las soluciones tecnológicas en la resolución de desafíos sociales y ambientales contemporáneos.

Impacto económico

Uno de los objetivos globales de este trabajo es: “Reducir los costos asociados a los análisis periódicos de calidad del agua al implementar un monitoreo continuo y automatizado de los datos.”

Este trabajo proporciona beneficios económicos, reduciendo al mínimo el recurso humano. Debido a su enfoque automatizado y de fácil uso y personalización.

Siendo el ESR, un lago artificial construido en 1888 con el fin de proveer agua para consumo humano, riego y energía hidroeléctrica, su calidad impacta directamente en sus objetivos, pudiendo perjudicar o mejorar lo económico.

Así también es uno de los centros turísticos más importantes del país con un extenso uso recreativo, recibiendo la localidad de Villa Carlos Paz aproximadamente unos 370.000 visitantes por quincena en verano según estimación oficial (Secretaría de Turismo de Villa Carlos Paz, 2015). La calidad de su agua se ha deteriorado con el tiempo y actualmente presenta un estado eutrófico avanzado (Instituto Nacional del Agua, 2019) generando serios problemas económicos (inversión en infraestructura sofisticada para potabilización del agua, remediación del lago y disminución del turismo, entre otros)

La limitación del uso del ESR a modo recreativo y la reducción en la actividad pesquera debido a la mala calidad del agua pueden tener repercusiones económicas negativas en la región. La disminución en el turismo y las actividades recreativas puede afectar los ingresos de las empresas locales, como hoteles, restaurantes y comercios, así como también reducir las oportunidades de empleo en el sector turístico y relacionado.

Impacto social

El ESR de la Provincia de Córdoba, construido en 1888, tiene su importancia como regulador de los aportes hídricos superficiales de la cuenca alta del Río Suquía y es una de las principales fuentes de abastecimiento de agua para la Ciudad de Córdoba para consumo humano y riego, como así también para la generación de energía hidroeléctrica de la región y recreación. El deterioro de las condiciones y calidad de las aguas de los ríos tributarios al embalse, junto a los aportes directos desde sus márgenes, a lo largo de décadas, condujeron al actual estado eutrófico del embalse. Asimismo, las variaciones en las condiciones climáticas e hidrológicas de la cuenca pueden afectar aún más este proceso. El conjunto de estos factores es capaz de producir serios inconvenientes en la calidad y estética del recurso, como así también en los procesos de potabilización.

Al implementar este sistema, permite tener visibilidad de los datos para la toma de decisiones que mejoren la gestión de la calidad del agua. Así también tener visibilidad rápida, clara y fiable del estado trófico del agua para que se sepa con qué calidad está llegando a las plantas potabilizadoras.

Así también al generar un sistema de gestión de datos hidrometeorológicos en tiempo real, se puede utilizar para más aplicaciones no solo de calidad de agua, como por ejemplo predecir sequías, inundaciones, crecidas de ríos.

Responsabilidad Social Universitaria (RSU)

Este apartado también fue solicitado para validar por la Secretaría de Proyección y Responsabilidad Social Universitaria de la UCC que la aplicación de este trabajo, certifique como RSU. La cual fue aprobada.

Se entienden como **PROGRAMAS DE RESPONSABILIDAD SOCIAL UNIVERSITARIA** a:

- Acciones que respondan a las necesidades de transformación de la sociedad, mediante el ejercicio de la docencia, la investigación y/o la proyección social, animadas por la búsqueda de la promoción de la justicia, la solidaridad, la equidad social y la promoción del desarrollo humano sustentable.
- De carácter estables en relación al grado de institucionalización adquirido, aseguramiento de la continuidad de las acciones iniciadas y la permanencia de los logros alcanzados.
- Cuyo destinatario externo sea sujeto de marginación, padezca situaciones de injusticia y/o carencias particulares.

- Que preste servicios o ejecute acciones de carácter gratuito no disponibles en el medio y/o a las cuales el destinatario externo no pueda acceder por su situación particular.
- Que permita la adquisición, ejercicio y/o consolidación de los contenidos académicos y de las áreas, cátedras y/o carreras involucradas.
- Que garantice la activa participación de alumnos, constituyéndose en un ejercicio académico-profesional validado institucionalmente.
- Que permita el enriquecimiento mutuo entre destinatario externo e interno, en términos no sólo académico-profesionales sino humanísticos, éticos y formativos.

En esta situación, la investigación y desarrollo del proyecto no solo aborda cuestiones técnicas y científicas, sino que también incorpora un enfoque de Responsabilidad Social Universitaria (RSU) de importancia crucial. La RSU es un componente integral de la misión de nuestra universidad, y este proyecto es un ejemplo concreto de cómo la formación y aplicación académica puede contribuir al bienestar de la comunidad y al desarrollo sostenible de la región.

Los siguientes puntos son aspectos RSU que contempla el trabajo:

- Contribución a la Gestión de la Calidad del Agua

El proyecto se centra en aplicar soluciones técnicas e informáticas para permitir a los especialistas y expertos en la materia, mejorar la calidad del agua en el ESR. Esto es esencial para garantizar la disponibilidad de agua de alta calidad para la comunidad, lo que directamente impacta en su consumo diario. Al proporcionar información precisa y en tiempo real sobre el estado del agua, el trabajo contribuye directamente a la gestión de la calidad del agua en la región.

- Carácter Estable y Continuidad de Acciones

El trabajo establece un sistema de gestión de datos sostenible y eficiente. Al desarrollar una plataforma tecnológica y una página web para visualizar y analizar datos en tiempo real, estás contribuyendo a la estabilidad y continuidad de la gestión de datos en el futuro. Esto asegura que las acciones iniciadas perduren y los logros alcanzados se mantengan a lo largo del tiempo.

- Destinatario Externo Vulnerable

La comunidad que se beneficia del proyecto incluye a aquellos que dependen del ESR para su suministro de agua, así como a quienes están expuestos a riesgos relacionados con la calidad del agua. Estas personas podrían considerarse sujetos de marginación o en situaciones de injusticia si no se abordan adecuadamente los problemas de calidad del agua. Por lo tanto el proyecto se centra en mejorar su calidad de vida y bienestar.

- Prestación de Servicios Gratuitos No Disponibles en el Medio

La plataforma desarrollada para la visualización de datos y la generación de alertas en tiempo real se ofrece de manera gratuita y no está disponible en el entorno local. Esto asegura que las personas no tengan barreras económicas para acceder a información crucial sobre la calidad del agua y su impacto en el consumo.

- Promoción de la Adquisición y Consolidación de Contenidos Académicos

El proyecto involucra la recopilación, procesamiento y análisis de datos, lo que implica una contribución significativa a la adquisición de conocimientos académicos en áreas como la minería de datos, la ingeniería de software y la tecnología de la información, con aplicación concreta en gestión de datos para investigación y estudios de calidad de agua. Además, proporciona una plataforma para que estudiantes y profesionales adquieran experiencia práctica en la mejora de la calidad del agua y su relación con el consumo.

- Participación Activa de Alumnos, Enriquecimiento Mutuo y Colaboración Interinstitucional

Este proyecto ha establecido una sólida colaboración con el Instituto Nacional del Agua, una institución clave en la gestión hídrica a nivel nacional. Esta colaboración demuestra el compromiso de nuestra universidad con la colaboración interinstitucional y la resolución de problemas del mundo real. Juntos, trabajamos en soluciones que benefician a la comunidad.

En resumen, el proyecto claramente se adapta a los criterios de los programas de Responsabilidad Social Universitaria (RSU). Contribuye a la sociedad, promueve la equidad social y la justicia, brinda servicios gratuitos a quienes lo necesitan, y enriquece tanto a la comunidad como a quienes están involucrados en él.

Impacto medioambiental

Aquí es importante recordar el concepto de eutrofización, que ya se ha definido en el marco teórico, dado que si el estado trófico del cuerpo de agua es de eutrofización, el impacto medioambiental será negativo. Por lo tanto, una de las consecuencias directas de implementar este sistema es que permita reducir dicho estado en los cuerpos de agua analizados. En consecuencia, el impacto medioambiental positivo asociado con la implementación y uso del sistema está directamente relacionado con los beneficios que representa tener un cuerpo de agua sin estado eutrófico.

Los beneficios son:

- Promoción de la biodiversidad: Al reducir la eutrofización, se restaura el equilibrio natural de los ecosistemas acuáticos, lo que favorece el florecimiento de una mayor diversidad de vida acuática, incluyendo peces y otras especies. Esto conlleva a la recuperación de los hábitats acuáticos y la preservación de especies en peligro.

- Conservación del hábitat: La disminución de las floraciones de fitoplancton y la formación de zonas muertas permite que los ecosistemas acuáticos recuperen su salud y funcionamiento normal. Esto mejora la calidad del agua, restaura la transparencia y la penetración de la luz solar, y fomenta el desarrollo de la vegetación acuática.

- Mejora de la calidad del suelo y la producción agrícola: Al reducir la contaminación del agua con nutrientes, se protegen las fuentes de agua utilizadas para la irrigación agrícola, lo que contribuye a la conservación del suelo y mejora la calidad de los cultivos. Esto promueve prácticas agrícolas más sostenibles, aumenta la productividad agrícola y garantiza la seguridad alimentaria.

- Mitigación del cambio climático: Los ecosistemas acuáticos saludables desempeñan un papel importante en la captura y almacenamiento de carbono. Al preservar y restaurar estos hábitats mediante la reducción de la eutrofización, se contribuye a la mitigación del cambio climático al mantener los sumideros de carbono naturales.

Conclusiones

En el transcurso de este trabajo final, tuvimos la oportunidad de consolidar y aplicar una amplia gama de conocimientos adquiridos en las áreas fundamentales de la Ingeniería en Sistemas, los cuales han constituido el núcleo de nuestro aprendizaje a lo largo de la carrera. Este proyecto exigía una comprensión y aplicación integral de las competencias propias de un Ingeniero en Sistemas, abarcando desde metodologías de trabajo hasta análisis de sistemas, pasando por ingeniería de software, arquitectura de redes, gestión de bases de datos, implementación de sistemas inteligentes y desarrollo web, entre otros aspectos.

La naturaleza interdisciplinaria de los objetivos del proyecto nos permitió no solo poner en práctica nuestras habilidades técnicas, sino también desarrollar una visión holística del proceso de diseño e implementación de soluciones en el ámbito de la ingeniería en sistemas. Esta experiencia ha reforzado nuestra capacidad para abordar problemas complejos de manera eficaz, integrando conocimientos de diversas áreas para desarrollar soluciones innovadoras y eficientes.

Este proyecto nos permitió comprender que las soluciones informáticas requieren una visión integral, abordando el problema desde múltiples perspectivas. Esto subraya la importancia de contar con una formación amplia y diversificada como profesionales de la ingeniería en sistemas. Asimismo, destaca la necesidad de colaborar en equipos multidisciplinarios compuestos por especialistas en distintas áreas, para diseñar e implementar soluciones efectivas y completas.

Además, este proyecto nos permitió identificar la creciente demanda de profesionales en Ingeniería en Sistemas en distintas disciplinas. Enfrentamos retos típicos del entorno profesional, incluyendo desafíos en estimaciones de tiempo, definición de alcances, trámites burocráticos y la gestión de expectativas de los stakeholders. Particularmente, dedicamos un tiempo considerable, mayor al anticipado, al proceso de análisis y adaptación de sistemas preexistentes para el desarrollo de nuevas soluciones. Este esfuerzo adicional resultó en desafíos para cumplir con todos los requerimientos dentro del plazo previsto de la manera más óptima.

Enfrentamos otro desafío significativo al integrarnos en un equipo interdisciplinario, compuesto por profesionales no familiarizados con los sistemas que estábamos proponiendo. Este escenario nos exigió, como ingenieros en sistemas, el desarrollo y la aplicación de habilidades blandas cruciales para facilitar una comunicación efectiva. Fue esencial para nosotros no solo "hablar un mismo idioma", sino también explicar de manera comprensible nuestros conceptos técnicos y entender sus necesidades específicas. Esta

dinámica fue clave para captar adecuadamente sus requerimientos y asegurar una colaboración productiva, especialmente durante las presentaciones realizadas al equipo de científicos del INA, donde tuvimos que esforzarnos por hacer comprensible el alcance y los detalles de nuestro proyecto.

Sin embargo, lo que más resaltamos es la importancia crítica del conocimiento técnico y la selección adecuada de tecnologías y herramientas. Esta experiencia reafirmó nuestra convicción sobre el valor real y la aplicabilidad profesional de lo aprendido durante nuestra carrera para satisfacer eficazmente las necesidades presentadas. Contribuir de manera efectiva a satisfacer los requerimientos de una institución nacional tan crucial como el INA ha sido enormemente gratificante para nosotros. Esta oportunidad no solo nos permitió aplicar nuestros conocimientos y habilidades en un contexto real y significativo, sino que también reforzó nuestra comprensión de la relevancia de nuestra formación en ingeniería en sistemas para el avance y el bienestar de la comunidad.

Bibliografía

https://intelequia.com/blog/post/ciclo-de-vida-del-software-todo-lo-que-necesitas-saber

https://www.ilimit.com/blog/integracion-continua-entrega-continua-despliegue-continuo/

https://www2.ucc.edu.ar/vida-ucc/vida-universitaria/programas-de-proyeccion-social-y-rsu/

https://aws.amazon.com/es/what-is/sdlc/

https://evotic.es/software-a-medida/ciclo-de-vida-del-software/

Lucas Ledesma. Apuntes de clase, Ingeniería en Software 3, UCC, año 2022.

https://kafka.apache.org/documentation/

https://debezium.io/documentation/reference/stable/connectors/postgresql.html

https://hop.apache.org/manual/latest/index.html

https://www.oracle.com/ar/database/what-is-a-relational-database/

Tesis de Maestría del Ing. Pablo Facundo Andreoni:
https://datamining.dc.uba.ar/datamining/Files/Tesis/Tesis_Pablo_Andreoni.pdf

Paper del INA sobre Caracterización de la calidad de agua y de variables meteorológicas relacionadas con eventos extremos de floración en el Embalse San Roque:
https://www.ina.gov.ar/archivos/publicaciones/2018_Pussetto%20et%20al_Caracterizaci%C3%B3n%20Calidad%20Agua%20Eventos%20Extremos.pdf

Paper del INA sobre Sistema de Gestión de Alertas en INA-CIRSA:
https://www.ina.gob.ar/ifrh-2016/trabajos/IFRH_2016_paper_82.pdf

Amé, V., Ferral, A., & Solís, V. (2017). Eutrofización en el Embalse San Roque y floraciones masivas de cianobacterias. Seguimiento por técnicas geoespaciales. Unciencia.

Colladón, L. (2018). Evaluación del desempeño de los sensores de nivel, a través del registro de pios de crecidas.

Colladón, L., & Vélez, E. (2011). Sistema de Monitoreo Automático EE Rios en las Sierras de Córdoba. Artículo presentado en Memorias del Quinto Simposio Regional sobre Hidráulica de Ríos, Santiago del Estero.

Rodríguez, M. I., Cossavella, A., Oroná, C., Larrossa, N., Avena, M., Rodriguez, A., Del Olmo, S., Bertucci, C., Muñoz, A., Castelló, E., Bazán, R., & Martínez, M. (2000). Estudios Preliminares de la Calidad de Agua y Sedimentos del Embalse San Roque relacionados al proceso de Eutroficación. Artículo presentado en XVIII Congreso Nacional del Agua, Santiago del Estero.

Roldán Pérez, G., & Ramírez Restrepo, J. J. (2008). Fundamentos de Limnología Neotropical. Medellín: Editorial Universidad de Antioquía.