
Ficha Resumen

Trabajo de Grado Maestría

Título: Arquitectura para dispositivos IoT del instituto OMICAS

1. Área de trabajo: Agricultura 4.0
2. Tipo de proyecto: Aplicado
3. Estudiante: Carlos David Tello Ruiz
4. Estudiante: Germán Andrés Caycedo Mutis
5. Correo electrónico: germanc97@javerianacali.edu.co, cdtello@javerianacali.edu.co
6. Director: Diego Quintero
7. Correo electrónico del director: diegoquintero424@hotmail.com
8. Co-Director: Luis Eduardo Tobón
9. Grupo o empresa que lo avala: OMICAS
10. Palabras clave: IoT, Arquitectura, Datos, Agricultura 4.0, Internet, Nube, Interoperabilidad, Integración.
11. Fecha de inicio (Desarrollo del proyecto): 01/01/2023
12. ODS que aplica el proyecto (Agenda 2030): Objetivo 2: Poner fin al hambre
13. Resumen: Este proyecto tiene como objetivo proponer una arquitectura para la recolección, homologación y administración de los datos producidos por los dispositivos IoT en el marco del instituto OMICAS, con el fin de aportar a la producción de la agricultura en Colombia y ayudar a suplir la mayoría de las necesidades con cultivos mas eficientes. El protocolo de comunicación inalámbrica que será empleado es LoRa debido a la decisión del instituto OMICAS. Para llevar a cabo este proyecto se hará un levantamiento de los requisitos de la arquitectura; sera diseñado un modelo de los datos para estandarizar la información recolectada; se definirá un patrón de integración e interoperabilidad; se diseñará y desarrollará una prueba de concepto de la arquitectura de datos; y, por ultimo se evaluará la fiabilidad de la arquitectura propuesta que cumpla con las necesidades del instituto OMICAS.
Palabras clave: Agricultura. Arquitectura. IoT. LoRa. OMICAS.



MONOGRAFÍA DE TRABAJO DE GRADO

Arquitectura para dispositivos IoT del instituto
ÓMICAS

Germán Andrés Caycedo Mutis
Carlos David Tello Ruiz

Director

MSc. Diego Armando Quintero Quiñones

Codirector

Dr. Luis Eduardo Tobón Llano

19 de abril de 2024

Agradecimientos

Queremos agradecer a nuestras familias las cuales han sido muy importantes en el desarrollo de este proyecto. Queremos agradecer a los miembros del Instituto OMICAS del proyecto PhenoAgro los cuales nos ayudaron en todo momento del proyecto y sin ellos no hubiera sido posible. finalmente nos gustaría agradecer a nuestro director Diego el cual en todo momento nos estuvo apoyando, animando y solucionando todas las dudas que tuviéramos, el fue un miembro indispensable del proyecto y por eso estamos muy agradecidos con el.

Glosario

- **On-Premise:** Se refiere a la ubicación de recursos o sistemas informáticos que están físicamente alojados en las instalaciones de una organización, en lugar de estar en la nube o en servidores externos.
- **Batch:** Proceso que implica la recopilación y el procesamiento de datos en grupos o lotes, en lugar de forma continua e interactiva.
- **Stream:** Flujo continuo de datos o información que se procesa y transmite en tiempo real.
- **IoT (Internet of Things):** Internet de las cosas; se refiere a la red de dispositivos y objetos físicos que están conectados a la Internet y pueden intercambiar datos y realizar acciones automatizadas.
- **LoRa (Long Range):** Protocolo de comunicación inalámbrica de largo alcance utilizado en IoT para la transmisión de datos a distancias mayores con eficiencia energética.
- **Gateway:** Dispositivo que actúa como intermediario entre dispositivos IoT y la infraestructura de red, permitiendo la transmisión de datos entre ellos.
- **API (Application Programming Interface):** Interfaz de programación de aplicaciones; un conjunto de reglas y protocolos que permite a diferentes componentes de software comunicarse entre sí.
- **Diagramas C4:** Conjunto de diagramas de arquitectura que se utilizan para comunicar y documentar la arquitectura de software de manera efectiva.
- **Metadatos:** Datos que proporcionan información sobre otros datos, como la descripción, el origen y la estructura de los datos.
- **Dragino:** Marca de productos y soluciones relacionadas con IoT y comunicaciones inalámbricas.
- **Enrutamiento:** Proceso de encontrar la mejor ruta para transmitir datos de una fuente a un destino a través de una red.
- **InfluxDB:** Sistema de gestión de bases de datos de series temporales diseñado para almacenar y consultar datos de tiempo y métricas.
- **Decodificar:** Proceso de transformar datos codificados o cifrados en un formato legible o utilizable.
- **Buffer:** Área de almacenamiento temporal utilizada para retener datos antes de su procesamiento o transferencia.

- **Bucket S3:** En el contexto de Amazon S3, se refiere a un contenedor de almacenamiento que almacena objetos (archivos) en la nube.
- **Base de Datos:** Sistema para el almacenamiento y recuperación de datos estructurados de manera eficiente.
- **Dashboard:** Panel de control visual que muestra datos e información clave de manera clara y accesible.
- **CSV (Comma-Separated Values):** Formato de archivo que utiliza comas u otros delimitadores para separar y estructurar datos en forma de tablas o listas. Es comúnmente utilizado para almacenar y transferir datos en una forma legible por máquina.
- **Catalogación:** Proceso de organización y descripción de elementos, como documentos, objetos o datos, para facilitar su identificación, búsqueda y recuperación. La catalogación implica la asignación de metadatos y etiquetas para categorizar y clasificar elementos de manera eficiente.
- **BPMN:** Es un estándar de modelado gráfico utilizado en la gestión de procesos de negocios. Proporciona una notación gráfica fácil de entender para representar los procesos de negocio en forma de diagramas, lo que permite a los analistas y usuarios de negocios comprender, diseñar, documentar y mejorar los procesos de manera eficiente. Los elementos en un diagrama BPMN representan diversas actividades, eventos, flujos de secuencia, decisiones y relaciones entre los procesos, lo que facilita la comunicación y la colaboración entre los miembros del equipo y las partes interesadas en la gestión de procesos empresariales.

Resumen

Este proyecto tiene como objetivo proponer, implementar y llevar a cabo una arquitectura, siguiendo las mejores prácticas y considerando factores críticos como el tiempo, el costo y el volumen de datos, para la recolección, homologación y administración de los datos generados por los dispositivos IoT en el marco del instituto ÓMICAS¹, con el fin de aportar a la producción de la agricultura en Colombia y ayudar a suplir la mayoría de las necesidades con cultivos más eficientes. El protocolo de comunicación inalámbrica que será empleado es LoRa debido a la decisión del instituto OMICAS. Para llevar a cabo este proyecto se hará un levantamiento de los requisitos de la arquitectura; será diseñado un modelo de los datos para estandarizar la información recolectada; se definirá un patrón de integración e interoperabilidad; se diseñará y desarrollará una prueba de concepto de la arquitectura de datos; y, por último se evaluará la fiabilidad de la arquitectura propuesta que cumpla con las necesidades del instituto ÓMICAS.

Palabras Clave: Agricultura - Arquitectura - IoT - LoRa - ÓMICAS.

¹Se refiere a la Optimización Multiescala In-silico de Cultivos Agrícolas Sostenibles (ÓMICAS), el cual forma parte del Ecosistema Científico del gobierno colombiano, establecida entre 2017 y 2018 para consolidar y validar una estrategia e infraestructura multiescala, multiinstitucional y multidisciplinaria para avanzar en los descubrimientos en la ciencia de las plantas y el desarrollo de nuevas soluciones tecnológicas para mejorar la productividad y la sostenibilidad agrícola (Jaramillo, et al, 2022).

Abstract

This project aims to propose, implement, and execute an architecture, following best practices and considering critical factors such as time, cost, and data volume, for the collection, standardization, and management of data generated by IoT devices within the framework of the ÓMICAS Institute². The wireless communication protocol to be used is LoRa due to its advantages. To carry out this project, a survey of architecture requirements will be conducted, a data model will be designed to standardize the collected information, an integration and interoperability pattern will be defined, a data architecture proof of concept will be designed and developed, and finally, the reliability of the proposed architecture that meets the needs of the ÓMICAS Institute will be evaluated.

Keywords: Agriculture - Architecture - IoT - LoRa - ÓMICAS.

²Referring to the Multiscale In-silico Optimization of Sustainable Agricultural Crops (ÓMICAS), which is part of the Colombian government's Scientific Ecosystem, established between 2017 and 2018 to consolidate and validate a multiscale, multi-institutional, and multidisciplinary strategy and infrastructure for advancing discoveries in plant science and the development of new technological solutions to enhance agricultural productivity and sustainability (Jaramillo, et al, 2022).

Índice general

1. Introducción	1
1.1. Planteamiento del problema	1
1.2. Formulación del problema	3
1.3. Objetivo General	4
1.4. Objetivos específicos	4
1.5. Resultados esperados	4
1.6. Criterios de Aceptación	4
1.7. Alcance	6
1.8. Justificación	7
2. Marco de Referencia	9
2.1. Bases Teóricas	9
2.1.1. Internet de las cosas (IoT)	9
2.1.2. La Interoperabilidad	10
2.1.3. Integración de los datos	10
2.1.4. Protocolos de comunicación	11
2.1.5. LoRa	11
2.2. Estado del Arte	13
2.3. Elección del Caso de Estudio	16
2.4. Análisis del proceso	16
2.4.1. Análisis cualitativo	16
2.5. Órbita de Rediseño	17
2.6. Método Heurístico de Rediseño de Proceso	18
2.6.1. Etapas:	19
2.7. Metodología Event Storming	20
3. Resultados y Discusión	21
3.1. Inicio	21
3.1.1. Registro de problemas	21
3.1.2. Clientes internos o externos	23
3.1.3. Vista operacional del proceso de negocio	23
3.1.4. Requerimientos de la arquitectura	23
3.1.5. Elección Plataforma de Despliegue	25
3.1.6. Restricciones de arquitectura	25
3.1.7. Decisiones de arquitectura	26
3.2. Diseño	27
3.3. Implementación	33

3.3.1. Capa de Dispositivos Físicos y Gateway	33
3.3.2. Capa de Ingesta de Datos	40
3.3.3. Capa de Almacenamiento y Procesamiento de Datos	44
3.3.4. Capa de Consumo de Datos	52
3.3.5. Capa de Metadatos y Catalogación	59
3.3.6. Validación	61
3.4. Conclusiones	63
3.5. Recomendaciones y lecciones aprendidas.	65
3.6. Trabajos futuros.	66
Bibliografía	69

Índice de figuras

2.1. Dominio de la Agricultura 4.0 y sus aplicaciones de ejemplo. (Araújo et al., 2021) . . .	14
2.2. Arquitectura y modelo en bloques de integración de datos (Kaur and Owusu, 2016).	15
2.3. Modelo de Integración de Datos (Kaur and Owusu, 2016).	15
2.4. Órbita de Rediseño. Tomada de Dustdar et al. (2008)	19
3.1. Diagrama de datos en BPMN.	27
3.2. Diagrama de contexto C4.	28
3.3. Diagrama de contenedor C4.	29
3.4. Diagrama de componentes C4.	30
3.5. Diagrama de despliegue.	31
3.6. Pantalla creación del gateway en consola AWS.	33
3.7. Pantalla gateway creado en consola AWS.	34
3.8. Pantalla perfil de dispositivo AWS.	34
3.9. Pantalla perfil de servicio AWS.	35
3.10. Pantalla perfil de servicio AWS.	36
3.11. Pantalla perfil de servicio AWS.	37
3.12. Pantalla especificaciones nuevo dispositivo Lora AWS.	38
3.13. Pantalla selección perfiles y destino nuevo dispositivo Lora AWS.	39
3.14. Pantalla Dispositivos LoRaWAN consola AWS.	40
3.15. Pantalla función Lambda de decodificar consola AWS.	41
3.16. Pantalla función Lambda de almacenar en InfluxDB consola AWS.	42
3.17. Pantalla función Lambda de almacenar en AWS TimeStream consola AWS.	42
3.18. Pantalla secuencia de entrega Kinesis Data Firehouse consola AWS.	43
3.19. Pantalla configuración destino Kinesis Data Firehouse consola AWS.	43
3.20. Pantalla configuración búfer Kinesis Data Firehouse consola AWS.	43
3.21. Pantalla configuración particionamiento Kinesis Data Firehouse consola AWS.	44
3.22. Pantalla monitoreo Kinesis Data Firehouse consola AWS.	44
3.23. Pantalla Base de Datos Timestream consola AWS.	45
3.24. Pantalla estructura de la tabla de Timestream consola AWS.	46
3.25. Pantalla resultados consulta tabla de Timestream consola AWS.	47
3.26. Pantalla objetos del Bucket S3 Raw Zone consola AWS.	48
3.27. Pantalla objetos del Bucket S3 Curated Zone consola AWS.	49
3.28. Pantalla Amazon EventBridge Programación consola AWS.	50
3.29. Pantalla Amazon EventBridge Destino consola AWS.	51
3.30. Pantalla función Lambda de procesar la información y almacenarla consola AWS.	52
3.31. Pantalla menú Dashboards AWS Managed Grafana.	53
3.32. Pantalla Dashboard MPC estación Cambridge AWS Managed Grafana.	54

3.33. Pantalla Dashboard Atmosféricas estación CIAT AWS Managed Grafana.	54
3.34. Pantalla Dashboard Suelo estación CIAT AWS Managed Grafana.	55
3.35. Pantalla Dashboard Suelo tendencias estación CIAT AWS Managed Grafana.	55
3.36. Pantalla editor sentencias SQL AWS Athena consola AWS.	56
3.37. Pantalla resultados sentencias SQL AWS Athena consola AWS.	57
3.38. Pantalla visualización de resultados en formato CSV.	58
3.39. Pantalla estructura tabla AWS Athena consola AWS.	58
3.40. Pantalla configuración AWS Crawler consola AWS.	59
3.41. Pantalla información de ejecución AWS Crawler consola AWS.	60
3.42. Pantalla configuración tabla AWS Data Catalog consola AWS.	61
3.43. Pantalla esquema tabla AWS Data Catalog consola AWS.	61
3.44. Arquitectura de referencia Smart Farm on AWS Services (2022)	67

Índice de cuadros

2.1. Tabla comparativa protocolos adaptada de Araújo et al. (2021).	11
3.2. Registro de problemas del proyecto PhenoAgro	22
3.3. Elección Plataforma de Despliegue	25
3.4. Restricciones de arquitectura.	25
3.5. Decisiones de arquitectura	26

Introducción

1.1. Planteamiento del problema

Según Araújo et al. (2021) en diferentes países se ha venido implementando tecnologías de la agricultura 4.0, las cuales con la ayuda de dispositivos IoT se pueden desarrollar diversas soluciones de agricultura, mejorando la eficiencia de los cultivos. Sin embargo, en Colombia el sector de la agricultura es conservador, el cual presenta practicas rezagadas y ha conllevado a un atraso en el sector en comparación a los otros países que presentan una mayor eficiencia en sus cultivos.

El instituto ÓMICAS es un programa de investigación que busca desarrollar e implementar diferentes estrategias que aporten a la mejora de las variedades agrícolas (ÓMICAS, 2021), todo lo anterior con el fin de aportar a uno de los ODS (objetivos de desarrollo sostenible), más específicamente el número 2 "Hambre Cero", que hace referencia a realizar un cambio en el sistema agroalimentario para dar respuesta y suplir el hambre de toda la población mundial, el cual se estima que para el 2050 sea de aproximadamente de 9700 millones de personas (Nations, 2022). Para dar cumplimiento a este ODS que está enfocado en poner fin al hambre, se considera muy importante aumentar la productividad agrícola y tener una producción agrícola sostenible (Nations, 2015).

Como se mencionó anteriormente, para lograr el objetivo principal, el instituto ÓMICAS realiza estudios para mejorar genéticamente las variedades agrícolas de arroz y caña de azúcar, de tal manera que tengan una mejor respuesta al estrés físico y biológico al que se puedan enfrentar para que su producción aumente y sea sostenible, es decir, que se logre obtener una mayor cantidad de cultivo minimizando el uso de recursos naturales, la generación de materiales tóxicos, residuos y contaminantes (Argentina, 2022)

Para llevar a cabo todos los estudios e investigaciones necesarias sobre los cultivos de arroz y caña de azúcar, el instituto ÓMICAS está implementando dispositivos IoT, los cuales permiten tener una gran cantidad de datos en tiempo real sobre diferentes variables agrónomas que requieren analizar. Estos dispositivos se programan dependiendo de los datos que se necesiten evaluar, como por ejemplo el color o el tamaño de las plantas, las cuales son útiles para obtener un fenotipo del cultivo en cuestión. Si bien, las variables a analizar son numerosas y todas son diferentes, lo que significa que son datos heterogéneos que llevan a tener una gran cantidad de tipos de información.

Ahora bien, al tener diferentes tipos de información y dispositivos que censan las variables que se encuentran en los cultivos, se obtiene una gran variedad de tipos de datos y diferentes canales

por los cuales la información de los dispositivos es transmitida, lo que hace que la información recolectada no pueda ser analizada fácilmente, para ello es necesario plantear una solución que permita interactuar con todos los datos, que se puedan analizar entre si, correlacionarse y arrojar información procesada que sea de gran utilidad para quienes hagan uso de ella.

Actualmente, el instituto ÓMICAS se ha centrado en desarrollar y recolectar información de los dispositivos IoT, más no en integrar, analizar y aprovechar estos datos obtenidos, debido a la falta de personal con conocimiento técnico en el área de integración de información, como resultado, ÓMICAS está perdiendo la oportunidad de identificar patrones de comportamiento en las plantas, lo cual indica que no se está utilizando correctamente la información que debería servir como insumo para los estudios e investigaciones que deben realizar para cumplir con su objetivo principal, lo que a su vez impide generar grandes insights para las investigaciones que adelanta el instituto.

1.2. Formulación del problema

¿Como lograr que los diferentes dispositivos IoT logren transmitir los datos recolectados y se integren entre si en un mismo ambiente?. ¿Qué método se puede aplicar para que la recolección de datos se pueda manipular sin importar de que dispositivo provenga?. ¿Qué áreas de investigación del proyecto están involucradas o interesadas en los datos recolectados?. ¿Como lograr inter-operabilidad de los datos recolectados entre las diferentes áreas de investigación del proyecto?.

1.3. Objetivo General

Proponer, diseñar e implementar una arquitectura completa para la gestión de datos de dispositivos IoT en el proyecto PhenoAgro del Instituto ÓMICAS, que incluye la visualización en tiempo real, análisis batch, conexión con plataformas heredadas, almacenamiento a largo plazo y gestión de dispositivos IoT, con el propósito de optimizar la disponibilidad y utilidad de los datos para respaldar las investigaciones del proyecto PhenoAgro y los objetivos del instituto.

1.4. Objetivos específicos

- Contar con un sistema de carga de información en tiempo real que permita la adquisición continua y eficiente de datos desde diversas fuentes IoT.
- Diseñar e implementar un dashboard de monitoreo en tiempo real que brinde a los usuarios la capacidad de visualizar y analizar datos en tiempo real de manera intuitiva y efectiva.
- Establecer un sistema de almacenamiento de información en batch que posibilite el análisis histórico de tendencias y patrones a partir de datos recopilados a lo largo del tiempo.
- Configurar una integración efectiva entre AWS y la infraestructura on-premise para el almacenamiento y la gestión de métricas capturadas, permitiendo una transferencia de datos fluida y segura entre ambas plataformas.

1.5. Resultados esperados

Tener definida e implementada la arquitectura para la recolección, homologación y administración de los datos producidos por dispositivos IoT que cumpla con las necesidades del proyecto ÓMICAS identificadas previamente, adicionalmente tener realizada una prueba de concepto de la arquitectura propuesta y los resultados obtenidos de la prueba después de hablar y obtener retroalimentación de las personas directamente afectadas por el proyecto. La arquitectura debe contar con una capa de stream para visualizar los datos en tiempo real por medio de dashboards, por otra parte se debe contar con una capa batch la cual va a contener los componentes de estructuración y almacenamiento de los datos obtenidos.

1.6. Criterios de Aceptación

Debido a que el proyecto es una solución para el proyecto de PhenoAgro del instituto ÓMICAS se definieron en conjunto con los ingenieros del instituto unos criterios de aceptación los cuales permiten definir si las soluciones aplicadas desarrolladas y entregadas cumplen con lo que ÓMICAS busca y necesita. La forma de obtención de estos criterios ha sido por medio de entrevistas al personal del proyecto, cabe aclarar que el personal del proyecto PhenoAgro son ingenieros enfocados al hardware por lo cual muchas definiciones de arquitectura no las dejaron claras y se han tomado

decisiones basadas en las personas que hacen parte del desarrollo de este proyecto (Arquitectura para dispositivos IoT del instituto ÓMICAS).

- AC1: Se debe contar con una plataforma visual que permita visualizar los datos enviados por los dispositivos IoT para monitorear el estado de cada estación.
- AC2: La plataforma debe poderse conectar por medio de LoraWAN a los dispositivos edge y por medio de internet a los Lora Gateway.
- AC3: Se debe conectar por lo menos a dos dispositivos edge (estaciones) y a un Lora Gateway ya que no se espera tener desarrollados mas dispositivos al momento que se ejecuta este proyecto (Arquitectura para dispositivos IoT del instituto ÓMICAS)
- AC4: Se debe de contar con un almacenamiento de los datos a largo plazo para su posterior análisis.

1.7. Alcance

Teniendo en cuenta que el diseño de la arquitectura va a contar con 3 componentes, a saber: recolección, integración y administración de los datos, se espera llegar al diseño de los componentes de recolección, integración, estructuración de los datos obtenidos de los dispositivos IoT, además, tener mecanismos de visualización de los datos como dashboards en tiempo real y una herramienta para consultar los datos almacenados a largo tiempo. Posteriormente realizar una prueba de concepto donde se implemente la arquitectura diseñada con todos sus componentes incluyendo la conexión con plataformas heredadas, visualizadores en tiempo real y la administración de dispositivos IoT.

Como restricción dada por el el proyecto PhenoAgro del instituto OMICAS la recolección e integración de los datos recolectados por los dispositivos IoT se debe dar entre dos dispositivos debido que al momento de desarrollarse este proyecto son los dos únicos componentes de hardware que van a estar disponibles. Por otra parte, hay que tener en cuenta que puede que los dispositivos no estén listos a tiempo para hacer las pruebas e integraciones lo que conlleva a simular por medio de software la interacción entre lo implementado en la arquitectura y lo que serian los dispositivos IoT que el instituto se encuentra desarrollando actualmente.

Una consideración importante que se debe tener encuentra dentro del alcance es que el desarrollo de los dispositivos IoT se encuentra en simultaneo al desarrollo e implementación de la arquitectura por lo tanto cualquier retraso e inconveniente puede impactar y retrasar en las pruebas e integración final cambiando el alcance a lo largo del proyecto.

1.8. Justificación

La cosecha manual es una operación laboriosa, lenta y, a veces, propensa a errores. Además, el costo asociado a la mano de obra humana está aumentando y se está reduciendo la disponibilidad de mano de obra calificada que acepte tareas repetitivas (Araújo, 2021).

Por esta razón, los agricultores buscan una mejor productividad y mejorar la calidad de las cosechas para las personas. Así es como las industrias relacionadas con la agricultura se están moviendo hacia automatizar procesos en la granja, que pueden incluir robots de cosecha agrícola.

En este sentido, la agricultura ha sido transformada con el uso de máquinas agrícolas inteligentes, control de plagas inteligentes, drones para monitorizar los cultivos y el ganado, uso de estaciones que captan variables agroclimáticas que han hecho posible la recolección y aprovechamiento de grandes cantidades de datos críticos con mínimos costos, lo que ha logrado un proceso agrícola más orientado a la información y potencialmente más productivo y eficiente (García, et al. 2019). Todo esto ha sido gracias a la implementación del IoT en la agricultura.

El uso del IoT tiene muchas ventajas para la agricultura desde su impacto económico en la reducción de los costos de la producción y en el tiempo de cosecha, hasta la eficiencia y productividad de los productos. En otras palabras, la implementación del IoT en la agricultura ayuda a que los cultivos sean más eficientes, a minimizar costos, a ganar tiempo, entre otros factores.

Teniendo en cuenta esto, y pensando en el crecimiento poblacional que se ha venido presentando y que se presentará con el transcurrir de los años se pensó en este proyecto que busca diseñar una arquitectura en la recolección, homologación y administración de los datos producidos por los dispositivos IoT.

Es de nuestro conocimiento que se han desarrollado diferentes proyectos en la región al respecto pero son sólo aproximaciones a las soluciones planteadas en la alianza ÓMICAS y no suplen todas las necesidades.

De esta manera, la propuesta de arquitectura que se tiene pensada para este proyecto integrará diferentes fuentes de datos como lo son sensores e información estática que pueda ser ingresada por personas al sistema. Este diseño de arquitectura aborda una carencia evidente en los proyectos previos, como se revelará en los antecedentes que se presentarán más adelante. Su enfoque en aspectos como la visualización en tiempo real, la implementación en la nube, la generación de valor mediante análisis de datos, la gestión de costos y la administración de la plataforma, marca una diferencia fundamental en la eficacia y capacidad de estos proyectos.

Por esta razón, se pensó en que esta propuesta de proyecto tiene viabilidad y podría ser aprovechada en la producción de la agricultura en Colombia. Sin embargo, se debe aclarar que en este

proyecto se planea llegar hasta obtener una prueba de concepto o Mínimo Producto Viable. En este sentido, se han propuesto algunos objetivos que serán descritos en el siguiente capítulo.

Marco de Referencia

En este capítulo se describen las bases teóricas que serán usadas para el desarrollo de este proyecto y después se presenta el estado del arte donde se muestran algunas de las investigaciones más recientes que se han realizado hasta el momento acerca de los diseños de arquitectura para la recolección, homologación y administración de los datos producidos por dispositivos IoT.

2.1. Bases Teóricas

2.1.1. Internet de las cosas (IoT)

Cuando se habla del Internet de las Cosas (IoT¹) se hace referencia a la relación que existe entre dispositivos físicos y digitales. El término de IoT “fue empleado por primera vez en 1999 por el pionero británico Kevin Ashton para describir un sistema en el cual los objetos del mundo físico se podían conectar a Internet por medio de sensores” (Rose, et al p. 13).

En la actualidad no existe una definición única para el IoT y diferentes grupos utilizan distintas definiciones para describir el IoT con sus atributos más importantes. Esto se debe a que es un concepto que ha tomado fuerza con el pasar de los años porque hay diferentes dispositivos IoT que realizan múltiples tareas. Este proyecto se enfocará en proponer una arquitectura que ayude a la recolección de los datos producidos por dispositivos IoT del proyecto OMICAS debido a que las aplicaciones agrícolas de IoT incluyen agricultura de precisión, monitoreo de ganado, invernadero inteligente, gestión de pesca y seguimiento del clima.

En este sentido, se tomará la definición dada por Araújo, et al. (2021), quienes mencionan que el IoT “es un término usado para referirse a la conectividad entre las “cosas” físicas y digitales mediante estándares y protocolos de comunicación que permitan la interoperabilidad” (p. 10).

Ahora bien, en el IoT se habla de “cosas” para referirse a los diferentes dispositivos físicos como sensores de temperatura, sensores de humedad, sensores de luz, cámaras, entre otros. También se habla de “cosas” digitales que se refieren a diferentes softwares como bases de datos, programas de procesamiento de datos, aplicaciones de transmisión de imágenes, entre otras aplicaciones. Un ejemplo muy conocido del IoT es cuando se conecta un sensor de temperatura con una base de

¹Se usarán las iniciales IoT para referirnos al Internet de las Cosas (Internet of Things).

datos mediante un estándar y un protocolo de comunicación, pues a pesar de que el sensor y la base de datos son dos cosas diferentes estas dos van a estar interoperando (Tschofenig, p. 2015).

2.1.2. La Interoperabilidad

Gracias a los avances tecnológicos que se han presentado, se ha logrado un desarrollo significativo de los diferentes sensores, ya que no solo realizan las respectivas mediciones, sino también transmiten la información sin necesidad de una conexión física, más específicamente lo hace en una pequeña parte del espectro electromagnético, permitiendo una transmisión remota, lo que la hace completamente útil para el contexto de la agricultura (Araújo et al., 2021).

En este sentido, al hablar de interoperabilidad según la IEEE en su glosario (IEE, 1991) se refiere a “la capacidad de dos o más sistemas o componentes de intercambiar información y utilizar la información intercambiada”. Por lo tanto, cada vez que un dispositivo físico y un sistema digital se comunican o intercambian información se habla de interoperabilidad entre estos dos componentes.

Teniendo en cuenta esta definición, cabe mencionar que trabajos como el de Zeng et al. (2019) muestran los enfoques para lograr la interoperabilidad como se demuestra en los estándares y las mejores prácticas, proyectos y productos en el amplio dominio de la organización del conocimiento. Es por esto que es tenida en cuenta en el desenvolvimiento de este proyecto.

Adicionalmente, la interoperabilidad entre los dispositivos y sistemas de la IoT ocurre en diferentes grados y en diferentes capas dentro de la pila de protocolos de comunicación entre los dispositivos (Rose, et al p. 49). Una de sus ventajas es facilitar la capacidad de escoger los dispositivos con las mejores características y al mejor precio e integrarlos de manera que funcionen juntos.

Un informe publicado en 2015 sostiene que “en promedio, la interoperabilidad es necesaria para crear un 40 por ciento del potencial valor que puede generar la Internet de las Cosas en diversos entornos” (Manyika, et al. p. 2).

2.1.3. Integración de los datos

Ahora bien, en este proyecto se debe realizar una integración de los datos para permitir su interoperabilidad desde su captura hasta su visualización. En este sentido, la integración de datos se puede describir como la correlación de los datos de diferentes fuentes con el objetivo de ayudar en la comprensión de un fenómeno. Esta integración permite generar conjuntos de datos que se encuentran correlacionados, cuyos datos son obtenidos de diferentes fuentes (López y Corrales, 2018).

Othman (2015) menciona un aspecto importante en la integración de los datos y es que dado que “los resultados agregados de los datos se pueden usar para tomar decisiones críticas, una estación base debe atestiguar la integridad de los resultados agregados antes de aceptarlos (p. 874).

2.1.4. Protocolos de comunicación

Los protocolos de comunicación son las tecnologías que permiten el intercambio de datos entre los diferentes componentes físicos y digitales. De esta manera, existen una amplia variedad de protocolos de comunicación dependiendo las especificaciones para las cuales se requiera. Los protocolos de comunicación inalámbrica que son comúnmente usados se pueden ver en la Tabla 1 con sus características básicas. Las columnas de izquierda a derecha corresponden a: el nombre de los protocolos de comunicación; los valores estándares de cada protocolo; la banda de frecuencia en Gigahercios y Megahercios; el rango de trasmisión en metros y kilometros; la velocidad de los datos en Megabits por segundo y Kilobit por segundo; el consumo de energía en vatios; y, el costo del protocolo, si es bajo, medio o alto. .

	Standard	Frequency Band	Transmission Range	Data Rate	Energy Consumption	Cost
Bluetooth	Bluetooth (Formerly IEEE 802.15.1)	2.4 GHz	10-100 m	1-3 Mb/s	0.1-1 W	Low
LoRaWAN	LoRaWAN	Various	2-15 km	0.3-50 kb/s	100 mW	Low
NFC	ISO/IEC 13157	13.56 MHz	0.1 m	424 kb/s	1-2 mW	Low
Mobile communication	2G-GSM, GPRS 3G-UMTS, CDMA2000 4G-LTE	865 MHz, 2.4 GHz	Entire mobile network area	2G: 50-100 kb/s 3G: 200 kb/s 4G: 0.1-1 Gb/s	1 W	Medium
RFID	Various Sigfox	13.56 MHz	100 m	423 kb/s	1 mW	Low
Sigfox	Sigfox	2400-2483.5 MHz	30-100 km	10-1000 b/s	122 mW	Low
Wi-Fi	IEEE 802.11 a/c/b/d/g/n	2.4, 3.6, 5, 60 GHz	100 m	6-780 Mb/s 6.75 Gb/s at 60 GHz	1 W	High
ZigBee	IEEE 802.15.4	2400-2483.5 MHz	100 m	250 kb/s	1 mW	Low

Cuadro 2.1: Tabla comparativa protocolos adaptada de Araújo et al. (2021).

De acuerdo con esta tabla, se debe señalar que los protocolos de comunicación inalámbrica que son más adecuados para el desarrollo de este proyecto debido a sus aplicaciones agrícolas basadas en IoT, son los de consumo de energía y costo menores porque tiene un buen alcance de trasmisión, es decir, Sigfox, ZigBee y LoRa (Araújo et al., 2021).

Debido a restricciones impuestas por parte del instituto OMICAS, el cual es el desarrollador del hardware y dueño del proyecto, el cual ha seleccionado previamente el protocolo de comunicación LoRa se adoptó el protocolo para la arquitectura propuesta.

2.1.5. LoRa

De acuerdo con la sesión anterior, cabe mencionar la importancia del protocolo de Largo Alcance (LoRa²) en este proyecto. LoRa es una tecnología de comunicación inalámbrica que combina el consumo de energía con un largo alcance efectivo.

Esta tecnología fue desarrollada por “Cycleo y adquirida por Semtech en 2012 que es muy prometedora para construir redes de acceso amplio con topología en estrella (a menudo denominadas redes de área amplia de baja potencia). Estos últimos brindan comunicación de largo alcance a miles de dispositivos a un costo mínimo y un gasto de energía limitado, lo que permite realizar redes IoT

²Se usará la abreviación LoRa para referirnos al Long Range.

urbanas a gran escala” (Boano, et al. p. 40).

Algunas de las ventajas de LoRa son su largo alcance y cobertura, la baja potencia, es un hardware de bajo coste y de alta capacidad. Por esto, es tomada en cuenta en el desarrollo de este proyecto para alcanzar los objetivos propuestos. Sin embargo, una de las ventajas clave de este protocolo de sensor es la complejidad reducida de los protocolos de red, debido a que el amplio rango de comunicación permite formar topologías donde los dispositivos finales de baja potencia se comunican directamente con uno más potente (Boano, et al).

Por otra parte, si pensamos en que la población mundial viene creciendo de manera acelerada, entonces cada vez son más los habitantes y sus necesidades. De esta manera, algunos estudios prevén que para el 2025 la población llegue a los 8 billones de personas (Food and Organization., 2017) haciendo que los recursos que necesiten también tengan mucha más demanda. Entre estos recursos necesarios para la supervivencia humana se encuentran los alimentos, elemento fundamental para el crecimiento poblacional y su existencia. Por lo tanto, para poder llevar este crecimiento acelerado de la población se estima que la producción alimenticia debe crecer entre un 60 % y 70 % (Food and Organization, 2016).

Sin embargo, hacer que la producción alimenticia crezca de esta manera no es fácil, pues existen factores como el cambio climático, la contaminación y el desgaste de suelos, que dificultan y plantean desafíos a la producción de alimentos (Rosegrant, et al. 2019).

La agricultura requiere un desarrollo proporcional al crecimiento poblacional y una toma de decisiones basada en la información obtenida a partir de los cultivos y de acuerdo con las necesidades de momento de cada población. Para esto, es preciso implementar y hacer uso de los recursos que brinda la tecnología. Por ejemplo, la tecnología trae consigo beneficios que facilitan a los agricultores tener conocimiento de la información en tiempo real de los cultivos. Gracias a esta información, los agricultores pueden tomar decisiones informadas que traen consecuencias positivas para la producción de los cultivos (Baierle, et al. 2022).

A fin de avanzar en la producción alimenticia y enfrentar los retos de la actualidad se han creado varias iniciativas, entre ellas la llamada “Agricultura 4.0” (Rose and Chilvers, 2018) que va de la mano a la “Industria 4.0”, dos revoluciones de cuarta generación que como gran avance tienen la implementación de modelos digitales e implementan avances tecnológicos como IoT, Computación en la Nube, Big Data, Inteligencia artificial (IA), entre otros (Rose and Chilvers, 2018).

Uno de los problemas que se presentan para la obtención de los datos de los cultivos es el difícil acceso a ellos, puesto que depende de la topografía que tenga el terreno. En este sentido, se debe tener en cuenta que algunas investigaciones muestran que el proceso entre la captura y la visualización de los datos requiere ciertos pasos.

Araújo, et al. 2021 y Baierle, et al. 2022 muestran que para facilitar esta recolección, gracias al IoT es posible utilizar dispositivos tecnológicos tales como robots, sensores, drones, los cuales tienen una mayor cobertura en menor tiempo, es decir, pueden llegar fácilmente (o instalarse) a lugares de difícil acceso, registrar una gran cantidad de datos en menos tiempo y transmitirlos en tiempo real a los Gateways que luego envían información a los servidores correspondientes para su procesamiento y almacenamiento.

Finalmente, la información es enviada a plataformas de visualización de datos, en donde las personas a cargo podrán visualizar y tomar las decisiones a partir de estos datos previamente procesados. En otras palabras, “el progreso de las tierras de cultivo se puede rastrear mediante imágenes satelitales, que luego se analizan mediante un motor de IA, y después, las aplicaciones móviles visualizan simultáneamente el resultado para que los agricultores puedan entender el estado de forma remota” (Liu, et al. 2021, p. 4329).

Como se mencionó anteriormente, la información recolectada por los diferentes dispositivos IoT, es procesada y almacenada gracias a la Big Data, ya que además de soportar grandes cantidades de información, la procesa y almacena en tiempo real, convirtiéndola en información con valor agregado una vez es implementada en conjunto con la IA, facilitando la toma de decisiones que debe tomar el agricultor (Araújo, et al. 2021).

2.2. Estado del Arte

En esa sesión se describirán algunos de los antecedentes de este proyecto que proponen algún tipo de arquitectura para la recolección y administración de los datos producidos por dispositivos IoT.

Un aspecto que se debe señalar es que en el 2022 el instituto ÓMICAS no cuenta con alguna implementación completa de Agricultura 4.0, pues sólo tiene sensores inteligentes que captan la información ambiental como luz, temperatura, humedad, características del suelo y la envían a un Gateway³ bajo protocolo Lora donde es posteriormente enviada a un servidor por medio de internet. Esta es solo una parte de IoT y de Agricultura 4.0 y faltan varios componentes para que se pueda tener un proyecto completo.

Por otra parte, en la actualidad existen varios proyectos donde se han implementado muchas más tecnologías de la Agricultura 4.0 e IoT. Así, en la Figura 1 tomada de Araújo et al. (2021) se pueden observar los diferentes dominios donde se puede aplicar la Agricultura 4.0 y sus respectivos ejemplos de aplicación.

Ahora la discusión se enfocará en el dominio de monitoreo y predicción el cual es el que se relaciona con el proyecto del instituto ÓMICAS. A continuación se evidencian tres proyectos de

³Uno de los elementos principales para la creación de sistemas conectados son los Gateways IoT, que nos permiten conectar nuestro dispositivo a internet, así como también periféricos que se puedan conectar al él (Tobar, 2016).

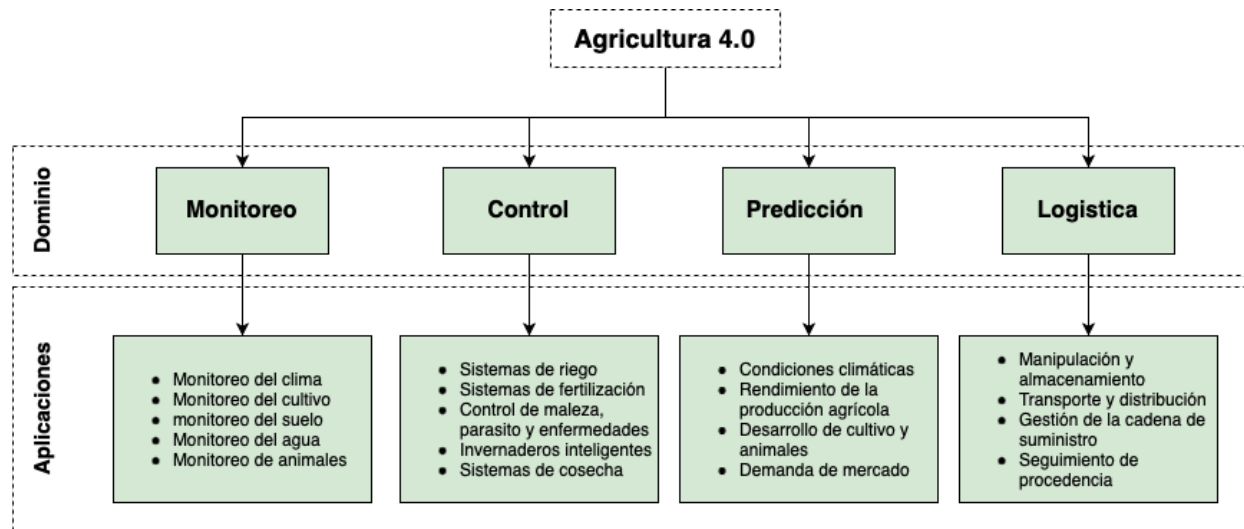


Figura 2.1: Dominio de la Agricultura 4.0 y sus aplicaciones de ejemplo. (Araújo et al., 2021)

aplicación de la Agricultura 4.0 donde se ha mejorado el monitoreo de los cultivos con implementación de sensores que recolectan datos para después integrarlos. Como soluciones que han planteado desde arquitecturas para el manejo de datos hasta modelos de integración de los datos.

En el primer proyecto [Kaur and Owusu \(2016\)](#) con el objetivo de optimizar y monitorear las actividades basadas en el sistema de crecimiento de cultivos, se propuso una arquitectura fundamentada en el manejo de datos, la cual recolecta y almacena información obtenida de diferentes sensores inalámbricos. A continuación en la figura 2.2a se puede evidenciar la arquitectura propuesta y en la figura 2.3a se muestra la forma de integración de los datos.

Por otro lado, con el mismo objetivo de la propuesta de [Kaur and Owusu \(2016\)](#), [López and Corrales \(2018\)](#) realizaron una integración de datos, en el cual los sensores que recolectan la información monitorean factores ambientales como lo es la temperatura y humedad del suelo de un cultivo de hongos, que determinan la resistencia del mismo.

Continuando, [Griffin et al. \(2009\)](#) emplearon imágenes satelitales de mediana y alta resolución con el fin de determinar el rendimiento de un cultivo, en este caso, se realiza la integración de información de las imágenes satelitales con variables climáticas como lo es temperatura y precipitación para obtener estadísticas de producción.

Los anteriores proyectos son aproximaciones muy cercanas como soluciones planteadas a la problemática que se tiene en el proyecto de ÓMICAS, sin embargo no alcanzan a cubrir todas las necesidades que se tienen debido a que no son arquitecturas ni proyectos que integren diferentes fuentes de datos como lo son sensores u otros aspectos como la visualización en tiempo real, la

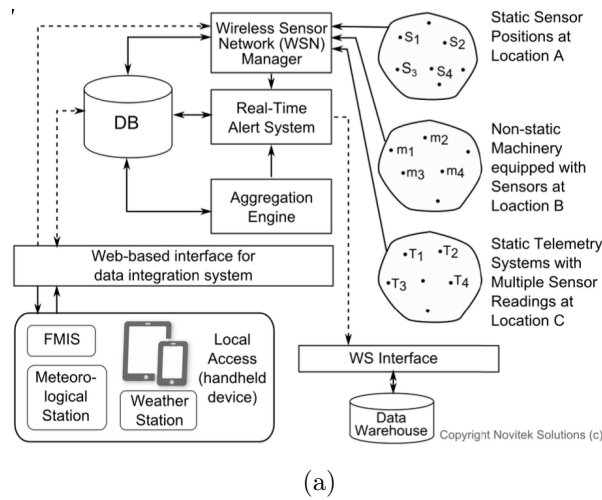


Figura 2.2: Arquitectura y modelo en bloques de integración de datos (Kaur and Owusu, 2016).

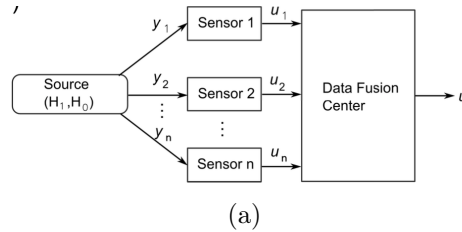


Figura 2.3: Modelo de Integración de Datos (Kaur and Owusu, 2016).

implementación en la nube, la generación de valor mediante análisis de datos, la obtención de KPIs, la gestión de costos y la administración de la plataforma. Por otra parte, las arquitecturas ya existentes no son pensadas en escalabilidad y alta disponibilidad que asegure que la arquitectura perdure en el tiempo ni pueda ser integrada a nuevas soluciones futuras.

Como metodología se realizó una investigación evolutiva, donde se analizan cambios a través del tiempo de manera iterativa. Además, se tiene una aproximación transversal, ya que se va a hacer una recolección de datos descriptivos, exploratorios, causales y se termina con una parte experimental donde se tuvo una prueba de concepto.

Se aplicó la metodología de desarrollo de software iterativa, donde se realizaron *sprints* cada dos semanas. Con base en esta metodología de desarrollo, se fue progresando en el proyecto de manera incremental, en el cual se retoman los resultados de los *sprints* anteriores para mejorar y seguir avanzando con el proyecto. A medida que se obtuvieron versiones del proyecto, se analizaron y se le refinaron o mejoraron las características hasta llegar a tener todos los requisitos completos.

Basado en los marcos de trabajo de las metodologías ágiles, en la implementación del proyecto fue necesario realizar adaptaciones a los momentos del proyecto, extendiendo por 2 ocasiones el sprint

que inicialmente tenía una duración de 2 semanas a 4 semanas, lo anterior debido a circunstancias como acceso a los dispositivos, retrasos por parte del Instituto, etc.

2.3. Elección del Caso de Estudio

Teniendo en cuenta los procesos que se desarrollan en el instituto OMICAS, se identifica el proyecto PhenoAgro, el cual es de gran impacto para la región debido a que se enfoca en el área de los cultivos agrícolas. En consecuencia el funcionamiento correcto y efectivo de este proyecto significa el beneficio de una gran parte de la población en la región. Además, después de una reunión con su Coordinador, Luis Eduardo Tobón, se evidencian algunas falencias y mejoras que se pueden realizar al proyecto, por lo tanto, se ve la oportunidad ideal de ser elegido como caso de estudio para este proyecto. También este caso cuenta con dispositivos OT, tecnologías IT y procesos de software lo cual es uno de los requisitos fundamentales que debe tener el caso de estudio.

2.4. Análisis del proceso

2.4.1. Análisis cualitativo

Analizar procesos de negocio es arte y ciencia [Dustdar et al. \(2008\)](#). El análisis cualitativo es el lado artístico del proceso de análisis.

En [Dustdar et al. \(2008\)](#) se trabajan un conjunto de técnicas para el análisis de proceso cualitativo. Primero se presentan dos técnicas enfocadas a identificar pasos innecesarios del proceso (análisis de valor agregado) y fuentes de desperdicios (análisis de desperdicios), de las cuales sólo se va a trabajar con la técnica de análisis de valor agregado. Luego, se presentan técnicas para identificar y documentar el problema en un proceso desde múltiples perspectivas, también se utilizan para analizar las causas fundamentales de los problemas, de las cuales se escoge la técnica del análisis de stakeholder para aplicar en el proceso de negocio. Las razones por las cuales se escogieron usar el método de valor agregado y la técnica de análisis de stakeholder fue debido a, primero al alcance del proyecto ya que la fase de diseño y prueba de concepto a la que se planea llegar no necesita de hacer otros tipos o métodos de análisis más profundos y el proyecto cuenta con un tiempo reducido. Segundo, el análisis de stakeholder también se ve reducido en una parte considerable, ya que como es planteado en [Dustdar et al. \(2008\)](#) abarca muchos actores y para el caso de estudio escogido estos actores no están presentes.

2.4.1.1. Análisis de Stakeholder

En el análisis de la operación de negocio según la investigación de [Dustdar \(2008\)](#), se debe tener en mente que siempre es posible mejorar. Esto se observa en numerosos casos donde, a pesar de optimizarse y cambiar continuamente, los procesos empresariales experimentan dificultades. Siempre habrá errores y pasos innecesarios que surgen en las actividades diarias. Por lo tanto, una tarea

esencial en este análisis es identificar y documentar los problemas que afectan el rendimiento de una operación.

El trabajo de un analista de operaciones incluye la identificación y documentación de los desafíos que influyen en la ejecución. Para ello, el analista recopilará información de diversas fuentes y llevará a cabo entrevistas con las partes involucradas, principalmente.

Los participantes experimentan el procedimiento desde una perspectiva interna. Es posible que noten defectos que surgen de las transferencias en el procedimiento, pero los clientes no siempre detectan estos defectos si se corrigen internamente. Generalmente, los participantes pueden proporcionar información sobre los residuos en la operación, que incluyen no solo defectos, sino también el desperdicio de transporte, movimiento y espera, que surge de las transferencias, así como el procesamiento excesivo.

Las partes externas pueden tener diversas inquietudes según su función en la operación. Los proveedores y subcontratistas se preocupan por mantener un flujo constante o creciente de trabajo en la operación, planificar sus actividades con anticipación y cumplir con los requisitos contractuales. En resumen, les preocupa la previsibilidad y pueden identificar oportunidades para mejorar sus interacciones entre sus propias operaciones y las operaciones en las que están integrados.

Es probable que el propietario de la operación y los directores operativos estén interesados en las métricas de rendimiento, como los tiempos de ciclo o tiempos de procesamiento prolongados, ya que estos están directamente relacionados con los costos laborales y la eficiencia. El propietario del proceso también puede centrarse en los defectos comunes y la sobreproducción. Además, el propietario del proceso y otros directores suelen estar preocupados por el cumplimiento de las políticas internas y regulaciones externas.

En el caso del proceso de negocio seleccionado, las partes involucradas, que se tendrán en cuenta para el análisis son: los coordinadores del proyecto que en este caso van a ser Simón Plata y Luis Eduardo Tobón .

2.5. Órbita de Rediseño

En la Figura 2.4 se puede distinguir un completo espectro de métodos de rediseño de procesos de negocio. Se visualizó este espectro como la órbita de rediseño como se muestra en la Figura. Como se explica en [Dustdar et al. \(2008\)](#) el eje vertical distingue los métodos transaccionales que se posicionan en el lado izquierdo de la Figura, de los métodos transformacionales en el lado derecho. Similarmente, el eje horizontal en la órbita de rediseño, muestra la diferencia entre los métodos creativos y los analíticos. El círculo interno de la órbita de rediseño contiene los métodos que se pueden caracterizar, como orientados hacia adentro, mientras que los métodos por fuera del círculo tienen una naturaleza hacia afuera.

Se distingue entre los métodos transaccionales y transformacionales, que el método transaccional apoya la identificación de problemas o cuellos de botella en un proceso y luego ayuda a resolverlo de forma incremental. Un método transaccional no desafía los cimientos existentes, por el contrario busca mejorar el proceso general gradualmente. Un método de transformación tiene como objetivo

lograr un avance, el cambio a gran escala. Este cuestiona los supuestos y principios fundamentales detrás de un proceso existente y su objetivo es romper radicalmente con ellos.

Los métodos de rediseño también difieren con respecto a su naturaleza, con el analítico y creativo como polos opuestos. El método de rediseño analítico se caracteriza por una base matemática y el uso de técnicas cuantitativas.

Un método de rediseño introspectivo tiene en cuenta como base la organización en la que se encuentra el proceso de negocio. En estos métodos los intereses y factores importantes de la organización son la parte central. Un método de rediseño enfocado en el exterior, tiene como principio los entes externos al proceso como en algunos casos el cliente o terceros.

Teniendo en cuenta las anteriores definiciones con las que se basó para escoger el método deseado y correcto para el rediseño del proceso de negocio, se analiza entre los métodos transaccionales y transformaciones, decidiéndose el uso de los métodos transaccionales, debido a que estos apoyan la identificación de problemas y ayuda a resolverlos, lo cual es lo que se ha venido trabajando en el proyecto. Por otra parte, buscando optimizar el proceso ya existente e irlo mejorando sin cambiar por completo el proceso, lo cual es lo que buscan hacer los métodos transaccionales. Continuando con el análisis y la decisión del método, se escoge un método que sea analítico, ya que se basa en cifras y cantidades, con lo que se cuenta en el proyecto y es lo que se busca mejorar, teniendo en cuenta los tiempos y cantidades de datos como factores importantes a mejorar en el proceso de negocio. Por último, se decide un método introspectivo, debido a que el proceso de negocio y los datos que se han recolectado de este son de la misma organización y de los actores internos del proceso como el dueño y gerente o monitor.

Contando con diferentes opciones dentro del grupo de métodos factibles para el rediseño, los cuales se pueden evidenciar en la órbita de rediseño en la Figura 2.4, para la decisión final se basa en el texto [Dustdar et al. \(2008\)](#), en el cual se explica muy bien y se toma como referencia de un método transaccional y analítico, se escoge el método Heurístico de rediseño de proceso.

2.6. Método Heurístico de Rediseño de Proceso

En el método heurístico se hace énfasis en la consideración sistemática de una extensa gama de principios de rediseño, lo que lo convierte en un enfoque analítico. Se enfoca en las heurísticas de rediseño, las cuales son una gran cantidad y es donde este método encuentra su gran fortaleza. Para la explicación del rediseño de procesos, se enfocará en el desafío técnico de generar un nuevo diseño de procesos. A continuación se explicarán la etapas que son importantes para el método y seguido, se explica en qué consisten las heurísticas de rediseño, las cuales son el factor más importante de este método.

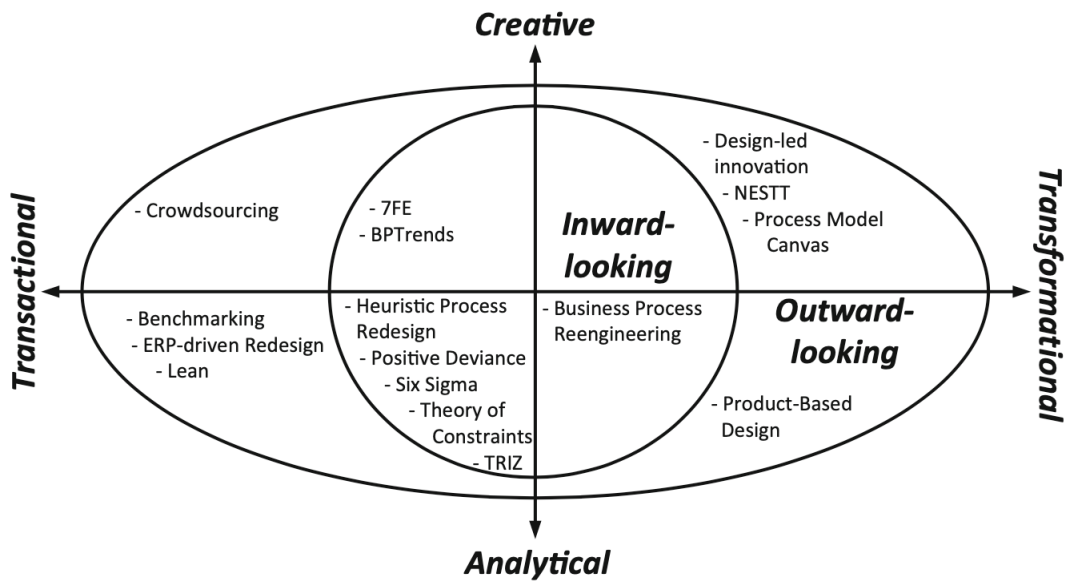


Figura 2.4: Órbita de Rediseño. Tomada de [Dustdar et al. \(2008\)](#)

2.6.1. Etapas:

1. **Inicio:** En la primera etapa, el proyecto de rediseño se configura. Hay muchas medidas organizacionales que se deben tener en cuenta, como por ejemplo, configurar el equipo del proyecto. Desde una mirada técnica los objetivos más importantes son:
 - a) Crear una comprensión de la situación existente
 - b) Configurar los objetivos para el proyecto de rediseño.
2. **Diseño:** Dados los resultados de la etapa inicial, la etapa de diseño hace uso de una lista fija de heurísticas de rediseño para determinar posibles acciones de mejora en el proceso existente. Para cada objetivo de desempeño, se necesita que el equipo del proyecto reflexione sobre las heurísticas relevantes que se puedan aplicar. Es deseable aplicar una heurística de rediseño si facilita la mejora del rendimiento deseada del proceso en consideración. Después de que se haya determinado qué heurísticas de rediseño serán útiles, cobra sentido ver si se pueden conformar grupos de estos. Para algunas heurísticas, puede tener sentido aplicarlas juntas, para otras no.
3. **Evaluación:** Esta es la etapa en la que se deben evaluar los diferentes escenarios de rediseño desarrollados en la etapa anterior. Esta evaluación se puede realizar en una forma cualitativa. En muchos entornos prácticos, se emplea una combinación de ambos cuando un grupo de expertos evalúa el atractivo de los diversos escenarios y donde los estudios de simulación se

usan para respaldar la elección de un escenario en particular que se desarrollará aún más, potencialmente hasta que sea implementado. En esta etapa puede ocurrir que ninguno de los resultados sea atractivo de seguir o se considere lo suficientemente importante como para establecer la mejora de desempeño deseable. Dependiendo del resultado exacto, una decisión puede ser ajustar los objetivos de desempeño, devolverse a la etapa de diseño o renunciar al proyecto de rediseño por completo.

2.7. Metodología Event Storming

Event Storming es un método interactivo y colaborativo que se utiliza para obtener una visión clara de cómo funciona un sistema de negocio. Se basa en la idea de que los eventos, o cosas que suceden en el negocio, son una excelente manera de entender el flujo de trabajo.

En un taller de Event Storming, todos los participantes, desde los expertos en el negocio hasta los desarrolladores de software, se reúnen para mapear estos eventos. Utilizan notas adhesivas de colores para representar diferentes aspectos del sistema, como eventos, comandos y actores.

Este proceso permite a todos los participantes tener una visión compartida del sistema y entender cómo interactúan sus partes. Es especialmente útil cuando se enfrentan a problemas complejos, ya que ayuda a desglosar la complejidad en partes manejables.

Para el desarrollo del proyecto PhenoAgro, se adoptó un enfoque basado en entrevistas con el personal del Instituto OMICAS. El objetivo principal de estas entrevistas era obtener una comprensión profunda del negocio, identificar las necesidades específicas que tenían en relación con el proyecto y descubrir los desafíos a los que se enfrentaban. En lugar de utilizar notas adhesivas, se optó por tomar notas detalladas durante las entrevistas para capturar toda la información proporcionada por el personal de OMICAS. Gracias a estas entrevistas, se pudo comprender y modelar con éxito el proyecto PhenoAgro y los problemas que se necesitaban resolver.

Resultados y Discusión

3.1. Inicio

En la parte inicial del diseño se crea una comprensión de la situación existente en el proyecto PhenoAgro del instituto OMICAS.

3.1.1. Registro de problemas

Debido a que el análisis de Stakeholder ayuda a identificar problemas desde diferentes perspectivas, se debe registrar y organizar esta información para poder identificar con claridad su impacto en el proceso. Los registros de problemas se van a documentar en forma de lista con una estructura previamente definida.

Para el proyecto PhenoAgro se identificaron los problemas indicados en el Cuadro 3.2, los cuales han sido ordenados de manera temporal según el orden de ejecución.

Cuadro 3.2: Registro de problemas del proyecto PhenoAgro

<p>Problema 1: Los datos se guardan de manera local</p> <p>Prioridad: 1</p> <p>Descripción: Una vez capturados los diferentes tipos de datos recolectados por los dispositivos IoT los datos de diferentes dispositivos se guardan de manera local</p> <p>Datos y suposiciones: El personal encargado de los dispositivos deben ir hasta el sitio del dispositivo y extraer la información de manera manual.</p> <p>Impacto cualitativo: La calidad de la información recolectada puede no ser la mejor al demorarse mucho en ser recolectada para su posterior uso.</p> <p>Impacto cuantitativo: No aplica</p>
<hr/> <p>Problema 2: Los datos se guardan de maneras diversas y en plataformas diversas</p> <p>Prioridad: 2</p> <p>Descripción: Debido a que los datos se recolectan de manera manual por diferentes personas, cada uno de los involucrados guardan los datos como ellos lo vean necesario y en la plataforma escogida por ellos mismos.</p> <p>Datos y suposiciones: Cada persona del proyecto que necesite un dato en particular debe buscar en diferentes partes y muchas veces debe consultar con el equipo si conoce donde se encuentra el dato.</p> <p>Impacto cualitativo: El personal debe de dedicar tiempo extra en la extracción y búsqueda del dato y este tiempo puede ser usado en su análisis.</p> <p>Impacto cuantitativo: 2 minutos x dato</p>
<hr/> <p>Problema 3: No hay visibilidad sobre los dispositivos IoT en uso.</p> <p>Prioridad: 3</p> <p>Descripción: Debido a que la manera de conexión y de recolección de los datos tomados por los dispositivos IoT (Se conectan de manera manual a un gateway que esta conectado a la red local y no tiene interfaz de visualización de los dispositivos IoT conectados, no hay forma de ver los dispositivos conectados y funcionando.</p> <p>Datos y suposiciones: Se necesita una persona en el campo que revise los dispositivos y verifique si se encuentran conectados.</p> <p>Impacto cualitativo: No aplica</p> <p>Impacto cuantitativo: Se elevan los costos de operación al tener una persona. 1 persona x (salario).</p>
<hr/> <p>Problema 3: No hay una conexión segura y estandarizada para consumir los datos</p> <p>Prioridad: 4</p> <p>Descripción: Los datos son recolectados desde los dispositivos y el gateway a un servidor que no tienen ningún protocolo de seguridad y no cuentan con un estándar para guardar y consumir los datos</p> <p>Datos y suposiciones: Se expone la información a cualquier problema de seguridad y además se vuelve complicado el proceso de consumir los datos generando gasto de tiempo extra.</p> <p>Impacto cualitativo: Se deja la posibilidad a filtración de datos y errores al momento de consumirlos.</p> <p>Impacto cuantitativo: Se elevan los costos de operación al tener una persona. 1 persona x (salario). + costos de una posible filtración de los datos</p>

3.1.2. Clientes internos o externos

El cliente para el proyecto es el instituto OMICAS, el cual es el que se ve beneficiado por el sistema y el cual necesita de la implementación de la integración IT/OT.

3.1.3. Vista operacional del proceso de negocio

Dentro del proceso existen 4 actividades:

- Recolectar los datos
- Enviar los datos de la estación al gateway
- Almacenar los datos
- Visualizar los datos recolectados

3.1.4. Requerimientos de la arquitectura

3.1.4.1. Requisitos Funcionales

- RF-01: El sistema debe ser capaz de conectar y comunicarse con dispositivos IoT a través del protocolo MQTT.
- RF-02: El sistema debe ser capaz de conectar y comunicarse con dispositivos IoT a través del protocolo LoraWan.
- RF-03: Los dispositivos IoT deben poder ser registrados y gestionados de forma remota desde una interfaz de administración.
- RF-04: El sistema debe recopilar datos de los dispositivos IoT de manera periódica y almacenarlos en una base de datos.
- RF-05: El sistema debe ser capaz de descryptar la información enviada por los dispositivos IoT para su procesamiento y almacenamiento.
- RF-06: El sistema debe proporcionar una herramienta de visualización en tiempo real que permita a los usuarios monitorear y analizar los datos provenientes de los dispositivos IoT de manera eficiente.
- RF-07: Los datos recopilados de los dispositivos IoT deben poder ser almacenados de manera segura durante períodos largos de tiempo, cumpliendo con las regulaciones de retención de datos aplicables.
- RF-08: Los usuarios deben poder exportar datos históricos almacenados en formatos estándar para su análisis externo o informes.

3.1.4.2. Requisitos No Funcionales - Requisitos de Integración

- RNF-01: El sistema debe ser altamente escalable para admitir un crecimiento futuro de dispositivos IoT de hasta 100 dispositivos conectados simultáneamente.

- RNF-02: El sistema debe garantizar la seguridad de la comunicación entre los dispositivos IoT y el servidor central mediante el uso de encriptación.

- RNF-03: El sistema debe garantizar la escalabilidad para admitir un número creciente de usuarios concurrentes que consumen información de los dispositivos IoT sin degradación significativa del rendimiento.

- RNF-04: La disponibilidad de los datos generados por los dispositivos IoT no debe superar un tiempo máximo de inactividad de 24 horas por semana. Los datos deben estar disponibles en todo momento, con un objetivo de tiempo de inactividad mínimo.

- RNF-05: El sistema debe ser capaz de integrarse de manera eficiente y bidireccional con las herramientas heredadas ya desarrolladas para el intercambio de datos recolectados. Debe seguir los protocolos y estándares de integración relevantes y asegurar la transferencia segura y confiable de datos entre el sistema y las herramientas heredadas.

3.1.4.3. Requisitos de Interfaz

- RI-01: La interfaz de usuario debe ser intuitiva y fácil de usar para los usuarios finales y los administradores.

- RI-02: La interfaz de la herramienta de visualización de los datos en tiempo real debe ser intuitiva y fácil de usar para usuarios de diferentes niveles de habilidad técnica. Debe proporcionar una visualización completa y clara de todos los datos requeridos por el usuario, permitiendo una navegación y manipulación eficientes de la información.

3.1.5. Elección Plataforma de Despliegue

Crterios de decisión	Justificación
Mayor cantidad de servicios IoT	AWS IoT es una plataforma de IoT líder en el mercado, que ofrece una amplia gama de servicios y herramientas para la gestión de dispositivos IoT, la recopilación y el análisis de datos, y la integración con otros servicios de AWS. Esto hace que AWS sea una opción ideal para proyectos de IoT.
Desarrolladores del proyecto tienen mayor conocimiento en AWS	El hecho de que los desarrolladores del proyecto tengan experiencia previa en AWS significa que la curva de aprendizaje será más corta y que el equipo podrá aprovechar al máximo las características y funcionalidades de AWS. Además, AWS ofrece una amplia gama de recursos de formación y documentación para ayudar a los desarrolladores a aprender y a mejorar sus habilidades en la plataforma.

Cuadro 3.3: Elección Plataforma de Despliegue

3.1.6. Restricciones de arquitectura

Debido a que el proyecto PhenoAgro ya se encontraba en la fase de desarrollo, se encontraron algunas restricciones impuestas por el equipo de OMICAS. Se realizaron varias entrevistas para entender un poco el funcionamiento de ellos y el stack tecnológico con el que ya contaban, por lo cual, se presentaron tres restricciones para la arquitectura en temas de infraestructura ya presente y protocolos de comunicación para los dispositivos IoT.

Restricción	Descripción
RA-01-BaseDatosInfluxDB	La arquitectura se debe conectar con la base de datos InfluxDB que se encuentra en el servidor del instituto y está expuesta con una ip publica.
RA-02-ProtocoloComunicacionSensores	El protocolo de comunicaciones de los dispositivos IoT debe ser LoraWAN y el proceso de autenticación OTAA v1.0.x.
RA-03-CodificaciónMensajes	Se debe de decodificar los mensajes bajo el protocolo de codificación definido por el instituto.

Cuadro 3.4: Restricciones de arquitectura.

3.1.7. Decisiones de arquitectura

Decisión	Descripción
DA-01-PlataformaDespliegue	La plataforma de despliegue de la arquitectura va a ser AWS debido a que el equipo de trabajo tiene mayor familiaridad con este servicio y es el proveedor en la nube con mayor caducidad de servicios IoT
DA-02-VisualizacionDatos	El dashboard con la información en tiempo real de los datos va a ser Grafana por el bajo coste y la familiaridad del equipo con esta plataforma.
DA-03-ModeloArquitectura	La arquitectura va a contar con dos partes, la parte Stream que es para los componentes de procesamiento en tiempo real y el componente Batch para el procesamiento de los datos transformados y almacenados a larga duración.
DA-04-BucketsPorCapas	Se contempla manejar dos tipos de capas de Buckets para el almacenamiento de datos, uno para los datos sin procesar y otro para los datos procesados.
DA-05-CicloDeVidaDatosStream	El tiempo de duración de los datos en la base de datos para la visualización en tiempo real será de un mes.

Cuadro 3.5: Decisiones de arquitectura

3.2. Diseño

Con base en el planteamiento de los requerimientos de la arquitectura se plantea una nueva arquitectura que cumpla con las necesidades del proyecto de PhenoAgro. Para entender un poco como se va a plantear la arquitectura y entender un poco más el flujo de los datos en PhenoAgro se diagramó el proceso bajo la notación de BPMN, se puede ver a continuación:

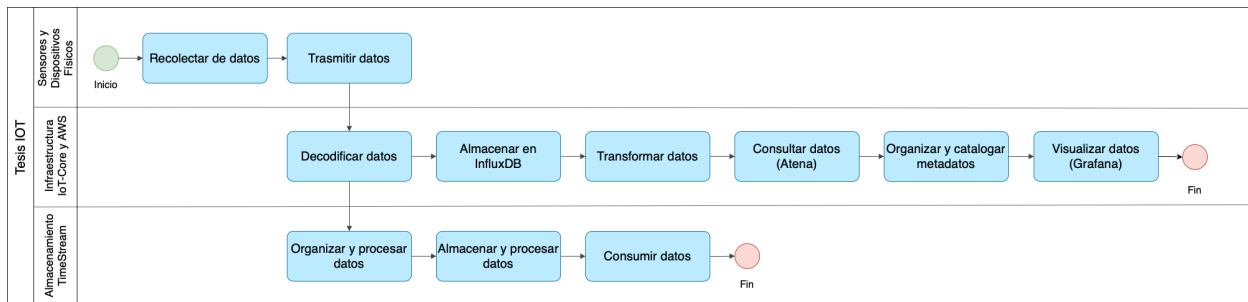


Figura 3.1: Diagrama de datos en BPMN.

A continuación para empezar a plantear la arquitectura se utilizó el modelo de documentación de arquitectura C4 donde se parte de un diagrama de contexto, en este diagrama se muestra el sistema en el contexto, el alcance del sistema incluidos los sistemas externos, los usuarios y los almacenes de datos. En la figura 3.2 se evidencia el planteamiento del contexto donde contamos con los usuarios, la base de datos para el almacenamiento de datos y los dispositivos físicos como sistemas externos.

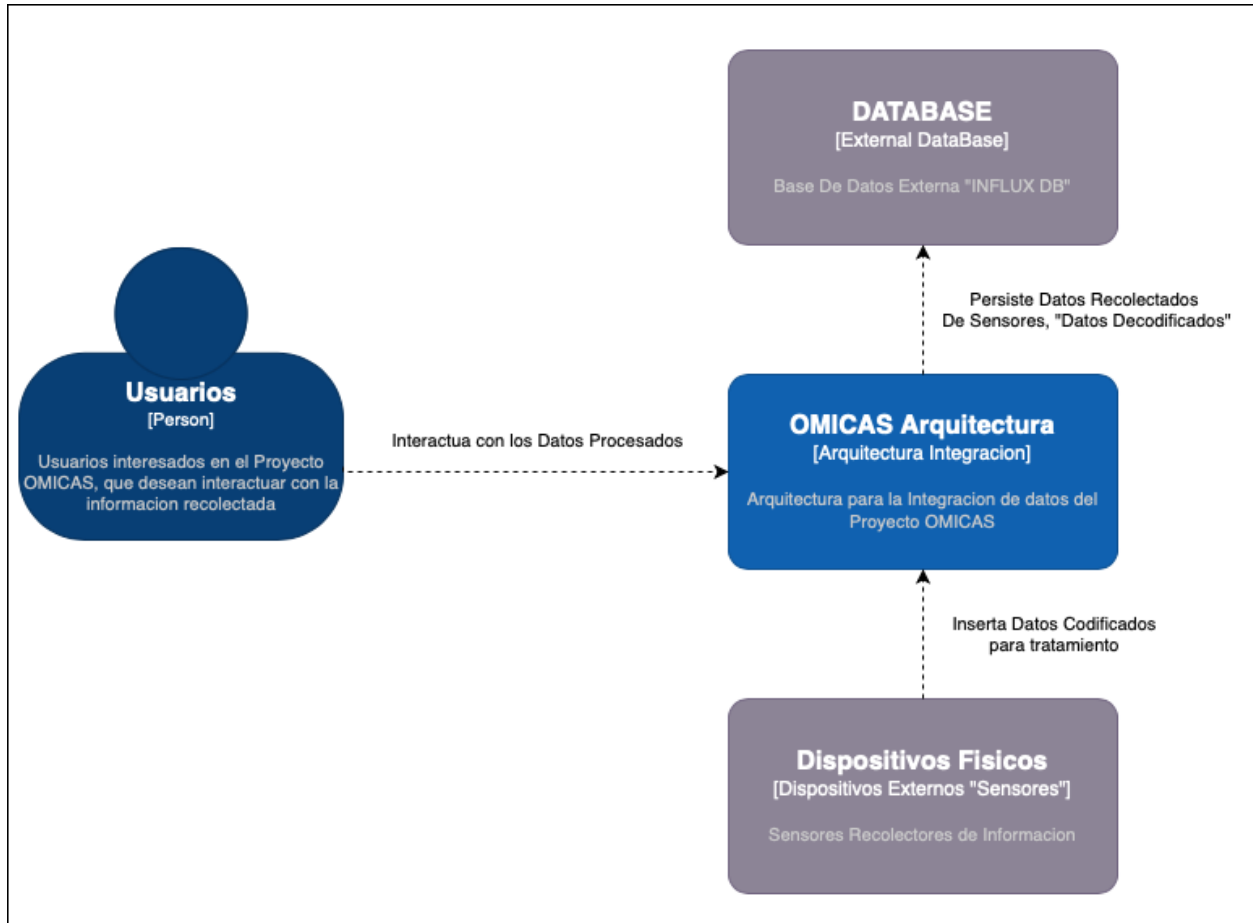


Figura 3.2: Diagrama de contexto C4.

Continuamos con el diagrama de contenedor donde se muestra los componentes de alto nivel del sistema y sus interacciones, como servidores web, bases de datos y API. En la figura 3.3 se evidencian los diferentes contenedores, se tiene un contenedor de consumo el cual va a contar con todos los componentes que se necesitan para que el usuario pueda interactuar con la plataforma. Siguiendo tenemos los contenedores de Batch y stream donde se va a realizar todo el procesamiento de la información y su distribución por el sistema. Por último se tiene el contenedor de Capa de ingesta el cual va a tener todos los componentes que se encargan de interactuar con los dispositivos físicos y obtener los datos de ellos.

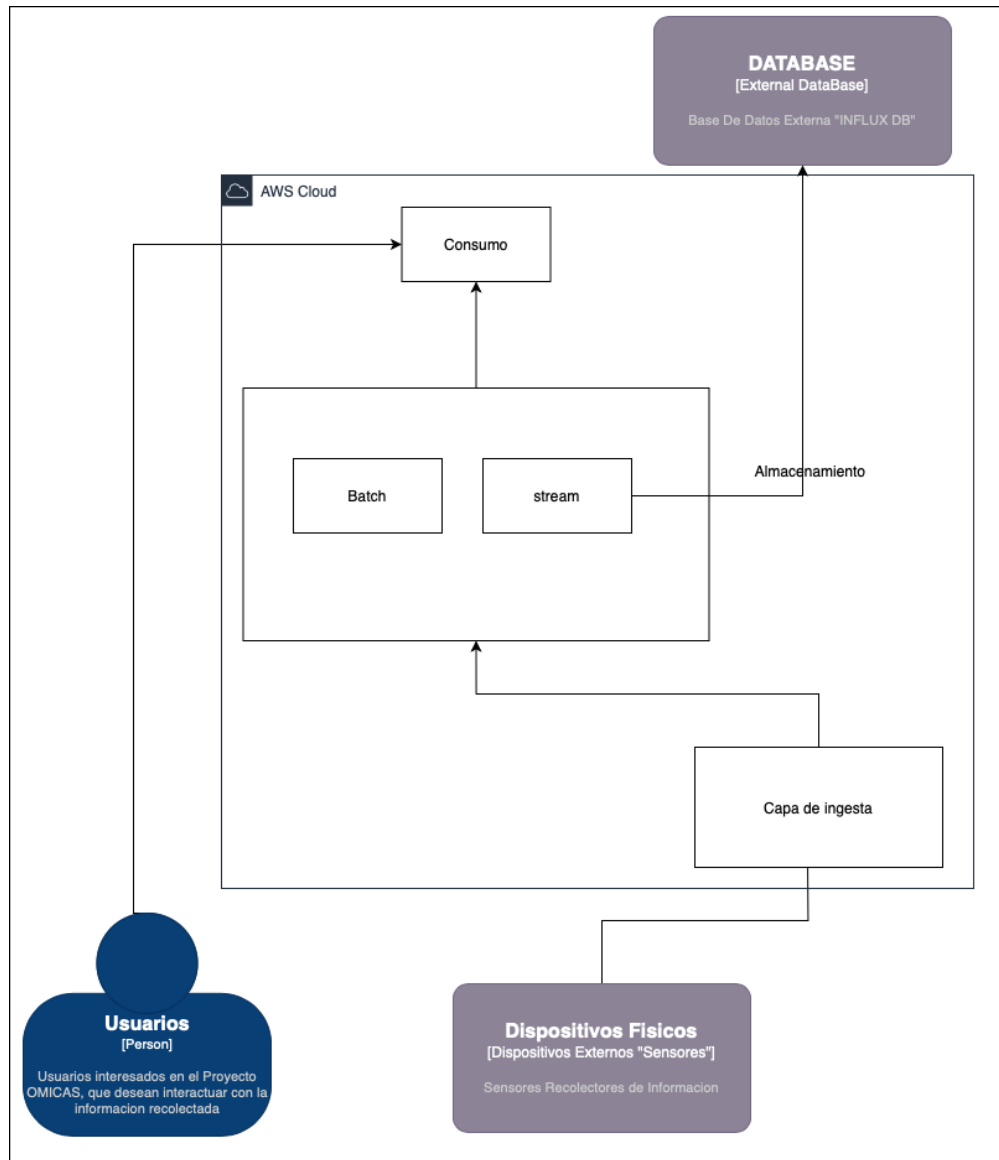


Figura 3.3: Diagrama de contenedor C4.

Para terminar con los diagramas C4, se realizó el diagrama de componentes el cual muestra los componentes internos de cada contenedor, como clases, módulos y bibliotecas.

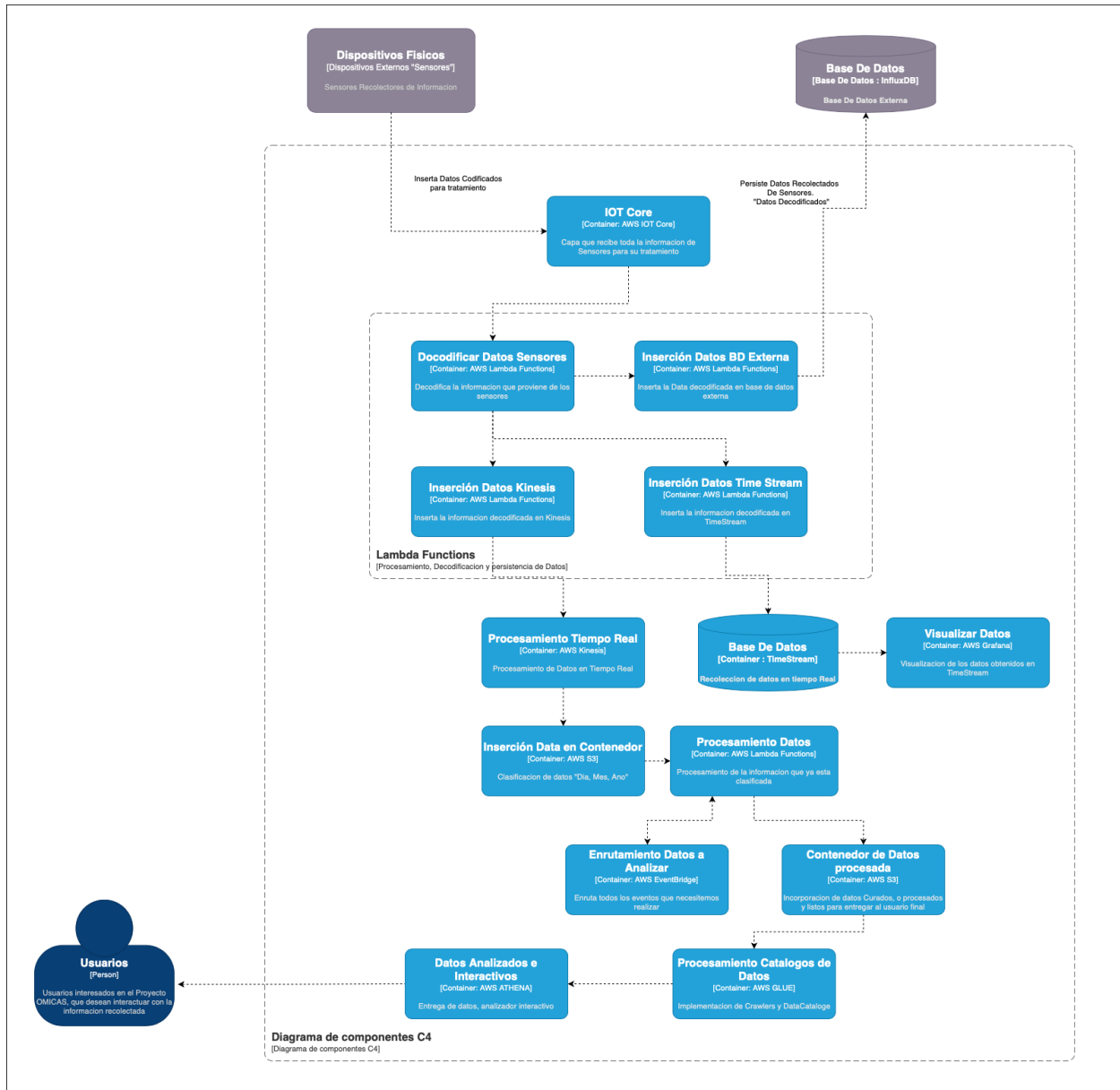


Figura 3.4: Diagrama de componentes C4.

Finalizando con los diagramas del diseño y después de seleccionar una plataforma para desplegar los componentes anteriormente diseñados se tiene el diagrama de despliegue 3.5 donde se tiene una imagen de como va a quedar la arquitectura implementada

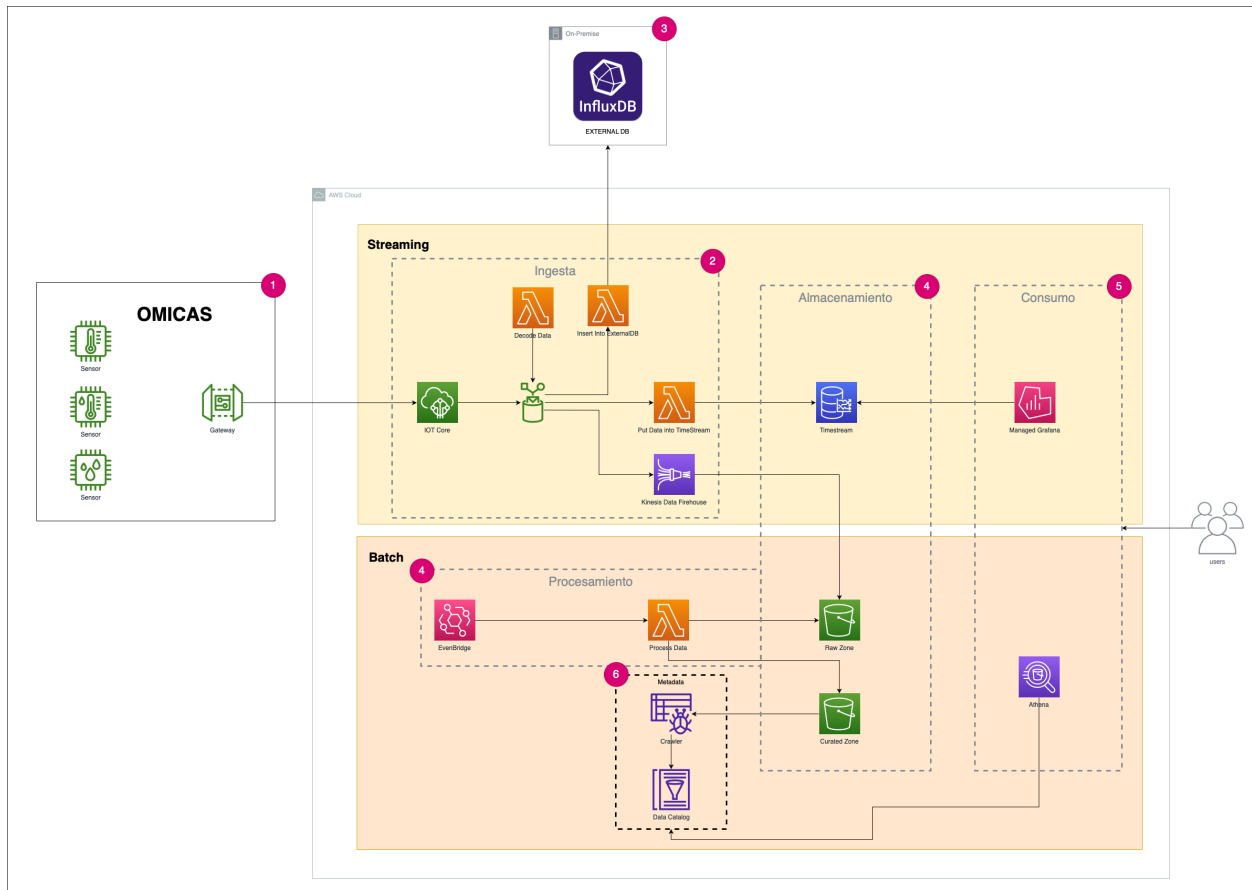


Figura 3.5: Diagrama de despliegue.

En este diagrama 3.5 contamos con las siguientes capas:

- **Capa de Dispositivos Físicos y Gateway:**

La primera capa de la arquitectura se compone de dispositivos físicos, que incluyen una amplia variedad de sensores, y una Gateway que facilita la transmisión de datos hacia la nube. Estos sensores son instrumentos especializados que recopilan información del entorno. La Gateway actúa como un punto de acceso y coordinación para asegurar que los datos se transmitan de manera eficiente.

- **Capa de Ingesta de Datos:**

En la capa de Ingesta de Datos, los datos capturados por los sensores se transmiten a través de la infraestructura de IoT-Core. Aquí, los datos recibidos se decodifican utilizando Lambda Functions específicas para luego ser transmitidos en su forma decodificada. Esta capa es esencial para la recopilación inicial de datos y su preparación para el procesamiento posterior.

- **Capa de Almacenamiento en InfluxDB:**

InfluxDB, una base de datos especializada, desempeña un papel fundamental en la arquitectura. Aquí, los datos decodificados se almacenan de manera eficiente y con un alto rendimiento. La base de datos InfluxDB la cual es una plataforma On-Premise proporcionada por el instituto OMICAS permite que el diseño e implementación nueva en AWS demuestre que puede conectarse con desarrollos anteriores y trabajar en paralelo sin comprometer en nada su funcionamiento.

- **Capa de Almacenamiento y Procesamiento de Datos:**

En esta etapa, los datos son almacenados en la base de datos TimeStream y los Bucket S3 posteriormente los datos almacenados en el Bucket S3 se organizan, clasifican y procesan para facilitar su consumo futuro. Este proceso implica la transformación de datos crudos en información significativa y útil para los usuarios finales.

- **Capa de Consumo de Datos:**

La capa de Consumo de Datos ofrece una interfaz de usuario final enriquecida. Utiliza herramientas avanzadas como Grafana y Athena para permitir a los usuarios finales comprender y analizar la información de manera efectiva. Grafana proporciona visualizaciones interactivas, mientras que Athena permite realizar consultas complejas en los datos almacenados.

- **Capa de Metadatos y Catalogación:**

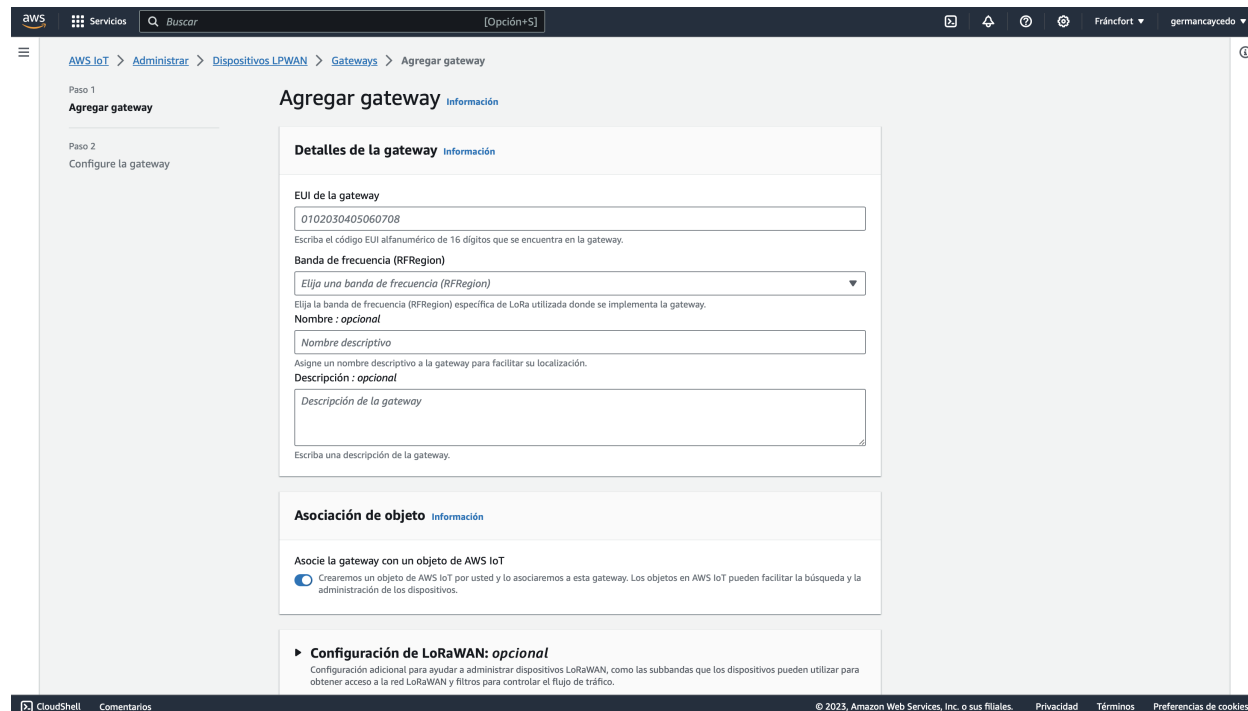
Para mantener un control completo sobre la información disponible, se emplean componentes como DataCatalog y Crawler. Estos elementos trabajan en conjunto para catalogar y mantener metadatos sobre los datos, lo que facilita la búsqueda y recuperación de información específica.

3.3. Implementación

En esta sección se va a trabajar sobre la implementación de la arquitectura diseñada en el capítulo anterior tomando como guía el diagrama de despliegue de la figura 3.5.

3.3.1. Capa de Dispositivos Físicos y Gateway

Para realizar la conexión hacia los dispositivos IoT se seleccionó el servicio de IoT Core de AWS. Lo primero que se realizó fue la conexión de el gateway Dragino que tiene el instituto OMICAS con el servicio de AWS, para esto, se creó un Gateway en la consola, la pantalla de creación de puede ver en la figura 3.6 y posteriormente se evidencia la creación correcta en la figura 3.7. El vinculo del gateway con la consola de AWS es necesario debido a que es el dispositivo por el cual se van a conectar las estaciones que contienen los sensores del Instituto a AWS y así recibir los datos que estas capturan y envían.



The screenshot shows the AWS IoT console interface for creating a gateway. The breadcrumb navigation is: AWS IoT > Administrar > Dispositivos LPWAN > Gateways > Agregar gateway. The page is divided into two main steps: Paso 1: Agregar gateway (selected) and Paso 2: Configure la gateway. The 'Agregar gateway' section contains three main panels: 1. 'Detalles de la gateway' with fields for 'EUI de la gateway' (0102030405060708), 'Banda de frecuencia (RFRegion)' (a dropdown menu), 'Nombre' (optional), and 'Descripción' (optional). 2. 'Asociación de objeto' with a radio button to 'Asocie la gateway con un objeto de AWS IoT' and a description of how objects facilitate search and management. 3. 'Configuración de LoRaWAN' (optional) with a description of subbands for device access. The footer includes 'CloudShell', 'Comentarios', and copyright information for Amazon Web Services.

Figura 3.6: Pantalla creación del gateway en consola AWS.

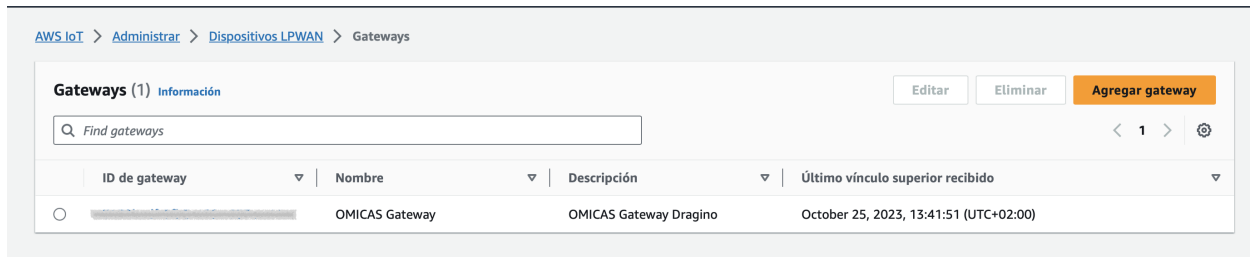


Figura 3.7: Pantalla gateway creado en consola AWS.

Posteriormente se deben crear los perfiles de los dispositivos y del servicio los cuales son necesarios para la creación y conexión de las estaciones ya que contienen toda la configuración necesaria. En la figura 3.8 se puede evidenciar la información que contiene el perfil del dispositivo. En la figura 3.9 se puede evidenciar la información que contiene el perfil del servicio.

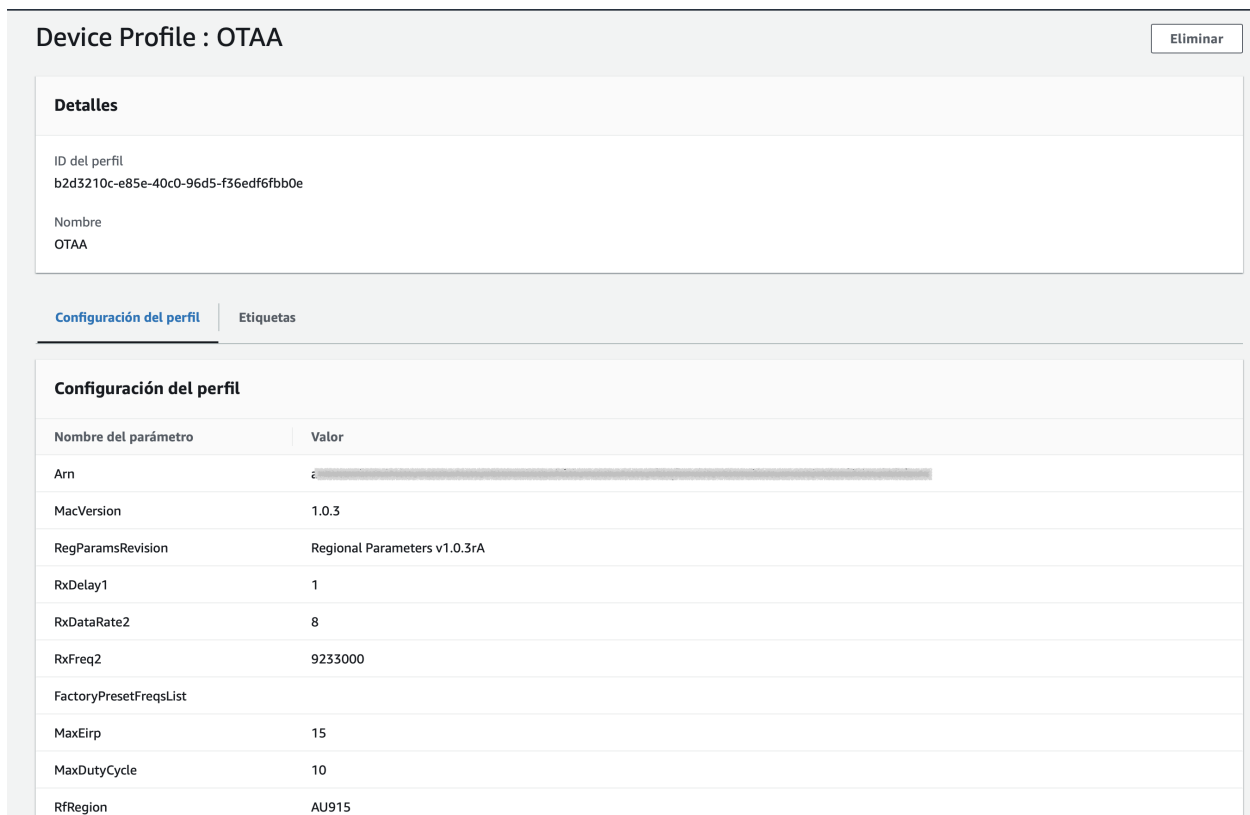
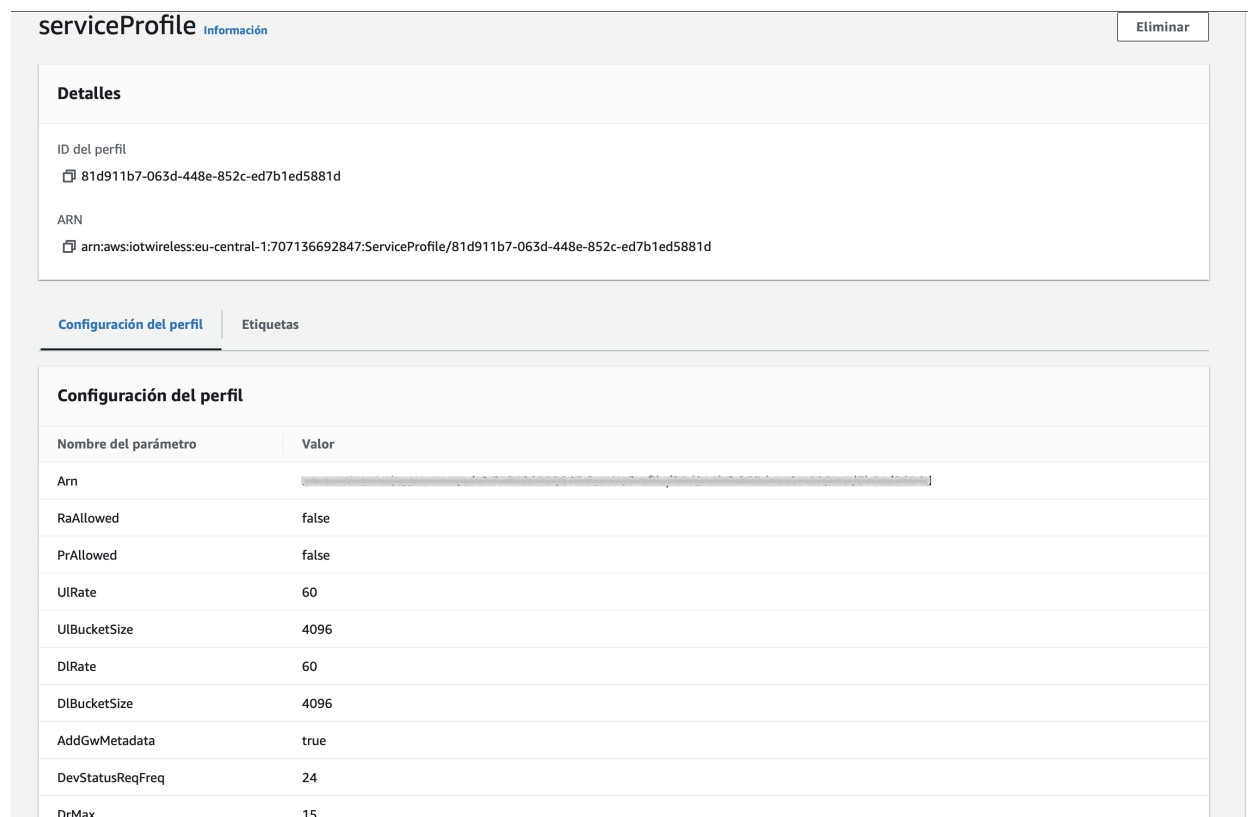


Figura 3.8: Pantalla perfil de dispositivo AWS.



The screenshot displays the AWS IoT Service Profile configuration interface. At the top, the title is "serviceProfile" with a sub-label "Información" and an "Eliminar" button. Below this, there are two main sections: "Detalles" and "Configuración del perfil".

Detalles

- ID del perfil: 81d911b7-063d-448e-852c-ed7b1ed5881d
- ARN: arn:aws:iotwireless:eu-central-1:707136692847:ServiceProfile/81d911b7-063d-448e-852c-ed7b1ed5881d

Configuración del perfil

Nombre del parámetro	Valor
Arn	
RaAllowed	false
PrAllowed	false
UIRate	60
UIBucketSize	4096
DIRate	60
DIBucketSize	4096
AddGwMetadata	true
DevStatusReqFreq	24
DrMax	15

Figura 3.9: Pantalla perfil de servicio AWS.

A continuación se creó la regla de enrutamiento de mensajes en la cual se debe especificar que es lo que se debe hacer con los mensajes que llegan de los dispositivos, como se puede evidenciar en la figura 3.10 tiene dos componentes importantes, el primero la instrucción SQL en el cual se debe especificar que información del mensaje se debe obtener y en el caso de este proyecto también se hace la decodificación del mensaje que llega codificado, para la decodificación se hace uso de una función Lambda.

Continuando con el segundo componente de esta pantalla se tienen las acciones, en esta parte se decide que hacer con el mensaje, en el caso de este proyecto se tienen cuatro acciones especificadas, la primera es republicar el mensaje en un tema de AWS IoT para poder visualizarlo y hacer una auditoría de los mensajes que llega. La segunda es la función Lambda la cual envía el mensaje a la plataforma heredada de influxDB con la que cuenta el instituto OMICAS. La tercera es la función Lambda que se encarga de escribir la información en la base de datos de TimeStream de AWS. La cuarta y última acción es de enviar el mensaje a un flujo de Amazon Kinesis Firehose.

Detalles

Descripción
IoTRule

ARN
[Redacted]

Tema
-

Fecha de creación
August 29, 2023, 17:44:26 (UTC+02:00)

Estado
Activo

Tema de ingesta básico
\$aws/rules/IoTRule

Instrucción SQL

Instrucción SQL
SELECT WirelessDeviceId, MessageId, aws_lambda("arn:aws:lambda:eu-central-1:707136692847:function:omicas-dev-decodeFunction", { "PayloadData":PayloadData }) as decodingoutput

Versión de SQL
2016-03-23

Acciones | Acción de error | Etiquetas

Acciones (4) Ver detalles

Las acciones se producen cuando se activa un evento. Se ejecutan en orden ascendente, hasta que todas las acciones se completan o se produce un error. Para agregar o eliminar acciones, deberá editar la regla.

Servicio	Acción
<input type="radio"/> Republish to AWS IoT topic	Volver a publicar mensajes en un tema de AWS IoT
<input type="radio"/> Lambda	Invocar una función de Lambda para pasar los datos del mensaje
<input checked="" type="radio"/> Lambda	Invocar una función de Lambda para pasar los datos del mensaje
<input type="radio"/> Kinesis Firehose stream	Enviar un mensaje a un flujo de Amazon Kinesis Firehose

Figura 3.10: Pantalla perfil de servicio AWS.

Una vez se tiene creada la regla de enrutamiento se debe crear un destino el cual se encarga de conectar los dispositivos LoRaWAN con la regla de enrutamiento. En la figura 3.11 se evidencian los detalles de este componente.

The screenshot displays the AWS IoT console interface for an IoT Destination. At the top, a breadcrumb trail shows the navigation path: [AWS IoT](#) > [Administrar](#) > [Dispositivos LPWAN](#) > [Destinos](#) > [IoT Destination](#). Below the breadcrumb, the page title is "IoT Destination" with a sub-link for "Información". To the right of the title are two buttons: "Editar" and "Eliminar".

The main content area is divided into two sections:

- Detalles:** A table with three columns. The first column, "Nombre de destino", contains the value "IoT Destination". The second column, "Expresión", contains "IoTRule". The third column, "Tipo de expresión", contains "RuleName". Below this, the "Descripción" field is empty, and the "Nombre de rol" field contains the value "AWSIoTWirelessDestination-1DwTEnBl".
- Etiquetas (0):** A section with a table header containing "Clave" and "Valor". Below the header, the text "Sin etiquetas" is displayed, followed by the message "No tiene ninguna etiqueta asociada a este recurso." To the right of this section is a button labeled "Administrar etiquetas".

Figura 3.11: Pantalla perfil de servicio AWS.

Ya se tienen creados los componentes para poder crear el dispositivo LoRaWAN en la consola de AWS IoT Core y así realizar la conexión con la estación del instituto OMICAS, para eso se debe crear y configurar un nuevo dispositivo en el menú de LPWAN, primero se deben de llenar todas las especificaciones del dispositivo como se ve en la figura 3.12, esta información la proporcionaron los ingenieros encargados del proyecto de PhenoAgro del instituto OMICAS.

[AWS IoT](#) > [Administrar](#) > [Dispositivos LPWAN](#) > [Dispositivos](#) > [Agregar dispositivo](#)

Paso 1
Configurar dispositivo

Paso 2 - *opcional*
Establecer la posición del dispositivo

Configurar dispositivo

Especificación de LoRaWAN y configuración de dispositivo inalámbrico [Información](#)

Especificación de dispositivo inalámbrico
Las especificaciones del dispositivo se componen de la versión LoRaWAN (1.1 o 1.0.x) y el proceso de autenticación (autenticación Over The Air o autenticación por personalización). Una vez seleccionados, los datos se cifran con una clave que AWS posee y administra por usted.

OTAA v1.0.x ▼

DevEUI
0102030405060708
Valor de DevEUI hexadecimal de 16 dígitos que se encuentra en su dispositivo inalámbrico.

AppKey
01020304050607080910111213141516
Valor de AppKey hexadecimal de 32 dígitos proporcionado por el proveedor de dispositivos inalámbricos.

AppEUI
0102030405060708
AppEUI hexadecimal de 16 dígitos proporcionado por el proveedor de dispositivos inalámbricos.

Nombre del dispositivo inalámbrico: *opcional*
Nombre del dispositivo
Un nombre descriptivo para facilitar la localización del dispositivo inalámbrico.

Descripción del dispositivo inalámbrico: *opcional*
Descripción del dispositivo inalámbrico.
Una descripción útil para el dispositivo inalámbrico.

Figura 3.12: Pantalla especificaciones nuevo dispositivo Lora AWS.

A continuación de deben de especificar el perfil de dispositivo y el perfil del servicio los cuales fueron creados anteriormente y por último se debe especificar el destino de los mensajes el cual también fue creado anteriormente. Se puede evidenciar en la figura 3.13

Perfiles

Perfil de dispositivo inalámbrico
Elija un perfil de dispositivo inalámbrico para que el dispositivo pueda transmitir los mensajes correctos a la gateway.

Elegir un perfil de dispositivo ▼

Perfiles del servicio
Elija un perfil de servicio.

Elegir un perfil de servicio ▼

Etiquetas: opcional
Una etiqueta es una marca que se asigna a un recurso de AWS. Cada etiqueta consta de una clave y un valor opcional. Puede utilizar etiquetas para buscar y filtrar los recursos o realizar un seguimiento de los costos de AWS.

No tiene ninguna etiqueta asociada a este recurso.

Agregar nueva etiqueta

Puede agregar hasta 50 etiquetas.

Elegir destino

Nombre de destino
Los destinos dirigen los mensajes LoRaWAN desde su dispositivo inalámbrico a otros servicios de AWS.

Nombre de destino ▼

Cancelar **Siguiente**

Figura 3.13: Pantalla selección perfiles y destino nuevo dispositivo Lora AWS.

Con los anteriores pasos terminados ya se puede ver el dispositivo creado en la consola de AWS en la opción de Dispositivos LoRaWAN en el servicio de AWS IoT Core como se evidencia en la figura 3.14

The screenshot shows the AWS IoT console interface for managing LoRaWAN devices. The breadcrumb navigation is 'AWS IoT > Administrar > Dispositivos LPWAN > Dispositivos'. The page title is 'Dispositivos LoRaWAN (7) Información'. There are buttons for 'Editar', 'Eliminar', and 'Agregar dispositivo inalámbrico'. A search bar contains the text 'Buscar dispositivos LoRaWAN'. Below the search bar is a table with the following data:

	ID de dispositivo	Nombre	Destino	Último vínculo super...	ARN	DevEUI
<input type="radio"/>	5f3b1b28-09a4-4068-a...	Primogénita	IoTDestination	October 25, 2023, 04:42...	arn:aws:iotwireless:eu-c...	
<input type="radio"/>	0f6ae382-b503-416b-8...	Satélite Cambridge	IoTDestination	October 25, 2023, 15:05...	arn:aws:iotwireless:eu-c...	
<input type="radio"/>	af5d1b31-376a-46b3-b...	Satélite DaarD	IoTDestination	-	arn:aws:iotwireless:eu-c...	
<input type="radio"/>	4fd4108c-abe2-47ac-93...	Satélite CIAT	IoTDestination	October 24, 2023, 04:01...	arn:aws:iotwireless:eu-c...	
<input type="radio"/>	cd4caaab-993a-4751-ad...	Satélite EJR	IoTDestination	-	arn:aws:iotwireless:eu-c...	
<input type="radio"/>	93bd7d58-bead-49be-8...	Tripas	IoTDestination	October 24, 2023, 22:44...	arn:aws:iotwireless:eu-c...	
<input type="radio"/>	6c3aa723-26bb-4aeb-b...	MPC-2	IoTDestination	August 31, 2023, 23:14...	arn:aws:iotwireless:eu-c...	

Figura 3.14: Pantalla Dispositivos LoRaWAN consola AWS.

Con los dispositivos agregados correctamente ya ellos se pueden conectar con el gateway Dragino también agregado previamente y así enviar los mensajes al servicio de AWS IoT Core finalizando con la implementación de la Capa de Dispositivos Físicos y Gateway.

Con la implementación de estos componentes de la capa de dispositivos físicos y gateway, de la arquitectura se cumple con los requisitos funcionales RF-01, RF-02 y RF-03. Además, se cumple con los requisitos no funcionales RNF-01, RNF-02 y con los requisitos de interfaz RI-01.

3.3.2. Capa de Ingesta de Datos

En la capa de ingesta de datos se tienen cuatro componentes, el primero es la función Lambda la cual se encarga de decodificar el mensaje que llega de los dispositivos LoRaWAN y entregar la información como un JSON la cual pueda ser utilizada posteriormente, figura 3.15.

Segundo, se tiene la función Lambda que se encarga de con el JSON decodificado escribir en la base de datos On-Premise InfluxDB del instituto OMICAS, figura 3.16.

Tercero, se tiene la función Lambda que se encarga de escribir la información que llega como mensaje del dispositivo y que ha sido decodificada en la base de datos AWS TimeStream la cual se usa para almacenar la información que va a ser visualizada en tiempo real, figura 3.17.

Cuarto, se cuenta con el servicio de AWS Kinesis Data Firehouse, ver figura 3.18, el cual se encarga de recopilar los mensajes y agruparlos durante una cantidad de tiempo especificada y una vez pasado ese tiempo los mensajes recibidos y agrupados son almacenados en el servicio de AWS

S3. En este servicio se pueden configurar diferentes variables como lo son: el destino que en el caso de este proyecto es un Bucket S3, se puede ver en la figura 3.19, también se puede configurar el almacenamiento en el búfer que se decidió hacer hasta un tiempo de 300 segundo o hasta que se cuenten con hasta 128 MiB en mensajes, se puede ver en la figura 3.20, por otra parte, se puede configurar como hacer el particionamiento dinámico al momento de almacenar la información que en este caso fue año/mes/día, se puede ver en la figura 3.21, finalmente se cuenta con una pantalla de monitoreo que cuenta con diferentes gráficas que permiten visualizar el comportamiento de la secuencia de entrega, ver en la figura 3.22

The screenshot displays the AWS Lambda console interface for the function 'omicas-dev-decodeFunction'. At the top, the breadcrumb navigation shows 'Lambda > Funciones > omicas-dev-decodeFunction'. The function name is prominently displayed, along with buttons for 'Limitación', 'Copiar ARN', and 'Acciones'. A notification bar indicates that the function belongs to an application and provides a link to manage it. The main section is titled 'Información general de la función' and contains several components: a central box for the function name and 'Layers' (0); a 'Funciones relacionadas' dropdown menu; a 'Descripción' field; 'Última modificación' set to 'anteayer'; 'ARN de la función' with a copy icon; 'Aplicación' set to 'omicas-dev'; and 'URL de la función' with an 'Información' link. On the left, there are boxes for 'API Gateway' and 'AWS IoT', each with an '+ Agregar desencadenador' button. A '+ Agregar destino' button is also present in the center.

Figura 3.15: Pantalla función Lambda de decodificar consola AWS.

The screenshot displays the AWS Lambda console for the function `omicas-dev-pushDecodeIntoInfluxDB`. The breadcrumb navigation shows `Lambda > Funciones > omicas-dev-pushDecodeIntoInfluxDB`. At the top right, there are buttons for `Limitación`, `Copiar ARN`, and `Acciones`. A notification bar states: `Esta función pertenece a una aplicación. Haga clic aquí para administrarla.`

The main section is titled `Información general de la función` and includes the following details:

- Función:** `omicas-dev-pushDecodeIntoInfluxDB`
- Layers:** (0)
- Funciones relacionadas:** `Seleccionar una función`
- Descripción:** -
- Última modificación:** `anteayer`
- ARN de la función:** `arn:aws:lambda:us-east-1:123456789012:func:omicas-dev-pushDecodeIntoInfluxDB`
- Aplicación:** `omicas-dev`
- URL de la función:** `Información`

On the left side, there are triggers for `API Gateway` and `AWS IoT`, along with a `+ Agregar desencadenador` button. A `+ Agregar destino` button is also present.

Figura 3.16: Pantalla función Lambda de almacenar en InfluxDB consola AWS.

The screenshot displays the AWS Lambda console for the function `omicas-dev-insertIntoTimeStream`. The breadcrumb navigation shows `Lambda > Funciones > omicas-dev-insertIntoTimeStream`. At the top right, there are buttons for `Limitación`, `Copiar ARN`, and `Acciones`. A notification bar states: `Esta función pertenece a una aplicación. Haga clic aquí para administrarla.`

The main section is titled `Información general de la función` and includes the following details:

- Función:** `omicas-dev-insertIntoTimeStream`
- Layers:** (0)
- Funciones relacionadas:** `Seleccionar una función`
- Descripción:** -
- Última modificación:** `anteayer`
- ARN de la función:** `arn:aws:lambda:us-east-1:123456789012:func:omicas-dev-insertIntoTimeStream`
- Aplicación:** `omicas-dev`
- URL de la función:** `Información`

On the left side, there are triggers for `API Gateway` and `AWS IoT`, along with a `+ Agregar desencadenador` button. A `+ Agregar destino` button is also present.

Figura 3.17: Pantalla función Lambda de almacenar en AWS TimeStream consola AWS.

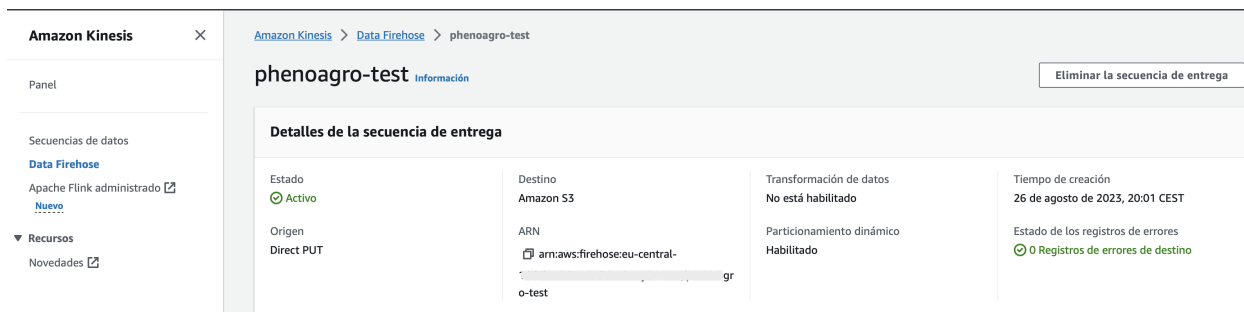


Figura 3.18: Pantalla secuencia de entrega Kinesis Data Firehouse consola AWS.

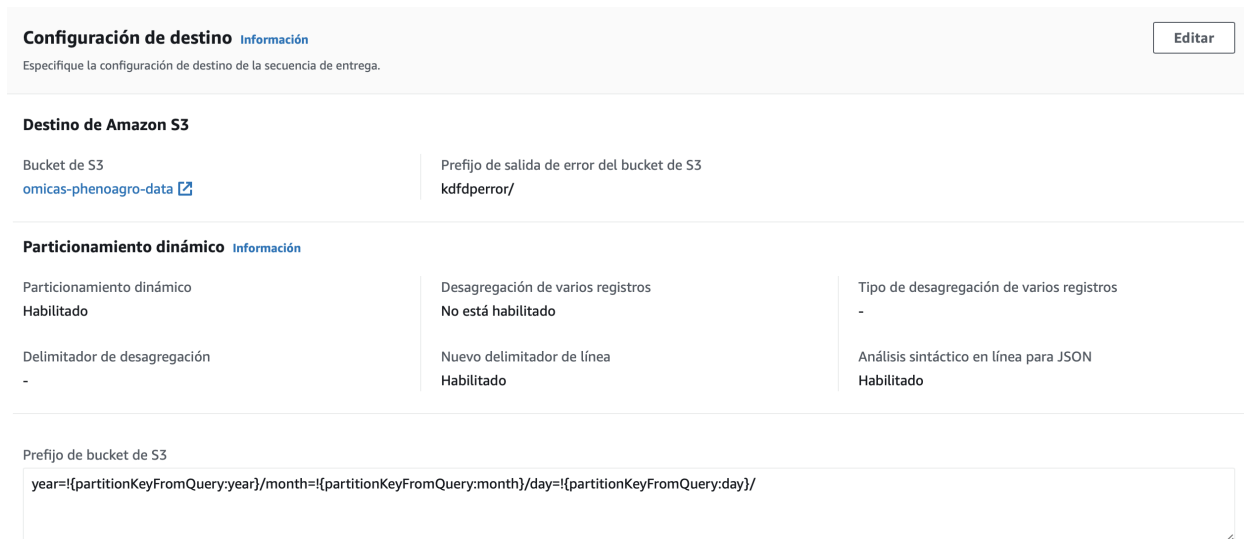


Figura 3.19: Pantalla configuración destino Kinesis Data Firehouse consola AWS.



Figura 3.20: Pantalla configuración búfer Kinesis Data Firehouse consola AWS.

Claves de particionamiento dinámico (3)	
<input type="text" value="Buscar claves de particionamiento dinámico"/> < 1 >	
Nombre de la clave	Expresión JQ
year	.decodingoutput.year
month	.decodingoutput.month
day	.decodingoutput.day

Figura 3.21: Pantalla configuración particionamiento Kinesis Data Firehouse consola AWS.

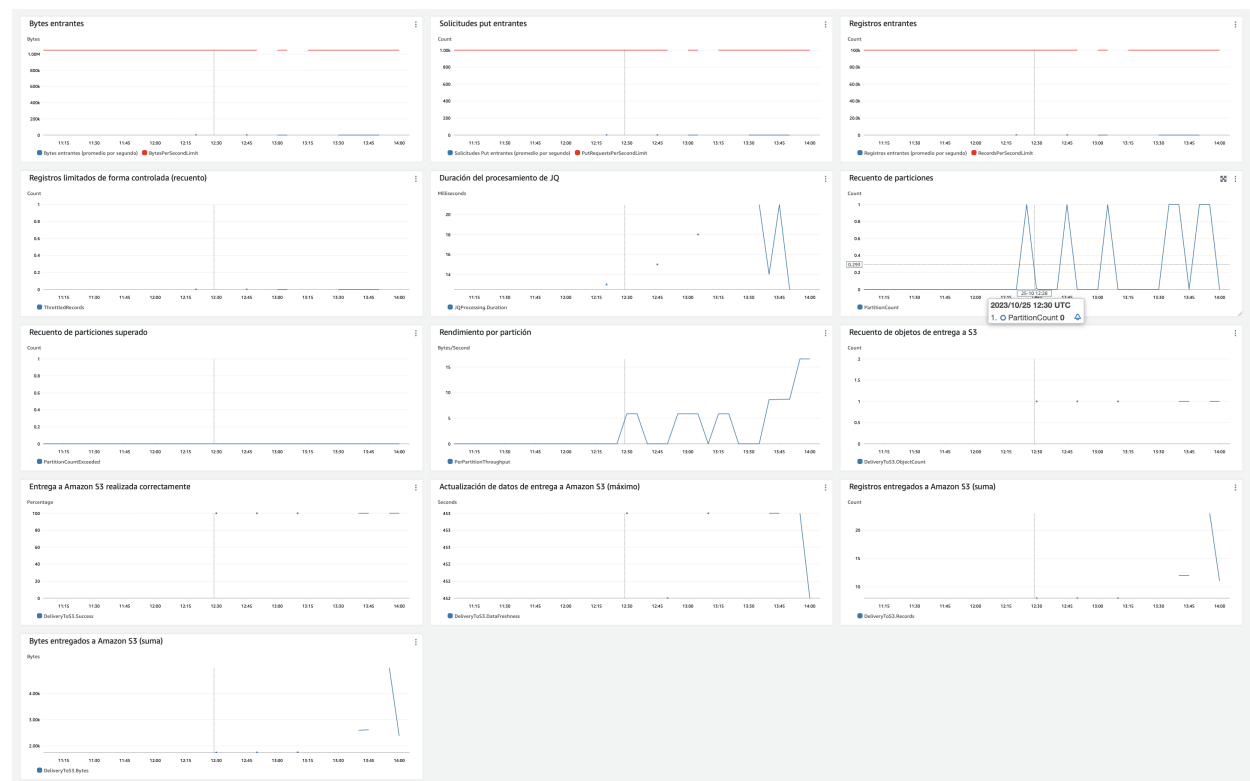


Figura 3.22: Pantalla monitoreo Kinesis Data Firehouse consola AWS.

Con la implementación de estos componentes de la capa de ingesta de datos de la arquitectura se cumple con los requisitos funcionales RF-04, RF-05 y RF-03. Además, se cumple con los requisitos no funcionales RNF-04, RNF-05 y con los requisitos de interfaz RI-01.

3.3.3. Capa de Almacenamiento y Procesamiento de Datos

En la capa de almacenamiento y procesamiento de datos se tienen cinco componentes, tres para almacenamiento y dos para el procesamiento. El primer componente de almacenamiento que se

implementó es la base de datos AWS TimeStream, esta es una base de datos de serie temporal la cual permite almacenar la información de una manera rápida y escalable, además puede ser consultada mediante SQL para el consumo de la información. En la figura 3.23 se puede evidenciar la información de la base de datos creada y la tabla que la compone. A continuación, se puede acceder a un editor de consultas donde en la figura 3.24 se puede evidencia los campos con los que cuenta la tabla y una vez realizada una consulta SQL en la figura 3.25 se puede evidenciar como queda guardada la información en la tabla.

The screenshot displays the AWS Timestream console interface. At the top, the breadcrumb navigation shows 'Timestream > Bases de datos > omicas-device-database'. The main header for the database is 'omicas-device-database', with 'Editar' and 'Eliminar' buttons to its right. Below this is a 'Resumen' (Summary) section containing a table with the following details:

Nombre	omicas-device-database	Hora de modificación (UTC)	30/9/2023, 3:46:15	Hora de creación (UTC)	26/7/2023, 3:23:51
ARN de base de datos	arn:aws:timestream:eu-central-1:123456789012:database/omicas-device-database	ARN de clave de KMS	arn:aws:kms:eu-central-1:123456789012:key/4c2b-9c36-247e2d7ad902		

Below the summary, there are tabs for 'Supervisión', 'Tablas', and 'Etiquetas'. The 'Tablas' tab is active, showing a table with one entry:

Nombre de la tabla	Hora de creación (UTC)
omicas-device-table	30/8/2023, 23:32:19

At the top of the 'Tablas' section, there are buttons for 'Crear copia de seguridad', 'Crear consulta programada', 'Editar', 'Eliminar', and 'Crear tabla'. A search filter is also present.

Figura 3.23: Pantalla Base de Datos Timestream consola AWS.

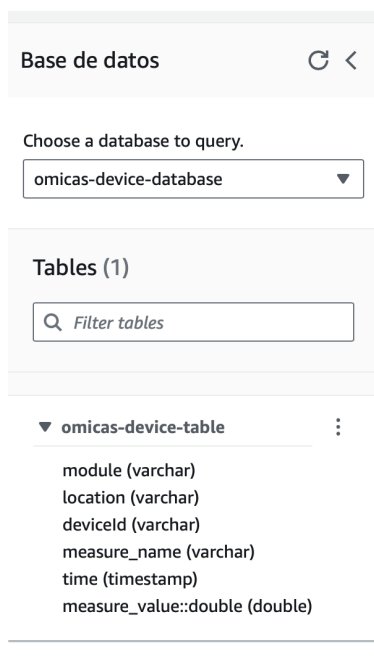


Figura 3.24: Pantalla estructura de la tabla de Timestream consola AWS.

Resultados de filas (10)

Q Filtrar

< 1 > ⚙

module	location	deviceId	measure_name	time	measure_value::double
Potencia	OmicasTest	[REDACTED]	Longitude	2023-10-25 15:41:27.039000000	0.0
Potencia	OmicasTest	[REDACTED]	Latitude	2023-10-25 15:41:23.977000000	0.0
Potencia	OmicasTest	[REDACTED]	SoilHum	2023-10-25 15:41:21.402000000	0.0
Potencia	OmicasTest	[REDACTED]	SoilTemp	2023-10-25 15:41:19.287000000	0.0
Potencia	OmicasTest	[REDACTED]	SoilPot	2023-10-25 15:41:19.242000000	0.0
Potencia	OmicasTest	[REDACTED]	SoilPhos	2023-10-25 15:41:17.098000000	0.0
Potencia	OmicasTest	[REDACTED]	SoilNit	2023-10-25 15:41:17.054000000	0.0
Potencia	OmicasTest	[REDACTED]	SoilCond	2023-10-25 15:41:15.094000000	0.0
Potencia	OmicasTest	[REDACTED]	SoilPh	2023-10-25 15:41:15.048000000	0.0
Potencia	OmicasTest	[REDACTED]	AirDir	2023-10-25 15:41:13.024000000	0.0

Figura 3.25: Pantalla resultados consulta tabla de Timestream consola AWS.

El segundo componente de almacenamiento sería el Bucket S3 denominado Raw Zone debido a que en este componente se guarda la información tal cual como llega de los dispositivos IoT para su procesamiento futuro, la estructura en la que se guardan los mensajes es la definida en Kinesis Data Firehouse de año/mes/día y el nombre de cada archivo comienza por el timestamp del mensaje y continúa con un id del mensaje generado automáticamente, lo anterior se puede evidenciar en la figura 3.26.

Amazon S3 ×

Amazon S3 > Buckets > omicas-phenoagro-data > year=2023/ > month=10/ > day=25/

day=25/ Copiar URI de S3

Objetos Propiedades

Objetos (38)

Los objetos son las entidades fundamentales que se almacenan en Amazon S3. Puede utilizar el [inventario de Amazon S3](#) para obtener una lista de todos los objetos de su bucket. Para que otras personas obtengan acceso a sus objetos, tendrá que concederles permisos de forma explícita. [Más información](#)

Copiar URI de S3 Copiar URL Descargar Abrir Eliminar Acciones Crear carpeta Cargar

Buscar objetos por prefijo

<input type="checkbox"/>	Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
<input type="checkbox"/>	phenoagro-test-6-2023-10-25-00-03-58-54d65dc4-ae95-3771-9059-1557d62fa958	-	25 Oct 2023 2:11:31 AM CEST	2.6 KB	Estándar
<input type="checkbox"/>	phenoagro-test-6-2023-10-25-00-10-20-b2ea1513-b050-3d94-83d2-19697ec45e5f	-	25 Oct 2023 2:17:54 AM CEST	4.3 KB	Estándar
<input type="checkbox"/>	phenoagro-test-6-2023-10-25-00-16-41-b66f100c-ac02-33be-b8b3-57ea9b10358e	-	25 Oct 2023 2:24:16 AM CEST	2.6 KB	Estándar
<input type="checkbox"/>	phenoagro-test-6-2023-10-25-00-23-06-dfdeabf6-f902-31b3-8ac3-1665b77ebe27	-	25 Oct 2023 2:30:41 AM CEST	4.3 KB	Estándar
<input type="checkbox"/>	phenoagro-test-6-2023-10-25-00-29-27-98be226c-0d43-341f-8976-6a7419aa5fbc	-	25 Oct 2023 2:37:01 AM CEST	5.1 KB	Estándar

Figura 3.26: Pantalla objetos del Bucket S3 Raw Zone consola AWS.

El tercer componente de almacenamiento es el Bucket S3 denominado Curated Zone debido a que en este componente se guarda la información después de ser procesada, el procesamiento ocurre una vez por día y se procesan todos los mensajes almacenados en el Bucket S3 denominado Raw Zone en un día se guardan en un solo archivo .parquet en el Bucket S3 denominado Curated Zone. La estructura en la que se guardan los mensajes es la definida en Kinesis Data Firehouse de año/mes/día y el nombre comienza por la palabra file y continua por un id único generado automáticamente, lo anterior se puede evidencia en la figura 3.27.

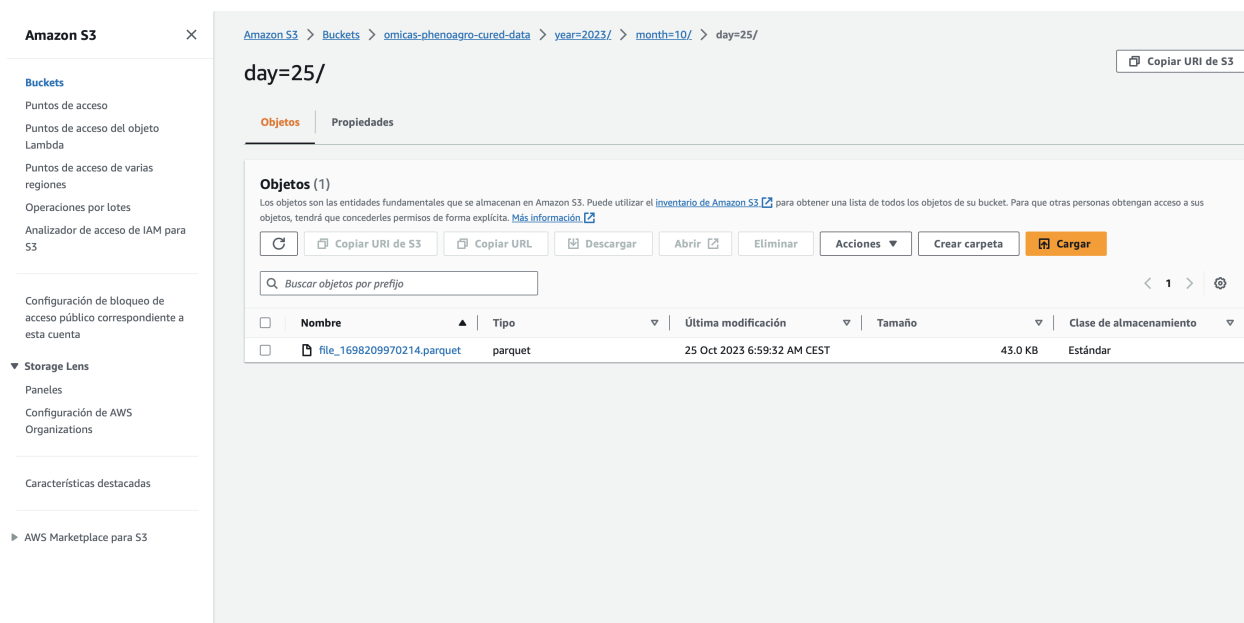


Figura 3.27: Pantalla objetos del Bucket S3 Curated Zone consola AWS.

Guardar la información en los Buckets S3 permite que se pueda almacenar en largos periodos de tiempos, en grandes cantidades de manera flexible y escalable a bajos costos.

Continuando con la parte de procesamiento de datos se tienen dos componentes, primero el componente EventBridge el cual se encarga de programar la ejecución de un servicio, en este caso se programa la ejecución de una función Lambda la cual se encarga de procesar la información que se encuentra en el Bucket S3 denominado Raw Zone y la guarda en el Bucket S3 Curated Zone. En la figura 3.28 se puede evidenciar como se ha realizado la programación de la ejecución para que se ejecute de manera diaria a las 23:59 y como zona horaria se toma América/Bogota. En la figura 3.29 se puede evidenciar como se ha configurado que el destino de la programación sea ejecutar la función Lambda encargada de procesar la información.

Detalles de programación

<p>Nombre de la programación dataProcessLambda</p> <p>Descripción dataProcessLambda</p> <p>Nombre del grupo de programación default</p>	<p>Estado ✔ Habilitado</p> <p>ARN de la programación arn:aws:scheduler:eu-central-</p> <p>Acción después de la finalización NONE</p>	<p>Hora de inicio de la programación -</p> <p>Hora de finalización de la programación -</p> <p>Zona horaria de ejecución America/Bogota</p>	<p>Intervalo de tiempo flexible -</p> <p>Fecha de creación Aug 28, 2023, 19:27:59 (UTC+02:00)</p> <p>Fecha de la última modificación Aug 29, 2023, 17:56:41 (UTC+02:00)</p>
--	---	--	--

Programación
Destino
Política de reintentos
Cola de mensajes fallidos
Cifrado

Programación

Expresión Cron [Información](#)

59
23
?

Minutos
Horas
Día del mes
Mes
Día de la semana
Año

Copiar expresión Cron

Siguientes 10 fechas de activación
La fecha y la hora se muestran en la zona horaria seleccionada para la que este horario está configurado en formato UTC, por ejemplo, "miércoles 9 de noviembre de 2022 a las 09:00 (UTC - 08:00)"

Wed, 25 Oct 2023 23:59:00 (UTC-05:00)
Thu, 26 Oct 2023 23:59:00 (UTC-05:00)
Fri, 27 Oct 2023 23:59:00 (UTC-05:00)

Figura 3.28: Pantalla Amazon EventBridge Programación consola AWS.

dataProcessLambda

Desactivar
Editar
Eliminar

Detalles de programación

Nombre de la programación dataProcessLambda	Estado ✔ Habilitado	Hora de inicio de la programación -	Intervalo de tiempo flexible -
Descripción dataProcessLambda	ARN de la programación arn:aws:scheduler:eu-central-	Hora de finalización de la programación -	Fecha de creación Aug 28, 2023, 19:27:59 (UTC+02:00)
Nombre del grupo de programación default	Acción después de la finalización NONE	Zona horaria de ejecución America/Bogota	Fecha de la última modificación Aug 29, 2023, 17:56:41 (UTC+02:00)

Programación
Destino
Política de reintentos
Cola de mensajes fallidos
Cifrado

Destino Información

Destino omicas-dev-processDataFunction	ARN del destino arn:aws:lambda:eu-central-	Rol de ejecución Amazon_EventBridge_Scheduler_LAMBDA_6c35be4205
Servicio AWS Lambda		
API Invoke		
Carga -		

Figura 3.29: Pantalla Amazon EventBridge Destino consola AWS.

El segundo componente de la capa de procesamiento de datos es la función Lambda que se encarga de recopilar todos los mensajes recolectados por los dispositivos en un día y los organiza en un solo archivo .parquet de manera estructurada para después la misma función almacenarlo el archivo en el Bucket S3 denominado Curated Zone. Ver figura 3.30.

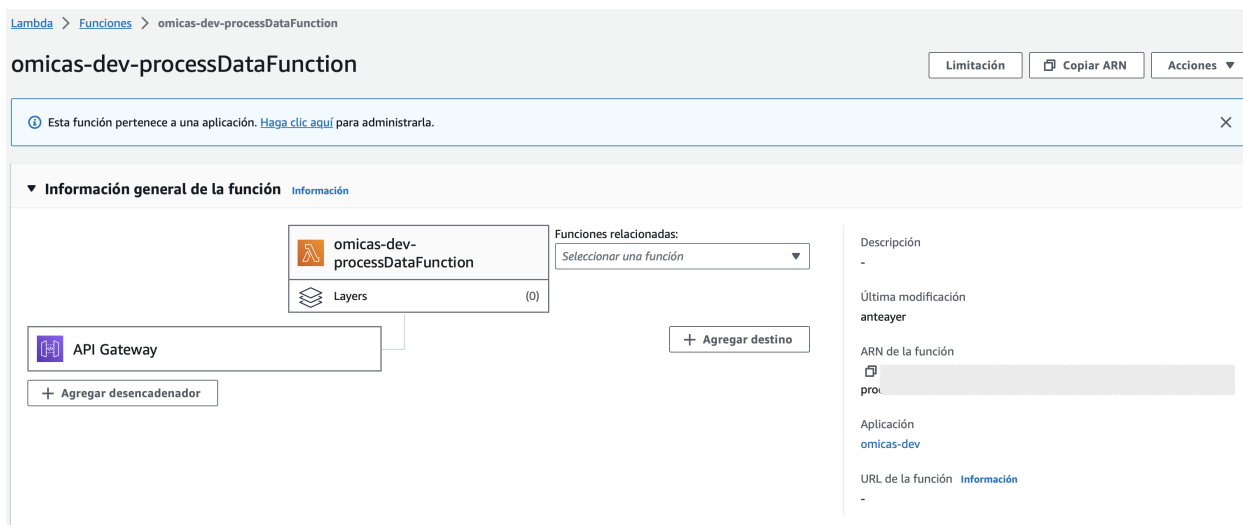


Figura 3.30: Pantalla función Lambda de procesar la información y almacenarla consola AWS.

Con la implementación de estos componentes de la capa de ingesta de datos de la arquitectura se cumple con los requisitos funcionales RF-07. Además, se cumple con los requisitos no funcionales RNF-04 y con los requisitos de interfaz RI-01.

3.3.4. Capa de Consumo de Datos

La capa de consumo de datos es la encargada de tener los componentes con los cuales el usuario va a poder interactuar, se implementaron dos, el primero es un servicio de Grafana manejado por AWS, con Grafana se habilita la posibilidad de crear dashboard para mostrar los datos en tiempo real que llegan de los dispositivos IoT, con estos dashboards el usuario puede monitorear las diferentes variables que los sensores en las estaciones de PhenoAgro obtienen. Para cada una de las estaciones se creó un grupo de dashboards como se evidencia en la figura 3.31, donde el Módulo Satélite Cambridge cuenta con tres dashboards, uno de variables atmosféricas, otro de variables de potencia y uno de variables de suelo.

El primer dashboard figura 3.32 muestra la información relacionada a las variables de potencia con las que cuenta la estación, Corriente y Voltaje de diferentes componentes, por otra parte, muestra la información relacionada a la ubicación de la estación.

El dashboard siguiente es el de variables atmosféricas figura 3.33 este dashboard muestra la información recolectada por los sensores de variables atmosféricas como lo son temperatura del aire, humedad, entre otros. Además, este dashboard cuenta con el valor en tiempo real y la tendencia de los valores recolectados en un intervalo de tiempo previamente seleccionado.

El dashboard de variables de suelo figura 3.34 también cuenta con dos partes, los datos en tiempo

real de las diferentes variables de suelo y la tendencia de los valores recolectados en un intervalo de tiempo previamente seleccionado figura 3.35.

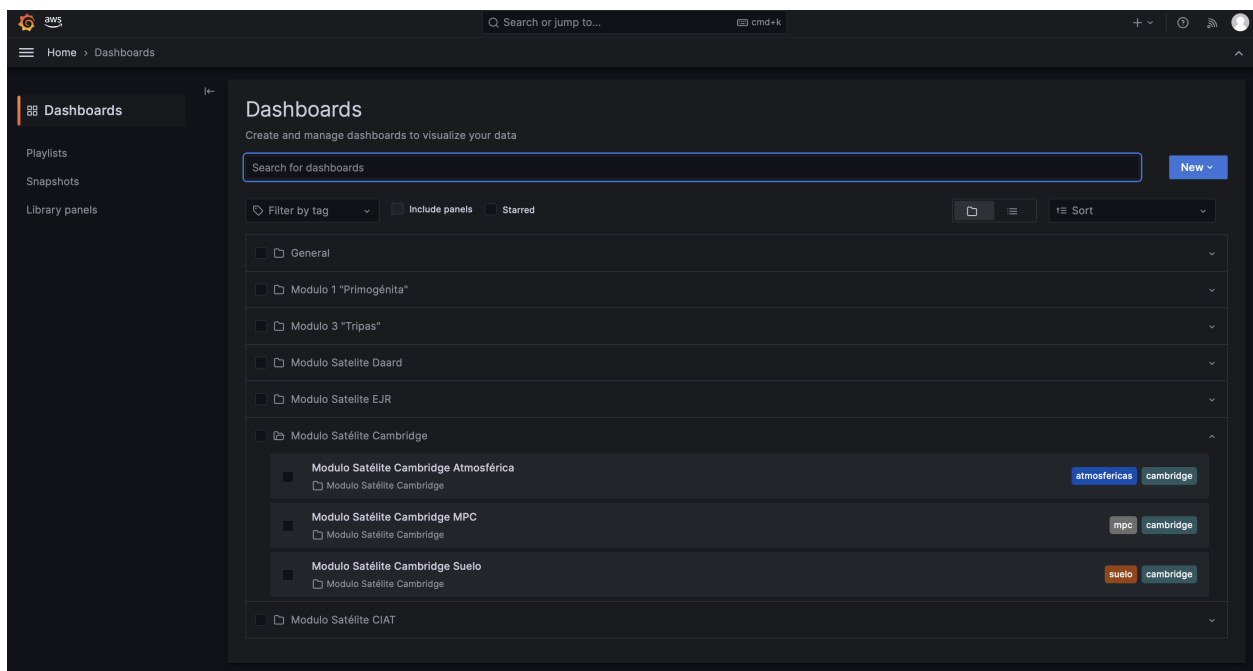


Figura 3.31: Pantalla menú Dashboards AWS Managed Grafana.

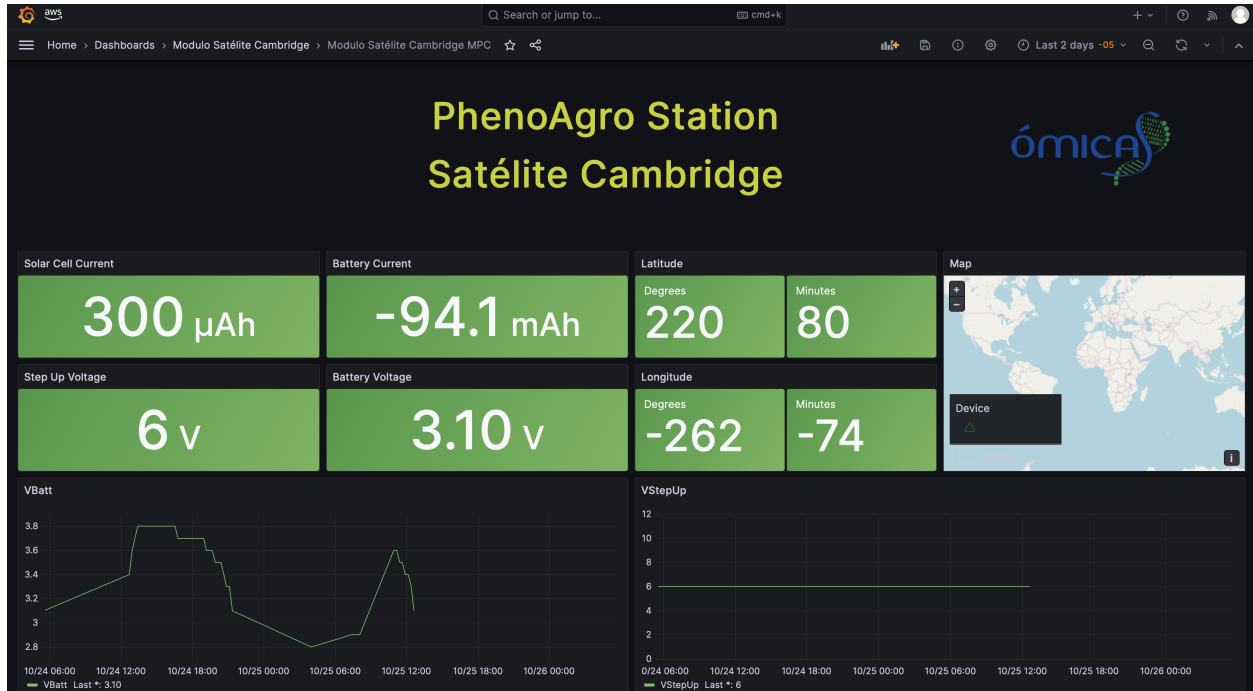


Figura 3.32: Pantalla Dashboard MPC estación Cambridge AWS Managed Grafana.

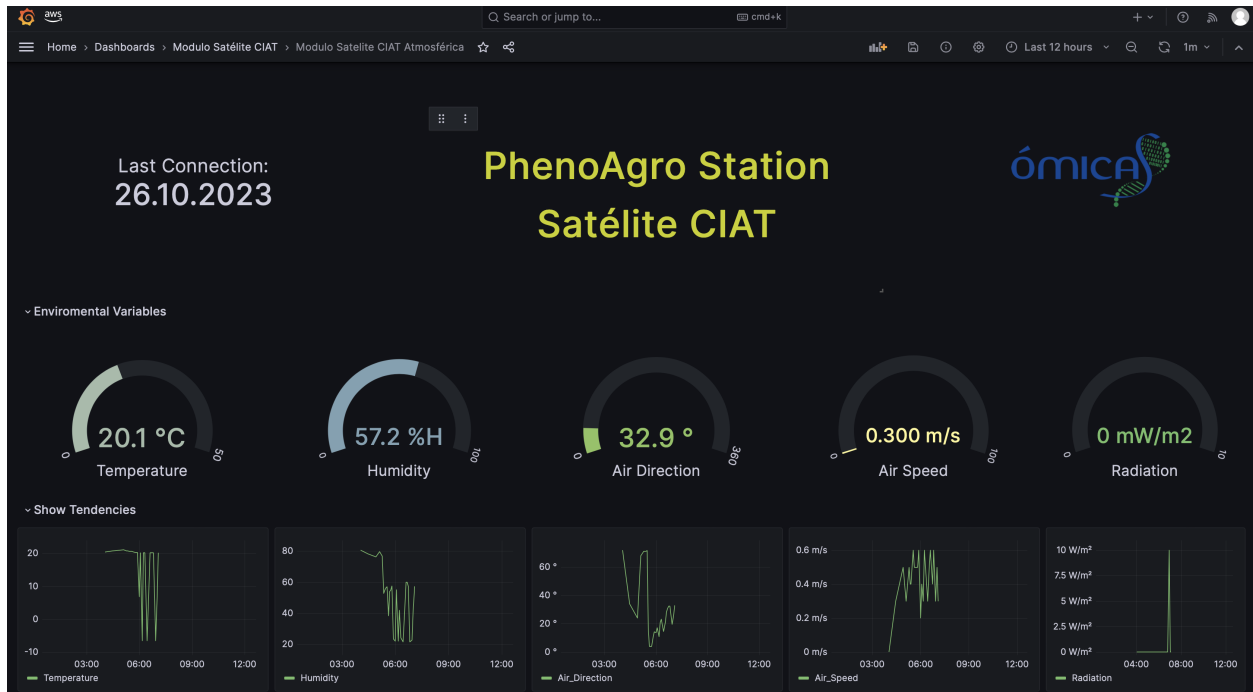


Figura 3.33: Pantalla Dashboard Atmosféricas estación CIAT AWS Managed Grafana.

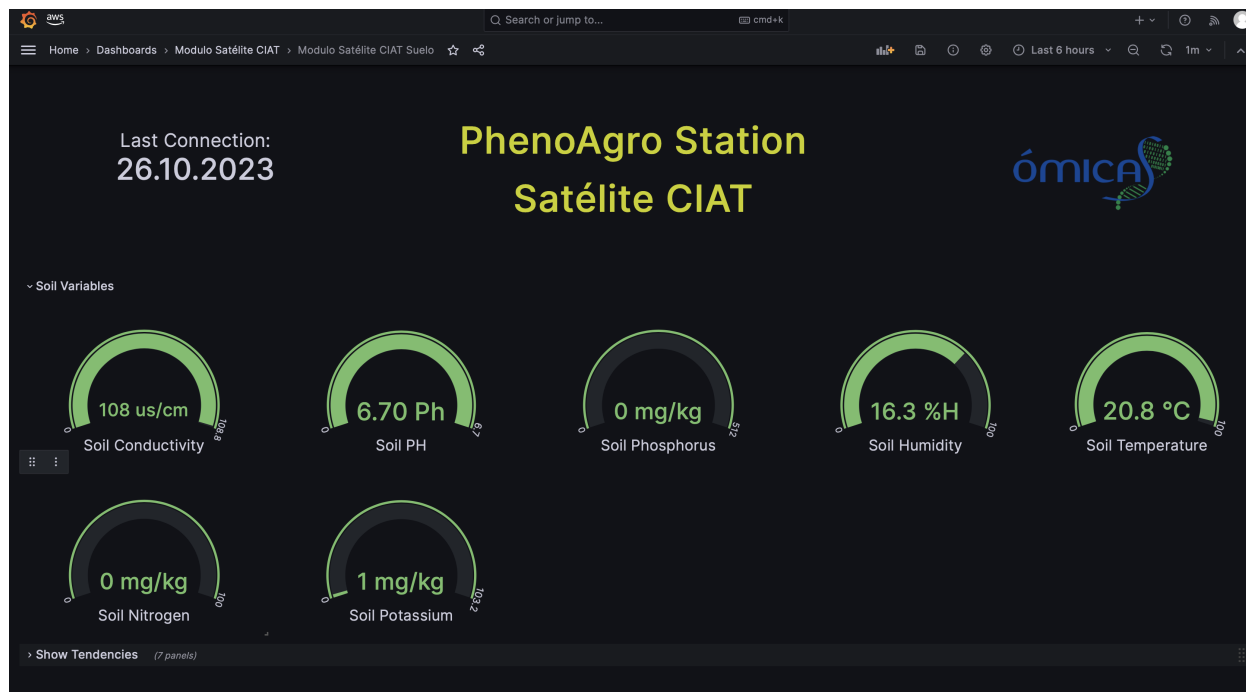


Figura 3.34: Pantalla Dashboard Suelo estación CIAT AWS Managed Grafana.

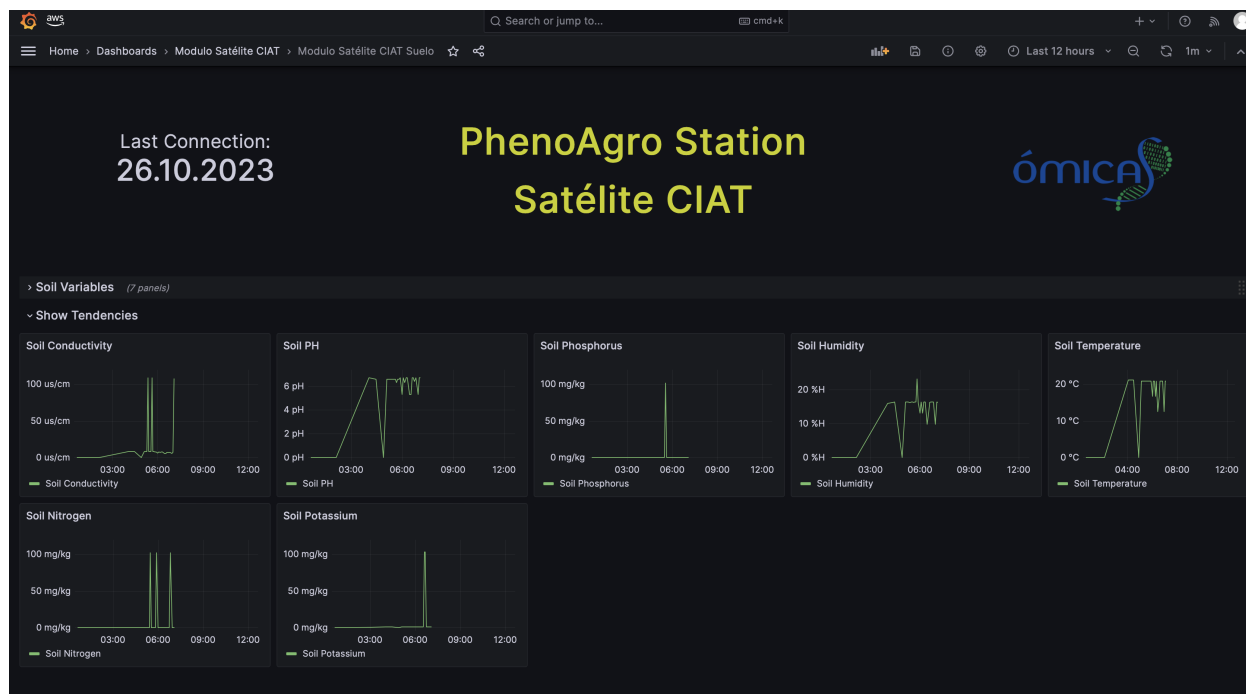


Figura 3.35: Pantalla Dashboard Suelo tendencias estación CIAT AWS Managed Grafana.

El segundo componente de la capa de consumo es el servicio de AWS Athena, este servicio permite consultar la información almacenada en los Buckets S3 con sentencias SQL de manera sencilla y eficiente. En la figura 3.36 se evidencia como se cuenta con un editor desde donde se ejecutan las sentencias SQL. Una vez hecha la consulta, el servicio cuenta con un visualizador de los resultados figura 3.37 y también cuenta con la habilidad de exportar los resultados en formato CSV figura 3.38. Por ultimo, con AWS Athena se puede visualizar la estructura de la información almacenada en el Bucket S3 a manera de tabla figura 3.39.

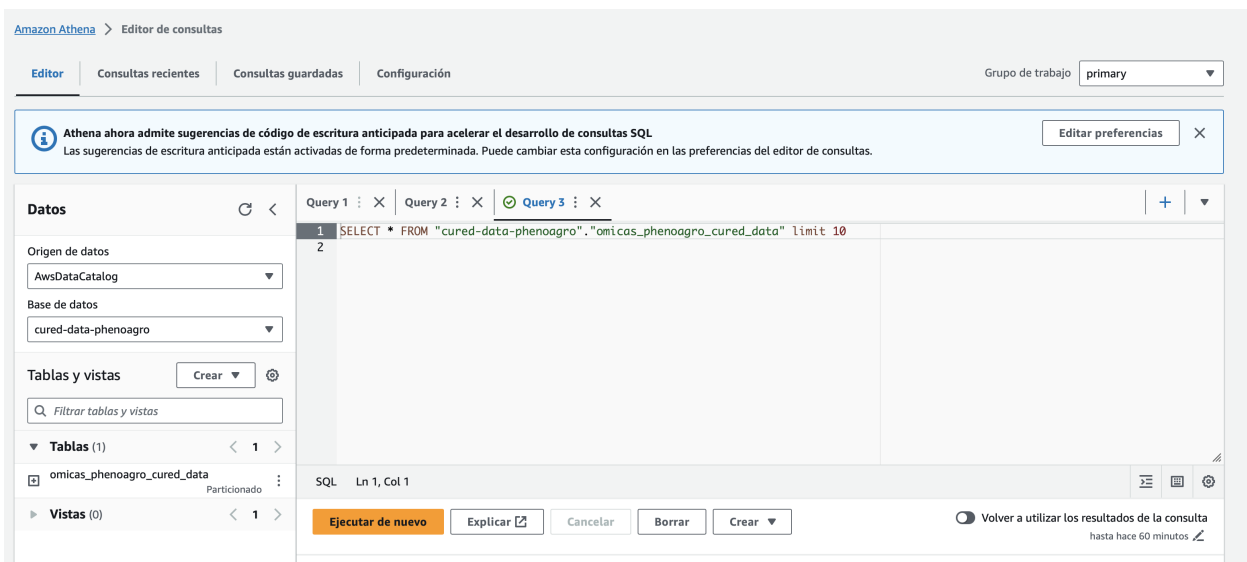


Figura 3.36: Pantalla editor sentencias SQL AWS Athena consola AWS.

Ejecutar de nuevo
Explicar
Cancelar
Borrar
Crear ▼

🔴 Volver a utilizar los resultados de la consulta
hasta hace 60 minutos

Resultados de la consulta
Estado de la consulta

✔ Completado
Tiempo en cola: 126 ms
Tiempo de ejecución: 1.108 seg
Datos analizados: 6.70 KB

Resultados (10)

Copiar

Descargar resultados

🔍

< 1 >

# ▼	modul e ▼	location ▼	deviceid ▼	measurename ▼	measurevalue ▼	time ▼	year
1	Potencia	Omicas	5...	CBatt	-202.4	2023-08-29 15:41:50.398	2023
2	Potencia	Omicas	...	CCell	154.1	2023-08-29 15:41:50.398	2023
3	Potencia	Omicas	...	CBatt	-202.3	2023-08-29 15:41:50.398	2023
4	Potencia	Omicas	...	CCell	154.2	2023-08-29 15:41:50.398	2023
5	Potencia	Omicas	...	CBatt	-202.1	2023-08-29 15:41:50.398	2023
6	Potencia	Omicas	...	CCell	154.9	2023-08-29 15:41:50.398	2023
7	Potencia	Omicas	...	CBatt	-202.1	2023-08-29 15:41:50.398	2023
8	Potencia	Omicas	...	CCell	154.9	2023-08-29 15:41:50.398	2023
9	Potencia	Omicas	...	VBatt	3.6	2023-10-11 04:59:30.273	2023
10	Potencia	Omicas	...	VStepUp	6.0	2023-10-11 04:59:30.273	2023

Figura 3.37: Pantalla resultados sentencias SQL AWS Athena consola AWS.

Con la implementación de estos componentes de la capa de ingesta de datos de la arquitectura se cumple con los requisitos funcionales RF-06, RF-08. Además, se cumple con los requisitos no funcionales RNF-03, RNF-04 , RNF-05 y con los requisitos de interfaz RI-01 y RI-02.

3.3.5. Capa de Metadatos y Catalogación

En la capa de Metadatos y Catalogación se implementaron dos componentes, AWS Crawler el cual se encarga de descubrir la información guardada en el Bucket S3 para así guardar la metadata en una tabla en AWS Data Catalog para poder ser consultada después en AWS Athena. En la figura 3.40 se evidencia las propiedades del Crawler creado y en la parte inferior el historiar de las ejecuciones del Crawler. En la figura 3.41 se evidencia la información resultante de la ejecución del Crawler, muestra las particiones añadidas de los nuevos datos encontrados en el Bucket S3 y la tabla en la que actualiza esta información.

The screenshot displays the AWS Glue console for the crawler 'cr-phenoAgro-data'. The top section shows the crawler's properties, including its name, description, IAM role (AWSGlueServiceRole-phenoAgro), database (cured-data-phenoagro), and state (READY). Below this, the 'Crawler runs' section shows a table of five completed runs with their respective start and end times, durations, and table changes.

Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
October 23, 2023 at 21:13:24	October 23, 2023 at 21:14:20	55 s	Completed	0.077	1 table change, 42 partition changes
September 11, 2023 at 21:21:03	September 11, 2023 at 21:22:02	58 s	Completed	0.041	1 table change, 6 partition changes
September 5, 2023 at 21:33:02	September 5, 2023 at 21:33:45	43 s	Completed	0.096	1 table change, 7 partition changes
August 29, 2023 at 15:45:21	August 29, 2023 at 15:46:06	45 s	Completed	0.041	1 table change, 1 partition change
August 28, 2023 at 18:58:10	August 28, 2023 at 18:59:02	51 s	Completed	0.082	1 table change, 1 partition change

Figura 3.40: Pantalla configuración AWS Crawler consola AWS.

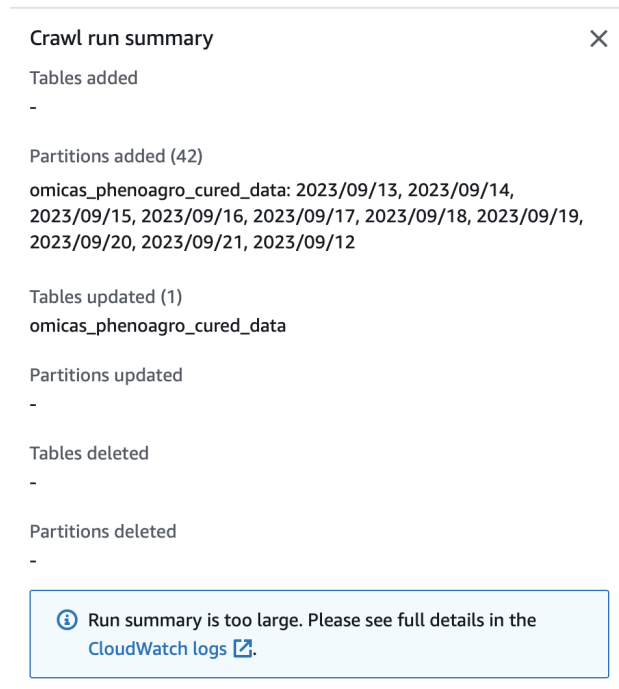


Figura 3.41: Pantalla información de ejecución AWS Crawler consola AWS.

El segundo componente de esta capa es el AWS Data Catalog, este servicio es el encargado de almacenar lo metadatos obtenidos previamente por el Crawler sobre los datos almacenados en el Bucket S3. En este servicio se crea una base de datos y una tabla desde donde se va a gestionar toda la metadata. En la figura 3.42 se evidencia la configuración de la tabla en la cual se ha especificado el nombre, la ubicación de los datos que en este caso es un Bucket S3 y la clasificación y formatos de los archivos que es Parquet. Posteriormente en la figura 3.43 se evidencia el esquema de la tabla el cual contiene todos los campos con los que cuentan los archivos con la información, este esquema se identifica automáticamente una vez se ejecuta el servicio de AWS Crawler por primera vez.

The screenshot shows the AWS Data Catalog console interface for the table 'omicas_phenoagro_cured_data'. The page includes a breadcrumb trail 'AWS Glue > Tables > omicas_phenoagro_cured_data', the table name, and a 'Last updated (UTC)' timestamp of 'October 23, 2023 at 21:14:18'. There are buttons for 'Version 5 (Current version)' and 'Actions'. Below this, there are tabs for 'Table overview' and 'Data quality New'. The main content area is divided into 'Table details' and 'Advanced properties'. The 'Table details' section contains a table with the following information:

Name	omicas_phenoagro_cured_data	Description	-	Database	cured-data-phenoagro	Classification	Parquet
Location	s3://omicas-phenoagro-cured-data/	Connection	-	Deprecated	-	Last updated	October 23, 2023 at 21:14:18
Input format	org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputFormat	Output format	org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutputFormat	Serde serialization lib	org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe		

Figura 3.42: Pantalla configuración tabla AWS Data Catalog consola AWS.

The screenshot shows the AWS Data Catalog console interface for the schema of the table 'omicas_phenoagro_cured_data'. The page includes tabs for 'Schema', 'Partitions', and 'Indexes'. The main content area is titled 'Schema (9)' and includes a search bar for 'Filter schemas'. Below this is a table with the following columns: '#', 'Column name', 'Data type', 'Partition key', and 'Comment'. The table contains 9 rows of schema information:

#	Column name	Data type	Partition key	Comment
1	module	string	-	-
2	location	string	-	-
3	deviceid	string	-	-
4	measurename	string	-	-
5	measurevalue	double	-	-
6	time	timestamp	-	-
7	year	string	Partition (0)	-
8	month	string	Partition (1)	-
9	day	string	Partition (2)	-

Figura 3.43: Pantalla esquema tabla AWS Data Catalog consola AWS.

Con la implementación de estos componentes de la capa de ingesta de datos de la arquitectura se cumple con los requisitos funcionales RF-06, RF-07, RF-08. Además, se cumple con los requisitos no funcionales RNF-03, RNF-04, RNF-05 y con los requisitos de interfaz RI-01.

3.3.6. Validación

La validación de la implementación anteriormente presentada se realizó en conjunto con las dos personas que trabajaban al momento de la realización del proyecto en PhenoAgro. Con ellos lo que se validó es que los dispositivos se pudieran conectar de manera satisfactoria a la plataforma implementada en AWS, una vez conectados se validó que enviaran la información que estaban recolectando de manera correcta, esto se hizo desde la consola de ellos local del dispositivo y por el otro lado desde la consola de AWS IoT Core. Una vez los datos eran recibidos se necesitaban visualizar de una manera intuitiva en los dashboards de Grafana lo cual se validó entrevistando a

los miembros de PhenoAgro lo cuales manifestaron satisfacción con lo entregado. Debido a que el proyecto PhenoAgro con contaba con ninguna persona con conocimientos en arquitectura de software la propuesta, el diseño y la elección de herramientas no fue posible validarlo con ellos y fue validado con los Directores del proyecto (Arquitectura para dispositivos IoT del instituto ÓMICAS) a manera de juicio de experto debido a que ellos son personas con la suficiente trayectoria y conocimientos en el área.

3.4. Conclusiones

- El potencial de las tecnologías IoT en el ámbito de la agricultura es inmenso y aún no ha sido completamente aprovechado, lo que abre una ventana de oportunidad para lograr beneficios significativos tanto para los cultivos como para los agricultores. La interconexión de dispositivos y sensores a través de la Internet de las Cosas (IoT) permite una monitorización en tiempo real de las condiciones agrícolas, como la temperatura, humedad, nivel de agua y calidad del suelo. Esto posibilita una gestión más precisa y eficiente de los cultivos, al proporcionar datos en tiempo real que permiten tomar decisiones informadas.
- El Instituto ÓMICAS dispone de un destacado departamento de hardware que ha desarrollado tecnologías avanzadas. Sin embargo, se enfrenta a un desafío crítico: la falta de un desarrollo de software que pueda aprovechar plenamente el potencial de su hardware y la información que recopila. Esta limitación actúa como una barrera para maximizar el alcance y el impacto de sus actividades. El desarrollo de software específico es esencial para la gestión, análisis y aprovechamiento de los datos recopilados por el hardware de ÓMICAS.
- Desarrollar herramientas de software para tecnologías IoT puede, de hecho, ser un proceso desafiante y costoso. En este contexto, contar con proveedores en la nube que ya han realizado este desarrollo puede ser una estrategia muy beneficiosa. Esto permite avanzar en proyectos como PhenoAgro de manera más ágil y eficiente, al tiempo que reduce la necesidad de emplear un gran número de recursos y personal.
- Mantener un contacto directo y colaborativo con el equipo que desarrolla el hardware es fundamental para garantizar la alineación y el éxito en la implementación de las soluciones de software en un proyecto IoT. La comunicación estrecha entre los equipos de hardware y software es esencial para adaptarse a cambios durante las etapas de desarrollo y pruebas y evitar retrasos significativos.
- La selección de servicios de proveedores de nube como AWS puede ser una tarea desafiante debido a la amplia gama de servicios que ofrecen. Contar con la asesoría de personas con experiencia en la plataforma es esencial para tomar decisiones informadas y optimizar el uso de los recursos en la nube. Estas personas pueden ofrecer orientación valiosa en aspectos como la selección de servicios adecuados, la arquitectura de infraestructura, la optimización de costos, seguridad, cumplimiento, optimización de rendimiento, migración y despliegue.
- La etapa de diseño de la arquitectura es de suma importancia en nuestro proyecto, ya que una vez que se ha definido un diseño sólido, la implementación se vuelve más ágil y efectiva. Un diseño bien elaborado sienta las bases para una ejecución sin contratiempos, y su meticulosa planificación contribuye significativamente a la agilidad del proceso. Cada detalle considerado en esta fase tiene un impacto directo en la eficiencia y calidad del proyecto en su totalidad, subrayando la relevancia crítica de este paso fundamental.

- En el contexto de nuestro proyecto, resulta evidente que cada capa desempeña un rol fundamental en el proceso, comenzando con la adquisición inicial de datos y culminando en su posterior análisis y aprovechamiento. La meticulosa integración de componentes especializados y el empleo de herramientas avanzadas no solo contribuyen a una gestión de datos integral, sino que también reflejan un enfoque técnico que optimiza la eficiencia y la efectividad en el entorno IoT. Esta sinergia entre las diferentes capas se convierte en el cimiento sobre el cual se sustenta el éxito de nuestro proyecto, permitiendo la gestión, interpretación y aplicación de datos de manera más ágil y precisa.

3.5. Recomendaciones y lecciones aprendidas.

- Estar comunicados con el equipo de hardware es necesario ya que los tiempos de los equipos pueden ser diferentes y es necesario el hardware para probar las soluciones de software implementadas.
- Debe existir un plan para poder probar los componentes de la arquitectura sin necesidad de tener que depender de otros equipos para poder avanzar de manera ágil en el proyecto.
- Existen una gran cantidad de servicios IoT que los proveedores en la nube ofrecen por lo cual tener bien delimitado el proyecto ayuda a saber cuáles seleccionar y hasta qué parte llegar.

3.6. Trabajos futuros.

Como parte de la planificación para futuros proyectos, el Instituto ÓMICAS está involucrado en diversas iniciativas y áreas donde existe la oportunidad de diseñar, ampliar e implementar la arquitectura presentada en este proyecto. Entre los desafíos actuales se encuentra la integración de los diversos repositorios de datos generados diariamente por el instituto en distintos proyectos, con el fin de facilitar su análisis y consulta.

Continuando con la posibilidad de continuar con el trabajo en el instituto ÓMICAS y el proyecto PhenoAgro se plantean unas recomendaciones y acciones de mejora las cuales puede impulsar el desarrollo: Contando con un mayor tiempo para el diseño y elección de herramienta se puede revisar con mayor rigurosidad otras herramientas para la visualización de los datos y construcción de Dashboards ya que en este documento le implemento Grafana, sin embargo, existen otras plataforma que pueden servir de igual manera y mejorar los costos. Además, se puede avanzar en la estandarización de los tiempos de recepción de la información debido a que es un tema importante para tener en cuenta al momento de seleccionar herramientas y su configuración ya que en el tiempo del desarrollo del proyecto actual no se tuvo en cuenta este tema por lo que el proyecto PhenoAgro estaba en fase de desarrollo y no estaba claro.

Es importante tener en cuenta que para desarrollos futuros en instituto no cuenta con ingenieros de software por lo cual la interacción y el conocimiento en común es restringido, entonces se recomienda siempre tener personas con el conocimiento del tema y una buena cantidad de interacción con el instituto ÓMICAS para lograr la mayor calidad de los productos.

Como otro trabajo futuro importante se pueden generar escenarios de calidad del proyecto los cuales permitan evaluar y validar de mejor manera la arquitectura y su implementación pudiendo dar más valor a lo desarrollado.

Por otra parte, como oportunidad de trabajos futuros es importante revisar la arquitectura de referencia que tiene AWS en su repositorio de arquitecturas para una granja inteligente con componentes de AWS ver figura 3.44. En esta arquitectura se puede resaltar que como trabajo futuro para PhenoAgro se puede implementar unos servicios de AWS IoT Device Defender y Device Manager para centralizar completamente la gestión de los dispositivos IoT desde AWS y tener una capa de seguridad robusta. Por otra parte, se pueden implementar componentes como Amazon Kinesis Video Stream el cual permitiría en un futuro transmisión de vídeo en tiempo real desde las estaciones de PhenoAgro para visualizar la misma estación y los cultivos. Por ultimo se deja planteado que es posible integrar otros componentes relacionados a Data Analytics como lo es Amazon Kinesis Data Analytics para poder aprovechar mejor los datos recolectados y almacenados. Como conclusión sobre los trabajos futuros al haber desarrollado e implementado la arquitectura con servicios de AWS permite complementar todo el ecosistema de PhenoAgro con otros productos que AWS ofrece que aumentarían el valor del proyecto para los usuarios finales.

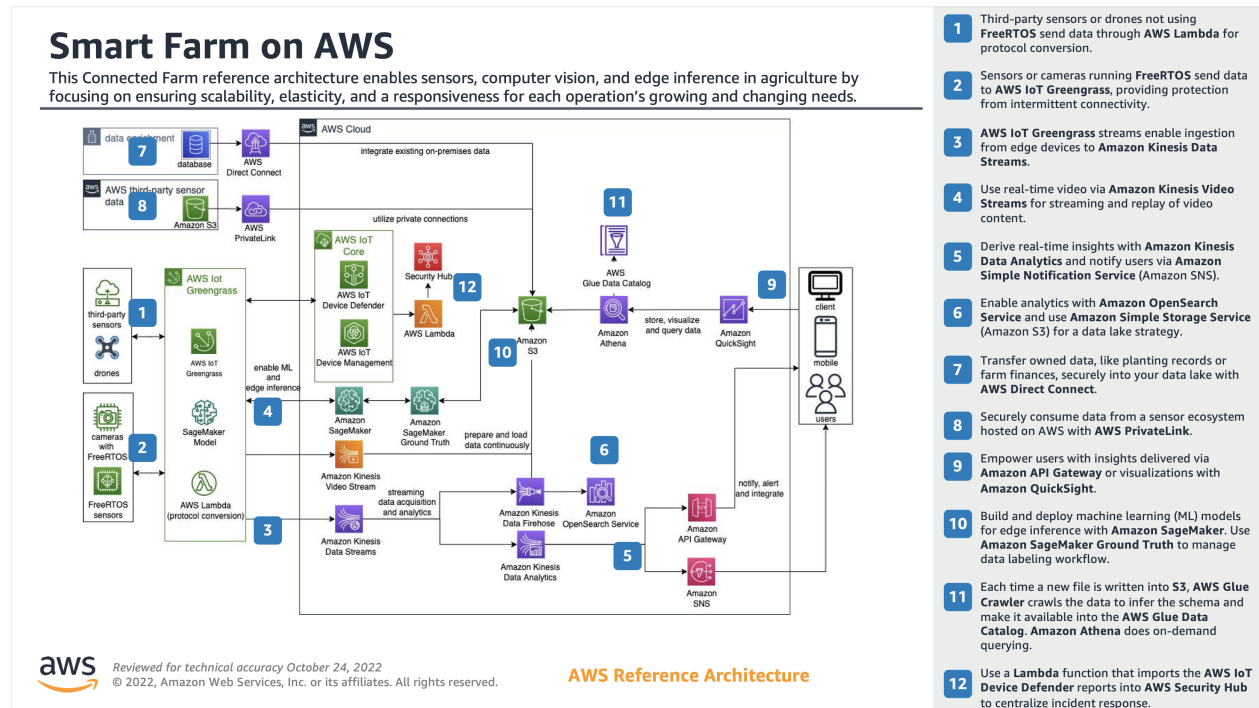


Figura 3.44: Arquitectura de referencia Smart Farm on AWS Services (2022)

Finalmente para facilitar trabajos futuros sobre la arquitectura presentada en este documento se comparte el link al repositorio con el código de su implementación: <https://github.com/cdtello/omicas.git>

Bibliografía

- Araújo, S. O., Peres, R. S., Barata, J., Lidon, F., and Ramalho, J. C. (2021). Characterising the agriculture 4.0 landscape—emerging trends, challenges and opportunities. *Agronomy*, 11(4).
- Argentina, G. (2022). Producción sostenible.
- Dustdar, S., Fiadeiro, J. L., and Sheth, A. (2008). *Business process management*, volume 64.
- Griffin, S., Kunz, A. E., and Manager, G. P. (2009). Data fusion/integration of high and medium resolution imagery for crop analysis.
- Kaur, B. and Owusu, R. K. (2016). Inverse problems and data fusion for crop production applications targeting optimal growth-fertilization. volume 2016-February, pages 108–114. Institute of Electrical and Electronics Engineers Inc.
- López, I. D. and Corrales, J. C. (2018). A smart farming approach in automatic detection of favorable conditions for planting and crop production in the upper basin of cauca river. In Angelov, P., Iglesias, J. A., and Corrales, J. C., editors, *Advances in Information and Communication Technologies for Adapting Agriculture to Climate Change*, pages 223–233, Cham. Springer International Publishing.
- Nations, U. (2015). Objetivos de desarrollo sostenible.
- Nations, U. (2022). Desafíos globales población.
- Services, A. W. (2022). Smart farm on aws.
- ÓMICAS, I. (2021). ¿qué es Ómicas?