



“Año del Bicentenario, de la consolidación de nuestra Independencia, y de la conmemoración de las heroicas batallas de Junín y Ayacucho”

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



“Sistema de Automatización de Carga de Datos y Modificación de Reportes en el Almacén de Datos”

Integrantes:

Nº	Código Universitario	Apellidos y Nombres
1.-	2017057528	Ccalli Chata, Joel Robert
2.-	2020067145	Anahua Coaquira, Mayner Gonzalo
3.-	2020066924	Zeballos Purca, Justin Zinedine
4.-	2020069046	Salinas Condori, Erick Javier
5.-	2018000654	Paco Ramos, Aaron Pedro

CURSO : “Inteligencia de Negocios”

SECCIÓN : “A”

DOCENTE : Ing. “Patrick Jose Cuadros Quiroga”

Tacna - Perú

2024



CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	Joel Ccalli Erick Salinas Aaron Paco Justin Zeballos	Mayner Anahua	Mayner Anahua	27/06/2024	Versión Original



***Sistema de Automatización de Carga de Datos y
Modificación de Reportes en el Almacén de Datos***
Informe de SAD
FD-04
Versión 1.0



Contenido

1.

¡Error! Marcador no definido.1.1.

¡Error! Marcador no definido.1.2.

¡Error! Marcador no definido.1.3.

¡Error! Marcador no definido.1.4.

¡Error! Marcador no definido.2.

¡Error! Marcador no definido.2.1.1.

¡Error! Marcador no definido.2.1.2.

¡Error! Marcador no definido.3.

¡Error! Marcador no definido.3.1.

¡Error! Marcador no definido.3.1.1.

¡Error! Marcador no definido.3.2.

¡Error! Marcador no definido.3.2.1.

¡Error! Marcador no definido.3.2.2.

¡Error! Marcador no definido.3.2.3.

¡Error! Marcador no definido.3.2.4.

¡Error! Marcador no definido.3.2.5.

¡Error! Marcador no definido.3.2.6.

¡Error! Marcador no definido.3.3.

¡Error! Marcador no definido.3.3.1.

¡Error! Marcador no definido.3.3.2.

¡Error! Marcador no definido.3.4.

¡Error! Marcador no definido.3.4.1.

¡Error! Marcador no definido.3.5.

¡Error! Marcador no definido.3.5.1.

¡Error! Marcador no definido.4.

¡Error! Marcador no definido.Escenario de Funcionalidad

8

Escenario de Usabilidad

8

Escenario de confiabilidad

9

Escenario de rendimiento

9

Escenario de mantenibilidad

9

Otros Escenarios

9

1. INTRODUCCIÓN:



1.1 Propósito (4+1)

El propósito de este proyecto es desarrollar un sistema automatizado que facilite la carga, almacenamiento y modificación de datos en archivos, así como la generación y actualización de reportes en un almacén de datos. Al utilizar S3 para el almacenamiento y IAM para la gestión de permisos, buscamos garantizar la seguridad, integridad y disponibilidad de los datos, permitiendo a los usuarios acceder a la información de manera eficiente y con los niveles de permiso adecuados. Este sistema no solo reducirá la carga de trabajo manual y los errores asociados, sino que también mejorará la precisión y la relevancia de los reportes generados.

1.2 Alcance

El alcance del proyecto incluye el desarrollo e implementación de un sistema automatizado de carga de datos, la modificación del reporte de la unidad 1 para que utilice el nuevo almacén de datos, y la capacitación a los usuarios sobre el nuevo sistema. También se realizarán pruebas y validaciones del sistema para garantizar su correcto funcionamiento, y se establecerán procedimientos de monitoreo y mantenimiento continuo. Adicionalmente, el proyecto facilitará el acceso a datos para la investigación biomédica, asegurando siempre el respeto a los derechos de los pacientes y el cumplimiento de todas las normativas y regulaciones aplicables.

1.3 Definición, siglas y Abreviaturas



Término	Definición
Actor	Usuario del sistema que puede participar De un caso de uso.
Arquitectura de software	Conjunto de elementos estáticos, propios del diseño intelectual del sistema, que definen y dan forma tanto al código fuente como al comportamiento del software en tiempo de ejecución. Naturalmente este diseño arquitectónico ha de ajustarse a las necesidades y requisitos del proyecto
Caso de Uso	Acciones que el sistema realiza, la cual proporciona un resultado de valor observable.
Diseño	Actividad creativa que proyecta objetos para después fabricarlos.
Escenario	Especifica el comportamiento y limita el interés de un área específica del sistema para 1 o varios stakeholders.
Paquetes	Agrupaciones de casos de uso y actores por funcionalidad que proveen.

1.4 Organización del documento

2. OBJETIVOS Y RESTRICCIONES ARQUITECTÓNICAS

2.1 Requerimientos Funcionales

Cuadro de requerimientos funcionales Inicial



Código	Requerimiento Funcionales Inicial
RFI 01	El sistema debe permitir la carga automática de datos desde múltiples fuentes.
RFI 02	Debe validar los datos durante el proceso de carga.
RFI 03	El sistema debe integrar los datos en el almacén de datos de manera estructurada.
RFI 04	Debe generar alertas en caso de errores durante la carga de datos.
RFI 05	El sistema debe ser capaz de programar cargas de datos periódicas.
RFI 06	Debe permitir la modificación de reportes para apuntar al nuevo almacén de datos.

2.2 Requerimientos No Funcionales – Atributos de Calidad

Código	Requerimientos no funcionales	Descripción
RNF 01	El sistema debe ser escalable para manejar volúmenes crecientes de datos.	Uso de Amazon S3 para almacenamiento escalable y Power BI para manejar grandes volúmenes de datos.
RNF 02	Debe tener una alta disponibilidad.	Uso de AWS S3 que proporciona alta disponibilidad y Power BI Service que también garantiza alta disponibilidad.
RNF 03	El tiempo de respuesta para la carga de datos no debe exceder los 10 minutos.	R Scripts optimizados para carga eficiente desde AWS S3 a Power BI.
RNF 04	El sistema debe cumplir con las normativas de seguridad de datos.	Uso de IAM Roles para control de acceso seguro y cifrado de datos en tránsito y en reposo en AWS S3.



Cuadro de requerimientos funcionales Final

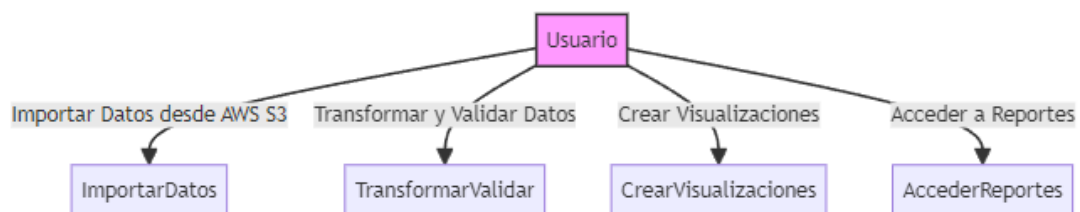


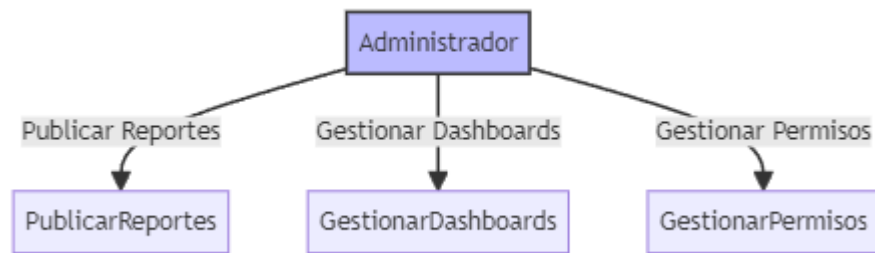
Código	Requerimiento Funcionales Finales	Descripción
RF 01	El sistema debe permitir la configuración de reglas de validación de datos personalizadas.	R Scripts permiten la validación y transformación de datos antes de cargarlos en Power BI.
RF 02	Debe soportar la carga de datos en tiempo real.	Integración con APIs y servicios de AWS para cargas en tiempo real.
RF 03	El sistema debe permitir la extracción y carga de datos mediante APIs.	Uso de R Scripts para conectarse a APIs y extraer datos hacia AWS S3.
RF 04	El sistema debe permitir la administración de permisos de usuario para la carga de datos.	Gestión de permisos mediante IAM Roles en AWS.
RF 05	El sistema debe permitir la conexión segura a través de IAM Roles para la gestión de accesos.	Implementación de IAM Roles para acceso seguro a AWS S3 desde Power BI.
RF 06	El sistema debe permitir la importación de datos desde Amazon S3 usando scripts de R.	R Scripts específicos para importar datos desde AWS S3 a Power BI.
RF 07	El sistema debe permitir la visualización y análisis de datos en Power BI Desktop y Power BI Service.	Uso de Power BI Desktop para desarrollo y Power BI Service para publicación y acceso a reportes.

3. REPRESENTACIÓN DE LA ARQUITECTURA DEL SISTEMA

3.1 Vista de caso de uso

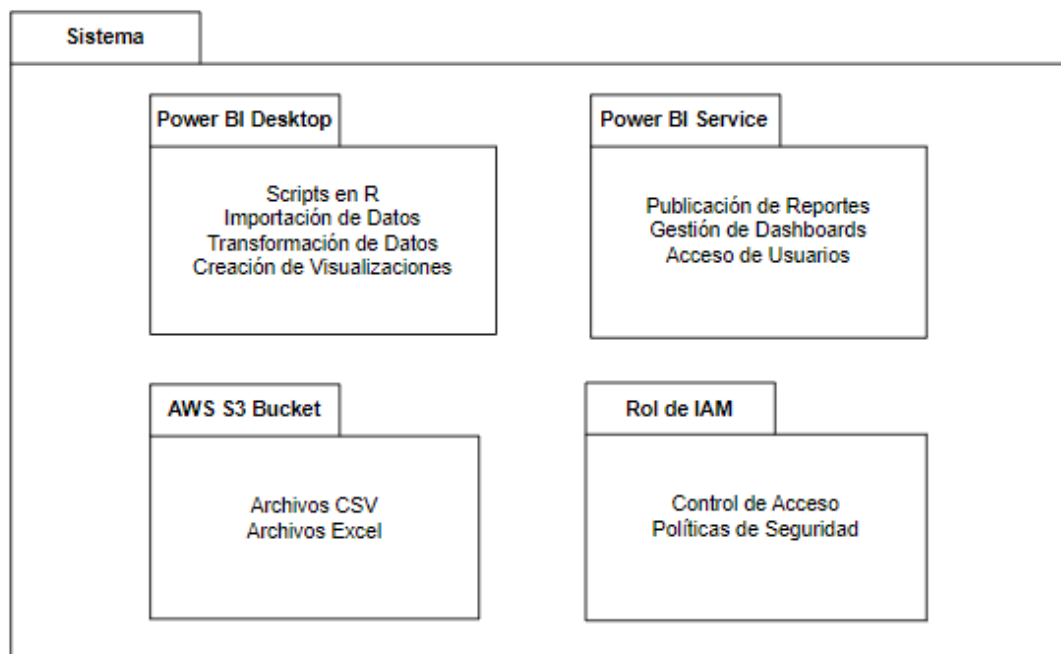
3.1.1 Diagrama de casos de uso



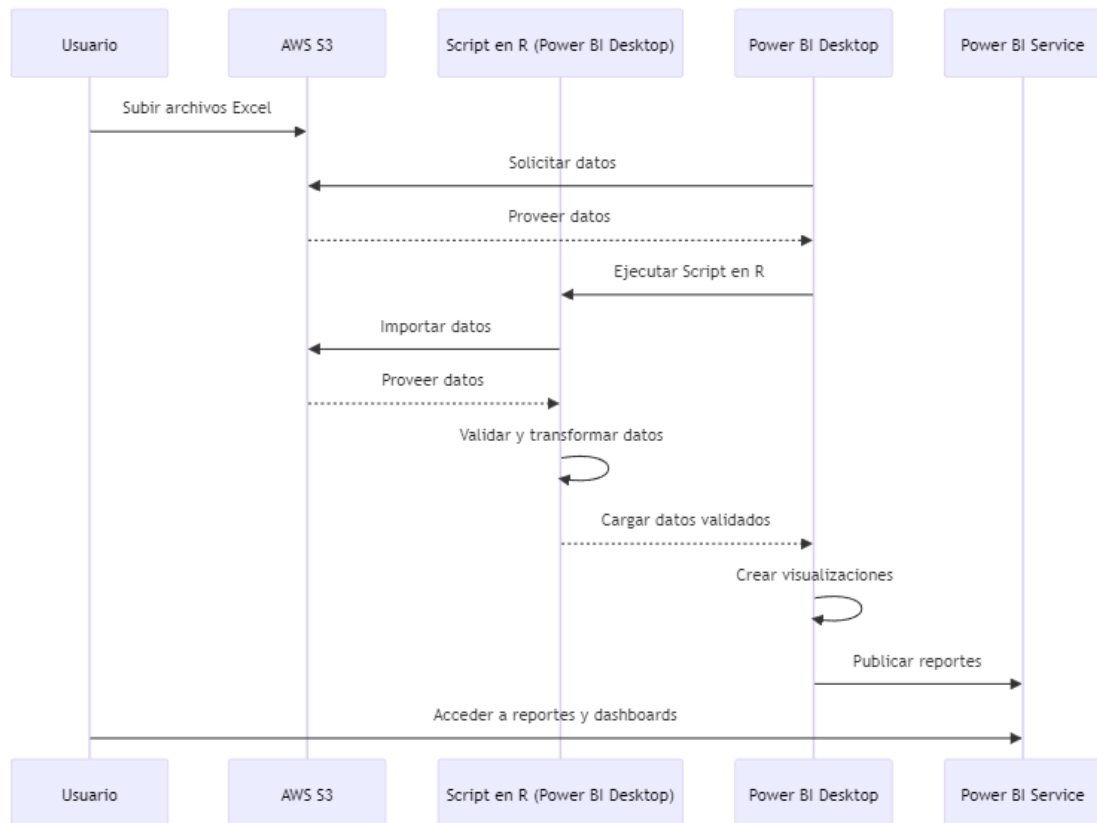


3.2 Vista Lógica

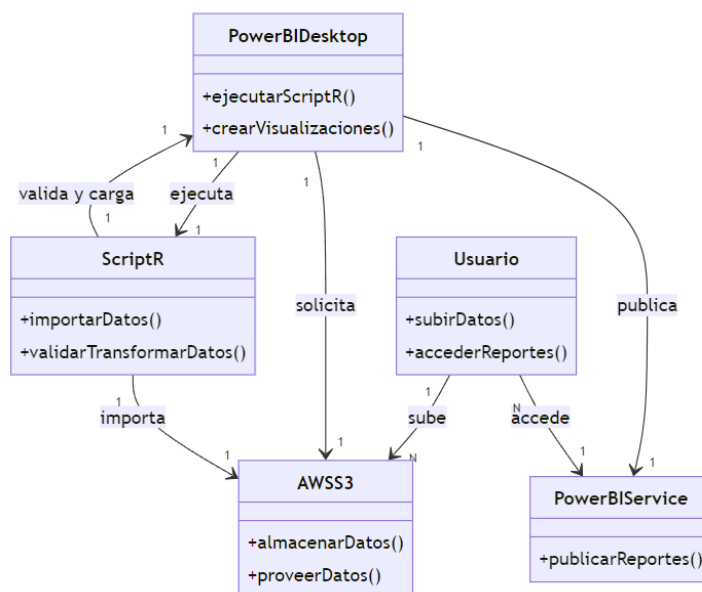
3.2.1 Diagrama de subsistemas (paquetes)



3.2.2 Diagrama de secuencia (vista de diseño)

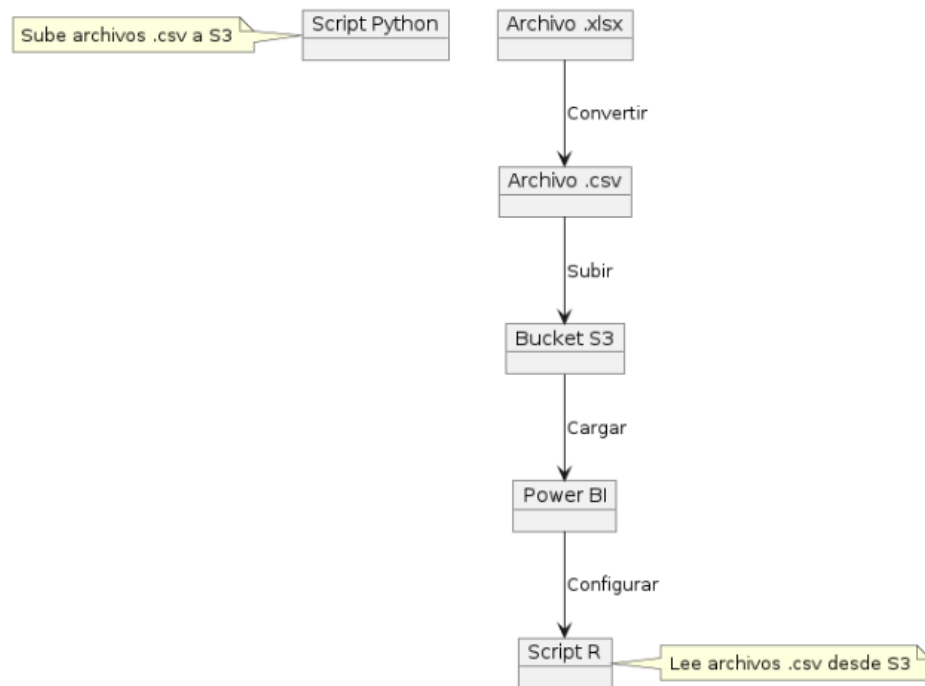


3.2.3 Diagrama de colaboración (vista de diseño)

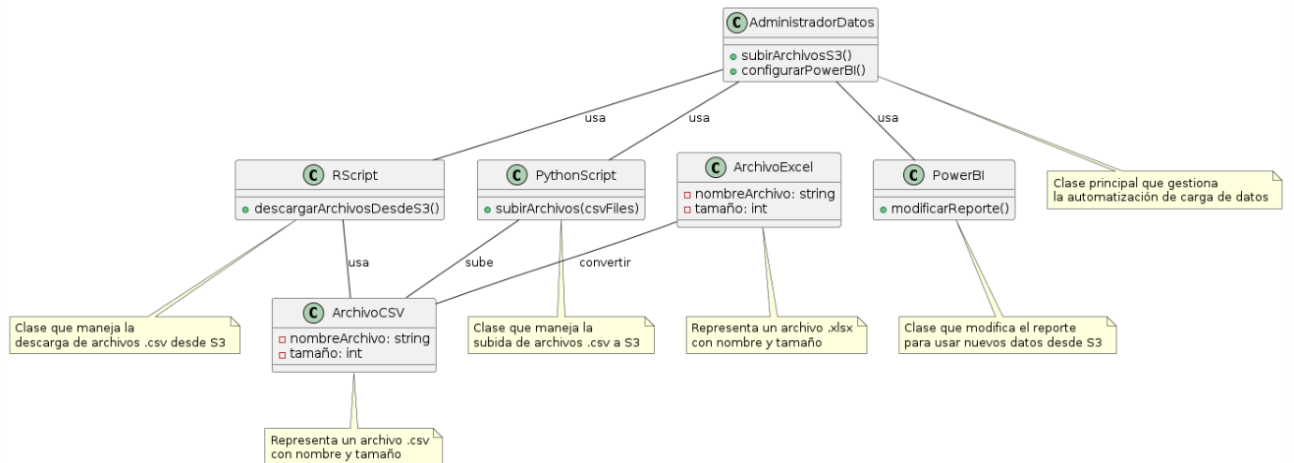


2

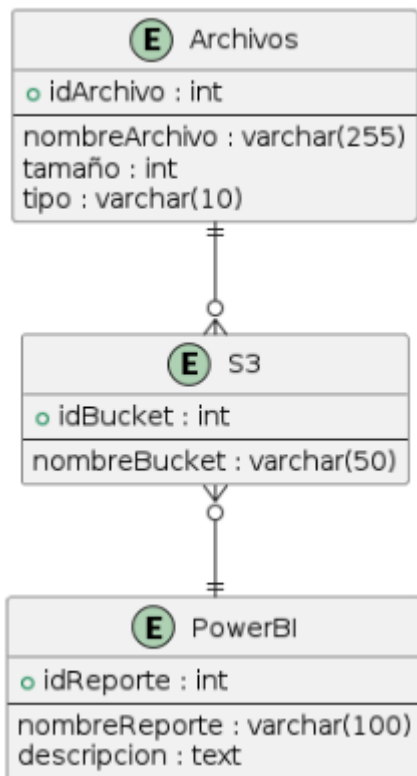
3.3.4 Diagrama de objetos



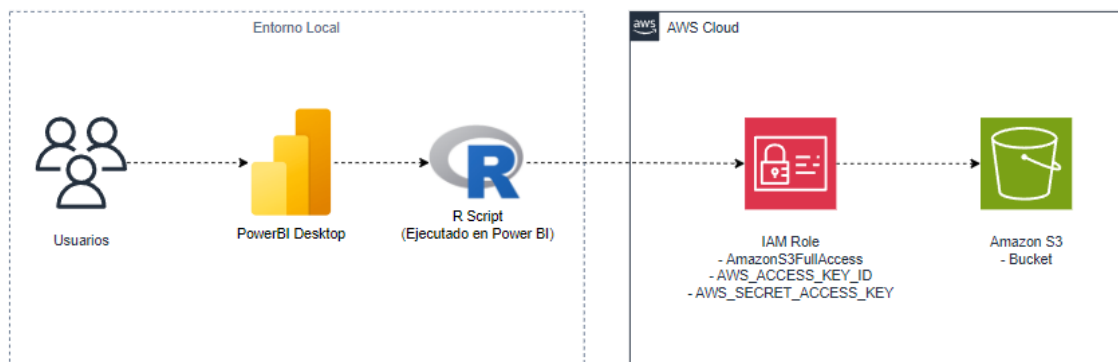
3.2.5 Diagrama de clases



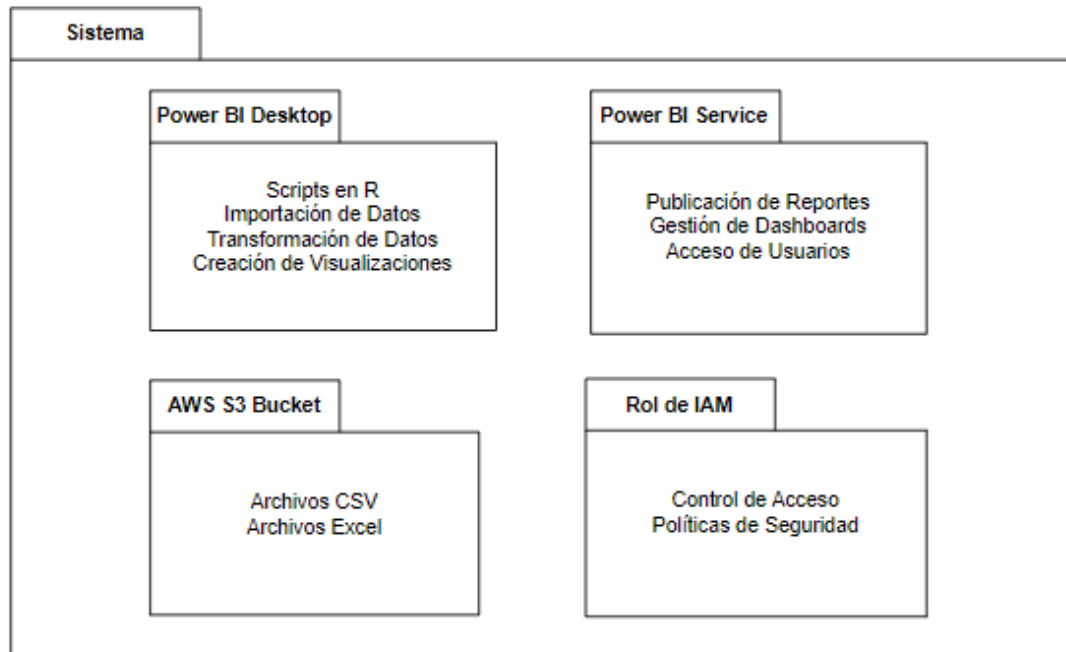
3.2.6 Diagrama de base de datos (relacional o no relacional)



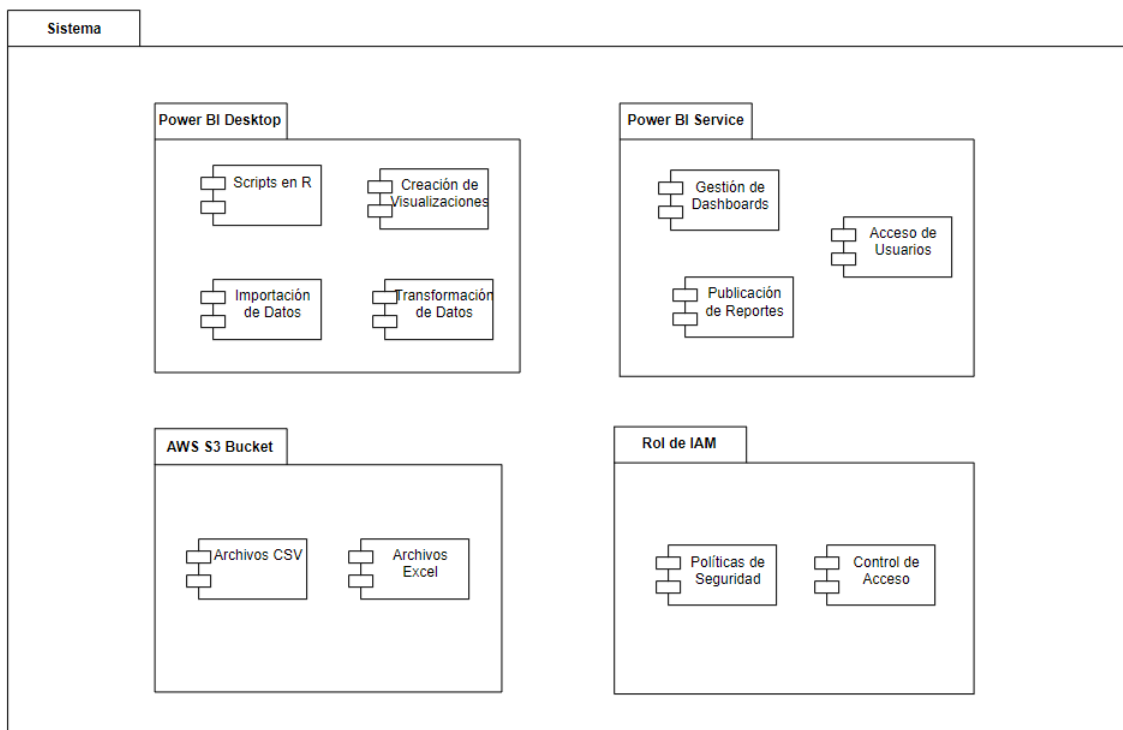
3.3 Vista de implementación



3.3.1 Diagrama de arquitectura de software (paquetes)

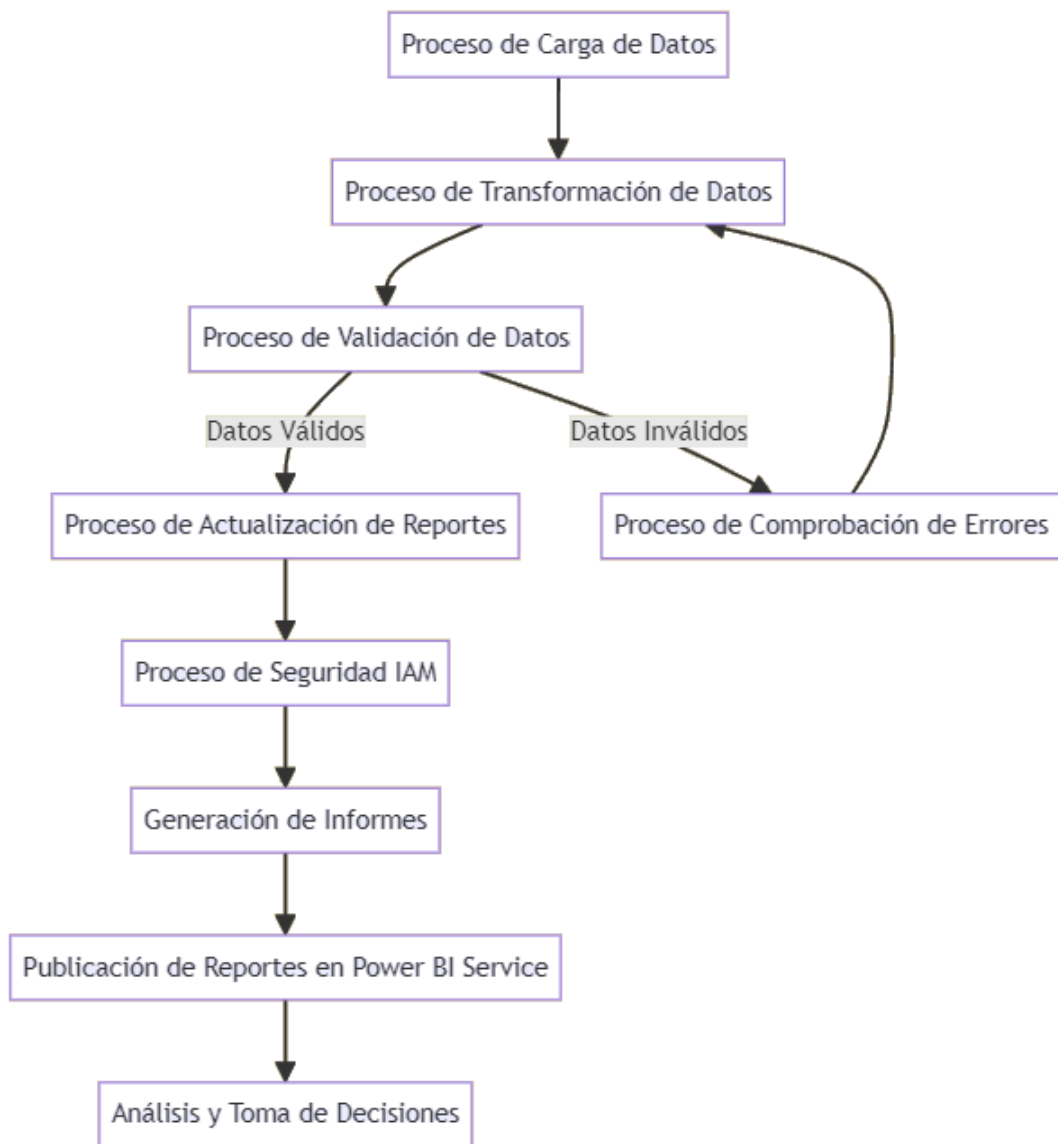


3.3.2 Diagrama de arquitectura del sistema (diagrama de componentes)



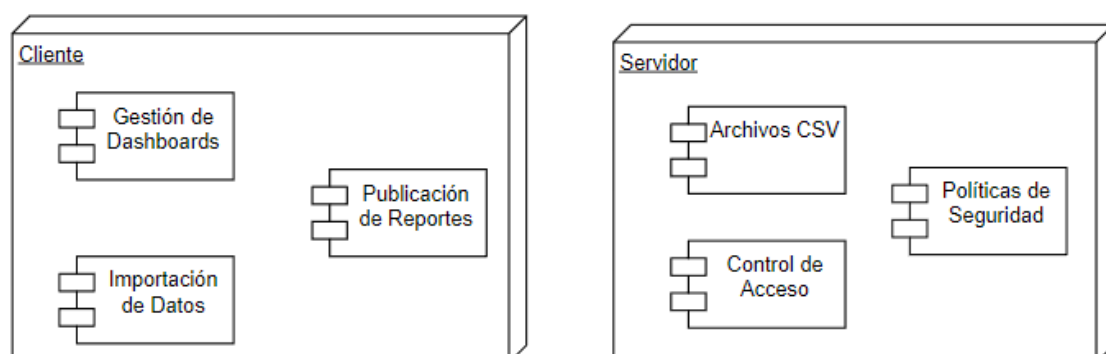
3.4 Vista de procesos

3.4.1 Diagrama de procesos del sistema (diagrama de actividad)



3.5 Vista de despliegue (vista física)

3.5.1 Diagrama de despliegue





4. ATRIBUTOS DE CALIDAD DEL SOFTWARE

4.1 Escenario de Usabilidad

Este escenario se enfoca en la interacción de los usuarios con el sistema. Dado que el sistema se utilizará para la carga de datos y generación de reportes en un entorno hospitalario, la usabilidad es crucial. Ejemplos de este escenario incluyen:

- Facilidad de uso: Los usuarios deben ser capaces de completar la carga de datos y generar reportes con mínima formación.
- Intuitividad: La interfaz debe ser intuitiva y fácil de navegar.
- Satisfacción del usuario: Los usuarios deben encontrar un sistema agradable y eficiente para realizar sus tareas.

4.2 Escenario de confiabilidad

La confiabilidad del sistema es esencial para garantizar que los datos se gestionan correctamente y sin errores. Ejemplos de este escenario incluyen:

- Disponibilidad: El sistema debe estar disponible el 99.9% del tiempo para permitir la carga de datos y generación de reportes sin interrupciones.
- Recuperación ante fallos: En caso de fallos, el sistema debe ser capaz de recuperar los datos sin pérdida.
- Precisión: Los datos cargados y los reportes generados deben ser precisos y libres de errores.

4.3 Escenario de rendimiento

El rendimiento del sistema es crítico para manejar grandes volúmenes de datos y generar reportes rápidamente. Ejemplos de este escenario incluyen:

- Tiempo de respuesta: El sistema debe cargar datos y generar reportes en menos de 10 minutos.
- Capacidad de procesamiento: El sistema debe ser capaz de manejar múltiples solicitudes de carga de datos y generación de reportes simultáneamente.
- Eficiencia: El uso de recursos del sistema (CPU, memoria) debe ser optimizado para garantizar un rendimiento óptimo.

4.4 Escenario de mantenibilidad

La mantenibilidad del sistema asegura que el software pueda ser actualizado y modificado con facilidad para adaptarse a nuevos requerimientos. Ejemplos de este escenario incluyen:

- Modularidad: El sistema debe estar diseñado de manera modular para facilitar las actualizaciones y modificaciones.



- Documentación: El sistema debe estar bien documentado para que los desarrolladores puedan entender y modificar el código fácilmente.
- Facilidad de actualización: Las actualizaciones del sistema deben poder realizarse sin interrumpir el servicio.

4.5 Otros Escenarios

Además de los escenarios mencionados, pueden existir otros atributos de calidad relevantes para el sistema. Ejemplos pueden incluir:

- Seguridad: El sistema debe proteger los datos sensibles contra accesos no autorizados mediante IAM Roles y otras medidas de seguridad.
- Compatibilidad: El sistema debe ser compatible con otros sistemas existentes en el hospital.
- Escalabilidad: El sistema debe poder escalar para manejar volúmenes crecientes de datos y usuarios en el futuro.
- Integración: Asegurar la integración fluida con las operaciones y sistemas actuales del hospital.