

# GENERACIÓN Y GESTIÓN DE DATOS HIDRO- METEOROLÓGICOS Y ESCENARIOS DE CAMBIO CLIMÁTICO REPÚBLICA DOMINICANA

## Producto 3

Plataforma CDMS (Sistema de  
Gestión de Datos Climáticos)  
para la República Dominicana

Trabajo realizado por:



Fecha: 24 de abril de 2023



# I RESUMEN EJECUTIVO

El objetivo del proyecto es desarrollar un Sistema de Gestión de Datos Climáticos (CDMS, por sus siglas en inglés) personalizado para la República Dominicana, que cumpla las normas internacionales y las necesidades locales específicas. Para ello, se parte de un análisis detallado de las normas exigidas por la Organización Meteorológica Mundial (OMM) y de las plataformas básicas existentes -identificadas como normas de referencia por el anuncio del proyecto- para identificar las características clave que deben integrarse en la plataforma CDMS dominicana.

Además, se lleva a cabo un análisis exhaustivo de las mejores plataformas de código abierto disponibles y adheridas a las normas internacionales de referencia, con el fin de identificar aquellas que potencialmente podrían utilizarse como punto de partida para el desarrollo de futuras plataformas. Se toma en cuenta la compatibilidad entre estas plataformas y los requisitos específicos de la República Dominicana, así como las normas tecnológicas y reglamentarias pertinentes. El objetivo final es la definición de una estructura de base de datos adecuada y una capa de Interfaz de Programación de Aplicaciones (API, por sus siglas en inglés) versátil y eficiente, que permita gestionar de forma eficaz y pertinente los requisitos del escenario dominicano.

Por último, el proyecto prevé la definición de desarrollos futuros, tanto imprescindibles como sugeridos, para que la plataforma CDMS sea versátil, esté preparada para el futuro y pueda ofrecer una experiencia de usuario completa y agradable.

El presente documento contiene los detalles del análisis realizado, así como el diseño de la plataforma CDMS, altamente adaptada a las necesidades específicas de la República Dominicana, incluyendo el plan de inversión para su implementación.

El documento es complementado en su parte inicial con una síntesis para tomadores de decisiones, que describe de manera resumida y llana todos los componentes de la propuesta.

## II            **ACRÓNIMOS**

<b>AI:</b>	Inteligencia Artificial
<b>ALB:</b>	Equilibrador de Carga de Aplicaciones
<b>API:</b>	Interfaz de Programación de Aplicaciones
<b>AWS:</b>	Servicios Web de Amazon
<b>BYU:</b>	Universidad Brigham Young
<b>CDMS/SGDC:</b>	Sistema de Gestión de Datos Climáticos
<b>CI/CD:</b>	Integración Continua/Implantación Continua
<b>DB:</b>	Base de Datos
<b>SGBD:</b>	Sistema de Gestión de Bases de Datos
<b>DDoS:</b>	Denegación de Servicio Distribuida
<b>DoS:</b>	Denegación de Servicio
<b>e2e:</b>	De Extremo a Extremo
<b>ECR:</b>	Registro Elástico de Contenedores
<b>EFS:</b>	Sistema de Archivos Elástico
<b>EKS:</b>	Servicio Kubernetes Elástico
<b>ET-CDMS:</b>	Equipo de Expertos en Sistemas de Gestión de Datos Climáticos
<b>FTP:</b>	Protocolo de Transferencia de Archivos
<b>GCOS:</b>	Sistema Mundial de Observación del Clima
<b>SIG:</b>	Sistemas de Información Geográfica
<b>GISC:</b>	Centros del Sistema Mundial de Información de la OMM
<b>GTS:</b>	Sistema Mundial de Telecomunicaciones
<b>IaC:</b>	Infraestructura como Código
<b>IoT:</b>	Internet de los Objetos
<b>JWT:</b>	Token Web JSON
<b>KPI:</b>	Indicadores Clave de Rendimiento
<b>LDC:</b>	Países menos Desarrollados
<b>MQ:</b>	Cola de Mensajes
<b>NHMS:</b>	Servicios Meteorológicos e Hidrológicos Nacionales
<b>POC:</b>	Prueba de Concepto
<b>REST:</b>	Transferencia de Estados Representativos

<b>SFTP:</b>	Protocolo Seguro de Transferencia de Archivos
<b>SOAP:</b>	Protocolo Simple de Acceso a Objetos
<b>SQL:</b>	Lenguaje de Consulta Estructurado
<b>IU:</b>	Interfaz de Usuario
<b>UML:</b>	Lenguaje Unificado de Modelización
<b>UX:</b>	Experiencia del Usuario
<b>VPC:</b>	Nube Privada Virtual
<b>WAF:</b>	Cortafuegos de Aplicaciones Web
<b>WIGOS:</b>	Sistema Mundial Integrado de Observación de la OMM
<b>WIS:</b>	Sistema de Información de la OMM
<b>WIS 2.0:</b>	Sistema de Información de la OMM 2.0
<b>WMO/OMM:</b>	Organización Meteorológica Mundial

# **III RESUMEN PARA TOMADORES DE DECISIONES**

## **III.A OBJETIVOS, CRITERIOS Y METODOLOGÍA**

El proyecto contempla la creación de un punto de referencia entre las tres plataformas definidas como los principales sistemas de referencia en la convocatoria del proyecto. Estas tres plataformas son un punto de referencia para identificar los requisitos mínimos de funcionalidad e interacción que la futura plataforma CDMS de la República Dominicana deberá cumplir necesariamente.

Además, el proyecto define una estructura de base de datos adecuada a las necesidades específicas del país, mediante el uso de las tecnologías más recientes y eficientes, junto con una capa de API versátil, completa y eficiente.

Por tanto, el punto de partida ha sido la definición de requisitos funcionales que, por un lado, respondieran a los criterios mínimos solicitados y, por el otro, cubrieran algunas funcionalidades necesarias en el escenario específico de la República Dominicana, donde la propiedad de las estaciones hidrometeorológicas está dividida entre diversas entidades productoras de datos.

Como punto de partida para el desarrollo futuro, se ha identificado una plataforma CDMS existente afiliada al proyecto OpenCDMS (respaldado y patrocinado por la OMM), que se utilizará como base para la construcción de la futura plataforma CDMS de la República Dominicana. La elección de OpenCDMS responde al objetivo de establecer e implementar las mejores prácticas y estándares de gestión de datos globales para el manejo de datos existentes y futuros, adhiriendo a estándares internacionalmente recomendados y auspiciados por la autoridad máxima a nivel global en tema hidrometeorológico.

Por lo tanto, con referencia entre las soluciones candidatas, la elección final propuesta está basada en parámetros tecnológicos, accesibilidad, personalización y cumplimiento de los estándares de referencia internacionales. El código abierto ha sido considerado como un criterio fundamental para la selección, debido a la capacidad intrínseca del software de código abierto para ser personalizado y adaptado a las necesidades específicas de un proyecto de manera más fácil que el software de código cerrado.

En la fase de análisis se estableció, además, con éxito una relación de colaboración con los equipos principales de los proyectos estudiados, lográndose un diálogo fructífero con los equipos de OpenCDMS y SURFACE.

Como último paso, se realizó un análisis de la plataforma seleccionada para determinar las áreas que requieren mejoramiento y con qué prioridad. Se establecieron dos niveles de prioridad de desarrollo: imprescindibles y opcionales. Los desarrollos imprescindibles u obligatorios se consideran indispensables y no negociables. Los desarrollos opcionales se dividen en módulos autónomos, cada uno independiente de los demás y con potencial de ser desarrollado e implementado por separado, sin comprometer la funcionalidad del sistema.

### III.B PLATAFORMAS DE REFERENCIA

Este análisis surge de las solicitudes de los entregables del proyecto y tiene como objetivo describir las funcionalidades ofrecidas por tres plataformas definidas como términos de comparación mínimos e imprescindibles: las de Colombia, Chile y México. Junto a estas tres plataformas se encuentra la plataforma nacional de Belice, identificada como una opción adicional más que satisfactoria por cumplir con todos los requisitos mínimos de entrega. El propósito de este análisis es crear una base de partida que defina las características mínimas con las cuales la plataforma CDMS de la República Dominicana debe contar, para diseñar el sistema en base a ellas.

La tabla a continuación resume el análisis comparativo entre las plataforma de los cuatro países analizados, en base a tres niveles de cumplimiento (verde = “disponible”, amarillo = “parcialmente disponible”, rojo = “no disponible”).

**Cuadro IIIA. Benchmark de plataformas existentes**

	COLOMBIA	CHILE	MÉXICO	BELICE
Mapa interactivo	●	●	●	●
Filtros por estación	●	●	●	●
Filtros por variable	●	●	●	●
Filtros de tiempo	●	●	●	●
Descargar/Exportar datos	●	●	●	●
Gráficos de datos por estación	●	●	●	●
Gráficos de datos por variable	●	●	●	●
Visualización de datos estadísticos	●	●	●	●

	COLOMBIA	CHILE	MÉXICO	BELICE
Visualización de datos estadísticos en el mapa	●	●	●	●
Análisis espacial (Heatmap)	●	●	●	●
Lista de estaciones	●	●	●	●
Comparación de estaciones	●	●	●	●

El detalle del benchmark entre las plataformas analizadas se encuentra en los capítulos 2 y 3 del documento.

### III.C REQUISITOS FUNCIONALES

El análisis de los requisitos funcionales requiere tomar en cuenta el rol del usuario final de los servicios climáticos, así como la entidad productora de datos hidrometeorológicos y el de la entidad administradora de la plataforma.

En el caso de la entidad productora, esta debe poder gestionar de forma autónoma sus propias estaciones, sin contar con las autorizaciones necesarias para gestionar estaciones de las demás entidades. En cuanto a la entidad administradora, las necesidades están relacionadas con la capacidad de gestionar tanto las entidades como las estaciones, independientemente de la entidad a la que pertenezcan. El sistema debe garantizar funcionalidades específicas para cada rol involucrado, de manera que se satisfagan las necesidades de gestión de las estaciones por parte de las entidades productoras y el control y monitoreo transversal por parte de la entidad administradora.

La tabla a continuación presenta la lista de los requisitos funcionales (RF) asociados a una plataforma CDMS.

**Cuadro IIIB. Cuadro resumen de los requisitos funcionales (RF) sugeridos (cada RF es numerado según el modelo RF-XX).**

Nombre	Descripción
RF-01	El sistema debe presentar un mapa con indicadores de las estaciones climáticas instaladas.
RF-02	El sistema debe permitir al usuario interactuar con el mapa a través de los filtros adecuados.

Nombre	Descripción
RF-03	El sistema debe permitir al usuario visualizar datos estadísticos directamente en el mapa.
RF-04	El sistema debe presentar una lista de estaciones climáticas.
RF-05	El sistema deberá permitir al usuario examinar los datos relativos a cada estación.
RF-06	El sistema debe permitir al usuario analizar datos y estadísticas relativas a una estación.
RF-07	El sistema debe permitir al usuario analizar datos y estadísticas relativas a una variable.
RF-08	El sistema debe ser capaz de realizar elaboraciones gráficas basadas en datos y estadísticas.
RF-09	El sistema deberá permitir al usuario comparar diferentes estaciones y/o variables.
RF-10	El sistema debe proporcionar un mapa dinámico que permita al usuario crear heatmaps.
RF-11	El sistema debe permitir que el usuario pueda descargar/guardar datos climáticos en su dispositivo.
RF-12	El sistema debe permitir a las entidades productoras de datos hidrometeorológicos insertar nuevas estaciones/plataformas.
RF-13	El sistema debe permitir a las entidades productoras de datos hidrometeorológicos modificar sus estaciones/plataformas.
RF-14	El sistema debe permitir que las entidades productoras de datos hidrometeorológicos eliminen sus estaciones/plataformas.
RF-15	El sistema debe permitir a las entidades productoras de datos hidrometeorológicos definir y aplicar umbrales de control de calidad sobre los datos relevantes para sus estaciones/plataformas.
RF-16	El sistema debe permitir a las entidades productoras de datos hidrometeorológicos importar datos, tanto nuevos como históricos.
RF-17	El sistema debe permitir al gobierno/gestor ingresar entidades productoras de datos hidrometeorológicos y/o estaciones/plataformas.
RF-18	El sistema debe permitir al gobierno/operador modificar entidades productoras de datos hidrometeorológicos y/o estaciones/plataformas.
RF-19	El sistema debe permitir al gobierno/operador eliminar entidades productoras de datos hidrometeorológicos y/o estaciones/plataformas.

La descripción detallada de los requisitos funcionales es contenida en el capítulo **3** del documento.



### III.D PLATAFORMAS ABIERTAS DE REFERENCIA

El objetivo de este benchmark es proporcionar al lector una visión general detallada de tres plataformas CDMS que forman parte del proyecto OpenCDMS. La razón de elegir estas tres plataformas (MCH, SURFACE y Climsoft) responde a lo siguiente: MCH ya se utiliza en la República Dominicana (la ONAMET está implementándola y el INDRHI tiene el interés en adherir a ella), mientras que SURFACE y Climsoft son las dos alternativas de código abierto con características más adecuadas para ser analizadas de manera comparativa para identificar la respuesta a las necesidades nacionales.

El análisis comparativo entre MCH, SURFACE y Climsoft es realizado en función de las características más importantes para una plataforma de gestión de datos climáticos. Según los objetivos y criterios definidos, **SURFACE resulta ser la mejor opción como punto de partida para el desarrollo de la futura plataforma CDMS para la República Dominicana**, puesto que cuenta con puntos fuertes notables, tanto por la tecnología utilizada como por fundamentarse en software de código abierto, esencial si se desea realizar personalizaciones adaptadas al contexto nacional.

El cuadro a continuación resume el análisis comparativo de las características de las tres plataformas analizadas.

**Cuadro IIIC. Resumen del análisis de las características de las plataformas analizadas**

	MCH	SURFACE	Climsoft
Integración con OpenCDMS	Programado: lanzamiento no programado con precisión	Planificado: lanzamiento inicial programado para abril de 2023	Planificado: lanzamiento inicial programado para mayo de 2023
Integración con WIS 2.0 y wis2box	No en desarrollo: disponible más adelante a través de la integración con OpenCDMS	En desarrollo y también disponible más adelante a través de la integración con OpenCDMS	No en desarrollo: disponible más adelante a través de la integración con OpenCDMS
Integración de los metadatos del WIGOS	Parcial: disponible más adelante a través de la integración con el proyecto OpenCDMS	Integración parcial – total en desarrollo	Parcial: disponible más adelante a través de la integración con el proyecto OpenCDMS
Disponibilidad	Aplicación de escritorio de Windows	Visor de web	Aplicación de escritorio de Windows
Licencia y accesibilidad	Código cerrado	Código abierto – licencia MIT	Código abierto – licencia MIT GPL3
Tecnología de base de datos	MySQL 5.1 & MyISAM	TimescaleDB	PostgreSQL & PostGIS o MySQL 8.x y MariaDB
Lenguaje de programación	Delphi/Object Pascal	Python	Python
Versionado de software	No	Sí	Sí

	MCH	SURFACE	Climsoft
Hoja de ruta	No disponible	Disponible y respetado	Disponible pero no respetado
Permisos y aislamiento de datos	Sistema de permisos implementado (con método desconocido), aislamiento de datos no disponible	Sistema de permisos implementado (a través de Django), sin aislamiento de datos disponible	Sistema de permisos implementado (a través de Django), sin aislamiento de datos disponible
Consultas de bases de datos de terceros sin archivar	No implementada	No implementada	No implementada

Informaciones detalladas sobre la evaluación comparativa de las tres plataformas analizadas están contenidas en el capítulo 4 del presente documento.

### **III.E PLATAFORMA SUGERIDA Y ESCENARIOS DE EVOLUCIÓN**

En todo el documento se detallan y justifican las razones detrás de la elección de SURFACE como sistema base para el desarrollo de la plataforma CDMS dominicana. Sin embargo, para fines de garantizar la implementación de una solución óptima para el contexto de la República Dominicana, son necesarios desarrollos adicionales, algunos de los cuales son imprescindibles u obligatorios, mientras que otros son considerados opcionales.

Respecto a los desarrollos obligatorios, la primera evolución necesaria desde SURFACE hacia la futura plataforma CDMS dominicana es el diseño e implementación de un sistema efectivo y relevante de aislamiento de datos, junto con las correspondientes autorizaciones. Se prevé implementar este sistema aprovechando el modelo de datos preexistente de SURFACE y enriqueciéndolo adecuadamente. Después de este desarrollo, será fácil implementar un sistema adicional de autorizaciones que permita a cada entidad gestionar sus propios datos sin generar conflictos, manteniendo el más alto nivel de seguridad e integridad de los datos.

Otra funcionalidad importante que falta en la actual capa API de SURFACE es la capacidad de importar datos hidrometeorológicos. La implementación de esta función abre varias posibilidades para la inserción de datos, gracias a la capacidad del API de comunicarse con otros componentes de software independientemente del lenguaje de programación utilizado para crearlos. La importación de datos es crucial para la plataforma, ya que no solo permite la inserción de nuevos datos recopilados, sino que

también brinda la oportunidad de ingresar datos históricos de los cuales los entes productores disponen.

Algunos desarrollos futuros de la plataforma CDMS dominicana, aunque no sean considerados imprescindibles, sí son sugeridos, con el objetivo de asegurar que la plataforma esté a la vanguardia tecnológica y sea lo más atractiva posible para cualquier usuario potencial, adaptándose a las evoluciones. Los desarrollos futuros opcionales se dividen en módulos autónomos, independientes entre sí y por lo tanto implementables por separado según las necesidades y consideraciones del cliente. Los módulos autónomos sugeridos son los siguientes:

- Módulo de funcionalidades de usuario: un módulo que implementa funcionalidades orientadas a la calidad de vida del usuario, como la posibilidad de guardar filtros y búsquedas, y gestionar accesos directos personalizados.
- Investigación y desarrollo (I+D): una propuesta de inversión que busca salvaguardar la integridad y precisión de los datos mediante investigaciones sobre el uso de algoritmos estadísticos vinculados a soluciones de vanguardia, como la teoría de juegos o la inteligencia artificial (IA).
- Módulo de UI/UX: que incluye una interfaz de usuario diseñada para incorporar todas las mejoras derivadas de la fase de investigación, así como desarrollar funciones y diseños que se ajusten mejor a los estándares contemporáneos. Este módulo incluye un nuevo diseño para dispositivos móviles y funciones de accesibilidad.

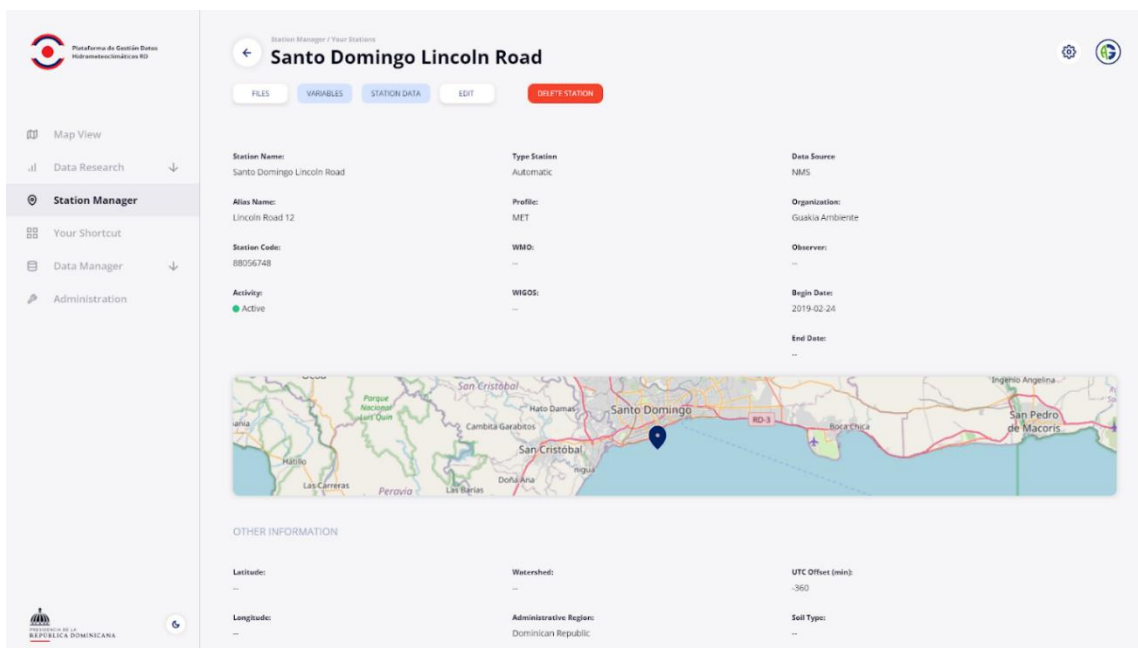


Figura IIIA. Gestión de estaciones con el módulo UX/UI personalizado

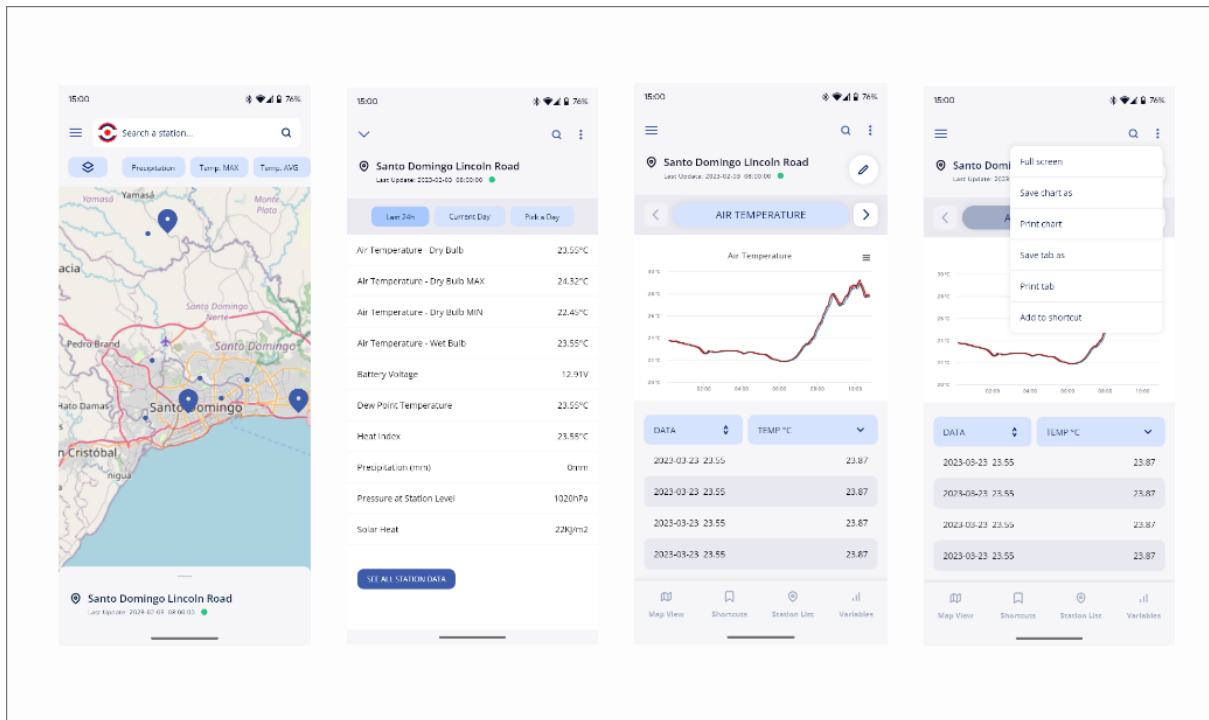


Figura IIIB. Diseño móvil con módulo UX/UI personalizado

En base al análisis detallado del contexto dominicano, las opciones presentadas son evaluadas como las más adecuadas para alcanzar los objetivos establecidos y garantizar la satisfacción de los diferentes usuarios de la plataforma CDMS en el país.

El detalle de la plataforma, incluyendo el análisis realizado, se encuentre en el capítulo 5.

### III.F ARQUITECTURA E INFRAESTRUCTURA

La arquitectura base del producto presenta tres capas principales interoperables:

- Capa de front-end: desarrollada en Angular.
- Capa de back-end: basada en la capa de back-end de Surface modificada adecuadamente, a su vez construida utilizando el framework Django.
- Capa de datos: desarrollada utilizando TimescaleDB.

Las tres capas de la arquitectura están estrechamente interconectadas y colaboran para proporcionar una experiencia de uso completa y funcional. En concreto, una vez que la aplicación Angular se distribuye a los clientes, es responsable de acceder y modificar los datos a través de la capa de API presente en el back-end. De hecho, la capa de back-end, utilizando el componente de modelo proporcionado por el framework Django, es la única responsable de acceder directamente a los datos.

Los administradores del sistema podrán acceder a las funcionalidades de administración a través de sus clientes de escritorio. Estas funcionalidades serán expuestas por el módulo Django Admin y, si es necesario, el módulo puede ser externalizado y dockerizado de forma separada. Cada capa de la arquitectura base se dockerizará por separado y contribuirá a la configuración física general de la arquitectura.

La arquitectura de la infraestructura presentada es cloud-agnostic y de código abierto, a excepción de algunos servicios específicos. La arquitectura cloud-agnostic permite la flexibilidad y portabilidad de los sistemas de software en diferentes plataformas en la nube, reduciendo el riesgo de dependencia de una sola plataforma.

Se eligió AWS como proveedor de servicios en la nube debido a su confiabilidad, amplia gama de servicios, herramientas de automatización y orquestación, herramientas de seguridad, opciones de precios flexibles y esfuerzos para la sostenibilidad ambiental. Aunque la infraestructura es compatible con la plataforma propuesta, muchas de las características y pautas seguirían siendo válidas para una infraestructura alternativa.

El detalle sobre la estructura de la infraestructura puede ser consultado en el capítulo 8.

### III.G PLAN DE INVERSIÓN

El plan de inversión presentado en el cuadro siguiente (en USD) excluye los costos de integración con plataformas y/o sistemas de TI existentes y relacionados con los entes productores de datos, puesto que la estimación de estos costos depende en gran medida del análisis que puede ser realizado solamente en fase de implementación, para determinar la posibilidad de acceder a las plataformas a ser integradas, una vez que se haya evaluado la necesidad de integración para cada caso.

**Cuadro IIID. Cuadro resumen de costos de la hipótesis sugerida en USD.**

DESARROLLO	COSTO (USD)
BACK-END Y CAPA API	50.625,00
FRONT-END PERSONALIZADO	98.700,00
CONSTRUCCIÓN DE INFRAESTRUCTURA	85.000,00
INVESTIGACIÓN Y DESARROLLO	40.000,00
SISTEMA DE REGISTRO DE USUARIO PÚBLICO	15.000,00
INTEGRACIÓN ENTRE DJANGO Y COGNITO	10.000,00

CREACIÓN DE GUÍAS DE USUARIO	12.000,00
CREACIÓN DE CAPACITACIÓN PARA USUARIOS	5.000,00
EJECUCIÓN DE LA PRUEBA DE PENETRACIÓN	15.000,00
<b>TOTAL</b>	<b>331.325,00</b>

Los costos indicados en el cuadro anterior deben ser complementados con los costos de ejecución de la infraestructura (calculados automáticamente con AWS Calculator), que ascienden a un costo único de **249,66 USD** (lo que eleva el costo total inicial a **331.574,66 USD**) y un costo de ejecución mensual de **2.457,00 USD** (equivalente a un costo de ejecución anual de **29.484,00 USD**).

El plan de inversión detallado se encuentra en el capítulo 9.

### **III.H HOJA DE RUTA, ENTREGABLES Y PLAN DE TRABAJO**

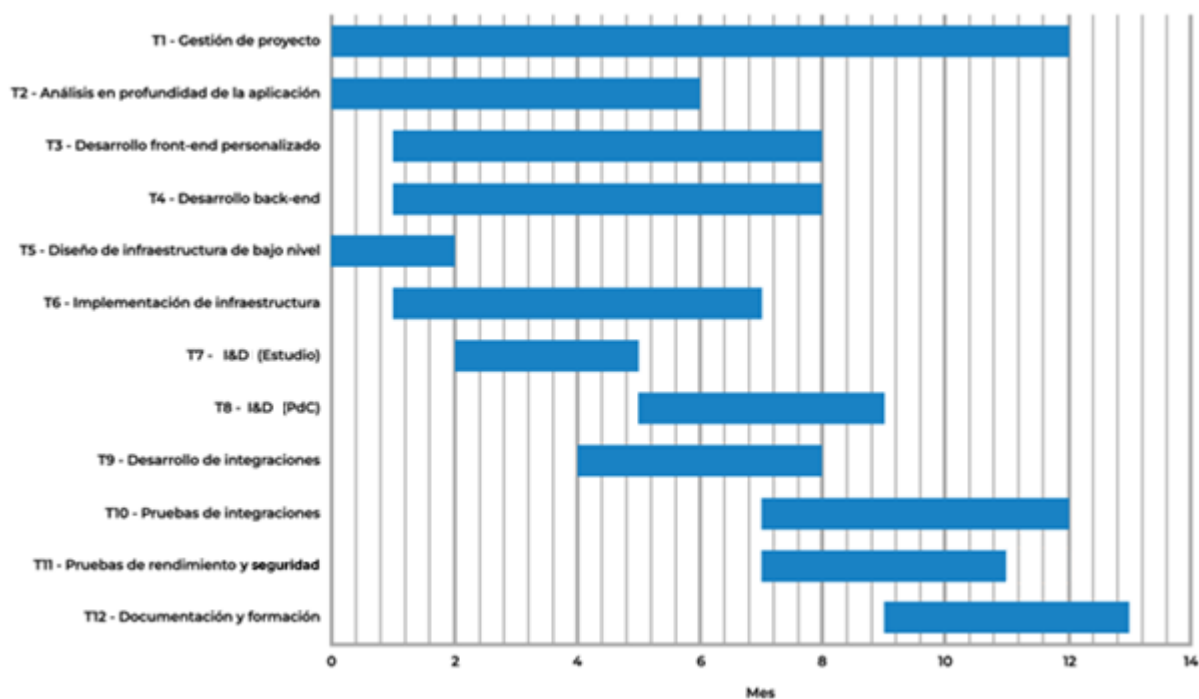
El escenario presentado es el resultado de un análisis detallado de los requisitos y objetivos del proyecto recopilados durante la fase de estudio de la plataforma. En función de esto, la opción descrita anteriormente es la más adecuada para alcanzar los objetivos establecidos y garantizar la satisfacción de las necesidades de todos los usuarios de la plataforma CDMS.

El plan de trabajo siguiente define las macroactividades para la implementación de la plataforma en un período de tiempo total de 13 meses. Muchas de las macroactividades se superponen temporalmente, ya que la metodología de desarrollo adoptada seguirá un enfoque iterativo ágil.

También se presenta el **Cuadro III E** de entregables esperados, es decir, la lista de entregas que deberá realizar el equipo de desarrollo para completar el proyecto en los plazos establecidos.

**Cuadro III.E. Entregables del proyecto**

ID	DESCRIPCIÓN	MES DE ENTREGA
D1	Elaboración de informes bimestrales sobre el estado de las actividades del proyecto	M2, M4, M6, M8, M10, M12
D2.1	Entorno de prueba con posibilidad de verificar las primeras funcionalidades implementadas por un subconjunto predefinido de usuarios finales del sistema.	M3
D3	Documento de análisis de bajo nivel de infraestructura	M3
D4.1	Documento I+D sobre calidad de datos	M6
D2.2	Primera instalación en entorno de producción	M7
D4.2	PoC I+D Demo	M10
D5	Informe de prueba de rendimiento y seguridad	M11
D6	Documentación y guías	M12



**Figura III.C. Gantt del plan de trabajo con planificación de las actividades previstas**

Información detallada sobre hojas de ruta, entregables y planes de trabajo se encuentra en el capítulo 10.

## IV ÍNDICE

I	RESUMEN EJECUTIVO.....	1
II	ACRÓNIMOS.....	2
III	RESUMEN PARA LOS RESPONSABLES DE LA FORMULACIÓN DE POLÍTICAS.....	4
III.A	OBJETIVOS, CRITERIOS Y METODOLOGÍA.....	4
III.B	PLATAFORMAS DE REFERENCIA.....	5
III.C	REQUISITOS FUNCIONALES.....	6
III.D	PLATAFORMAS ABIERTAS DE REFERENCIA.....	8
III.E	PLATAFORMA SUGERIDA Y ESCENARIOS DE EVOLUCIÓN.....	9
III.F	ARQUITECTURA E INFRAESTRUCTURA.....	11
III.G	COSTOS DEL SUPUESTO SUGERIDO.....	12
III.H	HOJA DE RUTA, ENTREGABLES Y PLAN DE TRABAJO.....	13
IV	ÍNDICE.....	15
1	INTRODUCCIÓN.....	18
1.1	OBJETIVOS.....	18
1.2	CRITERIOS.....	19
1.2.1	CUMPLIMIENTO DE LAS NORMAS INTERNACIONALES.....	20
1.2.2	DISPONIBILIDAD.....	29
1.2.3	ACCESIBILIDAD.....	30
1.2.4	TECNOLOGÍA.....	31
1.3	METODOLOGÍA.....	33
1.3.1	REQUISITOS FUNCIONALES.....	34
1.3.2	REQUISITOS NO FUNCIONALES.....	37
1.3.3	TECNOLOGÍA PARA EL CDMS DE LA REPÚBLICA DOMINICANA.....	38
1.3.4	ENFOQUES DE DESARROLLO.....	39
2	PLATAFORMAS DE REFERENCIA.....	42
2.1	COLOMBIA.....	43
2.2	CHILE.....	43
2.3	MÉXICO.....	44
2.4	BELICE.....	45
3	REQUISITOS FUNCIONALES.....	47
3.1	ANÁLISIS PRELIMINAR.....	47



3.1.1	ENCUESTA A LAS PARTES INTERESADAS.....	48
3.1.2	EVALUACIÓN COMPARATIVA DE LAS PLATAFORMAS EXISTENTES.....	52
3.2	REQUISITOS PROPUESTOS.....	54
3.2.1	DIAGRAMAS DE CASOS DE USO.....	55
4	REQUISITOS NO FUNCIONALES.....	75
4.1	ANÁLISIS COMPARATIVO.....	75
4.1.1	ENCUESTA A LAS PARTES INTERESADAS.....	76
4.1.2	ANÁLISIS DE LAS PLATAFORMAS CDMS.....	78
4.2	PLATAFORMA SUGERIDA .....	100
4.2.1	PORQUE EL CÓDIGO ABIERTO .....	101
5	PLATAFORMA CDMS PARA LA REPÚBLICA DOMINICANA.....	103
5.1	INTRODUCCIÓN A DJANGO .....	104
5.2	DEFINICIÓN DE LA ESTRUCTURA DE LA BASE DE DATOS.....	105
5.2.1	PLANIFICADOR DE TAREAS.....	106
5.2.2	GESTIÓN DE SERVICIOS PÚBLICOS .....	107
5.2.3	ALMACENAMIENTO DE DATOS CLIMÁTICOS.....	108
5.3	DEFINICIÓN DE FLUJO DE DATOS.....	110
5.3.1	DECODIFICADOR PARA LA IMPORTACIÓN DE DATOS .....	112
5.3.2	CONTROL DE CALIDAD .....	113
5.4	LAYER API.....	115
5.5	OBTENCIÓN DE DATOS DE FUENTES EXTERNAS.....	118
5.5.1	OPCIONES PARA LA RECOGIDA DE DATOS.....	119
5.6	EVOLUCIONES Y DESARROLLOS FUTUROS.....	123
5.6.1	DESARROLLOS OBLIGATORIOS.....	123
5.6.2	EVOLUCIONES OPCIONALES Y INTEGRACIÓN DE TECNOLOGÍAS CLAVE	125
6	DISEÑO DE UX Y UI PARA ESCRITORIO Y VISUALIZACIÓN WEB MÓVIL.....	136
6.1	DISEÑO UX.....	136
6.2	DISEÑO DE INTERFAZ DE USUARIO.....	137
6.3	ESCENARIO A - SURFACE.....	138
6.3.1	ANÁLISIS DE SURFACE.....	138
6.4	ESCENARIO B - UI PERSONALIZADA.....	143
6.4.1	FASE DE DISEÑO .....	144
6.4.2	ANÁLISIS FUNCIONAL .....	146
6.4.3	ACCESIBILIDAD .....	169
6.4.4	MARCO TÉCNICO PROPUESTO.....	173

7	REFERENCIA FINAL.....	177
8	REQUISITOS DE SISTEMA, SEGURIDAD Y GESTIÓN .....	179
8.1	ARQUITECTURA DE APLICACIONES .....	179
8.2	ARQUITECTURA DE INFRAESTRUCTURAS.....	180
8.2.1	CONTRATACIÓN .....	181
8.2.2	DIAGRAMA ARQUITECTÓNICO Y REQUISITOS DEL SISTEMA .....	183
8.2.3	REQUISITOS DE SEGURIDAD.....	188
8.2.4	ENTORNOS Y CONDUCTOS CI/CD (INTEGRACIÓN CONTINUA/DESPLIEGUE CONTINUO).....	192
8.2.5	SEGUIMIENTO .....	194
8.3	ASISTENCIA Y MANTENIMIENTO DEL SISTEMA.....	195
8.3.1	GUÍAS DEL USUARIO DE SOFTWARE.....	196
8.3.2	FORMACIÓN ADICIONAL.....	196
8.3.3	ASISTENCIA Y NOTIFICACIÓN DE ERRORES.....	197
9	COSTES.....	198
9.1	INTEGRACIÓN DE LOS CASOS DE USO.....	198
9.2	ARQUITECTURA DE APLICACIONES .....	203
9.3	ESCENARIOS DE INTEGRACIÓN Y ASISTENCIA AL USUARIO .....	204
9.4	CUADRO RESUMEN .....	205
10	ESCENARIO DE APLICACIÓN DEL PLAN DE TRABAJO .....	207
10.1	HIPÓTESIS PROPUESTA.....	207
10.2	TAREAS DEL PLAN DE TRABAJO.....	208
10.2.1	GANTT .....	211
10.4	ENTREGABLE.....	211
12	BIBLIOGRAFÍA.....	213
13	APÉNDICE A.....	220
14	APÉNDICE B - OPCIONES DE DISEÑO RELATIVAS ALA UI PERSONALIZADA .....	225
B.1	TIPOGRAFÍA.....	225
B.2	COLOR DEL TEXTO .....	226
B.3	PALETAS DE COLORES .....	226
B.4	DISPOSICIÓN Y ESPACIADO .....	227
B.5	ICONOS .....	228
B.6	EFEECTO VIDRIO .....	229
B.7	COMPONENTES.....	229

# 1 INTRODUCCIÓN

## 1.1 OBJETIVOS

Los objetivos de este proyecto son múltiples y ambiciosos. En primer lugar, se trata de definir con precisión los requisitos para el desarrollo de un Sistema de Gestión de Datos Climáticos (CDMS, por sus siglas en inglés) diseñado específicamente para la República Dominicana. Esto se logrará en primer lugar mediante un análisis detallado de las normas exigidas por la Organización Meteorológica Mundial (OMM) y las plataformas básicas existentes identificadas en el anuncio del proyecto como puntos de referencia mínimos. Sus funcionalidades se utilizan como punto de partida para identificar las características clave que se incorporarán a la plataforma dominicana.

Mientras se siguen los pasos de los resultados del proyecto, es crucial definir una estructura de base de datos apropiada que satisfaga las necesidades únicas del escenario dominicano, y que haga uso de las últimas y más relevantes tecnologías disponibles - incluyendo el desarrollo de una capa de *Interfaz de Programación de Aplicaciones* (API, por sus siglas en inglés) versátil y eficiente.

Un segundo objetivo del proyecto es realizar un análisis en profundidad de las mejores plataformas de código abierto disponibles en la actualidad y adheridas a las normas internacionales de referencia, centrándose en su funcionalidad, tecnología utilizada y facilidad de uso. Este análisis permitirá identificar posibles plataformas que puedan utilizarse como punto de partida para el futuro desarrollo de la plataforma nacional, teniendo en cuenta su compatibilidad con los requisitos específicos de la República Dominicana y las normas tecnológicas y reglamentarias pertinentes. Será importante evaluar detenidamente las distintas opciones disponibles y elegir la que mejor se adapte a las necesidades del proyecto.

Un tercer objetivo del proyecto es identificar futuros escenarios que personalicen el desarrollo de la plataforma CDMS específicamente para la República Dominicana. Para que esto ocurra de forma provechosa, se requiere una comprensión profunda de los requisitos y retos únicos del contexto singular del país, incluida la integración de los datos con los sistemas existentes, que se caracterizan por un alto grado de heterogeneidad en la instrumentación y la propiedad de los sistemas, y el desarrollo de funcionalidades para apoyar una densificación de la red de medición, que se encuentra en fase de planificación. La identificación de estos y otros escenarios permite desarrollar la plataforma CDMS para responder adecuadamente a las necesidades locales precisas, garantizando su pertinencia y eficacia.

Por último, el proyecto pretende identificar y definir futuros desarrollos, tanto obligatorios como sugeridos, para que la plataforma CDMS sea versátil, esté preparada para el futuro y pueda ofrecer una experiencia de usuario completa y agradable. Esto incluye identificar funciones adicionales que puedan mejorar las capacidades de la plataforma, explorar nuevas tecnologías que mejoren su rendimiento o sugerir mejoras en la interfaz de usuario para hacerla más intuitiva. Si se planifican estos desarrollos futuros, la plataforma CDMS puede diseñarse para adaptarse a los requisitos cambiantes y a los avances tecnológicos, garantizando su durabilidad y usabilidad a largo plazo.

## 1.2 CRITERIOS

Ante la necesidad de ejecutar un gran proyecto, es esencial establecer criterios claros que guíen las opciones de diseño y desarrollo. En el caso de un proyecto internacional, como el diseño de una plataforma de gestión de datos climáticos (CDMS - Climate Data Management System), los criterios adoptados pueden dividirse en dos grandes grupos: institucionales y normativos, y prácticos y tecnológicos.

Existen normas internacionales para tales sistemas, definidas y perfeccionadas por la Organización Meteorológica Mundial (OMM) y en constante evolución. Por lo tanto, como paso previo, se consideró necesario que la futura plataforma CDMS de la República Dominicana se ajustara lo más posible a estos estándares, incluyendo la posibilidad de formar parte de un proyecto fundamental como OpenCDMS<sup>[1][2]</sup>, antes de establecer cualquier otra forma práctica de desarrollo e implementación.

Posteriormente, se definieron con más detalle los criterios prácticos, que, al tratarse esencialmente de un sistema de software, son principalmente criterios tecnológicos. Estos criterios incluyen tanto aspectos técnicos propiamente dichos (por ejemplo, el SGBD o el lenguaje de programación utilizado) como la influencia de la tecnología adoptada en otras características, como la facilidad de personalización o la disponibilidad y accesibilidad del software. Cada uno de estos criterios se considera importante y tiene su propia sección en las páginas siguientes, independientemente de su naturaleza intrínseca.

---

<sup>1</sup> Sección 1.2.1.1

<sup>2</sup> [Sitio web oficial del proyecto](#)

## 1.2.1 CUMPLIMIENTO DE LAS NORMAS INTERNACIONALES

La adhesión y el cumplimiento de las normas internacionales son cruciales para garantizar la interoperabilidad, eficiencia y eficacia de las actividades y servicios en muchos ámbitos, incluidos los relacionados con el medio ambiente y el clima. El cumplimiento de las normas internacionales permite disponer de un conjunto común de estándares para la recopilación, gestión e intercambio de datos e información, garantizando que la información recopilada sea fiable, comparable y utilizable por todos los actores implicados (Zbinden & Ionescu, 2018).

En el sector meteorológico y climatológico, los estándares internacionales están definidos por la Organización Meteorológica Mundial (OMM)<sup>[3]</sup>, una organización intergubernamental de carácter técnico que se ocupa de la meteorología y que cuenta con 191 Estados miembros y territorios. La adhesión y conformidad a los estándares internacionales son particularmente importantes, ya que la información recopilada y compartida puede tener implicaciones significativas en la seguridad, salud y calidad de vida de las personas en todo el mundo, así como en la posibilidad de análisis y comparación entre diferentes regiones. Los estándares de metadatos WIGOS<sup>[4]</sup>, la adhesión a WIS 2.0<sup>[5]</sup> y el proyecto OpenCDMS<sup>[6]</sup> representan algunos de los desarrollos más recientes en este sector, que buscan mejorar la calidad y confiabilidad de los datos e información meteorológica y climatológica recopilada y compartida entre los diversos países y organizaciones.

La adhesión a las normas de metadatos de WIGOS garantiza que los datos meteorológicos y climatológicos se recopilen y gestionen de forma coherente y normalizada, mientras que la adhesión al SIO 2.0 permite el intercambio eficaz y seguro de esta información entre países y organizaciones (OMM, 2014b, 2015). El proyecto OpenCDMS, por su parte, pretende promover el intercambio abierto y el acceso gratuito a los datos meteorológicos y climatológicos, permitiendo a más investigadores y partes interesadas acceder a esta información y utilizarla con fines científicos, comerciales y de política pública (NOAA, 2020).

En resumen, la adhesión y el cumplimiento de las normas internacionales son cruciales para garantizar la eficacia y la eficiencia de las actividades y los servicios en el sector meteorológico y climatológico, con importantes implicaciones para la seguridad y la calidad de vida de las personas en todo el mundo (Zbinden & Ionescu, 2018).

---

<sup>3</sup> [Sitio oficial](#)

<sup>4</sup> Párrafo 1.2.1.3

<sup>5</sup> Párrafo 1.2.1.2

<sup>6</sup> Párrafo 1.2.1.1

## 1.2.1.1 PROYECTO OpenCDMS

Ya sea a nivel local, regional o global, una gestión efectiva de los datos ambientales, especialmente los datos hidrometeoroclimáticos, es crucial para el monitoreo, análisis, modelado y pronóstico de las observaciones del sistema terrestre, así como para la provisión eficiente y oportuna de todos los servicios y estudios relacionados, como requisito fundamental para la gestión sostenible del territorio.

Es evidente que uno de los elementos más importantes para una gestión efectiva de los datos del sistema terrestre es contar con un sistema de gestión de datos climáticos (CDMS) potente y versátil, que pueda almacenar y procesar de manera efectiva cualquier tipo de dato hidrometeoroclimático con componentes de series temporales, es decir, aquellos datos que tienen alguna referencia al momento o período de tiempo en que fueron recopilados.

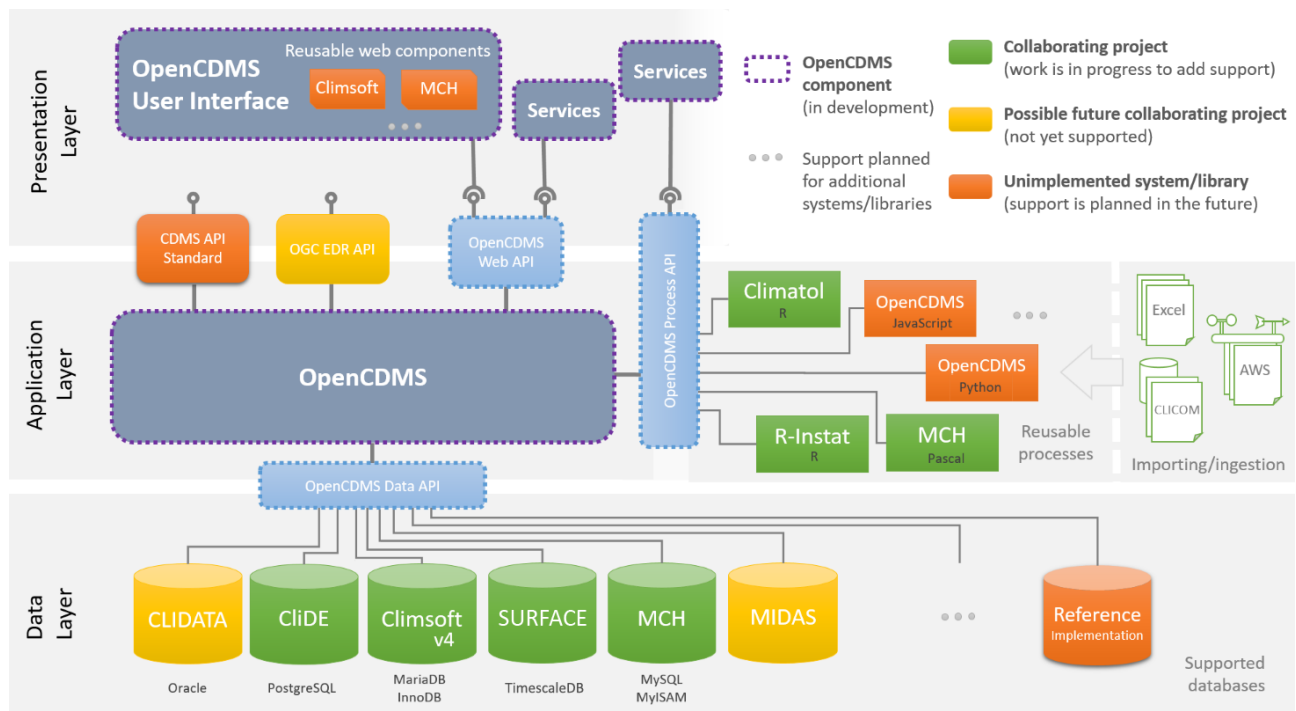


Figura 1. Diagrama arquitectónico de OpenCDMS

La idea detrás de la creación del proyecto OpenCDMS surgió durante el trabajo en las Especificaciones del Sistema de Gestión de Datos Climáticos en 2014 (WMO), y su desarrollo fue formalmente propuesto al año siguiente por el Equipo de Expertos en Sistemas de Gestión de Datos Climáticos (ET-CDMS) (Proyecto OpenCDMS, 2023). El Declaración de Política del ET-CDMS señala cómo actualmente se requiere una inversión significativa para mantener activos y funcionando los numerosos CDMS utilizados en todo el mundo para generar datos para análisis hidrometeoroclimáticos a nivel nacional, regional o global. Además, la mayoría de los CDMS existentes no cumplen con las Especificaciones del Sistema de Gestión de Datos Climáticos (WMO, 2014a; Proyecto

OpenCDMS, 2023). Esto significa que, además del mantenimiento habitual, sería necesario realizar inversiones significativas para cumplir con los requisitos de la WMO-No. 1131, lo que implicaría un potencial desperdicio de recursos al desarrollar las mismas funcionalidades en diferentes CDMS.

En este contexto, surge la consideración adicional de que ya existen varios CDMS de código abierto utilizados por los miembros de la WMO. El proyecto OpenCDMS tiene la intención e intuición de aprovechar el considerable potencial existente al racionalizar estos CDMS de código abierto, fusionándolos en un único CDMS de referencia de código abierto: la plataforma OpenCDMS. La idea detrás del proyecto ha recibido amplio apoyo y adhesión, lo que llevó a que OpenCDMS fuera adoptado como proyecto oficial de la Asociación AISBL OpenWIS en mayo de 2020 (Proyecto OpenCDMS, 2023).

Al ser basado en el modelo de código abierto, el proyecto OpenCDMS es fundamentalmente una comunidad de colaboradores que trabajan juntos para abordar las necesidades de gestión de datos hidrometeoroclimáticos tanto de países en desarrollo como desarrollados. El objetivo siempre es establecer e implementar mejores prácticas y estándares globales de gestión de datos para fomentar que las soluciones de gestión de datos existentes y futuras se adhieran a estos estándares y, cuando sea posible, al propio proyecto. Además, el grupo de desarrollo tiene como objetivo la creación y el soporte de soluciones de software gratuitas y de código abierto para ayudar a implementar las mencionadas mejores prácticas y estándares internacionales de referencia.

2019-2020	2021-2022	2023-2024	2025	2026-
<p>La decimoctava sesión del Congreso Meteorológico Mundial solicitó el desarrollo de un CDMS de fuente abierta de referencia</p> <p>OpenCDMS como un marco para la colaboración establecido como un proyecto de la Asociación Open WIS</p>	<p>Historias de usuario, requisitos funcionales y pruebas de aceptación para implementación de referencia</p> <p>Desarrollo de un modelo de datos de próxima generación para ser adoptado como estándar de la OMM</p>	<p>Implementación de referencia y modelo de datos de OpenCDMS presentados a la decimonovena sesión del Congreso Meteorológico Mundial para su aprobación.</p> <p>Pruebas de software en países de enfoque</p>	<p>Paquete de software de código abierto instalable OpenCDMS disponible</p> <p>Inicio de la implementación y capacitación</p>	<p>Implementación y capacitación continua</p> <p>Mantenimiento y mayor desarrollo del paquete de software OpenCDMS</p>
Prototipos y desarrollo de OpenCDMS			Implementación de OpenCDMS	
<p><b>Mayor colaboración con desarrolladores de CDMS y comunidades de usuarios, movilización de recursos, establecimiento de una comunidad de práctica que aplica principios de fuente abierta</b></p>				

**Figura 2. Cronología del proyecto OpenCDMS**

Así, los principales objetivos del proyecto OpenCDMS pueden resumirse en los siguientes puntos (OpenCDMS Project, 2023):

- Mejorar la interoperabilidad entre los CDMS existentes;
- Crear una implementación de referencia para un CDMS de nueva generación que cumpla plenamente las normas metodológicas y tecnológicas (con la orientación de grupos de expertos y desarrolladores);
- Trabajar para apoyar los proyectos existentes y colaborar con ellos con la intención de garantizar que los usuarios del CDMS sigan beneficiándose de todos los servicios preexistentes, incluido el software.

Además, OpenCDMS continuará siguiendo los principios establecidos en los Términos de Referencia de las Comisiones Técnicas de la OMM (2019a), garantizando que las soluciones desarrolladas e implementadas:

- Estén orientada al usuario y proporcionar observaciones hidrometeorológicas, además de procesamiento de datos y servicios, productos e información pertinentes para los miembros;
- Sean aplicable de forma realista, asequible y con gestión del ciclo de vida en toda la gama de miembros de la OMM;
- Se basen en los principios de modularidad y escalabilidad, en la medida de lo posible;
- Apliquen plena y provechosamente las normas y reglamentos vigentes de la OMM y otros organismos pertinentes;
- Utilicen y promuevan la interacción público-privada siempre que sea beneficiosa;
- Incorporen tecnologías de punta, óptimas y adaptadas a sus necesidades;
- Se basen en las necesidades de los usuarios, elaboradas en coordinación con la Comisión de Servicios y el Consejo de Investigación;
- Aprovechen las asociaciones y redes existentes entre comunidades de práctica dentro de las áreas de servicio, que son beneficiosas para los miembros de la OMM.

En conclusión, dada la importancia fundamental de la gestión de datos ambientales, especialmente los datos hidrometeorológicos, para el monitoreo, análisis, modelado y pronóstico de las observaciones del sistema terrestre, así como para brindar servicios eficaces y oportunos, **la adopción de un CDMS potente y versátil es esencial**. En este contexto, el proyecto OpenCDMS se presenta como una opción altamente recomendable, ya que tiene como objetivo racionalizar las numerosas soluciones CDMS de código abierto ya existentes, unificándolas en una única plataforma de referencia de código abierto. La adhesión al proyecto permitirá ahorrar recursos significativos al cumplir con los requisitos globales de gestión de datos hidrometeorológicos y mantener un CDMS que cumpla con diversas especificaciones y requisitos internacionales establecidos por la WMO, a los cuales se ha mencionado anteriormente y se continuará discutiendo en las secciones siguientes del documento. **Por lo tanto, se recomienda unirse al proyecto OpenCDMS para la gestión de datos hidrometeorológicos en la República Dominicana.**



### **1.2.1.2 DE GTS Y WIS A WIS 2.0**

En 1971, el Sexto Congreso Meteorológico Mundial aprobó el Manual sobre el nuevo Sistema Mundial de Telecomunicaciones (SMT) (OMM, 1971), iniciando así su vida operativa. El SMT se describe en el Manual como "El sistema mundial coordinado de instalaciones y arreglos de telecomunicaciones para la rápida recopilación, intercambio y distribución de observaciones e información procesada en el marco de la Vigilancia Meteorológica Mundial" (OMM, 2019b).

Durante los últimos cincuenta años, el SMT ha permitido un intercambio continuo y en tiempo real de datos hidrometeorológicos esenciales, proporcionando observaciones a los Centros Mundiales de Proceso de Datos y de Predicción y enviando información preprocesada a los Servicios Meteorológicos e Hidrológicos Nacionales (SMHN) - cuyo trabajo se basa en observaciones y otros datos producidos y compartidos libremente en tiempo real, de acuerdo con los principios de la Vigilancia Meteorológica Mundial (OMM, 1995). Aunque se han producido algunos pequeños avances en las tecnologías utilizadas para el intercambio de datos, el GTS ha mantenido inalterados sus fundamentos técnicos básicos. La aparición en las dos últimas décadas de una conectividad mundial cada vez más rápida y eficiente -y de banda ancha- a través de Internet está brindando nuevas oportunidades para la evolución del GTS (Giraud et al, 2021).

El Decimoquinto Congreso Meteorológico Mundial, celebrado en 2007, dio un paso importante hacia la mejora del sistema y la resolución de estos problemas, con la motivación principal de poder proporcionar acceso a los datos también a entidades no conectadas directamente al SMT (OMM, 2007). Esta consideración condujo al desarrollo paralelo del Sistema de Información de la OMM (SIO), destinado a suplir algunas de las carencias del SMT. El SIO implementa un catálogo de búsqueda y un caché global para permitir servicios adicionales de descubrimiento, acceso y recuperación a través de portales web, gestionados por quince Centros del Sistema Mundial de Información (GISC) designados, cada uno de ellos operado por un Miembro de la OMM.

El SIO también definió nuevas funciones para los Centros de la OMM en todo el mundo, reconociendo la necesidad de mejorar la comunicación y la coordinación entre los Miembros, y de facilitar el intercambio de datos más allá de la Vigilancia Meteorológica Mundial. Sin embargo, el sistema SIO sigue utilizando el GTS como servicio de infraestructura operativa para el intercambio de datos, sólo que con algunas ligeras mejoras, por lo que hereda la mayoría de sus limitaciones, ya sean intrínsecas al sistema o debidas al peso de los años.

El acceso a los datos GTS (en muchos casos incluso su búsqueda efectiva) requiere conocimientos especializados, disponibles sólo en una comunidad limitada de expertos, principalmente de los SMHN con más recursos a su disposición. Esto significa que los SMHN de los países miembros menos desarrollados no suelen estar bien equipados para acceder y utilizar esta valiosa fuente de datos en tiempo real, y que otras instituciones y

el público en general quedan totalmente excluidos. La puesta en marcha del SIO tenía como uno de sus principales objetivos que los usuarios de todo el mundo pudieran, en principio, buscar y acceder a los datos libremente o solicitar permiso a sus propietarios. Aunque permitió la publicación de muchos conjuntos de datos del SMT y de otras fuentes, el SIO nunca alcanzó plenamente su objetivo original de facilitar el acceso a los datos de la OMM.

Los 15 SIGC ofrecen varias interfaces web, pero sus portales son menos útiles de lo que se pretendía en un principio porque presentan demasiadas barreras para los usuarios no expertos (Giraud et al, 2021). La complejidad de la información dificulta el mantenimiento de un catálogo coherente y eficaz con metadatos significativos y de alta calidad. Esto significa que a menudo es casi imposible encontrar los datos que uno necesita sin la ayuda de los expertos del GTS. En última instancia, el catálogo del SIO se dirige a un público equivocado: el objetivo original de exponer el SIO a usuarios no expertos no se consiguió (Giraud et al, 2021).

El objetivo del diseño y la aplicación del SIO 2.0 es precisamente resolver los problemas antes mencionados. De este modo, pondrá los datos hidrometeorológicos al alcance de todos, derribando las barreras infraestructurales y facilitando el acceso a los intercambios de datos y servicios a toda la comunidad de la OMM, y a cualquier otra persona interesada (Giraud et al, 2021).

WIS 2.0 da prioridad al uso de la infraestructura web, a diferencia de GTS, que se basa principalmente en redes privadas. La adopción de una tecnología generalizada y bien entendida favorece enormemente un enfoque más participativo, en el que los usuarios dejan de ser meros observadores. Los SMHN pueden publicar sus datos como directorios de archivos o a través de API de servicios web, lo que permite a los usuarios ver o descargar exactamente lo que buscan sin necesidad de software especializado. El uso de la web también permite otras plataformas de acceso a los datos, como los Sistemas de Información Geográfica (SIG) o las aplicaciones móviles.

Sin embargo, esto no significa que todo el mundo pueda acceder a los datos sin restricciones. Es posible que se apliquen controles de seguridad para restringir el acceso cuando sea necesario, mediante el uso de sistemas de autenticación y autorización (Giraud et al, 2021). Esto permite al proveedor de datos mantener el control sobre quién puede acceder a los recursos publicados y poder exigir a los usuarios, si lo considera necesario, que acepten una licencia como condición para darles acceso.

El objetivo principal del SIO 2.0 es superar la mayoría de los problemas inherentes al SIO, empezando por su limitada eficacia a la hora de satisfacer las necesidades de la amplia comunidad de la OMM. Los Principios del SIO 2.0 (OMM, 2021a) constituyen el núcleo del proyecto: comprenden un conjunto de prácticas técnicas y operativas destinadas a modernizar y promover la accesibilidad de los recursos, los datos y la información, mejorando al mismo tiempo la eficiencia del intercambio físico de datos. Para ello se

adopta un enfoque orientado al codesarrollo y se trabaja con las organizaciones interesadas para hacer evolucionar iterativamente el sistema central.

Aunque la red de centros del SIO está bien establecida y es plenamente operativa, todavía hay regiones del mundo donde la disponibilidad de datos es más bien escasa. Por lo tanto, con el SIO 2.0, la mejora de la disponibilidad de datos sigue siendo una prioridad, apoyando especialmente a los países en desarrollo en el esfuerzo por mantenerse a la vanguardia, con el objetivo de que nadie se quede atrás. Una combinación de protocolos estándar y ligeros, tecnologías en la nube y la Internet pública permitirá a los PMA explotar los recursos ya disponibles siempre que sea posible, y abordar la complejidad de la empresa reduciendo las barreras tecnológicas y optimizando el intercambio de datos para tener en cuenta las limitaciones de la infraestructura (Giraud et al, 2021).

El SIO 2.0 es, por lo tanto, el siguiente y necesario paso en la infraestructura de intercambio de datos de la OMM, y desempeñará un papel fundamental a la hora de colmar la brecha de capacidad y promover la adaptación y la creación de resiliencia mundial ante el aumento de los riesgos meteorológicos y climáticos. Por este motivo, se considera necesario e imperativo que la plataforma CDMS propuesta en este documento se adhiera plenamente a ella, ya sea de forma nativa o a través de futuros desarrollos. En particular, la solución preferida es la integración con el software wis2box, que proporciona una serie de servicios destinados a introducir, transformar y publicar datos hidrometeorológicos mediante la aplicación de los requisitos fundamentales del SIO 2.0 (OMM, 2022).

### **1.2.1.3 WIGOS Y METADATOS WIGOS**

El Sistema Mundial Integrado de Observación de la OMM (WIGOS) es un marco de apoyo a todos los programas y actividades de observación y recopilación de datos de la Organización Meteorológica Mundial. Se trata principalmente de un conjunto de recomendaciones y directrices para todos los sistemas de observación de la OMM, ya pertenezcan directamente a la organización o estén copatrocinados por ella.

Por lo tanto, WIGOS pretende facilitar a los miembros de la OMM el uso de datos procedentes de sistemas gestionados y operados por una amplia gama de organizaciones y programas, con el objetivo adicional de satisfacer de forma iterativa las necesidades de los miembros en materia de observaciones hidrometeorológicas, necesidades que evolucionan constantemente. Un objetivo importante que debe alcanzarse es la interoperabilidad y compatibilidad de los datos entre distintos sistemas, con la idea básica de que esto puede lograrse mediante la aplicación de normas aceptadas internacionalmente, prácticas y procedimientos recomendados y el uso de normas compartidas de representación de datos.

Un aspecto clave de la aplicación de WIGOS es garantizar la máxima utilidad y utilidad de las observaciones y encuestas. Las observaciones sin metadatos sirven de poco: sólo cuando van acompañadas de metadatos adecuados (datos que describen los datos) se puede alcanzar y explotar todo el potencial del sistema de observación y encuesta.

Para ello se necesitan dos tipos complementarios de metadatos: metadatos de descubrimiento y metadatos de interpretación/descripción (u observacionales). Los metadatos de descubrimiento están diseñados para facilitar el descubrimiento, el acceso y la recuperación de datos. Pertenecen a los metadatos del SIO y se especifican y gestionan como parte del sistema. Los metadatos de interpretación/descripción u observacionales, por su parte, tienen por objeto facilitar la interpretación de los valores de los datos en el contexto al que pertenecen. Este segundo tipo de metadatos pertenece al conjunto de metadatos de WIGOS, que se considera necesario para el uso eficaz de las encuestas y observaciones de todos los sistemas de observación miembros de WIGOS.

El Manual sobre el Sistema mundial integrado de observación de la OMM (OMM-No. 1160) se refiere, en efecto, con respecto a los metadatos, a la necesidad de "Proporcionar información para que las observaciones puedan ser interpretadas" (2021b, Apéndice 2.1), mientras que el Principio de Vigilancia del Clima 2.2.1(c) del Sistema Mundial de Observación del Clima (SMOC) subraya la importancia de los metadatos al indicar que "Los detalles y la historia de las condiciones locales, los instrumentos, los procedimientos operativos, los algoritmos de procesamiento de datos y otros factores pertinentes para la interpretación de los datos (es decir, los metadatos) deben documentarse y tratarse con el mismo cuidado que los propios datos" (OMM, 2021b, Apéndice 2.2).

Las observaciones y encuestas de WIGOS constan de una gama muy amplia de datos de naturaleza y formato igualmente variados. La definición de una norma completa de metadatos que abarque todos los tipos de observaciones es, por naturaleza, compleja. Un usuario debe poder utilizar los metadatos de WIGOS para identificar las condiciones en las que se realizó la observación o medición y cualquier aspecto que pueda influir en su uso o comprensión. El objetivo último de los metadatos de WIGOS es, por tanto, describir la variable observada, las condiciones en las que se observó, cómo se midió o clasificó y cómo se procesaron los datos, con el fin de proporcionar a los usuarios una certeza razonable de que los datos son adecuados para el uso previsto.

Se han identificado diez categorías de metadatos WIGOS (OMM, 2019d, p. 2). Estas categorías definen la Norma de metadatos de WIGOS, y cada una de ellas consta de una o más subcategorías de metadatos (OMM, 2019d, p. 5-6). Todas las categorías enumeradas se consideran importantes para la documentación e interpretación de las observaciones realizadas, también para su uso en el futuro. Por lo tanto, la norma también hace referencia a muchos elementos que no son estrictamente necesarios para las aplicaciones que se centran en un uso más inmediato o específico de las observaciones. Las categorías no siguen un orden determinado -de importancia o no-, sino que reflejan la necesidad inherente de evaluar y documentar todos los

componentes de la observación que se consideran útiles para los fines últimos de todos los usuarios.

**Cuadro 1. Categorías de metadatos de WIGOS** (OMM, 2019d, p. 2)

Categoría	Descripción
1. Variable observada	Especifica las características básicas de la variable observada y de los conjuntos de datos resultantes. Incluye un elemento que describe la representatividad espacial de la observación, así como el compartimento biogeofísico descrito por la observación.
2. Objetivo de la observación	Especifica la(s) principal(es) área(s) de aplicación de la observación y el (los) programa(s) y red(es) de observación a los que está afiliada la observación.
3. Estación/plataforma	Especifica la instalación de observación (estación fija, equipo móvil, plataforma de teledetección, etc.) en la que se realiza la observación.
4. Medio ambiente	Describe el entorno geográfico en el que se realiza la observación. También proporciona un elemento no estructurado para metainformación adicional que se considera relevante para el uso adecuado de las observaciones pero que no puede clasificarse en otras secciones de metadatos de WIGOS.
5. Instrumentos y métodos de observación	Especifica el método de observación y describe las características de los instrumentos utilizados para llevar a cabo la observación. Debe utilizarse tantas veces como instrumentos intervengan en la generación de los datos.
6. Muestreo	Especifica cómo se utiliza el muestreo y/o el análisis para obtener los datos notificados, o cómo se recogen físicamente las muestras.
7. Tratamiento de datos e informes	Especifica cómo se transforman los datos brutos en variables observadas y cómo se comunican a los usuarios.
8. Calidad de los datos	Especifica la calidad de los datos y la trazabilidad de la observación.
9. Propiedad y política de datos	Especifica quién es el responsable y el propietario de la observación.
10. Contacto	Especifica dónde puede obtenerse información sobre la observación o el conjunto de datos.

En el **Cuadro 1** figura una composición esquemática de todas las categorías, con sus correspondientes descripciones. Algunos de estos elementos se implementan después en la construcción práctica de un modelo de datos utilizando varias subentidades individuales (por ejemplo, la posición geoespacial consiste en una combinación de elementos, como la latitud, la longitud, la altitud o un conjunto de coordenadas polares, así como una referencia a los métodos de geoposicionamiento utilizados). Se puede encontrar información más detallada sobre la categorización de los metadatos en el **Cuadro 1** del **Apéndice A** y en el documento oficial de la OMM (OMM, 2019d).

En conclusión, **se recomienda la adhesión al marco WIGOS**, sobre todo en lo que respecta a su exhaustiva norma de metadatos definida anteriormente. En realidad, WIGOS se desarrolló específicamente para facilitar el uso de datos procedentes de sistemas de observación gestionados por diferentes organizaciones y programas, garantizando la interoperabilidad y la compatibilidad de los datos. Sin embargo, es importante subrayar que las observaciones sin metadatos son de poca utilidad y que,

para obtener el máximo beneficio del uso de los datos, se necesitan metadatos adecuados. En particular, los metadatos de descubrimiento y los metadatos de interpretación/descripción son complementarios y esenciales para garantizar la máxima utilidad y facilidad de uso de las observaciones y conclusiones, y proporcionar a los usuarios una certeza razonable de que los datos son adecuados para el uso previsto.

## **1.2.2 DISPONIBILIDAD**

La disponibilidad de un determinado programa informático significa, ante todo, la facilidad con la que un usuario interesado puede obtenerlo, instalarlo y utilizarlo. Este análisis no se referirá a los programas informáticos genéricos, sino a las plataformas de gestión de datos hidrometeorológicos y, por ello, la disponibilidad debe entenderse más concretamente como la facilidad con la que el usuario puede proceder a la recogida, gestión, análisis y visualización de este tipo de datos. El grado de disponibilidad de los programas informáticos CDMS es un factor crucial para los profesionales en particular, ya que científicos, expertos en clima y responsables políticos tratan de comprender el cambio climático y desarrollar colectivamente políticas públicas eficaces para hacerle frente.

Como primer punto importante, la disponibilidad de código abierto de un CDMS ofrece varias ventajas a los usuarios. En primer lugar, el acceso al código fuente permite analizar libremente el software y, en caso necesario, adaptarlo a necesidades específicas (Budapest Open Access Initiative, 2002). Se pueden hacer modificaciones de todo tipo, desde añadir funcionalidades a la interfaz hasta desarrollar nuevos módulos para el análisis de datos, una posibilidad especialmente importante para quien tenga necesidades de análisis únicas o más específicas que las que permite la versión básica del programa. Este enfoque también ayuda a promover eficazmente la colaboración y el intercambio de conocimientos entre los usuarios, que pueden compartir sus módulos, contribuir con el código fuente y aportar comentarios sobre futuros desarrollos, fomentando el desarrollo continuo de software y promoviendo la innovación (Budapest Open Access Initiative, 2002).

Además de la disponibilidad de código abierto de las plataformas CDMS, también existen otras iniciativas como Open Data y Open Hardware que promueven la accesibilidad de los datos climáticos y las tecnologías de recopilación de datos (Barrett et al., 2012). Open Data es un movimiento que fomenta el intercambio público de datos hidrometeorológicos de forma gratuita y accesible para todos (Comisión Europea, 2016). Esto favorece una mejor comprensión de la misma y fomenta la elaboración de políticas públicas basadas en información más precisa y actualizada. El acceso a datos hidrometeorológicos abiertos es especialmente importante para los países en desarrollo, que pueden carecer de recursos para recopilar de forma independiente todos los datos que necesitarían.

El hardware abierto es otro movimiento, en una línea similar, que promueve la mayor disponibilidad posible de herramientas y tecnologías para la recopilación de datos climáticos (Pearce et al., 2017). Esta filosofía permite a los usuarios personalizar y crear herramientas en función de sus necesidades específicas, lo que mejora la eficacia y la calidad de la recogida de datos. Además, la disponibilidad de herramientas de hardware abierto puede ser especialmente beneficiosa para las organizaciones con presupuestos limitados que no pueden permitirse adquirir costosas herramientas comerciales (Smith, 2013; Höft et al., 2018).

La adopción de iniciativas como Open Data, Open Hardware y Open Source no solo es importante para lograr una mayor colaboración, intercambio de conocimientos y transparencia en el sector climático: también es crucial para optimizar los recursos, una consideración que adquiere cada vez más importancia a medida que pasa el tiempo y crece la concienciación al respecto (González et al., 2015). De hecho, la disponibilidad de enfoques gratuitos y compartidos puede ser especialmente ventajosa para las organizaciones con limitaciones presupuestarias, ya que les permite reorientar parte de sus recursos a otras actividades importantes, como la recopilación de datos, la elaboración y publicación de informes y la investigación, con especial atención a la evaluación de los efectos del cambio climático (Dimitrov et al., 2020).

Por último, la disponibilidad de un CDMS puede influir directamente en el éxito y la adopción del software por parte de los usuarios. Si el software es difícil de obtener o utilizar, es razonable que los usuarios o las instituciones prefieran alternativas más baratas. Al mismo tiempo, si la plataforma es fácil de obtener y utilizar, puede llegar a ser muy popular y estar muy extendida, lo que hace que estos factores sean especialmente críticos para su éxito a medio y largo plazo (López-Gómez et al., 2019).

### **1.2.3 ACCESIBILIDAD**

El concepto de accesibilidad del software, en general, se refiere a la capacidad de un programa informático para ser comprendido y utilizado fácilmente por todos los usuarios, independientemente de sus capacidades o de las tecnologías de asistencia que puedan utilizar (Gupta y Bajaj, 2019; Vanderheiden, 2019). En pocas palabras, el objetivo principal de la accesibilidad del software es garantizar que el usuario final pueda utilizar las funcionalidades del programa de forma sencilla e intuitiva, sin encontrar barreras o dificultades que impidan utilizar el producto para el fin para el que fue diseñado y desarrollado (Laureano-Cruces, 2020). Por lo tanto, la creación de software accesible se convierte en un requisito previo crucial cuando se trata de promover la inclusión y la equidad en el acceso a la información y la tecnología, mejorar la experiencia del usuario y aumentar la productividad para todos los implicados (Fujimoto y Ota, 2018; Vanderheiden, 2019).

El uso generalizado de dispositivos móviles y el crecimiento de las innovaciones tecnológicas han convertido la posibilidad de acceder a los datos a través de un visor web en un elemento crucial para el uso de la información climática (Florido-Benítez et al., 2020). Un visor web, de hecho, permite a los usuarios acceder al software en tiempo real y en cualquier momento, y tomar decisiones informadas de forma rápida y eficiente, con el único requisito previo de tener a mano un dispositivo capaz de conectarse a internet y una conexión no demasiado rudimentaria (Ganti et al., 2016; Patil et al., 2019). Este menor número de requisitos previos también hace que la accesibilidad del software sea un factor importante en situaciones tecnológicamente más desfavorecidas, como las zonas rurales o remotas y los PMA (Florido-Benítez et al., 2020).

El razonamiento adquiere especial importancia cuando se trata de acceder a datos hidrometeorológicos, ya que la posibilidad de acceder a la información desde cualquier lugar y en cualquier momento es crucial para quienes trabajan sobre el terreno o se desplazan constantemente, a menudo en situaciones en las que no disponer de un marco temporal puede marcar la diferencia. Sin embargo, no es necesario llegar tan lejos como las posibles emergencias: incluso para los meros aficionados, es importante poder tener una experiencia de consulta libre y eficiente en movimiento, especialmente ahora que existen múltiples tecnologías que apoyan este tipo de enfoque (Laureano-Cruces, 2020; Wang et al., 2020).

En conclusión, **la creación de software accesible no sólo beneficia a los usuarios, sino también a las organizaciones que lo utilizan. De hecho, al aumentar la eficacia y eficiencia de las operaciones, el software accesible puede incrementar la productividad y mejorar la satisfacción de los usuarios.** Además, crear software accesible supone abrir la puerta a un amplio abanico de usuarios que, de otro modo, quedarían excluidos del acceso a los datos hidrometeorológicos, promoviendo una mayor inclusión y equidad en el acceso a la información (Lazar, 2019; Nielsen, 2012).

## **1.2.4 TECNOLOGÍA**

En general, la elección de las tecnologías más adecuadas en el diseño de software es crucial para garantizar que, una vez desarrollada, la aplicación sea eficiente, segura y sostenible en el tiempo (Silberschatz, Korth, & Sudarshan, 2019). Las dos opciones tecnológicas más importantes a la hora de diseñar un Sistema de Gestión de Datos Climáticos son el tipo de base de datos y el lenguaje de programación, que, junto con otras decisiones tecnológicas, tanto primarias como secundarias, influyen enormemente en la escalabilidad y futuribilidad de la aplicación, dos conceptos que siempre es importante tener en mente si se quiere que el software tenga éxito a largo plazo.



Elegir el tipo adecuado de base de datos es crucial porque la base de datos es el corazón palpitante de una plataforma de gestión de datos climáticos (Elmasri & Navathe, 2019). La elección de la base de datos depende de las necesidades específicas de la aplicación y sus usuarios: en este caso, el software tiene que gestionar grandes cantidades de datos, y una base de datos relacional puede resultar la mejor opción, especialmente por la forma en que los datos hidrometeorológicos están estrechamente interconectados (Date, 2018). De hecho, las relaciones implementadas de forma nativa en este tipo de bases de datos pueden utilizarse para recrear este nivel de interconexión y complejidad, haciendo que la lógica de entrada y consulta de datos sea más coherente con el tipo de dominio funcional abordado (García-Molina, Ullman y Widom, 2009). Además, hay que tener en cuenta que en un CDMS la mayoría de los datos archivados tienen un componente temporal, es decir, están vinculados al instante concreto de tiempo en el que se registraron. Este factor enfrenta al desarrollo con la necesidad de introducir tecnologías específicamente diseñadas para facilitar el archivo y el análisis de series temporales (Levene, 2017). No todas las bases de datos relacionales tienen capacidades tan específicas, y la elección correcta puede mejorar significativamente el rendimiento de las aplicaciones y garantizar una experiencia de usuario fluida.

La elección del lenguaje de programación adecuado es igualmente importante. El lenguaje de programación determina en primer lugar la facilidad de desarrollo: los lenguajes más extendidos y fáciles de dominar conducen a un desarrollo más fluido, del mismo modo que ocurre lo contrario con los lenguajes más difíciles y con una distribución más especializada (Liu y Tretter, 2015). En segundo lugar, e igual de importante, el uso de un lenguaje más fácil de usar influye positivamente en la funcionalidad de la aplicación, su velocidad de ejecución y la facilidad de mantenimiento (Duan & Huang, 2021). Cada lenguaje de programación tiene sus pros y sus contras, y la elección del más adecuado depende de los requisitos específicos de la aplicación. Por ejemplo, si la aplicación requiere una alta velocidad de ejecución, un lenguaje de programación más cercano al lenguaje máquina podría ser la mejor opción. Por otro lado, si la aplicación requiere facilidad de desarrollo, un lenguaje de programación más extendido, con una gran comunidad de soporte -y disponibilidad de módulos y bibliotecas fácilmente implementables- puede ser la mejor opción. **Dado que los datos hidrometeorológicos suelen proceder de distintas fuentes y llegan al software en formatos diferentes, es importante elegir un lenguaje de programación que pueda manejar esta complejidad** (Zhao et al., 2020).

La escalabilidad y futuribilidad de la aplicación son igualmente importantes. La escalabilidad es importante para que la aplicación se adapte a las distintas necesidades operativas. Por ejemplo, una vez que el CDMS empieza a ser utilizado por cada vez más usuarios, puede ser necesario añadir nuevas funcionalidades para satisfacer las necesidades específicas de cada uno de ellos. Esto requeriría un aumento de los recursos informáticos y más potencia de procesamiento. Si la aplicación no se escala adecuadamente, el sistema puede no funcionar correctamente o ralentizarse, comprometiendo la calidad del análisis de datos (Guo et al., 2019). Además, la escalabilidad es importante para garantizar que la aplicación también pueda hacer frente a picos de tráfico repentinos, como durante fenómenos meteorológicos extremos.

En estas situaciones, la aplicación debe ser capaz de gestionar un gran volumen de solicitudes simultáneas sin experimentar ralentizaciones o caídas (Nag et al., 2017).

La futuribilidad es un concepto crucial en el diseño de un CDMS. En el ámbito de los servicios climáticos, la tecnología y los requisitos institucionales evolucionan constantemente (IPCC, 2018). En consecuencia, **el CDMS debe ser capaz de adaptarse a las nuevas tecnologías y a las necesidades de los usuarios a lo largo del tiempo** (OMM, 2019). Esto significa que la aplicación debe estar diseñada para integrarse fácilmente con otras herramientas y tecnologías emergentes, garantizando la máxima eficacia y calidad en el análisis de los datos hidrometeorológicos (OMM, 2019). Por ejemplo, si surgen nuevas herramientas o tecnologías que puedan mejorar el análisis de los datos climáticos, el CDMS debe ser capaz de integrar estas nuevas herramientas y utilizarlas de forma sinérgica, de modo que pueda ofrecer al usuario una visión completa y precisa de los datos climáticos (ECMWF, 2016). **Además, el CDMS debe diseñarse para garantizar la interoperabilidad con otros sistemas y aplicaciones, de modo que los datos climáticos puedan intercambiarse y compartirse fácilmente con otros usuarios o instituciones.** Esta característica es especialmente importante en una época en la que la colaboración entre instituciones y organizaciones es cada vez más esencial para hacer frente a los desafíos climáticos mundiales (OMM, 2019).

## 1.3 METODOLOGÍA

Esta sección del documento tiene como objetivo describir la metodología adoptada en el diseño de un Sistema de Gestión de Datos Climáticos (SGDC) para la República Dominicana. Este sistema está diseñado para gestionar los datos hidrometeorológicos producidos en el país como base de apoyo para la ordenación del territorio y la gestión de los riesgos del cambio climático.

El diseño comenzó con la construcción de un punto de referencia entre cuatro plataformas identificadas como los principales sistemas de referencia por el anuncio del proyecto<sup>7</sup>. El objetivo es identificar los requisitos mínimos de funcionalidad e interacción que deberá cumplir la futura plataforma CDMS en la República Dominicana. Esta evaluación comparativa se elaboró utilizando un enfoque de benchmarking, que permitió identificar los puntos fuertes y débiles de los sistemas analizados, con el objetivo de detectar oportunidades para mejorar las plataformas existentes o construir una nueva que responda mejor a las necesidades del proyecto y de las partes interesadas.

---

<sup>7</sup> Tres de las plataformas (Colombia, Chile y México) fueron identificadas directamente por el anuncio del proyecto. El cuarto (Belice) identificado por el equipo de desarrollo.

Tras analizar los resultados de la evaluación comparativa, el equipo del proyecto sugirió requisitos funcionales que tienen en cuenta tanto la evaluación comparativa entre las plataformas como otras consideraciones sobre la funcionalidad necesaria en el escenario específico de la República Dominicana. Uno de los criterios clave a la hora de seleccionar la plataforma es el uso de software de código abierto, que se considera ideal por su capacidad inherente de personalizarse y adaptarse a las necesidades específicas de un proyecto con mucha más facilidad que el software de código cerrado.

A continuación, se identificó una plataforma CDMS existente afiliada al proyecto OpenCDMS para utilizarla como base para construir la futura plataforma CDMS para la República Dominicana. El proceso de selección consistió en establecer un punto de referencia entre las soluciones candidatas, y la elección final se basó en parámetros tecnológicos, accesibilidad y personalización, y cumplimiento de las normas internacionales de referencia.

Una vez seleccionada la plataforma, el grupo de trabajo realizó un análisis para determinar qué áreas necesitaban más mejoras y con qué prioridad debían aplicarse. Así pues, se establecieron dos niveles de prioridades de desarrollo: obligatorias y optativas. Las mejoras obligatorias se consideraban indispensables y no negociables. Las mejoras opcionales se han dividido en módulos autónomos, cada uno independiente de los demás y que pueden desarrollarse e implantarse por separado. Este planteamiento permitió una mayor flexibilidad en el proceso de desarrollo y la posibilidad de priorizar las mejoras en función de las necesidades y las reacciones de las partes interesadas.

### **1.3.1 REQUISITOS FUNCIONALES**

Los requisitos funcionales son una parte crucial del análisis de requisitos de un sistema de software, como se destaca en varias fuentes en el campo de la ingeniería de software (Pressman, 2014; IEEE Std 830-1998; Cockburn, 2000; Sommerville, 2016). Según Pressman (2014), los requisitos funcionales definen las actividades que el sistema debe ser capaz de realizar y los resultados que debe ofrecer, formando así el núcleo del sistema y definiendo lo que el sistema debe hacer para satisfacer las necesidades de los usuarios y las partes interesadas.

La definición de los requisitos funcionales puede expresarse de varias formas, como por ejemplo mediante el uso de diagramas de casos de uso (Cockburn, 2000). Estos diagramas permiten representar la funcionalidad del sistema y las interacciones con los usuarios y otros sistemas. Sin embargo, el uso de diagramas de casos de uso debe ir acompañado de una descripción textual de los requisitos funcionales para garantizar que son completos y precisos. Además, los requisitos funcionales deben ser inequívocos y coherentes para garantizar que el sistema de software satisfaga las necesidades de los

usuarios y las partes interesadas durante todo el ciclo de vida del software (Sommerville, 2016).

La correcta definición de los requisitos funcionales es esencial para el desarrollo de un sistema de software que satisfaga las necesidades de los usuarios y las partes interesadas, como se destaca en diversas fuentes en el campo de la ingeniería de software (Pressman, 2014; Kotonya & Sommerville, 1998). La falta de comprensión de los requisitos funcionales puede dar lugar a un producto de software que no funcione según lo previsto o no satisfaga las necesidades del usuario. Además, están sujetos a posibles variaciones, cambios y ajustes en el transcurso de la implantación -generalmente tras la finalización del diseño y la primera fase de implantación- para garantizar que el sistema de software satisface las necesidades de los usuarios y las partes interesadas a lo largo de todo el ciclo de vida del software (Kotonya & Sommerville, 1998). En resumen, la definición precisa y completa de los requisitos funcionales es un paso fundamental en la realización de un sistema de software.

### **1.3.1.1 ANÁLISIS DEL CONTEXTO**

En informática, el análisis de contexto es una metodología utilizada para comprender el contexto específico en el que se utilizará un sistema o una aplicación. Esto incluye tanto el análisis del entorno tecnológico en el que funcionará el sistema como los requisitos de los usuarios y de funcionalidad más amplia.

Este tipo de análisis es una parte esencial del proceso de diseño del sistema, porque ayuda a identificar los requisitos del sistema y a comprender las peculiaridades específicas que habrá que desarrollar para que el sistema se integre plenamente en el entorno para el que se está desarrollando (Sommerville, 2016). Además, es importante porque ayuda a crear una visión global del problema que se pretende resolver y a desarrollar una solución que satisfaga las necesidades de todas las partes interesadas (Booch et al., 2007).

Por lo tanto, es necesario analizar el contexto específico de la República Dominicana, centrándose en la fragmentación de las estaciones/plataformas, cuya propiedad está dividida entre varias entidades, instituciones que todas o casi todas dependen de tecnologías diferentes y sistemas heterogéneos de recopilación de datos.

### **1.3.1.2 ENCUESTA A LAS PARTES INTERESADAS**

Una encuesta a las partes interesadas, también conocida como sondeo de partes interesadas, es un método esencial para recabar información y opiniones de todas las

partes implicadas en un proyecto. Según la definición de Freeman, una parte interesada es cualquier grupo o individuo que puede influir o verse influido por la consecución de los objetivos de la organización (2010). El principal objetivo de una encuesta a las partes interesadas es conocer a fondo las opiniones, expectativas, necesidades y preocupaciones de cada una de ellas. Esto puede ayudar a identificar los puntos fuertes y débiles del proyecto o de la organización, así como las áreas susceptibles de mejora. La importancia de implicar a las partes interesadas adecuadas en las decisiones de desarrollo de los proyectos es considerable y se pone de relieve en numerosos estudios (Clarkson, 1995; Mitchell et al., 1997; Johnson et al., 2008; Freeman, 2010).

En el contexto del diseño y desarrollo de un Sistema de Gestión de Datos Climáticos, las partes interesadas pueden ser múltiples y complejas, con intereses y objetivos diferentes. Por lo tanto, es crucial implicar a las partes interesadas adecuadas durante la fase de diseño y desarrollo del sistema para garantizar que se satisfacen sus necesidades y que el sistema es útil y sostenible a lo largo del tiempo. En este caso, se prestó especial atención a la situación de la propiedad de las emisoras/plataformas, que está fragmentada entre muchas instituciones y organizaciones diferentes. Se procuró que esta particularidad del escenario dominicano no se convirtiera en un punto débil para el desarrollo de un CDMS adecuado. Basándose en los resultados de estudios anteriores (AFD, 2021), la encuesta a las partes interesadas se llevó a cabo mediante entrevistas específicas, cuyos detalles figuran en el Producto 2 del proyecto "Diagnóstico de la situación actual". Este documento se centrará exclusivamente en los elementos clave para el diseño del CDMS.

### **1.3.1.3 EVALUACIÓN COMPARATIVA DE LAS PLATAFORMAS EXISTENTES**

La evaluación comparativa entre dos o más plataformas puede definirse como el proceso de evaluar comparativamente la funcionalidad o el rendimiento de una plataforma frente a otras, en función de criterios o requisitos previamente establecidos (Camp, 2005). Este proceso permite determinar los puntos fuertes y débiles de los sistemas analizados y, en consecuencia, identificar oportunidades para mejorarlos o construir una alternativa propia que responda mejor a las necesidades del comisario y de las partes interesadas.

La evaluación comparativa puede utilizarse para mejorar la calidad y la productividad del servicio identificando los procesos más eficientes utilizados por plataformas diseñadas para fines similares y adoptándolos en el propio entorno operativo para reducir costes y mejorar la productividad (Taticchi et al., 2015). Así pues, un análisis comparativo realizado de este modo es fundamentalmente una herramienta estratégica para la mejora continua, especialmente en lo que respecta a la adopción de las mejores prácticas (y la evitación de las peores prácticas o criticidades más en general)

en un sector determinado, en el caso de este documento para la creación de un Sistema de Gestión de Datos Climáticos.

La evaluación comparativa se utilizó en dos fases distintas durante el proceso de redacción de este documento:

1. Identificación y definición de las principales funcionalidades de las plataformas nacionales de Colombia, Chile y México, definidas como puntos de referencia mínimos en los entregables del proyecto. La plataforma de Belice se añadió a la evaluación comparativa porque se identificó como una opción adicional que cumple (y supera) todos los requisitos mínimos de entrega. Este punto de referencia se utiliza para definir los requisitos (funcionales y no funcionales) del proyecto y crear un punto de partida para identificar las funcionalidades esenciales de la futura plataforma dominicana. Además, el análisis de las cuatro plataformas se utilizó como base para la construcción del módulo gráfico opcional (UX/UI) para la futura plataforma CDMS en la República Dominicana.
2. Identificación de una plataforma CDMS existente afiliada al proyecto OpenCDMS, que se utilizará como base para construir la futura plataforma CDMS para la República Dominicana. La elección final se basará en parámetros tecnológicos, accesibilidad y personalización, así como en el cumplimiento de las normas internacionales de referencia.

### **1.3.2 REQUISITOS NO FUNCIONALES**

Los requisitos no funcionales definen las propiedades del sistema necesarias para garantizar una determinada calidad de servicio. Representan aspectos críticos del sistema que no pueden pasarse por alto en el diseño y la implementación, porque afectan directamente a la satisfacción del usuario, la eficacia del producto y la aceptación del mercado (Sikora, Nowak, & Czabanowska, 2019).

Los requisitos no funcionales pueden dividirse en varias categorías, como los requisitos de rendimiento (por ejemplo. tiempo de respuesta, velocidad de procesamiento, rendimiento), a la seguridad (por ejemplo. protección de datos, autenticación, autorización), a la usabilidad (por ejemplo. facilidad de uso, ergonomía, accesibilidad), escalabilidad (p. ej. gestión de la carga, adaptación a los cambios en el volumen de datos), mantenibilidad (por ejemplo. modularidad, extensibilidad, documentación), compatibilidad (p. ej. interoperabilidad, cumplimiento de estándares) y otros (Tantar, Berekmeri, & Miclea, 2018).

La gestión de requisitos no funcionales requiere el uso de técnicas y herramientas específicas, como modelización, evaluación, verificación y validación. Estas técnicas pueden utilizarse para medir la calidad del sistema, detectar posibles problemas y definir

criterios de aceptación de los requisitos. Además, es importante tener en cuenta los requisitos no funcionales durante todo el ciclo de vida del producto, desde la fase de diseño hasta la de prueba y mantenimiento, para garantizar que se alcanzan los objetivos de calidad fijados (Laplante, 2017).

En resumen, los requisitos no funcionales son esenciales para garantizar que el sistema satisfice las necesidades de usuarios y partes interesadas en términos de calidad, rendimiento, seguridad y usabilidad. Su gestión requiere el uso de técnicas específicas y su atención debe mantenerse durante todo el ciclo de vida del producto (Gacek & Arief, 2019; Lyytinen & Kautz, 2019).

### **1.3.3 TECNOLOGÍA PARA EL CDMS DE LA REPÚBLICA DOMINICANA**

Aunque el tema ya ha sido introducido y discutido en la sección 1.2.4, es importante considerar más a fondo la elección de la tecnología apropiada para el diseño de un sistema de gestión de datos hidrometeorológicos (CDMS) en el contexto específico de la República Dominicana. En este país, la gestión de los datos hidrometeorológicos está fragmentada y confiada a numerosas organizaciones de distinto tamaño, tanto públicas como privadas, con intereses diversos y sistemas y procesos tecnológicos completamente diferentes.

En consecuencia, **se sugiere encarecidamente el uso de una plataforma de código abierto, que pueda personalizarse más fácilmente en respuesta a las necesidades del proyecto, y, al mismo tiempo, la inclusión del futuro sistema CDMS dominicano en el proyecto OpenCDMS, y por tanto indirectamente en el sistema internacional regido por la OMM.** Esta elección tiene en cuenta el contexto local, donde la fragmentación de los sistemas de almacenamiento y recogida de datos es compleja de gestionar. La inclusión bajo un paraguas internacional como éste sirve tanto de garantía de calidad como de reaseguro para todas las partes implicadas, desde los directores hasta las organizaciones que gestionan actualmente los datos.

Además de estas consideraciones, **es crucial garantizar que la integración de esta futura plataforma en el paisaje local se produzca de la forma más fluida posible, superando posibles disputas relacionadas con la gobernanza del sistema u otras cuestiones derivadas de la fragmentación de intereses relacionada con la diferente propiedad de los sistemas de almacenamiento y recopilación de datos. Para ello, debe implantarse un sistema de aislamiento de datos completo y eficaz, con un protocolo de autorización adicional para proteger los intereses de todas las partes implicadas.**

En general, la consecución de los objetivos del proyecto depende, por tanto, de la capacidad de gestionar eficazmente estos retos y garantizar la integración armoniosa

del CDMS en el contexto específico de la República Dominicana. El resto de las opciones específicas adoptadas para alcanzar este objetivo se analizan detalladamente a lo largo de este documento.

## **1.3.4 ENFOQUES DE DESARROLLO**

Antes de entrar en los detalles del trabajo que se realizará, es importante dedicar unos párrafos a hablar de los enfoques de desarrollo que utilizará el grupo de trabajo. Un enfoque estructurado es esencial para garantizar una implantación eficaz y segura del software, sobre todo cuando se trata de software de código abierto en el que las contribuciones pueden proceder de diversas fuentes.

### **1.3.4.1 AGILE Y SCRUM**

Las metodologías ágiles ofrecen un marco sólido para gestionar proyectos complejos, permitiendo una planificación y coordinación eficaces de las actividades del proyecto. Agile es una filosofía de desarrollo de software que promueve el trabajo colaborativo, la adaptación a las necesidades cambiantes del proyecto y la entrega continua de funcionalidad.

Scrum es uno de los marcos ágiles más utilizados y consiste en un enfoque iterativo e incremental de la gestión de proyectos. En Scrum, el trabajo se organiza en "sprints", que son periodos de tiempo claramente definidos en los que el equipo se centra en la entrega de una característica o conjunto de características. Las actividades que se llevarán a cabo durante un sprint se determinan durante una reunión de planificación, en la que el equipo comparte lo aprendido en el sprint anterior y acuerda los objetivos del siguiente.

Durante el sprint, el equipo se reúne a diario en una reunión denominada "scrum diario" para debatir el progreso del trabajo e identificar cualquier impedimento. Al final del sprint, se organiza una revisión en la que el equipo presenta lo que ha desarrollado durante el sprint y evalúa su trabajo.

Scrum también incluye un papel de "propietario del producto", que representa al cliente o, en este caso, al desarrollador del proyecto, responsable de definir los requisitos y priorizar las funcionalidades que deben desarrollarse. De este modo, el equipo tiene una comprensión compartida de los requisitos y objetivos del proyecto y puede trabajar de forma eficaz y coordinada.



### **1.3.4.2 PRUEBAS AUTOMÁTICAS**

Para preservar la integridad y el buen funcionamiento del software de código abierto, es crucial aplicar pruebas automatizadas exhaustivas y sin regresiones, como las pruebas funcionales y de extremo a extremo (e2e), prestando especial atención a los detalles. Estas pruebas permiten detectar cualquier regresión o mal funcionamiento del software durante la fase de desarrollo, garantizando que los nuevos cambios introducidos no tengan efectos negativos en el software existente.

Automatizar la ejecución de estas pruebas en cada iteración del código permite verificar de forma continua y sistemática la calidad del software, minimizando el riesgo de error humano y garantizando que el software siga siendo estable, fiable y funcional a lo largo del tiempo. Además, estas pruebas permiten detectar y resolver a tiempo cualquier problema o anomalía antes de que el software se ponga a disposición de los usuarios finales, lo que garantiza una mejor experiencia de uso y reduce los costes de mantenimiento a largo plazo.

### **1.3.4.3 INTEGRACIÓN CONTINUA/IMPLANTACIÓN CONTINUA (CI/CD)**

La adopción de un enfoque de Integración Continua/Despliegue Continuo (CI/CD) es crucial para garantizar la integridad y calidad del software de código abierto. El CI/CD prevé la automatización de varias etapas del ciclo de desarrollo de software, como la compilación del código, la ejecución automática de pruebas y la liberación, lo que permite verificar inmediatamente los cambios realizados antes de integrarlos en el software principal.

La automatización de la compilación del software permite generar automáticamente versiones ejecutables del mismo, lo que permite detectar a tiempo errores de compilación o la falta de dependencias. La ejecución automática de las pruebas comprueba si el código modificado ha causado regresiones o problemas de funcionalidad, lo que permite identificar y corregir rápidamente cualquier defecto antes de su publicación.

La publicación automatizada de software, combinada con la compilación y las pruebas automatizadas, garantiza un flujo de publicación rápido y fiable. Esto significa que las correcciones de errores, las nuevas funciones o las mejoras del software pueden publicarse de forma rápida y segura, reduciendo el riesgo de acumulación de errores y simplificando el proceso de implantación.

El enfoque CI/CD garantiza que el software de código abierto publicado sea siempre estable y funcional, reduciendo el riesgo de errores críticos o fallos de funcionamiento. Además, la pronta identificación y corrección de los problemas ayuda a evitar daños en el software y garantiza una mejor experiencia del usuario.

## 2 PLATAFORMAS DE REFERENCIA

Este análisis se deriva directamente de las solicitudes de los entregables del proyecto, y pretende esbozar las funcionalidades ofrecidas por tres plataformas nacionales definidas en la licitación como términos mínimos e indispensables de comparación: la de Colombia<sup>[8]</sup>, la de Chile<sup>[9]</sup> y la de México<sup>[10]</sup>. A estas tres plataformas se une la plataforma nacional de Belice, ya que se ha identificado como una opción más que satisface con creces todos los requisitos mínimos de entrega. El objetivo último de este análisis es, por tanto, la creación de una línea base que defina los requisitos mínimos a cumplir para la posterior propuesta de implantación y desarrollo de una plataforma CDMS para la República Dominicana.

La evaluación comparativa se llevó a cabo principalmente mediante un análisis exploratorio de las páginas web puestas a disposición por las plataformas nacionales de los Estados enumerados en el apartado anterior, utilizando y simulando el papel de un usuario final, es decir, un usuario básico que pretende utilizar las herramientas que ofrece la plataforma.

También se contactó con las instituciones encargadas de las plataformas para obtener información adicional sobre la infraestructura de almacenamiento y recuperación de datos. La principal pregunta que había que responder era si el almacenamiento estaba centralizado o no, es decir, si las plataformas web consultaban una base de datos central para recuperar y visualizar los datos. El sistema colombiano resultó ser híbrido: a través de un servicio web, se realiza una consulta automática en momentos determinados a las bases de datos de las distintas entidades propietarias de los datos, para tenerlos disponibles -y archivados- en la plataforma central. En cambio, el sistema de Belice está centralizado y se examinará en detalle en los capítulos 4 y 5.

En cuanto a las otras dos plataformas -la chilena y la mexicana-, no se pudo obtener información sobre la política de almacenamiento de datos. La plataforma chilena depende de un servicio de gestión privado, mientras que la mexicana se gestiona con MCH, un CDMS que también se analizará con más detalle en el capítulo 4.

---

<sup>8</sup> [Consulta y Descarga de Datos Hidrometeorológicos \(ideam.gov.co\)](http://ideam.gov.co)

<sup>9</sup> [EXPLORADOR CLIMÁTICO \(cr2.cl\)](http://cr2.cl)

<sup>10</sup> [Información Estadística Climatológica \(conagua.gob.mx\)](http://conagua.gob.mx)

## 2.1 COLOMBIA

La primera funcionalidad importante de la plataforma colombiana es permitir la consulta y descarga de datos meteorológicos e hidrológicos recogidos en su territorio.

La interfaz gráfica presenta un mapa de Colombia marcado con marcadores circulares de diferentes tamaños, dependiendo de si la zona que marcan contiene más o menos estaciones/plataformas. Estos marcadores subdividen el mapa en macrozonas de forma natural y sobre ellos se superpone el número total de estaciones de la zona de referencia (representada gráficamente por el marcador circular). Estas zonas de referencia se hacen más o menos extensas en función del nivel de zoom fijado por el usuario; acercar o alejar la imagen sigue siendo la forma más inmediata de consultar el mapa a nivel puramente gráfico.

En cambio, para ver los datos directamente en el mapa, debe utilizar los filtros incluidos en el menú *Consulta*, donde puede seleccionar el periodo que le interesa, los parámetros y variables deseados y las estaciones cuyos datos desea ver. Los datos filtrados previamente se pueden exportar a través de la función de *descarga* o según la estación/plataforma en la que se tomaron a través de la función de menú denominada *Información especial*. Por último, es posible descargar recursos -como, por ejemplo, el catálogo nacional de estaciones u otras entidades y el glosario de variables- gracias a la funcionalidad *Recursos*. La interacción entre el usuario y la plataforma colombiana tiene lugar íntegramente dentro de una única página web.

## 2.2 CHILE

También en lo que respecta a la plataforma chilena, la principal funcionalidad es la consulta de datos hidrometeorológicos desde el mapa, de la forma más intuitiva posible.

La visualización de los datos en el mapa se hace, de nuevo, dinámica en función de los filtros establecidos por el usuario. Los marcadores correspondientes a cada estación del territorio adoptan una coloración diferente en función tanto de la variable seleccionada como de su valor absoluto medido por la estación/plataforma correspondiente. En términos más generales, el sistema prevé que la consulta pueda devolver tanto datos meteorológicos como hidrológicos, en función del tipo de estación y, por supuesto, de las variables filtradas por el usuario.

La plataforma también pone a su disposición un gráfico -un histograma para ser precisos- situado debajo del mapa. Este gráfico se rellena y mejora en función tanto de la estación seleccionada en el mapa como de los siguientes filtros adicionales:

- Variable (precipitaciones, temperatura media, caudal, etc.);
- Periodo de agregación temporal (mensual o anual);
- Periodo de tiempo (a partir de 1940);
- Estadísticas que se mostrarán en el mapa (medias, extremos, etc.) - esta selección es una de las que hace que cambie el color de los marcadores de las estaciones que se muestran en el mapa;
- Anomalías, es decir, la visualización y notificación de cualquier diferencia con la media histórica (la media de los últimos 30 años de encuestas). Esta selección también cambia el color de los marcadores de estación que aparecen en el mapa;
- Series temporales: selección y descarga del gráfico temporal (histograma) en función de todos los filtros seleccionados por el usuario.

En conclusión, al igual que en el caso de la plataforma colombiana, todas las interacciones con la plataforma chilena tienen lugar dentro de una única página web.

## 2.3 MÉXICO

Al igual que en las dos plataformas analizadas anteriormente, en el caso de la plataforma mexicana, todas las funciones pueden consultarse en una única página web. La interfaz, sin embargo, es diferente: en este caso, las distintas funciones disponibles se seleccionan mediante un menú desplegable.

La consulta de los datos del estudio hidrometeoroclimático se basa de nuevo en la consulta intuitiva de mapas y gráficos. En el caso de México, sin embargo, estas dos características no son dinámicas ni se generan en el momento de la consulta. Los gráficos y mapas son, de hecho, el resultado de elaboraciones realizadas por el CDMS de forma autónoma, sólo para ser consultados para su lectura en el portal web y visualizados por el usuario si así lo desea. Para mayor comprensión, cuando hablamos de mapas nos referimos a *mapas térmicos* y mapas poblados de datos, no al mapa genérico en el que se muestran indicadores que representan la ubicación geográfica de las estaciones.

Las funcionalidades que presenta la plataforma mexicana son las siguientes:

- Mapa térmico diario basado en datos de temperatura y precipitaciones;
- Mapa térmico mensual basado en datos de temperatura y precipitación, dividido según el año de medición que el usuario desee consultar;
- Mapa con información sobre las estaciones climáticas del territorio nacional (ubicación, información básica sobre cada estación, enlace a los datos estadísticos recopilados a partir de los resultados de la estación);

- Lista de estaciones por región. Los valores medios anuales y los extremos mensuales o anuales pueden exportarse desde la lista;
- Mapa de calor histórico, basado en datos recogidos en el periodo 1981-2010 para cada variable disponible;
- Climogramas de cada estación, seleccionables directamente desde el mapa;
- Seguimiento de la sequía con valores visibles en un mapa (mapa de calor) y un gráfico (histograma, en porcentajes);
- Índice de precipitación mensual (SPI) sobre el territorio nacional, disponible en intervalos de tiempo de 1 a 24 meses;

Además de las funciones mencionadas, todos los mapas pueden exportarse y descargarse en formato KMZ.

## 2.4 BELICE

La plataforma de Belice presenta como página principal un mapa interactivo<sup>[1]</sup>, poblado de marcadores que indican las estaciones registradas en el sistema. Algunas funciones -junto con los informes instantáneos sobre determinadas variables- están disponibles directamente haciendo clic en los marcadores de las estaciones, mientras que otras son accesibles desde un menú emergente situado a la izquierda de la pantalla. Los datos disponibles para consulta son tanto meteorológicos como hidrológicos y climáticos.

Las funcionalidades más relevantes de la plataforma pueden dividirse y clasificarse del siguiente modo:

- Mapa interactivo: al hacer clic en una estación se muestran los datos climáticos (último sondeo, últimas 24 horas), un enlace a los datos de la estación y los datos climáticos de la estación (*Informe de la estación*). Además, el mapa puede filtrarse en función de los tipos de estaciones que se deseen visualizar o modificarse mediante *superposiciones* según parámetros preseleccionados (incluidas cuencas hidrográficas, distritos y aguas nacionales);
- Sección *Estaciones*: sección desde la que se puede acceder a informes sobre las variables registradas por estaciones individuales y a los metadatos de dichas estaciones;
- Sección de *datos*: lista de datos diarios de las estaciones, filtrados por mes seleccionado; lista de datos horarios de las estaciones, filtrados por día seleccionado; exportación de datos (en formato .csv) según los filtros seleccionados (distrito, estaciones, variables, intervalo de tiempo);

---

<sup>1</sup> Hay tres posibles opciones de mapas: Open Street Map, World Imagery y World Street Map. El mapa por defecto es Open Street Map.

- Sección de control de *calidad*: selección y personalización de los umbrales que se utilizarán para el *control de calidad de los datos*;
- Sección *Productos*: Sección en la que es posible obtener informes procesados en tiempo real sobre los datos deseados, filtrados por estación, variable e intervalo de tiempo, con posibilidad de comparar estaciones y variables entre sí y generación automática de gráficos para acompañar a los informes. En esta sección también es posible generar análisis espaciales en un mapa según la variable seleccionada y la escala de colores deseada. Como última característica de la sección, existe la posibilidad de obtener un informe sobre los valores medios y extremos de las variables medidas (filtrados por mes y estación).

Como consideración adicional, cabe destacar que los mapas y gráficos son dinámicos, lo que los hace personalizables según las necesidades personales.

## **3 REQUISITOS FUNCIONALES**

Los requisitos funcionales constituyen una parte esencial del análisis de requisitos de un sistema informático, ya que definen las actividades que debe realizar el sistema para satisfacer las necesidades de los usuarios y las partes interesadas. Se centran en la funcionalidad del sistema, pero también pueden incluir requisitos de rendimiento, seguridad, fiabilidad y facilidad de uso. Una definición precisa de los requisitos funcionales es crucial para garantizar el desarrollo de un sistema informático que funcione correctamente y satisfaga las necesidades de usuarios y partes interesadas. La falta de comprensión de los requisitos funcionales puede dar lugar a un producto de software que no funcione como se espera o que no tenga en cuenta determinados requisitos funcionales importantes. Además, los requisitos funcionales están sujetos a variaciones y cambios a lo largo del ciclo de vida del software para garantizar que el sistema de software siga satisfaciendo las necesidades de los usuarios y las partes interesadas (Sommerville, 2016; Kotonya y Sommerville, 1998).

En el caso concreto de la redacción de este documento, la definición de los requisitos funcionales se basó en el análisis de las plataformas definidas en el anuncio del proyecto como términos mínimos e indispensables de comparación, en el sentido de que las funcionalidades presentadas por los proyectos de referencia son también un requisito incuestionable de este proyecto. Sin embargo, el caso de uso específico de la República Dominicana tiene requisitos funcionales adicionales<sup>[12]</sup>, mientras que otras características también figuran en secciones posteriores del documento<sup>[13]</sup>.

### **3.1 ANÁLISIS PRELIMINAR**

El objetivo del proyecto es crear una plataforma nacional para la gestión de datos hidrometeorológicos en la República Dominicana. En la actualidad, estos datos son gestionados por varias organizaciones, cada una de ellas con técnicas y herramientas de recogida y almacenamiento diferentes. El contexto en el que tendrá que operar el sistema es, por tanto, muy heterogéneo, y uno de los principales objetivos es recopilar estos datos en una plataforma única, tanto para su gestión como para su uso.

La primera fase del estudio preliminar incluye un análisis cualitativo de las partes interesadas para esbozar el contexto general en el que se introducirá el sistema, centrándose principalmente en el modo en que las distintas entidades gestionan los datos hidrometeorológicos. En cambio, los próximos pasos se centrarán en

---

<sup>12</sup> Sección 3.2

<sup>13</sup> Capítulo 5



comprender cómo otras plataformas similares satisfacen la demanda de visualización y consulta de datos, según los requisitos de la convocatoria del proyecto. La elaboración de estos análisis conducirá a la definición de los requisitos funcionales del sistema.

### 3.1.1 ENCUESTA A LAS PARTES INTERESADAS

Una encuesta a las partes interesadas es un instrumento muy útil para recabar información y opiniones de los participantes en un proyecto. Este tipo de análisis permite evaluar las necesidades y expectativas de las partes interesadas para definir mejor las prioridades y estrategias del proyecto y, en consecuencia, alcanzar sus objetivos con eficacia y eficiencia.

Las partes interesadas son una categoría muy amplia de agentes y pueden incluir, por ejemplo, individuos, organizaciones, empresas y otras instituciones que pueden verse afectadas o influir en el proyecto en cuestión. En el caso concreto del proyecto de gestión de datos hidrometeorológicos en la República Dominicana, la encuesta se realizó al personal responsable de la gestión de datos en las distintas entidades de producción que operan en el país. Se eligieron estos sujetos porque representan una fuente fiable de información sobre los datos gestionados y las metodologías utilizadas para recopilarlos y almacenarlos.

El objetivo de la encuesta era, por tanto, obtener una imagen precisa del contexto general en el que tendrá que operar el sistema, con el fin de definir los requisitos funcionales del proyecto. La información recabada a través de la encuesta permitió así identificar las necesidades de las partes interesadas, evaluar sus expectativas y definir con mayor precisión algunas de las prioridades del proyecto.

**Cuadro 2. Encuesta a las partes interesadas - requisitos funcionales.** Cada fila del cuadro corresponde a una institución diferente a la que se dirigieron las preguntas de la encuesta a las partes interesadas. Las entidades no se mencionan por su nombre, ya que la información es superflua a efectos del análisis realizado.

Papel del personal entrevistado	Procedimiento de introducción de datos en la base de datos	Tipo de datos gestionados por la BD	Tipo de estaciones/plataformas gestionadas por el Instituto
Interior IT Ayuda	Envío de fotos de un formulario en papel por Whatsapp.	Precipitación Temperatura	Climatología

Papel del personal entrevistado	Procedimiento de introducción de datos en la base de datos	Tipo de datos gestionados por la BD	Tipo de estaciones/plataformas gestionadas por el Instituto
IT	<p>Recuperación de archivos de datos por telemetría. Se utiliza un datalogger <i>Davis</i> para la comunicación entre la estación y el PC (con el software <i>Davis</i> instalado) llevado al campo ad hoc.</p> <p>Las estaciones no guardan datos directamente, o guardan muy pocos. La periodicidad de la recuperación de datos es variable, por lo que hay lagunas en los datos.</p> <p><i>Las estaciones son del tipo Davis Instrument Vantage Pro.</i></p>	Evapotranspiración Metadatos Horas de sol Precipitación Presión Radiación UV Temperatura Humedad Viento	Meteorología agrícola
Interior IT Ayuda	Recogida de datos sobre el terreno y posterior mecanografiado y archivado en la oficina.	Horas de sol Precipitación Presión Temperatura Humedad Viento	Climatología Meteorología agrícola Meteorología aeronáutica Sinópticos de superficie
Interior IT Ayuda	Recogida de datos sobre el terreno y posterior mecanografiado y archivado en la oficina.	Horas de sol Precipitación Presión Temperatura Humedad Viento Más	Climatología Meteorología agrícola Meteorología aeronáutica Sinópticos de superficie
Interior IT Ayuda	Rellenando la API de MCH directamente o mediante telemetría.	Metadatos Horas de sol Precipitación Presión Temperatura Humedad Viento	Climatología Meteorología agrícola Meteorología aeronáutica Sinópticos de superficie
Admin de plataforma Interior IT Ayuda	<p>Las estaciones transmiten datos en formato de texto, a través del teléfono móvil, a la app <i>Davis WeatherLink</i>.</p> <p>A continuación, los datos se extraen de WL y se introducen en una base de datos privada con una aplicación especialmente creada dentro de CLIMARED (programa informático de gestión). El Instituto tiene todo almacenado en un servidor local, pero también tiene una réplica en la nube para permitir el acceso a los datos fuera de la red local.</p>	Metadatos Horas de sol Precipitación Presión Temperatura Humedad Viento	Climatología Meteorología agrícola

Papel del personal entrevistado	Procedimiento de introducción de datos en la base de datos	Tipo de datos gestionados por la BD	Tipo de estaciones/plataformas gestionadas por el Instituto
Interior IT Ayuda	Los datos se almacenan directamente en <i>Davis WeatherLink</i> desde las estaciones/plataformas. No tiene su propia base de datos.  Se intenta recoger datos periódicamente para almacenarlos localmente. La recaudación no es totalmente automática.	Horas de sol Precipitación Presión Radiación UV Temperatura Humedad Viento	Climatología Meteorología agrícola
Interior IT Ayuda	Conexión directa a las estaciones y almacenamiento en base de datos propia sin pasar por <i>WeatherLink</i> . En contradicción con lo que han explicado otros institutos usuarios de la estación <i>Davis</i> .  La base de datos se aloja en un servidor local. No está claro si hay réplicas o copias de seguridad.  La base de datos se actualiza cada cinco minutos.	Horas de sol Precipitación Presión Temperatura Humedad Viento	Climatología Datos sobre el suelo Meteorología agrícola
Interior IT Ayuda	Los datos se transmiten a la plataforma en línea y se descargan en un servidor local cada cinco minutos.  La visualización en un servidor local se realiza a través del software <i>Davis</i> .  Las encuestas no incluyen muchas variables.	Precipitación Presión Temperatura Humedad Viento Más	Climatología Meteorología agrícola
Interior IT Ayuda	Como respuesta anterior.	Precipitación Temperatura Más	Climatología
Interior IT Ayuda	Forma parte de la red INDRHI. Hay uno que funciona con detección manual por parte de un observador, transmitiendo los datos a un operador que los digitaliza en un archivo Excel. Funciona en paralelo con una estación automática, que transmite la telemetría directamente al INDRHI.	Precipitación Temperatura Más	Climatología
Interior IT	Red de 95 pluviómetros. La recogida de datos es manual, desde una unidad llamada torre de control. Información enviada por WhatsApp diariamente (a las 6 de la mañana), inicialmente en un archivo Excel. A continuación, estos datos se envían a Biosalc, un programa informático agrícola, para su almacenamiento.  La estación climática (automática), por su parte, envía los datos directamente al productor, pero también hace una recogida local. La cronología está disponible en Lynks.	Balance hídrico Horas de sol Precipitación Presión Temperatura Texturas del suelo Humedad Viento	Climatología
Interior IT Ayuda	Estaciones <i>Davis</i> .	Más	Climatología Meteorología agrícola

Papel del personal entrevistado	Procedimiento de introducción de datos en la base de datos	Tipo de datos gestionados por la BD	Tipo de estaciones/plataformas gestionadas por el Instituto
Interior IT Ayuda	Transmisión de datos a través de la red telefónica.	Nivel hidrométrico	Estación hidrométrica
Interior IT Ayuda	<p>En los años ochenta existía Clipper Summer'87 guardado en una base de datos dBase.</p> <p>En 2014, se implementó un sistema global, denominado SNIH, para visualizar la red de telemetría (78 estaciones automáticas), con almacenamiento de datos en SQLite (dos años).</p> <p>Debido a la falta de inversión, se dañó el suministro eléctrico (había una reserva en Santiago). Las estaciones enviaban datos, pero éstos no podían guardarse en los servidores. Debido a este problema, los datos se almacenaron manualmente, pero algunas series se perdieron porque no podían recibir datos telemétricos en tiempo real.</p> <p>Los datos de estas estaciones se envían por satélite (GOES) y se conservan durante unos días; después, esta información se sustituye.</p> <p>El SNIH funcionó durante dos años, durante los cuales recogió datos de estaciones de telemetría. Después de apagar el servidor por un problema de suministro eléctrico, los datos se guardaron en hojas de Excel. El SNIH no pudo recibir datos de los ficheros Excel desde el momento en que se apagó el servidor; sin embargo, estos ficheros siguen generándose.</p> <p>En la base de datos SQLite está lo que había en la base de datos dBase, más los dos años de datos de telemetría que se recogieron mientras el servidor estaba funcionando.</p> <p>En última instancia, hay dos fuentes: SQLITE y archivos Excel.</p>	<p>Hidrometría</p> <p>Horas de sol</p> <p>Precipitación</p> <p>Presión</p> <p>Temperatura</p> <p>Humedad</p> <p>Viento</p>	<p>Climatología</p> <p>Meteorología agrícola</p> <p>Sinópticos de superficie</p>
Interior IT Ayuda	Migración de bases de datos dBase.	<p>Hidrometría</p> <p>Horas de sol</p> <p>Precipitación</p> <p>Presión</p> <p>Temperatura</p> <p>Humedad</p> <p>Viento</p>	<p>Climatología</p> <p>Meteorología agrícola</p> <p>Sinópticos de superficie</p>

Como puede verse, los métodos de recogida de datos son muy variados, y los más importantes parecen ser:

- Manual, mediante archivo de texto (por ejemplo, archivo Excel);
- Por telemetría;
- Por radio;
- A través de la red telefónica;
- A través de la API;

- Mediante programas informáticos, tanto de código abierto como privados.

Algunos de los entrevistados destacan el mal funcionamiento de la transmisión automática de datos, debido principalmente a averías y a una falta estructural de fondos para el mantenimiento, tanto rutinario como extraordinario. Las fuentes son extremadamente heterogéneas. Por este motivo, **la plataforma tendrá que ofrecer una o varias funcionalidades para la introducción de datos, que permitan tanto la introducción automática como la manual.**

La entrevista también es muy útil para hacerse una idea general de todos los tipos de estaciones/plataformas y datos que el sistema tendrá que gestionar y procesar. Desde el punto de vista de los requisitos funcionales, la necesidad se impone:

- Poder almacenar todo tipo de datos;
- Poder declarar todas las estaciones climáticas de que dispone la organización productora, pudiendo definir el tipo y la fuente de los datos;
- Poder gestionar y filtrar los datos en función de su tipo.

Además, la encuesta permitió esbozar un nuevo papel además del del usuario final: el de la institución. La nueva función es necesaria para poder subdividir la gestión de datos en función de la entidad a la que pertenezcan las estaciones, las encuestas y los datos archivados.

### 3.1.2 EVALUACIÓN COMPARATIVA DE LAS PLATAFORMAS EXISTENTES

La creación de esta tabla comparativa – resultado del análisis realizado en el **Capítulo 2** – pretende poner de relieve las diferencias entre las plataformas existentes para tener una visión general de las funcionalidades mínimas que deben realizarse según lo exigido por los entregables del proyecto.

**Cuadro 3. Evaluación comparativa de las plataformas existentes**

	COLOMBIA	CHILE	MÉXICO	BELICE
Mapa interactivo	●	●	●	●
Filtros por estación	●	●	● <sup>[14]</sup>	●
Filtros por variable	●	●	●	●

<sup>14</sup> Filtro parcial, sólo distingue entre estaciones activas y pasivas

	COLOMBIA	CHILE	MÉXICO	BELICE
Filtros de tiempo	●	●	●	●
Descargar/Exportar datos	●	●	●	●
Gráficos de datos por estación	●	●	● <sup>[15]</sup>	●
Gráficos de datos por variable	●	●	● <sup>[16]</sup>	●
Visualización de datos estadísticos	●	●	●	●
Visualización de datos estadísticos en un mapa	●	●	●	●
Análisis espacial (mapa de calor)	●	●	● <sup>[17]</sup>	●
Lista de estaciones	● <sup>[18]</sup>	●	●	●
Comparación de estaciones	●	●	●	●

Observando las distintas plataformas, se puede ver que todas presentan un mapa interactivo como característica común, pero hay algunas diferencias entre ellas. Por ejemplo, para Colombia, Chile y Belice, los filtros son completos y permiten visualizar los datos directamente en el mapa, mientras que para la plataforma mexicana sólo es posible ver todas las estaciones en el mapa, y el único filtro disponible sirve para distinguir entre estaciones activas e inactivas. Esta limitación hace que la plataforma mexicana sea menos adecuada como base para la creación del mapa interactivo.

Además del mapa interactivo, las plataformas también tienen una función común para descargar datos climáticos, con algunas diferencias en los filtros de exportación. Sin embargo, hay grandes diferencias en la visualización de datos estadísticos, que pueden mostrarse mediante una lista, un mapa o gráficos.

Cabe destacar que la plataforma mexicana no procesa los datos de forma dinámica, lo que la hace menos personalizable y versátil para las necesidades de los usuarios. En cambio, las demás plataformas ofrecen un mapa dinámico, que permite visualizar los datos estadísticos directamente en el mapa, en la ubicación de las estaciones de las que se tomaron.

<sup>15</sup> El gráfico no es dinámico, sino el resultado del procesamiento por software de terceros

<sup>16</sup> El gráfico no es dinámico, sino el resultado del procesamiento por software de terceros

<sup>17</sup> El análisis espacial no es dinámico, sino el resultado de un tratamiento realizado en otro lugar

<sup>18</sup> Lista de estaciones no visibles en la plataforma, pero descargables desde el sitio web

La posibilidad de analizar los datos a través de gráficos (histogramas) está menos presente, y sólo se encuentra en la plataforma chilena a través de la selección de una estación, y en la plataforma de Belice con filtros más detallados que incluyen tanto estaciones como variables climáticas. Además, salvo en el caso de la plataforma chilena, sólo es posible visualizar el listado de estaciones (en el caso colombiano sólo descargando un archivo .csv).

Por último, Belice dispone de una función única que permite la comparación entre estaciones, lo que permite al usuario analizar las diferencias entre los datos recogidos en distintas estaciones, incluso mediante un gráfico específico.

## 3.2 REQUISITOS PROPUESTOS

Esta sección describirá con más detalle las funcionalidades específicas que debe tener el sistema para cumplir los requisitos identificados anteriormente. Estas características fueron sugeridas por los análisis realizados anteriormente y se enumeran para que puedan clasificarse fácilmente y reutilizarse en el futuro.

En comparación con los análisis anteriores, en los que sólo se hacía referencia al usuario final, ahora se añaden dos nuevos papeles: el de la entidad productora y el de la Administración. En el caso de la entidad productora, tal y como se desprende de la fase de encuesta descrita en el apartado **3.1.1**, existe la necesidad de que las entidades puedan gestionar sus propias estaciones/plataformas de forma independiente, pero sin disponer al mismo tiempo de las autorizaciones necesarias para gestionar las estaciones/plataformas de otras entidades. Por otro lado, en lo que respecta a la Administración (o, más en general, al gestor/administrador), los requisitos están relacionados con la posibilidad de poder gestionar tanto entidades como emisoras/plataformas, independientemente de la entidad a la que pertenezcan.

En resumen, **el sistema debe garantizar funcionalidades específicas para cada rol implicado, con el fin de responder a las necesidades de gestión de la estación por parte de las entidades y de control cruzado y seguimiento por parte del gobierno y/o entidad administradora.** Las funcionalidades se enumeran claramente y se clasifican por categorías para facilitar su consulta y reutilización en el futuro.

**Cuadro 4. Requisitos funcionales (llamados con la abreviatura RF-XX) sugeridos**

Nombre	Descripción
RF-01	El sistema debe presentar un mapa del terreno nacional con marcadores que indiquen la ubicación de las estaciones climáticas instaladas.
RF-02	El sistema debe permitir al usuario interactuar con el mapa mediante filtros adecuados.

Nombre	Descripción
RF-03	El sistema debe permitir al usuario visualizar datos estadísticos de interés directamente en el mapa.
RF-04	El sistema debe presentar una lista de estaciones climáticas.
RF-05	El sistema debe permitir al usuario examinar los datos de cada estación.
RF-06	El sistema debe permitir al usuario poder analizar los datos y/o los tratamientos estadísticos relativos a una estación.
RF-07	El sistema debe permitir al usuario poder analizar datos y/o tratamientos estadísticos relacionados con una variable.
RF-08	El sistema debe ser capaz de realizar procesamientos gráficos y/o estadísticos basados en datos.
RF-09	El sistema debe permitir al usuario comparar diferentes estaciones y/o diferentes variables.
RF-10	El sistema debe proporcionar un mapa dinámico que permita al usuario crear mapas de calor según sus necesidades estableciendo y aplicando los filtros adecuados.
RF-11	El sistema debe permitir que el usuario pueda descargar/guardar datos climáticos en su dispositivo.
RF-12	El sistema debe permitir que las entidades productoras de datos hidrometeorológicos puedan introducir nuevas estaciones/plataformas de observación.
RF-13	El sistema debe permitir que las entidades productoras de datos hidrometeorológicos puedan modificar sus estaciones/plataformas de observación.
RF-14	El sistema debe permitir que las entidades productoras de datos hidrometeorológicos puedan eliminar sus propias estaciones/plataformas de observación.
RF-15	El sistema debe ofrecer a las entidades productoras de datos hidrometeorológicos la posibilidad de definir y aplicar umbrales de validación para el control de calidad de los datos hidrometeorológicos correspondientes a sus estaciones/plataformas de observación.
RF-16	El sistema debe permitir a las entidades productoras de datos hidrometeorológicos importar datos hidrometeorológicos, ya sean antiguos o nuevos.
RF-17	El sistema debe permitir al gobierno/gestor poder incluir nuevas entidades productoras de datos hidrometeorológicos y/o nuevas estaciones/plataformas de observación.
RF-18	El sistema debe permitir al gobierno/gestor poder cambiar las entidades productoras de datos hidrometeorológicos y/o las estaciones/plataformas de observación.
RF-19	El sistema debe permitir al gobierno/gestor poder eliminar las entidades productoras de datos hidrometeorológicos y/o estaciones/plataformas de observación.

### 3.2.1 DIAGRAMAS DE CASOS DE USO

Los diagramas de casos de uso son herramientas útiles para definir la funcionalidad de un sistema. Sin embargo, para utilizarlos correctamente es necesario conocer el lenguaje UML (*Unified Modeling Language*), diseñado para crear un lenguaje común de modelado visual rico tanto en semántica como en sintaxis. El UML se creó basándose en el paradigma orientado a objetos, que permite modelar objetos del mundo real.



El paradigma orientado a objetos es una metodología de diseño que permite crear software basado en objetos que contienen datos (atributos) y métodos (comportamiento). Un objeto es una instancia de una clase que define el modelo utilizado para crear objetos.

El UML es una combinación de varias notaciones orientadas a objetos, como el diseño orientado a objetos, la técnica de modelado de objetos y la ingeniería de software orientada a objetos. Se utiliza en ingeniería de software para describir los límites, la estructura y el comportamiento del sistema y los objetos que lo componen.

El UML tiene la ventaja de utilizar modelos fáciles de entender para el cliente y para el equipo de desarrollo de software que está familiarizado con la sintaxis. En el desarrollo de sistemas, se utilizan tres modelos de sistema diferentes que se centran en aspectos distintos:

1. Modelo funcional: los diagramas de casos de uso se utilizan para describir la funcionalidad del sistema desde el punto de vista del usuario.
2. Modelo de objetos: los diagramas de clases se utilizan para describir la estructura del sistema en términos de objetos, atributos, asociaciones y operaciones.
3. Modelo dinámico: para describir el comportamiento interno del sistema se utilizan diagramas de interacción (diagramas de secuencia), diagramas de máquinas de estados (diagramas de Statechart) y diagramas de actividades (diagramas de actividad).

En esta sección, nos centraremos en modelar el sistema desde un punto de vista funcional, utilizando diagramas de casos de uso.

Estos diagramas describen las funcionalidades o servicios que ofrece el sistema, tal y como los perciben y utilizan los actores que interactúan con él. Además, pueden utilizarse como herramienta para representar los requisitos funcionales del sistema.

En los diagramas de casos de uso UML se utilizan tres elementos principales:

1. Actores: representados gráficamente por un icono que representa a un hombre estilizado (stickman), los actores son formalmente clases con el estereotipo "actor" y representan un papel cubierto por un conjunto de entidades que interactúan con el sistema (incluidos usuarios humanos, otros sistemas de software, dispositivos de hardware, etc.). Cada rol corresponde a un conjunto de interacciones relacionadas que el actor realiza con el sistema.
2. Casos de uso: representados gráficamente como elipses que contienen el nombre del caso de uso, estos casos de uso pueden considerarse clases de comportamientos relacionados. En la práctica, representan una función o un servicio ofrecido por el sistema a uno o varios actores. Esta función debe ser exhaustiva y significativa desde el punto de vista de los actores implicados.

3. Relaciones: representan las conexiones entre los elementos del modelo y añaden semántica al modelo, definiendo su estructura y el comportamiento entre elementos. Existen cuatro tipos de relaciones en los Diagramas de Casos de Uso:
  - i. Asociación: relación fundamental que conecta a los actores con los casos de uso. Los actores pueden asociarse a cualquier número de casos de uso y viceversa.
  - ii. Generalización: las relaciones de generalización pueden expresarse tanto entre actores como entre casos de uso. Esta relación se basa en el concepto de generalización entre clases e indica que un elemento hijo hereda los atributos, operaciones y relaciones definidos por un elemento padre.
  - iii. Inclusión: representada por una línea de puntos con el estereotipo "incluye", esta relación entre casos de uso indica que la función representada por un caso de uso incluye completamente la función representada por el otro.
  - iv. Extensión: representada por una línea de puntos con el estereotipo "extender", esta relación entre casos de uso indica que la función representada por el caso de uso "extender" puede utilizarse en el contexto de la función "extendida", es decir, representa una especie de enriquecimiento de ésta.

También en el caso de este proyecto, la realización de los diagramas de casos de uso tuvo lugar tras un análisis de las funcionalidades necesarias a partir de la definición de los requisitos funcionales, con el objetivo de cumplir dichos requisitos<sup>[19]</sup>.

---

<sup>19</sup> En los diagramas UML propuestos, las funcionalidades coloreadas en azul representan las que serán objeto de un posible desarrollo a medida.

### 3.2.1.1 INTERACCIÓN CON EL MAPA

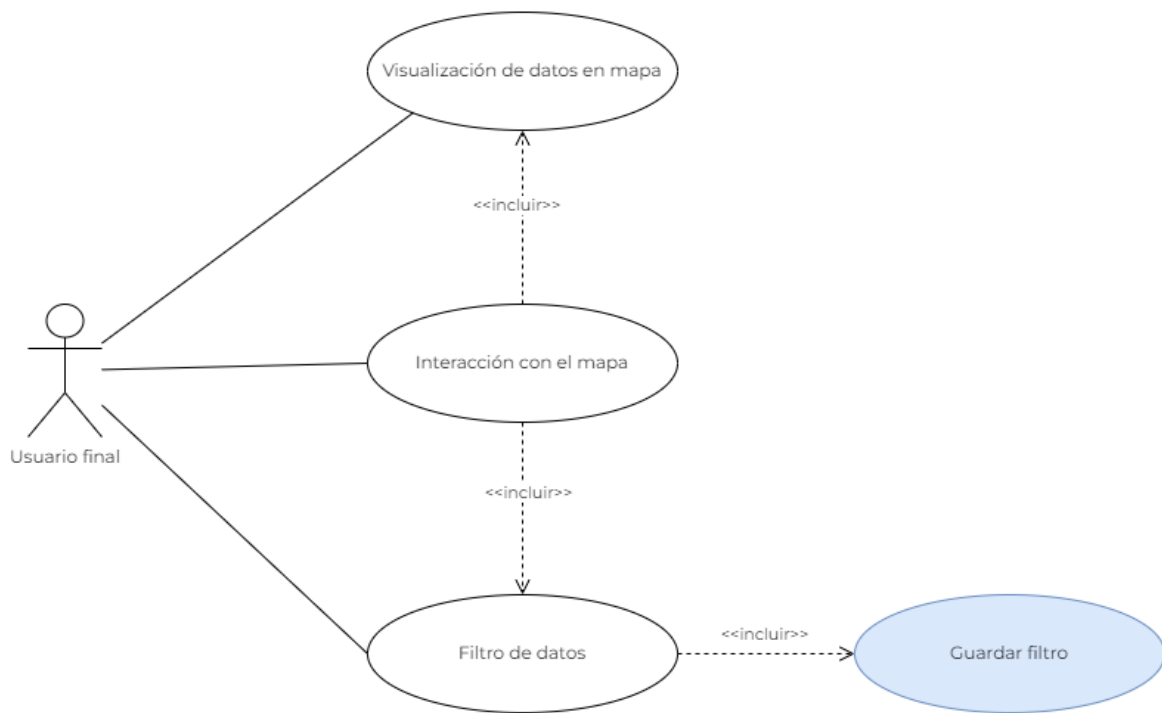


Figura 3. UC-01 - Visualización e interacción con el mapa

La primera función introducida es un mapa nacional interactivo que permite a los usuarios visualizar la ubicación de las estaciones meteorológicas mediante marcadores. Los usuarios pueden interactuar con el mapa para visualizar los datos climáticos directamente en la zona de interés donde se tomaron.

Además, el mapa interactivo permite filtrar los datos, por ejemplo, para reducir las zonas de interés o seleccionar las variables que se mostrarán, como la temperatura. Esta función de filtrado permite a los usuarios centrarse en los datos climáticos concretos que les interesan, sin tener que navegar por información superflua.

Cuadro 5. UC-01 - Visualización e interacción con el mapa

UC-01	Visualización e interacción con el mapa	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un usuario desea interactuar con los datos hidrometeorológicos del mapa.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Usuario final</li> </ul>	
	<b>Paso</b>	<b>Acción</b>

UC-01	Visualización e interacción con el mapa	
<b>Secuencia ordinaria</b>	<b>1</b>	El sistema muestra un <i>mapa interactivo</i> donde se pueden encontrar todas las estaciones.
	<b>2</b>	El actor <b>usuario final</b> <i>visualiza los datos hidrometeorológicos de las estaciones</i> haciendo clic sobre una de ellas en el mapa, representada por un marcador visual.
	<b>3</b>	El actor <b>usuario final</b> aplica determinados <i>filtros</i> .
	<b>4</b>	El sistema muestra las estaciones relacionadas con los filtros seleccionados.
	<b>5</b>	El actor <b>usuario final</b> guarda los <i>filtros</i> utilizados.
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	<b>3</b>	Si el <b>usuario final</b> está autenticado y ya ha utilizado el sistema, puede elegir <i>filtros guardados</i> previamente.
	<b>5</b>	Sólo puede utilizarse si el <b>usuario final</b> está autenticado.
<b>Requisitos funcionales</b>	RF-01, RF-02, RF-03	

### 3.2.1.2 DATOS DE LA ESTACIÓN

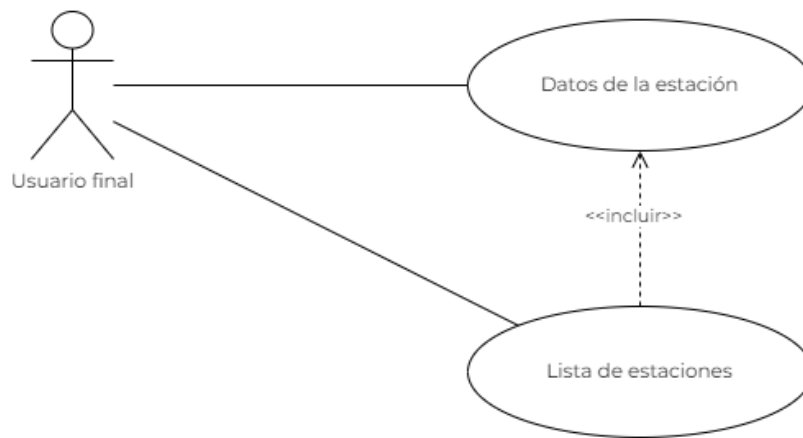


Figura 4. UC-02 - Datos de la estación

El objetivo de esta funcionalidad es proporcionar una lista exhaustiva de todas las estaciones hidrológicas y meteorológicas instaladas en todo el país. A continuación, los usuarios pueden acceder a una vista detallada de los datos de cada estación desde la lista proporcionada. Esto permite a los usuarios explorar los datos hidrológicos y meteorológicos de cada estación de forma rápida y eficaz.

Cuadro 6. UC-02 - Datos de la estación

UC-02	Datos de la estación	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un usuario desea visualizar los datos de una estación.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Usuario final</li> </ul>	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>usuario final</b> quiere ver todas las estaciones
	2	El sistema muestra una tabla con la <i>lista de estaciones</i> .
	3	El actor <b>usuario final</b> selecciona una estación de la lista.
4	El sistema muestra los <i>datos de la estación</i> .	
<b>Requisitos funcionales</b>	RF-04, RF-05	

### 3.2.1.3 ANÁLISIS DE LOS DATOS DE LAS ESTACIONES

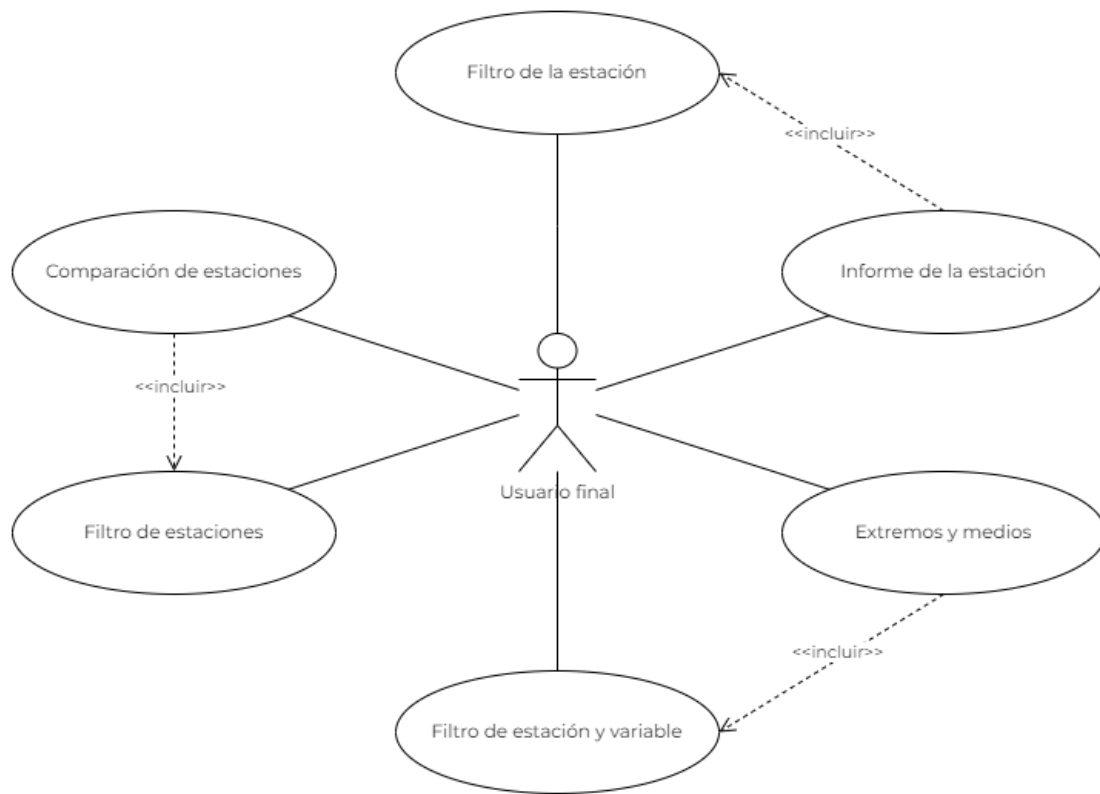


Figura 5. UC-03 - Análisis de los datos de las estaciones

La funcionalidad de *Análisis de Datos* que aquí se presenta tiene como objetivo proporcionar al usuario datos hidrológicos y meteorológicos procesados de forma sencilla y directa. Una de las principales funciones de esta sección es el *Informe de estación*, que permite a los usuarios ver los datos procesados en función de la estación seleccionada. Estos datos se presentan tanto en forma de lista como de histograma, lo que facilita el análisis y la comprensión de la información.

Otra función disponible es la *Comparación de estaciones*, que permite a los usuarios comparar estaciones en función de variables seleccionadas mediante filtros. De este modo, los usuarios pueden comparar diferentes estaciones hidrológicas y meteorológicas para comprender las diferencias en los datos.

Por último, la función *Extremos y Medias* permite a los usuarios visualizar las medias y los extremos (mín./máx.) de una variable específica en función del mes y la estación seleccionados. Esta función permite a los usuarios comprender patrones y tendencias de forma fácil y eficaz.

**Cuadro 7. UC-03 - Análisis de los datos de las estaciones**

UC-03	Análisis de los datos de las estaciones	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un usuario desea visualizar estadísticas sobre los datos de una estación hidrometeorológica.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Usuario final</li> </ul>	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>usuario final</b> solicita el <i>informe de una estación</i> , aplicando los <i>filtros</i> deseados.
	2	El sistema ofrece una interfaz en la que se pueden seleccionar los datos analizados en función de la <i>estación</i> y las <i>variables filtradas</i> (por ejemplo, la precipitación media diaria/mensual/anual de la estación); además, el sistema mostrará un gráfico de los valores de interés.
	3	El actor <b>usuario final</b> elige dos estaciones para <i>compararlas</i> utilizando el <i>filtro de estaciones</i> .
	4	El sistema proporciona una interfaz que muestra un <i>gráfico</i> basado en las estaciones seleccionadas y una lista de datos climáticos utilizados para la comparación.
	5	El actor <b>usuario final</b> desea visualizar estadísticas como <i>medias y extremos</i> mediante un <i>filtro</i> en el que puede elegir el mes, la estación y las variables en las que está interesado el usuario.
	6	El sistema muestra una lista de los datos estadísticos requeridos.
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	1	Si el <b>usuario final</b> está autenticado y ya ha utilizado el sistema, puede elegir <i>filtros guardados</i> previamente.
<b>Requisitos funcionales</b>	RF-06, RF-08, RF-09	

### 3.2.1.4 ANÁLISIS DE DATOS DE VARIABLES

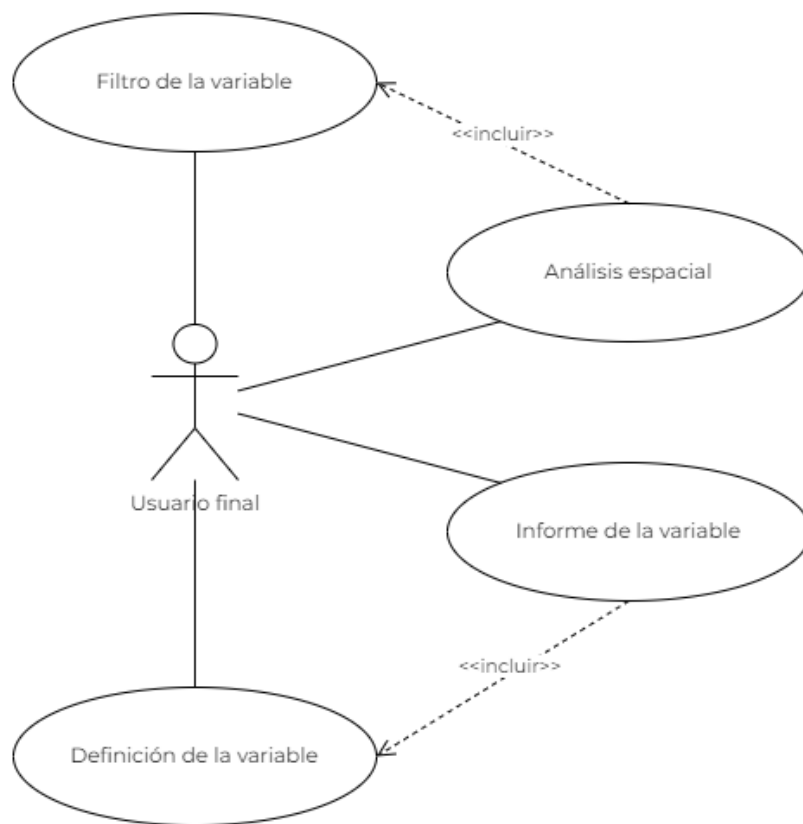


Figura 6. UC-04 - Análisis de datos de variables

En este caso, el análisis de los datos se refiere a las variables hidrológicas y meteorológicas del sistema. La primera funcionalidad se denomina "Informe de variables" y, al igual que "Informe de estaciones", procesa los datos en función de la variable seleccionada. Los resultados se muestran en forma de lista o de histograma.

La funcionalidad "Análisis espacial", por su parte, permite visualizar directamente los datos procesados en un mapa coloreado en función de la variable seleccionada y de la escala de referencia fijada para la coloración.

Cuadro 8. UC-04 - Análisis de datos de variables

UC-04	Análisis de datos variables	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un usuario desea visualizar estadísticas sobre datos hidrometeorológicos en función de las variables.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Usuario final</li> </ul>	
	<b>Paso</b>	<b>Acción</b>



UC-04	Análisis de datos variables	
<b>Secuencia ordinaria</b>	1	El actor <b>usuario final</b> elige la <i>variable</i> de interés para el <i>informe</i> .
	2	El sistema proporciona una interfaz que muestra un gráfico en función de la variable elegida.
	3	El actor <b>usuario final</b> desea visualizar el <i>análisis</i> espacial en función del <i>filtro de la variable</i> .
	4	El sistema muestra el mapa coloreado según la variable seleccionada y la escala establecida en los filtros.
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	1	Si el <b>usuario final</b> está autenticado y ya ha utilizado el sistema, puede elegir <i>filtros</i> guardados anteriormente.
	3	Si el <b>usuario final</b> está autenticado y ya ha utilizado el sistema, puede elegir <i>filtros</i> guardados previamente.
<b>Requisitos funcionales</b>	RF-07, RF-08, RF-10	

### 3.2.1.5 EXPORTACIÓN DE DATOS

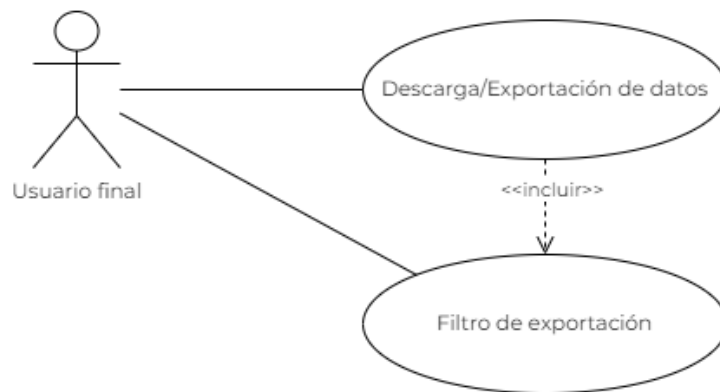


Figura 7. UC-05 - Exportación de datos

Además de la posibilidad de visualizar los datos, también queremos ofrecer a los usuarios la funcionalidad de exportarlos y descargarlos utilizando un filtro especial para seleccionar la estación, las variables y el periodo de tiempo de interés.

Cuadro 9. UC-05 - Exportación de datos

UC-05	Exportación de datos	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un usuario desea exportar datos hidrometeorológicos.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Usuario final</li> </ul>	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>usuario final</b> desea exportar datos hidrometeorológicos.
	2	El sistema muestra un <i>filtro</i> para elegir las estaciones y variables de interés, la fecha de inicio y fin de los datos a exportar (en función de cuándo se recogieron)
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	2	Si el <b>usuario final</b> está autenticado y ya ha utilizado el sistema, puede elegir filtros guardados anteriormente.
<b>Requisitos funcionales</b>	RF-11	

### 3.2.1.6 GESTIÓN DE ACCESOS DIRECTOS Y FILTROS GUARDADOS

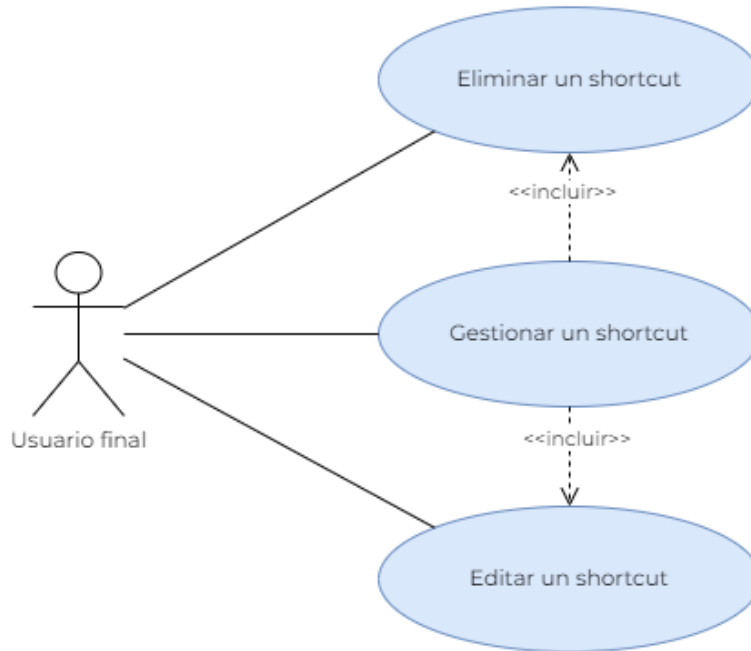


Figura 8. UC-06 - Gestión de accesos directos y filtros guardados

Queremos ofrecer a los usuarios la posibilidad de guardar los filtros utilizados en las funciones descritas anteriormente, para que puedan reutilizarlos en futuras visitas a la plataforma. Para ello, será posible gestionar todos los filtros guardados, permitiendo a los usuarios verlos, editarlos o eliminarlos.

Cuadro 10. UC-06 - Gestión de accesos directos y filtros guardados

UC-06	Gestión de <i>accesos directos</i> y filtros guardados	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un usuario desea gestionar accesos directos previamente guardados.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Usuario final</li> </ul>	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>usuario final</b> desea <i>gestionar los accesos directos</i> (filtros previamente guardados).
	2	El sistema muestra la lista de accesos directos guardados.
	3	<i>El actor usuario final puede modificar los accesos directos.</i>
	4	El actor <b>usuario final</b> puede <i>eliminar accesos directos</i> .

UC-06	Gestión de <i>accesos directos</i> y filtros guardados	
Excepciones	Paso	Acción
	1	Si el <b>usuario final</b> no está autenticado, no puede <i>gestionar los accesos directos</i> .

### 3.2.1.7 GESTIÓN DE LAS ESTACIONES SEGÚN LA ENTIDAD PRODUCTORA

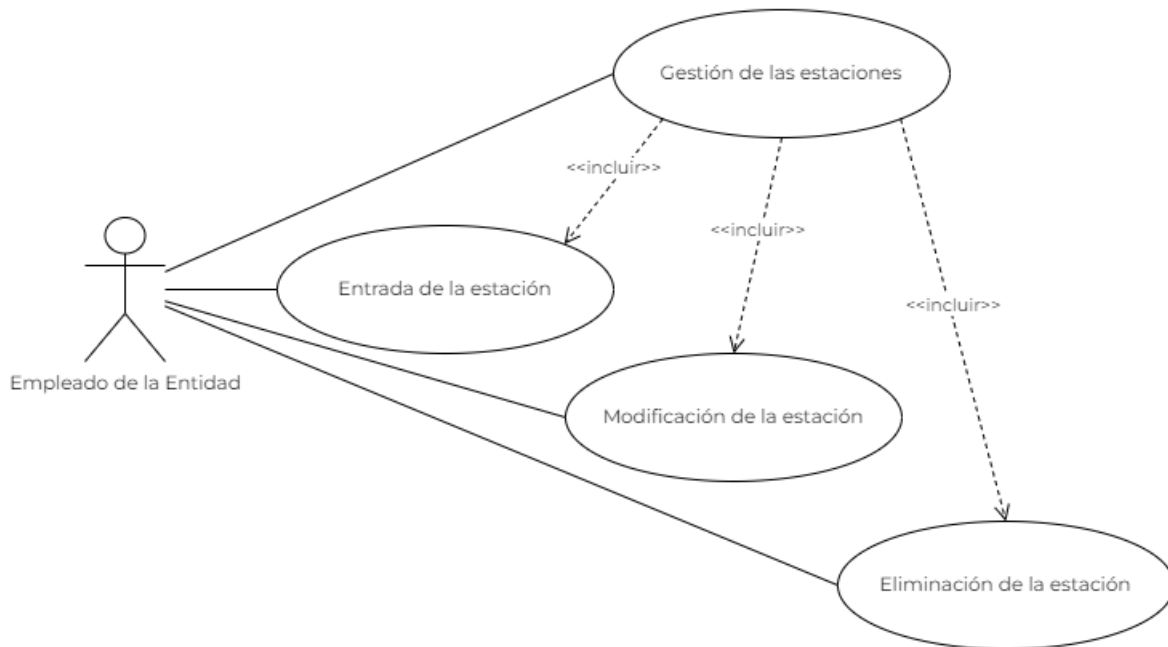


Figura 9. UC-07 - Gestión de estaciones

Esta funcionalidad está diseñada para permitir a las entidades productoras de datos hidrometeorológicos gestionar sus propias estaciones hidrológicas y/o meteorológicas. Esto incluye la posibilidad de:

- Añade una nueva estación;
- Modificación de una estación existente;
- Borrar una estación.

Sin embargo, estas acciones sólo pueden realizarse si las estaciones pertenecen a la organización o son gestionadas por ella, y si el usuario tiene permisos para añadir, editar o eliminar estaciones. La gestión de autorizaciones se describirá más adelante<sup>[20]</sup>.

Cuadro 11. UC-07 - Gestión de estaciones

UC-07	Gestión de estaciones
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un empleado de una entidad productora de datos hidrometeorológicos desea introducir, modificar o eliminar estaciones.

<sup>20</sup> Apartados 5.2.2 y 5.6.2.1

UC-07	Gestión de estaciones	
<b>Actores</b>	<ul style="list-style-type: none"> <li>Empleado de la entidad</li> </ul>	
<b>Condición previa</b>	El actor <b>empleado de la entidad</b> ya se ha autenticado y está habilitado en el nivel de permisos "entidad productora".	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>empleado de la entidad</b> quiere <i>entrar en una nueva estación</i> .
	2	El sistema muestra un formulario para <i>introducir</i> los datos de la <i>nueva estación</i> .
	3	El actor <b>empleado de la entidad</b> desea <i>modificar los datos de la estación</i> .
	4	El sistema muestra un formulario para <i>editar los datos de la estación</i> .
	5	El actor <b>empleado de la entidad</b> quiere <i>eliminar una estación</i> .
	6	El sistema <i>borra la estación</i> seleccionada.
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	2	Si el <b>empleado de la entidad</b> no tiene permiso para <i>introducir un nuevo puesto</i> , el sistema no lo permite.
	4	Si el <b>empleado de la entidad</b> no tiene permiso para <i>cambiar una estación</i> , el sistema no lo permite.
	6	Si el <b>empleado de la entidad</b> no tiene permisos para <i>eliminar una estación</i> , el sistema no lo permite.
<b>Requisitos funcionales</b>	RF-12, RF-13, RF-14	

### 3.1.2.8 CONTROL DE CALIDAD Y VALIDACIÓN DE DATOS

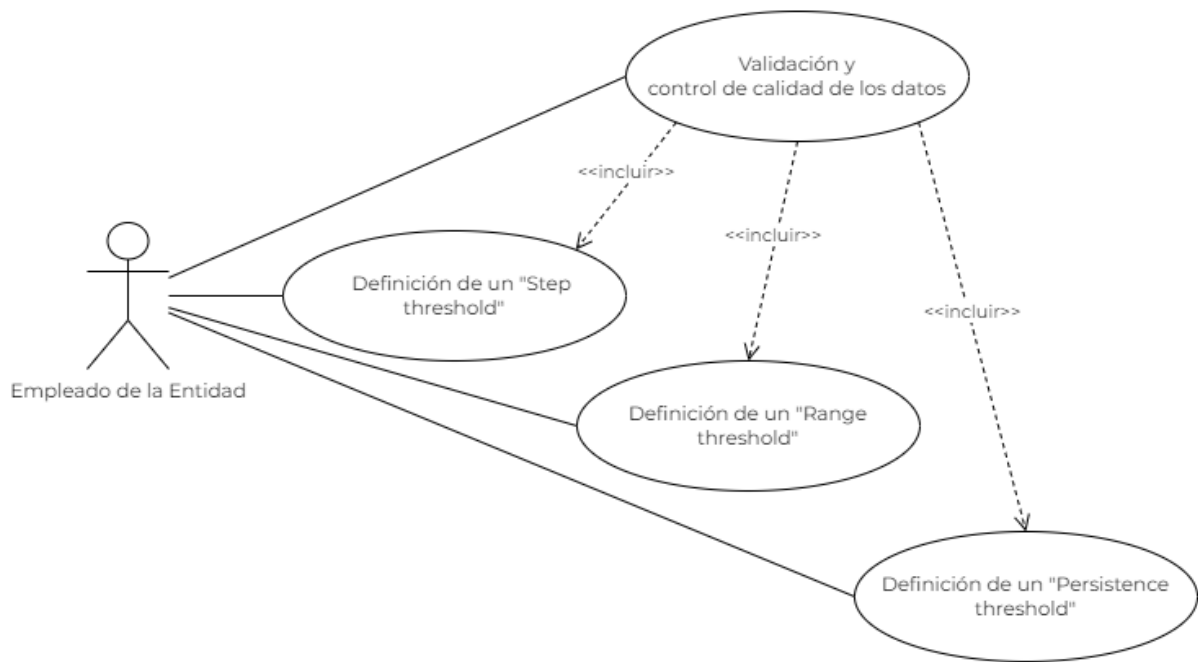


Figura 10. UC-08 - Control de calidad y validación de datos

Para garantizar un control de calidad adecuado, se desea ofrecer la posibilidad de definir umbrales para la validación de los datos con el fin de automatizar estas comprobaciones. Estos umbrales pueden definirse de forma global para todas las variables, o de forma más específica para las variables de una estación concreta. Los umbrales y su funcionamiento en la práctica se describen con más detalle en el apartado 5.3.2.

Cuadro 12. UC-08 - Control de calidad y validación de datos

UC-08	Control de calidad y validación de datos	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un empleado de una entidad productora de datos hidrometeorológicos desea definir umbrales para la validación de datos hidrometeorológicos y para el control de calidad.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>Empleado de la entidad</li> </ul>	
<b>Condiciones previas</b>	El actor <b>empleado de la entidad</b> ya se ha autenticado y está habilitado en el nivel de permisos "entidad productora".	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>empleado de la entidad</b> desea <i>validar los datos</i> para evitar valores atípicos.

UC-08	Control de calidad y validación de datos	
	2	El sistema muestra los tres tipos de <i>umbrales</i> que pueden definirse para que puedan seleccionarse.
	3	El actor <b>empleado de la entidad</b> desea <i>definir un</i> umbral de paso.
	4	El sistema muestra un formulario para <i>definir un "Umbral de paso "</i> en función de la estación y de las variables seleccionadas.
	5	El actor <b>empleado de la entidad</b> desea <i>definir un</i> umbral de rango.
	6	El sistema muestra un formulario para <i>definir un 'Umbral de Rango '</i> basado en la estación y las variables seleccionadas.
	7	El actor <b>empleado de la entidad</b> desea <i>definir un</i> umbral de persistencia.
	8	El sistema muestra un formulario para <i>definir un "Umbral de persistencia "</i> en función de la estación y las variables seleccionadas.
	Excepciones	Paso
2		Si el <b>empleado de la entidad</b> no tiene permiso para <i>establecer nuevos umbrales</i> , el sistema no lo permite.
Requisitos funcionales	RF-15	

### 3.1.2.9 IMPORTACIÓN DE DATOS HYDROMETEOCLIMATICOS

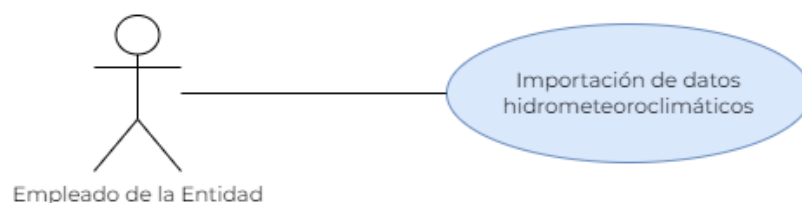


Figura 11. UC-09 - Importación de datos hidrometeorológicos

Se desea ofrecer a cada entidad productora de datos hidrometeorológicos la posibilidad de importar datos hidrometeorológicos, tanto nuevos como históricos, para permitir una evaluación de los datos a lo largo del tiempo mediante su análisis. Sin embargo, la importación de datos está sujeta a restricciones de permisos de usuario.

Cuadro 13. UC-09 - Importación de datos hidrometeorológicos



UC-09	Importación de datos hidrometeorológicos	
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un empleado de una entidad productora de datos hidrometeorológicos desea importar datos hidrometeorológicos, tanto nuevos como históricos.	
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Empleado de la entidad</li> </ul>	
<b>Condiciones previas</b>	El actor <b>empleado de la entidad</b> ya se ha autenticado y está habilitado en el nivel de permisos "entidad productora".	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>empleado de la entidad</b> desea <i>importar datos hidrometeorológicos</i> .
	2	El sistema muestra una interfaz para <i>importar datos</i> .
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	2	Si el <b>empleado de la entidad</b> no tiene permisos para <i>importar datos</i> , el sistema no lo permite.
<b>Requisitos funcionales</b>	RF-16	

### 3.2.1.10 GESTIÓN DE LAS ENTIDADES PRODUCTORAS

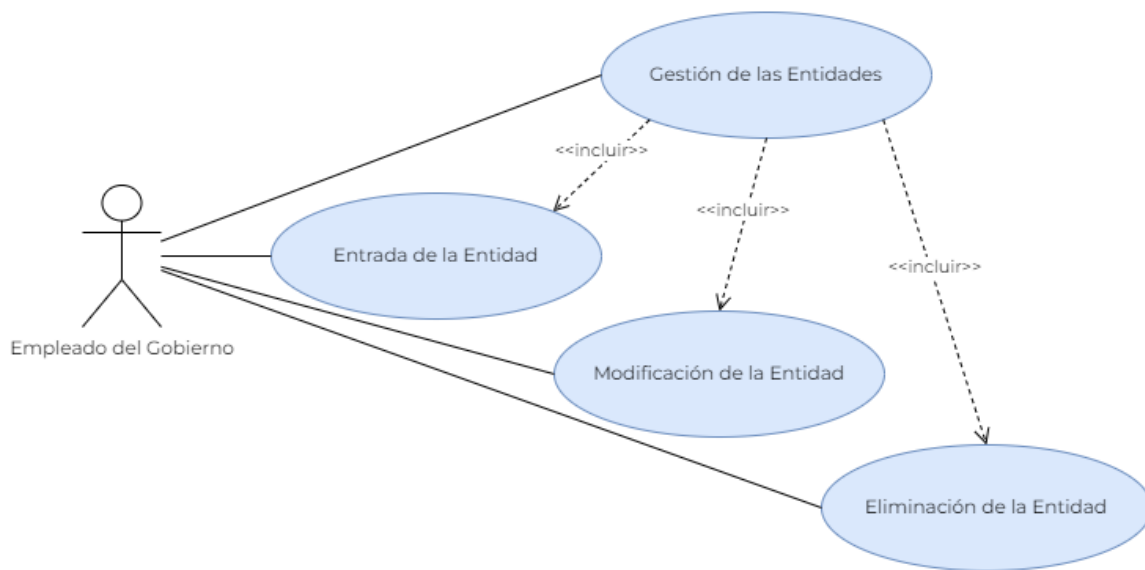


Figura 12. UC-10 - Gestión de entidades productoras de datos hidrometeorológicos

El objetivo de esta funcionalidad es proporcionar al gobierno (o a cualquier entidad que actúe como *superadministrador*) la capacidad de gestionar las entidades productoras, a saber:

- Introduzca una nueva entidad;
- Modificar una entidad existente;
- Eliminar una entidad.

Sin embargo, es importante tener en cuenta que estas acciones sólo pueden realizarse si el usuario dispone de los permisos necesarios para llevarlas a cabo. Más adelante se ofrecerá una descripción detallada de la gestión de autorizaciones<sup>[21]</sup>.

Cuadro 14. UC-10 - Gestión de organizaciones de productores

UC-10	Gestión de las entidades productoras
<b>Descripción</b>	El sistema debe comportarse como se describe en el siguiente caso de uso cuando un empleado de la administración (o superadministrador) desea insertar, modificar o eliminar entidades productoras.
<b>Actores</b>	<ul style="list-style-type: none"> <li>• Empleado de la administración</li> </ul>
<b>Condiciones previas</b>	El actor <b>empleado de la administración</b> ya se ha autenticado y está habilitado con el nivel de permiso 'Gobierno' o 'Superadmin'.

<sup>21</sup> Apartados 5. 2.2 y 5.6.2.1

UC-10	Gestión de las entidades productoras	
<b>Secuencia ordinaria</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <b>empleado de la administración</b> quiere <i>insertar una nueva entidad</i> .
	2	El sistema muestra un formulario para <i>introducir los datos de la nueva entidad</i> .
	3	El actor <b>empleado de la administración</b> quiere <i>modificar los datos de una entidad</i> .
	4	El sistema muestra un formulario para <i>modificar los datos de una entidad</i> .
	5	El actor <b>empleado de la administración</b> quiere <i>eliminar una entidad</i> .
	6	El sistema <i>borra la entidad</i> seleccionada.
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	2	Si el actor <b>empleado de la administración</b> no tiene permiso para <i>introducir una nueva entidad</i> , el sistema no se lo permite.
	4	Si el actor <b>empleado de la administración</b> no tiene permiso para <i>modificar una entidad</i> , el sistema no se lo permite.
6	Si el actor <b>empleado de la administración</b> no tiene permiso para <i>eliminar una entidad</i> , el sistema no se lo permite.	
<b>Requisitos funcionales</b>	RF-17. RF-18, RF-19	

## **4 REQUISITOS NO FUNCIONALES**

Como ya se ha explicado con más detalle en la introducción, los requisitos no funcionales son cruciales para garantizar la calidad del sistema y la satisfacción de usuarios y partes interesadas. Representan aspectos críticos del producto que no pueden pasarse por alto durante el diseño y la aplicación. Los requisitos no funcionales pueden dividirse en varias categorías, como rendimiento, seguridad, usabilidad, escalabilidad, mantenibilidad y compatibilidad. La gestión de estos requisitos requiere el uso de técnicas específicas, como la modelización y la verificación, para garantizar la consecución de los objetivos de calidad fijados. Además, la consideración de los requisitos no funcionales debe mantenerse durante todo el ciclo de vida del producto, desde la fase de diseño hasta las pruebas y el mantenimiento (Gacek & Arief, 2019; Lyytinen & Kautz, 2019).

### **4.1 ANÁLISIS COMPARATIVO**

El objetivo de esta sección del documento es ofrecer al lector una visión en profundidad de las tres plataformas CDMS elegidas como puntos de referencia. La razón de la elección de estas tres plataformas (MCH, SURFACE y Climsoft) puede resumirse de la siguiente manera: MCH ya está adoptada en República Dominicana por ONAMET, mientras que SURFACE y Climsoft son las dos alternativas de código abierto con las características más adecuadas para ser traídas como términos de comparación en un análisis comparativo. El análisis comparativo se realizará dividiéndolo en varias áreas temáticas, en función de lo que se consideren las características más importantes a comparar a la hora de elegir una plataforma de gestión de datos climáticos.

Cuando se utiliza correctamente, la evaluación comparativa es una herramienta muy importante, porque permite valorar las diferencias y similitudes entre productos, servicios o sistemas. Se trata de una estrategia que se pone en práctica cuando uno tiene que comparar distintas opciones para elegir la que mejor se adapta a sus necesidades y recursos. Además, la evaluación comparativa permite comparar objetivamente -o al menos lo más objetivamente posible- el rendimiento de productos o servicios a partir de criterios comunes y mensurables. La razón fundamental para adoptar esta estrategia es ayudar a los usuarios a evitar decisiones basadas principalmente en opiniones o creencias personales, o en publicidad engañosa, y a elegir con conocimiento de causa las opciones que realmente se adaptan a sus necesidades.

## 4.1.1 ENCUESTA A LAS PARTES INTERESADAS

La encuesta a las partes interesadas también incluyó temas relacionados con la definición de requisitos no funcionales, centrándose en las tecnologías utilizadas para almacenar los datos climáticos y para consultar su contenido.

**Cuadro 15. Encuesta a las partes interesadas: requisitos no funcionales.** Cada fila del cuadro corresponde a una institución diferente a la que se dirigieron las preguntas de la encuesta a las partes interesadas. Las entidades no se mencionan por su nombre, ya que la información es superflua a efectos del análisis realizado.

Papel del personal entrevistado	Software utilizado para la base de datos	¿Cuáles son las fuentes de estas bases de datos?	¿Existen API para acceder a los contenidos o consultarlos?	¿Existen una o varias interfaces web para acceder/consultar contenidos?
Interior ES Ayuda	Excel	Documentos físicos de apoyo	No	No
ES	Archivo textual	Ficheros de datos	No	No
Interior ES Ayuda	MySql	Documentos físicos de apoyo	No	No
Interior ES Ayuda	Estaciones voluntarias	Sin respuesta	No	No
Interior ES Ayuda	MCH	API Documentos físicos de apoyo Ficheros de datos Telemetría Otros DB	Sí	Sí
Plataforma Admin Interior ES Ayuda	MySQL.	API	No	Sí
Interior ES Ayuda	<i>Davis WeatherLink</i>	API	Sí	Sí

Papel del personal entrevistado	Software utilizado para la base de datos	¿Cuáles son las fuentes de estas bases de datos?	¿Existen API para acceder a los contenidos o consultarlos?	¿Existen una o varias interfaces web para acceder/consultar contenidos?
Interior ES Ayuda	Microsoft SQL Server	Telemetría Más	No	No
Interior ES Ayuda	Plataforma ClimaRed Software <i>Davis</i>	Otros DB	Sí	Sí
Interior ES Ayuda	Almacenamiento de datos en tarjeta de memoria (descarga periódica). datos almacenados en archivos .dat separados.	Ficheros de datos	No	No
Interior ES Ayuda	MySQL	Documentos físicos de apoyo Telemetría	No	No
Interior ES	<i>Biosalc</i> para 95 pluviómetros. <i>Linces</i> para estación meteorológica.	API Ficheros de datos WhatsApp	No	No
Interior ES Ayuda	<i>Davis WeatherLink</i>	API	Sí	Sí
Interior ES Ayuda	Software asociado al sistema de medición (desconocido)	Telemetría	No	No
Interior ES Ayuda	SQLite	API Ficheros de datos Telemetría	No	Sí
Interior ES Ayuda	Quién utiliza la base de datos de BYU	Base de datos dBase Documentos físicos de apoyo Ficheros de datos Otros DB	No	Sí

Cada productor de datos hidrometeoroclimáticos utiliza un software de base de datos y una fuente de datos diferentes. Las bases de datos más comunes son las basadas en SQL, como MySQL, SQLite, MCH (una de las bases de datos patrocinadas por la OMM, también basada en MySQL) y Microsoft SQL Server. En algunos casos, sin embargo, se utiliza una plataforma privada llamada WeatherLink, que se basa en el software

propietario de Davis para gestionar todo el flujo de datos, desde la recogida hasta la visualización. Davis también se basa en un SGBD de tipo SQL. Un último método de almacenamiento utilizado consiste en guardar los datos en hojas de cálculo (Excel o programas similares).

Como ya se ha comentado en el apartado **3.2**, uno de los requisitos funcionales de la plataforma es la posibilidad de importar datos hidrológicos y meteorológicos. En esta sección, gracias a la encuesta realizada, identificamos diferentes tecnologías utilizadas para el envío de datos a bases de datos, lo que esbozará los diferentes tipos de fuentes que deberá soportar la plataforma. Las fuentes de datos se han agrupado en seis categorías:

1. Documentos físicos de apoyo (por ejemplo, hojas de cálculo Excel);
2. API proporcionada por el software de gestión;
3. Telemetría;
4. Otras bases de datos (categoría genérica);
5. Ficheros de datos convertidos por el software de gestión;
6. Envío por WhatsApp.

Dado que estas fuentes de datos son de distintos tipos, serán necesarios diferentes métodos de importación para integrar la plataforma con las fuentes utilizadas actualmente y para soportar el uso de cualquier nueva fuente que se proponga en el futuro. Es importante completar los dispositivos existentes para garantizar el acceso a los datos históricos y utilizarlos con fines estadísticos.

Las dos últimas preguntas de la encuesta se refieren al uso de API o interfaces web para acceder a los datos y consultarlos. En este caso concreto, la implementación de estos servicios para la gestión de datos hidrológicos y meteorológicos no es un requisito obligatorio en el entregable del proyecto. En la actualidad, estos servicios de consulta de datos hidrometeorológicos sólo están disponibles para un número limitado de organismos productores. Sin embargo, su uso ofrece una notable mejora en todos los niveles de gestión y consulta de datos hidrológicos y meteorológicos.

## **4.1.2 ANÁLISIS DE LAS PLATAFORMAS CDMS**

Como ya se ha mencionado en la introducción del documento, el análisis comparativo de tres de las plataformas CDMS se desarrollará mediante comparaciones por áreas temáticas. Las áreas temáticas seleccionadas se ordenan de forma que se facilite la construcción de una argumentación cohesionada y lo más eficaz posible.

## 4.1.2.1 INTEROPERABILIDAD Y INTEGRACIÓN CON OpenCDMS

OpenCDMS se encuentra en la fase de desarrollo de su próxima iteración, que incorporará la primera versión de la nueva Norma del Modelo de Datos Climáticos. Está previsto que esta iteración se presente en el Congreso de la Organización Meteorológica Mundial de mayo de 2023. Para esta nueva versión está prevista la compatibilidad con una serie de sistemas CDMS existentes, como SURFACE, Climsoft, CliDE y MCH. A largo plazo, esto les garantizaría una integración y funcionalidad sin fisuras con la plataforma central de la OMM, una vez finalizado el desarrollo de la plataforma OpenCDMS.

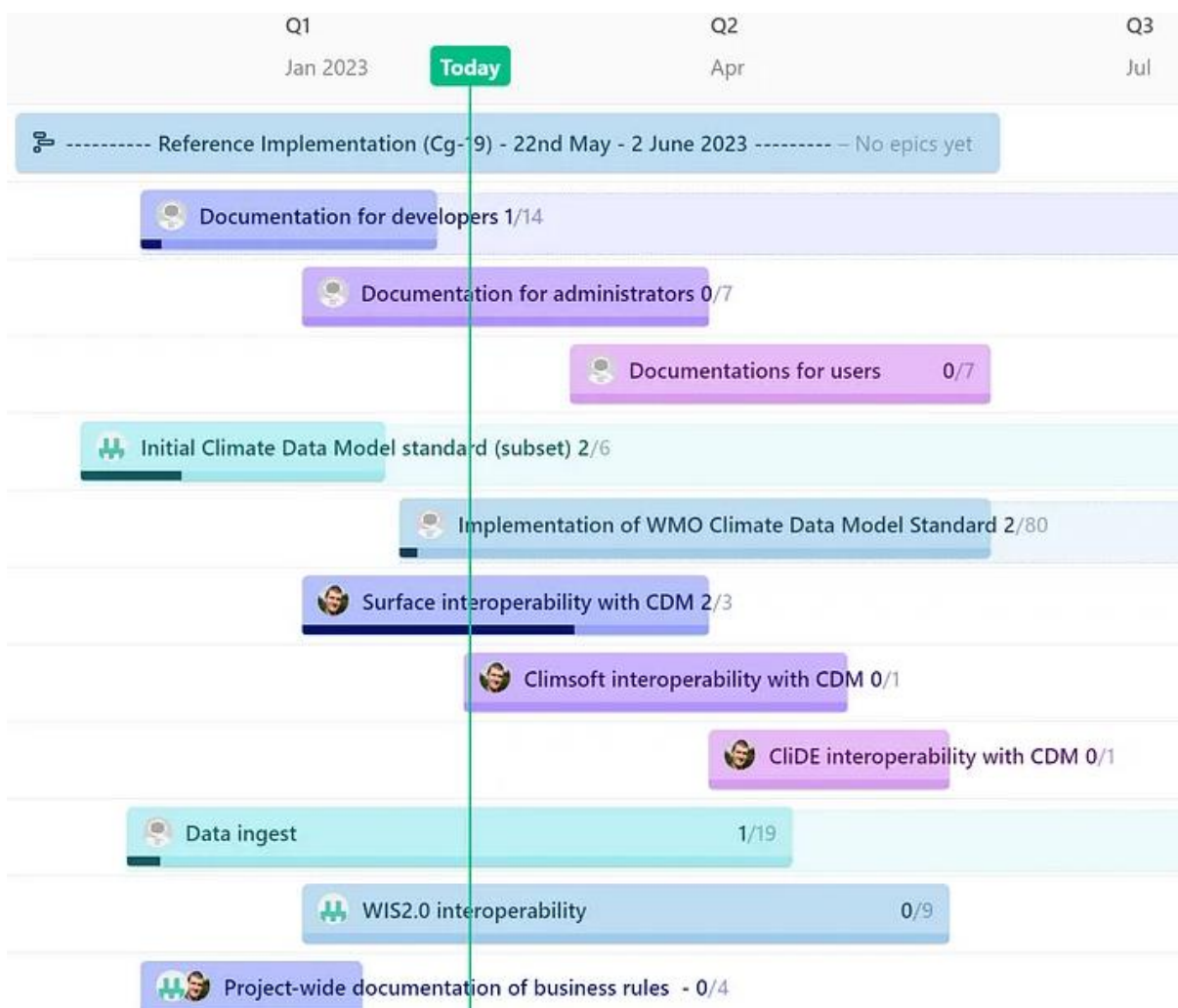


Figura 13. Hoja de ruta de interoperabilidad del proyecto OpenCDMS

Para que esto sea posible desde el punto de vista operativo, el equipo que trabaja en el desarrollo de OpenCDMS está trabajando en la creación de una capa de *interoperabilidad* o de *compatibilidad* (según el punto de vista y el nivel de integración). El objetivo último de estas capas es hacer que las bases de datos de los sistemas soportados aparezcan y se comporten de la misma nueva forma que el Modelo de Datos



Climáticos. En el caso de la *capa de interoperabilidad*, la integración será completa y efectivamente sin fisuras, creando un soporte bidireccional a nivel nativo, tanto de lectura como de escritura. En el caso de la capa de *compatibilidad*, sin embargo, la capa de integración consistirá básicamente en una capa de transformación de lectura, es decir, unidireccional de la base de datos compatible al Modelo de Datos Climáticos.

#### **4.1.2.1.1 MCH**

El proyecto OpenCDMS pretende mejorar la funcionalidad del MCH garantizando su integración con la norma SIO 2.0. Esto implica principalmente la capacidad de leer datos de la base de datos MCH a través de una *capa de compatibilidad*. Hasta la fecha, no se ha previsto una correspondencia bidireccional entre el nuevo modelo de datos climáticos y el *esquema* MCH, lo que excluye una *capa de interoperabilidad* completa.

Hay que tener en cuenta que el calendario para la compatibilidad de MCH con el modelo de datos climáticos de la OMM no está definido con precisión y no se incluye actualmente en la hoja de ruta completa de OpenCDMS, que actualmente se extiende hasta julio de 2023. No obstante, la capa de compatibilidad está prevista y, una vez realizada, permitirá a OpenCDMS interactuar con MCH.

#### **4.1.2.1.2 SURFACE**

Los desarrolladores de OpenCDMS y SURFACE tienen como objetivo principal implementar un mapeo bidireccional entre el nuevo estándar del Modelo de Datos Climáticos y el *esquema de* la base de datos SURFACE. El mapeo bidireccional permite tanto operaciones de lectura como de escritura, constituyendo a todos los efectos una *capa de interoperabilidad* completa. Aunque el desarrollo y la aplicación del nuevo MDL sigue siendo el principal objetivo de OpenCDMS, se compartirán actualizaciones periódicas de la situación en forma de estudios de casos, que demostrarán los avances logrados en la integración de OpenCDMS y SURFACE.

El desarrollo de la capa de interoperabilidad entre OpenCDMS y SURFACE ya ha comenzado y una versión inicial debería estar terminada en abril de 2023<sup>[22]</sup>. El equipo de OpenCDMS concede la máxima prioridad a la integración del MDL con la base de datos SURFACE, como puede verse fácilmente en cada actualización mensual del Sprint de Transformación. Este enfoque garantiza que la integración de los dos sistemas siga siendo una prioridad absoluta para cualquier versión del software o del MDL, incluidas las futuras.

---

<sup>22</sup> Hasta la fecha, no se prevén retrasos en el calendario.

### 4.1.2.1.3 CLIMSOFT

El argumento de la integración de Climsoft es similar al que se acaba de esgrimir para SURFACE. Climsoft forma parte de la hoja de ruta de OpenCDMS para el desarrollo de una correspondencia bidireccional entre la base de datos Climsoft y el Modelo de Datos Climáticos. Se trata, por tanto, también en este caso de la creación de una capa completa de interoperabilidad.

Según la hoja de ruta de desarrollo de OpenCDMS, el desarrollo de la capa de interoperabilidad entre OpenCDMS y Climsoft ya debería haber comenzado: de hecho, es la segunda capa prevista en orden cronológico, después de SURFACE y antes de CLIDE. Una versión inicial de esta capa de interoperabilidad debería estar terminada en mayo de 2023. Por tanto, puede decirse que el equipo de OpenCDMS también concede una prioridad razonable a la integración del MDL con la base de datos de Climsoft, y no sólo a la de SURFACE. No parece haber actualizaciones periódicas sobre el estado del desarrollo. Sin embargo, sigue siendo razonable suponer que la integración de ambos sistemas sigue siendo una prioridad para el proyecto OpenCDMS.

### 4.1.2.1.4 INTEROPERABILIDAD Y INTEGRACIÓN CON OpenCDMS - CUADRO RESUMEN

Cuadro 16. Cuadro sinóptico sobre interoperabilidad e integración con OpenCDMS

	MCH	SURFACE	Climsoft
Integración con OpenCDMS	Previsto - lanzamiento no programado con precisión	Previsto - lanzamiento inicial previsto para abril de 2023	Previsto - lanzamiento inicial previsto para mayo de 2023

### 4.1.2.2 INTEGRACIÓN CON WIS 2.0 Y wis2box

El Sistema Mundial de Telecomunicaciones (SMT) fue decisivo para el intercambio de datos hidrometeorológicos y climáticos en tiempo real. Sin embargo, el acceso a los datos del SMT requiere conocimientos especializados, lo que excluye a muchos usuarios y SMHN de países menos desarrollados. Para superar estas limitaciones, en 2007 se desarrolló el Sistema de Información de la OMM (SIO), pero este sistema sigue utilizando el SMT como servicio operativo de la infraestructura, heredando así muchas de las limitaciones del sistema original. El SIO 2.0, actualmente en desarrollo, pretende resolver estos problemas poniendo los datos hidrometeoclimáticos al alcance de todos, incluidos

los usuarios no expertos, con el uso de interfaces web de fácil manejo y metadatos de alta calidad.

La implantación del SIO 2.0 hará aún más importantes los datos hidrometeorológicos, que serán accesibles a todos y romperán las barreras infraestructurales. Además, el SIO 2.0 permitirá mejorar la comunicación y la coordinación entre los miembros de la OMM y facilitará el intercambio de datos más allá de la Vigilancia Meteorológica Mundial. WIS 2.0, junto con wis2box, una nueva herramienta de intercambio de datos, representan un importante paso adelante en la evolución del sistema GTS.

#### **4.1.2.2.1 MCH**

El proyecto WIS2-in-a-box es de 2021, mientras que el desarrollo de MCH es muy anterior, ya que el software y el manual de usuario de la base de datos datan de 2009 y aún se utilizan. Por lo tanto, parece poco probable que wis2box pueda soportar MCH de forma nativa sin desarrollos sustanciales de la plataforma, y es plausible que cualquier forma posible de compatibilidad sólo pueda ser a través de la capa de compatibilidad OpenCDMS.

#### **4.1.2.2.2 SURFACE**

Por lo que sabemos gracias a nuestros contactos con el equipo de desarrollo de SURFACE, la integración de WIS 2.0 y wis2box está en desarrollo activo, tanto de forma independiente como a través de la ya mencionada capa de interoperabilidad entre SURFACE y OpenCDMS.

#### **4.1.2.2.3 CLIMSOFT**

Por lo que respecta a Climsoft, el razonamiento es similar al expuesto anteriormente para MCH. Climsoft, en la actualidad, sigue siendo una aplicación de escritorio, con la última versión disponible públicamente que data de 2019. Una vez más, parece poco probable que pueda haber compatibilidad entre wis2box y Climsoft sin que se consiga a través de la capa de interoperabilidad proporcionada por el proyecto OpenCDMS.

#### 4.1.2.2.4 INTEGRACIÓN CON WIS 2.0 Y wis2box - CUADRO RESUMEN

Cuadro 17. Cuadro sinóptico sobre la integración con WIS 2.0 y wis2box

	MCH	SURFACE	Climsoft
Integración con WIS 2.0 y wis2box	No en desarrollo - disponible más adelante mediante integración con el proyecto OpenCDMS	En desarrollo y también disponible más adelante a través de la integración con el proyecto OpenCDMS	No en desarrollo - disponible más adelante mediante integración con el proyecto OpenCDMS

#### 4.1.2.3 INTEGRACIÓN DE METADATOS WIGOS

El Sistema Mundial Integrado de Observación de la OMM (WIGOS) es un conjunto de directrices para todos los sistemas de observación de la Organización Meteorológica Mundial (OMM), cuyo objetivo es facilitar el uso de datos procedentes de sistemas gestionados por diferentes organizaciones. El objetivo de WIGOS es garantizar la interoperabilidad y compatibilidad de los datos entre distintos sistemas mediante la aplicación de normas aceptadas internacionalmente y el uso de metadatos WIGOS. Los metadatos de descubrimiento e interpretación son necesarios para la utilización eficaz de las encuestas y observaciones de todos los sistemas de observación miembros de WIGOS.

El Manual sobre el Sistema Mundial Integrado de Observación de la OMM identifica diez categorías de metadatos de WIGOS, que definen la norma de metadatos de WIGOS (OMM, 2021b). Estas categorías se consideran de importancia fundamental para la documentación e interpretación de las observaciones realizadas, también y sobre todo para su uso futuro. Los metadatos de WIGOS tienen como objetivo proporcionar información sobre cómo se midió o clasificó la variable específica y cómo se procesaron los datos, con el fin de proporcionar a los usuarios una certeza razonable de que los datos son adecuados para el uso previsto. Como también se indica en los párrafos introductorios, la adhesión a la norma de metadatos WIGOS se considera indispensable.

##### 4.1.2.3.1 MCH

La complejidad más evidente en la aplicación de la norma de metadatos WIGOS es que, a falta de campos existentes que puedan representar dichos metadatos, el modelo de

datos debe modificarse necesariamente. Esto significa que, para todos los metadatos que no estén representados en la base de datos, hay que crear especialmente nuevos campos que puedan representarlos adecuadamente (es decir, para cada metadato que falte, hay que crear una columna especial en la tabla más adecuada del modelo de datos).

MCH es un CDMS completo en lo que respecta a su modelo de datos y, en consecuencia, ya dispone de campos especiales que pueden representar algunas de las categorías de metadatos sin necesidad de modificaciones especiales. El problema surge para todos aquellos metadatos que actualmente no son representables: el sistema está obsoleto y los cambios importantes en el modelo de datos implicarían que también habría que implementar nuevos cambios y desarrollos en la parte de la aplicación real. Esta hipótesis es poco probable, mientras que es mucho más plausible que, para el posible cumplimiento futuro de la norma de metadatos WIGOS, MCH vuelva a confiar en la capa de compatibilidad OpenCDMS. Capa que, sin embargo, actualmente no tiene una fecha definida de inicio o finalización para su desarrollo y aplicación<sup>[23]</sup>.

#### **4.1.2.3.2 SURFACE**

Como ya se ha especificado en la sección correspondiente de MCH, el cumplimiento de las normas WIGOS para metadatos requiere al menos una revisión parcial de la base de datos, para garantizar que los metadatos se corresponden realmente con los campos almacenados en la BD. El modelo de datos SURFACE se está actualizando para cumplir estos requisitos: las secciones de la plataforma que contienen metadatos que se adhieren a la norma ya están parcialmente etiquetadas y lo estarán cada vez más a medida que la colaboración con OpenCDMS y la OMM en general conduzca a un cumplimiento cada vez más estricto de las distintas normas internacionales, cumplimiento que está en el centro de los futuros desarrollos del proyecto SURFACE.

#### **4.1.2.3.3 CLIMSOFT**

También en este caso, el razonamiento relativo a Climsoft es muy similar al realizado anteriormente sobre MCH. Climsoft también es un CDMS completo y actualmente se basa en una aplicación de escritorio. Como MCH, basándonos en el estado actual de las cosas, cualquier cambio en el modelo de datos conllevaría necesariamente cambios y desarrollos adicionales también en la parte de la aplicación de escritorio. De nuevo, es mucho más probable que Climsoft confíe en la capa de interoperabilidad con OpenCDMS para el futuro cumplimiento de la norma de metadatos WIGOS. Esta capa, a diferencia de la capa MCH, es de interoperabilidad y no de compatibilidad, y además

---

<sup>23</sup> Sección 4.1.2.9.1

tiene una fecha límite de desarrollo e implementación -al menos para su primera iteración- con finalización prevista para mayo de 2023<sup>[24]</sup>.

#### 4.1.2.3.4 INTEGRACIÓN DE METADATOS WIGOS - TABLA RESUMEN

Cuadro 18. Cuadro sinóptico sobre la integración de metadatos en WIGOS

	MCH	SURFACE	Climsoft
Integración de metadatos WIGOS	Parcial - disponible más adelante a través de la integración con el proyecto OpenCDMS	Integración parcial - total en desarrollo	Parcial - disponible más adelante a través de la integración con el proyecto OpenCDMS

#### 4.1.2.4 DISPONIBILIDAD

Por disponibilidad de la plataforma se entiende la facilidad con la que uno puede obtenerla si está interesado en utilizarla. En este apartado también se analiza si la plataforma es o no de código abierto, y bajo qué tipo de licencia se pone a disposición de los usuarios que quieran instalarla en sus máquinas por el motivo que sea.

Cuando se opta por utilizar una plataforma preexistente en lugar de crear una desde cero, la facilidad de personalización se convierte en un factor esencial, en primer lugar, porque siempre hay aspectos que mejorar y, en segundo lugar, porque las plataformas preexistentes están diseñadas en función de necesidades que pueden no coincidir totalmente con las de la República Dominicana. **Si se quiere tener la mejor solución posible para el país y sus casos de uso, es imprescindible que la plataforma esté abierta a los cambios necesarios para que así sea.** Esto es imposible en el caso de las licencias de código cerrado y/o de usuario que no incluyen la posibilidad de modificación integral del software.

##### 4.1.2.4.1 MCH

MCH no está disponible gratuitamente para uso público, ya que no se ha publicado bajo una licencia de código abierto y el acceso a la aplicación sólo se concede previa solicitud.

<sup>24</sup> Sección 4.1.2.9.3

Esta limitación dificulta la posibilidad de analizar a fondo sus características y capacidades, sus puntos fuertes y sus puntos débiles.

Como MCH no es de código abierto, es difícil que alguien actualice o mejore el software. El acceso a la versión de producción puede estar restringido, sobre todo si la solicitud no procede de un instituto hidrometeorológico o climático.

#### **4.1.2.4.2 SURFACE**

SURFACE disfruta de las ventajas de estar disponible bajo la licencia de código abierto MIT, que es muy permisiva<sup>[25]</sup>. Entre las principales ventajas está la posibilidad de acceder y utilizar la aplicación en todo el mundo y en cualquier momento, así como la oportunidad para los desarrolladores de examinar y mejorar el producto y el código subyacente sin limitaciones. Estas contribuciones pueden devolverse al proyecto, lo que beneficia a toda la comunidad. Esta disponibilidad de código abierto garantiza que **SURFACE pueda personalizarse fácilmente para satisfacer las necesidades y requisitos específicos de la República Dominicana.**

#### **4.1.2.4.3 CLIMSOFT**

Climsoft es un proyecto libre y de código abierto, publicado bajo licencia GPL3 (General Public License versión 3)<sup>[26]</sup>. Esta licencia garantiza que el software siga siendo de código abierto y pueda ser utilizado, modificado y distribuido libremente por cualquiera. Climsoft es utilizado por los Servicios Meteorológicos e Hidrológicos Nacionales (SMHN) de 19 países en desarrollo, situados principalmente en el África subsahariana. Para que los usuarios puedan acceder fácilmente al software, el código fuente de Climsoft está disponible en los repositorios de Github.

Como en el caso de SURFACE, la accesibilidad del código abierto hace que Climsoft pueda personalizarse según los requisitos del proyecto o las necesidades de las organizaciones que lo adopten, aunque su configuración como aplicación de escritorio le resta versatilidad.

Junto con el código fuente, también está disponible una documentación bastante extensa y completa, que incluye manuales de usuario, guías de instalación y especificaciones técnicas. Esta documentación ayuda a los usuarios a configurar y

---

<sup>25</sup> <https://github.com/opencdms/surface/blob/main/LICENSE>

<sup>26</sup> <https://www.gnu.org/licenses/gpl-3.0.html>

utilizar el software y a resolver los problemas que puedan surgir, especialmente los más comunes o recurrentes.

#### 4.1.2.4.4 DISPONIBILIDAD – CUADRO RECAPITULATIVO

Cuadro 19. Cuadro recapitulativo de la disponibilidad actual

	MCH	SURFACE	Climsoft
Disponibilidad	Aplicación de escritorio - Sólo Windows	Visor web	Aplicación de escritorio - Sólo Windows

#### 4.1.2.5 ACCESIBILIDAD

La accesibilidad del software es crucial para garantizar que los usuarios finales puedan utilizar el producto de forma fácil e intuitiva, independientemente de sus capacidades o de la plataforma que utilicen. En el ámbito del acceso a los datos hidro-meteorológicos, la accesibilidad de los programas informáticos a través de un visor web es cada vez más importante con el avance de la tecnología y el uso generalizado de los dispositivos móviles. Gracias al visor web, los usuarios pueden acceder a la información climática desde cualquier lugar y en cualquier momento, utilizando un dispositivo móvil como un teléfono o una tableta. Esto es especialmente útil para quienes trabajan sobre el terreno o se desplazan constantemente, ya que les permite acceder a los datos en tiempo real y tomar decisiones informadas con rapidez y eficacia.

Además, un visor web permite acceder al programa a cualquier persona con acceso a Internet, independientemente del tipo de dispositivo utilizado para navegar. Sin embargo, para que un programa informático sea realmente accesible, debe estar diseñado para que todos los usuarios puedan acceder a sus funcionalidades de forma fácil e intuitiva, sin encontrar barreras ni dificultades. La creación de software accesible beneficia a todos los que lo utilizan, ya que aumenta la eficacia y eficiencia de las operaciones y la satisfacción de todas las categorías de usuarios.

##### 4.1.2.5.1 MCH

El MCH se desarrolló inicialmente en 1996 y desde entonces no ha sufrido ninguna actualización significativa en cuanto a diseño o enfoque infraestructural. Se limita esencialmente a una aplicación de escritorio que sólo puede instalarse en máquinas



Windows. El desarrollo de un visor web para MCH requiere la programación de una capa API adicional, ya que, por razones principalmente cronológicas, durante la creación del producto no se pensó en un visor web nativo. El alcance de las capacidades del MCH y hasta qué punto pueden utilizarse realmente a través de un visor web sigue siendo incierto. Las capas API desarrolladas para MCH y disponibles para su consulta se limitan a permitir la consulta de tablas y datos de bases de datos, sin ninguna sofisticación o procesamiento particular.

#### **4.1.2.5.2 SURFACE**

SURFACE se diseñó y desarrolló en 2018 a partir de una reescritura completa de un CDMS preexistente. Esta reciente reescritura permitió el uso de tecnología actualizada y un enfoque infraestructural más moderno que muchos sistemas de CDMS comparables. SURFACE, gracias a su diseño, puede utilizarse desde cualquier navegador web, independientemente del sistema operativo y, potencialmente, incluso del dispositivo. Por el momento, la experiencia está optimizada desde ordenadores de sobremesa o portátiles, pero el software puede actualizarse para que la experiencia sea utilizable también desde un teléfono o una tableta.

#### **4.1.2.5.3 CLIMSOFT**

Climsoft comenzó como Climsoft Desktop para Windows, por lo que se limitaba a una aplicación de escritorio que sólo podía instalarse y utilizarse en máquinas basadas en Windows. Esta aplicación inicial se implementó originalmente utilizando la base de datos Microsoft Access y posteriormente evolucionó con la adopción de Visual Basic y MySQL. La versión 4 se reescribió en Visual Basic .NET y se publicó como software de código abierto bajo licencia GPL v3.

Posteriormente, Climsoft evolucionó -al menos en lo que respecta a las intenciones declaradas de los desarrolladores- hacia una solución multidispositivo/multiplataforma rebautizada Climsoft CDMS. Este enfoque diferente, mucho más acorde con los tiempos que corren, se basa en el concepto de que Climsoft se instala en un servidor, local o remoto, y cada instalación de servidor puede incluir luego, según sea necesario, uno o varios componentes diseñados para exponer su API y aprovechar los diversos servicios web y microservicios programados. Todas las conexiones externas utilizan HTTPS como estándar de transmisión segura. La base de código de Climsoft es única, independientemente de si la aplicación final es de escritorio, web o móvil, lo que simplifica enormemente el desarrollo. En un principio, estaba prevista una versión web totalmente funcional y accesible, pero no está claro si llegará a desarrollarse y ponerse a disposición del público, o si el proyecto OpenCDMS se hará cargo de esa parte tras el desarrollo de la capa de interoperabilidad con Climsoft.

#### 4.1.2.5.4 ACCESIBILIDAD – CUADRO RECAPITULATIVO

Cuadro 20. Cuadro sinóptico de accesibilidad

	MCH	SURFACE	Climsoft
Accesibilidad	Fuente cerrada	Código abierto - Licencia MIT	Código abierto - licencia GPL3

#### 4.1.2.6 BASE DE DATOS

Un sistema de gestión de bases de datos (SGBD) es un programa informático que permite organizar, gestionar y acceder a los datos de forma eficaz y segura. El SGBD proporciona herramientas para la definición de la estructura de los datos, su manipulación y la gestión del acceso de los usuarios. Además, el SGBD ofrece funciones de seguridad para proteger los datos sensibles de accesos no autorizados. Un motor de base de datos (DB Engine), por su parte, es el motor de almacenamiento que realiza las operaciones de almacenamiento, búsqueda y recuperación de datos, utilizando complejos algoritmos para garantizar la redundancia y la integridad.

La velocidad y la escalabilidad son dos factores críticos a la hora de elegir un DBMS y un motor de DB. Una vez definida la estructura de datos e introducidos éstos en el sistema, la velocidad de recuperación de los datos resulta crucial para garantizar el funcionamiento eficaz del sistema. La velocidad es fundamental porque la mayoría de las aplicaciones requieren una respuesta inmediata a los datos solicitados. Si un sistema tarda en recuperar los datos, los usuarios pueden desmotivarse y dejar de utilizar la aplicación. Además, si la carga de trabajo aumenta, la velocidad del sistema puede disminuir, comprometiendo su eficacia.

También hay que tener en cuenta que, **a medida que aumentan el volumen de datos y la complejidad de los análisis, la escalabilidad del sistema se convierte en un factor importante para tener en cuenta a la hora de elegir el DBMS y el DB Engine.** La escalabilidad, de hecho, se refiere a la capacidad del sistema para manejar grandes cantidades de datos y adaptarse a las necesidades futuras del proyecto.

##### 4.1.2.6.1 MCH

La versión estándar de MCH utiliza MySQL (MySQL 5.1) como sistema de gestión de bases de datos (SGBD). La versión 5.1 es de 2008 y, por tanto, anterior a la era de los macrodatos.

Además, no ha recibido ayudas desde hace más de una década (las ayudas finalizaron en diciembre de 2013). La compatibilidad de OpenCDMS con esta versión de MySQL es muy limitada y no se admite directamente. Además, MCH tiene una versión básica con un número predeterminado de tablas, pero la personalización que requiere cada organización para satisfacer sus necesidades específicas da lugar a una falta de normalización. La forma en que está estructurado el MCH hace muy improbable que haya siquiera dos organizaciones diferentes que tengan exactamente la misma estructura de tablas (lo que técnicamente se llama un esquema).

La actualización del SGBD utilizado por el MCH a una versión más reciente y funcional no sería en absoluto un proceso sencillo. Muchos de los nombres de tablas y columnas del esquema de la base de datos MCH utilizan palabras que, en versiones más recientes de MySQL, están reservadas (es decir, no pueden utilizarse como nombres de dichas tablas o campos porque están reservadas por el sistema). Para corregir este problema, estas palabras reservadas deben cambiarse o encerrarse en superíndices invertidos al restaurar el archivo .sql tras el proceso de actualización. Por ejemplo, mientras que "data" no es una palabra reservada en MySQL 5.1, se considera como tal en versiones más recientes, y por lo tanto no se puede utilizar. Una consecuencia directa de esto es que las implementaciones MCH existentes detenidas en MySQL 5.1, que son la gran mayoría, no pueden actualizarse directamente a una versión más reciente de MySQL sin modificar manualmente el esquema. Manualmente, porque cada implementación de base de datos personalizada debe examinarse cuidadosamente antes de la actualización para garantizar que las tablas personalizadas cumplen las normas de la última versión del SGBD.

El uso de MySQL 5.1 como DBMS va unido al uso de MyISAM como motor de almacenamiento en la mayoría de las versiones de producción de MCH, ya que MyISAM era el motor de almacenamiento por defecto de MySQL hasta la versión 5.5. Sin embargo, MyISAM no admite ningún tipo de claves *foráneas*, lo que supone un problema importante para quienes deseen integrar esta funcionalidad en MCH. Todas las claves externas de las tablas no estándar deben añadirse manualmente, por las mismas razones explicadas anteriormente en relación con la actualización del RDMS. Una vez más, esta actividad no puede automatizarse por completo, lo que la convierte en un proceso potencialmente largo y complejo.

Se están realizando esfuerzos para migrar el MCH a las versiones más modernas y actualizadas de la base de datos MySQL (MySQL 8.x) y del motor de almacenamiento MariaDB. Sin embargo, debido a las restricciones y limitaciones mencionadas en los párrafos anteriores, el proceso de migración dista mucho de ser sencillo y directo. Sigue siendo incierto si las ventajas de la migración compensarían las posibles complejidades y riesgos y si, por el contrario, la adopción de un sistema alternativo de gestión de bases de datos sería una solución más viable y eficaz a medio y largo plazo.

#### 4.1.2.6.2 SURFACE

SURFACE utiliza TimescaleDB como SGBD. La lógica de TimescaleDB está diseñada específicamente para manejar grandes cantidades de *series temporales* y datos *espaciales*, que constituyen la mayoría de los datos de un CDMS. TimescaleDB es intuitiva, totalmente compatible con SQL y ofrece una escalabilidad comparable a la de las bases de datos NoSQL. También utiliza como base la sólida y fiable plataforma PostgreSQL. Juntos, estos dos elementos le permiten alcanzar un rendimiento de consulta entre 10 y 100 veces (dependiendo del escenario) más rápido que PostgreSQL, InfluxDB y MongoDB<sup>[27]</sup>.

TimescaleDB, al estar basado en PostgreSQL, utiliza el mismo robusto motor de almacenamiento incorporado, ofreciendo una uniformidad sin fisuras en todo el sistema de gestión de bases de datos. La falta de soporte para motores alternativos al motor PostgreSQL estándar puede resultar en una solución algo menos versátil. Sin embargo, el motor incorporado compensa esto con sus potentes capacidades de rendimiento, y las soluciones específicas adoptadas por TimescaleDB se desarrollaron e implementaron ad hoc para el motor de gestión estándar de PostgreSQL.

#### 4.1.2.6.3 CLIMSOFT

Climsoft admite más de una solución tanto en lo que respecta al sistema de gestión de bases de datos como al motor de almacenamiento. El manual de Climsoft recomienda utilizar PostgreSQL con la extensión espacial PostGIS para su uso junto con el servidor de bases de datos Climsoft CDMS. PostGIS es una de las bases de datos de código abierto más maduras para datos espaciales. Además, en el momento en que se desarrolló Climsoft, PostGIS era la única opción con soporte integrado para campos ráster (necesarios para la integración de datos de malla).

Como alternativa a la combinación PostgreSQL más PostGIS, Climsoft también admite la opción MySQL/MariaDB. Sin embargo, algunas de las funcionalidades más avanzadas pueden no estar disponibles si se elige esta segunda opción, debido principalmente a que MySQL no es compatible con OGC (limitaciones de espacio).

El rendimiento de ambas soluciones está a la altura de los tiempos, sobre todo en el caso de la primera opción (PostgreSQL + PostGIS), pero subsiste el problema de la escalabilidad, que se aborda con más éxito mediante el uso de TimescaleDB, optimizado tanto para los datos espaciales como para los datos de series temporales (que constituyen la cantidad más importante de datos de entrada almacenados en un

---

<sup>27</sup> [Notas de rendimiento](#)

sistema de gestión de bases de datos climáticas, ya que las distintas mediciones están inextricablemente ligadas al instante en que se toman).

#### 4.1.2.6.4 BASE DE DATOS – CUADRO RECAPITULATIVO

Cuadro 21. Cuadro sinóptico de la tecnología de bases de datos

	MCH	SURFACE	Climsoft
Tecnología de bases de datos	MySQL 5.1 y MyISAM	TimescaleDB	PostgreSQL y PostGIS o MySQL 8.x y MariaDB

#### 4.1.2.7 LENGUAJE DE PROGRAMACIÓN

La elección del lenguaje de programación desempeña un papel crucial para facilitar la actualización y el mantenimiento del software. Los lenguajes antiguos suelen ser difíciles de mantener, tanto por la dificultad de implementar funcionalidades más acordes con un uso moderno de las herramientas informáticas como por la dificultad de encontrar personas competentes en su uso.

Por otra parte, los lenguajes de programación más recientes suelen tener una base de usuarios mucho mayor, lo que facilita mucho la resolución de problemas y la búsqueda de personal adecuado. Además, suelen soportar librerías de funciones muy extensas que facilitan enormemente la implementación de funcionalidades que llevarían bastante más tiempo si hubiera que escribirlas desde cero.

##### 4.1.2.7.1 MCH

MCH se escribió en Delphi/Object Pascal, un lenguaje que hoy en día apenas se utiliza para desarrollar software nuevo y se emplea casi exclusivamente para mantener software heredado.

##### 4.1.2.7.2 SURFACE

Cuando se estaba diseñando SURFACE, el equipo de desarrollo optó por utilizar Python como lenguaje de programación, debido al amplio soporte y a la cantidad de bibliotecas

disponibles. La comunidad de usuarios de Python es muy amplia, ya que este lenguaje se encuentra desde hace años entre los cinco lenguajes de programación más populares y demandados, como indican diversas encuestas<sup>[28][29]</sup> y sondeos<sup>[30]</sup>. Una consecuencia fundamental de una comunidad tan extendida es la facilitación de la resolución rápida de problemas, mientras que en Pascal no está garantizado el mismo nivel de apoyo y capacidad de resolución de problemas.

### 4.1.2.7.3 CLIMSOFT

Climsoft utiliza Python como lenguaje principal del lado del servidor, debido a su versatilidad y fiabilidad en la construcción de back-ends web y, al mismo tiempo, a la peculiaridad de disponer de un conjunto extremadamente rico de librerías científicas y geoespaciales, muy convenientes en la gestión de un CDMS. Además, Python es adecuado tanto para programadores noveles -por su facilidad de aprendizaje y su amplia red de soporte y documentación- como para desarrolladores experimentados -que pueden aprovechar al máximo sus capacidades y rendimiento-.

Las capas de back-end y API se desarrollan con Django. Django (GeoDjango para ser precisos) es un framework de Python, que se utiliza ampliamente en una gran variedad de aplicaciones debido a su amplio conjunto de características, su enfoque en la seguridad, su facilidad y velocidad de desarrollo y su flexibilidad general. El framework Django es una elección adecuada para construir la capa API, pero su verdadera fuerza en el caso de la implementación en un CDMS reside en su potencial desde una perspectiva científica y geoespacial.

### 4.1.2.7.4 LENGUAJE DE PROGRAMACIÓN - CUADRO RECAPITULATIVO

Cuadro 23. Cuadro resumen de lenguajes de programación

	MCH	SURFACE	Climsoft
Lenguaje de programación	Delphi/Object Pascal	Python	Python

<sup>28</sup> <https://bootcamp.berkeley.edu/blog/most-in-demand-programming-languages/>

<sup>29</sup> <https://www.northeastern.edu/graduate/blog/most-popular-programming-languages/>

<sup>30</sup> <https://survey.stackoverflow.co/2022/>

## **4.1.2.8 VERSIONADO DE SOFTWARE**

*El versionado de software* es un proceso esencial para gestionar el código fuente de un programa informático, que permite llevar un registro de los cambios y actualizaciones realizados e identificar la versión concreta instalada. En este contexto, el seguimiento de versiones y la gestión de esquemas de bases de datos son igualmente importantes. Estos procesos contribuyen a garantizar que todas las partes interesadas estén informadas sobre la evolución del software y los motivos de los cambios.

Un control de versiones adecuado evita graves problemas en un proyecto de software, ya que permite determinar qué cambios se han hecho en el código y quién los ha hecho, sobre todo si varias personas trabajan al mismo tiempo en el mismo proyecto. Esto garantiza la capacidad de reproducir el comportamiento del software en versiones anteriores, identificar a los autores de las modificaciones y evaluar su calidad, y gestionar los conflictos entre distintas versiones del código fuente.

A esto hay que añadir las implicaciones para la seguridad de un control de versiones adecuado, ya que permite a los desarrolladores detectar y corregir rápidamente vulnerabilidades y agujeros de seguridad en el código, evitando posibles daños tanto a los datos como a la reputación de la organización que utiliza el software.

Por último, el control de versiones también evita problemas de compatibilidad entre distintas versiones del software y entre el software y las bases de datos asociadas, lo que reduce las dificultades en la migración de datos y el mantenimiento del software, así como el riesgo de errores de programación y mal funcionamiento.

### **4.1.2.8.1 MCH**

MCH carece de un control adecuado del código fuente y de las versiones, lo que dificulta la determinación de la versión concreta del software instalado y de la parte responsable de los cambios y actualizaciones realizados en el código o en cualquier parte del software, así como de los motivos de dichos cambios. Además, en una base de datos MCH no es posible determinar cómo ha cambiado el esquema a lo largo del tiempo.

### **4.1.2.8.2 SURFACE**

Toda la base de código de SURFACE es accesible a través del control de código fuente y fácilmente rastreable. Esto permite a las partes interesadas seguir la evolución del software, quién hizo los cambios, cuándo se hicieron y las razones que los motivaron. Además, el versionado del esquema de la base de datos se registra meticulosamente

mediante migraciones de software, lo que permite seguir los cambios y modificaciones de la base de datos SURFACE a lo largo del tiempo. Cualquier cambio en el esquema de la base de datos SURFACE se registra cuidadosamente mediante migraciones de software<sup>[31]</sup>.

### 4.1.2.8.3 CLIMSOFT

Climsoft, al ser un proyecto de código abierto, pone su código fuente a disposición del público. Los interesados pueden acceder al código fuente en los repositorios de Github, una popular plataforma para compartir y colaborar en proyectos de código abierto.

Además de alojar el código fuente, la página Github de Climsoft tiene una sección dedicada a la supervisión de las versiones del software. En esta sección se registran las distintas versiones de la base de código mediante las etiquetas y descripciones correspondientes a los diversos cambios y actualizaciones realizados.

Este enfoque de la gestión de proyectos no sólo ayuda a los usuarios a mantenerse informados sobre el desarrollo de software, sino que también permite a los desarrolladores colaborar eficazmente. Los desarrolladores pueden hacer un seguimiento de los cambios realizados por los demás en el código base, lo que garantiza que todo el mundo conoce el estado de las cosas y evita conflictos o trabajo redundante. El uso de GitHub u otras plataformas similares también permite a la comunidad de usuarios de software aportar comentarios, informar de problemas y sugerir mejoras para futuras versiones.

### 4.1.2.8.4 VERSIONADO DE SOFTWARE – TABLA RESUMEN

Cuadro 24. Cuadro sinóptico sobre el versionado de software

	MCH	SURFACE	Climsoft
Versiones de software	No	Sí	Sí

<sup>31</sup> <https://github.com/opencdms/surface/tree/main/api/wx/migrations>



## **4.1.2.9 HOJA DE RUTA**

La hoja de ruta es una herramienta de planificación que describe las principales etapas e hitos cruciales en el proceso de desarrollo de un proyecto a medio o largo plazo. Es un documento de referencia que establece la dirección estratégica del proyecto, sus prioridades y las actividades que deben llevarse a cabo para alcanzar sus objetivos. Una hoja de ruta bien definida ayuda a quienes trabajan en el desarrollo a gestionar el proyecto de forma más eficiente, identificando los recursos necesarios y definiendo el calendario para la realización de las actividades.

Además de todo esto, la hoja de ruta permite a todas las partes implicadas, desde las partes interesadas hasta los clientes, tener una visión clara del estado del proyecto y los resultados previstos, lo que mejora la comunicación y la colaboración entre las partes implicadas. Una hoja de ruta precisa y bien planificada es esencial para el éxito del proyecto, ya que ayuda a evitar imprevistos y sorpresas por el camino y a garantizar el control de los costes y los recursos empleados.

### **4.1.2.9.1 MCH**

El MCH no tiene una hoja de ruta publicada, lo que dificulta determinar la dirección del proyecto y las prioridades futuras. No obstante, cabe suponer que el proyecto pretende incorporar los avances ya establecidos en SURFACE.

De la hoja de ruta de los hitos pasados se deduce que el esfuerzo en el desarrollo de MCH se centra en la creación y mejora de un sistema con visor/cliente web, y en la creación de una API para la consulta de la BD de forma remota. La última versión de producción disponible de MCH es de 2018: en fechas posteriores, solo se pueden encontrar las API, que solo se prepararon para consulta<sup>[32]</sup>.

### **4.1.2.9.2 SURFACE**

SURFACE cuenta con una hoja de ruta bien definida y accesible al público, elaborada en colaboración con la Organización Meteorológica Mundial (OMM). El objetivo es incorporar los últimos y más avanzados estándares tecnológicos en beneficio de todos los usuarios. Como se ha descrito anteriormente, el objetivo actual de la hoja de ruta de desarrollo de SURFACE es la integración con la nueva iteración de OpenCDMS y la

---

<sup>32</sup> Encontrará una versión actualizada del cliente web MCH en este enlace: <http://hydromch.com:3000/>

primera versión del nuevo Modelo de Datos Climáticos, y se espera que los trabajos finalicen en abril de 2023.

### 4.1.2.9.3 CLIMSOFT

La versión actual de Climsoft es la 4.1 (más concretamente, la 4.1.12). La hoja de ruta publicada indica que esta versión se remonta a 2019 y, según la hoja de ruta publicada, debería haber alcanzado la versión 6. La versión 6 dejaría obsoleta la versión de sobremesa y pasaría a ser totalmente compatible con la versión multidispositivo, con toda la funcionalidad desarrollada anteriormente para la versión de sobremesa.

Sin embargo, por el momento, esta versión 6 no está disponible, y la versión web de Climsoft está aún en fase beta, no soportando actualmente toda la funcionalidad necesaria para la gestión y el tratamiento de datos CDMS.

Como se puede ver en la hoja de ruta del sitio web de Climsoft, está prevista la integración de Climsoft en OpenCDMS, con un trabajo inicial de desarrollo de la integración programado para mayo de 2023<sup>[33]</sup>.

### 4.1.2.9.4 HOJA DE RUTA – CUADRO RECAPITULATIVO

Cuadro 25. Cuadro resumen de la hoja de ruta

	MCH	SURFACE	Climsoft
Mapa de carreteras	No disponible	Disponible y respetado	Disponible pero no respetado

### 4.1.2.10 AUTORIZACIONES Y AISLAMIENTO DE DATOS

En la gestión de bases de datos, la asignación de autorizaciones de acceso a los datos - especialmente para su modificación y/o supresión- es crucial para garantizar la seguridad y protección de la información sensible. En este contexto concreto, como es el de la gestión de estaciones hidrológicas y meteorológicas por diferentes entidades, el escenario está bien perfilado: es importante que cada entidad productora sólo pueda

---

<sup>33</sup> Mientras que en el caso de SURFACE estamos seguros del estado de los trabajos, en lo que respecta a Climsoft no podemos determinar si la hoja de ruta se está cumpliendo o no.

acceder a sus propias estaciones con permisos de edición y no a las de otras entidades, conservando los permisos de visualización de los datos de todas las demás estaciones. Para lograrlo, **se necesita una estrategia eficaz de aislamiento de datos.**

El aislamiento de los datos puede aplicarse mediante diversos mecanismos, como la definición de reglas de acceso específicas para cada entidad o la división de los datos en esquemas de base de datos separados. Cuando se aplica correctamente una estrategia de aislamiento de datos, cada entidad puede gestionar sus estaciones de forma independiente, sin riesgo de conflictos o accesos no autorizados.

La importancia del aislamiento de datos en este contexto es crucial y resulta imprescindible. Esto se debe, por un lado, a que los datos hidrometeorológicos se utilizan a menudo para tomar decisiones importantes en diversos sectores, por lo que deben ser lo más fiables posible. Por otro lado, depende del hecho de que, sin permisos restrictivos de este tipo, podrían surgir conflictos entre las distintas entidades, como consecuencia de modificaciones no autorizadas, relacionadas con posibles manipulaciones intencionadas o errores humanos. Una gestión adecuada de los datos es, por tanto, esencial para garantizar la seguridad y protección de la información y tomar decisiones fundamentadas basadas en datos correctos.

Los tres sistemas (MCH, SURFACE<sup>[34]</sup> y Climsoft) disponen de su propio sistema de gestión de usuarios y de las autorizaciones asociadas a ellos, pero ninguno de ellos tiene la funcionalidad de separar las autorizaciones de forma que se garantice una granularidad de aislamiento de datos funcional para los objetivos del proyecto. Por lo tanto, todo el sistema debe desarrollarse en los tres casos.

#### **4.1.2.10.1 AUTORIZACIONES Y AISLAMIENTO DE DATOS – CUADRO RECAPITULATIVO**

**Cuadro 26. Cuadro recapitulativo sobre autorizaciones y aislamiento de datos**

	MCH	SURFACE	Climsoft
Autorizaciones y aislamiento de datos	Sistema de autorización implantado (método desconocido), aislamiento de datos no disponible	Sistema de autorización implementado (vía Django), aislamiento de datos no disponible	Sistema de autorización implementado (vía Django), aislamiento de datos no disponible

<sup>34</sup> El sistema de gestión de usuarios SURFACE se analiza con más detalle en la sección **5.2.2**.

#### **4.1.2.11 CONSULTAS A BASES DE DATOS DE TERCEROS SIN ARCHIVAR**

Entre las diversas opciones disponibles para consultar datos de distintas fuentes, también existe la opción de construir un cliente que permita consultar datos de distintas fuentes sin necesidad de una única base de datos central en la que almacenar toda la información. En su lugar, los datos se almacenan localmente en dispositivos individuales o en bases de datos descentralizadas más pequeñas a las que se accede a través de una red distribuida.

Este enfoque ofrece numerosas ventajas, como una mayor seguridad y privacidad, un menor riesgo de pérdida o corrupción de datos y una mayor escalabilidad y flexibilidad. También puede ayudar a reducir los costes y la complejidad asociados a la gestión de una base de datos centralizada, sobre todo en sistemas a gran escala u organizaciones con equipos dispersos geográficamente.

Sin embargo, el uso de clientes para consultar bases de datos de terceros sin almacenamiento centralizado también presenta algunos inconvenientes. Por ejemplo, puede resultar más difícil mantener la coherencia de los datos entre los distintos nodos de la red, y puede ser más complicado gestionar los derechos de acceso a los datos. Además, la latencia de la red puede afectar al rendimiento general del sistema. Con las mismas tecnologías utilizadas para el desarrollo de la base de datos, la plataforma necesita por tanto un mayor nivel de optimización -y la incorporación de tecnologías más punteras, como una capa de caché- para mantenerse a la altura de la velocidad de consulta de una base de datos centralizada.

Actualmente, ninguno de los tres sistemas (MCH, SURFACE y Climsoft) admite la *consulta* de bases de datos de terceros sin almacenamiento de datos. En cuanto al MCH, habría que desarrollarlo desde cero. SURFACE y, en cierta medida, Climsoft ya disponen de una capa API REST estructurada, lo que facilita el desarrollo de esta funcionalidad más que en el caso de MCH.

##### **4.1.2.11.1 CONSULTAS A BASES DE DATOS DE TERCEROS SIN ALMACENAMIENTO – TABLA RESUMEN**

**Cuadro 27. Cuadro recapitulativo de las consultas a bases de datos de terceros sin almacenamiento**

	MCH	SURFACE	Climsoft
Consultas de bases de datos de terceros sin almacenamiento	No se ha aplicado	No se ha aplicado	No se ha aplicado

## 4.2 PLATAFORMA SUGERIDA

Como se destaca en el análisis comparativo de la sección 3.1.2, SURFACE y su visor web cumplieron plenamente las características exigidas en el anuncio del proyecto, superando incluso en funcionalidad a las demás plataformas de referencia.

En este capítulo, sin embargo, nos hemos centrado en un análisis comparativo de los sistemas CDMS conformes al proyecto OpenCDMS y a las normas de referencia internacionales. Como puede verse en la tabla resumen, SURFACE parece ser la mejor opción como punto de partida para el desarrollo de la futura plataforma CDMS para la República Dominicana. Sus puntos fuertes son considerables, tanto por la tecnología utilizada como por su naturaleza de código abierto, elemento esencial si se quiere personalizar un software sin límites en lugar de desarrollarlo completamente desde cero.

**Cuadro 28. Análisis resumido de las características de las plataformas abiertas**

	MCH	SURFACE	Climsoft
Integración con OpenCDMS	Previsto - lanzamiento no programado con precisión	Previsto - lanzamiento inicial previsto para abril de 2023	Previsto - lanzamiento inicial previsto para mayo de 2023
Integración con WIS 2.0 y wis2box	No en desarrollo - disponible más adelante mediante integración con el proyecto OpenCDMS	En desarrollo y también disponible más adelante a través de la integración con el proyecto OpenCDMS	No en desarrollo - disponible más adelante mediante integración con el proyecto OpenCDMS
Integración de metadatos WIGOS	Parcial - disponible más adelante a través de la integración con el proyecto OpenCDMS	Integración parcial - total en desarrollo	Parcial - disponible más adelante a través de la integración con el proyecto OpenCDMS
Disponibilidad	Aplicación de escritorio - Sólo Windows	Visor web	Aplicación de escritorio - Sólo Windows

	MCH	SURFACE	Climsoft
Licencia y accesibilidad	Fuente cerrada	Código abierto - Licencia MIT	Código abierto - licencia GPL3
Tecnología de bases de datos	MySQL 5.1 y MyISAM	TimescaleDB	PostgreSQL y PostGIS o MySQL 8.x y MariaDB
Lenguaje de programación	Delphi/Object Pascal	Python	Python
Versiones de software	No	Sí	Sí
Mapa de carreteras	No disponible	Disponible y respetado	Disponible pero no respetado
Autorizaciones y aislamiento de datos	Sistema de autorización implantado (método desconocido), aislamiento de datos no disponible	Sistema de autorización implementado (vía Django), aislamiento de datos no disponible	Sistema de autorización implementado (vía Django), aislamiento de datos no disponible
Consultas de bases de datos de terceros sin almacenamiento	No aplicado	No se ha aplicado	No se ha aplicado

Con base en los objetivos y criterios definidos, y a la luz de los resultados de los análisis descritos anteriormente, la elección sólo puede recaer en SURFACE como base para el desarrollo del CDMS dominicano. En los capítulos siguientes, definiremos lo que se considera necesario para crear una plataforma ideal para la situación específica de la República Dominicana. También se espera que la futura plataforma pueda convertirse en una referencia para cualquiera que se enfrente a situaciones similares en el desarrollo de un CDMS.

Antes de proceder a la descripción de futuros desarrollos, es importante reflexionar brevemente sobre las características inherentes al software de código abierto y por qué este criterio se consideró de suma importancia para la elección del proyecto.

## 4.2.1 PORQUE EL CÓDIGO ABIERTO

La elección de una solución de código abierto para la plataforma CDMS en la República Dominicana estuvo motivada por las diversas razones expuestas en varias secciones del documento. En este subcapítulo se presentan algunas ideas para completar el análisis.

El uso de software de código abierto ofrece ventajas como costes reducidos o licencias gratuitas, lo que permite ahorrar en licencias y costes de distribución. Además, en este caso concreto, el apoyo a SURFACE y a sus distintas iteraciones también está previsto a medio y largo plazo, ya que el sistema forma parte integrante del proyecto OpenCDMS y, por consiguiente, se encuentra bajo la tutela más o menos directa de la OMM<sup>[35]</sup>. En el caso de SURFACE, la licencia es libre y muy permisiva<sup>[36]</sup>.

La permisividad de la licencia permite personalizar el código fuente para adaptarlo a las necesidades específicas de desarrolladores y usuarios, lo que ofrece una mayor flexibilidad. Esta flexibilidad y capacidad de personalización son fundamentales para el proyecto en cuestión, ya que **las modificaciones necesarias para adaptar el software a las necesidades de la República Dominicana sólo pueden realizarse si el código puede consultarse y personalizarse libremente.**

La transparencia del código fuente de los programas informáticos de código abierto también permite detectar y resolver cuanto antes los fallos o vulnerabilidades de seguridad gracias a la inspección de la comunidad de desarrolladores, con la que se puede colaborar activamente para intercambiar ideas, recursos y mejoras. **El equipo del proyecto colaboró estrechamente con la comunidad de desarrolladores de SURFACE durante la ejecución del proyecto, debatiendo a fondo todas las consideraciones relativas al software, su uso y futuros desarrollos antes de proponerlos.**

**En conclusión, la elección de una solución de código abierto para la plataforma CDMS de la República Dominicana ofrece ventajas como la reducción de costes, una mayor flexibilidad y transparencia del código fuente, lo que permite una personalización precisa y la capacidad de identificar y resolver cualquier problema de seguridad a tiempo. Esta elección se ha evaluado cuidadosamente y se considera la solución más adecuada a las necesidades del proyecto.**

---

<sup>35</sup> Apartados 4.1.2.1, 4.1.2.2 y 4.1.2.3

<sup>36</sup> Sección 4.1.2.4.2

# 5 PLATAFORMA CDMS PARA LA REPÚBLICA DOMINICANA

Las razones de la elección de SURFACE como sistema base para el desarrollo de la futura plataforma CDMS dominicana han sido ampliamente detalladas y justificadas en el capítulo anterior y a lo largo de todo el documento.

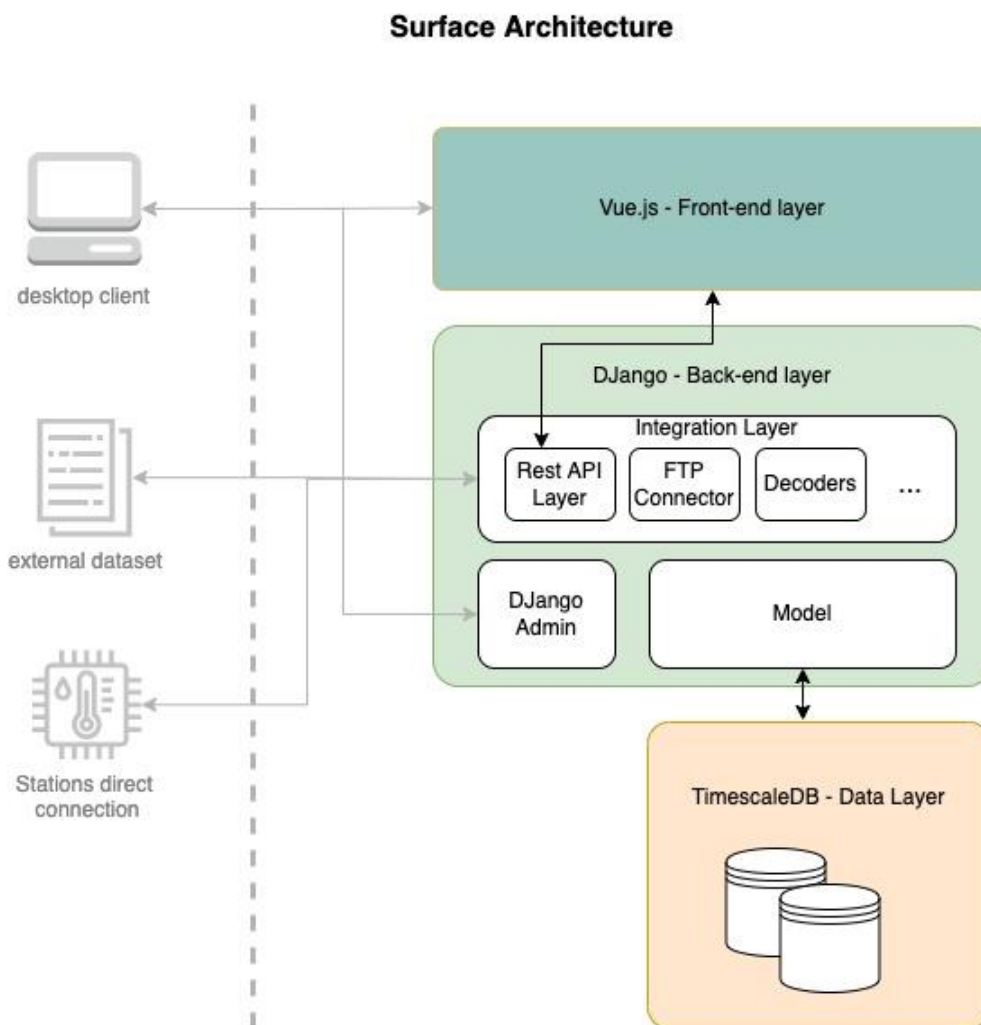


Figura 14. Diagrama arquitectónico de SURFACE

El objetivo de este capítulo es, en primer lugar, ilustrar los principios básicos del funcionamiento del sistema SURFACE y, a continuación, debatir los futuros desarrollos necesarios para adaptar la plataforma al contexto específico de la República Dominicana. Además, se presentará una sección opcional, pero que se considera importante, para garantizar que la plataforma esté a la última.

Cabe señalar que el equipo de desarrollo de SURFACE fue extremadamente útil a la hora de proporcionar aclaraciones y apoyo durante la fase de estudio y redacción del



documento, y se espera que esta fructífera colaboración continúe en el caso del futuro desarrollo de la plataforma CDMS para la República Dominicana.

## 5.1 INTRODUCCIÓN A DJANGO

La capa back-end de SURFACE, como se ha mencionado en los párrafos anteriores, está desarrollada en Python. Para ser más precisos, el back-end está gestionado por Django<sup>[37]</sup>, un popular framework web (Instagram y Spotify, por ejemplo, se basan en Django) basado en el lenguaje de programación Python. Django tiene una interfaz de administración integrada que permite la gestión de plantillas, usuarios y permisos de usuario/grupo.

Django es un framework web de código abierto escrito en Python, conocido por su robustez y escalabilidad, que permite desarrollar aplicaciones web complejas y escalables de forma eficiente. Sigue el modelo MVT (Modelo-Vista-Plantilla), una arquitectura de desarrollo que divide el código en tres componentes principales: el Modelo (Model), la Vista (View) y la Plantilla (Template).

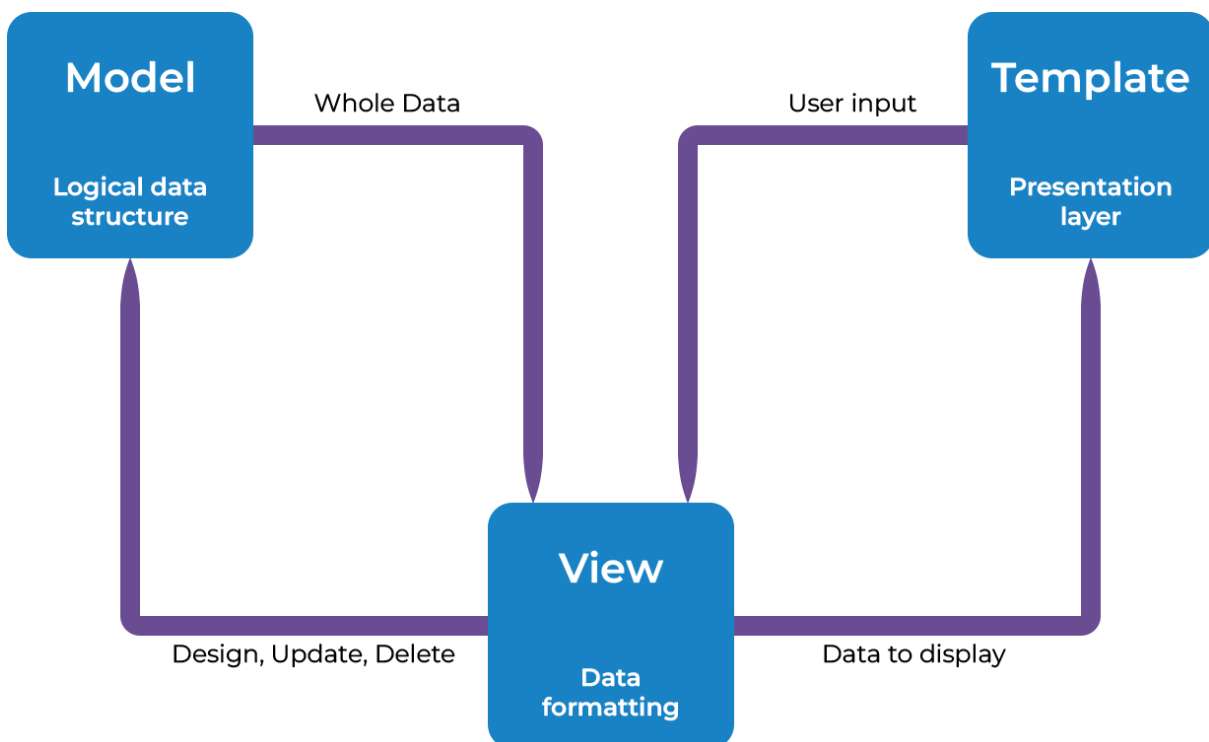


Figura 15. Patrón de diseño MVT (Model View Template)

---

<sup>37</sup> [Sitio web oficial](#)

El Modelo de Django es responsable de definir la estructura de datos de la base de datos, proporcionando una abstracción de los datos mediante la definición de campos, tipos de datos y relaciones entre las entidades del sistema. El ORM (Object-Relational Mapping) integrado simplifica la gestión de la base de datos, permitiendo manipular los datos como objetos Python.

La Vista de Django maneja la lógica de negocio de la aplicación, procesando las peticiones del usuario, interactuando con las plantillas para obtener los datos necesarios y pasándolos a las plantillas para su renderización en HTML. Las vistas de Django ofrecen una gran flexibilidad en el manejo de peticiones y respuestas, permitiendo personalizar el comportamiento de la aplicación según necesidades específicas.

La plantilla Django define la presentación de los datos en el front-end. Las plantillas permiten crear páginas HTML dinámicas, incluyendo datos dinámicos de las plantillas y manejando la lógica de presentación, como bucles y condiciones. Las plantillas de Django son muy flexibles y permiten separar la lógica de presentación de la lógica de negocio.

Además del modelo MVT, Django ofrece muchas funcionalidades integradas, como el sistema de enrutamiento para el procesamiento de URL, la gestión de autenticación de usuarios, la gestión de sesiones, la gestión de formularios y mucho más. Debido a su facilidad de uso, seguridad integrada y escalabilidad, Django es ampliamente utilizado para el desarrollo de aplicaciones web de diversos tamaños y complejidad.

## **5.2 DEFINICIÓN DE LA ESTRUCTURA DE LA BASE DE DATOS**

La estructura de la base de datos de SURFACE<sup>[38]</sup> puede dividirse conceptualmente en tres partes funcionalmente distintas. Estas tres partes están completamente separadas entre sí a nivel relacional, es decir, las tablas que componen una de las partes no están vinculadas por claves foráneas a ninguna tabla de ninguna de las otras dos partes. En cualquier caso, las tres partes se comunican entre sí, al estar vinculadas por relaciones conceptuales o funcionales (aunque a otros niveles). A nivel de la base de datos, es decir, a nivel puramente tabular y a nivel del código subyacente, las tres partes están claramente divididas. Al mismo tiempo, las relaciones entre tablas de una misma área funcional son densas y a veces intrincadas, sobre todo en lo que respecta a la parte de la base de datos que se ocupa del almacenamiento de datos hidrometeorológicos.

---

<sup>38</sup> El diagrama SURFACE ER está disponible [aquí](#). El modelo de datos SURFACE está disponible [aquí](#)

Dos de las tres partes pueden clasificarse como básicamente administrativas, aunque se ocupan de tareas bastante diferentes. La interfaz de estas partes administrativas se implementa a través de *Django Admin*, una aplicación por defecto incluida en el framework que proporciona una interfaz de administración lista para usar para el modelo de aplicación.

El panel de administración de Django proporciona funcionalidades como crear, editar, ver y borrar registros en la base de datos de la aplicación. Además, se puede personalizar fácilmente la interfaz de administración para satisfacer las necesidades específicas de su sitio, utilizando la API de personalización proporcionada por Django.

La implementación de la gestión de modelos dentro del framework Django hace innecesario implementar un programa separado de administración de bases de datos para todas las funciones avanzadas de bases de datos. Además de esta ventaja puramente técnica, una de las principales razones de su popularidad es su gran comunidad de usuarios y soporte. La combinación de estos dos factores hace que la probabilidad de que lo que quieras implementar en tu producto ya lo hayan hecho antes otros sea muy alta, pudiéndose insertar inmediatamente en el proyecto a través de un paquete de código abierto. Este es el caso, por ejemplo, del *programador de tareas Celery Beat*, que también está disponible como biblioteca de funciones para Django, y es de esta forma como se implementa en la base de datos y el back end de SURFACE.

## 5.2.1 PLANIFICADOR DE TAREAS

Esta sección de la base de datos (y el back-end, con el que se comunica) se encarga de gestionar la automatización de tareas, que se inician y ejecutan a intervalos regulares definidos por el administrador del sistema. Una tarea periódica se define como cualquier función destinada a ejecutarse automáticamente a intervalos regulares, o en determinados momentos del día o de la semana (o en cualquier periodo de tiempo). Todas las tablas de esta parte funcional de la BD llevan el prefijo *django\_celery\_beat*, que simplemente indica cómo se implementan las funciones de programación de tareas a través de la biblioteca Celery Beat para Django.

La programación de tareas periódicas utiliza la zona horaria UTC por defecto, pero para mayor comodidad puede cambiar la zona horaria por defecto para alinearla con la suya. Una de las pocas desventajas de utilizar Celery Beat como una biblioteca Django - es decir, como un programador de base de datos - es que si se cambia la zona horaria de la programación de tareas, la programación tiene que ser reiniciada manualmente para hacer efectivo el cambio.

Un ajuste que puede resultar especialmente útil para planificar tareas periódicas es el horario solar, que programa automáticamente las tareas a realizar en función de eventos

solares condicionados por la latitud y la longitud, como la salida y la puesta del sol, o el crepúsculo (ya sea civil, náutico o astronómico). Todos los eventos solares se calculan utilizando la hora UTC y, por lo tanto, no se ven afectados por la configuración de la zona horaria. El horario solar es lo suficientemente sofisticado como para calcular automáticamente la hora de realizar las tareas incluso en regiones polares, donde el sol puede no salir o ponerse todos los días. La única excepción es para la *tarde\_solar*, definida formalmente como el momento en que el sol atraviesa el meridiano celeste. Un evento planificado basado en el *solar\_noon* se ejecutará todos los días, incluso si el sol está por debajo del horizonte.

En resumen, el programador de tareas SURFACE es muy versátil y está bien integrado con la capa de backend y la BD. Su funcionalidad es especialmente interesante si se tiene en cuenta cómo trata la base de datos los datos hidrometeorológicos y climáticos. Desde este punto de vista, es crucial que el componente de automatización de tareas tenga la posibilidad de integrarse fuertemente con la naturaleza inextricablemente temporal (ligada al ciclo día/noche, así como al ciclo estacional) de una base de datos para almacenar y procesar datos hidrometeoclimáticos.

## **5.2.2 GESTIÓN DE SERVICIOS PÚBLICOS**

La gestión de usuarios de SURFACE también se realiza íntegramente mediante el framework Django, tanto en lo que se refiere a la capa back-end como al registro en la base de datos de todo lo necesario para el correcto funcionamiento del sistema. Django tiene un sistema de autenticación de usuarios que gestiona cuentas, grupos, permisos y sesiones basado en cookies. El sistema de autenticación de Django gestiona tanto la autenticación como la autorización: en resumen, la diferencia entre ambos conceptos es que la autenticación verifica que un usuario es quien dice ser, y la autorización determina lo que un usuario autenticado puede hacer. En términos más generales, el término autenticación se utiliza a menudo, también en este documento, para referirse a la combinación de autenticación y autorización.

El sistema básico de autenticación en Django está diseñado para ser muy genérico y, cuando se utiliza como independiente, no proporciona algunas de las funcionalidades más avanzadas que se encuentran comúnmente en los sistemas de autenticación web - funcionalidad que puede, sin embargo, ser implementada mediante el uso de paquetes de terceros. Como ya se mencionó en la sección anterior, Django tiene un sistema de autorización incorporado, que proporciona una manera de asignar permisos a los usuarios a través de grupos de permisos específicos. Estas autorizaciones se basan tanto en la naturaleza de los objetos en los que se permiten cambios o modificaciones como en el nivel de modificación permitido en cada uno de estos objetos. El funcionamiento de estas autorizaciones es bastante estándar y puede resumirse a grandes rasgos del siguiente modo:

- El acceso a la visualización de objetos está restringido a los usuarios con permisos de "visualización" o "edición" para ese tipo de objeto.
- El acceso para ver el formulario "añadir" y añadir un objeto a una lista (por ejemplo, añadir una estación meteorológica) está restringido a los usuarios con el permiso "añadir" para ese tipo de objeto.
- El acceso para ver la lista de edición, ver el formulario de "edición" y editar un objeto está restringido a los usuarios con permiso de "edición" para ese tipo de objeto.
- El acceso para eliminar un objeto está restringido a los usuarios con la autorización "eliminar" para ese tipo de objeto.

Así pues, el sistema de permisos no es especialmente sofisticado, al menos en su versión básica integrada de forma nativa en SURFACE<sup>[39]</sup>. Los grupos de usuarios son una forma sencilla y natural de facilitar la gestión de los permisos. De hecho, son una forma genérica de categorizar a los usuarios para poder aplicarles autorizaciones o cualquier otra etiqueta. No hay limitación en el número de grupos a los que puede pertenecer un usuario. Un usuario de un grupo tiene automáticamente los permisos concedidos a ese grupo. Por ejemplo, si el grupo Editores de estaciones tiene el permiso *can\_edit\_any\_station*, cualquier usuario de ese grupo tendrá permiso para editar cualquiera de las estaciones de la base de datos.

Además de las autorizaciones, los grupos son una forma cómoda de clasificar a los usuarios para asignarles una etiqueta o una funcionalidad ampliada. Por ejemplo, es posible crear un grupo de "usuarios especiales" y escribir un código que les permita acceder a una parte restringida del sitio (al estilo de "*sólo miembros*") o enviar mensajes de correo electrónico con la restricción de que los destinatarios de estos mensajes pertenezcan a un grupo específico - por ejemplo, una alerta meteorológica o un informe periódico sobre el clima, o más sencillamente un boletín clásico.

### 5.2.3 ALMACENAMIENTO DE DATOS CLIMÁTICOS

Esta es la sección de la base de datos que se ocupa de la parte decididamente más importante de un Sistema de Gestión de Datos Climáticos, es decir, el almacenamiento y la gestión propiamente dichos de los datos hidrometeorológicos recogidos sobre el terreno o a partir de estaciones automáticas (o de cualquier tipo de punto de recogida).

Las tablas de esta sección de la base de datos llevan, salvo en un par de casos, el prefijo *wx* (abreviatura de *meteorología*) con un guión bajo entre *wx* y el nombre de la tabla (ejemplos: *wx\_station*, *wx\_wmostationtype*, *wx\_dcpmessages*, etc.). Las tablas con el prefijo *wx* también existen en la sección de gestión de usuarios (en particular en la

---

<sup>39</sup> En la sección 5.6.1.1 se detalla la evolución futura a este respecto.

gestión de grupos y permisos): estas tablas no están relacionadas funcionalmente con el almacenamiento de datos hidrometeorológicos, pero el prefijo se inserta para indicar que el tipo de permisos está estrictamente relacionado con el dominio funcional de la base de datos, y las tablas se construyen ad hoc para ello.

Como ocurre con todas las bases de datos relacionales, la importancia de una tabla sobre otra suele adivinarse a partir de dos características.

El primero, que en realidad también se aplica a las bases de datos no relacionales, es el número de campos: es decir, el número de columnas en las que se pueden introducir datos. Asumiendo como premisa incuestionable que la base de datos se ha diseñado correctamente, la importancia de una tabla suele ser directamente proporcional al número de columnas que tiene, que sería entonces literalmente la cantidad de datos que contiene. Cuantos más datos contiene una tabla, más importante es, incluso intuitivamente.

La segunda, propia de las bases de datos relacionales, es el número de relaciones que tiene dicha tabla -en lenguaje técnico, el número de claves foráneas-, tanto entrantes como salientes.

Tanto por razones lógicas como de diseño de la base de datos, las tablas más importantes son -por orden de relevancia y complejidad- *wx\_station* y *wx\_variable*. Como puede adivinarse por el nombre, el primer cuadro trata de la definición de las estaciones meteorológicas y el segundo de la de las variables físicas asociadas a las mediciones instrumentales. La importancia de estos dos cuadros también se ve subrayada por el hecho de que la mayoría de las consultas de datos estándar de SURFACE se agrupan por estaciones o variables. Por otro lado, es bastante intuitivo que, si algún usuario quiere consultar datos hidrometeorológicos, querrá hacerlo a partir de una determinada localización geográfica (es decir, de una determinada estación) o en función de una variable -como, por ejemplo, consultar los valores máximos y mínimos de temperatura, o las precipitaciones-. Sin embargo, como la base de datos también está concebida para uso científico, las variables físicas que pueden introducirse en ella son de cualquier tipo y admiten cualquier instrumentación. Por cierto, incluso se está desarrollando una parte de la base de datos dedicada a los datos sobre olas y mareas, con la posibilidad de captar este tipo de datos a intervalos subminutales.

Otras tablas importantes, desde el punto de vista de la consulta, son *daily\_summary*, *monthly\_summary* y *last24\_summary*. Estas tres tablas, como también puede adivinarse fácilmente por sus nombres, se ocupan del almacenamiento de resúmenes diarios, mensuales y de las últimas veinticuatro horas, desglosados por estación y variable. Estos resúmenes se calculan automáticamente a intervalos regulares de tiempo, y son gestionados en parte por el programador de tareas de Celery Beat y en parte por las tablas *daily\_summarytask*, *monthly\_summarytask* y *last24\_summarytask*, que forman parte de la sección de la base de datos que se está analizando aquí. El cálculo automático de estos resúmenes es muy cómodo porque permite visualizarlos de forma

mucho más rápida y sencilla que una función similar implementada con cálculos a la carta, con la ventaja añadida de disponer de un historial de resúmenes disponible y archivado.

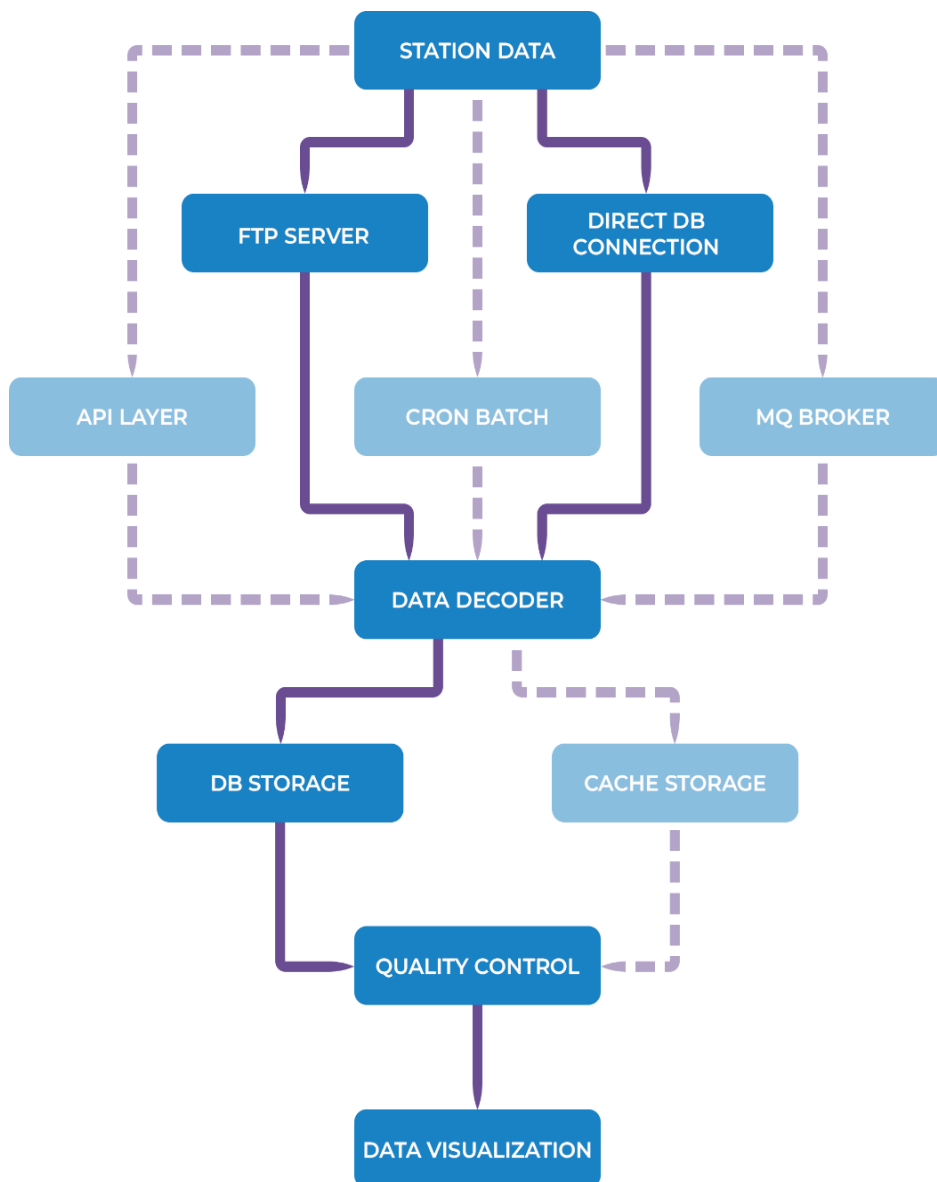
Sin entrar en demasiados detalles y sin llegar a describir demasiadas tablas, lo que no es el objetivo de este documento, hay que hacer una última mención a las tablas que representan los puntos de entrada en la base de datos de los datos que aún no se han procesado, es decir, que aún no han pasado por el control de calidad integrado en SURFACE. El almacenamiento de los datos previos y posteriores al control de calidad permite al sistema modificar los parámetros de control, en caso necesario, sin pérdida de datos. Esta característica no es estrictamente necesaria para el correcto funcionamiento de un Sistema de Gestión de Datos Climáticos, pero tenerla sin duda lo hace más versátil que no tenerla, además de proteger los datos de errores humanos al establecer los umbrales de control de calidad.

Para más detalles técnicos sobre la estructura de la base de datos SURFACE, se adjuntan tanto una descripción detallada tabla por tabla en forma de archivo .sql como el diagrama ER extrapolado del volcado de la base de datos.

## **5.3 DEFINICIÓN DE FLUJO DE DATOS**

El objetivo de esta sección del documento es ilustrar el flujo de datos estándar en el CDMS de SURFACE. Por flujo de datos entendemos su recorrido estándar desde la medición de los instrumentos en las estaciones hasta su consulta y uso por el usuario de la plataforma. Al no tratarse de un documento técnico, la descripción de este flujo se realizará de forma más bien discursiva, primando la claridad y proporcionando una visión general del funcionamiento de esta parte del sistema.

Por lo que respecta a la parte inicial del flujo de datos, es decir, el camino que va de la instrumentación al almacenamiento, hay que distinguir en primer lugar entre los dos componentes fundamentales del sistema SURFACE: el servidor de BD y los servidores FTP. La primera es una máquina utilizada como servidor donde se monta la base de datos de producción, con tareas de almacenamiento de datos y mantenimiento de copias de seguridad tanto de la base de datos como de los datos ya agregados. El segundo tipo de máquina también se utiliza como servidor, pero en este caso se trata de uno o varios servidores FTP, es decir, ordenadores que ejecutan un programa de servidor FTP para alojar y/o almacenar archivos. La máquina FTP también tiene un sistema automatizado de copias de seguridad, y en ambos casos estas copias se realizan a una hora determinada, una vez al día (normalmente a medianoche, hora local).



**Figura 16. Diagrama de flujo de datos en SURFACE y el futuro CDMS dominicano**

En las flechas azules continuas y no discontinuas se destaca esquemáticamente el flujo actual de datos, desde su recogida en las estaciones/plataformas hasta su visualización por el usuario final a través del visor web. En azul, con una opacidad del 50% y flechas de puntos, se destacan en cambio los posibles desarrollos futuros para el flujo de datos<sup>[40]</sup>.

El flujo de datos estándar es parcialmente diferente dependiendo de si los datos proceden de una estación satélite o no. Para las estaciones no satélite, se suele utilizar un servidor FTP como máquina intermedia entre las estaciones y la máquina donde está instalada la base de datos, mientras que las estaciones satélite pasan por alto la máquina intermedia y van directamente a almacenar los datos en el servidor de producción con la base de datos montada. Los datos transmitidos por las estaciones de satélite, tras una descodificación de entrada para hacerlos utilizables por la estructura de la base de datos, se almacenan inmediatamente en las tablas correspondientes. Sin embargo, la descodificación de la entrada es una característica compartida por ambos flujos, y es una función integrada en el sistema de gestión de datos.

A diferencia de las estaciones de satélite, los datos de las estaciones que no son de satélite se someten a pasos intermedios adicionales antes de registrarse en la base de

<sup>40</sup> Apartado 5.5.1



datos. De hecho, la ingesta de datos procedentes de estaciones no satelitales está diseñada para que el servidor o servidores FTP sean un paso intermedio. El servidor FTP actúa básicamente como el principal centro de almacenamiento y comunicación para las estaciones no satelitales, importando regularmente los datos de estas estaciones y guardando una copia de seguridad de estos datos en bruto. Las dos máquinas -DB y FTP- están diseñadas para comunicarse directa y automáticamente entre sí. Tras almacenarse en el servidor FTP, los datos no procedentes de satélites se transmiten al servidor de la base de datos para su tratamiento e inclusión en el sistema final para su consulta. El sistema está diseñado para gestionar eficazmente la entrada de datos, comparando los nuevos datos recibidos de las estaciones con los datos existentes en el servidor FTP y omitiendo automáticamente todos los datos que no se hayan actualizado desde la última entrada.

En los dos casos considerados, la introducción de datos se realiza a intervalos regulares y, una vez configurada correctamente, está totalmente automatizada. Una vez registrados los datos en la base, el sistema los procesa automáticamente rellenando las tablas de consulta, con lo que quedan disponibles tanto para su posterior análisis y tratamiento como para su utilización por el usuario final. A partir de este punto, es de hecho el usuario quien elige interactivamente cómo consultar los datos en función de la amplia funcionalidad del sistema y de sus propias necesidades.

### **5.3.1      DECODIFICADOR PARA LA IMPORTACIÓN DE DATOS**

Los datos que hay que almacenar en la base de datos suelen proceder de distintos tipos de estaciones, sean o no de satélite. Los distintos tipos de estaciones miden variables diferentes en función de la instrumentación con la que están equipadas y de la finalidad para la que están pensadas o diseñadas. Además, incluso las estaciones destinadas al mismo fin pueden no registrar las mismas variables, sobre todo si tienen una instrumentación diferente: pensemos, por ejemplo, en una pequeña estación automática en una zona remota en comparación con una gran estación con personal siempre in situ.

Para superar este problema, suele haber dos tipos de soluciones: una manual y otra automatizada, además de todas las formas híbridas apropiadas. La solución manual suele consistir en introducir manualmente los datos medidos en un archivo de formato y forma idénticos, que luego se introduce en el sistema de *ingestión de datos* CDMS. Como es fácil adivinar, este tipo de solución es más exigente en cuanto a los recursos humanos necesarios y mucho más lenta que un proceso automatizado, en el que las lecturas de las estaciones se transforman mediante procesos automatizados en archivos listos para el sistema de ingestión de datos.

Independientemente de cuál de las dos soluciones se adopte en la práctica, suele ser necesario algún tipo de descodificador, es decir, un algoritmo que transforme los datos de tal manera que puedan almacenarse en la base de datos. Un ejemplo sencillo de este tipo de proceso es un script que transforma un archivo .csv(valoresseparados por comas) con los valores de las lecturas en su interior en un comando que los registra realmente en la base de datos. SURFACE ya está equipada de forma nativa con seis tipos diferentes de descodificadores, y una de las grandes ventajas de la plataforma es que se pueden escribir y registrar fácilmente descodificadores adicionales para facilitar la automatización del registro de lecturas hidrometeorológicas. No hay que subestimar la importancia de esta función, ya que facilita la ampliación de los procesos de ingestión de datos mediante la automatización del mayor número posible de procesos.

### **5.3.2 CONTROL DE CALIDAD**

Independientemente del tipo de estación y, por consiguiente, del camino que sigan los datos antes de ser almacenados, todos los datos están sujetos a un control de calidad por defecto, que debe ajustarse manualmente. Esto significa que, por defecto, no se establecen valores de ningún tipo para controlar la calidad de los datos, que se almacenan en el sistema tal y como proceden de las lecturas instrumentales. Al mismo tiempo, el control de calidad de los datos está automatizado: si se establecen los valores, se realiza automáticamente en cada ingestión de datos, lo que da lugar a la notificación y exclusión de datos sospechosos. No obstante, hay que recordar que el objetivo del control de calidad no es sólo excluir los datos con valores sospechosos, sino sobre todo informar de ellos a los responsables para que investiguen las razones de estas mediciones y, posiblemente, solucionen el problema instrumental (que suele ser la causa predominante de las mediciones extrañas).

El hecho de que los valores del control de calidad deban ajustarse manualmente hace que el proceso sea algo menos fluido que con un control de calidad preestablecido. Al mismo tiempo, esta característica permite afinar el control estación por estación y variable por variable, y modificarlo en cualquier momento según las necesidades. De hecho, es posible establecer valores umbral en función de cada variable de una estación específica o, de forma más general, en función de una variable independientemente de la estación en la que se mida. Jerárquicamente, los valores umbral de las estaciones ganan a los de las variables: si se fijan ambos, son los de las estaciones los que se utilizan cuando se aplica el control de calidad.

Hay tres tipos de umbrales disponibles: *rango*, *paso* y *persistencia*. Tienen finalidades y características diferentes, que describiremos brevemente en las siguientes secciones de este documento.

### **5.3.2.1 UMBRAL DE RANGO**

El objetivo principal del umbral de rango es comprobar si los datos presentan valores atípicos extraños, es decir, mediciones de valores extremos -en una u otra dirección- que son difíciles de explicar como procedentes de fenómenos meteorológicos o tendencias climáticas normales. La forma de realizar este control es extremadamente sencilla: se introduce manualmente un rango de valores aceptables, es decir, entre un máximo y un mínimo. Si los valores medidos por la instrumentación están fuera de este rango, la medición se notifica como sospechosa.

Tomemos como ejemplo el sensor de una estación que mide la temperatura del aire. Este sensor, por razones de clima local, tiene un rango establecido entre 0 °C y 50 °C. Si el sensor mide una temperatura del aire de -5 °C, el sistema lo notifica inmediata y automáticamente como dato sospechoso y lo excluye del procesamiento hasta que se notifique manualmente como válido.

### **5.3.2.2 UMBRAL DE PASO**

A diferencia del umbral de rango, el umbral de paso no se centra en la detección de valores atípicos en las mediciones, sino que adopta el enfoque de comprobar que los valores medidos por la instrumentación no están demasiado alejados (o incluso demasiado cerca) entre una medición y la siguiente. El funcionamiento práctico es muy sencillo y directo: se introducen manualmente dos valores, uno para el paso mínimo y otro para el paso máximo. El valor mínimo del paso garantiza que dos mediciones consecutivas no estén demasiado próximas, es decir, que la diferencia en valor absoluto entre las dos mediciones no sea inferior al umbral establecido. El valor de paso máximo funciona de forma similar, pero en su lugar se asegura de que dos mediciones consecutivas no estén demasiado separadas, es decir, que la diferencia en valor absoluto entre las dos mediciones no supere el umbral establecido. Si se da cualquiera de estos casos, la medición se notifica como sospechosa. Ambos tipos de umbral de paso pueden ser muy útiles, pero en general, el valor del paso máximo es más importante, al menos en lo que respecta a la mayoría de las variables.

Un ejemplo rápido de aplicación del umbral por pasos puede ser un sensor que detecte la temperatura del agua, con el umbral mínimo fijado en 0 °C y el umbral máximo fijado en 1 °C. En este caso, si no se detecta ninguna diferencia de temperatura entre una medición y la siguiente, todo va bien para el sistema (un umbral mínimo de 0 equivale a un umbral por defecto que no se ajusta y, por tanto, nunca detectará errores o problemas). Al mismo tiempo, si hubiera una diferencia incluso infinitesimalmente superior a 1 °C entre una medición y la siguiente, el sistema la detectaría como sospechosa e informaría de ello inmediatamente.

### **5.3.2.3 UMBRAL DE PERSISTENCIA**

El último umbral del sistema de control de calidad de SURFACE es el umbral de persistencia. De los umbrales analizados hasta ahora, es sin duda el que más se centra en detectar si el equipo funciona mal o no. El concepto que subyace a este umbral es, de hecho, evaluar si el equipo es potencialmente defectuoso o funciona mal. El planteamiento con el que se aplica es mediante el establecimiento de tres parámetros fundamentales, a saber, el intervalo de tiempo, la ventana temporal y la varianza mínima. Intervalo y ventana son conceptos bastante similares: la ventana es el "intervalo" de tiempo máximo en el que puede realizarse el análisis, mientras que el intervalo es la medida de tiempo de las "subventanas" en las que se divide la ventana principal cuando se realiza el análisis propiamente dicho. Básicamente, estos dos parámetros sirven para determinar la amplitud del intervalo de tiempo sobre el que se realiza el análisis.

La varianza mínima, en cambio, es muy simple y literal: si la varianza entre intervalos dentro de la misma ventana está por debajo de este valor, los datos se marcan automáticamente como sospechosos.

Un ejemplo de la aplicación de este umbral es igualmente sencillo y directo. Un sensor que detecta la temperatura del aire tiene un umbral fijado en una ventana de un día dividida en intervalos de seis horas cada uno, con una varianza mínima de 0,1. Si la varianza de los valores de los sensores en estos intervalos no es superior a 0,1, los datos no superan el control de calidad. En el caso de valores que deben variar regularmente a lo largo de un periodo de tiempo -como, por ejemplo, la temperatura del aire a lo largo de un día-, un informe de fallo de control de calidad suele equivaler directamente a un fallo del sensor o a un error humano en la introducción de datos.

## **5.4 LAYER API**

En el desarrollo de software, la capa API (*Application Programming Interface*) representa un conjunto de interfaces de programación y protocolos que permiten la comunicación y el intercambio de datos entre distintos componentes o sistemas de software, entre los que la capa API actúa como intermediaria, permitiéndoles comunicarse de forma segura y fiable. En la práctica, esta capa puede considerarse un puente entre la lógica empresarial del software y su implementación técnica. Las interfaces de programación expuestas por la capa API definen los métodos y parámetros que pueden utilizar los distintos componentes de software para interactuar con el sistema, lo que ofrece mayor modularidad y escalabilidad. Gracias a estas características, el back-end y el front-end pueden separarse, haciendo que la aplicación sea menos monolítica y simplificando la gestión de errores y el mantenimiento.

La capa API puede implementarse utilizando diversas tecnologías, como REST (*Representational State Transfer*), GraphQL, SOAP(*Simple Object Access Protocol*) u otras. En cualquier caso, el objetivo de la capa API es simplificar la comunicación entre componentes de software, reducir la complejidad del sistema y mejorar la interoperabilidad entre servicios, así como la mantenibilidad del software.

Es importante señalar que SURFACE ya cuenta con una capa API implementada a través de la tecnología REST, que se ha documentado aún más a través de Swagger para facilitar la referencia.

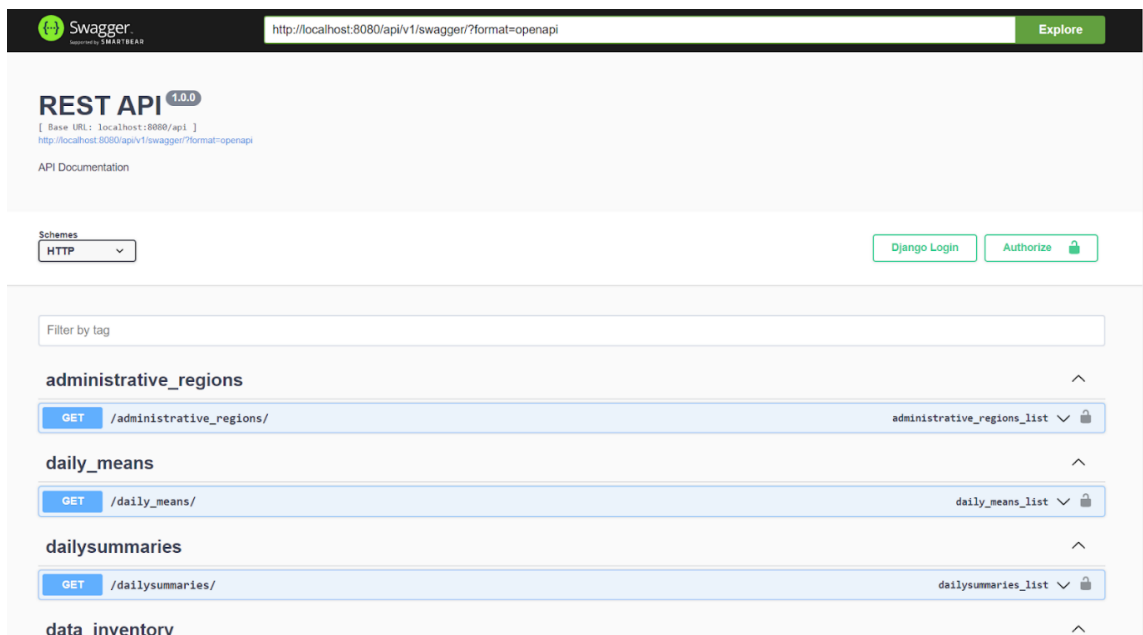


Figura 17. Ejemplo de documentación de la API SURFACE

Swagger es una herramienta de código abierto que permite a los desarrolladores crear documentación interactiva para sus API de forma automatizada y visualmente atractiva. En otras palabras, Swagger proporciona a los desarrolladores una herramienta para describir la funcionalidad y las características de sus API de forma clara y detallada. Además, Swagger sigue la especificación OpenAPI<sup>[41]</sup>, que es el principal estándar para este tipo de documentación. Debido a su popularidad, muchas herramientas y marcos de desarrollo de API han integrado la compatibilidad con la especificación OpenAPI, lo que simplifica la creación y gestión de la documentación.

Para ver la documentación de Swagger, basta con abrir una página del navegador en <https://editor.swagger.io/> y copiar el archivo YAML<sup>[42]</sup> en la parte izquierda del sitio. A continuación, la documentación de la API se generará automáticamente en la parte

<sup>41</sup> <https://spec.openapis.org/oas/v3.1.0>

<sup>42</sup> Disponible [aquí](#)

derecha del sitio, mostrando todos los métodos expuestos listados con la URL que los identifica, así como el método HTTP utilizado para llegar a ellos.

Method	URL	Action
<b>stations</b>		
GET	/stations/	stations_list
POST	/stations/	stations_create
PUT	/stations/	stations_update
DELETE	/stations/{id}/	stations_delete
<b>stations_metadata</b>		
GET	/stations_metadata/	stations_metadata_list
POST	/stations_metadata/	stations_metadata_create
GET	/stations_metadata/{id}/	stations_metadata_read
PUT	/stations_metadata/{id}/	stations_metadata_update
PATCH	/stations_metadata/{id}/	stations_metadata_partial_update
DELETE	/stations_metadata/{id}/	stations_metadata_delete
<b>stations_variables</b>		
GET	/stations_variables/	stations_variables_list
GET	/stations_variables/stations/	stations_variables_stations_list
<b>step_threshold</b>		
GET	/step_threshold/	step_threshold_list
POST	/step_threshold/	step_threshold_create
PATCH	/step_threshold/	step_threshold_partial_update
DELETE	/step_threshold/	step_threshold_delete

Figura 18. Ejemplo de documentación de la API SURFACE

En concreto, si seleccionamos uno de los métodos, se mostrará:

- La descripción del método;
- Los parámetros necesarios;
- El modelo de la respuesta que se devolverá.

En el caso mostrado en la **Figura 19**, se documenta el método que enumera todas las estaciones. Este método requiere un parámetro llamado "página" (la lista de todas las estaciones se divide en páginas, y cada página se caracteriza por un número fijo de estaciones) y, en respuesta, muestra el modelo de datos de respuesta, es decir, cómo se mostrarán los datos de la estación si la solicitud tiene éxito.

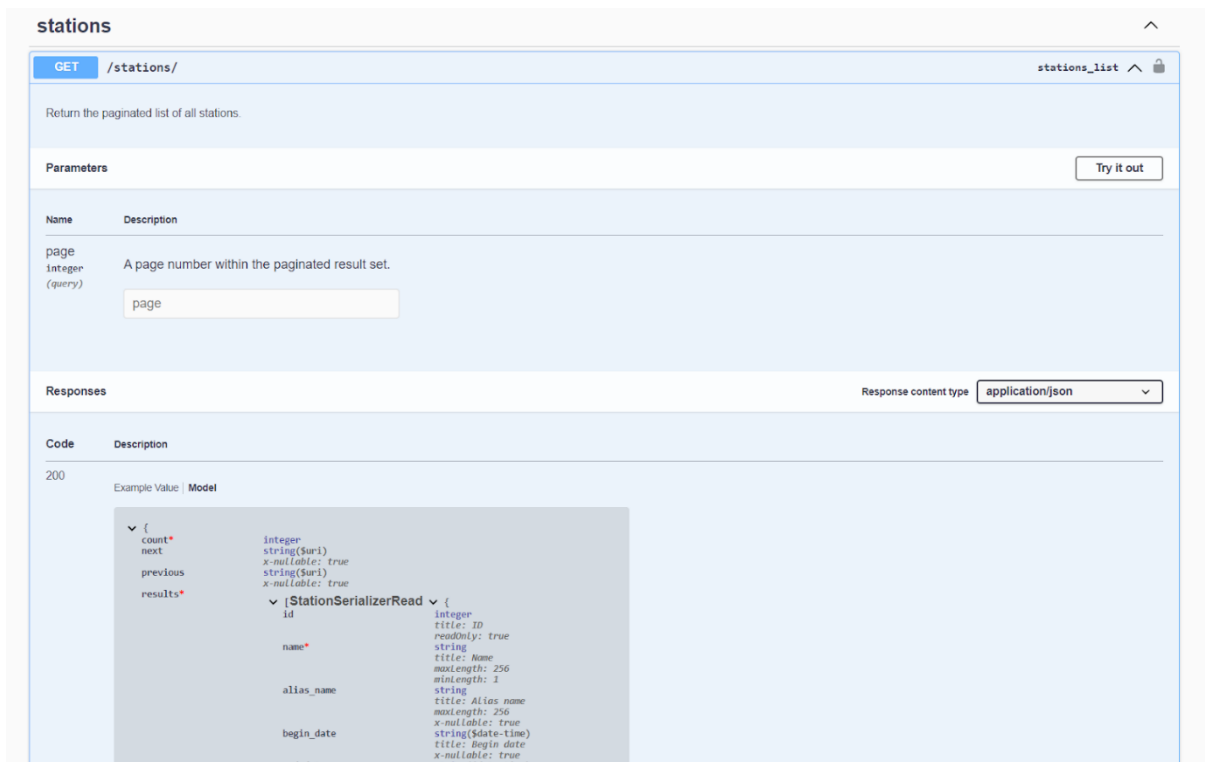


Figura 19. Detalles de la documentación del método de *las estaciones*

Del análisis del Swagger completo se desprende que la mayor parte de las funcionalidades requeridas por la plataforma ya han sido implementadas vía API. Sin embargo, hay algunas características que aparecen en los diagramas de casos de uso que aún no se han desarrollado, entre ellas:

- Gestión de accesos directos y filtros guardados;
- Importación de datos hidrometeorológicos;
- Gestión de Institutos (inserción, modificación, supresión).

La interacción con el mapa y el análisis espacial no se mencionaron, ya que requieren un tratamiento gráfico superior al nivel de la API.

## 5.5 OBTENCIÓN DE DATOS DE FUENTES EXTERNAS

La integración de datos procedentes de diversos institutos y plataformas de encuesta en una única base de datos es una cuestión compleja y estratégica que requiere la definición de soluciones eficaces para agregar, procesar y gestionar grandes cantidades de datos procedentes de fuentes heterogéneas.

El tema se enmarca en un contexto en el que las instituciones que operan en el territorio dominicano se encuentran a menudo utilizando diferentes plataformas, aplicaciones y sistemas de recogida y tratamiento de datos. Los escenarios que contempla el proyecto implican tanto el uso de sistemas ya implantados por la plataforma SURFACE como el desarrollo de nuevas soluciones para atender necesidades específicas o escenarios particulares.

## **5.5.1 OPCIONES PARA LA RECOGIDA DE DATOS**

Las *mejores prácticas* son un conjunto de normas, procedimientos y enfoques recomendados para alcanzar objetivos específicos. En el contexto de la recopilación de datos, las mejores prácticas se refieren a las metodologías recomendadas para adquirir datos de forma eficaz y eficiente, garantizando la calidad, fiabilidad y seguridad de la información recopilada.

En cuanto a la recopilación de datos, es aconsejable adoptar una estrategia de centralización de la información en la futura plataforma CDMS de la República Dominicana. Esto significa recopilar datos de distintas fuentes y almacenarlos en el sistema central. Esto puede hacerse almacenando directamente los datos recogidos por las plataformas de encuestas o consultando las bases de datos de terceros existentes y copiando y almacenando la información en la plataforma central. Este enfoque permite un control más preciso de los datos recogidos, garantiza la coherencia de la información y simplifica la gestión y el mantenimiento tanto de la información como del sistema de información.

También es importante considerar la consulta de bases de datos de terceros sin almacenar los datos permanentemente como opción de acceso a los datos, aunque no se recomienda. Un sistema de este tipo implicaría el uso de una capa de caché para reducir la necesidad de almacenar datos sensibles de forma permanente. Hacerlo podría ayudar a reducir los riesgos que conlleva el manejo de datos sensibles, como la violación de la privacidad o la pérdida de datos, y facilitar parcialmente el proceso de gobernanza. Sin embargo, este enfoque sería considerablemente más complejo de aplicar desde el punto de vista informático y, en algunos escenarios, podría dar lugar a un acceso más lento a los datos, ya que requeriría más pasos para acceder a la información almacenada en otro lugar<sup>[43]</sup>.

Durante el desarrollo de la plataforma, será importante elegir el escenario más adecuado en función de la complejidad y las características del sistema informático del instituto

---

<sup>43</sup> Apartado 5.6.2.1



que se vaya a integrar y de las peticiones específicas de los institutos que prefieran no almacenar sus datos en sistemas que no estén bajo su completo control.

A continuación se ofrecerá una visión general de las opciones de adquisición de datos previstas para el futuro sistema CDMS en la República Dominicana. De este modo se ofrecerá una visión general de los principales escenarios previstos por el proyecto para el proceso de transformación de los datos desde su recogida hasta su disponibilidad para la consulta.

### **5.5.1.1 OPCIÓN A: SERVIDOR FTP**

Como ya se ha descrito en la sección **5.3**, los servidores FTP del CDMS SURFACE ya están configurados para desempeñar un papel importante en el flujo de datos estándar. Actualmente, este sistema implica el uso de un servidor FTP como máquina intermedia para los datos procedentes de estaciones no satelitales. Estos datos se importan periódicamente al servidor FTP y se guardan como copias de seguridad sin procesar. El servidor FTP y el servidor de base de datos están diseñados para comunicarse directamente entre sí de forma automática. Tras almacenarse en el servidor FTP, los datos no procedentes de satélites se transmiten al servidor de la base de datos para su tratamiento e inclusión en el sistema final para su consulta. Una vez registrados en la base de datos, los datos se procesan automáticamente y rellenan las tablas de consulta, quedando disponibles para su posterior análisis y uso por parte del usuario final. La introducción de datos está automatizada y se realiza a intervalos regulares, lo que garantiza un flujo de datos eficaz y fiable.

### **5.5.1.2 OPCIÓN B: CAPA API**

La capa API es una ayuda fundamental en la integración de datos dentro de un sistema informático. Mediante el uso de los métodos proporcionados por la capa API, los datos pueden introducirse en el sistema de forma automática o manual. Esta flexibilidad está garantizada por la presencia de un software que permite la conexión a la API REST, lo que posibilita una comunicación estandarizada y fiable con el sistema.

Sin embargo, para poder utilizar esta funcionalidad, los datos deben presentarse en un formato compatible con el sistema. Por esta razón, en lo que respecta a la entrada de datos hidrometeorológicos, será necesario implementar decodificadores<sup>[44]</sup> capaces de

---

<sup>44</sup> Algunos ya existen, como se ve en el apartado **5.3.1**

transformar y procesar los datos desde el formato de entrada al requerido por el modelo de datos necesario para su almacenamiento en la base de datos.

El uso de la capa API para la integración de datos permite así simplificar enormemente el proceso de introducción de datos en el sistema, al tiempo que garantiza la flexibilidad necesaria para adaptarse a las necesidades específicas de cada instituto.

### **5.5.1.3 OPCIÓN C: CONEXIÓN DIRECTA A LAS ESTACIONES**

El software SURFACE ya proporciona una conexión directa entre la base de datos y las estaciones o plataformas topográficas, sin centros intermedios como el servidor FTP. Esta dinámica se aplica actualmente en el flujo de datos de las estaciones automáticas por satélite, cuyos datos recogidos se envían directamente al sistema de ingesta de datos de la base, que los descodifica y los inserta en las tablas correspondientes para que estén listos para su consulta a intervalos regulares<sup>[45]</sup>.

El objetivo del proyecto es ampliar este sistema para incluir cualquier tipo de estación automática y utilizarlo como columna vertebral para la densificación de la red de plataformas de detección, como se menciona entre los objetivos del proyecto<sup>[46]</sup>. Una parte importante de esta estrategia de expansión es la planificación de un sistema de estaciones automáticas de bajo coste que puedan cubrir las lagunas existentes en el territorio dominicano.

Aunque aún están por definir los detalles exactos de cómo se realizarán y aplicarán estas estaciones, es importante prever un sistema de software adecuado para gestionar y almacenar los datos de estas estaciones. Este sistema deberá ser capaz de manejar un gran volumen de datos procedentes de distintas fuentes, garantizando su integridad, seguridad y disponibilidad para su posterior análisis y consulta.

El objetivo es crear una infraestructura robusta y ampliable que pueda soportar la expansión de la red de estaciones automáticas, permitiendo recoger datos de diversas fuentes sin obstáculos. Habrá que tener en cuenta aspectos como la normalización de los datos procedentes de distintas estaciones, la gestión de errores e interrupciones de conexión, la sincronización de datos y la planificación de las operaciones de ingestión de datos para garantizar que éstos estén listos para su consulta y análisis.

Además, la planificación del sistema de estaciones automáticas de bajo coste también tendrá que considerar la disponibilidad de recursos de hardware y conectividad en las diferentes zonas del territorio dominicano, con el fin de garantizar que las estaciones puedan enviar datos a la base de datos de forma fiable y continua, evaluando las

---

<sup>45</sup> Véase también el apartado **5.3**

<sup>46</sup> Sección **1.1**

opciones tecnológicas y de infraestructura más adecuadas, de forma que se garantice la mayor fiabilidad posible de todo el sistema.

#### **5.5.1.4 OPCIÓN E: INTRODUCCIÓN MANUAL DE DATOS**

No hay mucho que decir sobre esta opción, que generalmente se desaconseja debido a la imposibilidad inherente de tener la misma fiabilidad y velocidad que un sistema automatizado de ingestión de datos a intervalos regulares.

En pocas palabras, un operador introduce manualmente los datos en un archivo con un formato determinado, como una hoja de cálculo con un formato especial o un archivo .csv. Los archivos compilados a mano se introducen en el sistema de ingesta de datos que, tras interpretarlos mediante un descodificador especial, los inserta en la base de datos donde, a partir de ese momento, se tratan exactamente igual que el resto de los datos.

#### **5.5.1.5 OPCIÓN F: BROKER MQ**

El término broker MQ hace referencia a un software especializado en la gestión de mensajes y transacciones en un entorno distribuido. En concreto, un broker MQ (Message Queue) actúa como intermediario entre las distintas aplicaciones que se comunican entre sí mediante la transmisión de mensajes.

El broker MQ proporciona una infraestructura de mensajería fiable, escalable y segura que permite a las aplicaciones comunicarse entre sí de forma asíncrona, es decir, sin tener que esperar una respuesta inmediata de la otra parte. El broker recibe los mensajes enviados por las aplicaciones, los almacena y los transmite a los destinatarios según reglas definidas, garantizando la entrega de los mensajes incluso en caso de fallos temporales de la aplicación o de la red.

Este escenario podría hacer posible que las estaciones envíen datos hidrometeorológicos directamente al broker MQ, con la plataforma "consumiendo" estos mensajes tan pronto como estén disponibles y almacenándolos en la base de datos. Además de la escalabilidad, esta elección podría mejorar la disponibilidad de la plataforma, ya que la recepción y el tratamiento de los datos tendrían lugar de forma asíncrona y, por tanto, sólo cuando el sistema dispusiera de los recursos necesarios.

## **5.6 EVOLUCIONES Y DESARROLLOS FUTUROS**

Esta sección del documento representa el resultado del análisis destinado a identificar las partes de SURFACE que más debían mejorarse y, en consecuencia, a establecer prioridades de aplicación. A partir de este análisis, se definieron dos niveles de prioridades de desarrollo: mejoras obligatorias y mejoras opcionales.

Las mejoras obligatorias se consideraron esenciales y no negociables para garantizar el buen funcionamiento de la plataforma. En cambio, las mejoras opcionales se han dividido en módulos autónomos, que pueden desarrollarse e implantarse por separado sin interferir con otros aspectos de la plataforma.

### **5.6.1 DESARROLLOS OBLIGATORIOS**

#### **5.6.1.1 AUTORIZACIONES Y AISLAMIENTO DE DATOS**

La primera evolución obligatoria de SURFACE a la futura plataforma CDMS dominicana es, sin duda, el desarrollo y la implantación de un sistema de aislamiento de datos eficaz y pertinente.

Como se identifica en varias partes del documento<sup>[47]</sup>, empezando por la sección de requisitos funcionales<sup>[48]</sup>, SURFACE carece por completo de la gestión de aislamiento de datos - una característica que es absolutamente necesaria implementar en el contexto dominicano, donde la propiedad de las plataformas y sistemas de detección se divide entre muchas entidades productoras diferentes. Por tanto, el escenario está bien perfilado: es importante que cada entidad sólo pueda acceder con permisos de edición a sus propias estaciones y no a las de otras entidades, mientras que conserva los permisos de visualización de los datos de todas las demás estaciones.

Como ya se ha mencionado, el aislamiento de datos puede aplicarse mediante diversos mecanismos. La forma de proceder en este caso es explotar el modelo de datos SURFACE existente, con la adición de tablas especiales para soportar una estructura de aislamiento desde el nivel de la base de datos. Básicamente, se enriquecerá el modelo de datos existente para que refleje una situación en la que las plataformas de detección puedan dividirse en función de la entidad encargada de ellas. Una vez desarrollados estos cambios, será sencillo implantar el consiguiente sistema de autorización adicional

---

<sup>47</sup> Por ejemplo, en la sección **4.1.2.10**

<sup>48</sup> Capítulo **3**

que permita a las entidades individuales gestionar sus datos sin generar conflictos y manteniendo el nivel de seguridad e integridad de los datos en el nivel más alto posible.

### **5.6.1.2 FUNCIONALIDAD DE IMPORTACIÓN A TRAVÉS DE LA CAPA REST API**

Una de las características que faltan en la actual capa API es la posibilidad de importar datos hidrometeorológicos. La idea es proporcionar un punto final accesible externamente que pueda ser utilizado por usuarios o máquinas autenticados para introducir datos no sólo a través de la plataforma web. Esta elección abre varias posibilidades de introducción de datos, tanto manual como automática, gracias a la capacidad de la API para comunicarse con otros componentes de software, independientemente del lenguaje de programación utilizado para implementarlos. La conexión y el envío de datos, tal y como se ha descrito anteriormente, tienen lugar según los principios de la arquitectura REST.

La importación de datos es un punto crucial para la plataforma, ya que no sólo permite introducir datos recién recopilados, sino que también ofrece la oportunidad de introducir datos históricos que posean las entidades productoras. Estos datos históricos desempeñan un papel importante en la elaboración de estadísticas y el análisis de datos hidrometeorológicos. Pueden utilizarse para identificar tendencias a largo plazo, comparar los datos actuales con los históricos e identificar cualquier cambio significativo en el comportamiento del clima o la hidrología a lo largo del tiempo.

La API de importación de datos hidrometeorológicos permitirá a los usuarios o máquinas autenticadas enviar datos en diferentes formatos, como .csv, .json o .xml, en función de las necesidades de la aplicación. Estos datos pueden enviarse de forma sincrónica o asincrónica, en función de la complejidad de los datos y del rendimiento requerido por la plataforma. También será posible definir restricciones y comprobaciones de validación de datos durante la importación para garantizar la integridad y coherencia de los datos introducidos.

Con la API de importación de datos hidrometeorológicos, los usuarios podrán automatizar el proceso de introducción de datos, por ejemplo mediante la vinculación de sistemas automáticos de adquisición de datos o la integración con otras fuentes de datos externas. Esto mejorará la eficacia del proceso de adquisición de datos y garantizará la actualización puntual de la información en la plataforma.

La implementación de una API de importación de datos hidrometeorológicos es, por tanto, un importante añadido a la actual capa de API, que permite una mayor flexibilidad en la introducción de datos, la automatización del proceso de adquisición de datos y el enriquecimiento de la base de datos histórica de la plataforma. Esto permitirá a los

usuarios utilizar la API de forma más versátil, permitiendo importar datos de distintas fuentes y en diferentes formatos, y garantizando un flujo de datos continuo y fiable dentro de la plataforma.

## **5.6.2 EVOLUCIONES OPCIONALES Y INTEGRACIÓN DE TECNOLOGÍAS CLAVE**

Tras tratar el tema ineludible de las evoluciones obligatorias de las plataformas, esta sección del documento se centrará en las evoluciones opcionales. El desarrollo de algunas de estas funcionalidades, como se verá en las secciones dedicadas a ellas, sigue siendo muy recomendable.

### **5.6.2.1 CONSULTAS A BASES DE DATOS DE TERCEROS**

En el contexto de la evolución de la plataforma opcional, se está estudiando la introducción de la posibilidad de consultar bases de datos de terceros, además de la operación estándar con consulta directa al sistema centralizado. Sin embargo, es importante señalar que, según los requisitos recopilados<sup>[49]</sup>, no es posible considerar una solución que incluya únicamente una capa de consulta sin almacenamiento dedicado. Esto se debe a que muchas entidades productoras pueden no disponer de sistemas informáticos propios para la gestión de datos.

Como posible solución, se contempla una opción híbrida, aunque no recomendable, para el caso de que las entidades productoras dispongan de un sistema informático adecuado pero no deseen compartir toda la información. En ese caso, podría buscarse un compromiso entre la necesidad de acceder a la información de terceros y la protección de la información sensible.

Es importante destacar que una solución híbrida puede conllevar una mayor complejidad en la gestión de los datos y puede no ser la opción ideal para garantizar un acceso completo y sin restricciones a la información requerida. Sin embargo, en algunos casos, podría considerarse una opción si la entidad productora dispone de un sistema informático adecuado pero tiene restricciones para compartir datos.

No obstante, la intención del proyecto sigue siendo continuar trabajando con las entidades productoras y evaluar caso por caso las mejores soluciones para garantizar un

---

<sup>49</sup> Capítulos 3 y 4

acceso eficaz y seguro a la información requerida, respetando al mismo tiempo las políticas y restricciones de cada entidad.

Aún no se han definido en detalle los métodos concretos mediante los que se implementaría la capa de consulta de bases de datos de terceros, pero la idea básica es implementar una capa de caché, un tipo de tecnología que se describirá en la siguiente sección.

### **5.6.2.1.1 CAPA DE CACHÉ**

Las capas de caché son una tecnología basada en el almacenamiento temporal (caché) de datos de uso frecuente en una memoria rápida y de alto acceso, como la RAM. Estos datos se guardan para poder recuperarlos rápidamente en el futuro, sin tener que acceder constantemente a dispositivos de almacenamiento más lentos, como discos duros o bases de datos.

El almacenamiento en caché se utiliza desde hace tiempo como tecnología para acelerar el rendimiento de las aplicaciones, reducir el tiempo de latencia y mejorar la eficiencia. Con soluciones tecnológicas más modernas, las capas de caché también pueden utilizarse en diferentes contextos, como en sitios web para almacenar contenido estático, a menudo imágenes o archivos CSS, o en aplicaciones empresariales para mejorar el rendimiento de las consultas a bases de datos, que es la aplicación que nos ocupa.

Sin embargo, configurar y gestionar un sistema de caché puede ser complejo y requerir conocimientos avanzados de infraestructura de red, gestión de memoria y tecnologías de caché. Por este motivo, se han desarrollado varios programas especializados en caché, como Apache Ignite<sup>[50]</sup> o Redis<sup>[51]</sup>, para simplificar la creación y gestión de capas de caché.

Apache Ignite es una plataforma de computación distribuida y almacenamiento en caché en memoria que ofrece una amplia gama de funcionalidades, como el almacenamiento en caché de datos, el procesamiento distribuido, el análisis en tiempo real y el aprendizaje automático. Ignite está diseñado para manejar grandes cantidades de datos y ofrecer un rendimiento de alta velocidad almacenando en memoria los datos de uso frecuente. Una de las características distintivas de Ignite es su capacidad para distribuir datos de forma transparente entre varios nodos de un clúster, lo que permite a las aplicaciones acceder a los datos de forma rápida y escalable.

---

<sup>50</sup> <https://ignite.apache.org/>

<sup>51</sup> <https://redis.io/>

Redis, por su parte, es un sistema de caché en memoria de código abierto, flexible y de alto rendimiento que ofrece una amplia gama de estructuras de datos, como cadenas, listas, conjuntos, mapas y más. Redis es conocido por su velocidad de acceso a la memoria y su capacidad para manejar grandes volúmenes de datos en tiempo real. Redis también admite funciones avanzadas como la persistencia en disco, la replicación de datos y la escalabilidad horizontal, lo que lo convierte en una opción popular para muchas aplicaciones que requieren capas de caché de alto rendimiento.

Tanto Apache Ignite como Redis son soluciones de almacenamiento en caché en memoria que ofrecen un alto rendimiento y funciones avanzadas para mejorar el rendimiento de las aplicaciones. Mientras Ignite es una completa plataforma de computación distribuida que ofrece una amplia gama de funcionalidades, Redis es un sistema de caché en memoria flexible y de alto rendimiento. Ambos programas se utilizan en diversos contextos de aplicación para acelerar el rendimiento de las aplicaciones y mejorar la eficacia de la gestión de datos.

## **5.6.2.2 INVESTIGACIÓN Y DESARROLLO**

Para mantenerse al día con las nuevas tecnologías y tendencias, es fundamental dedicar tiempo y recursos a la innovación y experimentación. Destinar un presupuesto a la investigación y desarrollo permite invertir en la evolución del producto y mantenerse competitivo y actualizado, desarrollando nuevos servicios, mejorando los existentes y experimentando soluciones que pueden generar nuevas oportunidades.

En este caso específico, se propone integrar en el proyecto algunas tecnologías clave que aborden los temas de integridad, seguridad y calidad de los datos. En los próximos párrafos se describirán de manera resumida tres posibles integraciones relacionadas con estos temas: blockchain, teoría de juegos e inteligencia artificial.

Es importante tener en cuenta que la línea de investigación y desarrollo está vinculada a los temas en sí y no a las tecnologías específicas que se utilizarán, siendo la única considerada imprescindible la adopción de un sistema de hardware abierto para complementar las plataformas de detección estándar.



### 5.6.2.2.1 INTRODUCCIÓN A BLOCKCHAIN

La tecnología Blockchain se basa en una estructura distribuida, compartida e inmutable que permite el registro seguro y transparente de las transacciones entre las partes implicadas. Esta tecnología ofrece varias ventajas:

- **Descentralización:** Blockchain elimina la necesidad de una autoridad central para validar las transacciones, lo que garantiza la reducción del riesgo de fraude y manipulación por parte de terceros.
- **Seguridad:** La criptografía avanzada utilizada en la cadena de bloques protege los datos y garantiza la identificación segura de las partes implicadas en una transacción.
- **Transparencia:** La cadena de bloques permite a todas las partes ver las transacciones en tiempo real y verificar la autenticidad de los datos registrados.
- **Rapidez y eficacia:** blockchain elimina intermediarios y reduce los tiempos y costes de las transacciones.
- **Inmutabilidad:** La cadena de bloques garantiza la permanencia e integridad de los datos registrados.

Un aspecto importante de la tecnología blockchain es la integridad de los datos, especialmente cuando se manejan datos críticos que requieren un alto grado de precisión, confiabilidad y ausencia de modificaciones, ya que pueden tener un impacto significativo en las actividades para las cuales se utilizan.

El concepto de consenso distribuido es fundamental en la tecnología blockchain y se logra a través de algoritmos de consenso que permiten a los nodos de la red verificar y aprobar las transacciones de forma autónoma. Existen varios algoritmos de consenso utilizados en las blockchains, pero los más comunes son el Prueba de Trabajo (*Proof of Work, PoW*) y el Prueba de Participación (*Proof of Stake, PoS*). Una vez que se alcanza el consenso, la transacción se agrega a la cadena de bloques como un nuevo bloque de datos. Este bloque se cifra y se vincula al bloque anterior de la cadena, creando un registro inmutable e incorruptible de transacciones pasadas.

La blockchain puede utilizar los contratos inteligentes (*smart contracts*), que son programas autónomos ejecutados en la red, para gestionar el almacenamiento de datos. Los contratos inteligentes permiten automatizar la gestión de los datos y las transacciones, lo que permite establecer reglas para la ejecución de operaciones específicas o garantizar que las transacciones se realicen solo si se cumplen ciertas condiciones.

Además, los contratos inteligentes pueden utilizarse para gestionar el acceso a los datos en la blockchain. Por ejemplo, se puede programar un contrato inteligente para permitir el acceso a los datos solo a usuarios específicos o para permitir el acceso solo si se cumplen ciertas condiciones.

Otra característica importante del almacenamiento de datos en la blockchain es su distribución en todos los nodos de la red. Esto significa que cada nodo tiene una copia idéntica de la cadena de bloques, lo que garantiza que los datos estén siempre disponibles y accesibles, incluso en caso de fallos o mal funcionamiento de un nodo individual o de una sección completa de la red. Esto contribuye a la resiliencia y confiabilidad de la blockchain como sistema de almacenamiento de datos.

### **5.6.2.2.2 IMPORTANCIA DE LAS ESTADÍSTICAS EN LA CALIDAD DE LOS DATOS**

Esta sección ilustrará la importancia de utilizar estadísticas para evaluar la calidad de los datos. Disponer de datos precisos y fiables es crucial en varios ámbitos de la investigación científica, entre ellos la recopilación y el análisis de datos climáticos. El análisis estadístico de datos proporciona las herramientas necesarias para detectar anomalías y evaluar la calidad de los datos, garantizando la validez de las conclusiones extraídas de ellos. La recogida de datos climáticos es una actividad compleja y exigente en la que interviene una amplia gama de sensores y dispositivos de vigilancia. Un ejemplo exitoso de análisis estadístico de datos climáticos ha sido el uso de técnicas de regresión para identificar y corregir errores en datos recogidos de sensores de temperatura (Menne et al., 2010). Estas técnicas se han utilizado con distintos fines:

- Identificación de anomalías y problemas: el análisis estadístico puede ayudar a detectar anomalías o problemas en los datos recogidos. Por ejemplo, si una temperatura detectada por el sensor se desvía significativamente de las temperaturas históricas registradas durante el mismo periodo, podría indicar un problema en la detección y requerir una evaluación cuidadosa.
- Evaluar el impacto del cambio climático: el análisis estadístico puede identificar tendencias y variaciones a lo largo del tiempo, lo que permite detectar posibles cambios climáticos, como el aumento de las temperaturas, en la región investigada (Distrito Nacional República Dominicana, 2023).
- Orientar las acciones futuras: a partir de los resultados del análisis estadístico, se pueden formular recomendaciones para mejorar la medición y el seguimiento de las variables hidrometeorológicas y climatológicas. Estas recomendaciones pueden servir para adoptar nuevas medidas que garanticen la calidad de los datos recogidos y mejoren la comprensión del clima y sus cambios a lo largo del tiempo.

Un ejemplo exitoso de análisis estadístico de datos climáticos es el uso de técnicas de agrupamiento y análisis multivariado para identificar relaciones entre variables climáticas como temperatura, presión atmosférica y humedad (Straus, 2019). Este tipo de análisis ha permitido identificar patrones climáticos y predecir condiciones meteorológicas basándose en datos históricos. El análisis estadístico de datos climáticos es un área de investigación en constante evolución, con un continuo desarrollo de

nuevas técnicas y herramientas para garantizar la confiabilidad de los resultados. La importancia del análisis estadístico y la detección de posibles anomalías ha sido ampliamente reconocida en la comunidad científica, como lo demuestran numerosas publicaciones científicas que destacan la eficacia de estas metodologías.

El uso del aprendizaje automático (*machine learning*) en combinación con la estadística puede ser útil para detectar la calidad de los datos e identificar posibles anomalías. Por ejemplo, modelos de aprendizaje automático como el análisis de componentes principales (PCA), la regresión logística o las redes neuronales pueden entrenarse para detectar anomalías en los datos climáticos, como valores anómalos de temperatura o humedad. Estos modelos pueden ser utilizados para identificar errores en los datos, mientras que el análisis estadístico puede utilizarse para evaluar la calidad de los datos identificados por el aprendizaje automático. Por ejemplo, pruebas estadísticas pueden ser utilizadas para verificar si las anomalías en los datos son significativas o aleatorias. Además, el análisis estadístico puede ser utilizado para analizar los datos de manera más profunda, identificando relaciones o tendencias entre las variables climáticas.

### **5.6.2.2.3 PROPUESTA DE INVESTIGACIÓN Y DESARROLLO**

El uso de hardware abierto para la recopilación de datos climáticos está cada vez más extendido como una solución de bajo costo para el monitoreo del clima. La creación de una amplia red de estaciones meteorológicas es una operación compleja y costosa, pero el uso de dispositivos de hardware de código abierto a bajo costo puede hacer esta operación más accesible. Estos dispositivos se caracterizan por tener una arquitectura modular y una comunidad de desarrolladores amplia y colaborativa, que puede contribuir a mejorar la funcionalidad y compatibilidad con diferentes entornos. El uso de hardware de código abierto también permite una mayor flexibilidad y adaptabilidad para agregar nuevos sensores y funciones en el futuro. En particular, se ha desarrollado un plan de investigación e implementación basado en el uso de estaciones meteorológicas equipadas con sensores para la detección y recopilación de datos climáticos.

La cadena de bloques (*blockchain*) representa una solución efectiva para garantizar la integridad de los datos recopilados por los sensores climáticos. Gracias a su estructura descentralizada e inmutable, la cadena de bloques permite registrar de manera segura y transparente la información recopilada por los sensores, sin la necesidad de una autoridad central para verificar la autenticidad de los datos. En particular, cada estación meteorológica puede registrar de manera transparente y segura la información relevante dentro de la red de la cadena de bloques. Esta información puede ser validada a través de mecanismos de consenso que requieren la aprobación de múltiples nodos para su inclusión en la cadena de bloques. El uso de esta tecnología también permite garantizar el origen y la procedencia de los datos, lo que permite rastrear todo el recorrido de los datos recopilados por el sensor, desde su creación hasta su

almacenamiento en la cadena de bloques. Además, la tecnología de la cadena de bloques ofrece un alto nivel de seguridad y protección de datos, ya que cada bloque en la cadena se cifra con algoritmos avanzados, garantizando la imposibilidad de modificar los datos una vez que han sido registrados en la cadena de bloques.

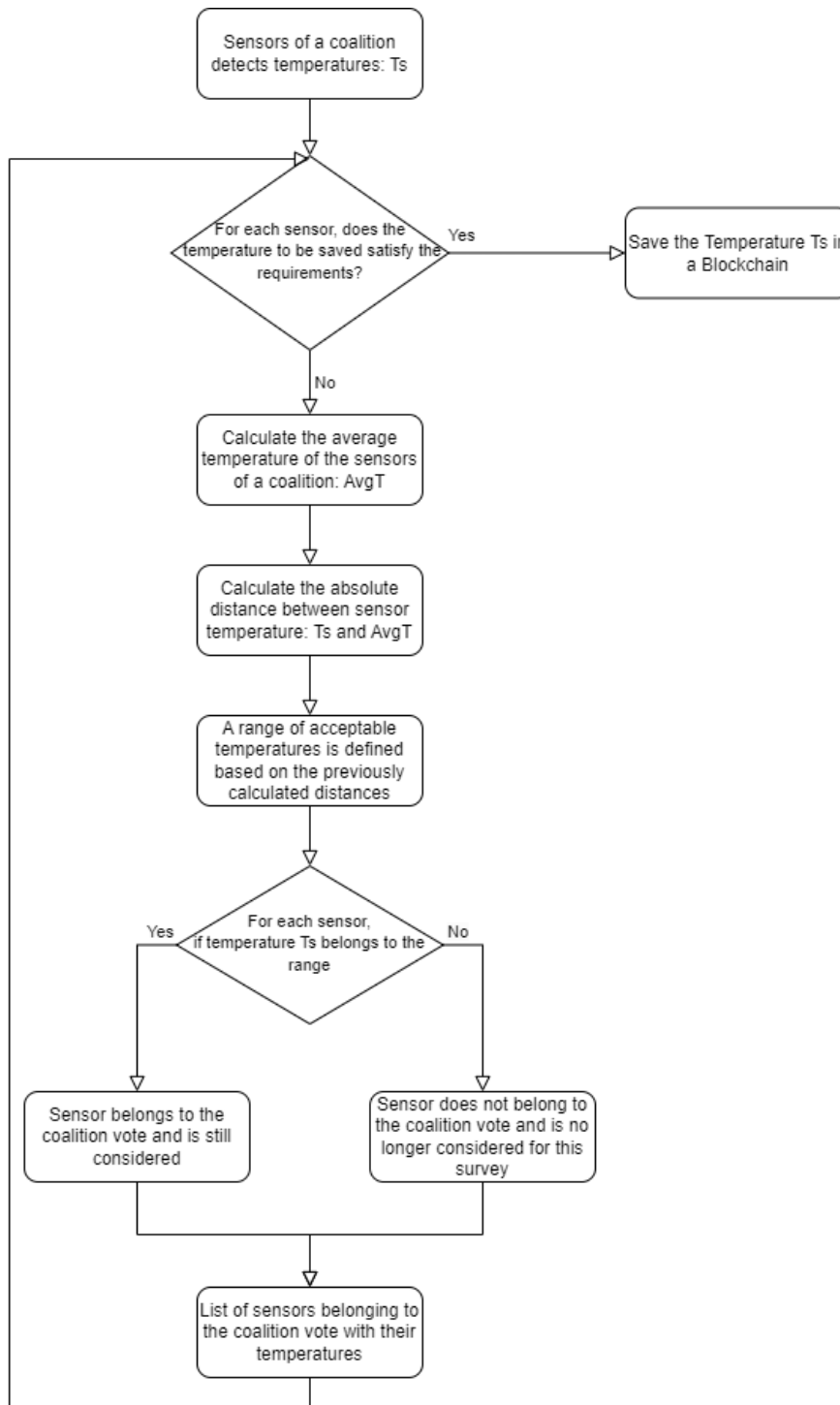
Es necesario profundizar en el plan de investigación para determinar qué datos serán certificados en la cadena de bloques, evitando problemas relacionados con las tarifas de gas, el rendimiento y la escalabilidad. El objetivo es limitar la cantidad de información almacenada en una cadena de bloques a solo la información esencial y relevante para la aplicación específica, evitando sobrecargar la cadena de bloques con datos innecesarios o superfluos. Este enfoque se basa en varios principios, incluyendo mantener la cadena de bloques liviana y compacta para garantizar eficiencia en términos de espacio de almacenamiento, tiempos de transacción y costos de transacción.

Además, la escalabilidad es un aspecto crucial a considerar. Reducir la cantidad de datos almacenados en la cadena de bloques permite disminuir el tamaño general de la cadena, simplificando su gestión y permitiendo una mayor escalabilidad. Una solución propuesta es incluir en la cadena de bloques solo los datos provenientes de estaciones meteorológicas ubicadas en sitios estratégicos, capaces de proporcionar información de mayor importancia en comparación con otras estaciones. Se gestionarán mecanismos de prioridad entre las estaciones, de modo que los datos provenientes de estaciones con mayor prioridad tengan preferencia en la inclusión en la cadena de bloques en comparación con aquellos provenientes de estaciones de menor prioridad. El objetivo es minimizar las tarifas de gas y la sobrecarga de datos en la cadena de bloques.

Una posible solución podría ser integrar la cadena de bloques directamente en la fuente de generación de datos, para garantizar la confiabilidad del origen de los datos mismos. Las lógicas de evaluación de la calidad de los datos podrían implementarse a través de contratos inteligentes, pero deberán ser invocadas por los clientes. Por lo tanto, sería deseable adoptar una solución en la que la cadena de bloques se integre directamente con el sensor físico, evitando el uso de capas intermedias que podrían ser vulnerables y causar inconsistencias en los datos (Perez, 2018).

#### **5.6.2.2.4 USO DE LA TEORÍA DE JUEGOS**

Además de lo discutido en los párrafos anteriores, el uso de un algoritmo basado en la teoría de juegos puede ser de vital importancia para la calidad de los datos (Casada-Vara et al., 2018). La teoría de juegos es una disciplina de las matemáticas aplicadas que estudia el comportamiento estratégico de los individuos en situaciones de interacción. En el contexto de los datos climáticos, puede aplicarse para comprender cómo diferentes actores pueden interactuar entre sí e influir en los resultados generales. En este sentido, se desarrollará un algoritmo basado en la teoría de juegos que fomentará la cooperación entre los sensores instalados en una zona específica.



**Figura 20**

Una vez establecido el número de jugadores, será importante determinar la distancia mínima entre los sensores para garantizar una comunicación confiable y reducir las discrepancias en la detección de los datos climáticos. Esto será objeto de más estudios e investigaciones para abordarse adecuadamente.

La distancia mínima entre los sensores es un parámetro importante utilizado para definir una "coalición" de sensores que colaboran para garantizar una comunicación confiable

y reducir las discrepancias en la detección de los datos climáticos. Por ejemplo, en una estación climática, es común tener al menos tres copias del mismo sensor para crear una mayoría de decisiones y formar un grupo de sensores cooperativos dentro de la misma coalición. Cada coalición tiene asociada una función característica del juego que define su comportamiento.

Sin embargo, el número máximo de sensores que pueden formar parte de una coalición está limitado por restricciones presupuestarias para la construcción e implementación física de los sensores, mientras que el número mínimo de sensores en una coalición se establece previamente.

Inicialmente, será necesario establecer el número mínimo de jugadores, lo cual será objeto de investigación, ya que requerirá un análisis preliminar cuidadoso para seleccionarlo adecuadamente. Basándose en algunas consideraciones, se ha establecido que al menos tres jugadores son necesarios para garantizar la formación de una mayoría en las decisiones que se toman.

Dado que un sensor individual no tiene la capacidad de determinar si los datos recopilados son correctos o no, es necesario que colabore con otros sensores dentro de su coalición para verificar la precisión de las mediciones climáticas y corregirlas si es necesario. Esto se logra a través de un mecanismo de mayoría, en el cual la coalición toma decisiones basadas en la mayoría de los votos para establecer el valor correcto de los datos recopilados.

### **5.6.2.2.5 USO DE AI**

En esta etapa, el uso de Inteligencia Artificial (IA) se presenta como una opción efectiva para el análisis de la calidad de los datos provenientes de los sensores de IoT (*Internet de las cosas*). Gracias a la IA basada en patrones de datos, es posible mejorar significativamente la precisión, eficiencia y automatización del análisis de calidad de datos (Marr, 2018).

La IA puede desempeñar un papel clave en la detección de errores en los datos. Una oportunidad que brinda el uso de la IA en el análisis de calidad de datos es la capacidad de detectar patrones y tendencias ocultas en los datos. Esto puede permitir identificar posibles problemas o fallas en el sistema de IoT, así como detectar tipos específicos de errores que ocurren con mayor frecuencia en ciertas condiciones ambientales o períodos del año. Esto permite tomar medidas preventivas o correctivas de manera oportuna. Además, la IA puede identificar datos faltantes o datos fuera de escala, lo que podría indicar un mal funcionamiento del sensor o una compromiso de los datos.

La inteligencia artificial puede analizar grandes cantidades de datos provenientes de sensores de IoT para identificar correlaciones y relaciones entre variables que pueden no

ser inmediatamente evidentes para los analistas humanos. Además, se puede utilizar para predecir el rendimiento futuro de los sensores, garantizando que siempre funcionen a su máximo potencial.

El uso de esta tecnología también permite automatizar el proceso de análisis de calidad de datos, reduciendo la necesidad de intervención humana. La IA puede realizar controles de manera continua y en tiempo real, lo que permite una respuesta inmediata ante los errores. Esto reduce la dependencia de los operadores humanos y mejora la eficiencia del proceso de análisis de calidad de datos.

Es fundamental destacar que la IA basada en patrones de datos requiere una preparación y limpieza adecuada de los datos de entrenamiento para garantizar su confiabilidad y precisión. Los datos de entrenamiento inadecuados o no representativos podrían comprometer la calidad de los resultados obtenidos.

En resumen, la adopción de IA ofrecería beneficios significativos y sería una elección estratégica para mejorar la eficiencia operativa y la calidad de los datos. Gracias a las capacidades de aprendizaje automático y análisis de datos, la IA puede identificar rápidamente errores, duplicados e inconsistencias, lo que permite tomar decisiones basadas en información cada vez más precisa y confiable. Además, como se mencionó anteriormente, ahorra tiempo y costos al evitar la necesidad de realizar actividades de limpieza de datos manualmente.

### **5.6.2.3 SISTEMAS DE REGISTRO DE SERVICIOS PÚBLICOS**

La plataforma SURFACE no dispone actualmente de un sistema automatizado de registro de usuarios. Sin embargo, si se desea desarrollar más funcionalidades para los usuarios registrados, es esencial crear simultáneamente un sistema de registro moderno y eficaz para usuarios simples que no requiera la aprobación humana y no dependa totalmente del registro manual de usuarios.

El mismo sistema podría utilizarse para la creación de usuarios de tipo productor o gubernamental. Sin embargo, los mecanismos específicos para definir dichos registros dependen de los requisitos de gobernanza, que se definirán en una fase posterior. Será importante definir tanto la entidad responsable de gestionar a los usuarios una vez implantada la plataforma como la forma de gestionar a estos usuarios dentro del sistema de registro.

#### **5.6.2.4 FUNCIONALIDAD, INTERFAZ Y EXPERIENCIA DEL USUARIO**

Entre las opciones de desarrollo recomendadas para la plataforma, se recomienda encarecidamente ampliar la funcionalidad para los usuarios, como se comenta brevemente en la sección de requisitos funcionales<sup>[52]</sup>.

En particular, estas funcionalidades tendrán como objetivo mejorar la experiencia del usuario mediante la introducción de mejoras en la calidad de vida, como la posibilidad de guardar filtros personalizados y accesos directos, pero sobre todo mediante el desarrollo de una experiencia de usuario (UX) y una interfaz de usuario (UI) específicas para la futura plataforma CDMS dominicana, incluyendo opciones de accesibilidad y diseños móviles<sup>[53]</sup>.

---

<sup>52</sup> Apartado **3.2.1.6**

<sup>53</sup> Capítulo **6**



## 6 DISEÑO DE UX Y UI PARA ESCRITORIO Y VISUALIZACIÓN WEB MÓVIL

El diseño es una fase clave para garantizar el éxito de un producto o servicio, y diseñar cualquier producto sin tener en cuenta las necesidades de los usuarios puede conducir a graves errores. Muchas veces interactuamos con productos cuyo funcionamiento no entendemos, y solemos sentirnos culpables por ello, pero en realidad el problema se debe a un mal diseño del propio producto (Norman, 2019).

En la sociedad moderna, donde los sitios web y las aplicaciones desempeñan un papel cada vez más importante en la vida cotidiana, resulta esencial situar al usuario en el centro del proceso de diseño y desarrollo. Es necesario diseñar para los usuarios y no para uno mismo: el diseño *centrado en el ser humano* es, por tanto, esencial.

Incluso en el mundo de las nuevas tecnologías, existen herramientas para un buen diseño, entre ellas el diseño UX (*User Experience*) y UI (*User Interface*), dos conceptos fundamentales que se introducirán en los párrafos siguientes.

### 6.1 DISEÑO UX

El diseño UX es una disciplina en la que intervienen diversas competencias y conocimientos, como la psicología cognitiva, la usabilidad, la investigación de mercados, el diseño gráfico y la ingeniería. El objetivo principal es crear productos y servicios fáciles de usar, agradables y que satisfagan las necesidades de los usuarios.

Para lograrlo, el diseñador de UX debe utilizar un enfoque basado en la empatía y la comprensión de las necesidades del usuario. Esto significa que tiene que estudiar el comportamiento de los usuarios, sus hábitos, necesidades y expectativas para diseñar soluciones adaptadas a sus necesidades.

El proceso de diseño en UX suele basarse en una serie de iteraciones, en las que los prototipos se prueban con los usuarios para recabar opiniones y mejorar continuamente el producto. De este modo, el diseñador de UX puede comprobar si las soluciones propuestas responden a las necesidades de los usuarios y si son eficaces desde el punto de vista de la usabilidad.

Para alcanzar los objetivos del diseño UX, existen normas específicas que los diseñadores pueden seguir, como las normas ISO y las directrices de accesibilidad web. Además, hay muchos autores y expertos en el campo del diseño UX que han desarrollado métodos,

técnicas y patrones de diseño que pueden utilizarse para mejorar la calidad de la experiencia del usuario.

Para ayudar a comprender mejor el concepto de diseño de UX, se puede hacer referencia a las diez *Heurísticas* de Nielsen (1994a, 1994b), que representan un conjunto de reglas que todo producto debe seguir para garantizar una experiencia de usuario satisfactoria y fácil de usar:

- **Visibilidad del estado del sistema:** el sistema debe mantener siempre informado al usuario (feedback).
- **Correspondencia entre el sistema y el mundo real:** el sistema debe referirse a patrones psicológicos del mundo real, acercándose así al usuario.
- **Control y libertad del usuario:** hay que dar al usuario el control sobre el producto y las acciones que realiza y ayudarlo a prevenir errores.
- **Coherencia y normas:** definimos y utilizamos un lenguaje coherente en todo el sistema.
- **Prevención de errores:** ayudamos al usuario, con diversas herramientas, a prevenir errores.
- **Reconocer en lugar de recordar:** minimizamos el esfuerzo mental del usuario haciendo que todo sea reconocible y que las instrucciones de uso del sistema sean visibles.
- **Flexibilidad y eficacia de uso:** damos a los usuarios experimentados la oportunidad de facilitar los procesos mediante atajos y otras herramientas.
- **Diseño estético y minimalista:** el usuario necesita datos claros y visibles, por eso ofrecemos interfaces de usuario sencillas y "sin ruido".
- **Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores:** siempre damos indicaciones claras de los errores y ayudamos a los usuarios a resolverlos.
- **Documentación:** si el sistema es muy complejo, proporcionamos documentación clara y completa como herramienta adicional.

Siguiendo estas (y otras) reglas y mediante la investigación y el análisis, el diseñador de UX es capaz de diseñar productos eficaces y fáciles de usar.

## 6.2 DISEÑO DE INTERFAZ DE USUARIO

El diseño de interfaz de usuario (UI) es el diseño de los elementos visuales de un producto digital, servicio digital o sitio web con los que se relaciona un usuario. El diseñador de interfaz de usuario se encarga entonces de diseñar el aspecto visual y funcional de los elementos de la interfaz de usuario -como diseños, botones, iconos, menús, formularios, etc.- de manera que los usuarios puedan interactuar con el producto de forma intuitiva y satisfactoria. El diseñador de interfaz de usuario aplica los mismos procesos que el

diseñador de experiencia de usuario a la parte visual del producto, diseñando un sistema e iterándolo en función de la investigación y los comentarios.

## 6.3 ESCENARIO A - SURFACE

Esta sección ofrece un rápido análisis de SURFACE, una plataforma de gestión y consulta de datos meteorológicos de Belice. El front-end de SURFACE se diseñó con Vuetify, un framework JavaScript gratuito basado en Vue.js. Este framework ofrece una biblioteca de elementos diseñados según las directrices de la segunda versión del Material Design de Google.

Material Design es uno de los principales lenguajes visuales de código abierto diseñados y utilizados por Google. La dependencia de un lenguaje visual de este tipo tiene dos críticas fundamentales: la primera es que crea una estandarización excesiva de todos los productos; la segunda es que, con el tiempo de diseño y el corto ciclo de vida de estas directrices, siempre se corre el riesgo de ofrecer a los usuarios algo "viejo". El lado positivo, sin embargo, es el gran saber hacer que hay detrás del diseño de las directrices.

### 6.3.1 ANÁLISIS DE SURFACE

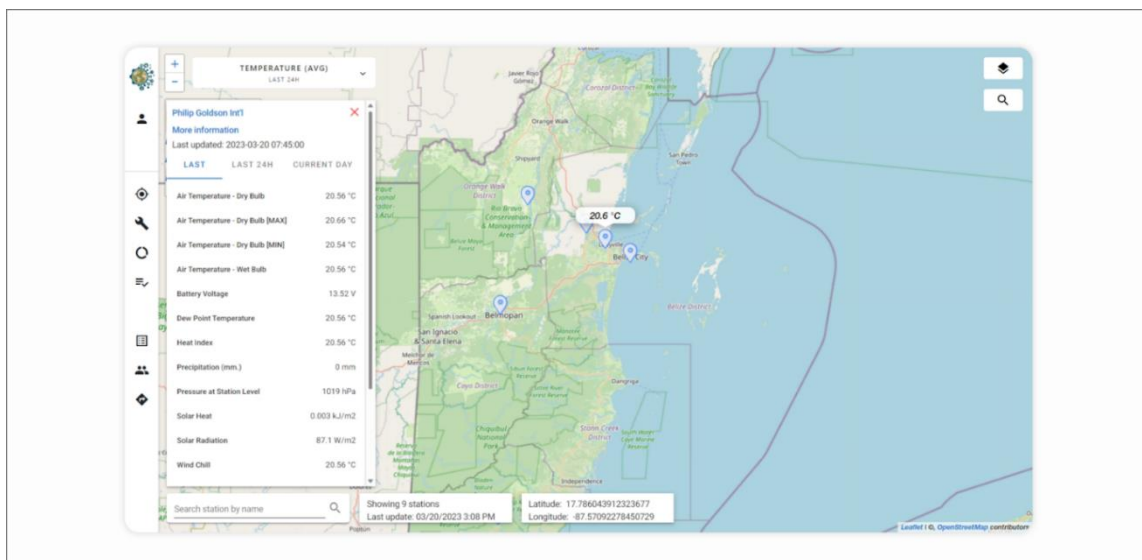


Figura 21

SURFACE es una plataforma que se ha diseñado centrándose medianamente en cuestiones relacionadas con la experiencia del usuario. Esto significa que, para las funciones básicas, hay algunos trucos que ayudan al usuario con la navegación. Al

mismo tiempo, algunas opciones pueden resultar confusas o llevar al usuario a estados de frustración debido a una mala orientación.

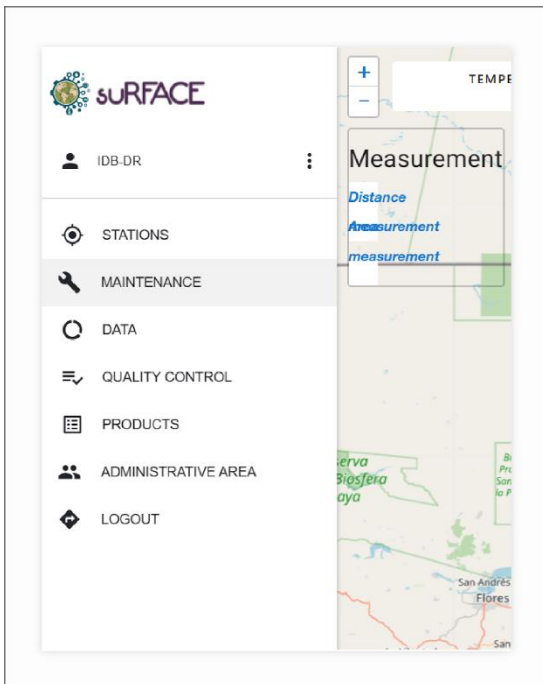


Figura 22

Intentemos hacer un análisis parcial de la pantalla principal del mapa, para entender juntos qué se ha diseñado bien y qué se podría mejorar, y también para dejar claro cómo unos pequeños toques pueden ser decisivos en una buena experiencia de usuario.

En esta pantalla, el usuario dispone de un mapa en el que visualizar información geolocalizada. Una buena práctica consiste en comprimir el menú de funciones de la izquierda (Navigation Rail, **Fig. 22**). De este modo, el usuario dispone de más espacio para ver el mapa. Lo malo es que no hay opción de bloquear la barra expandida o comprimida, y cuando pasas el puntero por encima, aunque sea por accidente, aparece una animación que comprime y descomprime todo el mapa de forma molesta.

Es excelente la presencia de información especial para aumentar el conocimiento del usuario como, por ejemplo, el número de estaciones mostradas y la hora de la última actualización.

Negativo, sin embargo, es el posicionamiento de los elementos y herramientas en el mapa: la barra de búsqueda suele estar arriba o arriba a la derecha. Moverlo a la parte inferior izquierda sin razón aparente aumenta la frustración del usuario. Lo mismo podría aplicarse a los botones de zoom (+ y -) a la izquierda, cuando normalmente están situados a la derecha.

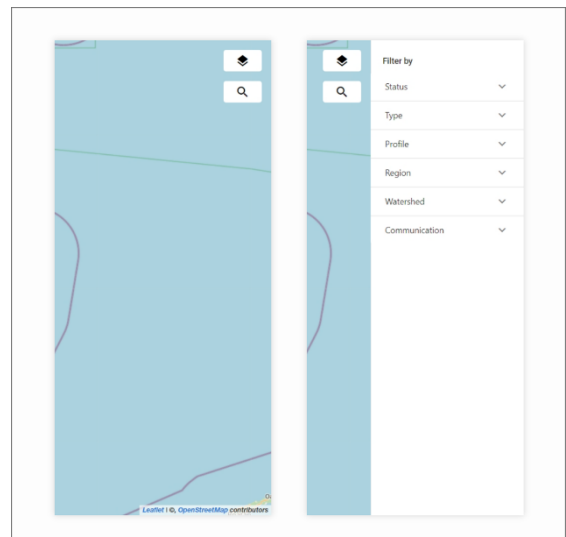


Figura 23

Un elemento crítico es la hoja lateral (**Fig. 23**) que abre la función de filtrado de estaciones (por el momento, el icono es probablemente incorrecto debido a algún desarrollo en curso de la plataforma). Cuando el usuario pulsa el botón, se abre la hoja lateral y puede seleccionar los distintos filtros. Aunque la información inmediata que el usuario ve en el mapa es muy buena, no lo es tanto la gestión de la navegación: no parece haber posibilidad de cerrar la sección de filtros. Sólo después de varios intentos dispares te das cuenta de que puedes cerrarlo pulsando el botón por segunda vez.

También se echa de menos en el diseño de superficies la atención a la coherencia de la interfaz de usuario (**Fig. 24**). Si examináramos el conjunto de iconos, nos daríamos

cuenta de que no se aprecia ninguna coherencia estilística. Algunos iconos están rellenos, mientras que otros sólo tienen un contorno. Esto hace que el usuario no tenga sentido del orden y el resultado son iconos que, aunque alineados, tienen distinto peso visual. Además, como puede verse en la figura siguiente, los iconos del mapa tienen un estilo completamente distinto. Un buen compromiso se encuentra en el conjunto de botones que tienen un solo tipo gráfico y son bastante reconocibles en toda la plataforma. Otra nota es el uso del color rojo, que, aunque puede ser una elección estilística, debería limitarse a los mensajes de error.

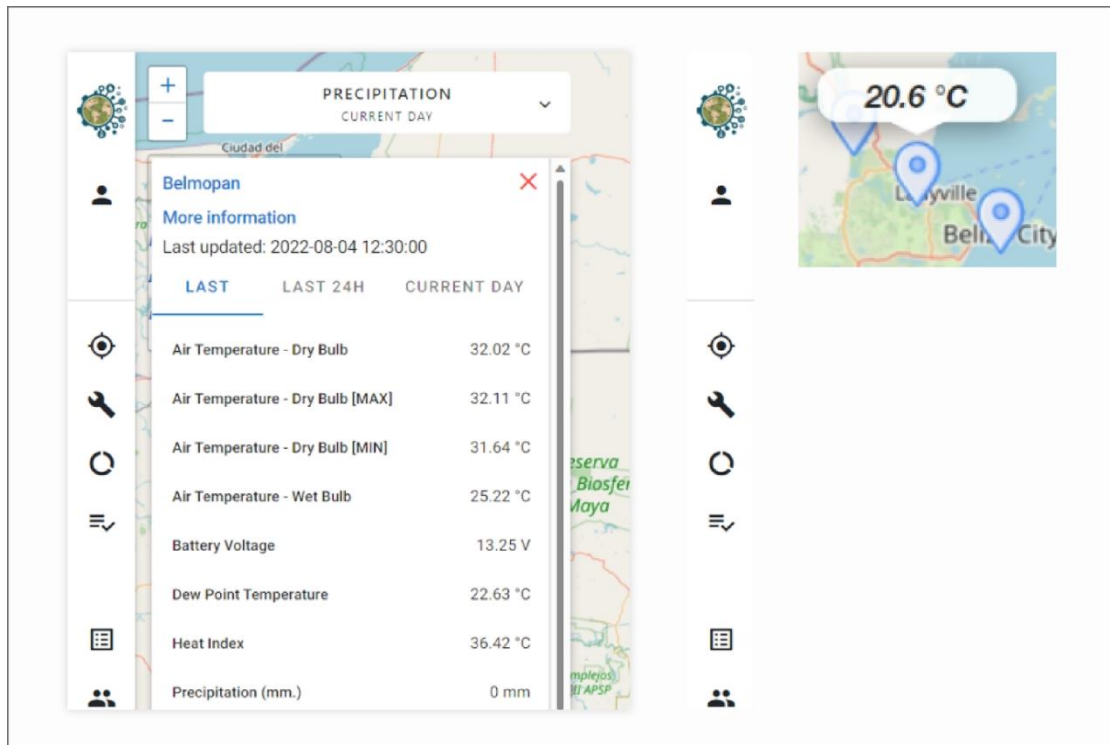


Figura 24

El mérito absoluto se debe sin duda a la investigación interna de las estaciones meteorológicas. Cuando el usuario empieza a escribir, la plataforma hojear el contenido de la búsqueda mostrando todas las estaciones meteorológicas que tienen las letras escritas en su nombre (**Fig. 25**). De este modo, se ayuda al usuario de forma proactiva: la plataforma disminuye el estrés del usuario y, al mismo tiempo, ayuda a prevenir errores.

Otras cuestiones importantes son las relacionadas con el espaciado entre elementos y la capacidad de respuesta. Aquí entran en juego tanto un problema de diseño como una cuestión de instrumentación -de ahí el uso del marco-, lo que no permite una total libertad de diseño.

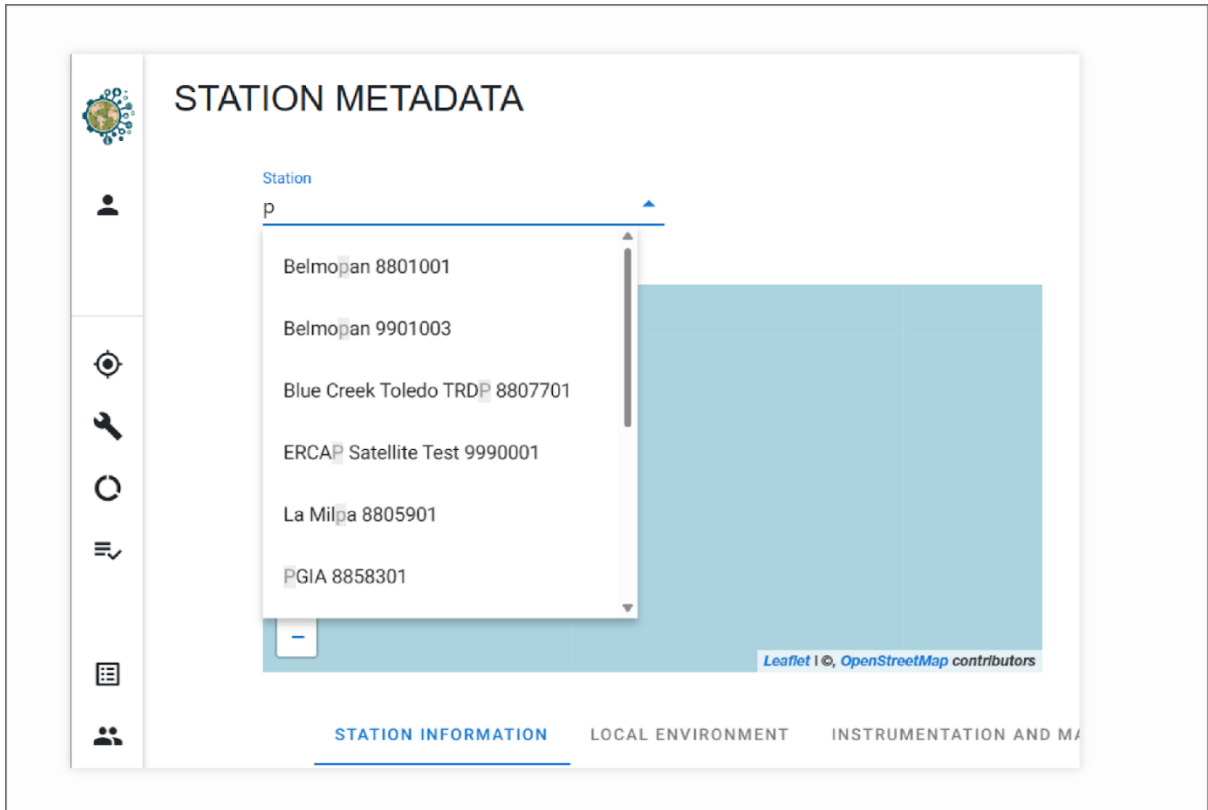


Figura 25

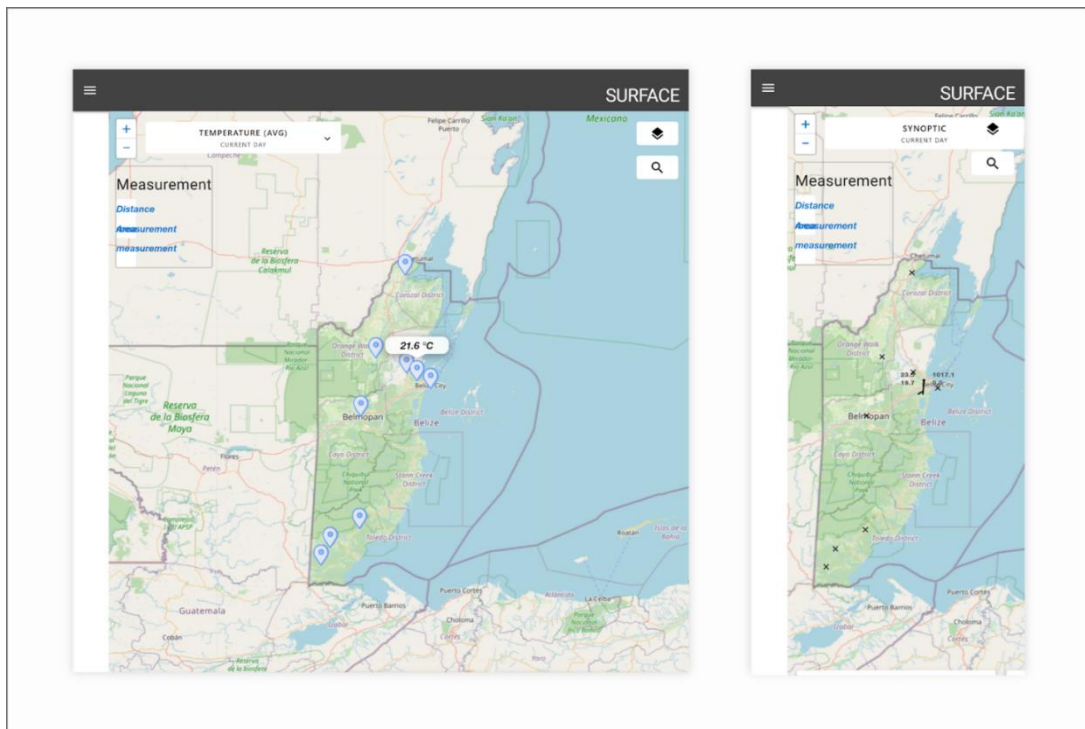


Figura 26

En esta primera pantalla, relacionada con la vista del mapa, observamos que, en primer lugar, hay algunos problemas con el menú. Es interesante ver cómo algunos componentes clave para el usuario, como la búsqueda de emisoras, desaparecen al cambiar el tamaño de la pantalla utilizada (**Fig. 26**).

No hay muchos cambios en la pantalla de la lista de estaciones (**Fig. 27**). Mientras que a nivel de escritorio (pantalla izquierda) hay una mejor gestión, vemos como la vista móvil siempre tiene los problemas de posicionamiento y gestión de contenidos.

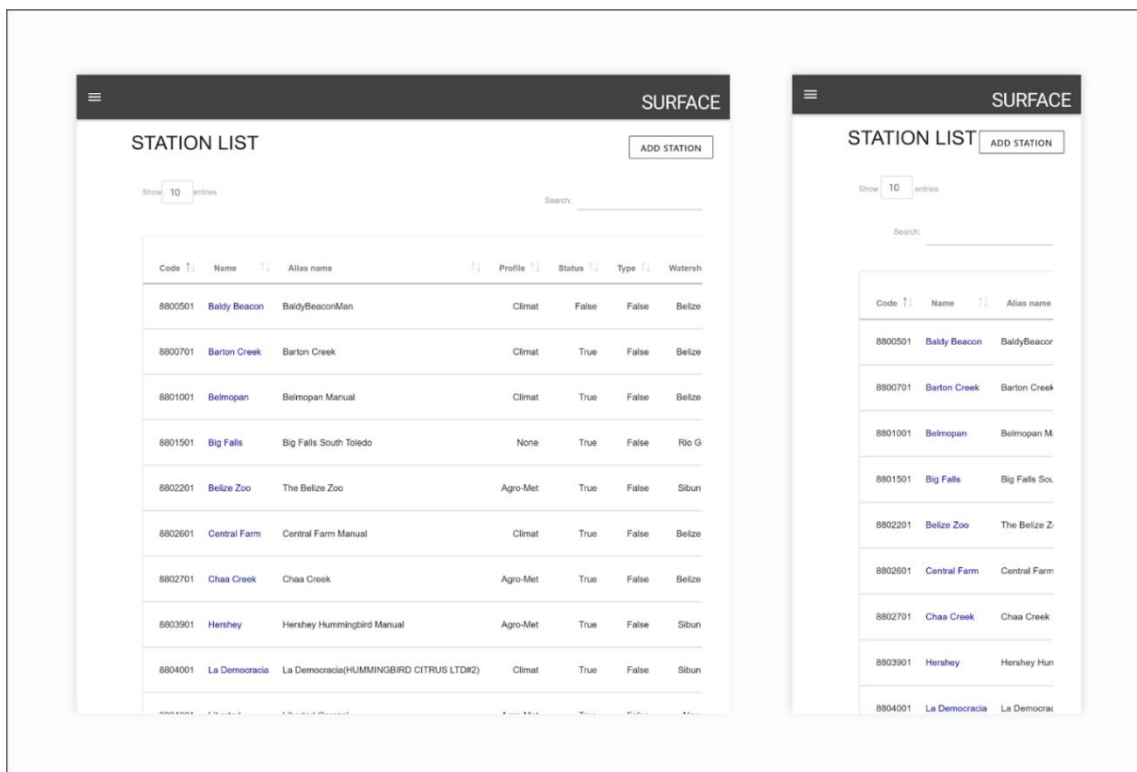


Figura 27

Lo mismo ocurre con la pantalla Datos de la estación (**Fig. 28**), que si consigue mantener la integridad a nivel de escritorio (pantalla izquierda), se vuelve completamente loco en la vista móvil.

Esta cuestión puede parecer menor, pero con el aumento exponencial del uso de dispositivos móviles, hay que tener en cuenta las necesidades de todos los usuarios. Por supuesto, es posible optar por omitir ciertas funciones de la pantalla del móvil para evitar que se realicen acciones muy complejas en pantallas pequeñas: esto aumentaría la susceptibilidad a errores y la percepción negativa del usuario.

Sin embargo, es imprescindible ofrecer una visualización adecuada a través de los dispositivos móviles.

En la parte final de este análisis, también hay que prestar poca atención a los usuarios como grupo de individuos diferentes. Este ámbito del diseño de la experiencia del usuario está cobrando cada vez más importancia en el diseño de productos para

permitir que las barreras entre personas con capacidades diferentes se derriben en la medida de lo posible<sup>[54]</sup>.

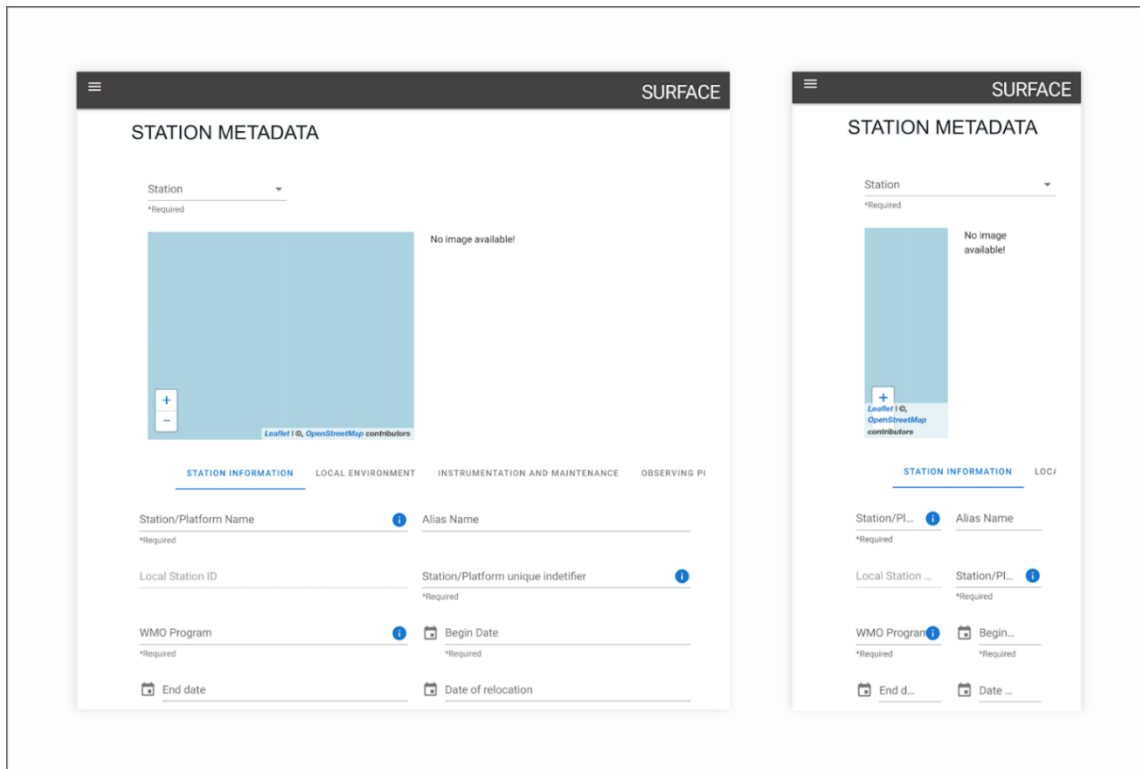


Figura 28

Actualmente, la plataforma carece de herramientas como la elección del idioma, la posibilidad de mostrar la interfaz en modo oscuro o con una opción de alto contraste, y la gestión de elementos de lectura de pantalla.

## 6.4 ESCENARIO B - UI PERSONALIZADA

Este proyecto reviste especial importancia por tratarse de una plataforma de gestión y consulta de datos meteorológicos de importancia nacional y alcance internacional. Un proyecto de esta envergadura debe diseñarse y estructurarse necesariamente con arreglo a estrictas normas de calidad y no puede, por tanto, prescindir del cuidado en lo que respecta a la Experiencia de Usuario y la Interfaz de Usuario. Esta consideración se hace teniendo en cuenta que muy a menudo los sitios web de las administraciones públicas, o de las instituciones, suelen ser poco intuitivos, antiguos y poco participativos. Este statu quo es un verdadero problema para los ciudadanos, que deberían poder

---

<sup>54</sup> Para más detalles, consulte el apartado 6.4.3



interactuar sin problemas con los canales institucionales a través de sitios y plataformas bien mantenidos y fáciles de usar.

La interfaz de usuario personalizada diseñada y presentada en los párrafos siguientes pretendía incorporar todas las mejoras derivadas de la fase de investigación de la competencia mediante el desarrollo de funciones y diseños de interfaz de usuario que cumplieran mejor las normas de UX y diseño de interfaz de usuario.

## **6.4.1 FASE DE DISEÑO**

### **6.4.1.1 PRIMER ENFOQUE DE DISEÑO UX**

Para el primer hito de la plataforma y su interfaz de usuario personalizada, se llevó a cabo un análisis preliminar de la competencia con el fin de identificar problemas relacionados con la experiencia y la interfaz de usuario.

Para apoyar el diseño de estos pasos, se creó internamente documentación de gestión de flujos centrada en las acciones de los usuarios. Con este fin, se crearon User Personas para representar a distintos tipos de usuarios con necesidades diferentes.

Como paso siguiente, se crearon wireframes de baja fidelidad para comprender el diseño y la gestión de los contenidos. A continuación, se crearon wireframes de alta fidelidad utilizando una interfaz de usuario estándar y de código abierto como Material Design 3.

Durante esta fase se racionalizaron algunas funciones y se introdujeron pequeños cambios basados en las necesidades de los usuarios. Todo el proceso de diseño se centró en atender las necesidades de los usuarios.

### **6.4.1.2 DISEÑO DE LA INTERFAZ DE USUARIO**

La interfaz de usuario (IU) es la parte del proceso de diseño que se centra en los aspectos visuales y gráficos del producto. Se basa en wireframes creados durante la fase de diseño de la experiencia del usuario (UX) y se perfecciona y enriquece para lograr el resultado final del producto.

El principal objetivo del diseño de interfaz de usuario es presentar la información de forma clara y sencilla, teniendo siempre en cuenta las necesidades de los usuarios, así como el aspecto estético del producto final.

Durante el diseño de la interfaz de usuario se define el estilo de elementos como la forma, el color, la tipografía y la aplicación de espacios coherentes. El estilo definido debe ser coherente en todo el producto para crear una experiencia visual unificada.

Para una descripción más detallada de las opciones de diseño, consulte el **APÉNDICE B**.

## 6.4.2 ANÁLISIS FUNCIONAL

### 6.4.2.1 VISTA DEL MAPA

El Mapa es la parte central y fundamental del proyecto, ya que ofrece a los usuarios una visualización inmediata de las estaciones meteorológicas y sus datos. También sirve de página de inicio del proyecto y, por tanto, es la primera página con la que interactúan los usuarios (**Fig. 29**).

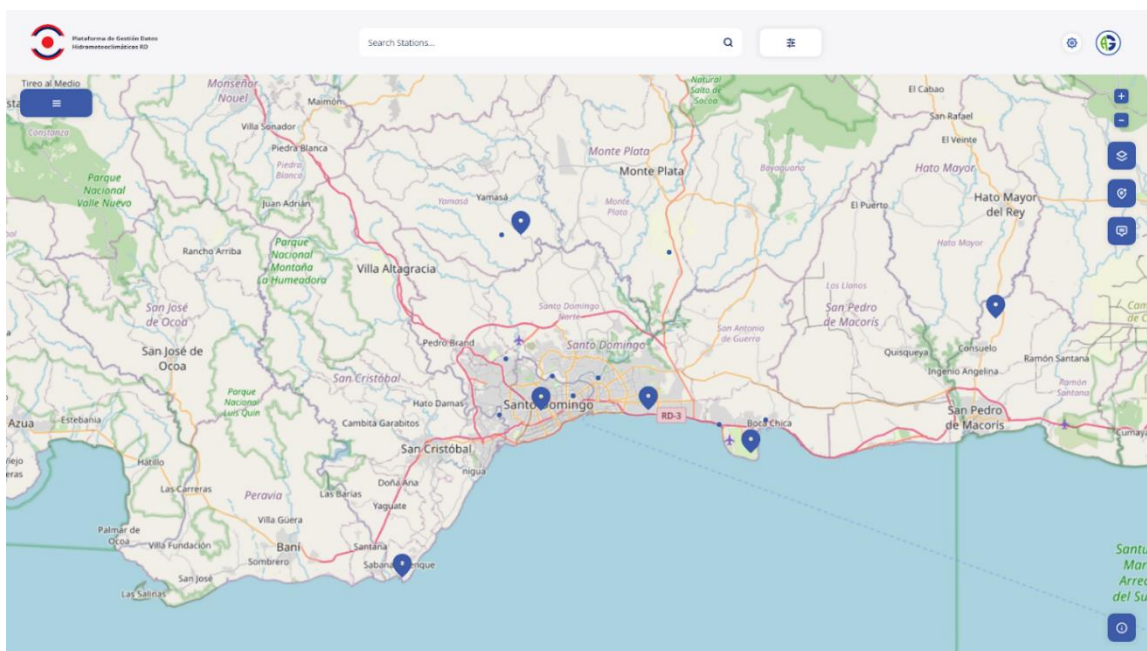


Figura 29

En la parte superior de la pantalla hay una sección con el logotipo del proyecto, una barra de búsqueda para buscar estaciones meteorológicas por nombre o código y un botón para filtrar la búsqueda. En la esquina superior derecha hay un botón para acceder a la configuración y otro para gestionar el perfil de usuario.

Como es habitual, hay un botón con el icono de un menú de hamburguesas para abrir la navegación de la plataforma (**Fig. 29**), mientras que en la parte derecha de la pantalla hay botones para la gestión del mapa, como el zoom, la capa del mapa, la función de selección múltiple y la función de visualización de etiquetas. Además, en la parte inferior hay información sobre el sistema, como el número de estaciones presentes, la última actualización, los créditos y mucho más.

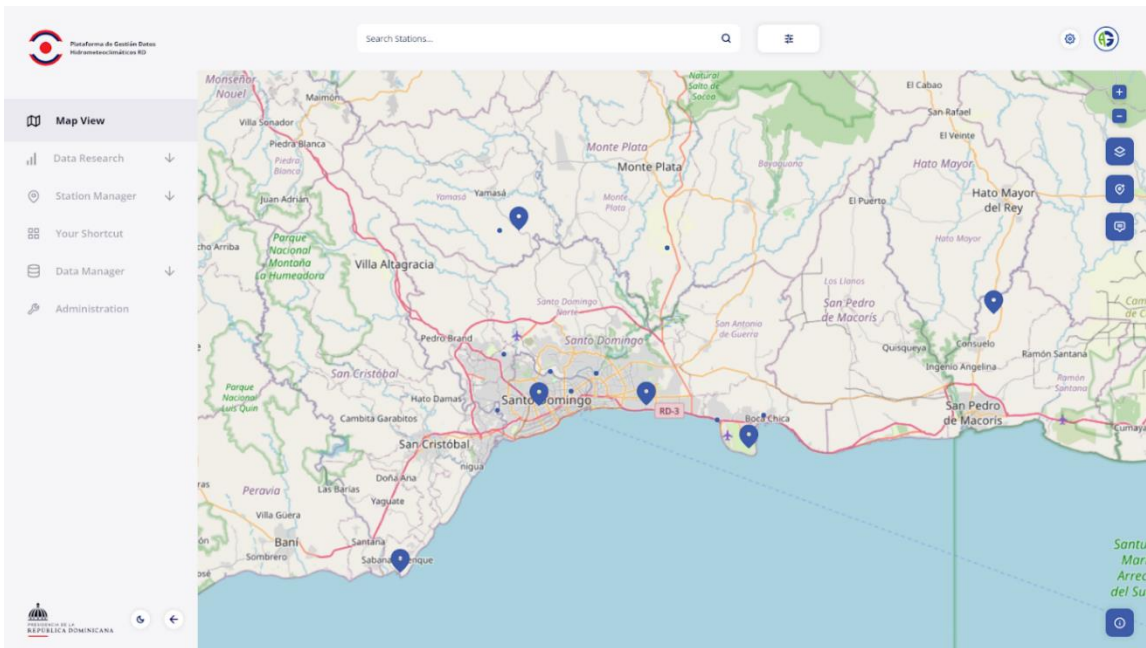


Figura 30

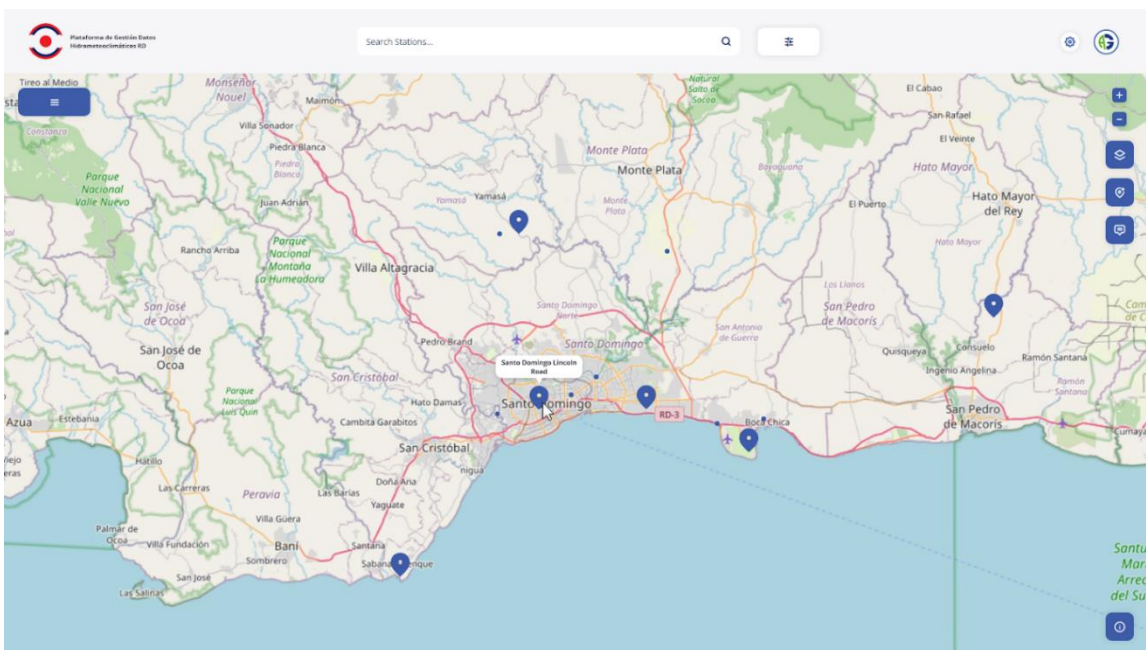


Figura 31

La **Figura 30** muestra el menú abierto, que permanece fijo una vez abierto, pero ofrece al usuario la opción de cerrarlo con una flecha en la parte inferior. En el menú, la opción "Vista del mapa" resalta la posición actual del usuario.

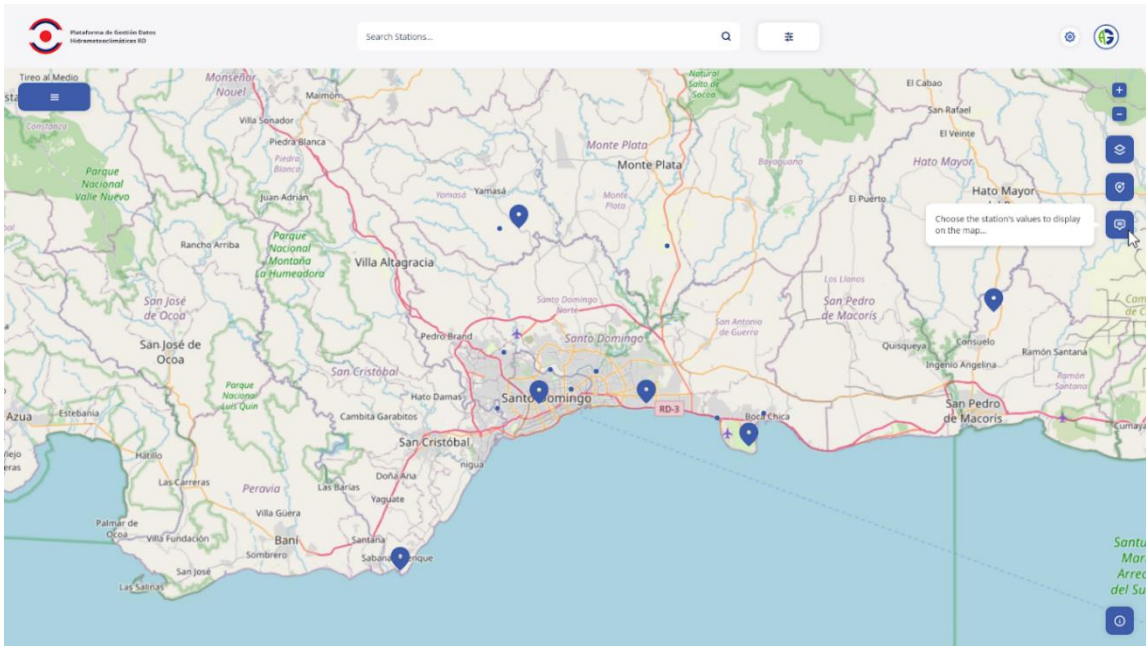


Figura 32

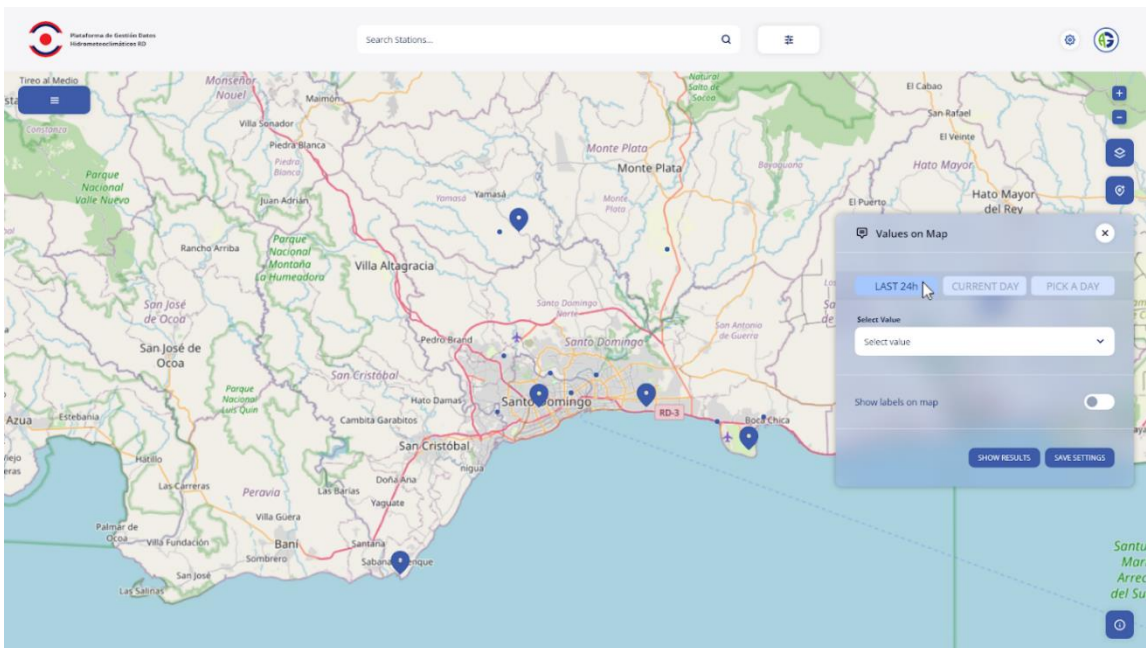


Figura 33

Cuando el usuario pasa el ratón por encima de una estación meteorológica, aparece un pequeño tooltip con el nombre de la estación (**Fig. 31**). El usuario también tiene la opción de visualizar determinados tipos de datos directamente en el mapa mediante la función "Valores en el mapa" (en **Fig. 32** en **Fig. 36**).

Cuando el usuario pulsa el botón, se abre un cuadro de diálogo con opciones de visualización (en **Fig. 33** en **Fig. 35**).

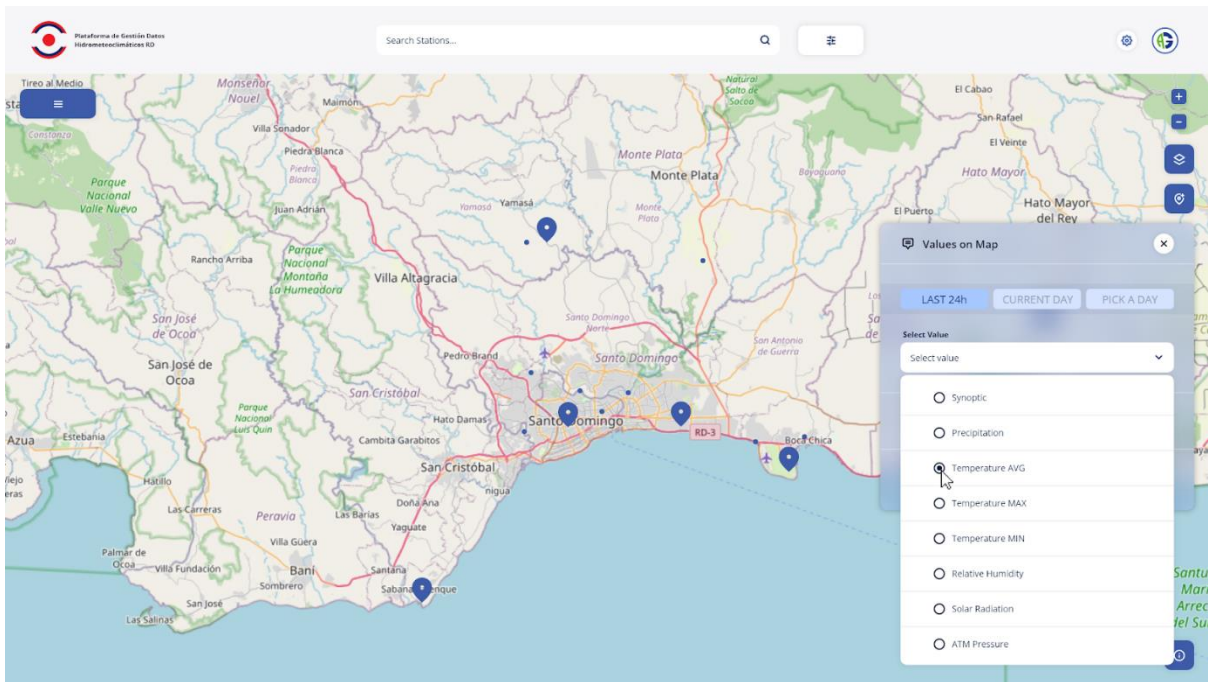


Figura 34

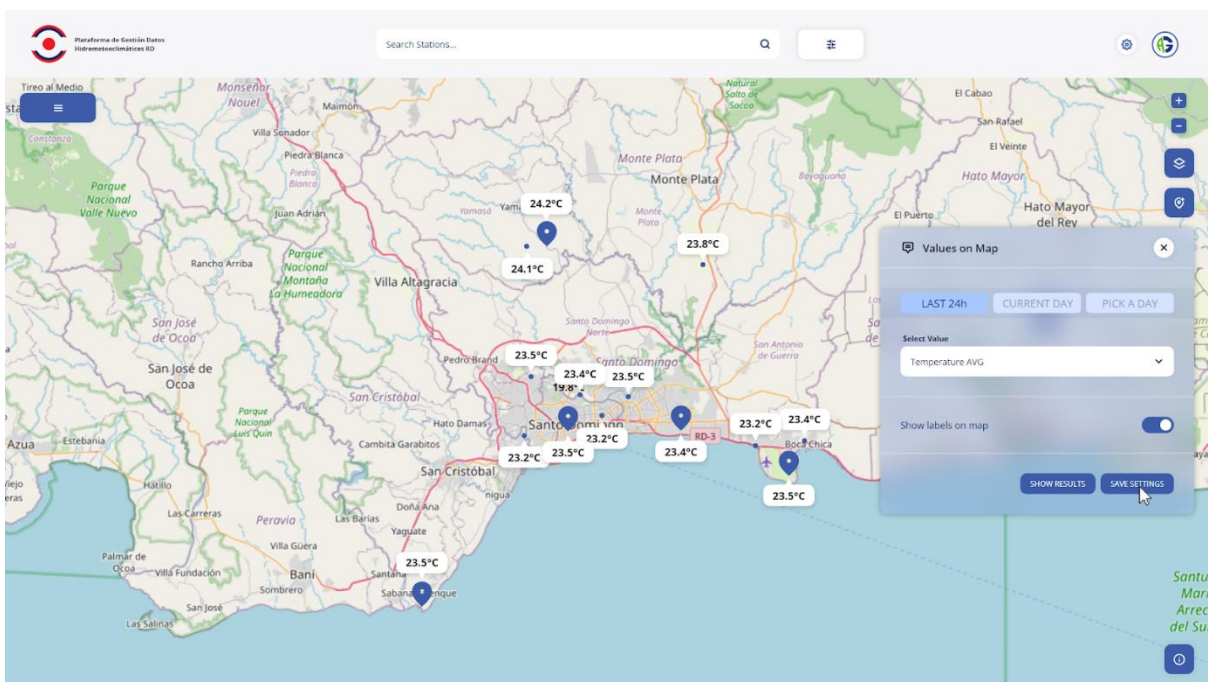


Figura 35

En la vista de mapa, el usuario puede ver todos los datos relacionados con una variable para todas las estaciones meteorológicas del mapa. Al pasar el ratón por encima de los mapas, se despliega un tooltip que muestra también el nombre de la estación meteorológica (Fig. 36).

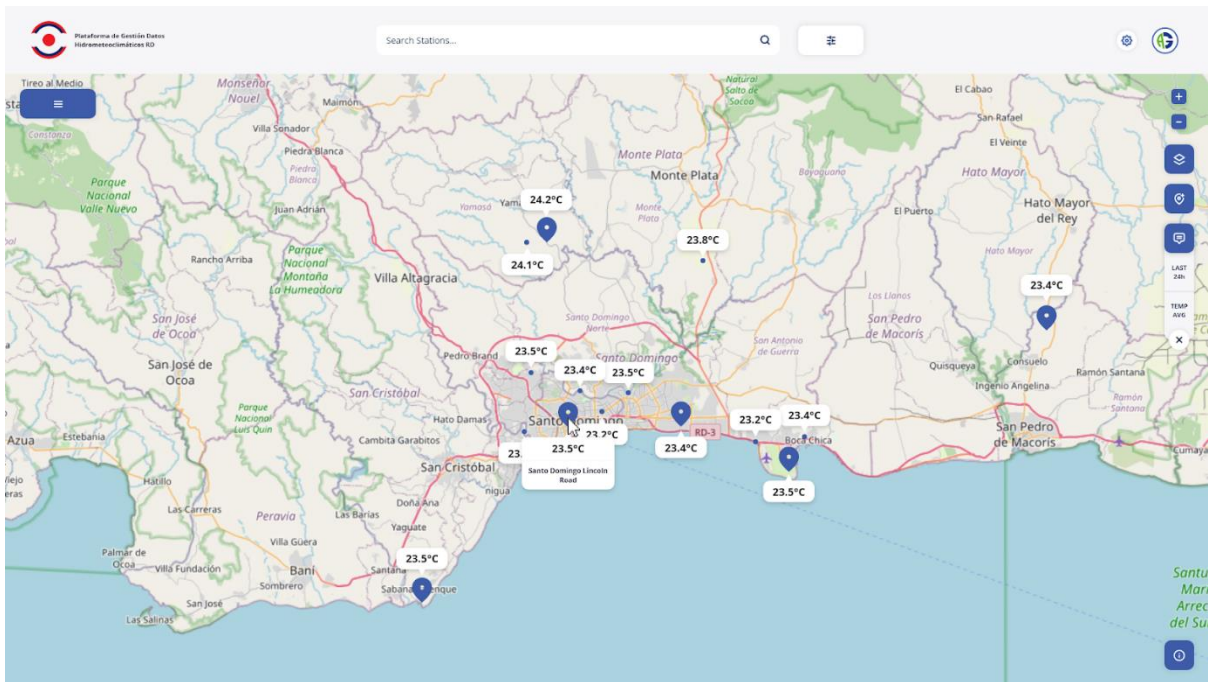


Figura 36

Cuando el usuario hace clic en una estación meteorológica, la plataforma abre un nuevo cuadro de diálogo en el que se muestran todos los datos de la estación según la configuración (Fig. 37). El diálogo muestra el nombre de la estación, la última actualización y datos que pueden filtrarse por tiempo. El usuario puede cerrar el cuadro de diálogo o acceder a la página de detalles del Informe de Estación.

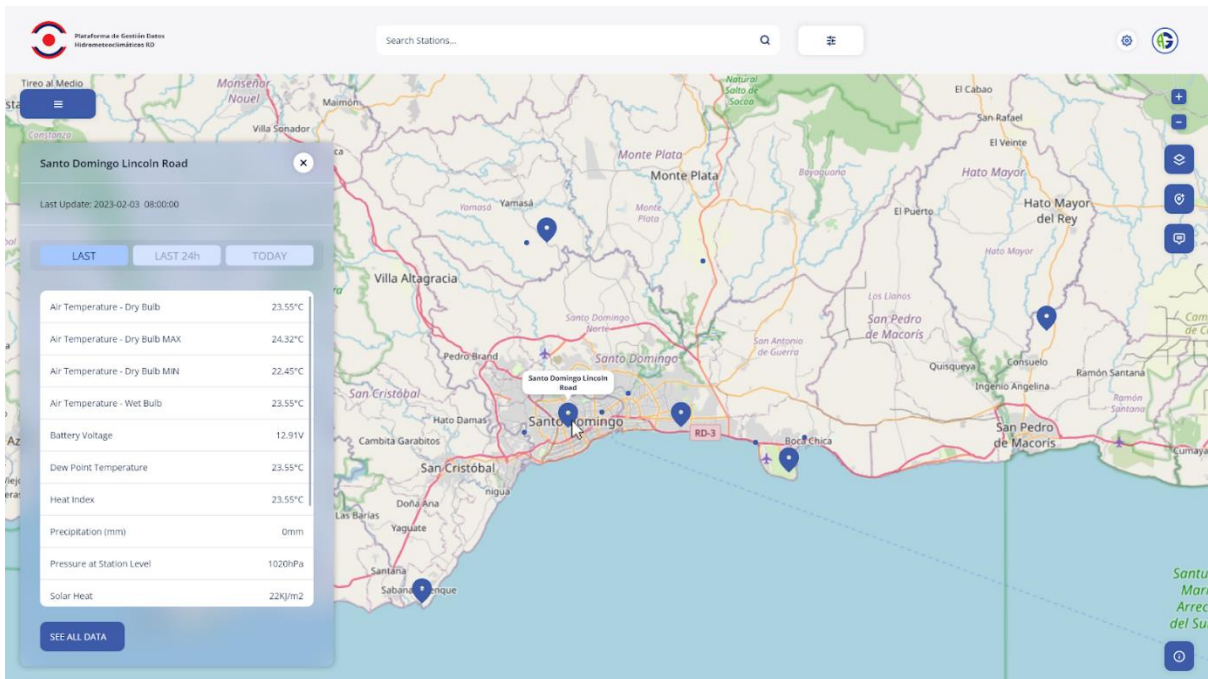


Figura 37

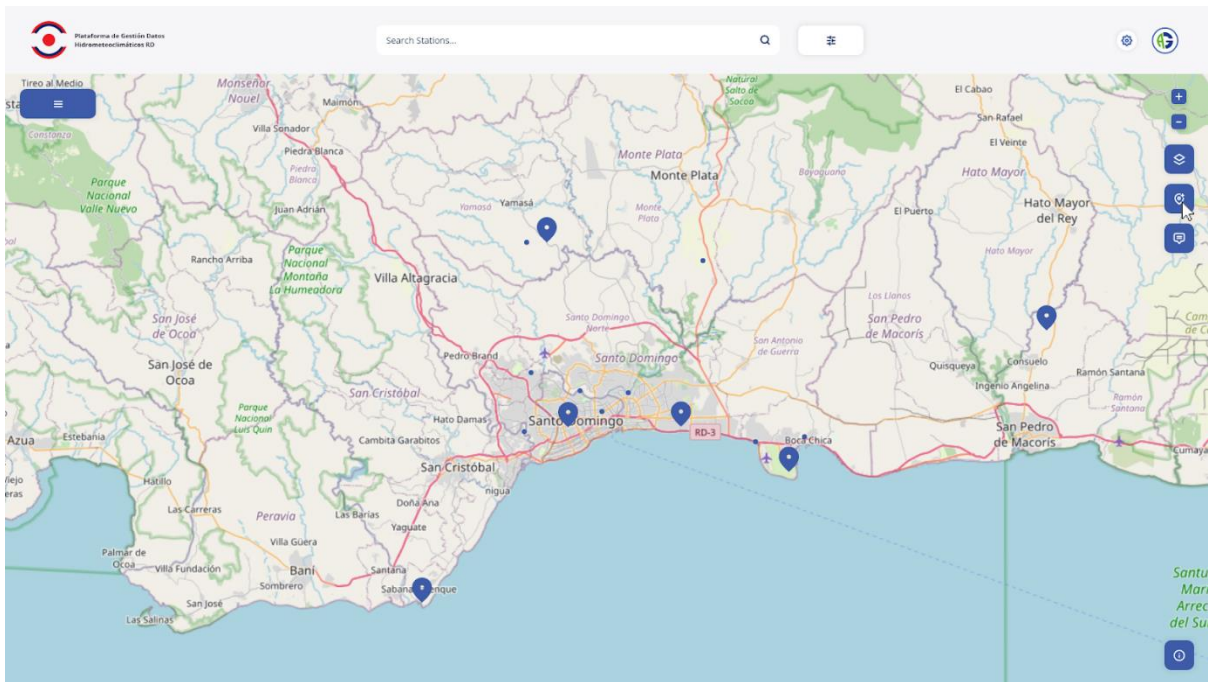


Figura 38

A partir de la investigación realizada durante la fase de diseño, se constató que un usuario podría necesitar seleccionar los datos de algunas estaciones que están agrupadas de una manera particular (es decir, que no responden a agrupaciones estándar como Estado, Provincia, etc.). Para ello se ha diseñado una nueva función denominada "Selección múltiple" (de **Fig. 38** a **Fig. 40**).

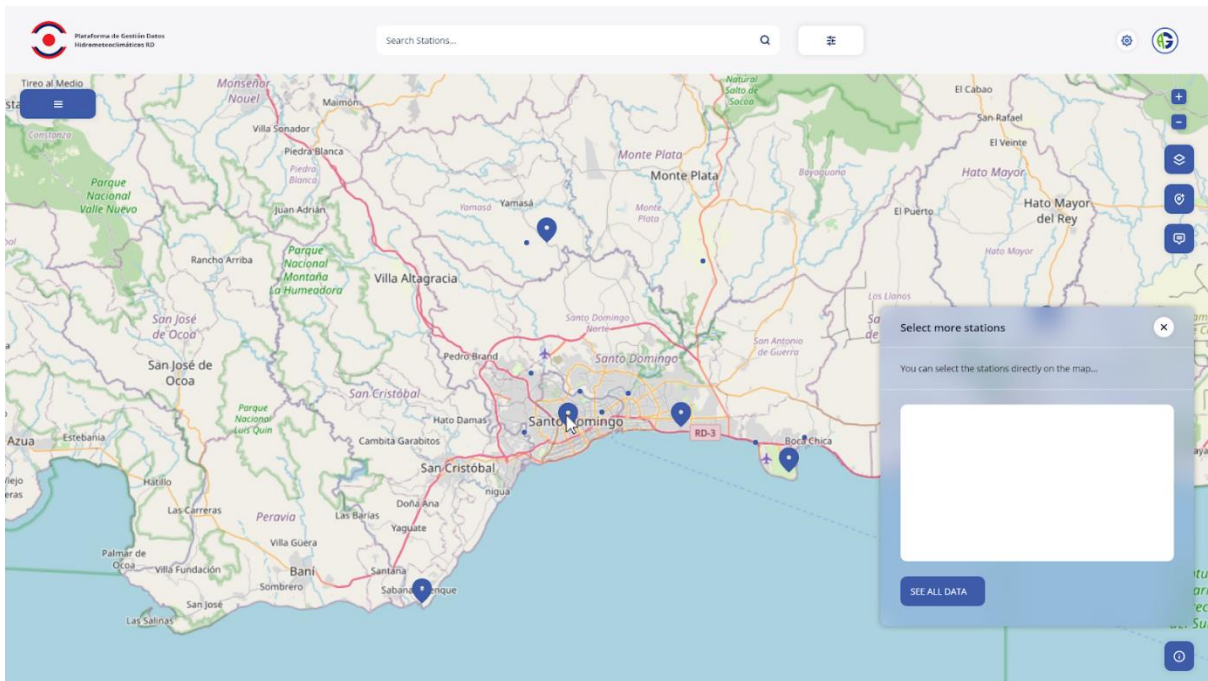


Figura 39



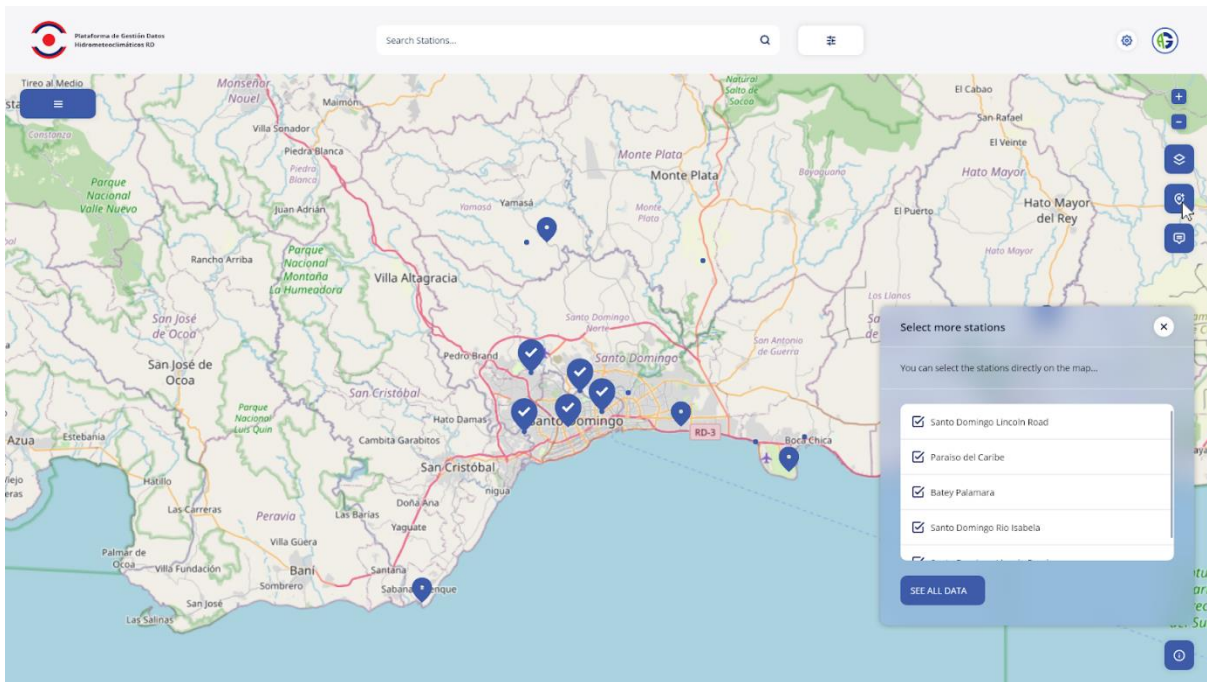


Figura 40

El usuario puede pulsar el botón de función y empezar a hacer clic en las estaciones. En este caso, al hacer clic en las estaciones no se abre un cuadro de diálogo, sino que simplemente se añade la estación sobre la que se ha hecho clic a la lista (Fig. 40).

Una vez seleccionadas todas las estaciones deseadas, el usuario puede cerrar la función cancelándola o hacer clic en "Ver todos los datos" y será redirigido a la sección de comparación de estaciones con las estaciones ya listadas.

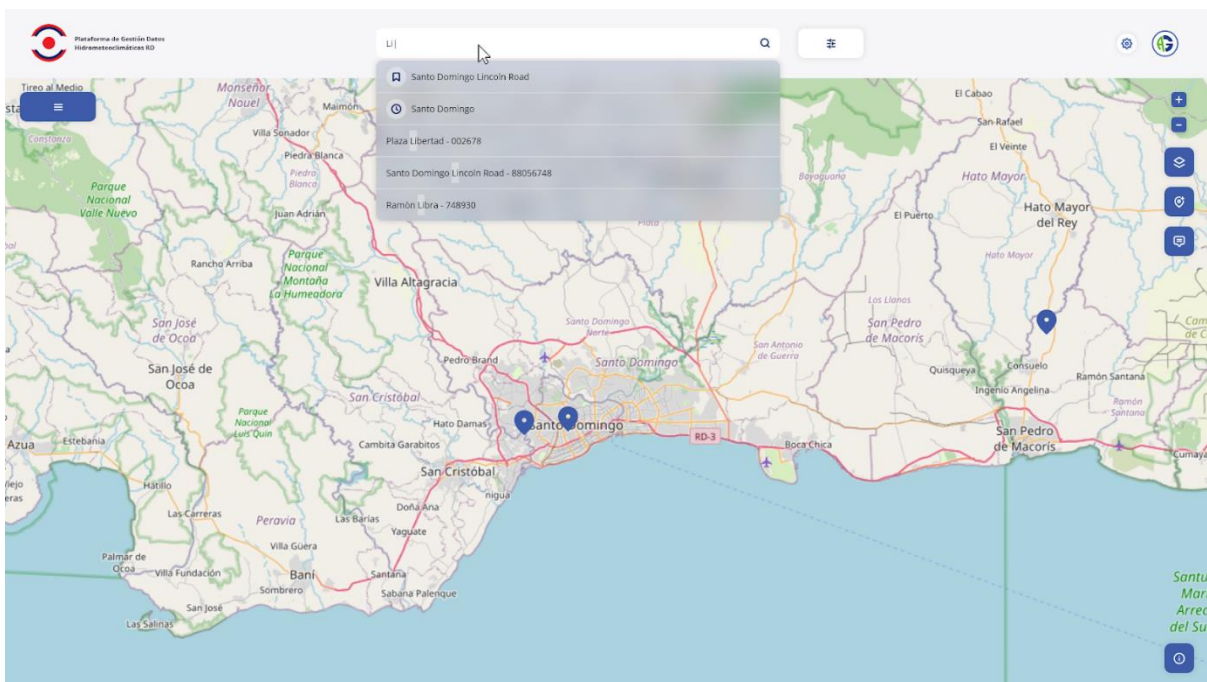


Figura 41

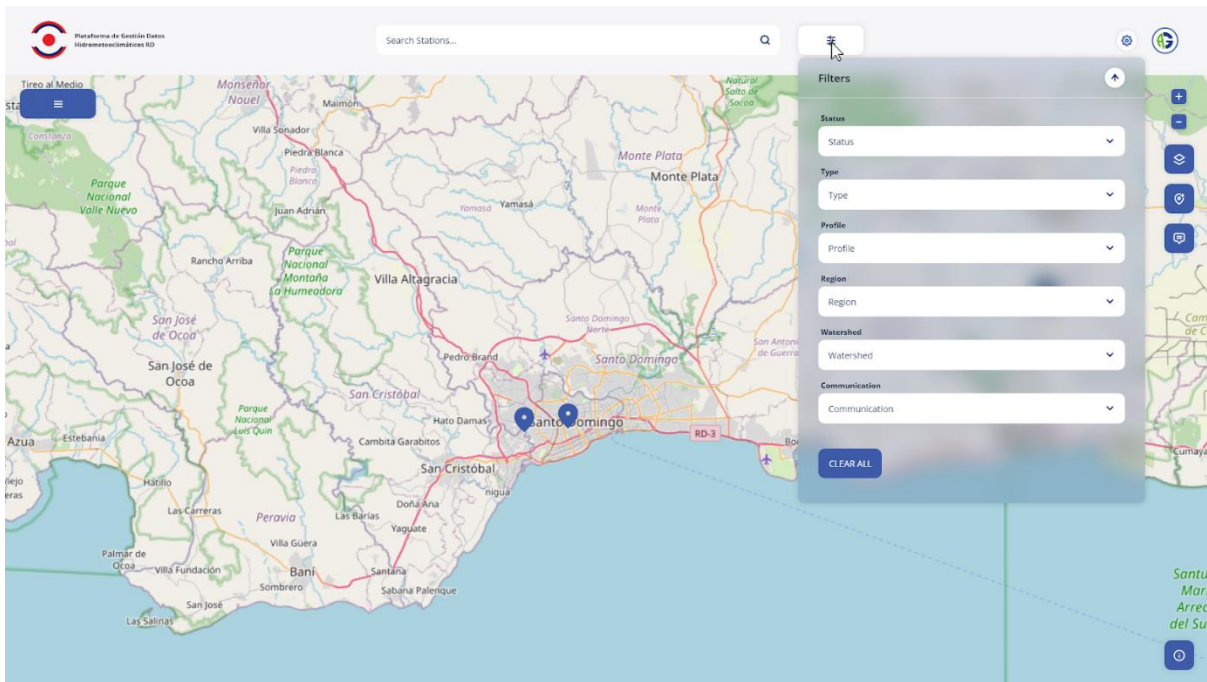


Figura 42

Además, el usuario puede filtrar la búsqueda mediante filtros de búsqueda (**Fig. 42**). A medida que se seleccionan, las estaciones meteorológicas que no cumplen los requisitos se ocultan en el mapa.

Finalmente, como última función de la Vista del Mapa, vemos cómo el usuario puede buscar una estación. Al hacer clic en la barra de búsqueda y empezar a escribir, la plataforma muestra inmediatamente resultados compatibles (**Fig. 41**), incluidos los elementos guardados en "Tus accesos directos" y los elementos del historial de búsqueda.

## 6.4.2.2 INFORME DE LA ESTACIÓN

Con la función "Informe de estación", el usuario puede buscar una estación concreta y acceder a todos los datos relacionados con ella.

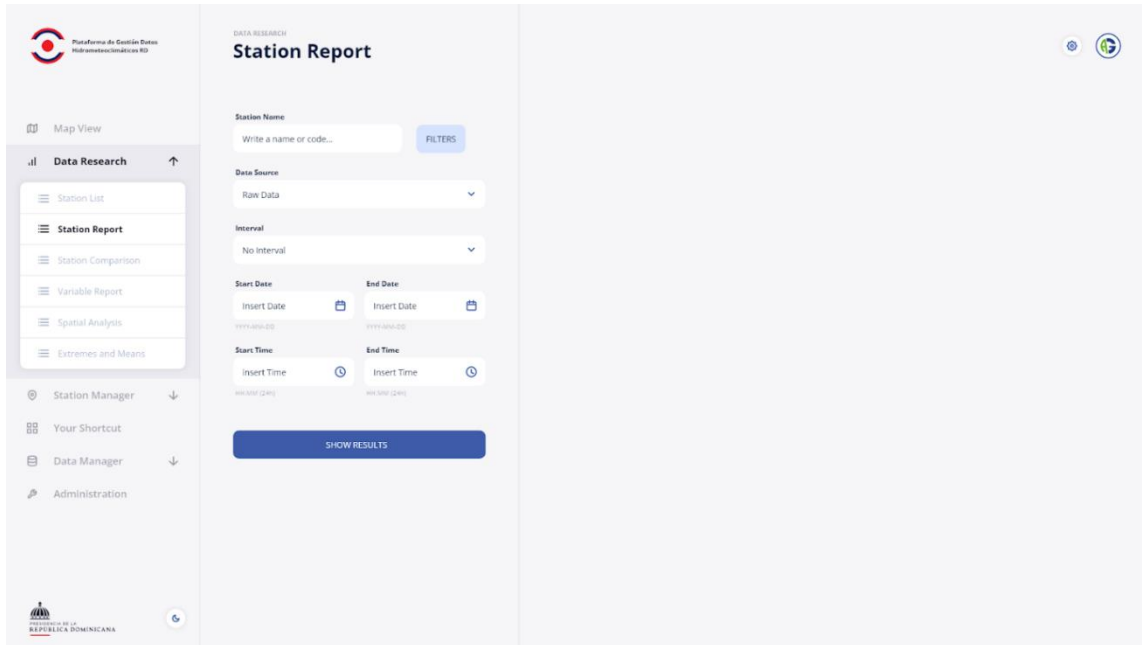


Figura 43

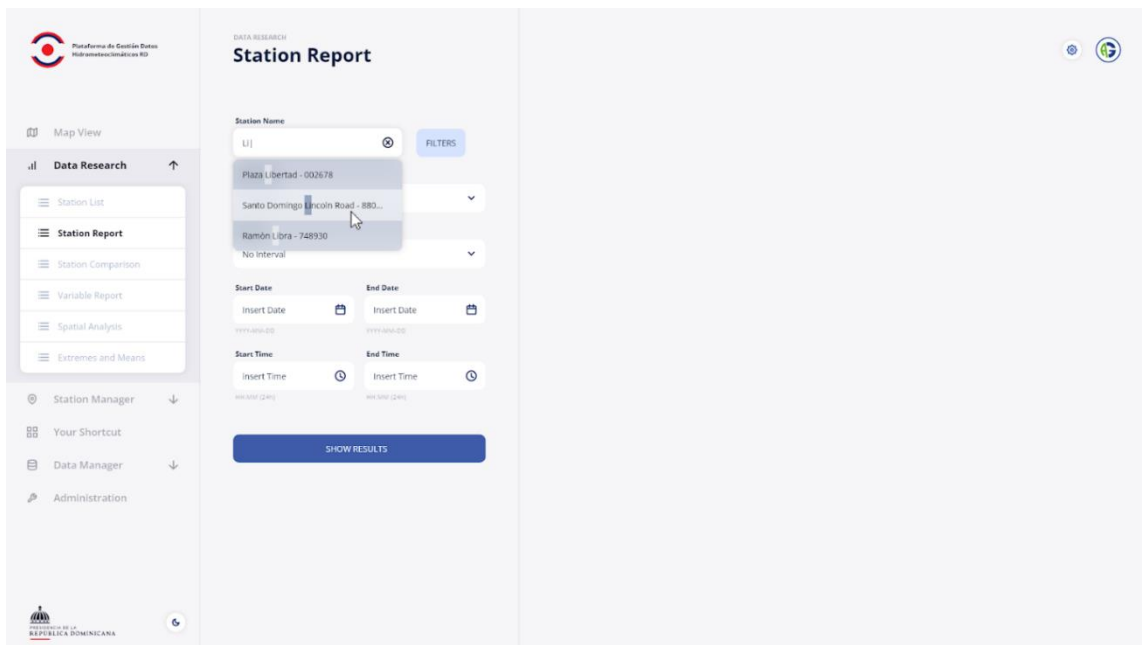


Figura 44

A la izquierda de la pantalla se encuentra el "cajón de navegación", con todos los elementos del menú y el nombre de la página correspondiente resaltado. Para facilitar

la navegación del usuario dentro de la plataforma, en la parte superior de la pantalla hay unas migas de pan que muestran la cabecera de la página (**Fig. 43**).

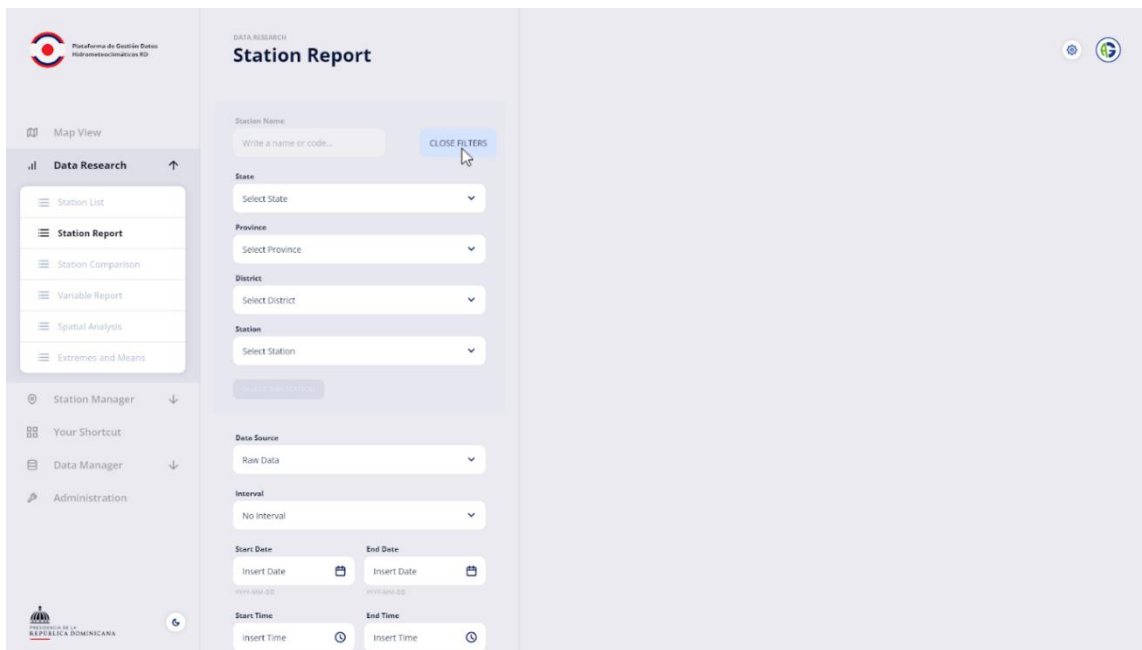


Figura 45

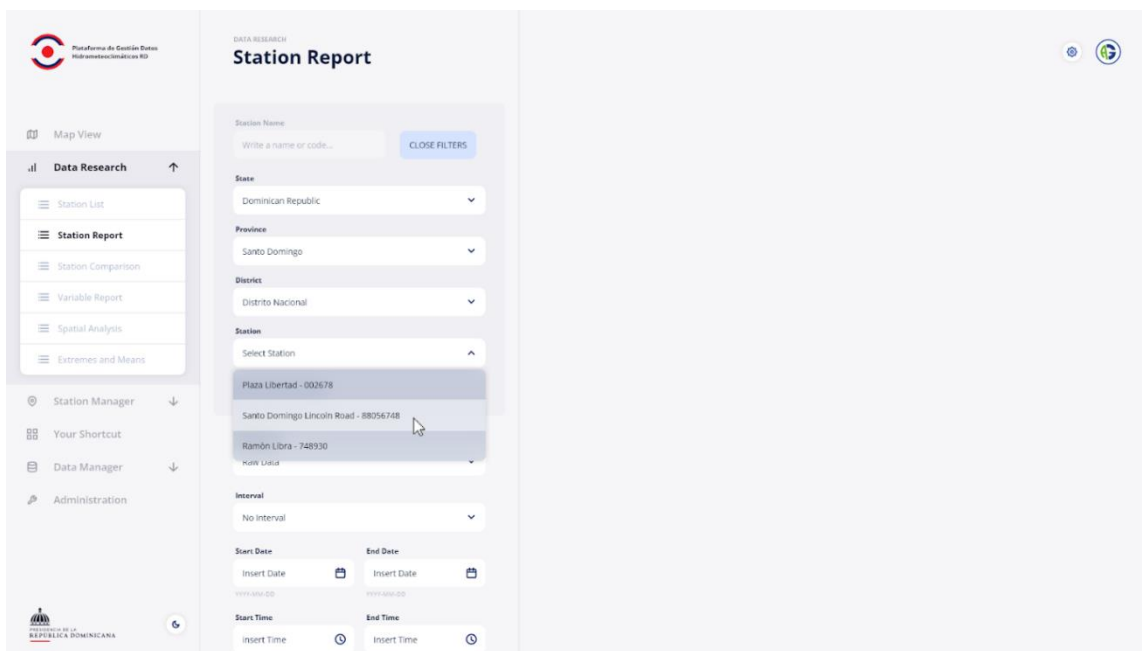


Figura 46

El usuario puede buscar el nombre o el código de la estación en la barra de búsqueda y, a continuación, especificar los demás datos (**Fig. 44**). Además, si el usuario necesita encontrar una estación sin conocer el nombre o el código, puede utilizar la función "Filtros" (**Fig. 45**). Al hacer clic en el botón "Filtros" se despliega una sección en la que el usuario puede introducir filtros para localizar la estación meteorológica concreta (**Fig. 46**).

Una vez que el usuario ha seleccionado la estación, ya sea mediante una búsqueda directa o utilizando filtros, puede ver los datos pulsando el botón "Mostrar resultados" (Fig. 47).

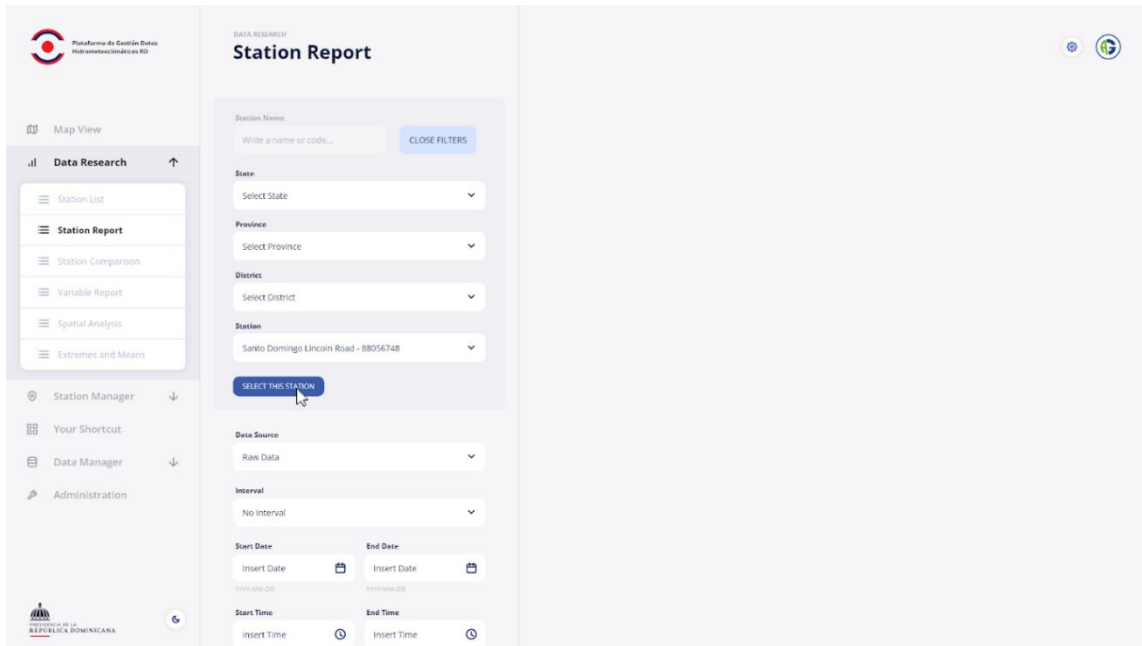


Figura 47

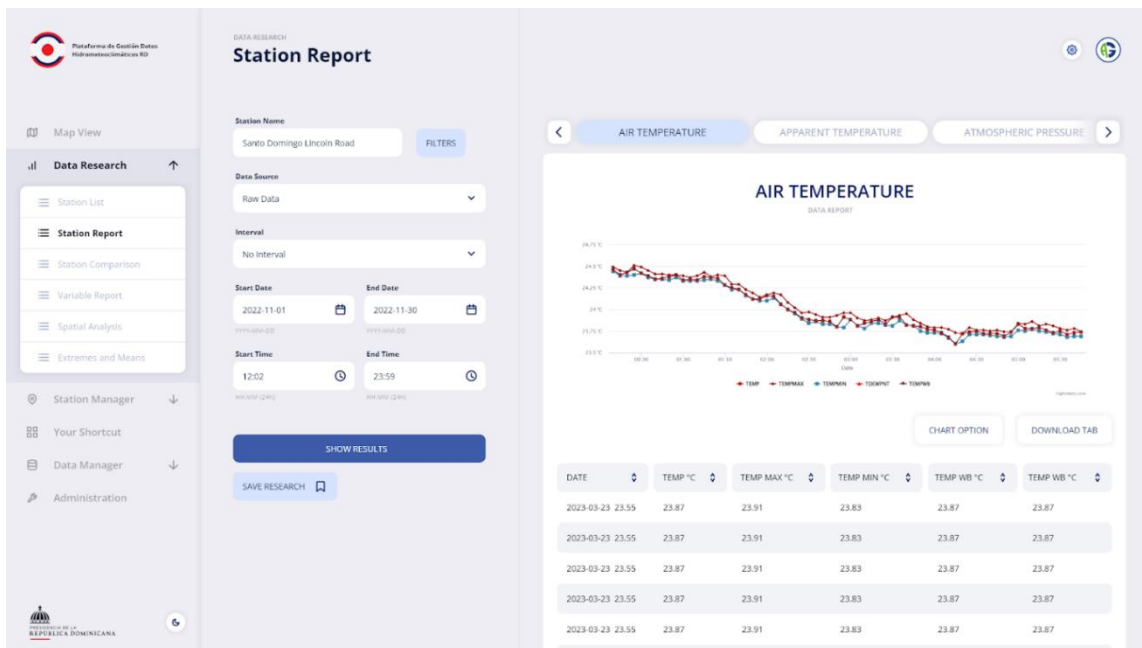


Figura 48

En esta pantalla, el usuario tiene acceso a toda la base de datos de la estación meteorológica elegida (Fig. 48). Los datos se muestran en la parte izquierda de la pantalla mediante gráficos y tablas. En la parte superior, el usuario puede elegir el tipo de datos que desea visualizar. En la sección en la que el usuario ha introducido los datos

para localizar la estación meteorológica, se añade un botón para guardar la búsqueda en la sección "Tus atajos".

Los gráficos y tablas pueden descargarse en distintos formatos, como se describe en los capítulos anteriores (**Fig. 49**).

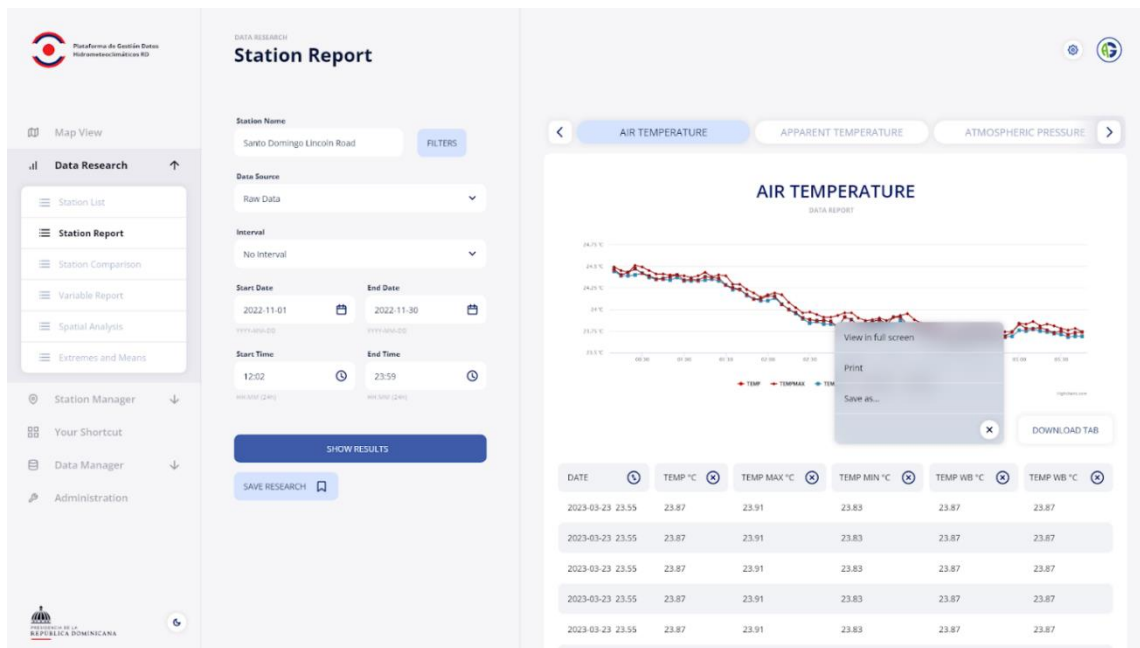


Figura 49

### 6.4.2.3 LISTA DE ESTACIONES

La sección "Lista de estaciones" ofrece una visión completa de las estaciones en formato tabular (Fig. 50).

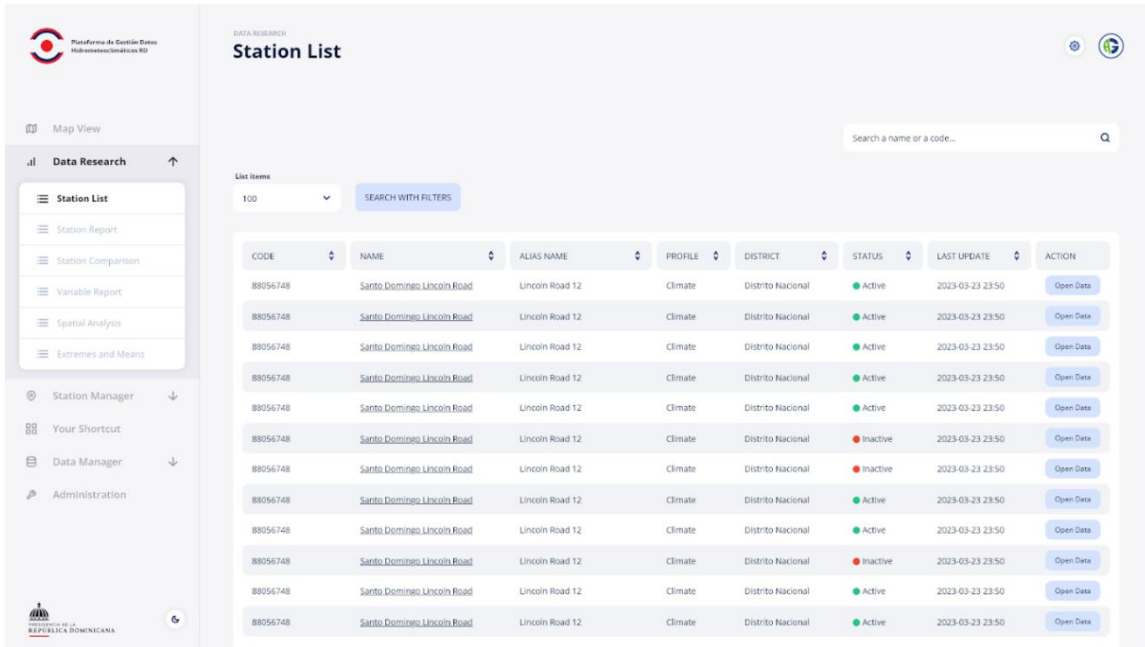


Figura 50

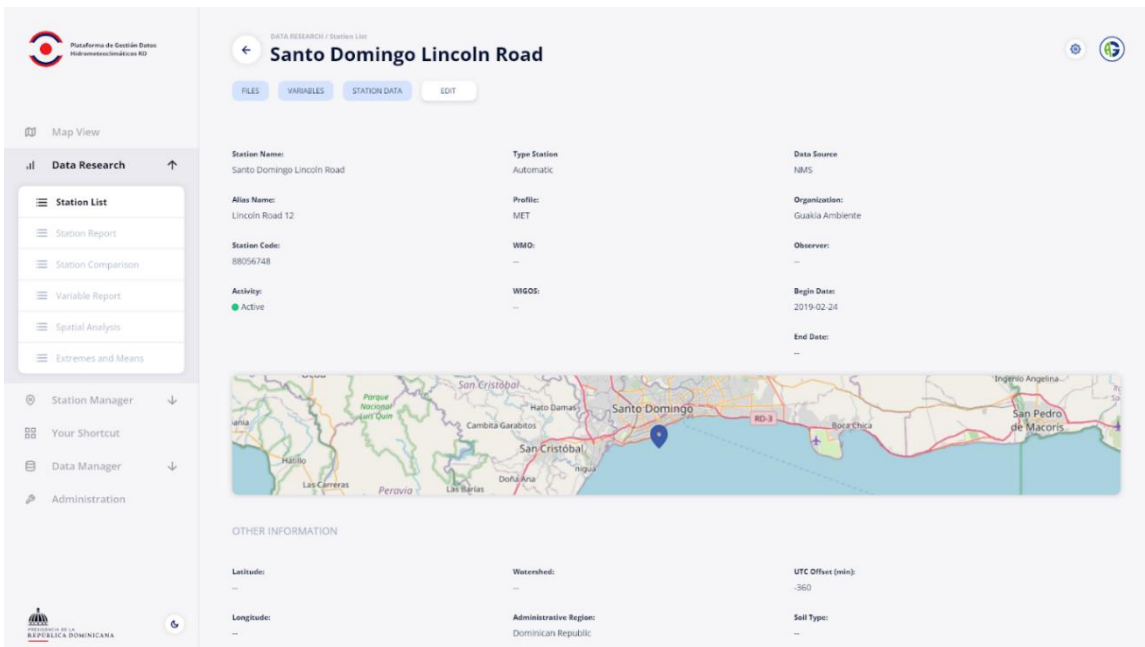


Figura 51

El usuario puede ver la lista de estaciones en la pantalla, organizadas en una tabla que muestra los datos clave. Entre estos datos, resulta especialmente útil mostrar el estado actual y la última actualización. Es posible ordenar la lista según los datos que figuran en

la parte superior de la tabla, como código, nombre, alias, perfil, distrito, estado y última actualización.

El nombre de la estación, subrayado, está vinculado a los metadatos de la estación hidrometeorológica (**Fig. 51**), mientras que la CTA (*Call to Action*) de cada fila de la tabla muestra los datos meteorológicos de la estación (**Fig. 50**).

Platforma de Gestión Datos Hidrometeorológicos RD

DATA RESEARCH

### Station List

Search a name or a code...

List Items: 100

Profile | District | Status | Last Update | SAVE RESEARCH

CODE	NAME	ALIAS NAME	PROFILE	DISTRICT	STATUS	LAST UPDATE	ACTION
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data
88056748	<u>Santo Domingo Lincoln Road</u>	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Open Data

Figura 52

Además, el usuario tiene la opción de utilizar filtros para obtener una visión más específica de la lista según sus necesidades. Una vez configurados, los filtros pueden guardarse en "Sus accesos directos" (**Fig. 52**).



## 6.4.2.4 COMPARACIÓN DE ESTACIONES

La función "Comparación de estaciones" permite al usuario ver simultáneamente los datos de varias estaciones y seleccionarlos mediante una lista gestionable por filtros.

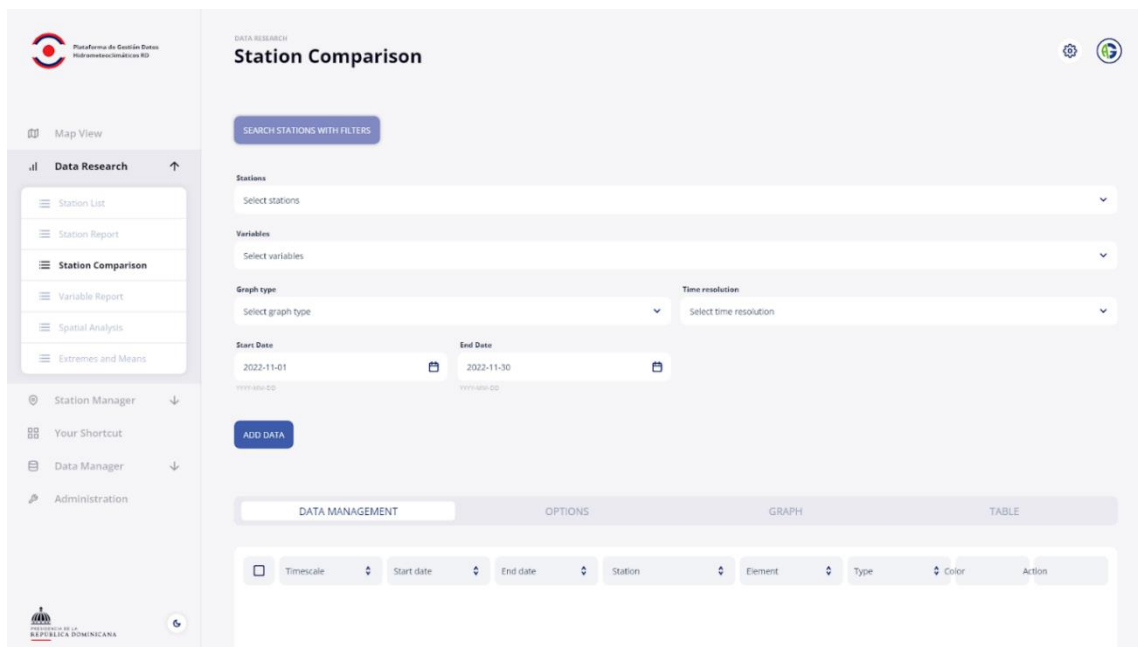


Figura 53

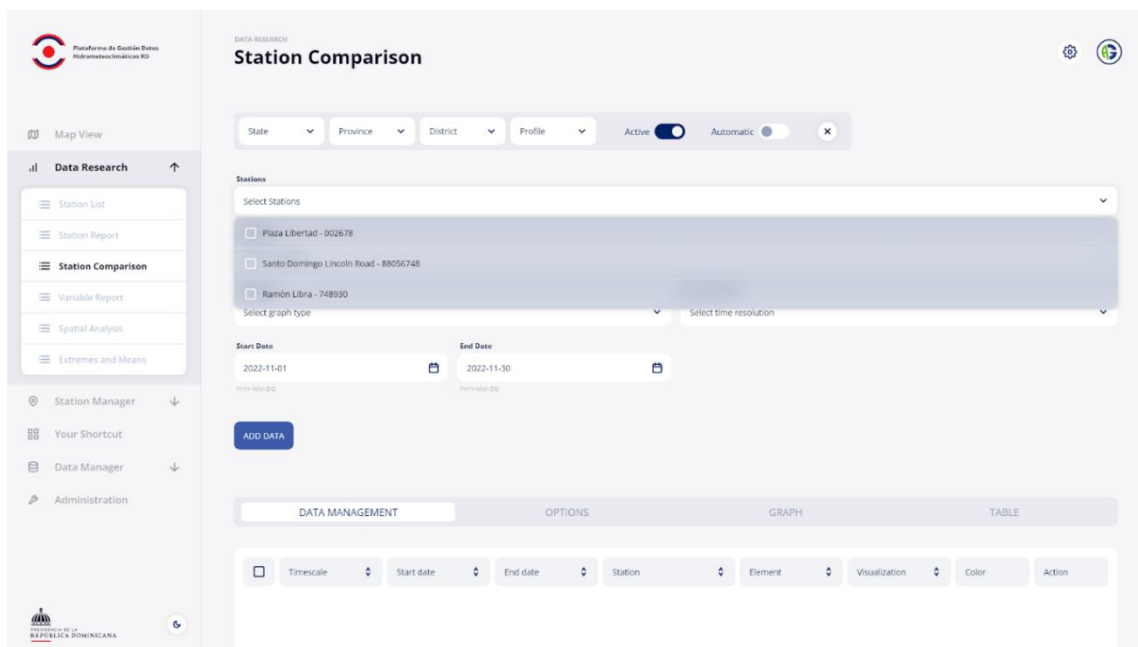
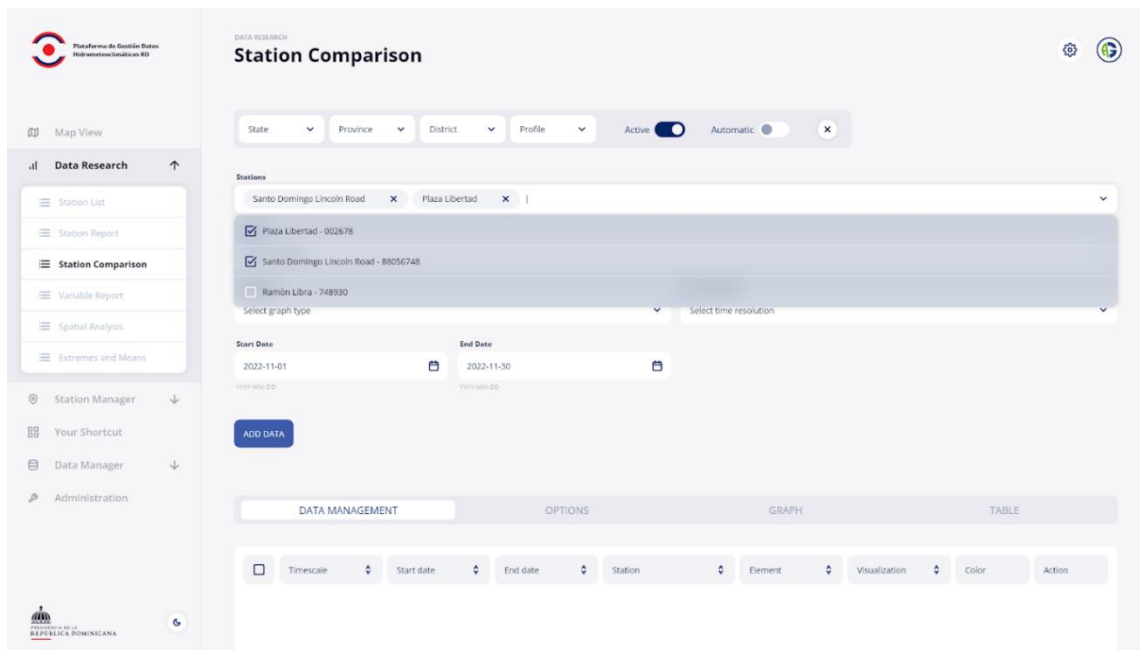


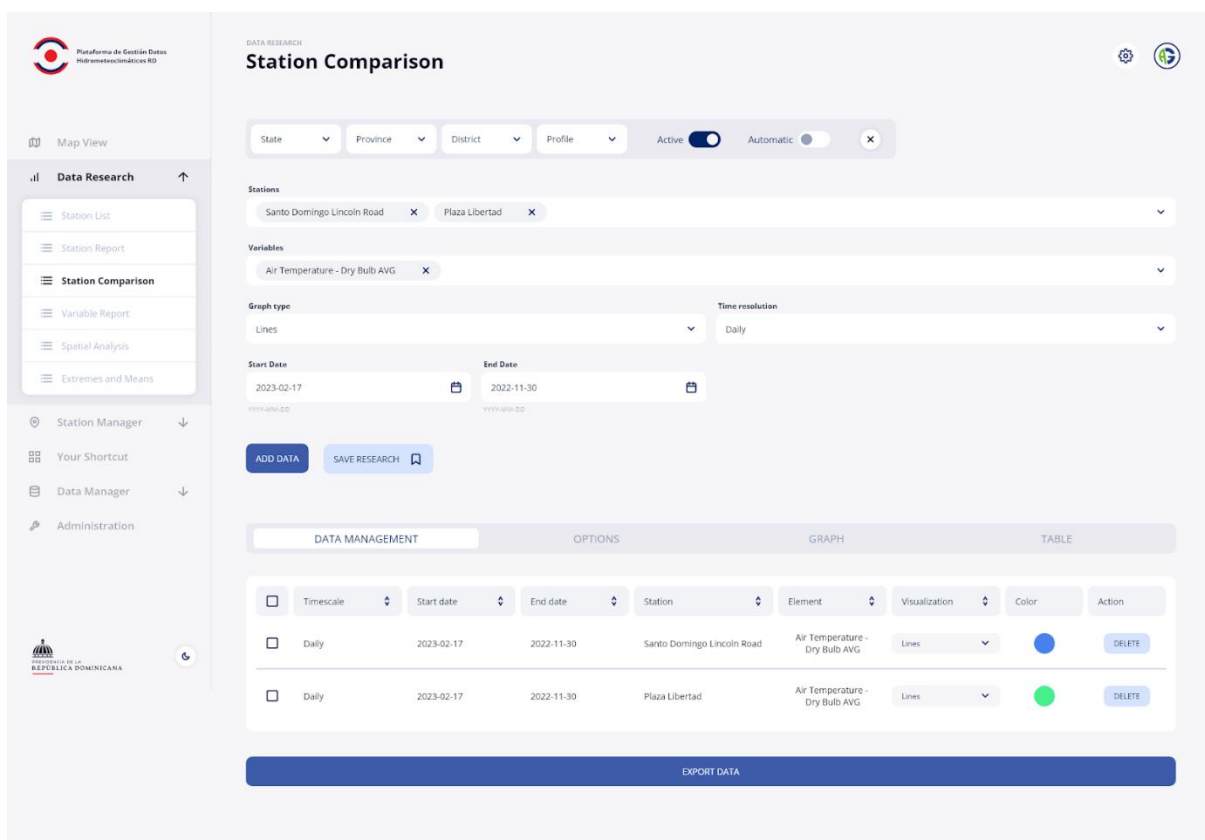
Figura 54

En la captura de pantalla que se muestra en la **Figura 53**, puede ver que puede hacer clic en "Estaciones" para abrir la lista completa de todas las estaciones disponibles. Como

la lista puede ser muy larga, se han incluido filtros de búsqueda para facilitar la selección, como se muestra en la **Figura 54**.



**Figura 55**



**Figura 56**

Una vez que el usuario ha seleccionado todas las estaciones deseadas, debe rellenar el resto del formulario y hacer clic en "Añadir datos", como se muestra en la **Figura 55**.

A partir de ese momento, el usuario puede personalizar los parámetros de visualización de los datos y ver gráficos y tablas como se muestra en la **Figura 56**.

## 6.4.2.5 INFORME DE VARIABLES

En la sección "Informe de variables", el usuario puede ver los datos de varias variables medidas en varias estaciones. El proceso es similar al de la sección "Comparación de estaciones", pero en este caso existe la posibilidad de introducir varias variables simultáneamente (Fig. 57 y Fig. 58).

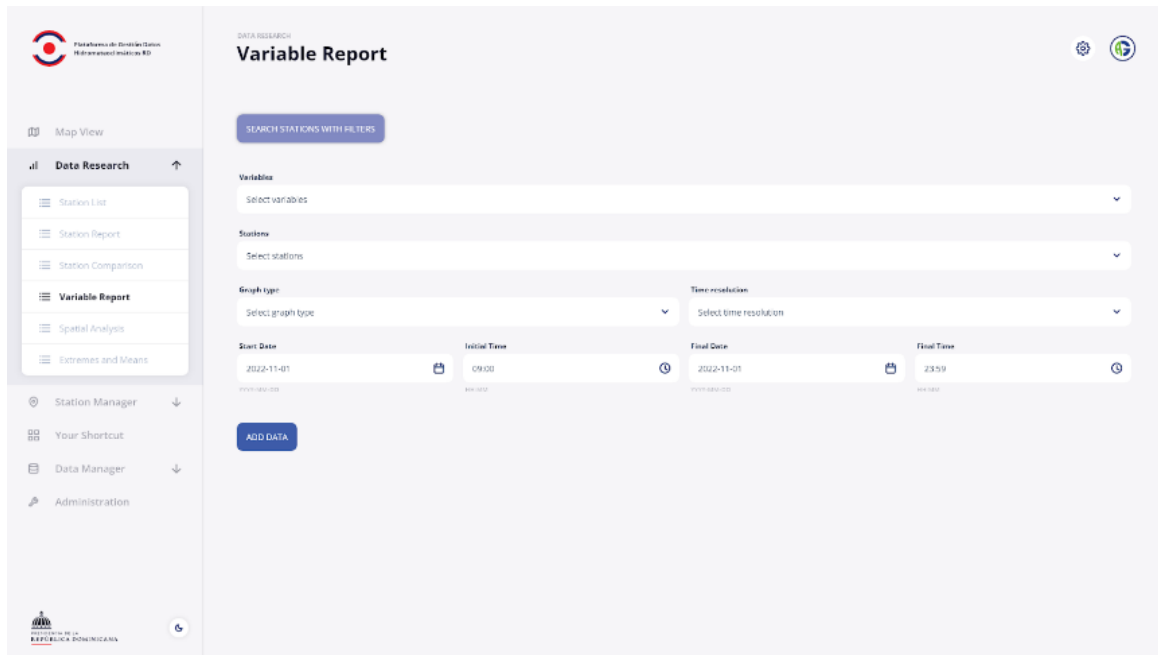


Figura 57

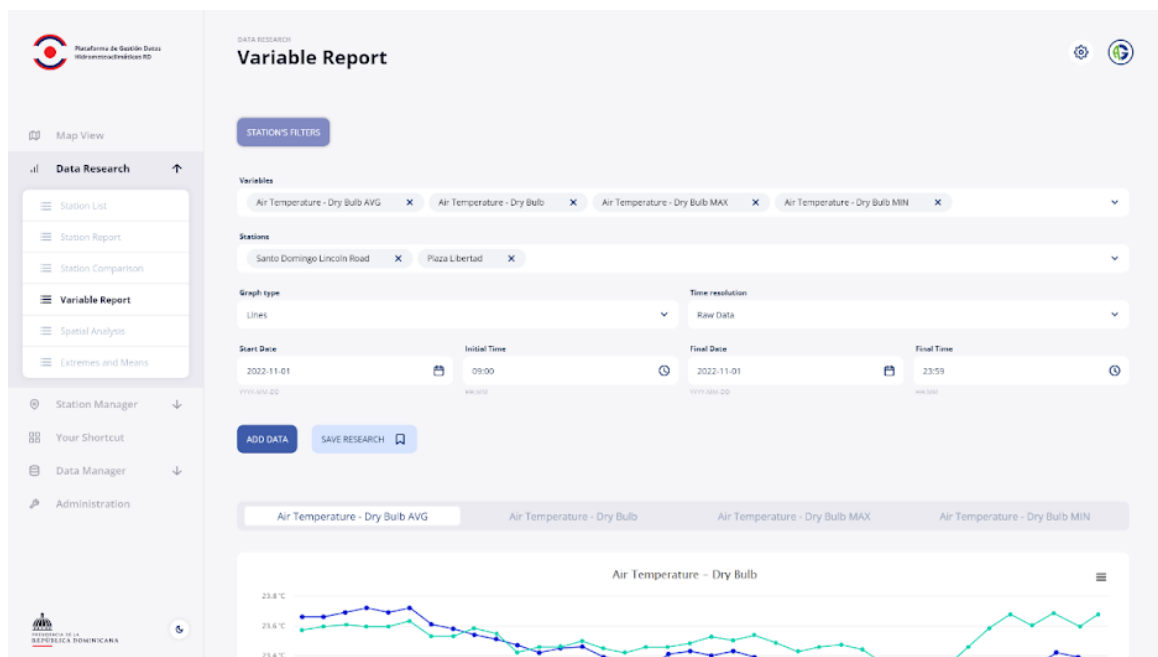


Figura 58

## 6.4.2.6 ANÁLISIS ESPACIAL

La sección "Análisis espacial" ofrece a los usuarios la posibilidad de visualizar los datos gráficamente en un mapa, utilizando colores para una representación visual. Estos mapas pueden exportarse e imprimirse, al igual que las tablas y los gráficos.

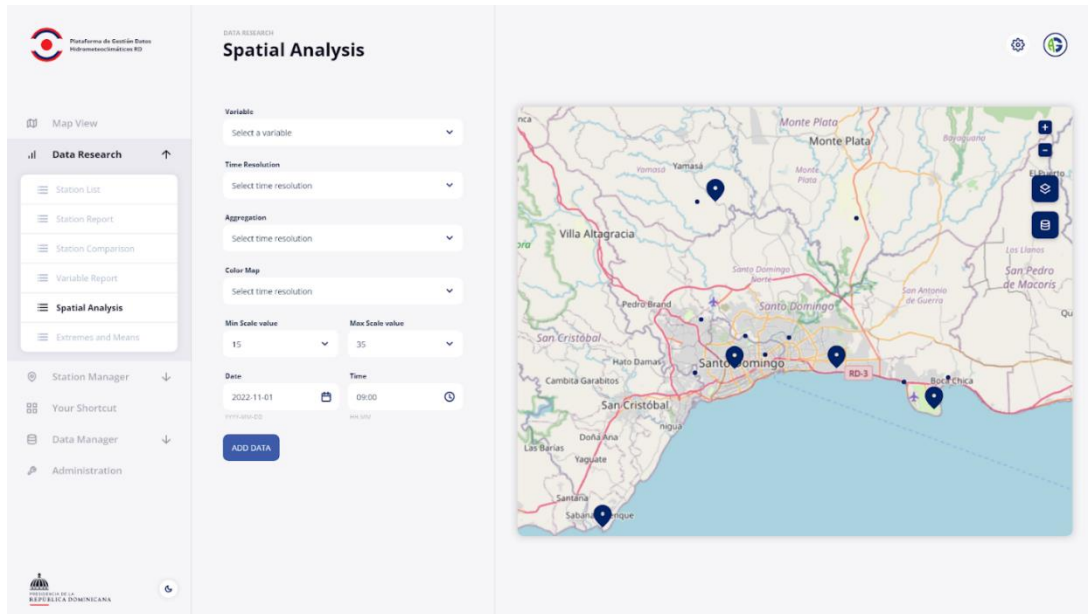


Figura 59

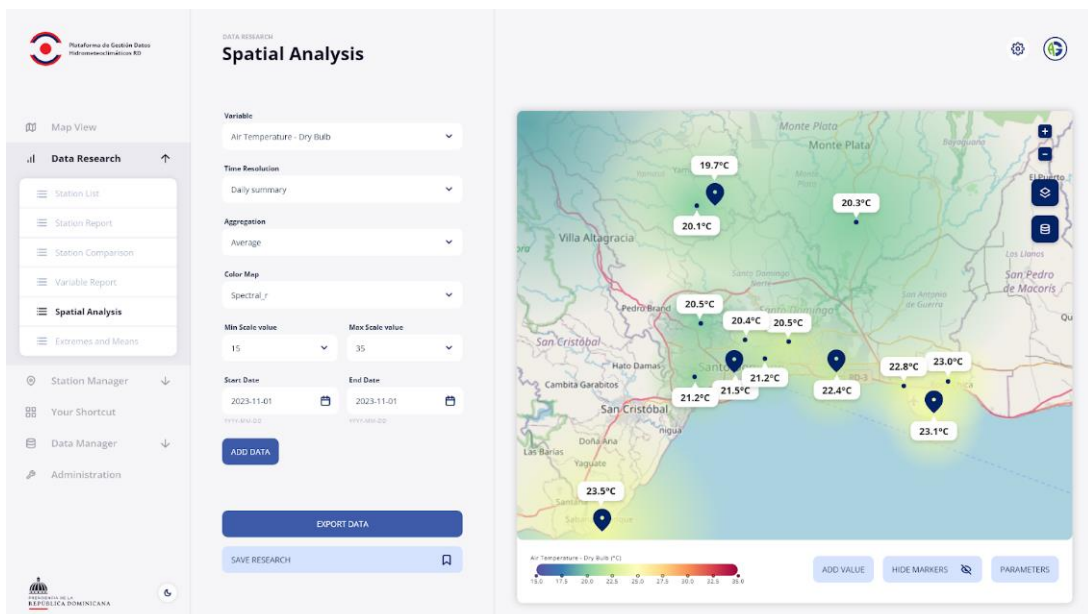


Figura 60

Los usuarios tienen control sobre el tipo de datos que se mostrarán y el método de agregación utilizado (Fig. 59 y Fig. 60). Además, podrán elegir entre distintos tipos de mapas y colores, para obtener archivos que respondan a sus necesidades específicas.

## 6.4.2.7 GESTOR DE ACCESOS DIRECTOS

La sección Gestor de accesos directos está diseñada para ofrecer a los usuarios una herramienta similar a los marcadores del navegador. A menudo, quienes trabajan o estudian pueden necesitar consultar la misma página de una plataforma en varias ocasiones.

Para evitar la repetición de búsquedas cada vez que el usuario necesite consultar determinados datos, ofrecemos la posibilidad de guardar búsquedas y páginas de interés para el usuario (**Fig. 61**).

Los accesos directos estarán disponibles directamente en el menú de navegación, con lo que el acceso a las páginas guardadas será más rápido y cómodo.

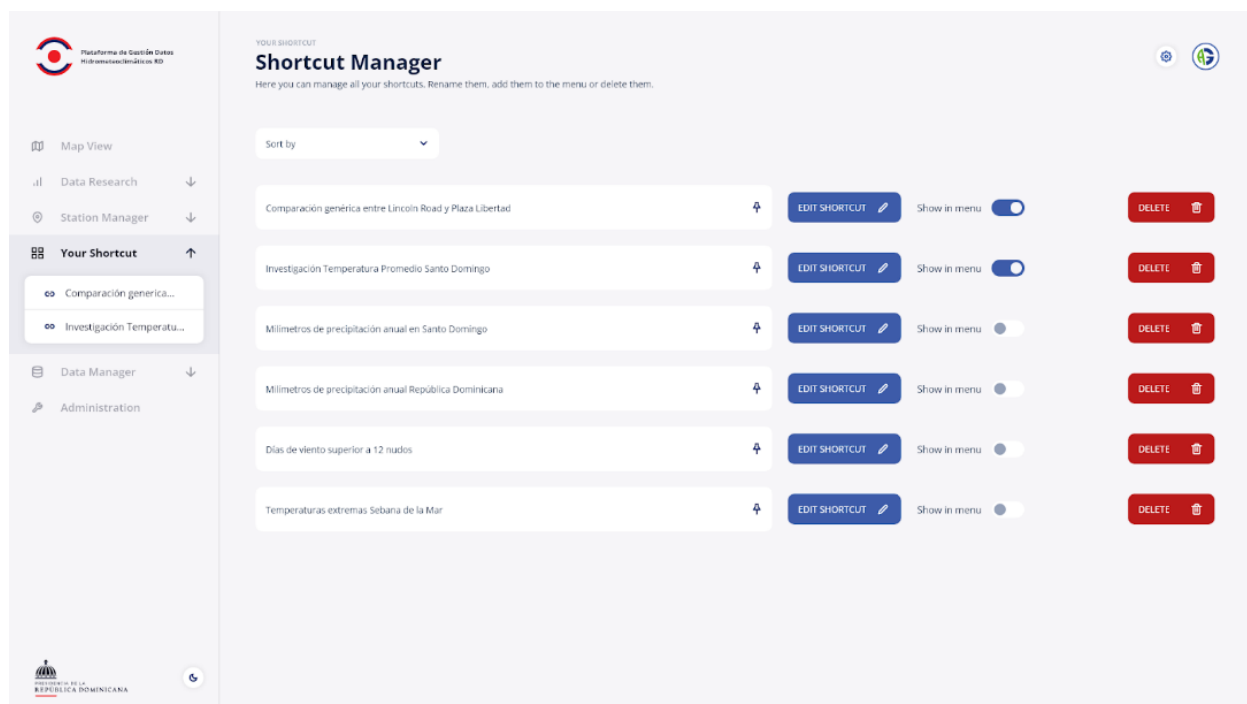


Figura 61

## 6.4.2.8 JEFE DE ESTACIÓN

La sección "Gestor de estaciones" permite a los usuarios consultar y gestionar las estaciones meteorológicas de su propiedad, siempre que dispongan de los permisos necesarios.

CODE	NAME	ALIAS NAME	PROFILE	DISTRICT	STATUS	LAST UPDATE	ACTION
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Inactive	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt
88056748	Santo Domingo Lincoln Road	Lincoln Road 12	Climate	Distrito Nacional	Active	2023-03-23 23:50	Edt

Figura 62

En la primera pantalla, el usuario puede ver todas las estaciones propiedad de la organización de productores con la que está acreditado (Fig. 62), con la posibilidad de editarlos pulsando el botón "Editar" (Fig. 64) o pulsando sobre el nombre para acceder a la vista de metadatos (Fig. 63).

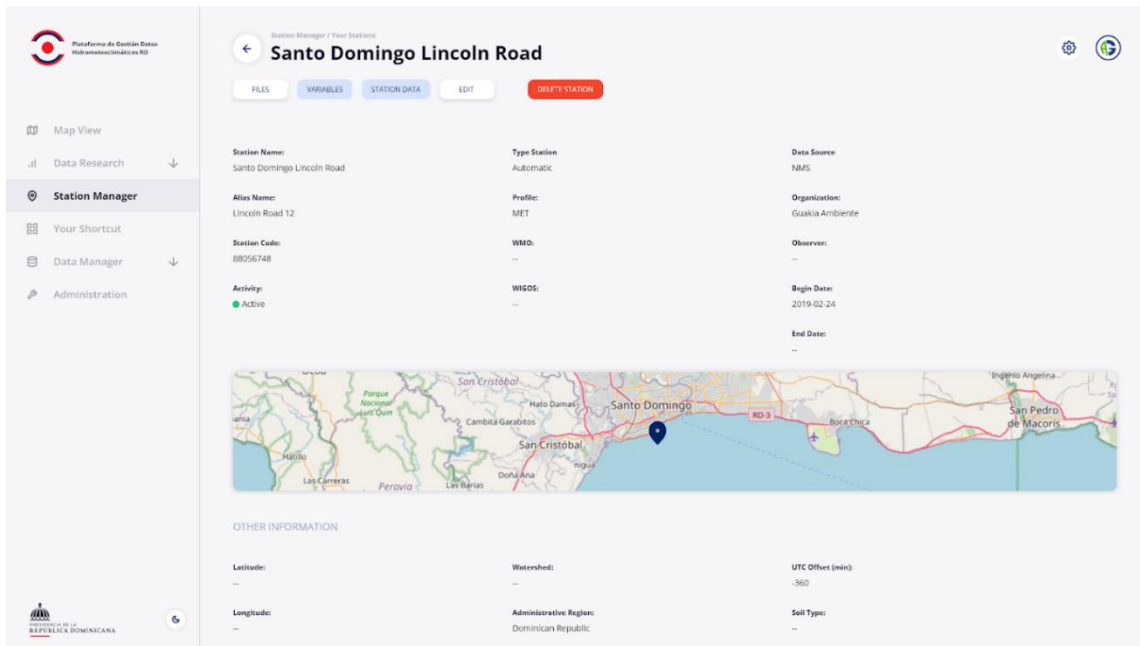


Figura 63

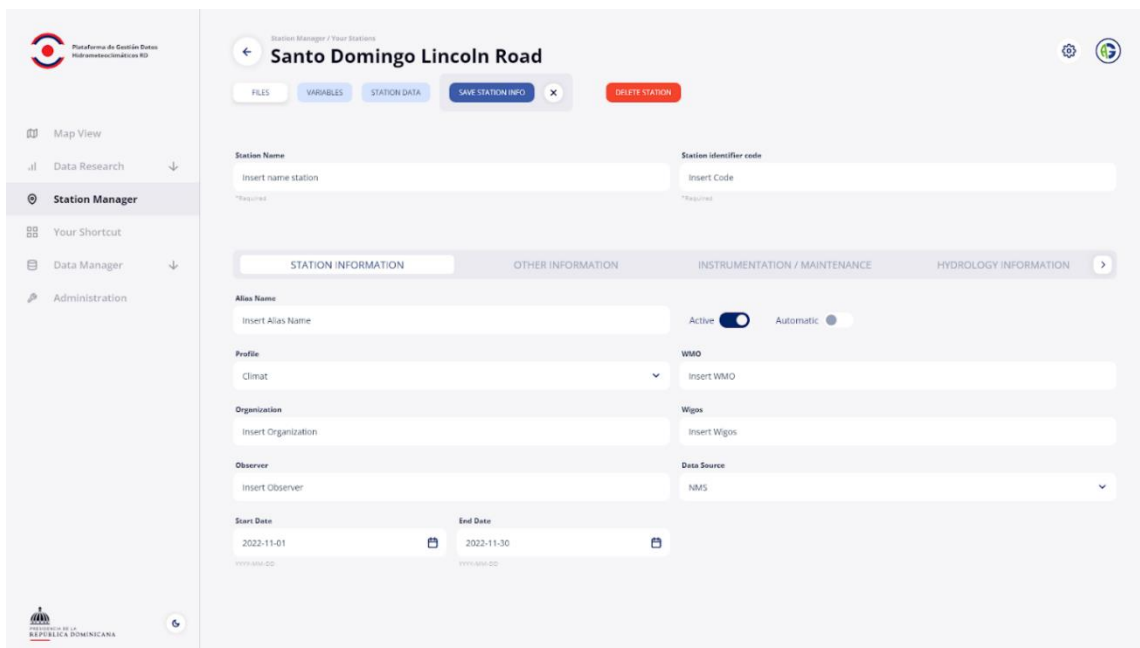


Figura 64

Además, el usuario tiene la opción de gestionar variables específicas de la estación (Fig. 65 y Fig. 66).



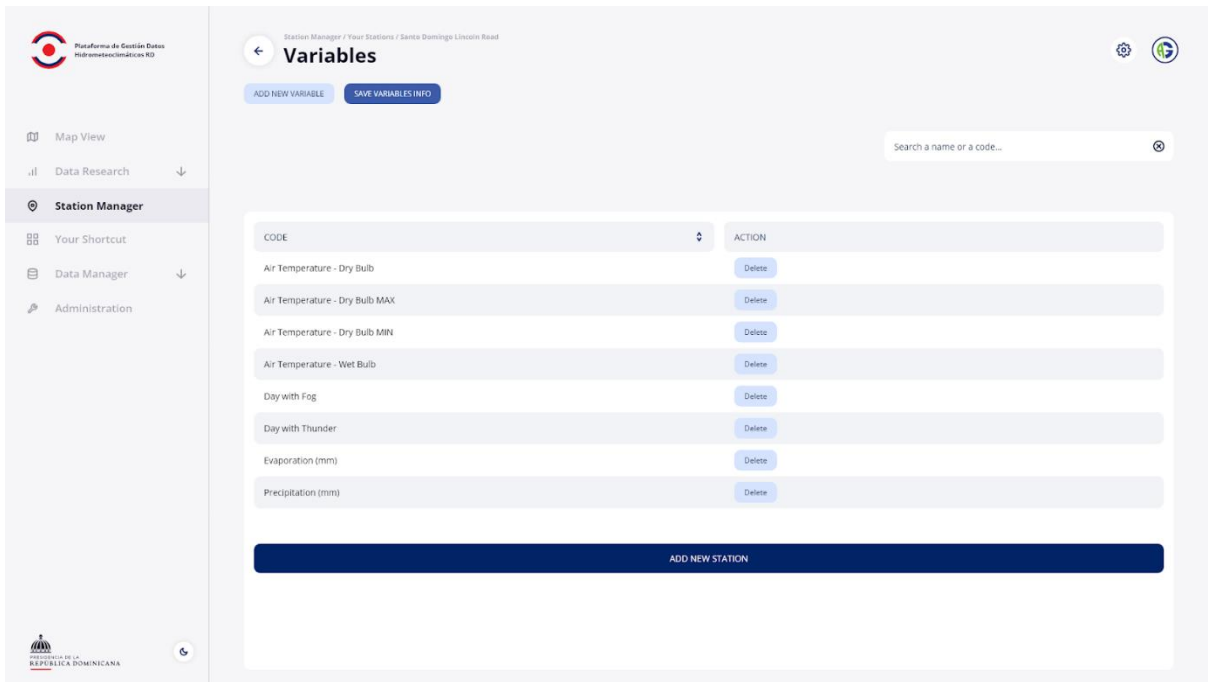


Figura 65

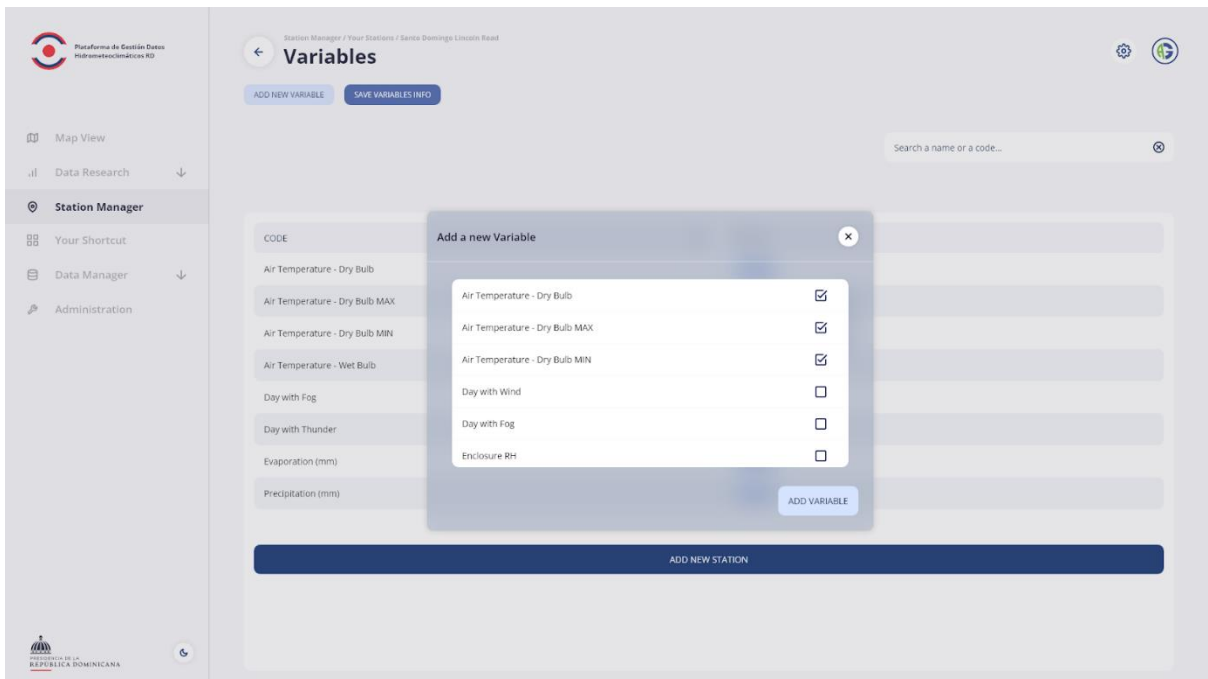


Figura 66

### 6.4.3 ACCESIBILIDAD

La accesibilidad se refiere a la capacidad de los sistemas y productos para prestar servicios y proporcionar información utilizable por todos los usuarios, sin discriminación, incluidos los que tienen capacidades diferentes. Garantizar el acceso a datos y servicios es crucial, especialmente cuando se trata de servicios públicos. Este concepto está respaldado por varias normas, entre ellas las Pautas de Accesibilidad al Contenido en la Web (WCAG) de 2018 del Consorcio World Wide Web (W3C).

Un aspecto importante de la accesibilidad se refiere a la posibilidad de acceso por parte de usuarios que utilizan distintos dispositivos y tamaños de pantalla. En este contexto, es esencial diseñar layouts responsivos y versiones móviles bien diseñadas.

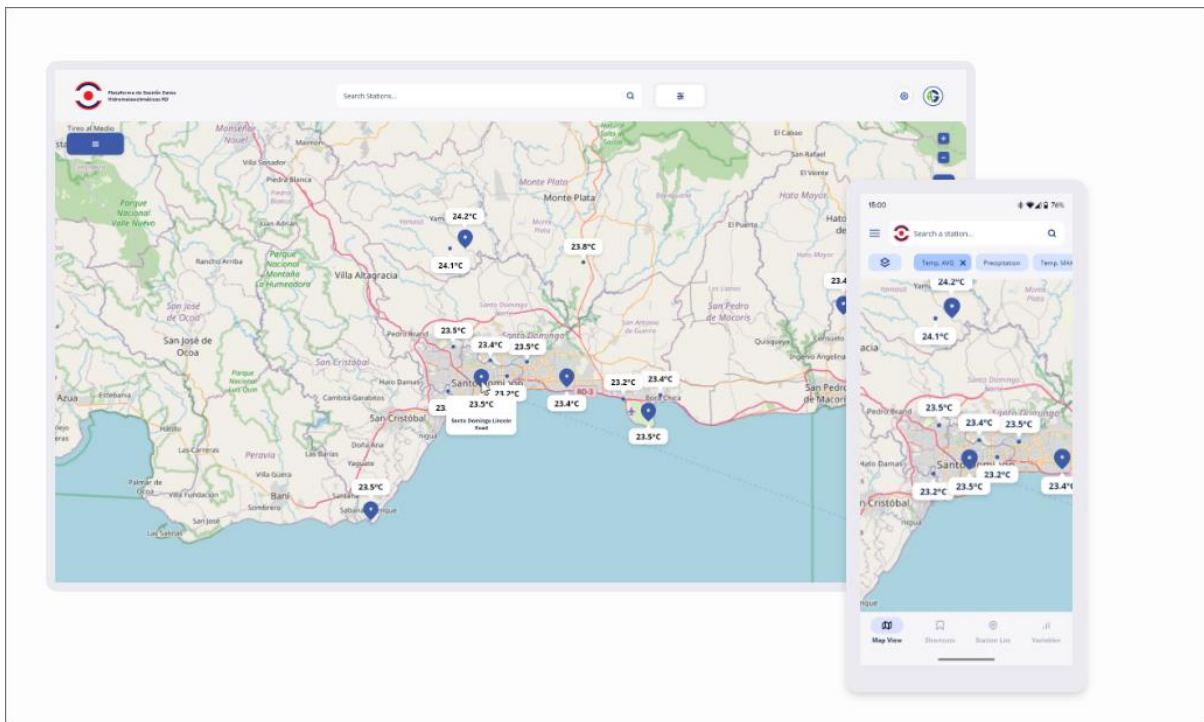


Figura 67

El objetivo de la maquetación móvil es ofrecer una experiencia de usuario completa y sin fisuras a los usuarios de teléfonos inteligentes, evitando errores en la transposición de contenidos del escritorio al móvil. Para lograrlo, es necesario utilizar patrones y componentes típicos del diseño visual para dispositivos móviles. Las siguientes imágenes muestran algunos ejemplos de cómo la plataforma puede utilizarse sin problemas en pantallas móviles.

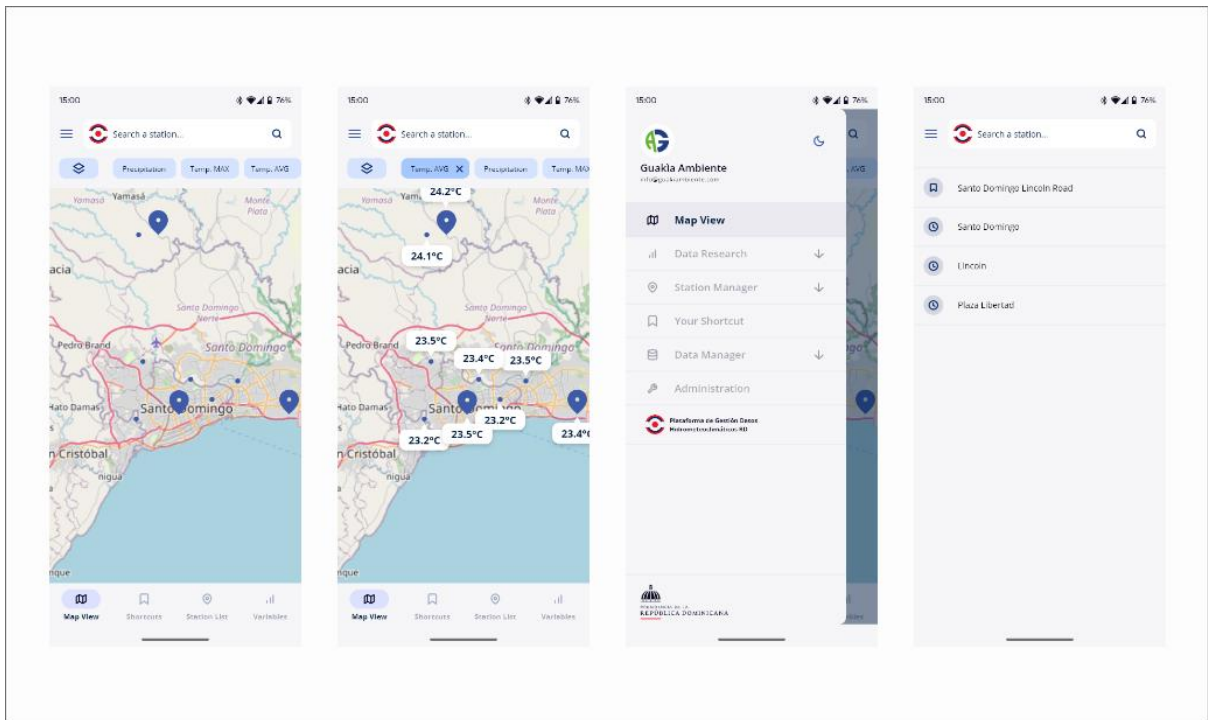


Figura 68. Vista del mapa, cajón de navegación y búsqueda

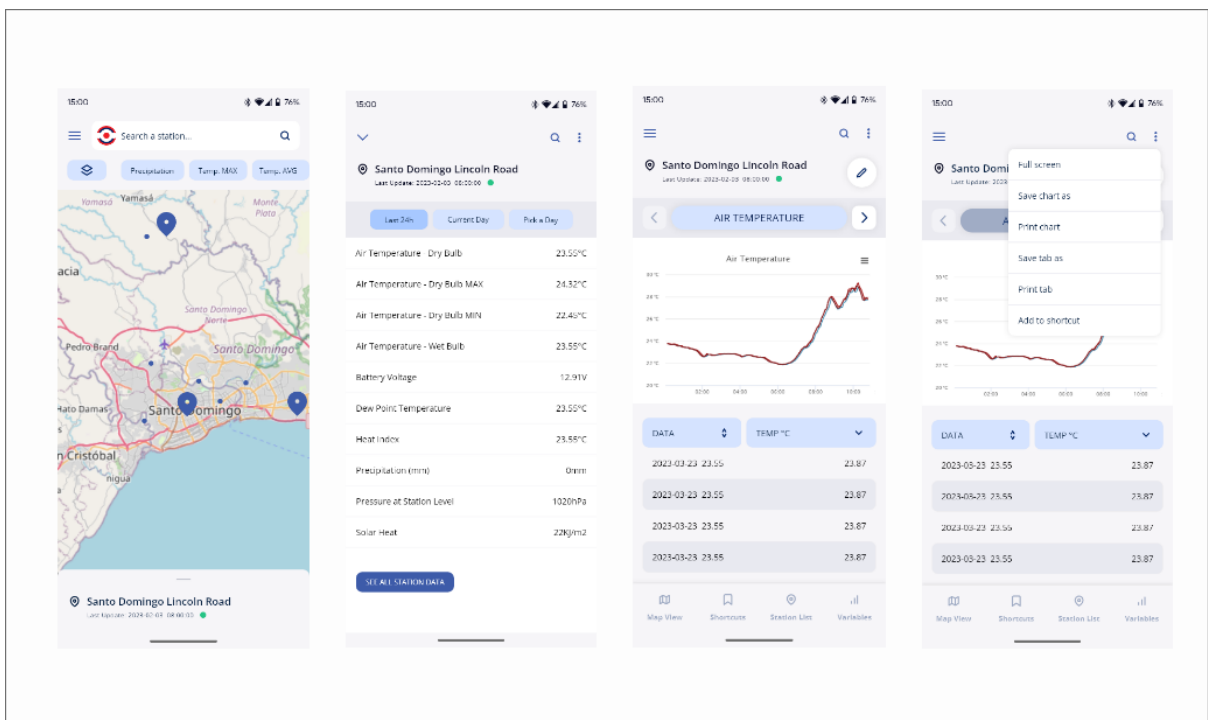


Figura 69. Selección de una estación, Datos de la estación desde la vista del mapa e Informe de la estación

Otra importante herramienta de accesibilidad es el modo oscuro, diseñado para aumentar el nivel de accesibilidad. A menudo se piensa que el modo oscuro es sólo un elemento gráfico estético, pero en realidad tiene muchas ventajas funcionales.

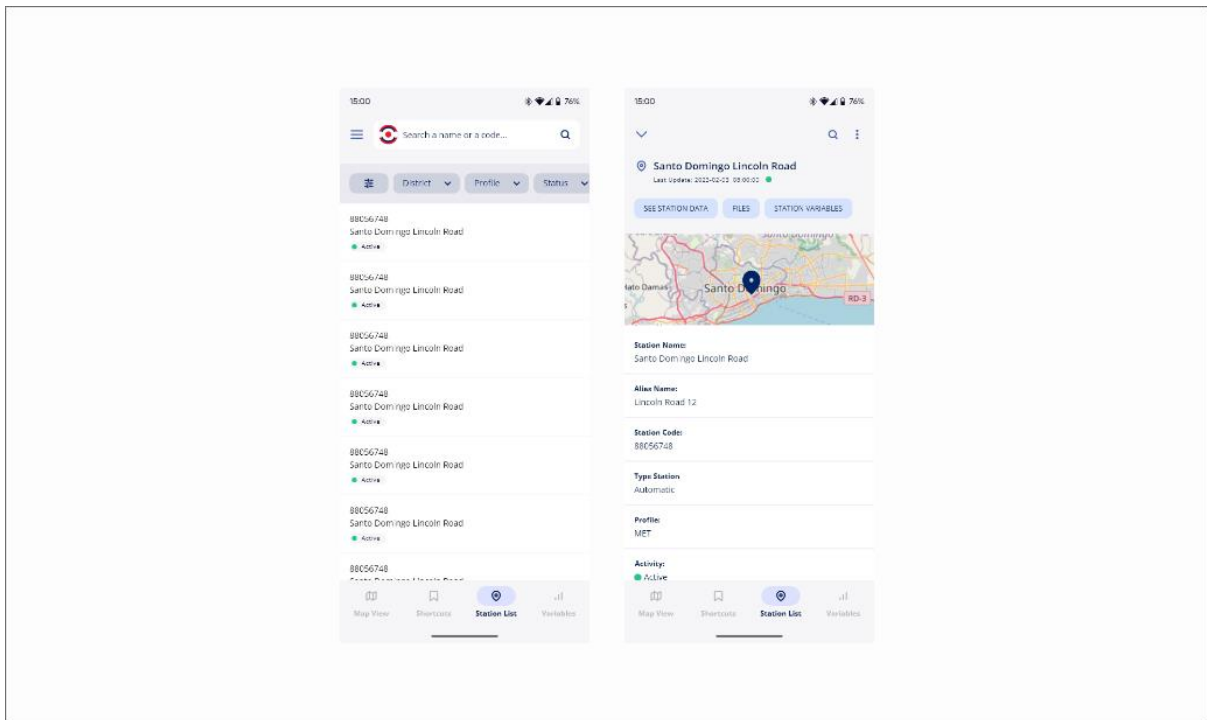


Figura 70. Listado de estaciones y metadatos de estaciones

En primer lugar, el modo oscuro reduce la exposición a la luz blanca que puede fatigar los ojos y provocar dolores de cabeza a muchas personas.

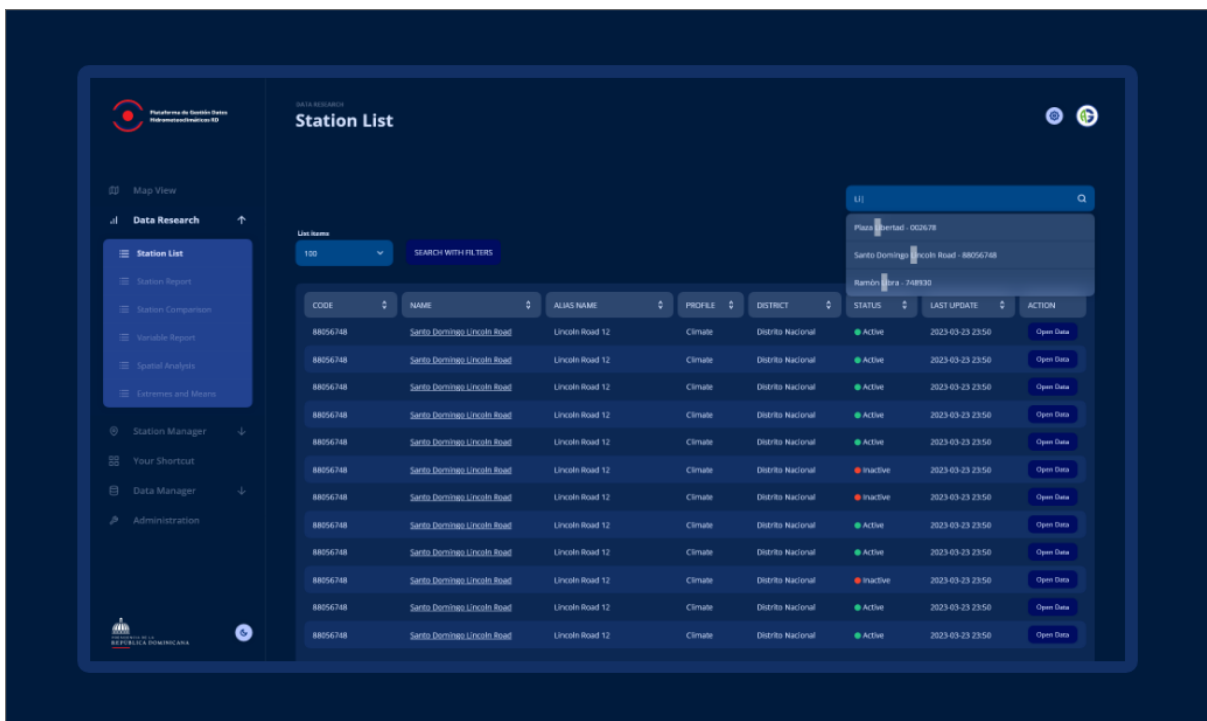


Figura 71. Versión oscura de Station List

Además, el modo oscuro es especialmente útil para personas con trastornos visuales como fotofobia, sensibilidad a la luz y astenopía. Además, el modo oscuro contribuye al

ahorro de energía, ya que en las pantallas OLED y AMOLED utilizadas en smartphones, tabletas y portátiles permite reducir el consumo de energía, alargando así la duración de la batería del dispositivo.

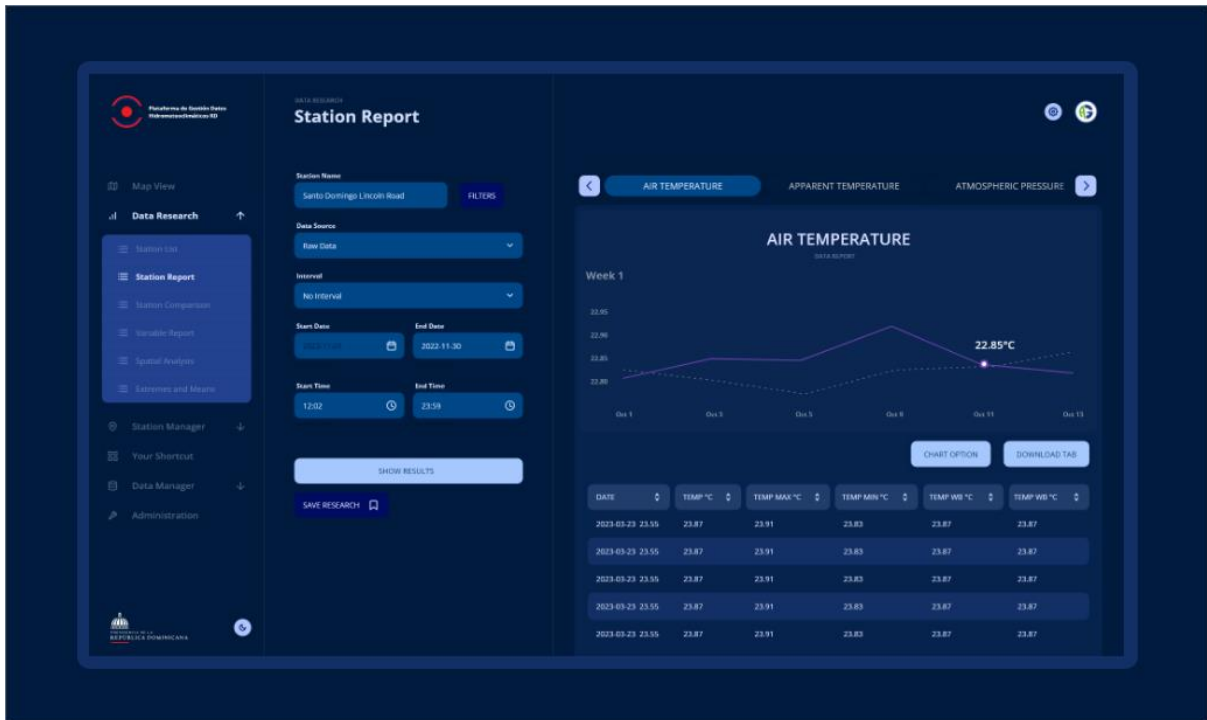


Figura 72. Versión oscura de Station Report

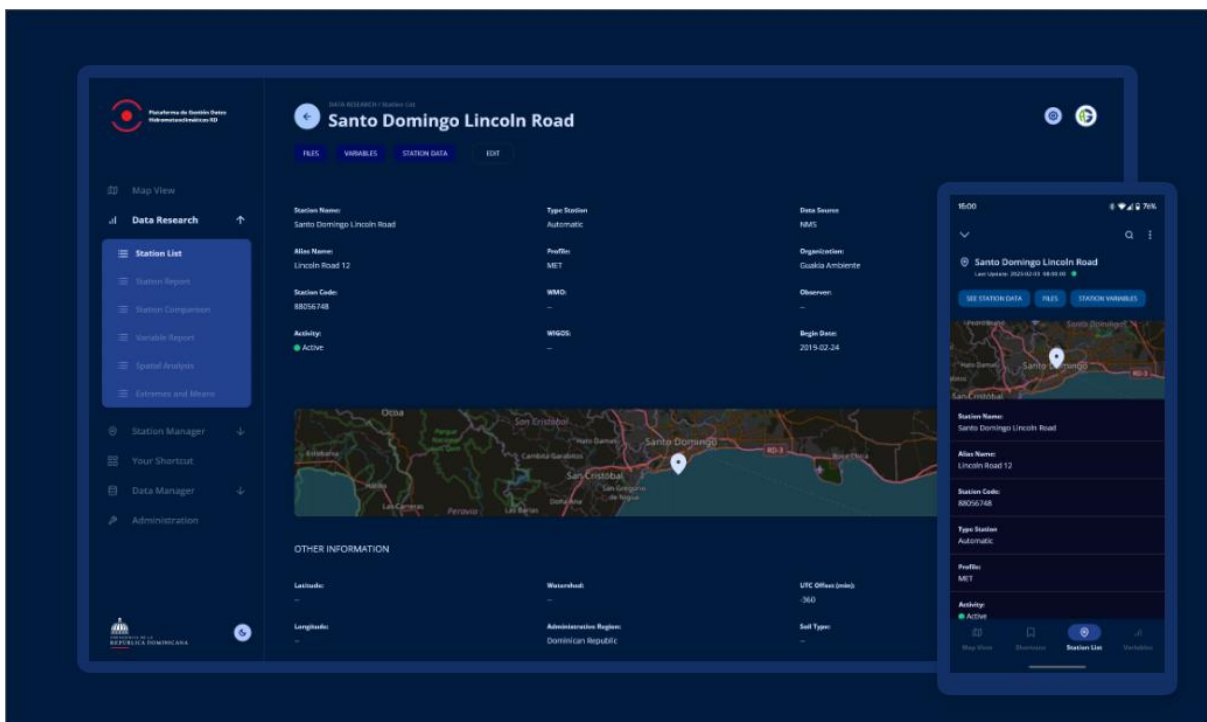


Figura 73. Versiones oscuras, de escritorio y móvil de los metadatos de la estación

## 6.4.4 MARCO TÉCNICO PROPUESTO

Para la realización de la nueva capa front-end personalizada, que incluirá los cuadros de mando de la aplicación accesibles a los usuarios finales en dispositivos de escritorio y móviles, se adoptarán las mejores prácticas y convenciones de codificación para garantizar un código seguro, sin errores y fácil de mantener, basado en arquitecturas y marcos sólidos. Además, se prestará especial atención a la experiencia del usuario, la usabilidad y la adaptabilidad del diseño para garantizar una aplicación multiplataforma utilizable en distintos tipos de dispositivos.

Para lograr estos objetivos, se propone utilizar Angular<sup>[55]</sup> como marco de diseño de aplicaciones y plataforma de desarrollo. Angular ofrece una solución eficaz y sofisticada que cumple los últimos estándares del sector, incluidos los de HTML5. Además, Angular permite crear aplicaciones multiplataforma, mejorar el rendimiento de los productos y aumentar la productividad del desarrollo.

El uso de Angular, con la aplicación de los patrones y las mejores prácticas adecuadas, garantizará la plena compatibilidad de la aplicación con diferentes dispositivos móviles y la capacidad de adaptarse a diferentes formatos de pantalla.

Además, para la creación de gráficos dentro de la aplicación, se propone utilizar Apache Echarts<sup>[56]</sup>, una potente herramienta de visualización para crear gráficos intuitivos, interactivos y altamente personalizables. Apache Echarts soporta diferentes tipos de gráficos con fines estadísticos y de análisis, como gráficos de líneas, de barras, de tarta, de velas y otros, además de soportar la localización geográfica con mapas y mapas de calor, y la visualización de datos multidimensionales con treemap, sunburst y otros.

### 6.4.4.1 ARQUITECTURA PERSONALIZADA DE EL LAYER DE FRONT-END

La arquitectura front-end propuesta para la plataforma se basa en varios objetivos clave. En primer lugar, se diseñó para garantizar una abstracción adecuada entre las capas de aplicación, permitiendo una clara separación de responsabilidades. Además, el flujo de datos es unidireccional, lo que garantiza una organización lineal y bien definida de la información dentro de la aplicación. La gestión reactiva del estado se implementó para permitir una actualización eficiente de las interfaces de usuario en respuesta a los cambios en el estado de la aplicación. Además, se adoptó un diseño modular, dividiendo el código en componentes independientes y autónomos. Por último, los componentes se organizaron de tal forma que algunos actúan como "contenedores" para manejar la

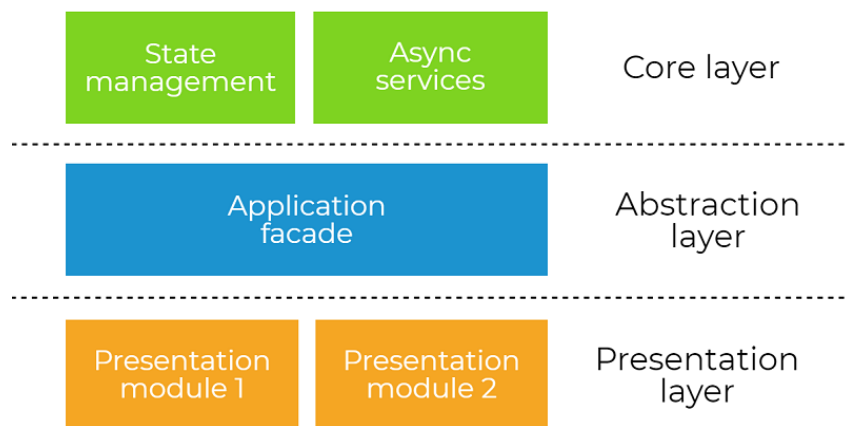
---

<sup>55</sup> [Sitio web oficial](#)

<sup>56</sup> <https://echarts.apache.org/en/index.html>

lógica de la aplicación, mientras que otros actúan como "presentaciones", centrándose únicamente en el aspecto visual.

Las capas de la aplicación de alto nivel se dividen en tres elementos principales: núcleo, abstracción y presentación. Esta división permite distintos niveles de responsabilidad que se comunican entre sí de forma selectiva. Por ejemplo, la capa de presentación sólo puede interactuar con el núcleo a través de la capa de abstracción, como se muestra en la **Figura 73**. Esto mantiene una clara separación entre la lógica de presentación y la lógica de negocio de la aplicación, mejorando la mantenibilidad y comprensibilidad del código.



**Figura 74. Capa de aplicación**

El uso de una arquitectura con capas separadas ofrece varias ventajas. En primer lugar, permite una mejor organización del código, con la lógica de negocio separada de la lógica de presentación, lo que facilita la comprensión del flujo de la aplicación. Además, la separación de tareas entre las distintas capas mejora la mantenibilidad del código, ya que permite modificar o añadir nuevas funcionalidades con menos impacto en los componentes existentes. Además, la gestión sensible del estado permite una interfaz de usuario más sensible y una mejor experiencia general del usuario.

#### **6.4.4.1.1 DISEÑO MODULAR**

Para garantizar un mejor mantenimiento del sistema, se decidió adoptar la estrategia de subdividirlo en pequeños módulos denominados "módulos de características", cada uno de los cuales ofrece una funcionalidad específica. Cada módulo de características sigue la misma arquitectura por capas descrita en la sección anterior, con una separación en capas de núcleo, abstracción y presentación. Esta estructura nos permite cargar módulos en modo perezoso en el navegador, es decir, cargarlos sólo cuando se utilizan realmente, mejorando así el tiempo de carga global de la aplicación.

### 6.4.4.1.2 FLUJO DE DATOS UNIDIRECCIONAL

En el contexto del desarrollo del front-end del producto, dos conceptos fundamentales son el flujo unidireccional de datos y la propagación de cambios. Estos conceptos son esenciales para garantizar la coherencia de los datos dentro de la aplicación y asegurarse de que la información mostrada está siempre actualizada y es coherente.

El flujo de datos unidireccional se refiere a la forma en que los datos se mueven a través de la aplicación en una sola dirección, es decir, desde el estado (State) al componente y no viceversa. Esto significa que los datos fluyen en una sola dirección, lo que evita los bucles de actualización y facilita la comprensión y el seguimiento del flujo de datos en la aplicación.

La propagación de cambios se refiere al mecanismo por el cual los cambios en los datos se notifican a los componentes afectados. Los objetos de estado, que contienen datos, pueden emitir notificaciones o eventos cuando cambian sus valores. A continuación, estas notificaciones se propagan a los componentes suscritos a estos eventos, lo que les permite reaccionar y actualizarse en consecuencia.

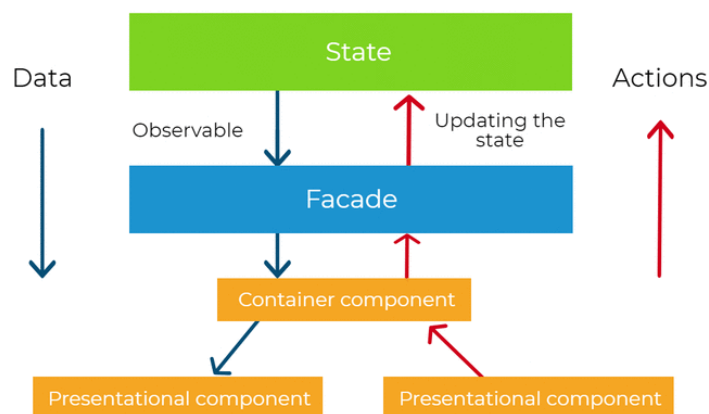


Figura 75. Gestión de los estados

Este enfoque de propagación de cambios a través de flujos observables, como con Redux o NgRx, asegura que los cambios de datos se produzcan de forma centralizada y controlada, garantizando la coherencia de los datos en toda la aplicación. En otras palabras, el objeto de Estado se considera la única fuente de verdad de los datos, lo que evita incoherencias o la duplicación de información en distintos puntos de la aplicación.

Este tipo de gestión reactiva del estado permite la coherencia de los datos dentro de la aplicación, garantizando que los datos mostrados por los distintos componentes estén siempre alineados y reflejen el estado actual de la aplicación. Esto es crucial para crear una experiencia de usuario coherente y fiable en la interfaz de usuario del producto.



### **6.4.4.1.3 GESTIÓN REACTIVA DEL ESTADO**

Un estado es un objeto JavaScript que contiene la estructura de datos de la aplicación. Puede poblarse con llamadas HTTP/HTTPS, notificaciones push para servicios en tiempo real e interacciones de los usuarios. Para manejar estos datos, se utilizó el patrón arquitectónico Redux (NgRx).

Redux permite gestionar los estados de una aplicación en un contenedor de estados predecible, basado en unos pocos principios clave. Uno de ellos es el concepto de "fuente única de la verdad", que significa que los datos de la aplicación residen en una ubicación centralizada. Esto evita la dispersión de datos en distintos repositorios y simplifica el seguimiento de los cambios de estado de los datos, especialmente cuando la aplicación crece en complejidad.

Además, en Redux, el estado es de sólo lectura, lo que significa que los componentes sólo pueden leer el estado, pero no pueden cambiarlo directamente. Para realizar cambios en el estado, se pueden enviar "acciones" a través del envío, delegando el cambio de estado en un punto central.

Un componente que desee utilizar un dato puede referirse a un "Observable" o suscribirse a un cambio de estado de un dato. En otras palabras, el componente escucha un dato concreto y recibe una notificación cada vez que cambia su valor.

Este patrón se adapta bien tanto a los sistemas que requieren actualizaciones en tiempo real de las partes del cliente (por ejemplo, a través de WebSocket) como a los sistemas en los que el cliente debe actualizar el contenido en modo de sondeo.

## 7 REFERENCIA FINAL

En esta sección se realizará una comparación detallada entre las plataformas analizadas anteriormente (descritas en la sección 3.1.2) y la propuesta de proyecto de la futura plataforma para la República Dominicana (DR CDMS).

**Cuadro 28. Evaluación comparativa de las plataformas existentes con la incorporación del sistema propuesto para la República Dominicana.**

	COLOMBIA	CHILE	MÉXICO	BELICE	DR CDMS
Mapa interactivo	●	●	●	●	●
Filtros por estación	●	●	● [57]	●	●
Filtros por variable	●	●	●	●	●
Filtros de tiempo	●	●	●	●	●
Descargar/Exportar datos	●	●	●	●	●
Gráficos de datos por estación	●	●	● [58]	●	●
Gráficos de datos por variable	●	●	● [59]	●	●
Visualización de datos estadísticos	●	●	●	●	●
Visualización de datos estadísticos en un mapa	●	●	●	●	●
Análisis espacial (mapa de calor)	●	●	● [60]	●	●
Lista de estaciones	● [61]	●	●	●	●
Comparación de estaciones	●	●	●	●	●

<sup>57</sup> Filtro parcial, sólo distingue entre estaciones activas y pasivas

<sup>58</sup> El gráfico no es dinámico, sino el resultado del procesamiento por software de terceros

<sup>59</sup> El gráfico no es dinámico, sino el resultado del procesamiento por software de terceros

<sup>60</sup> El análisis espacial no es dinámico, sino el resultado del procesamiento por un programa informático externo

<sup>61</sup> Lista de estaciones no visibles en la plataforma, pero descargables desde el sitio web

	COLOMBIA	CHILE	MÉXICO	BELICE	DR CDMS
Autenticación	●	●	●	●	●
Guardar y gestionar filtros y accesos directos	●	●	●	●	●
Gestión de estaciones	●	●	●	●	●
Control de calidad	●	●	●	●	●
Importación de datos	●	●	●	●	●
Gestión del Instituto	●	●	●	●	●

Las funcionalidades que faltaban, identificadas en los análisis realizados durante el documento, se añadieron al CDMS de DR para garantizar el pleno cumplimiento de la petición inicial de ofrecer al menos la misma funcionalidad que las demás plataformas existentes. En comparación con las plataformas de Colombia, Chile y México, se constató la ausencia de funcionalidades de autenticación.

Además, para responder a las necesidades específicas del contexto dominicano, se implementaron funcionalidades adicionales que no están presentes en las otras plataformas. Por ejemplo, la funcionalidad de importación de datos se añadió en respuesta al análisis y las necesidades que surgieron en el contexto de la República Dominicana. Aunque otras plataformas podrían ofrecer métodos de agregación de datos, esta funcionalidad específica no está disponible a través de la plataforma, por lo que se destacó como ausente. Otro ejemplo es la gestión de institutos, que se introdujo para permitir la integración de datos de diferentes entidades nacionales dominicanas en la plataforma, proporcionando así una solución para la gestión de dichos datos. No se puede determinar si otras plataformas ofrecen esta función, por lo que se ha marcado como ausente.

**En conclusión, el CDMS de RD respeta plenamente la petición inicial de ofrecer al menos la misma funcionalidad que las otras plataformas existentes, y también ofrece funcionalidad adicional para satisfacer las necesidades específicas del contexto dominicano.**

## 8 REQUISITOS DE SISTEMA, SEGURIDAD Y GESTIÓN

En este capítulo presentaremos una potencial arquitectura física diseñada para todo el sistema propuesto para la futura plataforma CDMS en República Dominicana, con énfasis en los aspectos de confiabilidad, tolerancia a fallas, seguridad y garantía de continuidad del servicio para los usuarios.

Debe tenerse en cuenta que la propuesta puede estar sujeta a variaciones significativas, que podrán aplicarse en función de los escenarios que se prefieran para la futura implantación del sistema, y de los requisitos y *KPI (Indicadores Clave de Rendimiento)* adicionales que se determinen en una fase posterior del proyecto.

### 8.1 ARQUITECTURA DE APLICACIONES

El objetivo de esta sección es presentar la arquitectura básica del sistema (**Figura 75**), objeto de la arquitectura física que se presentará más adelante. Para ello, se partió de la base de que optaríamos por la implementación de la capa de front-end personalizada, que ya ha sido ampliamente debatida y presentada en capítulos anteriores.

Como se muestra en la figura, la arquitectura básica del producto tiene tres capas interoperativas principales:

- Capa front-end: desarrollada en Angular<sup>[62]</sup>.
- Capa back-end: basada en la capa back-end Surface convenientemente modificada, que a su vez se construye utilizando el framework Django<sup>[63]</sup>.
- Capa de datos: desarrollada con TimescaleDB<sup>[64]</sup>.

Los tres niveles de arquitectura están estrechamente interconectados y trabajan juntos para ofrecer una experiencia de usuario completa y funcional. En concreto, la aplicación Angular, una vez distribuida a los clientes, se encarga de acceder y modificar los datos a través de la capa API en el back-end. De hecho, la capa de back-end, mediante el uso del componente Modelo proporcionado por el framework Django, es la única encargada de acceder directamente a los datos. Todo el acceso a la base de datos se realiza a través de la capa de integración, cuyos componentes se describen en la sección **5.5.1**.

---

<sup>62</sup> Descrito brevemente en el apartado **6.4.4**

<sup>63</sup> Sección **5.1** en adelante

<sup>64</sup> Sección **4.1.2.6.2**

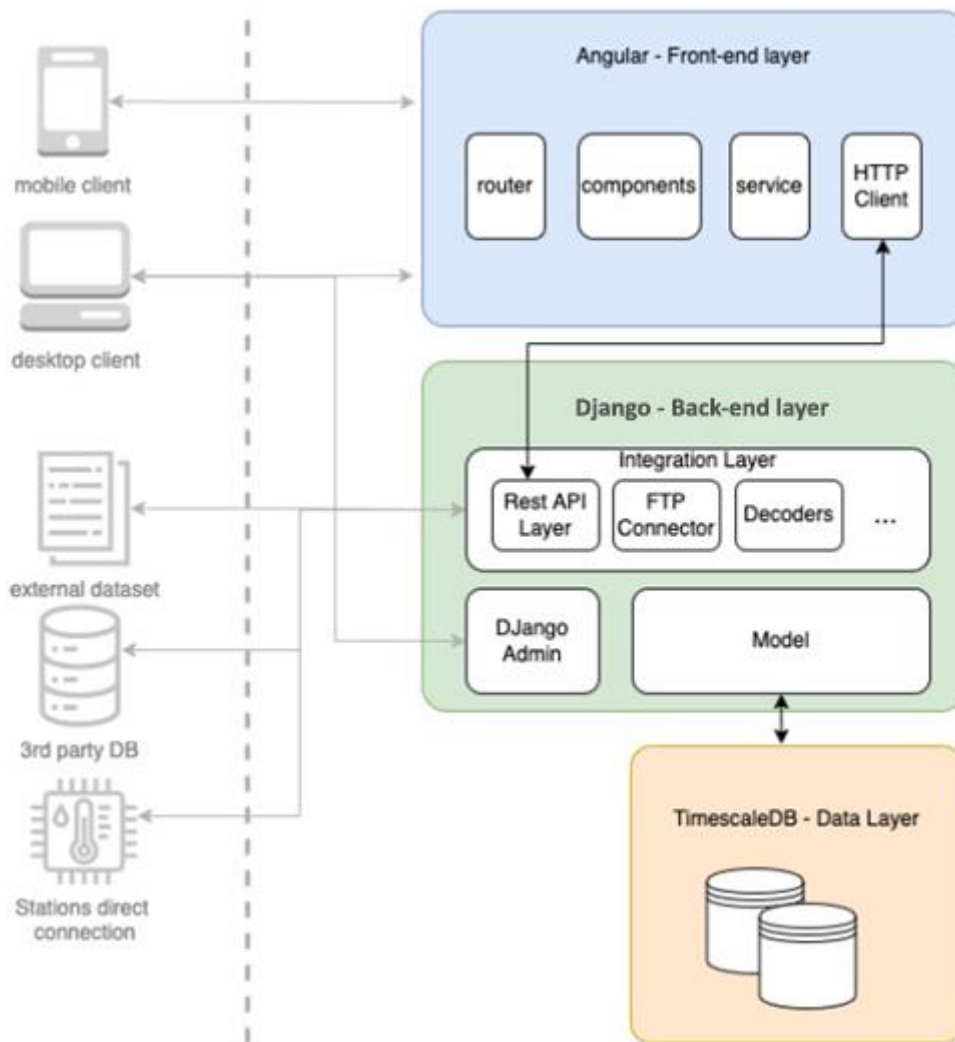


Figura 76. Arquitectura básica del sistema

Los usuarios administradores del sistema podrán acceder a las funciones de administración a través de sus clientes de escritorio. Estas funcionalidades serán expuestas por el módulo Django Admin y, si es necesario, el módulo puede ser externalizado y dockerizado por separado. Cada capa de la arquitectura básica se acoplará por separado y contribuirá a la configuración física general de la arquitectura, que se describirá con más detalle en los párrafos siguientes.

## 8.2 ARQUITECTURA DE INFRAESTRUCTURAS

La siguiente arquitectura y sus características se desarrollaron sobre la base de una serie de supuestos. Estas hipótesis se basaron en la información y los requisitos recopilados para la creación de este documento, y se eligieron la mejor solución para alcanzar los objetivos del proyecto. Sin embargo, es importante señalar que durante la aplicación de la arquitectura habrá que validar todos los supuestos.

La arquitectura de infraestructura presentada en este capítulo se define siguiendo, en la medida de lo posible, los principios de *agnóstico de la nube* y de código abierto, con la excepción de algunos servicios específicos. Una arquitectura agnóstica de nube está diseñada para ser independiente de la plataforma de nube específica en la que se ejecuta. Esto significa que la arquitectura puede implantarse en múltiples plataformas en la nube sin necesidad de realizar cambios significativos en el código o la infraestructura subyacentes. El objetivo de una arquitectura agnóstica de la nube es proporcionar mayor flexibilidad y portabilidad a los sistemas de software, permitiéndoles ejecutarse en diferentes plataformas en la nube en función de las necesidades de la organización o la aplicación, reduciendo el riesgo de dependencia de una única plataforma en la nube.

## 8.2.1 CONTRATACIÓN

### 8.2.1.1 PROVEEDOR DE LA NUBE

El primer supuesto es que la solución está alojada en una infraestructura de nube. Recomendamos esta opción por las diversas ventajas que ofrece frente a una infraestructura tradicional con servidores propietarios:

- **Escalabilidad:** con la capacidad de ampliar rápidamente los recursos según sea necesario, la infraestructura en nube permite añadir o eliminar CPU, RAM, almacenamiento y ancho de banda sin tener que comprar hardware adicional ni realizar costosas actualizaciones.
- **Flexibilidad:** la modularidad de la infraestructura en nube permite personalizar el entorno de alojamiento y elegir los recursos necesarios en función de las necesidades específicas de la aplicación. Además, la distribución de la aplicación en varias regiones geográficas mejora la disponibilidad y la latencia en todo el mundo.
- **Reducción de costes:** el uso de una infraestructura en nube evita los costes iniciales de adquisición de hardware y software, así como los costes de mantenimiento y actualización de servidores propios. Además, los servicios en la nube ofrecen facturación por uso, lo que permite pagar sólo por los recursos que realmente se utilizan, sin tener que adquirir recursos en exceso para hacer frente a picos repentinos de tráfico.
- **Seguridad:** los servicios en nube ofrecen un alto nivel de seguridad mediante medidas de seguridad físicas y lógicas, cifrado de datos en tránsito y en reposo, autenticación multifactor y supervisión continua del acceso y la actividad de los usuarios.

Sobre la base de los requisitos recopilados, la infraestructura de la nube es totalmente compatible para alojar la plataforma propuesta. Sin embargo, si se eligiera una opción diferente por razones de gobernanza u otros requisitos aún no identificados, muchas de las características, mejores prácticas y directrices que se describirán en la arquitectura física también seguirían siendo válidas para una infraestructura alternativa.

Para diseñar una arquitectura que permitiera deducir los costes de infraestructura con la mayor precisión posible, fue necesario seleccionar un proveedor de servicios en la nube, recayendo la elección en Amazon Web Services (AWS)<sup>[65]</sup>.

AWS ofrece una amplia gama de servicios entre todos los proveedores de servicios en la nube disponibles en el mercado y es conocido por su fiabilidad, con un tiempo de actividad del servicio del 99,99%. Esto significa que los clientes pueden contar con que los servicios en la nube estarán siempre disponibles y pueden escalar rápida y fácilmente los recursos en la nube en función de sus necesidades empresariales. Además, AWS proporciona herramientas de automatización y orquestación para simplificar el proceso de escalado.

En cuanto a la seguridad, AWS ofrece una amplia gama de herramientas para proteger los datos y las aplicaciones, como cifrado, acceso controlado, seguridad de red y mucho más. AWS también cumple las principales normas de seguridad, como ISO 27001, SOC 1/2/3 y PCI DSS.

Por último, AWS ofrece una amplia gama de opciones de precios flexibles, como el pago por uso y las instancias reservadas, que permiten a los clientes ahorrar costes a largo plazo. Además, AWS proporciona herramientas de monitorización de costes para ayudar a los clientes a realizar un seguimiento de sus costes de infraestructura en la nube.

Otro aspecto examinado se refiere al impacto medioambiental del servicio. En la actualidad, es difícil determinar con exactitud qué proveedor de nube es el "más ecológico" del mercado, ya que las empresas utilizan diferentes métricas para evaluar el impacto ambiental de sus operaciones e infraestructura en la nube. Sin embargo, hay algunos proveedores que destacan por sus esfuerzos en materia de sostenibilidad medioambiental.

Amazon Web Services (AWS) ha tomado medidas para reducir el impacto ambiental de sus actividades, como el uso de energías renovables como fuente primaria de electricidad para sus centros de datos. Además, AWS lanzó el programa "AWS Sustainability" para ayudar a los clientes a mejorar la sostenibilidad de sus negocios mediante el uso de servicios en la nube.

---

<sup>65</sup> [Sitio web oficial](#)

### 8.2.1.2 DISPONIBILIDAD Y RENDIMIENTO

A continuación, se presentan otros supuestos, detallados siguiendo una serie de temas relacionados con la disponibilidad y el rendimiento de la plataforma:

- **Retraso en la sincronización de datos:** la plataforma debe garantizar que los datos climáticos se sincronicen en tiempo real o, como máximo, con un retraso de 45 minutos. Tiempo de actividad: la plataforma debe garantizar un tiempo de actividad del servicio del 99,7% durante el año, es decir, no debe estar inactiva más de 26,28 horas en total durante el año.
- **Tiempo máximo de inactividad:** El tiempo máximo de inactividad, incluidas las sesiones de mantenimiento del sistema, no debe ser superior a tres horas consecutivas durante el año.
- **Periodo de cambio:** si la plataforma requiere mantenimiento o actualizaciones, debe establecerse un periodo concreto (por ejemplo, noches o fines de semana) en el que puedan llevarse a cabo para minimizar el impacto en los usuarios.
- **Recuperación en caso de catástrofe:** debe establecerse un plan de recuperación en caso de catástrofe que defina las medidas a tomar en caso de emergencia (por ejemplo, pérdida de datos debido a un suceso natural) para garantizar el restablecimiento del servicio en el menor tiempo posible.
- **Rendimiento y escalabilidad:** la plataforma debe ser capaz de gestionar un determinado número de solicitudes de datos por segundo o por minuto sin retrasos ni bloqueos y tener la capacidad de realizar un escalado del rendimiento en función de las necesidades de un periodo concreto.

### 8.2.2 DIAGRAMA ARQUITECTÓNICO Y REQUISITOS DEL SISTEMA

La **Figura 76** muestra una arquitectura altamente escalable y resistente basada en el uso de determinados servicios en la nube de AWS que garantizan una alta disponibilidad, seguridad y redundancia.

La Virtual Private Cloud (VPC) permite aislar la infraestructura de red del entorno público y crear una red privada virtual en la que están presentes las subredes, lo que proporciona mayor seguridad y control sobre la red. Las dos zonas de disponibilidad proporcionan redundancia a los servicios que componen la arquitectura, garantizando la continuidad del servicio incluso en caso de interrupción en una de las zonas.



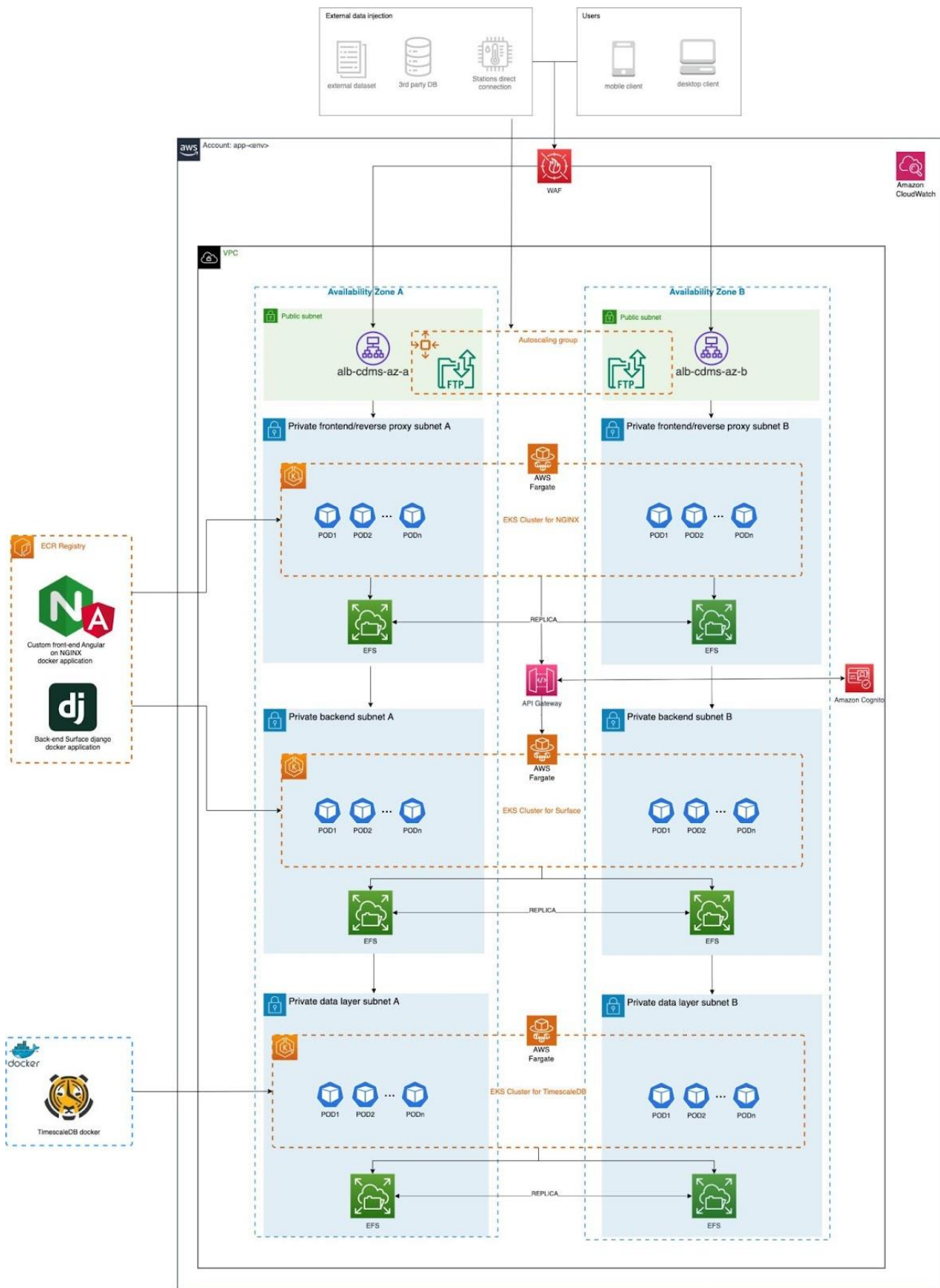


Figura 77. Diagrama de la arquitectura propuesta del entorno de producción

Dentro de cada zona de disponibilidad, se propone crear perímetros de red bien definidos a través de diferentes subredes. Aislar las distintas capas de una aplicación en subredes separadas ofrece varias ventajas:

- **Seguridad:** permite aislar los distintos componentes de la arquitectura en subredes separadas, lo que proporciona mayor seguridad y reduce el riesgo de ciberataques.
- Control de **acceso:** permite gestionar el control de acceso a distintas subredes de forma específica, por ejemplo, permitiendo el acceso sólo a determinados usuarios o aplicaciones.
- **Fiabilidad:** permite distribuir los recursos en subredes separadas, lo que garantiza una mayor resistencia y disponibilidad en caso de interrupción del servicio.
- **Escalabilidad:** permite añadir o eliminar recursos fácilmente en una subred específica sin afectar a otras subredes.
- **Gestión del tráfico de red:** permite gestionar el tráfico de red de forma específica para cada subred, lo que garantiza una mayor eficacia y rendimiento para la aplicación.

Cada capa de aplicación está delimitada por diferentes subredes entre las dos zonas de disponibilidad. Para cada zona de disponibilidad, existen tres subredes privadas para las tres capas de aplicación (capa de front-end, capa de back-end y capa de datos) y una única subred pública para el acceso desde el exterior. Las subredes propuestas son las siguientes:

- **Subred pública:** Esta subred es accesible desde la red pública y permite el acceso a un *Application Load Balancer* (ALB) que equilibra el tráfico entrante a los diferentes recursos de la arquitectura. Además, desde esta subred se accede a una interfaz SFTP (*Secure File Transfer Protocol*) que permite depositar los flujos de datos procedentes del perímetro de inyección de datos externos en el EFS (*Elastic File System*) al que se accede desde la capa de back-end.
- **Subred privada 1 (Subred privada para el front-end/proxy inverso):** esta subred está conectada a la subred pública y a la subred privada 2, que permite el acceso a la capa back-end de la solución. En concreto, esta subred proporciona acceso a una imagen docker NGINX con la aplicación Angular personalizada, alojada en la capa de orquestación EKS (*Elastic Kubernetes Service*) gestionada por AWS Fargate.
- **Subred Privada 2 (Subred Privada para el back-end):** Esta subred está conectada a la Subred Privada 1 y a la Subred Privada 3, lo que permite el acceso a la capa de datos. A través de esta subred se puede acceder a la imagen docker de la plataforma Surface y a su capa API alojada en la capa de orquestación EKS gestionada por AWS Fargate. Además, la capa API está protegida por el servicio Amazon API Gateway, que permite a los desarrolladores crear, publicar, mantener, monitorizar y proteger API para aplicaciones y servicios web.

- **Subred privada 3 (Subred privada para la capa de datos):** Esta subred está conectada a la subred privada 2 y proporciona acceso a la imagen docker de la base de datos de productos TimescaleDB, hospedada por la capa de orquestación EKS administrada por AWS Fargate.

Cada subred tiene acceso a un servicio AWS EFS, que permite el almacenamiento de todo el contenido (configuraciones, contenido estático y datos) y la replicación *entre regiones*, garantizando así una alta disponibilidad y redundancia de los datos. El EFS es único y se puede acceder a él desde diferentes subredes. En el diagrama, para simplificar, se ha mostrado una instancia del EFS en cada subred.

Todos los contenedores Docker mencionados se almacenarán en un ECR (*Elastic Container Registry*), a excepción de TimescaleDB, que se recuperará directamente del repositorio del fabricante. El uso de un registro ECR simplifica la gestión de imágenes Docker, permitiendo un versionado y despliegue eficientes y seguros.

Además, todas las solicitudes externas se filtran a través de un WAF (*Web Application Firewall*). AWS WAF es un servicio de protección de aplicaciones web que permite filtrar el tráfico HTTP y HTTPS entrante a las aplicaciones web.

## 8.2.2.1 ORQUESTACIÓN DE CONTENEDORES

Amazon Web Services (AWS) ofrece dos servicios de orquestación de contenedores: Amazon Elastic Container Service (ECS) y Amazon Elastic Kubernetes Service (EKS). Ambos servicios permiten administrar y ejecutar contenedores en AWS, pero presentan diferencias sustanciales. En particular, el uso de EKS, tal como se sugiere en la arquitectura propuesta, tiene ciertas ventajas sobre ECS en lo que respecta a evitar la dependencia de un proveedor.

La dependencia de un proveedor es una situación en la que una empresa está atada a un proveedor tecnológico concreto, lo que se traduce en costes elevados y dificultades para migrar a otros proveedores. Entre las ventajas de EKS frente a ECS para evitar la dependencia de un proveedor se encuentran:

- **Normalización:** EKS se basa en Kubernetes, una infraestructura de código abierto para la orquestación de contenedores. Esto significa que las aplicaciones de Kubernetes pueden ejecutarse en cualquier nube o entorno local que admita Kubernetes, no solo en AWS. En cambio, ECS tiene una API propia específica de AWS que requiere conocimientos más especializados que Kubernetes.
- **Portabilidad:** dado que Kubernetes es una infraestructura de código abierto, las aplicaciones de Kubernetes pueden trasladarse fácilmente de un entorno a otro, sin estar atadas a AWS. Esto significa que las aplicaciones pueden ejecutarse en

diferentes nubes o entornos locales, sin tener que reescribirse. En cambio, con ECS, las aplicaciones están vinculadas a la API propietaria de AWS y puede resultar más difícil migrarlas a otros entornos.

- **Comunidad:** Kubernetes cuenta con una gran comunidad de código abierto que contribuye al desarrollo de la infraestructura y proporciona asistencia y recursos a sus usuarios. Esto significa que las aplicaciones Kubernetes pueden beneficiarse de un soporte más amplio que las aplicaciones ECS, que dependen principalmente del soporte de AWS.

En general, la adopción de EKS puede ayudar a evitar la dependencia de un proveedor, ya que se basa en Kubernetes, una tecnología de código abierto respaldada por una gran comunidad. Esto permite a las empresas migrar sus aplicaciones a diferentes entornos sin estar atadas a AWS u otros proveedores específicos.

### 8.2.2.1.1 AWS FARGATE

AWS Fargate es un servicio informático de contenedores totalmente administrado que permite ejecutar contenedores sin necesidad de administrar la infraestructura subyacente. En otras palabras, Fargate permite ejecutar contenedores como se ejecutan máquinas virtuales, sin preocuparse de la configuración y gestión de la infraestructura subyacente.

Fargate se puede utilizar en combinación con Amazon Elastic Kubernetes Service (EKS) de varias maneras, pero el uso recomendado es ejecutar pods de Kubernetes en Fargate sobre EKS. De este modo, Fargate actúa como una alternativa de ejecución a los nodos EKS EC2 tradicionales, lo que significa que es posible ejecutar pods de Kubernetes en Fargate sin tener que gestionar manualmente la infraestructura subyacente.

Utilizar Fargate junto con EKS ofrece varias ventajas, como:

- **Gestión simplificada de la infraestructura:** Fargate permite ejecutar contenedores sin tener que gestionar la infraestructura subyacente. Esto significa que no hay necesidad de preocuparse por la configuración y gestión de los nodos EKS EC2.
- **Costes reducidos:** al utilizar Fargate, los costes pueden reducirse en comparación con el uso de nodos EKS EC2, ya que sólo se paga por el tiempo de ejecución real de los contenedores.
- **Mayor escalabilidad:** con Fargate, el número de contenedores puede ampliarse rápidamente en función de las necesidades de la aplicación sin tener que preocuparse por la gestión de la infraestructura.

## 8.2.2.2 TERRAFORM VS CloudFormation

Para estructurar el trabajo de acuerdo con los principios agnósticos de la nube, se sugiere utilizar Terraform<sup>[66]</sup> en lugar de CloudFormation, el servicio propietario de AWS, para el aprovisionamiento y la gestión de la infraestructura.

Terraform es una herramienta de código abierto desarrollada por HashiCorp que permite definir la infraestructura como código (IaC) mediante un lenguaje declarativo. De este modo, Terraform automatiza el proceso de creación, modificación y destrucción de recursos de infraestructura en nubes públicas (como AWS, Google Cloud Platform y Microsoft Azure) y en infraestructuras locales.

Terraform ofrece numerosas ventajas sobre CloudFormation de AWS, entre las que se incluyen:

- **Multi-nube:** Terraform puede soportar múltiples proveedores de nubes públicas, permitiendo la creación de infraestructuras multi-nube.
- **Declarativo:** Terraform se basa en un lenguaje declarativo, lo que significa que usted define el estado deseado de la infraestructura y Terraform se encarga de alcanzar este estado, manejando todos los detalles técnicos.
- **Modular:** Terraform permite organizar el código en módulos reutilizables, facilitando la creación de infraestructuras complejas.

## 8.2.3 REQUISITOS DE SEGURIDAD

En los párrafos siguientes, detallaremos brevemente todas las características de seguridad que ofrece la arquitectura propuesta del futuro CDMS en la República Dominicana.

### 8.2.3.1 AWS WAF (WEB APPLICATION FIREWALL)

Con AWS WAF (*Web Application Firewall*), se pueden crear reglas de seguridad personalizadas para proteger las aplicaciones web de ataques como la inyección SQL, el scripting entre sitios (XSS) y el scripting fraccionado (FRAG). Este servicio se basa en un conjunto de reglas que pueden ser definidas por el usuario o utilizadas por defecto proporcionadas por AWS. Además, se pueden crear reglas personalizadas para proteger las aplicaciones web de vulnerabilidades conocidas. AWS WAF cubre muchas de las

---

<sup>66</sup> [Sitio web oficial](#)

vulnerabilidades más comunes y proporciona protección avanzada para aplicaciones web.

### **8.2.3.2 SFTP (SECURE FILE TRANSFER PROTOCOL)**

SFTP (*Secure File Transfer Protocol*) es un protocolo seguro de transferencia de archivos que ofrece mayor seguridad que el tradicional FTP (*File Transfer Protocol*). Algunas ventajas de seguridad que SFTP ofrece sobre FTP son:

- **Cifrado:** SFTP utiliza un potente cifrado para proteger los datos durante la transferencia. Los datos se cifran tanto durante la transferencia como durante el almacenamiento en los servidores.
- **Autenticación:** SFTP admite la autenticación basada en claves públicas y privadas, que es más segura que la autenticación basada en nombre de usuario y contraseña utilizada en FTP.
- **Integridad de los datos:** SFTP proporciona mecanismos para garantizar la integridad de los datos transmitidos, protegiendo los archivos de manipulaciones no autorizadas.

### **8.2.3.3 API GATEWAY**

Amazon API Gateway, como ya se ha mencionado, es un servicio de AWS que permite a los desarrolladores crear, publicar, mantener, monitorizar y proteger API (interfaces de programación de aplicaciones) para aplicaciones y servicios web.

El API Gateway proporciona un único punto final para las aplicaciones que utilizan la API, lo que facilita la gestión y el control del acceso a los recursos y permite escalar fácilmente los recursos en función de las necesidades de la aplicación.

La API de Gateway también ofrece funciones avanzadas de supervisión y análisis que permiten a los desarrolladores hacer un seguimiento de las métricas clave (por ejemplo, número de solicitudes y tiempo de respuesta) y realizar análisis de datos para mejorar el rendimiento de las aplicaciones. Además, incluye funciones de seguridad avanzadas como autenticación de usuarios, gestión de accesos y protección contra ataques *dedenegación de servicio distribuido* (DDoS).

### 8.2.3.4 COGNITO AWS Y GESTIÓN DE UTILIDADES

Hoy en día, las aplicaciones informáticas se integran cada vez más con servicios de inicio de sesión único (SSO) para mejorar sus funciones de seguridad de forma eficaz y sin problemas. Mediante el uso de servicios SSO, se puede aprovechar un flujo de trabajo básico ya implantado, probado y disponible en la nube, que incluye el inicio de sesión, el registro, el restablecimiento de contraseñas, la autenticación de dos factores (2FA), la verificación del correo electrónico y el número de teléfono, el envío de correos electrónicos de bienvenida y mucho más.

Entre los servicios SSO, AWS Cognito es uno de los más cómodos y flexibles. Permite la personalización del flujo de trabajo mediante el uso de Lambda Triggers. En particular, durante los eventos de autorización, es posible llamar a funciones personalizadas de AWS Lambda donde se desee, con el fin de realizar las operaciones deseadas. También se pueden crear flujos de autenticación personalizados. En el caso de que la aplicación se aloje en AWS y se gestione mediante Terraform, también resulta muy cómodo configurar e instalar Cognito como un recurso adicional en la propia infraestructura, utilizando el enfoque de Infraestructura como código (IaC).

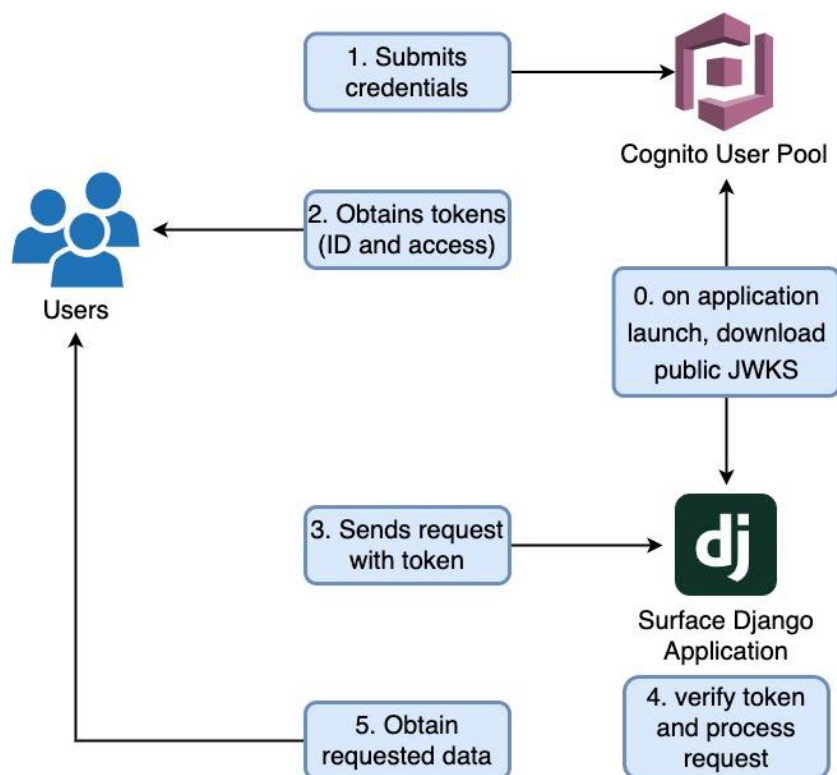


Figura 78. Integración prevista entre Django y Cognito

Es importante destacar la importancia de utilizar Cognito para garantizar la seguridad informática y no para exponer directamente los mecanismos de autenticación que ofrece Surface a través del framework Django. De hecho, el uso de Cognito puede mejorar enormemente la seguridad del sistema, ya que proporciona una arquitectura

segura y robusta para gestionar las credenciales de los usuarios. También puede ayudar a prevenir problemas de seguridad y vulnerabilidades asociadas a la gestión de las contraseñas de los usuarios, como el hash incorrecto de las contraseñas.

Para utilizar el servicio de autenticación de Amazon Cognito para autenticar usuarios en una aplicación web cliente-servidor, es necesario integrarlo con la capa de back-end de la plataforma de aplicaciones, en particular con el framework Django. El proceso de autenticación consta de varios pasos numerados del 0 al 5, como se muestra en la **Figura 77**, que destacan los puntos de integración entre AWS Cognito y la aplicación.

Tras una autenticación correcta en Cognito, la API de Amazon Cognito devuelve tres tokens: el `id_token`, el `access_token` y el `refresh_token`. El `id_token` contiene información personal como el nombre, el correo electrónico y el número de teléfono del usuario, mientras que el `access_token` se utiliza para acceder a los recursos protegidos de la aplicación y no contiene información personal. El `refresh_token` se utiliza para obtener un nuevo `access_token` cuando expira el anterior.

Sin embargo, para verificar la firma de estos tokens, el intercambio de claves JWKS entre Cognito y la aplicación debe haberse completado con éxito. JWKS es un formato estandarizado para representar un conjunto de claves criptográficas públicas utilizadas para firmar y verificar JSON Web Tokens (JWT)<sup>[67]</sup>.

La respuesta de Cognito que contiene los tokens JWT también incluye una URL pública (conocida como URL JWKS) desde la que la aplicación web puede descargar el JWKS que contiene la clave pública utilizada para firmar los tokens JWT. A continuación, la aplicación web descarga el JWKS de la URL proporcionada por Cognito y lo utiliza para verificar la firma del token JWT recibido de Cognito. Sólo si la firma del token es válida, la aplicación web autoriza al usuario a acceder a los recursos protegidos.

### **8.2.3.5 MINIMIZACIÓN DE RIESGOS, PRUEBAS DE PENETRACIÓN Y ANÁLISIS ESTÁTICO**

La implementación de un sistema seguro es de suma importancia, especialmente en el contexto actual donde las amenazas informáticas son cada vez más frecuentes y sofisticadas. La lista TOP TEN de OWASP<sup>[68]</sup> representa un punto de partida casi imprescindible para desarrollar un código que sea lo más inmune posible a las principales vulnerabilidades conocidas.

---

<sup>67</sup> [Profundización en el tema](#)

<sup>68</sup> <https://owasp.org/www-project-top-ten/>



Sin embargo, un código seguro no es una garantía suficiente. Es necesario probar el sistema de manera exhaustiva utilizando técnicas como las pruebas de penetración. Este tipo de actividad implica simular ataques externos al sistema con el fin de identificar posibles puntos débiles o vulnerabilidades. Dado la sensibilidad de los datos recolectados y almacenados en el sistema, resulta indispensable probar la plataforma misma para garantizar su máxima impenetrabilidad desde el exterior. Estas actividades se llevarán a cabo antes de que el sistema se ponga en funcionamiento durante al menos diez días, con el objetivo de garantizar la máxima confiabilidad y seguridad posibles.

Además, el análisis estático del código es otra técnica para garantizar la máxima seguridad posible. En este tipo de actividad, el código se analiza exhaustivamente en busca de posibles vulnerabilidades o errores de programación. Gracias a esta actividad, será posible identificar posibles errores o vulnerabilidades del sistema durante la fase de desarrollo, sin tener que esperar a que el sistema esté en producción. El uso de una de las herramientas recomendadas por OWASP<sup>[69]</sup> permite automatizar gran parte de este proceso, reduciendo el riesgo de errores humanos y acelerando la detección de posibles problemas.

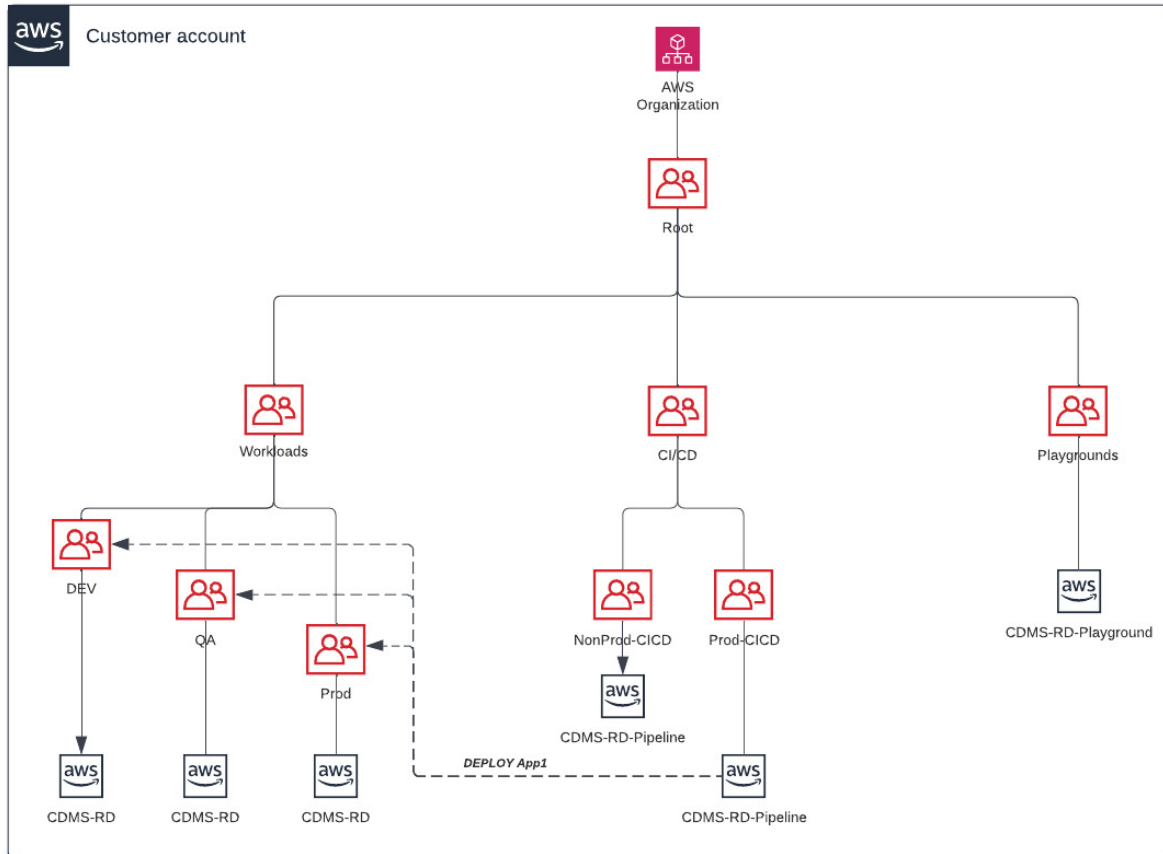
## **8.2.4 ENTORNOS Y CONDUCTOS CI/CD (INTEGRACIÓN CONTINUA/DESPLIEGUE CONTINUO)**

En esta sección se presenta el diseño de la *zona de aterrizaje* (**Figura 79**). En AWS, la zona de aterrizaje es una arquitectura de referencia que representa un entorno básico seguro y escalable que puede utilizarse como punto de partida para implementar nuevas infraestructuras en AWS.

En la práctica, una zona de aterrizaje proporciona un conjunto de prácticas recomendadas, directrices y herramientas para configurar una infraestructura básica, incluidas cuentas de AWS, CI/CD, gobernanza, sandboxes y gestión del acceso, con el objetivo de crear un entorno predecible, escalable y seguro para implementar aplicaciones y servicios.

---

<sup>69</sup> [https://owasp.org/www-community/Source\\_Code\\_Analysis\\_Tools](https://owasp.org/www-community/Source_Code_Analysis_Tools)



**Figura 79. Diseño de la zona de aterrizaje**

La estructura jerárquica de la zona de desembarco está organizada en tres OU (unidades organizativas):

1. OU de cargas de **trabajo**: Esta OU está diseñada para contener las principales cargas de trabajo de la organización, como aplicaciones y servicios web. Esta OU alberga los principales recursos, como máquinas virtuales (EC2), servidores, bases de datos y aplicaciones, que dan soporte a las actividades principales de la organización. Además, en esta OU se aplican las principales políticas de seguridad de la organización, como la seguridad de los datos, la gestión del acceso y el cumplimiento de la normativa. Hay tres subunidades organizativas en las cargas de trabajo de la OU:
  - a. **OU DEV**: donde los desarrolladores pueden trabajar en sus propios proyectos y probarlos antes de su publicación en fases posteriores.
  - b. **OU QA (Quality Assurance)**: los desarrolladores prueban las aplicaciones en un entorno controlado, similar al de producción, para detectar cualquier problema o error antes de su lanzamiento final.
  - c. **OU PROD**: OU principal donde la aplicación se ejecuta en producción y se pone a disposición de los usuarios finales.
2. **CI/CD OU**: Esta OU está diseñada para contener los recursos que soportan el proceso de Integración Continua y Despliegue Continuo (CI/CD) de la organización. Aquí se crean y gestionan los entornos de desarrollo, prueba y

producción, y se aplican las políticas de seguridad específicas del canal CI/CD. Las herramientas de automatización, como Jenkins, Travis CI o CircleCI, que automatizan la compilación, prueba y publicación de software, también se gestionan en esta OU.

En la OU CI/CD existen dos sub-unidades organizativas: Non-Prod-CICD y Prod-CICD. El canal Prod-CICD se utiliza para el despliegue en los tres entornos DEV, QA y STAGING, mientras que el canal Non-Prod-CICD sólo se utiliza para probar el canal y crear una versión de producción estable que se utilizará en el canal Prod-CICD. Esto ayuda a garantizar que la implantación sea precisa y que los cambios no causen problemas inesperados o tiempos de inactividad en el entorno de producción. Además, el uso de una canalización separada de no producción permite a los desarrolladores probar y validar los cambios antes de lanzarlos al entorno de producción, lo que reduce el riesgo de errores críticos y permite resolver cualquier problema antes de lanzarlos al entorno de producción.

3. **Playgrounds OU:** Esta OU está diseñada para permitir a los equipos de desarrollo explorar nuevas tecnologías y realizar pruebas sin comprometer la seguridad y fiabilidad de los recursos básicos de la organización. Aquí, los equipos pueden crear entornos de desarrollo y prueba independientes, que les permiten experimentar con nuevos marcos, bibliotecas y servicios sin riesgo de causar problemas en el sistema principal. En esta OU se aplican políticas de seguridad menos restrictivas que en las otras OUs, pero aún alineadas con las principales políticas de seguridad de la organización.

## 8.2.5 SEGUIMIENTO

Para la monitorización de toda la plataforma y la gestión de los recursos en la nube, se propone el uso del servicio AWS CloudWatch. El servicio permite a los usuarios supervisar y recopilar datos operativos, registros de aplicaciones y métricas de una amplia gama de fuentes.

La importancia de AWS CloudWatch para monitorizar los recursos informáticos en la nube es crucial, ya que permite a los usuarios monitorizar en tiempo real las métricas operativas y las actividades de todos los servicios de AWS que utilizan, así como recopilar y analizar datos de registro para poder detectar problemas de rendimiento, seguridad y gestión de recursos.

Las funcionalidades específicas que se sugiere implementar para la plataforma CDMS en República Dominicana son:

- **Métricas de monitorización:** CloudWatch ofrece una amplia gama de métricas de monitorización para todos los servicios de AWS, como la utilización de la CPU,

el uso de la memoria, el tiempo de respuesta de las solicitudes de API y muchas otras. Los usuarios pueden utilizar estas métricas para realizar un seguimiento del rendimiento de sus recursos de AWS e identificar cualquier problema de rendimiento.

- **Alarmas:** CloudWatch permite a los usuarios configurar alarmas para las métricas que monitorizan el rendimiento de los recursos de AWS. Las alarmas pueden configurarse para enviar notificaciones por correo electrónico, SMS u otras opciones cuando una métrica supera un umbral específico.
- **Registros de aplicaciones:** CloudWatch permite a los usuarios recopilar y analizar registros de aplicaciones de los recursos de AWS, lo que proporciona una visibilidad detallada de la actividad de las aplicaciones y el rendimiento del sistema. Estos datos pueden utilizarse para identificar posibles problemas de seguridad o de gestión de recursos.
- **Monitorización personalizada:** los usuarios también pueden configurar métricas de monitorización personalizadas para sus recursos de AWS utilizando sus propios scripts o aplicaciones. Esto permite a los usuarios supervisar las métricas específicas de sus necesidades operativas.

## 8.3 ASISTENCIA Y MANTENIMIENTO DEL SISTEMA

Simultáneamente al desarrollo, antes y después de la puesta en producción oficial<sup>[70]</sup>, es aconsejable diseñar e implantar un sistema de soporte que contemple la elaboración de documentación relevante para el uso de la plataforma, en forma de guías de uso del sistema para las distintas personas implicadas, incluyendo, si se considera necesario, un proceso de formación tanto para los empleados de las organizaciones productoras como para los empleados de la Administración, o cualquier persona que asuma el rol de administrador del sistema.

La documentación es un componente clave de cualquier sistema informático, y la creación de guías de usuario específicas para cada función puede ayudar a garantizar que todos los usuarios reciban la formación adecuada sobre el uso del sistema y puedan sacarle el máximo partido, utilizándolo de la mejor manera posible.

Un enfoque estructurado de la documentación y la formación también puede ayudar a reducir los costes de asistencia posteriores y minimizar el riesgo de errores o problemas al utilizar el sistema. Una documentación y una formación bien organizadas pueden ayudar a aumentar la satisfacción de los usuarios y la productividad, y al mismo tiempo garantizar que la transición del sistema actual a la futura plataforma CDMS dominicana sea lo más fluida posible, en beneficio de todas las partes implicadas.

---

<sup>70</sup> Capítulo 10

### 8.3.1 GUÍAS DEL USUARIO DE SOFTWARE

El diseño de una guía para el uso de programas informáticos de gestión de datos climáticos requiere un enfoque metódico y estructurado con el fin de proporcionar instrucciones detalladas y comprensibles a las distintas personas implicadas. La guía debe redactarse con un lenguaje claro y preciso, evitando el lenguaje técnico demasiado especializado y utilizando imágenes explicativas para facilitar la comprensión.

El documento debe estructurarse para proporcionar información específica a cada figura implicada en el sistema:

- **Usuarios finales sencillos:** la guía debe explicar cómo acceder al programa informático, explicar las funciones básicas del sistema y dar instrucciones sobre las operaciones que se pueden realizar. Además, la guía debe ilustrar con precisión cómo personalizar la experiencia del usuario.
- **Organizaciones de productores:** la guía debe ofrecer información completa sobre cómo introducir datos meteorológicos en las estaciones, explicar las funciones de seguimiento de las mediciones y proporcionar instrucciones detalladas sobre la gestión de los datos recogidos, desde la creación de las estaciones hasta la definición de los parámetros de medición, la gestión de la comunicación con el sistema central y el control de calidad de los datos recogidos.
- **Superadmin y sysadmin:** el documento debe proporcionar instrucciones específicas sobre la gestión del sistema, incluidos los permisos de acceso, la gestión de datos sensibles, la creación de cuentas de usuario y las funciones avanzadas del software. Los administradores de sistemas, por su parte, deben recibir aclaraciones sobre cómo instalar, configurar y mantener la infraestructura y el software.
- **Desarrolladores:** la guía debe presentar una visión completa de la estructura del sistema utilizado, las API disponibles, los procedimientos de desarrollo de aplicaciones y las mejores prácticas de programación.

En una fase posterior del proyecto, como alternativa o en paralelo a los documentos mencionados, también podría explorarse el uso de otras herramientas más innovadoras para guiar a los usuarios, como chatbots, guías contextuales o videoguías.

### 8.3.2 FORMACIÓN ADICIONAL

La creación de una guía puede no ser suficiente para garantizar el uso correcto y eficaz del software por parte de las distintas categorías de usuarios. Por este motivo, podría ser conveniente flanquear la guía con formación específica para cada figura implicada en el

sistema, con el fin de proporcionar formación teórica y práctica sobre la funcionalidad del software y, en la medida de lo posible, incluir ejercicios de simulación de las actividades que los usuarios tendrán que realizar en su trabajo diario.

En general, el objetivo de la formación debe ser garantizar que todos los usuarios del programa informático adquieran los conocimientos necesarios para utilizarlo correctamente y eficazmente, maximizando tanto la productividad como la facilidad de interacción. No obstante, las modalidades de esta formación deberán definirse en detalle en una fase posterior y más madura del proyecto.

### **8.3.3 ASISTENCIA Y NOTIFICACIÓN DE ERRORES**

En el marco del proyecto de desarrollo de la plataforma de gestión de datos hidrometeorológicos, está previsto ofrecer un servicio de notificación de errores en una fase posterior, es decir, después de que la plataforma se haya liberado para su uso fuera del desarrollo. Se considera que este servicio es esencial para que los usuarios puedan informar de cualquier problema que encuentren en el uso de la plataforma.

Para garantizar que los usuarios puedan utilizar la plataforma de forma eficaz y sin problemas, está previsto ofrecer un servicio de asistencia en el futuro. La consideración de los comentarios de los usuarios, sus opiniones y sugerencias, se considera un aspecto clave para mejorar constantemente la plataforma y ofrecer la mejor solución posible a sus necesidades.

## 9 COSTES

En este capítulo se detallarán los costes previstos para todos los desarrollos contemplados en el documento. La moneda elegida para las estimaciones de costes es el dólar estadounidense (USD). El análisis de costes es uno de los primeros pasos fundamentales en el diseño de un nuevo sistema informático. **Sin embargo, en esta fase, es importante tener en cuenta que el análisis debe excluir los costes de integración con plataformas y/o sistemas informáticos preexistentes pertenecientes a las entidades productoras. Ello se debe a que la estimación de estos costes depende en gran medida del análisis posterior a la posibilidad de acceso a las plataformas que deben integrarse, y de la evaluación de si es necesario o no integrarlas.**

### 9.1 INTEGRACIÓN DE LOS CASOS DE USO

La estimación de costes de desarrollo de software es una actividad crucial para garantizar el éxito de un proyecto. Entre las metodologías utilizadas para estimar los costes, una de las más extendidas es la de los Puntos de Caso de Uso (PCU). Los UCP representan un método de estimación de costes basado en la complejidad de los casos de uso<sup>[71]</sup>, es decir, las acciones específicas que un sistema de software debe ser capaz de realizar. Esta metodología se basa en el análisis de casos de uso, al que se asocia un valor numérico que representa la complejidad del propio caso de uso (Baldassarre et al., 2002).

Una vez establecidos los valores de todos los casos de uso del sistema, se calculó el total de Puntos de Caso de Uso, que se utilizó como base para estimar los costes de desarrollo del software. El cálculo de los UCP es una metodología muy eficaz para estimar los costes de desarrollo, ya que tiene en cuenta la complejidad real del sistema y permite estimar con precisión el tiempo y el coste necesarios para su implantación.

#### Cuadro 28. Costes de integración de los casos de uso en UCP.

Estimación de los costes previstos en UCP (Use Case Points) tanto en el caso de utilizar el front-end estándar como el front-end personalizado. El coste de las pruebas incluye tanto su desarrollo como su ejecución.

DESARROLLO	API DE COSTES Y CAPAS DE BACK-END (UCP)		COSTE FRONTAL ESTÁNDAR (UCP)		COSTE FRONTAL PERSONALIZADO (UCP)	
CASO 1: VISUALIZACIÓN E INTERACCIÓN CON EL MAPA	ANÁLISIS	3,125	ANÁLISIS	3,75	ANÁLISIS	3
	DESARROLLO	1,0375	DESARROLLO	1,25	DESARROLLO	9

<sup>71</sup> Descrito en el capítulo 3

DESARROLLO	API DE COSTES Y CAPAS DE BACK-END (UCP)		COSTE FRONTAL ESTÁNDAR (UCP)		COSTE FRONTAL PERSONALIZADO (UCP)	
	PRUEBA	2,0875	PRUEBA	2,5	PRUEBA	6
	<b>SUBTOTAL UC-01</b>	<b>6,25</b>	<b>SUBTOTAL UC-01</b>	<b>7,5</b>	<b>SUBTOTAL UC-01</b>	<b>18</b>
<b>CASO 2: DATOS DE LA ESTACIÓN</b>	ANÁLISIS	1,5625	ANÁLISIS	1,875	ANÁLISIS	3
	DESARROLLO	0,52125	DESARROLLO	0,625	DESARROLLO	3
	PRUEBA	1,04125	PRUEBA	1,25	PRUEBA	3
	<b>SUBTOTAL UC-02</b>	<b>3,125</b>	<b>SUBTOTAL UC-02</b>	<b>3,75</b>	<b>SUBTOTAL UC-02</b>	<b>9</b>
<b>CASO PRÁCTICO 3: ANÁLISIS DE DATOS DE ESTACIONES</b>	ANÁLISIS	2,5	ANÁLISIS	3,33	ANÁLISIS	4
	DESARROLLO	0,8375	DESARROLLO	3,33	DESARROLLO	12
	PRUEBA	1,6625	PRUEBA	3,33	PRUEBA	8
	<b>SUBTOTAL UC-03</b>	<b>5</b>	<b>SUBTOTAL UC-03</b>	<b>10</b>	<b>SUBTOTAL UC-03</b>	<b>24</b>
<b>CASO 4: ANÁLISIS DE DATOS VARIABLES</b>	ANÁLISIS	2,5	ANÁLISIS	3,4375	ANÁLISIS	5,5
	DESARROLLO	0,8375	DESARROLLO	1,14625	DESARROLLO	5,5
	PRUEBA	1,6625	PRUEBA	2,29125	PRUEBA	5,5
	<b>SUBTOTAL UC-04</b>	<b>5</b>	<b>SUBTOTAL UC-04</b>	<b>6,875</b>	<b>SUBTOTAL UC-04</b>	<b>16,5</b>
<b>CASO 5: EXPORTACIÓN DE DATOS</b>	ANÁLISIS	1,25	ANÁLISIS	0,625	ANÁLISIS	1,5
	DESARROLLO	0,4125	DESARROLLO	0,20875	DESARROLLO	0,5
	PRUEBA	0,8375	PRUEBA	0,41625	PRUEBA	1
	<b>SUBTOTAL UC-05</b>	<b>2,5</b>	<b>SUBTOTAL UC-05</b>	<b>1,25</b>	<b>SUBTOTAL UC-05</b>	<b>3</b>
<b>CASO PRÁCTICO 6: GESTIÓN DE ACCESOS DIRECTOS Y FILTROS GUARDADOS</b>	ANÁLISIS	3,5375	ANÁLISIS	2,1875	ANÁLISIS	3,5
	DESARROLLO	3,5375	DESARROLLO	0,72875	DESARROLLO	3,5
	PRUEBA	3,5375	PRUEBA	1,45875	PRUEBA	3,5
	<b>SUBTOTAL UC-06</b>	<b>10,625</b>	<b>SUBTOTAL UC-06</b>	<b>4,375</b>	<b>SUBTOTAL UC-06</b>	<b>10,5</b>
<b>CASO PRÁCTICO 7: GESTIÓN DE LAS ESTACIONES EN FUNCIÓN DE LA ENTIDAD PRODUCTORA</b>	ANÁLISIS	3,64625	ANÁLISIS	3,5416	ANÁLISIS	4,25
	DESARROLLO	10,9375	DESARROLLO	3,5416	DESARROLLO	12,75
	PRUEBA	7,29125	PRUEBA	3,5416	PRUEBA	8,5
	<b>SUBTOTAL UC-07</b>	<b>21,875</b>	<b>SUBTOTAL UC-07</b>	<b>10,625</b>	<b>SUBTOTAL UC-07</b>	<b>25,5</b>



DESARROLLO	API DE COSTES Y CAPAS DE BACK-END (UCP)		COSTE FRONTAL ESTÁNDAR (UCP)		COSTE FRONTAL PERSONALIZADO (UCP)	
<b>CASO PRÁCTICO 8:</b> CONTROL DE CALIDAD Y VALIDACIÓN DE DATOS	ANÁLISIS	1,25	ANÁLISIS	2,5	ANÁLISIS	4
	DESARROLLO	0,4125	DESARROLLO	0,8375	DESARROLLO	4
	PRUEBA	0,8375	PRUEBA	1,6625	PRUEBA	4
	<b>SUBTOTAL UC-08</b>	<b>2,5</b>	<b>SUBTOTAL UC-08</b>	<b>5</b>	<b>SUBTOTAL UC-08</b>	<b>12</b>
<b>CASO PRÁCTICO 9:</b> IMPORTACIÓN DE DATOS HIDROMETEOCLIMÁTICOS	ANÁLISIS	2,8125	ANÁLISIS	3,4375	ANÁLISIS	5,5
	DESARROLLO	0,9375	DESARROLLO	1,14625	DESARROLLO	5,5
	PRUEBA	1,875	PRUEBA	2,29125	PRUEBA	5,5
	<b>SUBTOTAL UC-09</b>	<b>5,625</b>	<b>SUBTOTAL UC-09</b>	<b>6,875</b>	<b>SUBTOTAL UC-09</b>	<b>16,5</b>
<b>CASO 10:</b> GESTIÓN DE ENTIDADES PRODUCTORAS	ANÁLISIS	3,64625	ANÁLISIS	3,5416	ANÁLISIS	4,25
	DESARROLLO	10,9375	DESARROLLO	3,5416	DESARROLLO	12,75
	PRUEBA	7,29125	PRUEBA	3,5416	PRUEBA	8,5
	<b>SUBTOTAL UC-10</b>	<b>21,875</b>	<b>SUBTOTAL UC-10</b>	<b>10,625</b>	<b>SUBTOTAL UC-10</b>	<b>25,5</b>
<b>CASOS PRÁCTICOS DE INTEGRACIÓN TOTAL</b>	ANÁLISIS SUBTOTAL	25,83	ANÁLISIS SUBTOTAL	27,5625	ANÁLISIS SUBTOTAL	38,5
	SUBTOTAL DESARROLLO	30,475	SUBTOTAL DESARROLLO	16,355	SUBTOTAL DESARROLLO	68,5
	SUBTOTAL PRUEBA	28,0	SUBTOTAL PRUEBA	22,9575	SUBTOTAL PRUEBA	53,5
	<b>TOTAL</b>	<b>84,375</b>	<b>TOTAL</b>	<b>66,875</b>	<b>TOTAL</b>	<b>160,5</b>

La conversión entre Use Case Points (UCP) y coste por Use Case Point en dólares depende de varios factores, como el tipo de proyecto, la complejidad de los casos de uso y el nivel de experiencia y conocimientos del equipo de desarrollo. En general, el coste por PCU puede calcularse dividiendo el coste total del proyecto por el número de PCU. Por ejemplo, si el coste total del proyecto es de 100.000,00 USD y el número de PCU es de 500, el coste por PCU es, por división trivial, de 200,00 USD.

En el caso de este proyecto, se tomó como referencia el caso de uso 2 para estimar el coste en dólares por UCP. A continuación, el coste de desarrollo se dividió por el número de OPC y el valor resultante, ligeramente superior a 600,00 USD, se redondeó a la baja y se multiplicó por los Puntos de Caso de Uso de cada caso de uso para obtener la estimación final de costes.

### Cuadro 29. Costes de integración de los casos de uso en USD.

Costes estimados en USD (dólares estadounidenses) tanto para el front-end estándar como para el personalizado. El coste de las pruebas incluye tanto su desarrollo como su ejecución.

DESARROLLO	COSTE BACK-END Y CAPA API (USD)		COSTE FRONTAL ESTÁNDAR (USD)		COSTE DEL FRONT-END PERSONALIZADO (USD)	
<b>CASO 1:</b> VISUALIZACIÓN E INTERACCIÓN CON EL MAPA	ANÁLISIS	1.875,00	ANÁLISIS	2.250,00	ANÁLISIS	1.800,00
	DESARROLLO	625,00	DESARROLLO	750,00	DESARROLLO	5.400,00
	PRUEBA	1.250,00	PRUEBA	1.500,00	PRUEBA	3.600,00
	<b>SUBTOTAL UC-01</b>	<b>3.750,00</b>	<b>SUBTOTAL UC-01</b>	<b>4.500,00</b>	<b>SUBTOTAL UC-01</b>	<b>10.800,00</b>
<b>CASO 2:</b> DATOS DE LA ESTACIÓN	ANÁLISIS	937,50	ANÁLISIS	1.125,00	ANÁLISIS	1.800,00
	DESARROLLO	312,50	DESARROLLO	375,00	DESARROLLO	1.800,00
	PRUEBA	625,00	PRUEBA	750,00	PRUEBA	1.800,00
	<b>SUBTOTAL UC-02</b>	<b>1.875,00</b>	<b>SUBTOTAL UC-02</b>	<b>2.250,00</b>	<b>SUBTOTAL UC-02</b>	<b>5.400,00</b>
<b>CASO PRÁCTICO 3:</b> ANÁLISIS DE DATOS DE ESTACIONES	ANÁLISIS	1.500,00	ANÁLISIS	2.000,00	ANÁLISIS	2.400,00
	DESARROLLO	500,00	DESARROLLO	2.000,00	DESARROLLO	7.200,00
	PRUEBA	1.000,00	PRUEBA	2.000,00	PRUEBA	4.800,00
	<b>SUBTOTAL UC-03</b>	<b>3.000,00</b>	<b>SUBTOTAL UC-03</b>	<b>6.000,00</b>	<b>SUBTOTAL UC-03</b>	<b>14.400,00</b>
<b>CASO 4:</b> ANÁLISIS DE DATOS VARIABLES	ANÁLISIS	1.500,00	ANÁLISIS	2.062,50	ANÁLISIS	3.300,00
	DESARROLLO	500,00	DESARROLLO	687,50	DESARROLLO	3.300,00
	PRUEBA	1.000,00	PRUEBA	1.375,00	PRUEBA	3.300,00
	<b>SUBTOTAL UC-04</b>	<b>3.000,00</b>	<b>SUBTOTAL UC-04</b>	<b>4.125,00</b>	<b>SUBTOTAL UC-04</b>	<b>9.900,00</b>
<b>CASO 5:</b> EXPORTACIÓN DE DATOS	ANÁLISIS	750,00	ANÁLISIS	375,00	ANÁLISIS	900,00
	DESARROLLO	250,00	DESARROLLO	125,00	DESARROLLO	300,00
	PRUEBA	500,00	PRUEBA	250,00	PRUEBA	600,00
	<b>SUBTOTAL UC-05</b>	<b>1.500,00</b>	<b>SUBTOTAL UC-05</b>	<b>750,00</b>	<b>SUBTOTAL UC-05</b>	<b>1.800,00</b>
<b>CASO PRÁCTICO 6:</b> GESTIÓN DE ACCESOS DIRECTOS Y FILTROS GUARDADOS	ANÁLISIS	2.125,00	ANÁLISIS	1.312,50	ANÁLISIS	2.100,00
	DESARROLLO	2.125,00	DESARROLLO	437,50	DESARROLLO	2.100,00
	PRUEBA	2.125,00	PRUEBA	875,00	PRUEBA	2.100,00

DESARROLLO	COSTE BACK-END Y CAPA API (USD)		COSTE FRONTAL ESTÁNDAR (USD)		COSTE DEL FRONT-END PERSONALIZADO (USD)	
	SUBTOTAL UC-06	6.375,00	SUBTOTAL UC-06	2.625,00	SUBTOTAL UC-06	6.300,00
<b>CASO PRÁCTICO 7:</b> GESTIÓN DE LAS ESTACIONES EN FUNCIÓN DE LA ENTIDAD PRODUCTORA	ANÁLISIS	2.187,50	ANÁLISIS	2.125,00	ANÁLISIS	2.550,00
	DESARROLLO	6.562,50	DESARROLLO	2.125,00	DESARROLLO	7.650,00
	PRUEBA	4.375,00	PRUEBA	2.125,00	PRUEBA	5.100,00
	<b>SUBTOTAL UC-07</b>	<b>13.125,00</b>	<b>SUBTOTAL UC-07</b>	<b>6.375,00</b>	<b>SUBTOTAL UC-07</b>	<b>15.300,00</b>
<b>CASO PRÁCTICO 8:</b> CONTROL DE CALIDAD Y VALIDACIÓN DE DATOS	ANÁLISIS	750,00	ANÁLISIS	1.500,00	ANÁLISIS	2.400,00
	DESARROLLO	250,00	DESARROLLO	500,00	DESARROLLO	2.400,00
	PRUEBA	500,00	PRUEBA	1.000,00	PRUEBA	2.400,00
	<b>SUBTOTAL UC-08</b>	<b>1.500,00</b>	<b>SUBTOTAL UC-08</b>	<b>3.000,00</b>	<b>SUBTOTAL UC-08</b>	<b>7.200,00</b>
<b>CASO PRÁCTICO 9:</b> IMPORTACIÓN DE DATOS HIDROMETEOCLIMÁTICOS	ANÁLISIS	1.687,50	ANÁLISIS	2.062,50	ANÁLISIS	3.300,00
	DESARROLLO	562,50	DESARROLLO	687,50	DESARROLLO	3.300,00
	PRUEBA	1.125,00	PRUEBA	1.375,00	PRUEBA	3.300,00
	<b>SUBTOTAL UC-09</b>	<b>3.375,00</b>	<b>SUBTOTAL UC-09</b>	<b>4.125,00</b>	<b>SUBTOTAL UC-09</b>	<b>9.900,00</b>
<b>CASO 10:</b> GESTIÓN DE ENTIDADES PRODUCTORAS	ANÁLISIS	2.187,50	ANÁLISIS	2.125,00	ANÁLISIS	2.550,00
	DESARROLLO	6.562,50	DESARROLLO	2.125,00	DESARROLLO	7.650,00
	PRUEBA	4.375,00	PRUEBA	2.125,00	PRUEBA	5.100,00
	<b>SUBTOTAL UC-10</b>	<b>13.125,00</b>	<b>SUBTOTAL UC-10</b>	<b>6.375,00</b>	<b>SUBTOTAL UC-10</b>	<b>15.300,00</b>
<b>CASOS PRÁCTICOS DE INTEGRACIÓN TOTAL</b>	ANÁLISIS SUBTOTAL	15.500,00	ANÁLISIS SUBTOTAL	16.562,50	ANÁLISIS SUBTOTAL	23.100,00
	SUBTOTAL DESARROLLO	18.287,50	SUBTOTAL DESARROLLO	9.812,50	SUBTOTAL DESARROLLO	41.100,00
	SUBTOTAL PRUEBA	16.837,50	SUBTOTAL PRUEBA	13.750,00	SUBTOTAL PRUEBA	34.500,00
	<b>TOTAL</b>	<b>50.625,00</b>	<b>TOTAL</b>	<b>40.125,00</b>	<b>TOTAL</b>	<b>98.700,00</b>

## 9.2 ARQUITECTURA DE APLICACIONES

Para que la estimación de costes del diseño y la implementación de la infraestructura fuera lo más fiable posible en esta fase, se contrató a un especialista en AWS.

**Cuadro 30. Costes de implantación de la infraestructura en USD.**

DESARROLLO	COSTE (USD)
DISEÑO DE ARQUITECTURA DE BAJO NIVEL	15.000,00
SUPERVISIÓN DE LA APLICACIÓN	5.000,00
DESARROLLO Y PRUEBAS DE INFRAESTRUCTURAS	65.000,00
<b>TOTAL</b>	<b>85.000,00</b>

Los costes de ejecución de la infraestructura son el resultado del cálculo automático de la Calculadora de AWS<sup>[72]</sup>, con los parámetros establecidos cumpliendo íntegramente con lo descrito y detallado en el Capítulo 8.

**Cuadro 31. Costes de ejecución de las infraestructuras en USD.**

DESARROLLO	REGIÓN	COMISIONES INICIALES (USD)	COSTE MENSUAL (USD)
AMAZON EKS	Este de EE. UU. (Norte de Virginia)	0,00	584,00
CORTAFUEGOS DE APLICACIONES WEB DE AWS (WAF)	Este de EE. UU. (Norte de Virginia)	0,00	150,00
NUBE PRIVADA VIRTUAL (VPC) DE AMAZON	Este de EE. UU. (Norte de Virginia)	0,00	315,72
AMAZON EC2	Este de EE. UU. (Norte de Virginia)	249,66	0,00

<sup>72</sup> Documento disponible [aquí](#)

DESARROLLO	REGIÓN	COMISIONES INICIALES (USD)	COSTE MENSUAL (USD)
REGISTRO DE CONTENEDORES ELÁSTICOS DE AMAZON	Este de EE. UU. (Norte de Virginia)	0,00	1,00
AMAZON COGNITO	Este de EE. UU. (Norte de Virginia)	0,00	50,75
AMAZON CLOUDWATCH	Este de EE. UU. (Norte de Virginia)	0,00	12,00
PASARELA API DE AMAZON	Este de EE. UU. (Norte de Virginia)	0,00	350,00
SISTEMA DE ARCHIVOS ELÁSTICO DE AMAZON (EFS)	Este de EE. UU. (Norte de Virginia)	0,00	409,60
IP ELÁSTICA DE AMAZON	Este de EE. UU. (Norte de Virginia)	0,00	7,30
AWS FARGATE	Este de EE. UU. (Norte de Virginia)	0,00	576,63
<b>TOTAL</b>	<b>n/a</b>	<b>249,66</b>	<b>2.457,00</b>

### 9.3 ESCENARIOS DE INTEGRACIÓN Y ASISTENCIA AL USUARIO

La última parte del análisis de costes se llevó a cabo internamente utilizando la experiencia del equipo de desarrollo, excepto en el caso de la estimación de costes de la capa caché, para la que se recurrió a un consultor externo. En cuanto a la parte de realización de guías de usuario y formación, se remite a lo descrito en los apartados **8.3.1** y **8.3.2**. Como se especifica en el párr. **5.6.2.3** todavía no se ha implantado un sistema

público de registro gratuito de usuarios, y su integración también figura entre los costes estimados en este apartado. Por último, pero no menos importante, está la planificación e implementación de las actividades de pruebas de penetración, mencionadas en el párrafo **8.2.3.4**.

**Cuadro 32. Costes de los escenarios de integración y asistencia al usuario en USD.**

DESARROLLO	COSTE (USD)
INVESTIGACIÓN Y DESARROLLO	40.000,00
CAPA DE CACHÉ PARA CONSULTAR BASES DE DATOS DE TERCEROS SIN ALMACENAMIENTO	80.000,00
SISTEMA DE REGISTRO DE SERVICIOS PÚBLICOS	15.000,00
INTEGRACIÓN ENTRE DJANGO Y COGNITO	10.000,00
CREACIÓN DE GUÍAS DEL USUARIO	12.000,00
REALIZACIÓN DE FORMACIÓN DE USUARIOS (32 HORAS DE FORMACIÓN INTERACTIVA EN VÍDEO)	5.000,00
PRUEBAS DE PENETRACIÓN (AL MENOS 10 DÍAS DE ACTIVIDAD DE PRUEBA DE PENETRACIÓN ANTES DE QUE EL SISTEMA ENTRE EN VIVO)	15.000,00

## 9.4 CUADRO RESUMEN

**Cuadro 33. Cuadro recapitulativo de los costes en USD.**

DESARROLLO	COSTE (USD)
BACK-END Y CAPA API	50.625,00
FRONT-END ESTÁNDAR (Opción 1)	40.125,00
FRONT-END A MEDIDA (Opción 2)	98.700,00
REALIZACIÓN DE INFRAESTRUCTURAS	85.000,00
INVESTIGACIÓN Y DESARROLLO (Opcional – recomendado al menos en una etapa posterior)	40.000,00
CACHE LAYER PARA INTERROGACIÓN DE BASE DE DATOS DE TERCEROS SIN ALMACENAMIENTO (Opcional)	80.000,00
SISTEMA DE REGISTRO DE SERVICIOS PÚBLICOS	15.000,00

INTEGRACIÓN ENTRE DJANGO Y COGNITO	10.000,00
CREACIÓN DE GUÍAS DEL USUARIO	12.000,00
REALIZACIÓN DE FORMACIÓN DE USUARIOS (32 HORAS DE FORMACIÓN INTERACTIVA EN VÍDEO)	5.000,00
PRUEBAS DE PENETRACIÓN (AL MENOS 10 DÍAS DE ACTIVIDAD DE PRUEBA DE PENETRACIÓN ANTES DE QUE EL SISTEMA ENTRE EN VIVO)	15.000,00

## 10 ESCENARIO DE APLICACIÓN DEL PLAN DE TRABAJO

En este capítulo, se describirán los motivos para elegir un escenario específico de entre las opciones sugeridas. Además, se presentarán las macrotareas que constituyen el plan de trabajo y se ilustrará un diagrama de Gantt, en el que figurará el calendario de todas las actividades previstas, con indicación de los plazos y recursos necesarios para cada una de ellas. Por último, se presentará la tabla de entregables previstos, es decir, la lista de entregables que el equipo de desarrollo tendrá que realizar para completar el proyecto dentro del plazo previsto.

### 10.1 HIPÓTESIS PROPUESTA

El escenario presentado es el resultado de un minucioso análisis de los requisitos y objetivos del proyecto recogidos durante la fase de estudio de la plataforma. De hecho, se considera que, sobre la base de toda la información recopilada hasta el momento, este escenario es la opción más adecuada para alcanzar los objetivos fijados y garantizar la satisfacción del usuario final.

En concreto, resumiremos a continuación una serie de macrofunciones, desarrollos y actividades (descritas en detalle en el Capítulo 5) que se consideran beneficiosas para los fines del proyecto y que, por tanto, se decidió incluir en el plan de trabajo. En las descripciones, se evitarán los desarrollos clasificados como obligatorios en el capítulo 5.

Un desarrollo clasificado como opcional, pero que se considera extremadamente funcional a efectos del proyecto, es el desarrollo de la interfaz personalizada relativa a la capa front-end. Durante la fase de estudio se detectaron varias deficiencias en el front-end SURFACE actualmente disponible. En concreto, la aplicación no es *receptiva* y la experiencia del usuario suele ser engorrosa. Por otro lado, el front-end personalizado se desarrollaría en modo *responsive by design*, y con una experiencia de usuario notablemente mejorada que, como suele ocurrir con las capas de presentación (front-end), se traduciría en una tarjeta de visita y una experiencia positivas para todos los usuarios del servicio.

La otra funcionalidad que se considera útil integrar en la plataforma es la posibilidad de registro público para los usuarios finales que consulten los datos, aspecto que aportaría varias ventajas:



- Posibilidad de que los usuarios guarden extracciones y las asocien a su perfil de usuario.
- Posibilidad de que los operadores de la plataforma sean más conscientes de cómo utilizan el sistema los usuarios.
- Posibilidad de que los operadores de la plataforma estén en contacto directo con sus usuarios por todas las razones que sean necesarias (comunicar el mantenimiento del sistema, notificar a los usuarios eventos de interés e iniciativas, etc.).

En lo que respecta a las integraciones de entrada de datos, las únicas funcionalidades de integración que se consideran útiles actualmente, y que están contempladas en el plan de trabajo, son la capa API, el servidor FTP. Actualmente no es necesario contemplar las otras modalidades propuestas.

Además, para un proyecto de esta magnitud, es recomendable considerar la inclusión de la línea de investigación y desarrollo propuesta. Esta inversión proporcionaría dos importantes beneficios: en primer lugar, la creación de una base de innovación que garantice la evolución continua del sistema, manteniéndolo constantemente actualizado y alineado con las tendencias del mercado; en segundo lugar, el establecimiento de una red de relaciones con universidades e institutos de investigación con el objetivo de lograr los mismos objetivos y crear un efecto de resonancia a nivel internacional para el proyecto.

Para la construcción del plan de trabajo no se tuvo en cuenta la capa de caché. A pesar de presentarse como un desarrollo opcional, según la información recopilada hasta la fecha, no parece ser un desarrollo necesario para la prestación de servicios. De hecho, no se encontraron problemas críticos específicos en relación con el intercambio de datos climáticos que hicieran útil este enfoque.

## **10.2 TAREAS DEL PLAN DE TRABAJO**

El plan de trabajo (**Cuadro 34**) define las macroactividades del proyecto situándolas en un periodo global de 12 meses. Muchas de las macroactividades se solapan temporalmente en el plan de trabajo, ya que, como se describe en la sección **1.3.4.1**, la metodología de desarrollo adoptada será un enfoque ágil iterativo, y no en cascada.

**Cuadro 34. Plan de trabajo**

TAREA	DESCRIPCIÓN	OBJETIVOS	MES DE INICIO DE LAS OBRAS	MES DE FINALIZACIÓN
T1 - GESTIÓN DE PROYECTOS	Planificación y seguimiento de las actividades, desde un punto de vista funcional y técnico. Facilitar el proceso de comunicación entre las distintas partes interesadas.	Planificación general de proyectos, programación, seguimiento y elaboración de informes.	M1	M12
T2 - ANÁLISIS EN PROFUNDIDAD DE LA SOLICITUD	Definir análisis funcionales y técnicos detallados (diseño de bajo nivel) que se transmitirán a los equipos de desarrollo para su aplicación.	Creación de documentos de análisis funcional y técnico.	M1	M6
T3 - DESARROLLO FRONT-END PERSONALIZADO	Desarrollar el código necesario para la implementación de la capa de front-end.	Desarrollo, despliegue y pruebas funcionales de la aplicación Angular.	M2	M8
T4 - DESARROLLO BACK-END	Desarrollar el código necesario para la implementación de la capa back-end.	Desarrollo, despliegue y pruebas funcionales de la aplicación Surface Django.	M2	M8
T5 - DISEÑO DE INFRAESTRUCTURAS DE BAJO NIVEL	Definir el análisis de bajo nivel de la arquitectura de la infraestructura.	Elaboración de un documento de diseño de infraestructura de bajo nivel.	M1	M2
T6 - IMPLANTACIÓN DE INFRAESTRUCTURAS	Implantar la infraestructura definida en el análisis de bajo nivel.	Desarrollo y despliegue de la infraestructura en la nube.	M2	M7
T7 - MÓDULO BLOCKCHAIN/AI DE I+D	Iniciar el estudio de investigación para identificar una posible solución para la calidad de los datos mediante la aplicación de tecnologías innovadoras.	Elaboración de trabajos de investigación.	M3	M5
T8 - I+D MÓDULO BLOCKCHAIN/AI POC	Implementar el PoC del módulo blockchain/AI para la calidad de los datos.	Aplicación de la PdC	M6	M9

TAREA	DESCRIPCIÓN	OBJETIVOS	MES DE INICIO DE LAS OBRAS	MES DE FINALIZACIÓN
T9 - DESARROLLO DE INTEGRACIONES	Desarrollar integraciones para la recopilación de datos de estaciones y plataformas externas.	Desarrollo, despliegue y prueba de integraciones para el suministro de datos externos.	M4	M8
T10 - PRUEBAS DE INTEGRACIÓN	Integraciones de prueba para la recopilación de datos de estaciones y plataformas externas.	Visualización de los primeros datos desde el exterior mediante la plataforma.	M7	M12
T11 - PRUEBAS DE RENDIMIENTO Y SEGURIDAD	Pruebe la capacidad de la plataforma para manejar grandes cargas de trabajo y mantener el rendimiento deseado en condiciones de uso del mundo real. Probar las capacidades de seguridad de la plataforma.	Obtener una plataforma capaz de prestar el servicio incluso en condiciones de carga, y que respete todos los requisitos de seguridad establecidos	M8	M11
T12 - DOCUMENTACIÓN Y FORMACIÓN	Redacción de documentación para usuarios y operadores de plataformas. Inicio de las actividades de formación.	Elaboración de guías y documentos. Programas de formación.	M9	M13

## 10.2.1 GANTT

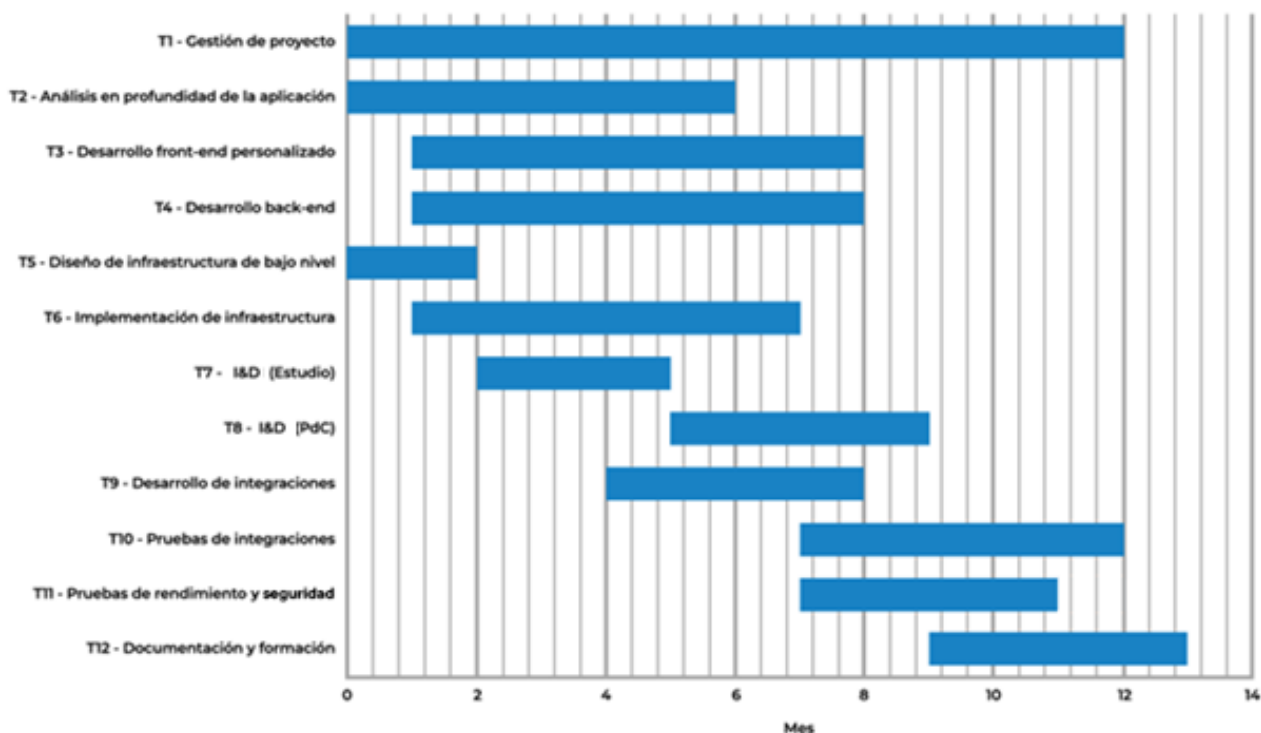


Figura 80. Diagrama de Gantt

## 10.4 ENTREGABLE

Cuadro 35. Resultados del proyecto

ID	DESCRIPCIÓN	MES DE ENTREGA
D1	Elaboración de informes bimensuales sobre las actividades del proyecto	M2, M4, M6, M8, M10, M12
D2.1	Entorno de pruebas con posibilidad de verificación de las primeras funcionalidades implementadas por un subconjunto predefinido de usuarios finales del sistema.	M3
D3	Documento de análisis de infraestructuras de bajo nivel	M3
D4.1	Documento de I+D sobre la calidad de los datos	M6
D2.2	Primera instalación en un entorno de producción	M7
D4.2	Demo PoC I+D	M10

D5	Informe de las pruebas de rendimiento y seguridad	M12
D6	Documentación y guías	M12

## 12 BIBLIOGRAFÍA

AFD. (2021). *Apoyo al sector Agrícola de la República Dominicana en un contexto de cambio climático (Componente 2). Evaluación de la situación de los servicios climáticos para la agricultura en República Dominicana*. SalvaTerra, EGIS, IRD, Meteodyn, Guakía Ambiente, Fundación REDDOM, y Carbonium. Santo Domingo.

Baldassarre, M. T., Caivano, D., & Visaggio, G. (2002). *Estimating software development effort through use case points*. *Information and Software Technology*, 44(14), 869-877. doi: 10.1016/S0950-5849(02)00090-0

Barrett, T., Keenan, M., & Martinez, J. (2012). *Open-source software for environmental data management and analysis*. In *Environmental Software Systems. Frameworks of eEnvironment*. Springer, Berlin, Heidelberg.

Booch, G., Rumbaugh, J., & Jacobson, I. (2007). *The unified modeling language user guide*. Addison-Wesley Professional.

Budapest Open Access Initiative. (2002, accessed 2023). *Budapest open access initiative*. Retrieved from <http://www.soros.org/openaccess/read.shtml>

Camp, R. C. (2005). *Benchmarking: The search for industry best practices that lead to superior performance*. ASQ Quality Press.

Casado-Vara, R., de la Prieta, F., Prieto, J., & Corchado, J. M. (2018). *Blockchain framework for IoT data quality via edge computing*. In *Proceedings of the 1st Workshop on Blockchain-enabled Networked Sensor Systems (BlockSys'18)*, November 4, 2018, Shenzhen, China (pp. 39-44). ACM. Retrieved from: <https://doi.org/10.1145/3282278.3282282>

Clarkson, M. (1995). *A stakeholder framework for analyzing and evaluating corporate social performance*. *Academy of Management Review*, 20(1), 92-117.

Cockburn, A. (2000). *Writing effective use cases*. Addison-Wesley Professional.

Date, C. J. (2018). *An introduction to database systems*. Addison-Wesley.

Dimitrov, N., Bölükbaşı, Ö., & Alkan Olsson, J. (2020, accessed 2023). *Comparison of different open source software for data management and analysis of climate and hydrological data*. *Environmental Modelling & Software*, 127, 104651. Retrieved from <https://doi.org/10.1016/j.envsoft.2020.104651>

- Duan, C., & Huang, J. (2021). *An Empirical Comparison of Programming Languages for Data Science Tasks*. *Journal of Data and Information Science*, 6(2), 20-29.
- ECMWF. (2016, accessed 2023). *Copernicus Climate Change Service User Requirements Document*. Retrieved from [https://climate.copernicus.eu/sites/default/files/2020-02/C3S\\_UserRequirementsDocument\\_2016.pdf](https://climate.copernicus.eu/sites/default/files/2020-02/C3S_UserRequirementsDocument_2016.pdf)
- Elmasri, R., & Navathe, S. B. (2019). *Fundamentals of database systems*. Pearson.
- European Commission. (2016, accessed 2023). *Open Data*. Retrieved from <https://ec.europa.eu/digital-single-market/en/open-data>
- Florido-Benítez, M., Sánchez-Fernández, L., García-Peñalvo, F. J., & Therón, R. (2020). *Web-Based Viewer for Climate Data Accessibility*. *Sensors*, 20(7), 1922. Retrieved from <https://doi.org/10.3390/s20071922>
- Fujimoto, M., & Ota, K. (2018, accessed 2023). *Towards a Comprehensive Approach to Software Accessibility*. In *Proceedings of the 13th International Conference on Software Technologies (ICSOFT 2018)* (pp. 391-398). Retrieved from <https://doi.org/10.5220/0006856803910398>
- Freeman, R. E. (2010). *Strategic management: A stakeholder approach*. Cambridge University Press.
- Gacek, C., & Arief, B. (2019). *Nonfunctional Requirements*. In *Handbook of Software Engineering* (pp. 283-310). Springer.
- Ganti, R. K., Ye, F., & Lei, H. (2016, accessed 2023). *Mobile crowdsensing: current state and future challenges*. *IEEE Communications Magazine*, 54(7), 141-147. Retrieved from <https://doi.org/10.1109/MCOM.2016.7502248>
- Garcia-Molina, H., Ullman, J. D., & Widom, J. (2009). *Database systems: The complete book*. Pearson.
- Giraud, R., Tandy, J., Eyre, J., Spears, T., Kralidis, T., Varley, R., & Fucile, E. (2021, accessed 2023) *Modernizing Data Exchange for Earth System Monitoring and Prediction* (online edition). Retrieved from <https://public.wmo.int/en/resources/bulletin/modernizing-data-exchange-earth-system-monitoring-and-prediction>
- Gonzalez, R., Kitchin, R., & Perkins, C. (2015, accessed 2023). *Open and Big Data in Europe: A Growth Engine or a Missed Opportunity?* *The Spatial Journal*, 29(2), 107-115. Retrieved from <https://doi.org/10.1080/2373566X.2015.1063393>

Guo, S., Song, H., Chen, Z., & Guo, L. (2019). *Scalable and Efficient Data Analytics for Internet of Things Using FaaS*. IEEE Internet of Things Journal, 6(1), 824-834.

Gupta, M., & Bajaj, V. (2019). *Overview of Software Accessibility*. In *Proceedings of the 6th International Conference on Computing for Sustainable Global Development (INDIACom 2019)* (pp. 372-376). Retrieved from <https://doi.org/10.1109/INDIACom.2019.8726863>

Höft, N., Stasch, C., Koziol, A., Reussner, R., & Schneider, K. (2018). *Open hardware in scientific research: An assessment of status quo and prospects for the future*. PloS one, 13(7), e0200933.

Hossain, M. S., Kowalkowski, C., Islam, M. M., & Meena, P. K. (2020). *A comparative analysis of consumer preference for online shopping platforms in the United States*. Journal of Retailing and Consumer Services, 57, 102179.

IEEE Standards Association. (1998). *IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications*.

IPCC. (2018, accessed 2023). *Global Warming of 1.5°C. An IPCC Special Report on the impacts of global warming of 1.5°C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty*. Retrieved from <https://www.ipcc.ch/sr15/>

Johnson, G., Scholes, K., & Whittington, R. (2008). *Exploring corporate strategy: Text and cases*. Pearson Education.

Joint Committee for Guides in Metrology (JCGM). (2012, accessed 2023). *International Vocabulary of Metrology – Basic and General Concepts and Associated Terms (VIM)* (Third edition, JCGM 200:2012). Retrieved from [http://www.bipm.org/utils/common/documents/jcgm/JCGM\\_200\\_2012.pdf](http://www.bipm.org/utils/common/documents/jcgm/JCGM_200_2012.pdf)

Kotonya, G., & Sommerville, I. (1998). *Requirements engineering: Processes and techniques*. John Wiley & Sons.

Laplante, P. A. (2017). *What Every Engineer Should Know About Software Engineering*. CRC Press.

Laureano-Cruces, A. L. (2020, accessed 2023). *Design and development of accessible software for people with disabilities*. Journal of Software Engineering and Applications, 13(8), 450-463. Retrieved from <https://doi.org/10.4236/jsea.2020.138031>



Lazar, J. (2019, accessed 2023). *Web accessibility: A foundation for research*. Synthesis Lectures on Assistive, Rehabilitative, and Health-Preserving Technologies, 10(3), 1-205. <https://doi.org/10.2200/S00958ED1V01Y201901ARH017>

Levene, M. (2017). *An introduction to search engines and web navigation*. John Wiley & Sons.

Liu, Q., & Tretter, D. (2015). *Assessing Programming Languages for Data Science*. Journal of Computational and Graphical Statistics, 24(1), 1-11.

Lopez-Gomez, I., Palacios-Diaz, M. P., & Garcia-Sanchez, F. (2019, accessed 2023). *Open source software adoption in firms: a model of technology acceptance with two different perspectives*. Information Systems Frontiers, 21(5), 1087–1102. Retrieved from <https://doi.org/10.1007/s10796-018-9843-y>

Lyytinen, K., & Kautz, K. (2019). *Non-functional Requirements in Software Engineering*. In *The Essence of Software Engineering* (pp. 135-153). Springer.

Marr, B. (2018). *Artificial Intelligence And Blockchain: 3 Major Benefits Of Combining These Two Mega-Trends*. Forbes. Retrieved from: <https://www.forbes.com/sites/bernardmarr/2018/03/02/artificial-intelligence-and-blockchain-3-major-benefits-of-combining-these-two-mega-trends/?sh=33a62ef44b44>

Menne, M. J., Williams Jr, R. C., & Gallagher, N. (2010). *The Effect of Data Quality on the Stability of Climate Classifications*. Journal of Climate, 23(8), 2000-2013. Retrieved from: <https://doi.org/10.1175/2009jcli3227.1>

Meteoblue. (accessed 2023). Distrito Nacional, Republica Dominicana, 18.47°N 69.89°O, 24m slm. Cambio climático Santo Domingo. Retrieved from: [https://www.meteoblue.com/it/climate-change/santo-domingo\\_repubblica-dominicana\\_3492908](https://www.meteoblue.com/it/climate-change/santo-domingo_repubblica-dominicana_3492908)

Mitchell, R. K., Agle, B. R., & Wood, D. J. (1997). *Toward a theory of stakeholder identification and salience: Defining the principle of who and what really counts*. Academy of Management Review, 22(4), 853-886.

Nag, A., Das, S., & Dutta, A. (2017). *Scalable and efficient data management for internet of things using AWS Lambda*. International Journal of Communication Systems, 30(1), e2957.

National Oceanic and Atmospheric Administration (NOAA). (2020, accessed 2023). *OpenCDMS: An open source climate data management system*. Retrieved from <https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/opencdms>

Nielsen, J. (2012, accessed 2023). *Usability 101: Introduction to usability*. Nielsen Norman Group. Retrieved from <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>

OpenCDMS Project (accessed 2023) *OpenCDMS*. Retrieved from <https://www.opencdms.org>

Patil, S., Banerjee, N., & D'Mello, S. K. (2019, accessed 2023). *Mobile crowdsourcing: A review and analysis of current approaches and challenges*. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 9(2), 1-35. Retrieved from <https://doi.org/10.1145/3300899>

Pearce, J. M., Blair, C. M., Laciak, K. J., Andrews, R., Nosrat, A., & Zelenika-Zovko, I. (2017). *3-D printing of open source appropriate technologies for self-directed sustainable development*. *Journal of Sustainable Development*, 10(1), 17.

Perez, G. (2018). *Blockchain: re-immaginare i processi, re-immaginare il business*. Retrieved from: <https://www.zerounoweb.it/resource-center/blockchain-re-immaginare-i-processi-re-immaginare-il-business/>

Pressman, R. S. (2014). *Software engineering: a practitioner's approach (8th ed.)*. McGraw-Hill.

Sikora, M., Nowak, M., & Czabanowska, K. (2019). *An Approach to Software Engineering Based on Non-functional Requirements*. In *2019 IEEE 23rd International Conference on Computer Supported Cooperative Work in Design (CSCWD)* (pp. 484-489). IEEE.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). *Database system concepts*. McGraw-Hill Education.

Smith, L. (2013). *The advantages and disadvantages of open-source software for business*. *American Journal of Business Education*, 6(3), 317–324. Retrieved from <https://doi.org/10.19030/ajbe.v6i3.7718>

Soltani, S., Abedi, M., & Kazemi, S. M. (2021). *A novel method for multi-criteria decision making in comparative analysis of health information systems*. *Health and Technology*, 11(1), 127-141.

Sommerville, I. (2016). *Software engineering (10th ed.)*. Pearson.

Straus, D. M. (2019). *Clustering Techniques in Climate Analysis*.

Tantar, A. A., Berekmeri, A., & Miclea, L. (2018). *Non-functional Requirements in Software Engineering: A Systematic Mapping Study*. *Procedia Computer Science*, 126, 1814-1823.

Taticchi, P., Tonelli, F., Cagnazzo, L., & Lettieri, E. (2015). *Benchmarking sustainability practices in the automotive industry: A review*. *Journal of Cleaner Production*, 108, 1063-1075. doi: 10.1016/j.jclepro.2015.06.123

Vanderheiden, G. (2019, accessed 2023). *Web accessibility: What does it mean and why is it important?*. *Synthesis Lectures on Assistive, Rehabilitative, and Health-Preserving Technologies*, 11(1), 1-91. Retrieved from <https://doi.org/10.2200/S00983ED1V01Y201904ARH016>

Wang, X., Chen, Y., Huang, J., Chen, D., & Chen, J. (2020, accessed 2023). *GeoAI: Spatial computing using AI and deep learning*. *ISPRS International Journal of Geo-Information*, 9(1), 7. Retrieved from <https://doi.org/10.3390/ijgi9010007>

World Meteorological Organization (WMO). (1971). *Sixth World Meteorological Congress: Abridged Report with Resolutions (WMO-No. 292)* (1971 edition). WMO.

World Meteorological Organization (WMO). (1995). *Twelfth World Meteorological Congress: Abridged Final Report with Resolutions (WMO-No. 827)* (1995 edition). WMO.

World Meteorological Organization (WMO). (2007). *Fifteenth World Meteorological Congress: Abridged Final Report with Resolutions (WMO-No. 1026)* (2007 edition). WMO.

World Meteorological Organization (WMO). (2014a). *Climate Data Management System Specifications (WMO-No. 1131)* (2014 edition). WMO.

World Meteorological Organization (WMO). (2014b). *Guide to the WMO Integrated Global Observing System (WIGOS)*. (2014 edition). WMO.

World Meteorological Organization (WMO). (2015). *Guide to the WMO Information System (WIS)*. (2015 edition). WMO.

World Meteorological Organization (WMO). (2019a). *World Meteorological Congress: Abridged Final Report of the Eighteenth session (WMO-No. 1236)* (2019 edition). WMO.

World Meteorological Organization (WMO). (2019b). *Technical Regulations, Volume I: General Meteorological Standards and Recommended Practices (WMO-No. 49)* (2019 edition, updated in 2021). WMO.

World Meteorological Organization (WMO). (2019c). *Vision for Global Observing Systems in 2040 (WMO-1185)* (2019 edition). WMO.

World Meteorological Organization (WMO). (2019d). *WIGOS Metadata Standard (WMO-No. 1192)* (2019 edition). WMO.

World Meteorological Organization (WMO). (2021a). *Commission for Observation, Infrastructure and Information Systems: Abridged Final Report of the First Session (WMO-No. 1251)* (2021 edition). WMO.

World Meteorological Organization (WMO). (2021b). *Manual on the WMO Integrated Global Observing System (WMO-No. 1160): Annex VIII to the WMO Technical Regulations* (2021 edition). WMO.

World Meteorological Organization (WMO). (2022, accessed 2023). *WMO Information System (WIS) 2.0 in a Box (INFCOM-2/INF. 6.3.1(1))*. Retrieved from <https://filecloud.wmo.int/share/s/iaA3ggA0QW-xGsKILa08sg>

Zbinden, R., & Ionescu, R. (Eds.). (2018). *Interoperability and standardization in meteorology, oceanography and climate*. Cham, Switzerland: Springer

Zhao, Z., Liu, Y., Wang, H., & Jiang, Z. (2020). *Data-intensive application for science gateway: a case study on weather research and forecasting*. *Cluster Computing*, 23(1), 847-858.

Cuadro A1. Nombres y definiciones de los elementos (OMM, 2019d)

Cada elemento se clasifica como obligatorio (M), condicional (C) u opcional (O). Un asterisco (\*) indica que el elemento es necesario para el examen continuo de las necesidades de la OMM. Una almohadilla(#) indica que es aceptable registrar un elemento obligatorio con un valor nilReason (que indica que los metadatos son desconocidos, no son aplicables o no están disponibles), ya sea en cualquier circunstancia o de acuerdo con las especificaciones declaradas ad hoc para ese elemento (OMM, 2019).

Categoría	ID	Nombre	Descripción	M, C, O
1. Variable observada	1-01	Variable observada - medición	Variable que se pretende medir, observar o derivar, incluido el contexto biogeofísico	M*
	1-02	Unidades de medida	Cantidad escalar real, definida y adoptada por convención, con la que se puede comparar cualquier otra cantidad del mismo tipo para expresar la relación de las dos cantidades como un número (JCGM, 2012, referencia n°. 1.9)	C*
	1-03	Ampliación temporal	Periodo de tiempo cubierto por una serie de observaciones, incluida la fecha/hora de medición (historial de mediciones).	M*
	1-04	Extensión espacial	Volumen típico georreferenciado espacialmente cubierto por las observaciones.	M*
	1-05	Representatividad	Extensión espacial de la región en torno a la observación de la que es representativa	O
2. Objetivo de la observación	2-01	Ámbito(s) de aplicación	Contexto(s) principal(es) o aplicación(es) prevista(s) para la observación, o aquel(s) con los requisitos más estrictos	O*
	2-02	Afiliación al programa/red	Programa(s) y red(es) mundial(es), regional(es) o nacional(es) a los que está asociada la emisora/plataforma	M
3. Estación/ andén	3-01	Región de origen de los datos	Región de la OMM	C*
	3-02	Territorio de origen de los datos	Nombre del país o territorio donde se encuentra el lugar de observación	C*
	3-03	Nombre de la estación/plataforma	Nombre oficial de la emisora/plataforma	M
	3-04	Tipo de estación/plataforma	Categorización del tipo de instalación de observación en la que se lleva a cabo una observación	M*
	3-06	Identificador único de estación/plataforma	El identificador de estación WIGOS, globalmente único para cada instalación de observación (estación/plataforma)	M*
	3-07	Localización geoespacial	Posición en el espacio que define la ubicación de la estación/plataforma de observación en el momento de la observación	M*

Categoría	ID	Nombre	Descripción	M, C, O
	3-08	Método de comunicación de datos	Método de comunicación de datos entre la estación/plataforma y una instalación central	O
	3-09	Estado operativo de la estación	Estado operativo declarado de la estación	M
	3-10	Grupo de estaciones/plataformas	Indica que una estación/plataforma forma parte de una agrupación de estaciones/plataformas, identificadas por un nombre y una descripción.	O
4. Medio ambiente	4-01	Cobertura superficial	La cobertura (bio)física observada en la superficie terrestre en las proximidades de la observación	O
	4-02	Esquema de clasificación de la cubierta de superficie	Nombre y referencia o enlace al documento que describe el sistema de clasificación de la cubierta de superficie	C#
	4-03	Topografía o batimetría	Forma o configuración de un accidente geográfico, representada en un mapa mediante curvas de nivel.	O
	4-04	Eventos en el centro de observación	Descripción de acciones humanas o acontecimientos naturales en la instalación o cerca de ella que puedan influir en la observación	O
	4-05	Información del sitio	Información no formalizada sobre el lugar y el entorno en el que se realiza una observación y que puede influir en ella.	O
	4-06	Rugosidad de la superficie	Clasificación del terreno en función de la longitud de la rugosidad aerodinámica	O
	4-07	Zona climática	La clasificación climática de Köppen de la región en la que se encuentra la instalación de observación. El esquema de clasificación climática de Köppen-Geiger divide los climas en cinco grandes grupos (A, B, C, D, E), cada uno con diferentes tipos y subtipos	O
5. Instrumentación y métodos de observación	5-01	Fuente de la observación	Origen del conjunto de datos descrito por los metadatos	M
	5-02	Método de medición/observación	El método de medición/observación utilizado	M#
	5-03	Especificaciones de instrumentación	Capacidades intrínsecas del método de medición/observación para detectar el elemento designado, incluidos el alcance, la estabilidad, la precisión, etc.	O*
	5-04	Estado de funcionamiento del instrumento	El estado de un instrumento con respecto a su funcionamiento	O
	5-05	Distancia vertical del sensor	Distancia vertical del sensor desde un nivel de referencia (especificado), como el suelo, la cubierta de una plataforma marina donde se encuentra el sensor o la superficie del mar.	C*

Categoría	ID	Nombre	Descripción	M, C, O
	5-06	Configuración de la instrumentación	Descripción de cualquier apantallamiento o configuración/ajuste de la instrumentación o equipos auxiliares necesarios para realizar la observación o para reducir el impacto de influencias extrañas en la observación.	C#
	5-07	Programa de control de instrumentos	Descripción del programa de calibración o verificación de los instrumentos	O
	5-08	Resultado de la comprobación del instrumento	El resultado de la comprobación de un instrumento, incluida la fecha, la hora, la posición, el tipo de norma y el período de validez.	C#
	5-09	Modelo y número de serie del instrumento	Datos del fabricante, número de modelo, número de serie y versión del firmware, si procede.	O
	5-10	Mantenimiento rutinario de los instrumentos	Descripción del mantenimiento que se realiza regularmente en un instrumento	O
	5-11	Mantenimiento	Identificador de la organización o persona que ha realizado la actividad de mantenimiento	O
	5-12	Localización geoespacial	Localización geoespacial del instrumento/sensor	C#
	5-13	Actividades de mantenimiento	Descripción del mantenimiento realizado en el instrumento	O
	5-14	Estado de observación	Estado de observación oficial	O
	5-15	Visualización de instrumentos	Grado en que un instrumento se ve afectado por influencias externas y refleja el valor de la variable observada.	C#
6. Muestreo	6-01	Procedimientos de muestreo	Procedimientos necesarios para obtener una muestra	O
	6-02	Tratamiento de muestras	Tratamiento químico o físico de la muestra antes del análisis	O
	6-03	Estrategia de muestreo	La estrategia utilizada para generar la variable observada	O*
	6-04	Periodo de muestreo	Periodo de tiempo durante el cual se realiza una medición	O
	6-05	Resolución de muestreo espacial	La resolución espacial se refiere al tamaño del objeto observable más pequeño. La resolución intrínseca de un sistema de formación de imágenes viene determinada principalmente por el campo de visión instantáneo del sensor, que es una medida del área del terreno vista por un único elemento detector en un instante de tiempo determinado	O
	6-06	Intervalo de tiempo de muestreo	Periodo de tiempo entre el inicio de dos periodos de muestreo consecutivos	O
	6-07	Hora base diurna	Hora a la que se refieren las estadísticas diurnas	C#
	6-08	Programa de observación	Programa de observación	M#

Categoría	ID	Nombre	Descripción	M, C, O
7.7. Tratamiento de datos e informes	7-01	Métodos y algoritmos de tratamiento de datos	Una descripción del tratamiento utilizado para generar la observación y una lista de los algoritmos utilizados para obtener el valor resultante.	O
	7-02	Centro de tratamiento/análisis	Centro donde se procesa la observación	O
	7-03	Período de referencia	Período de tiempo en el que se informa de la variable observada	M*
	7-04	Intervalo de referencia espacial	Rango espacial en el que se informa de la variable observada	C*
	7-05	Software/procesador y versión	Nombre y versión del software o procesador utilizado para obtener el valor del elemento	O
	7-06	Nivel de datos	Nivel de procesamiento de datos	O
	7-07	Formato de los datos	Descripción del formato en el que se proporciona la variable observada	O
	7-08	Versión del formato de datos	Versión del formato de datos en el que se proporciona la variable observada	O
	7-09	Periodo de agregación	Periodo de tiempo durante el cual se agregan las muestras/observaciones individuales	O
	7-10	Hora de referencia	Base temporal a la que se refieren la fecha y la hora	O
	7-11	Datos de referencia	Datos de referencia utilizados para convertir la cantidad observada en cantidad notificada	C
	7-12	Resolución numérica	Medida del detalle con que se expresa una cantidad numérica	O
	7-13	Puntualidad (de los informes)	El tiempo que suele transcurrir entre la finalización de la observación o la recogida de datos y el momento en que se comunican los datos.	O
	7-14	Programa de Intercambio Internacional	Permite distinguir si las observaciones se realizan o no a escala internacional.	M
8.8. Calidad de los datos	8-01	Incertidumbre de medición	Parámetro no negativo, asociado al resultado de una medición, que caracteriza la dispersión de los valores razonablemente atribuibles a la observación/medición.	O*
	8-02	Procedimiento utilizado para estimar la incertidumbre	Una referencia o enlace a un documento que describa los procedimientos y/o algoritmos utilizados para obtener la declaración de incertidumbre.	C#
	8-03	Bandera de calidad	Una lista ordenada de calificadores que indican el resultado de un proceso de control de calidad aplicado a la observación	O
	8-04	Sistema de señalización de calidad	Referencia al sistema utilizado para calificar la calidad de la observación	C



Categoría	ID	Nombre	Descripción	M, C, O
	8-05	Trazabilidad	Declaración que define la trazabilidad a un patrón, incluida la secuencia de patrones de medición y calibraciones utilizados para relacionar un resultado de medición con una referencia (JCGM, 2012, número de referencia 2.42).	C*
9. Propiedad y política de datos	9-01	Organización del control	Nombre de la organización propietaria de la observación	M
	9-02	Política de datos/restricciones de uso	Detalles sobre el uso y las limitaciones de datos impuestas por la organización de control	M*
10. Contacto	10-01	Contacto (punto focal designado)	Contacto principal (punto focal designado) para el recurso	M

# 14 APÉNDICE B - OPCIONES DE DISEÑO RELATIVAS A LA UI PERSONALIZADA

## B.1 TIPOGRAFÍA

La familia tipográfica seleccionada para nuestro proyecto es Open Sans, creada por el diseñador Steve Matteson (**Figura B1**). Esta fuente presenta formas abiertas y un aspecto neutro pero agradable, y está optimizada para impresión, web e interfaces móviles, con excelentes características de legibilidad. También incluye conjuntos de caracteres para ISO Latin 1, Latin CE, griego y cirílico, así como una versión de fuente variable (Google Font, para ser precisos).

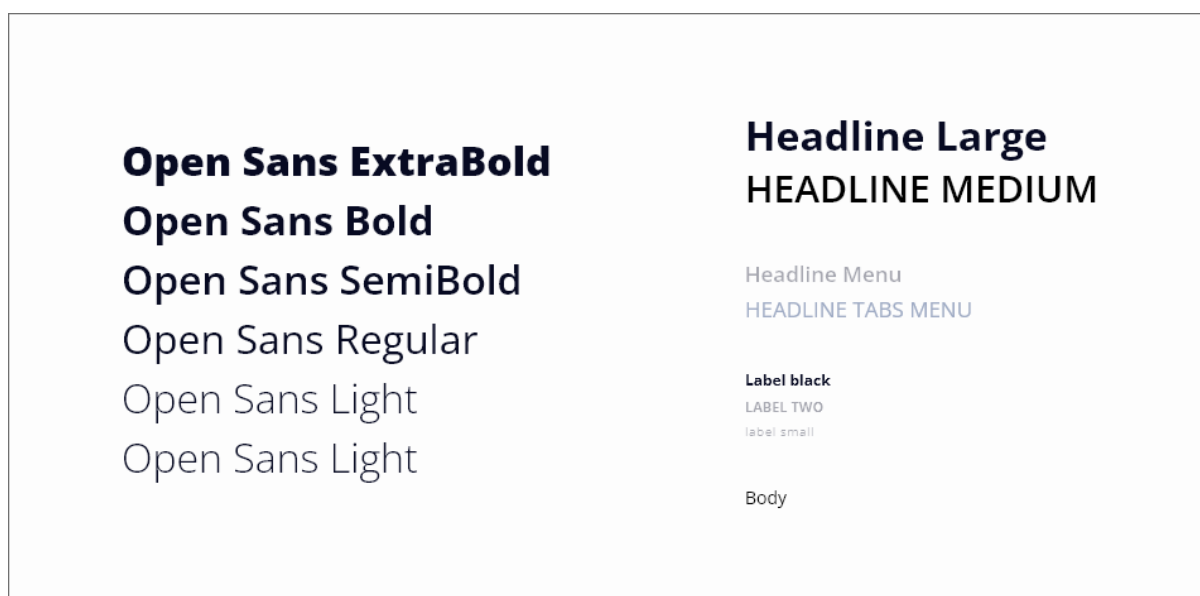


Figura B1

Un aspecto ventajoso es que la fuente Open Sans se distribuye bajo la licencia Open Font Licence, lo que significa que puede utilizarse en productos y proyectos impresos o digitales para usos comerciales o de otro tipo, y está disponible para su descarga gratuita en la plataforma Google Font.

## B.2 COLOR DEL TEXTO

El color del texto se ha seleccionado cuidadosamente para cumplir las normas mínimas de contraste entre colores. Los enlaces textuales, a menos que se resalten mediante botones, se identifican subrayando el texto.

## B.3 PALETAS DE COLORES

La selección de colores se hizo cuidadosamente para respetar la identidad del promotor del proyecto y ajustarse a la función institucional del producto final.

Se eligieron colores primarios, secundarios y neutros, teniendo los primeros una función predominante y siendo fácilmente visibles. Los colores secundarios se utilizan para las opciones de segundo nivel y los elementos secundarios, mientras que los colores neutros se utilizan para los elementos básicos y el fondo de la interfaz.

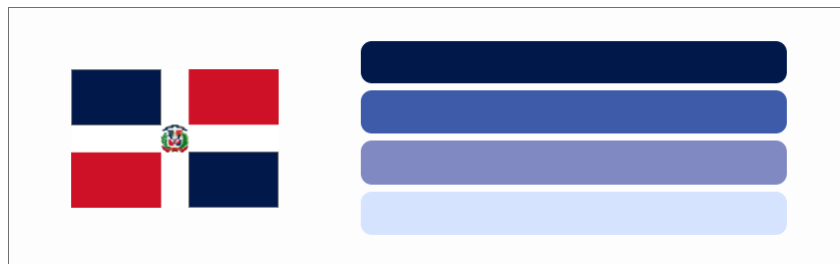


Figura B2



Figura B3

El color principal se tomó del azul de la bandera de la República Dominicana(**figura B2**), pero se adaptó para que fuera compatible con el uso en pantalla. Posteriormente, se crearon diversas variantes, creando una paleta que servirá de pauta para los modos claro y oscuro(**Figuras B3 y B4**).



Figura B4

## B.4 DISPOSICIÓN Y ESPACIADO

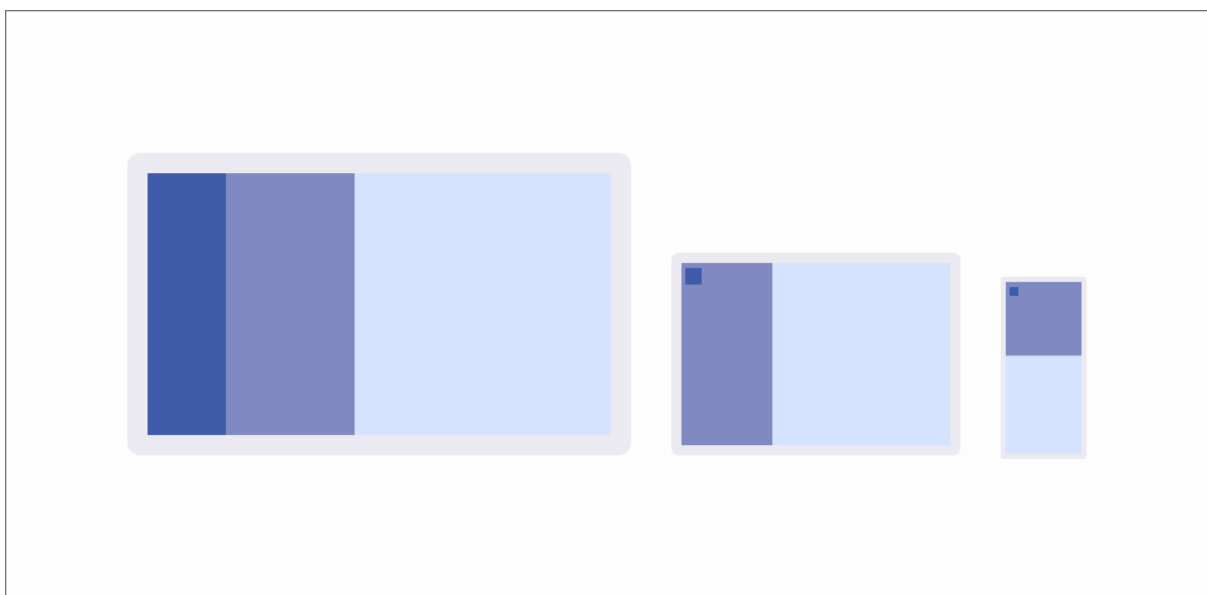


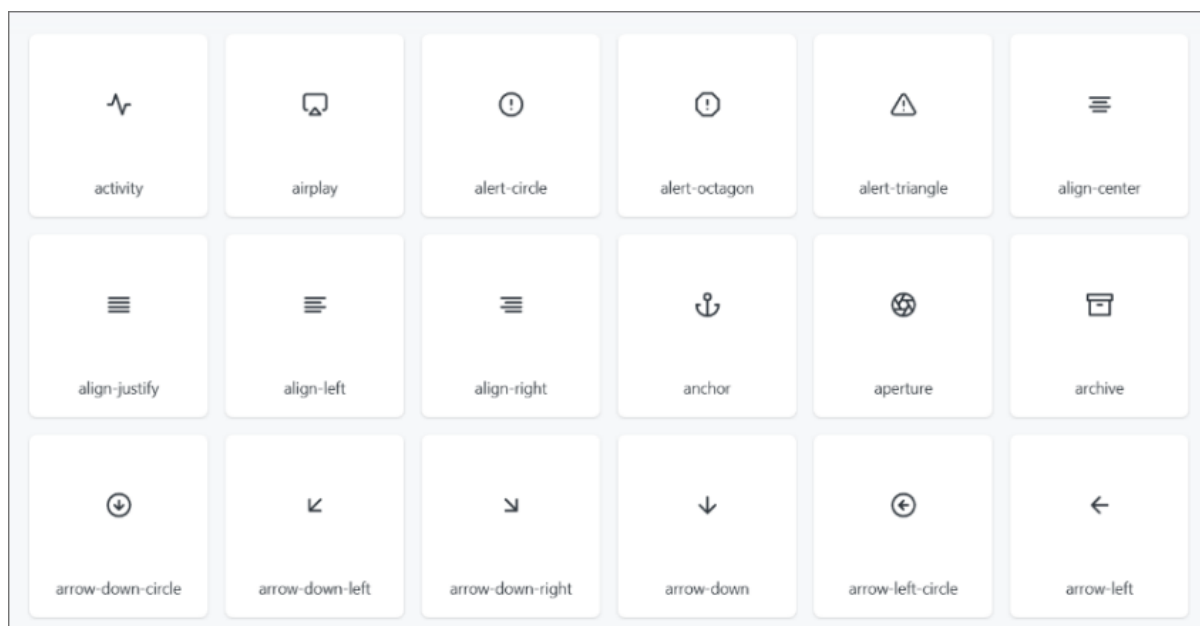
Figura B5

Los diseños de las páginas se elaboraron cuidadosamente para que encajaran adecuadamente en distintos tipos de pantallas y dispositivos (véase **la Figura B5**). El espaciado se ha ajustado sistemáticamente en múltiplos de 8 píxeles, salvo en algunas situaciones específicas indicadas en el documento Sistema de diseño.

## B.5 ICONOS

Para garantizar una coherencia visual óptima, es importante utilizar iconos de un conjunto específico. En este proyecto, hemos optado por utilizar iconos de Feather Icons(**Figura B6**), un conjunto de código abierto con iconos sencillos y eficaces.

Cualquier otro icono personalizado se diseñará siguiendo el mismo estilo elegido. En general, un icono debe ser autoexplicativo y no requerir explicaciones escritas. Sin embargo, si los comentarios o las pruebas demuestran que los usuarios no entienden el icono, se añadirán consejos o etiquetas, o se modificará o sustituirá el propio icono.



**Figura B6**

## B.6 EFECTO VIDRIO



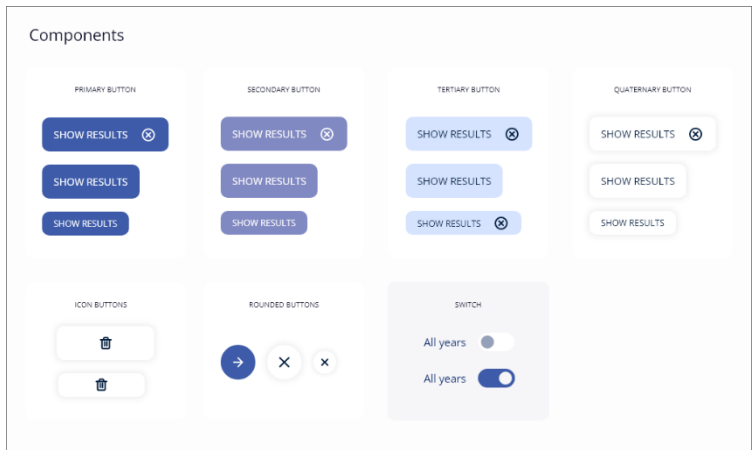
Figura B7

Para introducir una innovación en el contexto institucional que promueva una interfaz de usuario moderna sin comprometer la coherencia de un contexto similar, se decidió incorporar algunos elementos de cristalmorfismo (véase **la Figura B7**). Estos elementos se utilizarán de forma cuidadosa en las versiones posteriores a la consecución del hito, con el fin de destacar determinadas funcionalidades.

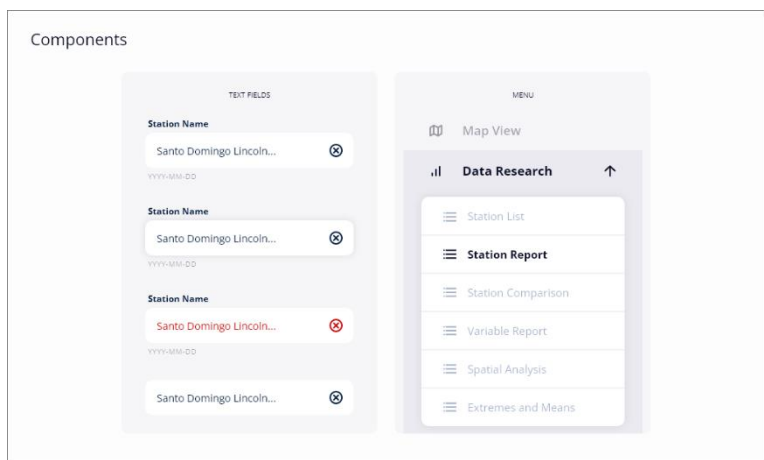
## B.7 COMPONENTES

Los componentes son los elementos visuales que se diseñan utilizando las herramientas descritas anteriormente y forman la parte visual de la interfaz de usuario (**Figura B8 a Figura B11**). Una vez diseñados, estos componentes se incorporan y organizan en un kit de diseño completo que se pone a disposición de los interesados.

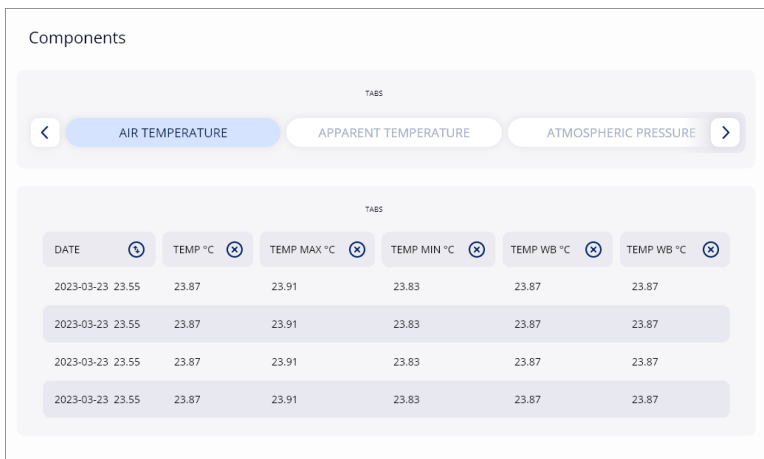
Los componentes incluyen botones, iconos, menús, barras de navegación, campos de texto, pestañas, cuadros de diálogo, interruptores, selectores de fecha y hora, etc.



**Figura B8**



**Figura B9**



**Figura B10**

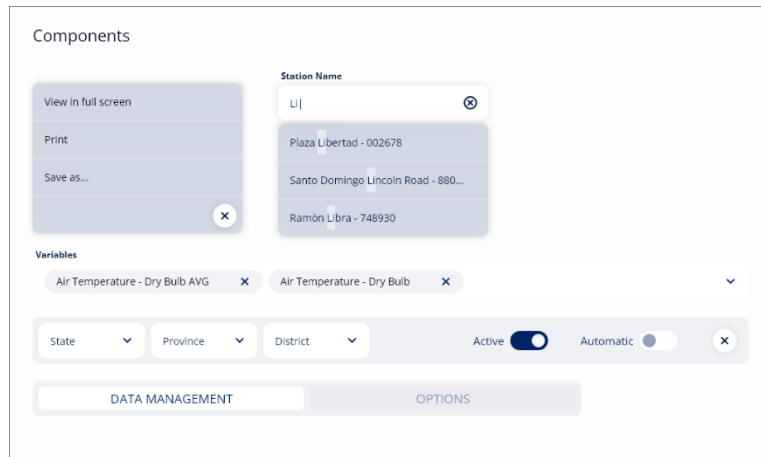


Figura B11