



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA
CARRERA DE COMPUTACIÓN

**DISEÑO Y DESARROLLO DE UN SISTEMA CLOUD-NATIVE PARA LA GESTIÓN Y
AUTOMATIZACIÓN DEL PROCESO DE AUTORIZACIÓN DEL DOCUMENTO
ELECTRÓNICO GUÍAS DE REMISIÓN DEL SRI EN EL GRUPO INDUSTRIAL GRAIMAN**

Trabajo de titulación previo a la obtención del
título de Ingeniero en Ciencias de la Computación

AUTOR: JUAN JAVIER MALO VEGA

TUTORA: ING. BERTHA KATERINE TACURI CAPELO, MSC.

Cuenca - Ecuador

2026

I. CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Juan Javier Malo Vega con documento de identificación N° 0104435540 manifiesto que:

Soy el autor y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 19 de enero del 2026

Atentamente,

Juan Javier Malo Vega

0104435540

II. CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Yo, Juan Javier Malo Vega con documento de identificación N° 0104435540, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del Proyecto técnico: “Diseño y desarrollo de un sistema Cloud-Native para la gestión y automatización del proceso de autorización del documento electrónico guías de remisión del SRI en el Grupo Industrial Graiman”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 19 de enero del 2026

Atentamente,

Juan Javier Malo Vega

0104435540

III. CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Bertha Katerine Tacuri Capelo con documento de identificación N° 0102972627, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO Y DESARROLLO DE UN SISTEMA CLOUD-NATIVE PARA LA GESTIÓN Y AUTOMATIZACIÓN DEL PROCESO DE AUTORIZACIÓN DEL DOCUMENTO ELECTRÓNICO GUÍAS DE REMISIÓN DEL SRI EN EL GRUPO INDUSTRIAL GRAIMAN, realizado por Juan Javier Malo Vega con documento de identificación N° 0104435540, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 19 de enero del 2026

Atentamente,

Ing. Bertha Katerine Tacuri Capelo, MSc.

0102972627

IV. DEDICATORIA Y AGRADECIMIENTO

A) DEDICATORIA

El presente logro académico lo ofrezco principalmente a mis padres, Andrés y Verónica, quienes han constituido la base inquebrantable de mi vida. Agradezco profundamente su respaldo absoluto y el ejemplo de tenacidad que me brindaron, pues fueron la semilla que despertó mi pasión por aprender y superarme día a día.

De igual forma, a mi pareja, cuya presencia, comprensión y aliento constante resultaron vitales para no decaer durante las extensas jornadas de desvelo y trabajo.

B) AGRADECIMIENTO

Extiendo mi profunda gratitud a la Universidad Politécnica Salesiana y a la carrera de Computación. Gracias a todo su cuerpo docente por transmitirme no solo teoría, sino vivencias que han moldeado mi perfil como ingeniero.

Reconozco especialmente la labor de mi tutora, la Ing. Bertha Tacuri, cuya guía precisa, tiempo y disposición para aconsejarme fueron determinantes para llevar este proyecto a buen puerto.

Al Grupo Industrial Graiman y a su departamento de tecnología, por brindarme la oportunidad de intervenir en un escenario real. Su confianza, apertura y la provisión de recursos han sido esenciales para madurar profesionalmente.

Por último, a mis amigos Alberto y Juan, por su lealtad incondicional y por convertir esta etapa universitaria en una experiencia mucho más grata y llevadera.

V. RESUMEN

Este proyecto de titulación se centra en la construcción de una solución de software, fundamentada en una arquitectura nativa de la nube, con el propósito de automatizar la gestión y autorización de Guías de Remisión electrónicas dentro del Grupo Industrial Graiman. El desafío principal que se busca resolver es la optimización del ciclo de emisión y validación de dichos comprobantes fiscales ante el Servicio de Rentas Internas (SRI), asegurando que el sistema permanezca disponible y sea escalable frente a una alta demanda transaccional.

En cuanto a la ejecución técnica, se utilizó una metodología apoyada en microservicios contenerizados, los cuales son orquestados mediante Kubernetes. Un componente crítico del sistema es su módulo de firma digital, que opera bajo el estándar XAdES-BES. Esto garantiza tanto la integridad como la autenticidad de los datos, en estricto apego a la normativa legal ecuatoriana. Adicionalmente, se integraron patrones de diseño específicos para administrar la concurrencia y el uso de colas de procesamiento, facilitando así una interacción fluida y resiliente ante fallos con los servicios web del ente regulador.

El producto final es una infraestructura tecnológica que logra procesar, firmar, autorizar y generar los PDF 's de las Guías de Remisión de forma totalmente desatendida. Esto ha permitido disminuir significativamente la necesidad de operación manual y ha mejorado los tiempos de respuesta del proceso. El sistema garantiza la trazabilidad completa de cada documento, desde su generación hasta su autorización final, cumpliendo con los estándares de seguridad y eficiencia exigidos por la entidad tributaria y las necesidades operativas del grupo industrial.

PALABRAS CLAVE:

Automatización de Procesos, Cloud-Native, Facturación Electrónica, Guías de Remisión, Kubernetes, Servicio de Rentas Internas (SRI), Firma Digital XAdES.

VI. ABSTRACT

This degree project addresses the design and development of a computer system based on a cloud-native architecture aimed at managing and automating the authorization process of Electronic Waybills (Guías de Remisión) for the Graiman Industrial Group. The central problem lies in the need to optimize the flow of issuance and validation of these tax documents before the Internal Revenue Service (SRI), ensuring high availability and scalability in the face of large transaction volumes.

For the solution implementation, a methodology based on the development of containerized microservices orchestrated by Kubernetes was employed. The system core integrates robust components for the digital signing of documents under the XAdES-BES standard, ensuring the integrity and authenticity of the information required by Ecuadorian regulations. Furthermore, design patterns for concurrency management and processing queues were implemented, allowing efficient and fault-tolerant communication with the SRI web services.

As a result, a technological platform capable of processing, signing, authorizing, and generating PDF files of Electronic Waybills in an unattended manner was obtained, reducing manual intervention and response times. The system ensures complete traceability of each document, from its generation to its final authorization, complying with the security and efficiency standards demanded by the tax authority and the operational needs of the industrial group.

KEYWORDS:

Process Automation, Cloud-Native, Electronic Invoicing, Electronic Waybills, Kubernetes, Internal Revenue Service (SRI), XAdES Digital Signature.

VII. ÍNDICE DE CONTENIDO

INTRODUCCIÓN.....	10
PROBLEMA.....	11
1) DESCRIPCIÓN DEL PROBLEMA	11
2) ANTECEDENTES	11
3) IMPORTANCIA Y ALCANCES	11
4) DELIMITACIÓN.....	12
OBJETIVOS GENERALES Y ESPECÍFICOS.....	13
A) OBJETIVO GENERAL	13
B) OBJETIVOS ESPECÍFICOS	13
REVISIÓN DE LA LITERATURA O FUNDAMENTOS TEÓRICOS.....	14
1) FACTURACIÓN ELECTRÓNICA.....	14
2) FIRMA DIGITAL Y ESTÁNDARES XAdES.....	14
3) ARQUITECTURAS EMPRESARIALES MODERNAS Y ENFOQUE CLOUD-NATIVE.....	14
4) CONTENERIZACIÓN Y ORQUESTACIÓN CON KUBERNETES	15
5) CONCURRENCIA, COLAS Y PATRONES DE CONFIABILIDAD	15
6) OBSERVABILIDAD Y PORTAL WEB DE TRAZABILIDAD.....	15
7) SEGURIDAD Y CUMPLIMIENTO	16
MARCO METODOLÓGICO.....	17
A) EL NÚCLEO DE AUTOMATIZACIÓN	17
i) Enfoque de Desarrollo y Arquitectura.....	17
ii) Implementación del Flujo de Procesamiento.....	17
iii) El ciclo de vida del documento transita por:	17
iv) Paradigmas y Patrones de Diseño Aplicados.....	18
v) Gestión de Datos y Persistencia.....	18
vi) Estrategia de Seguridad	19
vii) Infraestructura y DevOps.....	19
B) EL SERVICIO WEB	19
i) Arquitectura del Backend	19
ii) Arquitectura del Frontend y Patrón BFF	20
iii) Estrategia Unificada de Seguridad y Autenticación	20
iv) Módulos Funcionales y Observabilidad	20
v) Escalabilidad y Despliegue.....	21
RESULTADOS.....	22
1) OPTIMIZACIÓN DE TIEMPOS DE RESPUESTA.....	22
2) CAPACIDAD DE PROCESAMIENTO Y ESCALABILIDAD	22
3) EFICIENCIA EN EL CONSUMO DE RECURSOS.....	22
4) ESTABILIDAD Y TOLERANCIA A FALLOS	23
CRONOGRAMA	24
A) OBJETIVOS.....	24
B) CALENDARIO	25
PRESUPUESTO.....	26
CONCLUSIONES.....	27

RECOMENDACIONES	28
USO DE IA GENERATIVA (GENAI)	29
REFERENCIAS BIBLIOGRÁFICAS	30
ANEXOS	31
A) ÍNDICE DE FIGURAS	31

VIII. INTRODUCCIÓN

Esta transformación en el modelo de gestión tributaria tiene como fin no solo perfeccionar el control fiscal y la recaudación, sino también refinar la operatividad de las compañías, disminuyendo gastos y tiempos administrativos. En este contexto, la Guía de Remisión se posiciona como un documento esencial para la logística y el transporte. Su generación correcta y oportuna es indispensable para asegurar que la cadena de suministro no sufra interrupciones.

No obstante, para corporaciones de la magnitud del Grupo Industrial Graiman, administrar un flujo masivo de transacciones diarias convierte la emisión de estas guías en un reto técnico considerable. Las arquitecturas tradicionales o monolíticas suelen presentar deficiencias al intentar escalar frente a picos de demanda, creando cuellos de botella que retrasan los despachos y merman la eficiencia operativa. La obligación de validar documentos en tiempo real, junto con los rigurosos estándares de seguridad y firma electrónica, hace imperativo el uso de soluciones tecnológicas que sean modernas, resilientes y altamente escalables.

Este trabajo de titulación presenta una solución técnica avanzada para abordar dicha problemática. La propuesta se basa en el paradigma cloud-native, implementando una arquitectura de microservicios orquestada a través de Kubernetes. Este enfoque permite fragmentar la complejidad del proceso de autorización en componentes independientes, favoreciendo así el escalado automático, la tolerancia a fallos y la integración continua.

A lo largo del documento se describe el ciclo integral de desarrollo del sistema, partiendo del análisis de los requerimientos normativos y operativos, hasta la implementación de módulos críticos para la firma digital bajo el estándar XAdES-BES y la comunicación asíncrona con los servicios del SRI. El propósito fundamental es demostrar cómo la tecnología de vanguardia puede convertir un trámite obligatorio en una ventaja competitiva de automatización y eficacia.

IX. PROBLEMA

1) DESCRIPCIÓN DEL PROBLEMA

Previo a la implementación de este proyecto, el Grupo Industrial Graiman (GIG) dependía de un proveedor externo para gestionar la emisión y autorización de los comprobantes electrónicos requeridos por el SRI. Esta subordinación tecnológica y operativa deriva en dos inconvenientes graves que comprometían tanto la continuidad del negocio como la eficiencia financiera de la empresa.

2) ANTECEDENTES

La raíz del problema se encuentra en las actualizaciones de la normativa legal y en las limitaciones técnicas del servicio anterior. Específicamente, a partir del 1 de enero de 2026, el SRI estableció la obligatoriedad de que la emisión y autorización de los comprobantes electrónicos se realice de manera inmediata, es decir, dentro del mismo día de su generación. El servicio contratado no garantizaba esa inmediatez requerida, lo cual ponía en riesgo el cumplimiento legal del GIG. Anteriormente, hasta el año 2025, la normativa permitía transmitir los comprobantes hasta en 4 días hábiles posteriores, pero la nueva disposición eliminó esa flexibilidad, volviendo obsoleta la capacidad de respuesta del proveedor externo frente a los nuevos tiempos regulatorios.

Adicionalmente, la infraestructura que se poseía carecía de un sistema centralizado de control, trazabilidad interna y monitoreo sobre los documentos electrónicos enviados. No se disponía de una plataforma propia para visualizar en tiempo real el estado de los documentos enviados al SRI, lo que dificultaba la detección de errores, el seguimiento para fines de auditoría interna y el cumplimiento de políticas de calidad en los procesos contables.

3) IMPORTANCIA Y ALCANCES

La importancia de abordar este problema radicó en la necesidad urgente de recuperar la autonomía tecnológica y optimizar los recursos del grupo empresarial.

Impacto Económico: El servicio de terceros representaba un costo estimado superior a los 30.000 USD anuales, generando una fuerte dependencia externa. La implementación de una solución propia permitió eliminar este gasto recurrente, redirigiendo la inversión hacia el fortalecimiento de la infraestructura interna.

Impacto Operativo: El volumen de comprobantes que manejaban las empresas del grupo era elevado, por lo que se requería que el sistema pudiera procesar múltiples documentos en paralelo de forma eficiente. La solución propuesta se enfocó en soportar picos de carga sin afectar la capacidad de respuesta ni el tiempo de autorización, garantizando la escalabilidad que el servicio anterior no ofrecía.

Beneficiarios: El proyecto benefició directamente al área contable y logística del GIG, al proveer herramientas de monitoreo en tiempo real, y a la organización en general, al asegurar el cumplimiento estricto de la normativa tributaria vigente desde el año 2026.

4) DELIMITACIÓN

Delimitación Espacial: El proyecto se desarrolló e implementó para la infraestructura tecnológica del GIG, cuyas operaciones principales se centralizan en Cuenca, Ecuador, pero con alcance a nivel nacional.

Delimitación Temporal: El desarrollo abarcó el periodo de transición normativa entre el año 2025 y la entrada en vigencia de las nuevas reglas del SRI en enero de 2026.

Delimitación Tecnológica: La solución propuesta se estructuró y delimitó en dos fases de desarrollo claramente diferenciadas:

- **Fase de automatización:** Se centró en el desarrollo del núcleo del sistema, responsable del procesamiento masivo desatendidos. Esta fase abarcó la lógica de extracción de datos del ERP, el armado XML base, la firma electrónica bajo estándar XAdES-BES, la comunicación directa con los servicios web del SRI, la generación de archivo PDF y la notificación al cliente. Es el componente principal que garantiza la inmediatez y el cumplimiento normativo.
- **Fase de sistema web:** Comprendió el desarrollo de una plataforma para el control humano, subdividida en:
 - **Backend:** Servicios API encargados de la lógica de negocio, consultas, gestión y uso de roles.
 - **Frontend:** Interfaz gráfica para que el personal pueda visualizar la trazabilidad, descargar los documentos (PDF/XML) y monitorear el estado de las autorizaciones en tiempo real.

En ambas fases se utilizó una arquitectura Cloud-Native sobre Kubernetes, integrándose con los sistemas ERP existentes (SAP/Dynamics) únicamente a nivel de lectura de datos, sin modificar su código fuente.

X. OBJETIVOS GENERALES Y ESPECÍFICOS

A) OBJETIVO GENERAL

Diseñar y desarrollar un sistema cloud-native para la gestión y automatización del proceso de autorización del documento electrónico Guías de Remisión del SRI en el Grupo Industrial Graiman.

B) OBJETIVOS ESPECÍFICOS

- Analizar los procesos actuales de generación, emisión y autorización del documento electrónico guías de remisión en GIG para identificar requerimientos funcionales.
- Diseñar e implementar un sistema para la gestión del documento electrónico en cloud-native basado en una arquitectura modular y escalable mediante microservicios contenedorizados en Kubernetes, para la recolección, firmado, envío y almacenaje de documentos electrónicos a los servicios web del SRI conforme la versión 2.32.
- Desarrollar un módulo web de trazabilidad y monitoreo, empleando un backend REST con FastAPI y un frontend en React, que permita visualizar el estado de cada documento enviado, generar alertas en caso de demoras o errores, y proveer funcionalidades de auditoría, garantizando así control y transparencia en todo el proceso de facturación electrónica.
- Evaluar el desempeño del sistema implementado mediante pruebas piloto y métricas de rendimiento, comparando el tiempo promedio de autorización, la tasa de errores o rechazos corregidos automáticamente, y el impacto en la reducción de costos operativos.

XI. REVISIÓN DE LA LITERATURA O FUNDAMENTOS TEÓRICOS

Para el desarrollo del presente trabajo de titulación, es fundamental establecer las bases teóricas y normativas que sustentan la solución tecnológica. A continuación, se describen los conceptos clave relacionados con la tributación electrónica, la seguridad criptográfica, las arquitecturas modernas de software y los patrones de confiabilidad.

1) FACTURACIÓN ELECTRÓNICA

En el contexto ecuatoriano, la facturación electrónica constituye un mecanismo estandarizado para la emisión, firma, envío y validación de documentos fiscales, abarcando facturas, guías de remisión, retenciones y notas de crédito o débito. Desde el punto de vista técnico, cada comprobante se estructura como un archivo XML que debe apegarse rigurosamente a las definiciones de esquema XSD establecidas por el ente regulador.

La fase de autorización se gestiona mediante servicios web que utilizan el protocolo SOAP, los cuales notifican el estado del trámite (recibido, autorizado, rechazado o con errores). Es crucial mencionar que, con la normativa vigente desde 2026, se impuso el requisito de transmisión y autorización inmediata. Esto fuerza a los sistemas corporativos a minimizar la latencia y a garantizar una trazabilidad exhaustiva para fines de auditoría. En este escenario, métricas como la velocidad de respuesta y la gestión de errores se han vuelto indicadores fundamentales de desempeño.

2) FIRMA DIGITAL Y ESTÁNDARES XAdES

La autenticidad y la integridad de los comprobantes electrónicos se aseguran mediante la firma digital, basada en una infraestructura de clave pública (PKI). Este sistema emplea certificados X.509, claves privadas bajo control del emisor y sellos de tiempo.

El SRI ha adoptado el estándar XAdES-BES (XML Advanced Electronic Signatures - Basic Electronic Signature), el cual permite firmar el archivo XML sin alterar su estructura. Para una gestión adecuada, se deben seguir prácticas de seguridad como:

- Resguardo estricto de los archivos de certificados.
- Uso de técnicas de derivación de claves para proteger secretos.
- Cifrado en reposo para las contraseñas y almacenamiento seguro de la clave privada.
- Verificación de la validez del certificado antes de ejecutar la firma.

3) ARQUITECTURAS EMPRESARIALES MODERNAS Y ENFOQUE CLOUD-NATIVE

El desarrollo de software corporativo ha evolucionado desde estructuras monolíticas hacia arquitecturas de microservicios. Este cambio busca separar los dominios del negocio, facilitar la actualización independiente de módulos y permitir un escalado de recursos ajustado a la demanda real.

La filosofía Cloud-Native se enfoca en crear aplicaciones diseñadas para operar en infraestructuras automatizadas y efímeras. Sus fundamentos incluyen el uso de contenedores con imágenes inmutables, la inyección de configuraciones en el entorno y una capacidad de observación basada en logs, métricas y trazas distribuidas. Este enfoque es indispensable para cumplir con la inmediatez requerida en la autorización, asegurando que cada transacción sea auditable.

4) CONTENERIZACIÓN Y ORQUESTACIÓN CON KUBERNETES

Para materializar la arquitectura de microservicios, se emplea Docker como estándar de empaquetado y ejecución, lo que asegura la consistencia entre los distintos entornos. Paralelamente, Kubernetes funciona como el sistema orquestador, encargado de gestionar despliegues que sean reproducibles y controlados de manera eficiente.

Entre los recursos clave de Kubernetes utilizados destacan:

- **Deployments y ReplicaSets:** Para asegurar la disponibilidad de las aplicaciones.
- **Services, Load Balancer e Ingress:** Para la exposición y enrutamiento de tráfico.
- **ConfigMaps y Secrets:** Para desacoplar la configuración y las credenciales del código fuente.
- **Probes (Liveness/Readiness):** Para garantizar la salud operativa, evitando enrutar tráfico a instancias no saludables.

Kubernetes proporciona tolerancia a fallos mediante la reprogramación automática de pods caídos y habilita el autoescalado horizontal (HPA) basado en la demanda. Esto permite que, ante picos de emisión de guías de remisión, el sistema aumente sus réplicas y ejecute trabajos en paralelo para sostener la inmediatez regulatoria sin sobredimensionar la infraestructura permanentemente.

5) CONCURRENCIA, COLAS Y PATRONES DE CONFIABILIDAD

El manejo eficiente de la concurrencia es vital para reducir la latencia y elevar el rendimiento. Esto se aborda en dos niveles complementarios:

- **Nivel de Pod:** Paralelismo mediante hilos (threads) para optimizar operaciones de entrada y salida (I/O) como el firmado, consultas a bases de datos, y comunicación con el SRI.
- **Nivel Distribuido:** Uso de colas compartidas donde múltiples pods consumen tareas, equilibrando la carga y evitando la duplicidad de trabajo.

Para fortalecer la confiabilidad del sistema frente a intermitencias del servicio tributario, se aplican patrones de diseño robustos:

- **Reintentos con retroceso exponencial:** Para manejar errores transitorios sin saturar el servicio destino.
- **Disyuntores:** Para aislar fallos y evitar errores en cascada.
- **Patrón Outbox:** Implementación de una bandeja de entrada transaccional para publicar y consumir eventos sin perder consistencia de datos.
- **Idempotencia:** Uso de claves únicas para impedir envíos duplicados durante los reintentos, preservando la exactitud y trazabilidad de extremo a extremo.

6) OBSERVABILIDAD Y PORTAL WEB DE TRAZABILIDAD

La observabilidad moderna integra registros estructurados, métricas y trazas distribuidas para monitorear el ciclo de vida completo de cada documento. Para un proceso fiscal crítico se definen tres tipos de métricas:

- **Métricas de negocio:** Tiempo promedio de autorización, tasa de rechazo, y documentos procesados por minuto.

- **Métricas técnicas:** Latencia de servicios SOAP, errores clasificados por código del SRI, tiempo de cómputo de firma, tamaño de payloads y uso de recursos como CPU y memoria.
- **Datos de auditoría:** Tiempo entre emitido y autorizado, guardado de todo el ciclo de vida del documento y número de autorización.

Para la visualización y gestión, se fundamenta el uso de tecnologías como FastAPI para backend y Next.js para frontend para construir portales que ofrezcan vistas en tiempo real, filtros avanzados, alertas y acceso detallado a las evidencias XML/PDF, garantizando el control operativo.

7) SEGURIDAD Y CUMPLIMIENTO

La seguridad de la información amplía la garantía de no repudio de la firma digital mediante varias capas de protección:

- **Cifrado en reposo:** Uso de algoritmos AES para proteger secretos y respaldos.
- **Resguardo de credenciales:** Implementación de algoritmos PBKDF2 para la derivación de claves asociadas a las contraseñas de los certificados.
- **Seguridad en la transmisión:** Empleo de protocolos TLS para cifrar las comunicaciones de extremo a extremo.
- **Gestión de accesos:** Adopción estricta del principio de privilegio mínimo en el acceso a bases de datos, sistemas de almacenamiento y recursos del clúster.

Adicionalmente, se establecen políticas rigurosas para la retención de datos en cumplimiento con la ley, lo cual asegura la integridad del sistema, facilita la auditoría (trazabilidad) y garantiza la resiliencia necesaria para la recuperación ante fallos.

XII. MARCO METODOLÓGICO

A) EL NÚCLEO DE AUTOMATIZACIÓN

La ejecución de este proyecto se rigió por una estrategia de ingeniería de software modular e incremental, poniendo énfasis en la seguridad, la capacidad de escalado y la robustez de la solución. En las siguientes líneas se describe la metodología constructiva y la propuesta técnica que sustenta el componente principal de automatización.

i) Enfoque de Desarrollo y Arquitectura

Para edificar el sistema, se optó por una arquitectura nativa de la nube (Cloud-Native) fundamentada en microservicios desacoplados. Esta elección metodológica permitió fragmentar la complejidad del proceso de autorización ante el SRI en fases autónomas; cada una de estas fases se despliega mediante contenedores Docker y es administrada a través de la orquestación de Kubernetes (AKS).

El corazón de la automatización se estructuró como un flujo de procesamiento secuencial (pipeline), en el que cada microservicio ejecuta una función atómica y específica. Los módulos esenciales desarrollados son:

- **Extracción (ERP/Adicional):** Conjunto de microservicios dedicados a la interacción e interfaz con los sistemas heredados.
- **Construcción (Base):** Componente encargado de generar y estructurar el archivo XML estandarizado.
- **Seguridad (Firmado):** Módulo criptográfico responsable de aplicar la firma electrónica bajo el estándar XAdES-BES.
- **Gestión (Envío/Auth):** Cliente SOAP para la comunicación con el SRI.
- **Salida (PDF/Notificación):** Generación de representaciones gráficas y entrega al usuario final.

ii) Implementación del Flujo de Procesamiento

Se diseñó un flujo de trabajo secuencial de 7 etapas que garantiza la integridad transaccional del documento. La metodología para el manejo de estados se basa en el patrón de Cola de Procesos con Bloqueo Distribuido.

Para evitar condiciones de carrera y duplicidad de envíos en un entorno concurrente, se implementó un mecanismo de bloqueo optimista en la base de datos SQL Server utilizando sentencias FOR UPDATE SKIP LOCKED. Esto permite que múltiples pods o instancias de microservicios procesen documentos en paralelo sin interferir entre sí.

iii) El ciclo de vida del documento transita por:

- **ERP:** Extracción de datos base.
- **ADICIONAL:** Extracción de información corporativa complementaria.
- **BASE:** Generación de la estructura XML bajo ficha técnica 2.32.

- **FIRMADO:** Aplicación de firma electrónica con certificados P12.
- **ENVÍO/AUTH:** Transmisión al SRI y polling de estado hasta obtener respuesta.
- **PDF:** Renderizado del RIDE con códigos de barras.
- **NOTIFICACIÓN:** Envío de correo electrónico al cliente.

iv) Paradigmas y Patrones de Diseño Aplicados

La solución se construyó utilizando el lenguaje Python 3.12, aplicando estrictamente la Programación Orientada a Objetos para modelar la lógica de negocio tributaria.

Jerarquía de Clases y Abstracción: Se implementó una clase base abstracta Comprobante de la cual heredan los tipos de documentos específicos. Esto facilita la extensión del sistema a nuevos documentos sin modificar el núcleo de procesamiento.

Patrones de Diseño: Se aplicaron patrones de la industria para resolver problemas comunes de diseño:

- **Factory Pattern:** Para la instanciación dinámica de comprobantes basada en el tipo de documento recibido.
- **Template Method:** Utilizado en los esquemas de validación para definir el esqueleto del algoritmo de procesamiento, permitiendo que cada subclase implemente sus pasos específicos.
- **Singleton:** Para la gestión eficiente de conexiones a base de datos y clientes de almacenamiento.
- **Repository Pattern:** Para abstraer la capa de acceso a datos y centralizar las consultas SQL.

Concurrencia y Asincronía: Para maximizar el rendimiento, se empleó un enfoque híbrido:

- **Multithreading:** Uso de hilos para paralelizar operaciones de E/S dentro de cada worker.
- **Asincronía:** Implementada en el microservicio de generación de PDFs para manejar altas cargas de renderizado sin bloquear el hilo principal.

v) Gestión de Datos y Persistencia

La gestión de la persistencia de datos se apoya en Azure SQL Database, empleando un esquema relacional debidamente normalizado. Para la interacción y manipulación de la información se recurrió a SQLAlchemy; este ORM no solo simplifica la administración de transacciones, sino que también desacopla la lógica de la aplicación del motor de base de datos específico.

Dentro del diseño del esquema, resaltan entidades esenciales como PROCESO_COLA, encargada de orquestar las tareas en espera, y DOCUMENTO_CABECERA, que resguarda los metadatos del sistema. De manera complementaria, se integraron tablas dedicadas a la auditoría, tales como TIEMPOS_HISTORICO y ERRORES_HISTORICO, las cuales son fundamentales para evaluar métricas de rendimiento por etapa y realizar un rastreo detallado de cualquier anomalía.

vi) Estrategia de Seguridad

La seguridad fue un eje transversal en la metodología de desarrollo, implementando:

- **Criptografía:** Uso de algoritmos AES y derivación de claves PBKDF2 para proteger contraseñas de certificados y configuraciones sensibles en la base de datos.
- **Gestión de Secretos:** Integración con Kubernetes Secrets para evitar la exposición de credenciales en el código fuente.
- **Aislamiento:** Los certificados digitales se almacenan cifrados en Azure Blob Storage y solo son accesibles mediante tokens temporales (SAS).

vii) Infraestructura y DevOps

La metodología de despliegue se automatizó mediante Azure Pipelines, estableciendo un flujo de Integración y Despliegue Continuo (CI/CD).

- **Containerización:** Cada microservicio se empaqueta en imágenes Docker optimizadas basadas en python:3.12-slim.
- **Orquestación:** Se definieron manifiestos de Kubernetes para recursos como Deployments, Services, ConfigMaps y CronJobs para limpieza de temporales.
- **Resiliencia:** Se configuraron sondas de salud Liveness y Readiness Probes para garantizar que el tráfico solo se dirija a instancias saludables, y políticas de reintento con backoff exponencial para tolerar fallos transitorios en los servicios del SRI.

B) EL SERVICIO WEB

Como elemento complementario al motor de automatización, se construyó una solución web que sirve como panel de mando y control para la organización. Esta herramienta faculta la administración, supervisión y fiscalización de los comprobantes electrónicos, concentrando las reglas operativas y los mecanismos de seguridad dentro de una estructura técnica que, pese a estar desacoplada, conserva una fuerte cohesión funcional.

i) Arquitectura del Backend

La lógica central de la plataforma web se fundamenta en la versión 3.12 de Python, aprovechando el framework FastAPI; esta elección técnica responde a la necesidad de gestionar procesos asíncronos con un desempeño superior. El sistema se erige sobre una arquitectura de microservicios que adopta patrones de diseño estrictos para asegurar su sostenibilidad a largo plazo:

- **Patrón Repository y ORM Híbrido:** Se logró desacoplar el acceso a la información mediante SQLAlchemy 2.0, operando en una modalidad dual. Esta configuración asigna el driver mssql+aiodbc para las lecturas asíncronas de alta demanda, mientras que utiliza mssql+pyodbc para las transacciones síncronas críticas, optimizando así la interacción con Azure SQL Database.
- **Patrón DTO (Data Transfer Objects):** La integridad de los datos se asegura mediante Pydantic, definiendo esquemas estrictos de entrada y salida que validan cada petición antes de que sea procesada por la lógica de negocio.

- **Orquestación de Infraestructura:** El backend integra el cliente oficial de Kubernetes, permitiendo realizar operaciones DevOps programáticas directamente desde la API, sin exponer el clúster directamente a internet.

ii) Arquitectura del Frontend y Patrón BFF

La interfaz de usuario se desarrolló como una Single Page Application (SPA) moderna utilizando Next.js y React. La arquitectura implementa el patrón BFF (Backend For Frontend) a través de las API Routes de Next.js. Este diseño permite que el servidor de frontend actúe como un proxy seguro, ocultando la topología de la red interna y manejando la inyección de credenciales antes de enviar las peticiones al backend Python.

El desarrollo se apoyó en TypeScript en modo estricto para garantizar la seguridad de tipos, y Tailwind CSS para un diseño de interfaz declarativo y responsivo. Para la optimización del ciclo de construcción y despliegue en contenedores, se utilizó Bun como entorno de ejecución y gestor de paquetes.

iii) Estrategia Unificada de Seguridad y Autenticación

La seguridad se abordó de manera transversal, conectando ambas capas mediante un flujo de autenticación híbrido y robusto:

- **Autenticación Federada (OAuth 2.0):** El frontend integra MSAL (Microsoft Authentication Library) para delegar la autenticación a Azure Active Directory, garantizando que solo personal corporativo del Grupo Industrial Graiman pueda acceder.
- **Gestión de Sesiones Seguras:** Una vez validada la identidad en Azure, el backend emite un token JWT propio, firmado con algoritmo RS256, que contiene los roles y permisos de la aplicación (RBAC). Este token se almacena en el navegador mediante Cookies HttpOnly, mitigando completamente los riesgos de ataques XSS.
- **Criptografía de Datos:** Para la protección de activos sensibles, como las contraseñas de los certificados digitales P12 y las configuraciones del sistema, se implementó cifrado simétrico AES-256 y derivación de claves con SHA-256, asegurando que ninguna credencial crítica se almacena en texto plano en la base de datos.

iv) Módulos Funcionales y Observabilidad

La solución web integra un conjunto de utilidades de gestión estructuradas en módulos funcionales:

- **Tableros de Control y Analítica:** Mediante el uso de Recharts, se despliegan gráficos interactivos que reflejan los indicadores operativos procesados por el servidor. Esto hace posible monitorear en tiempo real variables como el caudal de transacciones, los porcentajes de error y la latencia en las autorizaciones.
- **Control Documental:** Un componente robusto de consulta y filtrado que brinda acceso al repositorio de documentos, permitiendo verificar la trazabilidad integral de cada proceso y descargar los archivos PDF generados al momento.

- **Administración de Infraestructura (Kubernetes):** Panel técnico especializado para supervisar el estado de los pods, examinar registros en vivo y administrar recursos del clúster vía API. Esto facilita las operaciones de mantenimiento sin requerir acceso directo a la consola de la nube.

v) Escalabilidad y Despliegue

En cuanto a la infraestructura, el servicio web opera bajo un esquema de contenedores Docker orquestados por Kubernetes. Se habilitó un Horizontal Pod Autoscaler para el backend, el cual ajusta automáticamente las réplicas en función del uso de CPU y memoria ante variaciones de carga. Paralelamente, el frontend se ejecuta como un servicio optimizado en modo standalone, garantizando un consumo eficiente de recursos.

XIII. RESULTADOS

La adopción de la arquitectura nativa de la nube para el manejo de Guías de Remisión en el Grupo Industrial Graiman ha derivado en indicadores tangibles de mejora en cuanto a fiabilidad, capacidad de escalado y eficiencia operativa. En los apartados siguientes se exponen los datos obtenidos tras la puesta en producción y la ejecución de las pruebas de estrés.

1) OPTIMIZACIÓN DE TIEMPOS DE RESPUESTA

Una meta fundamental de esta iniciativa fue alinearse con el requisito de inmediatez estipulado por el SRI para el ejercicio fiscal 2026. Al cotejar el desempeño del esquema previo frente al nuevo desarrollo interno, se observa una disminución significativa en el tiempo total de vida del documento.

- **Situación Anterior:** La gestión de autorizaciones, al estar subordinada a un proveedor tercerizado, sufría de retardos considerables que variaron entre 8 y 24 horas, lapso comprendido desde el registro inicial en el ERP hasta la validación definitiva.
- **Resultados Obtenidos:** Con la nueva arquitectura de microservicios, el tiempo promedio de procesamiento completo se ha reducido a un rango de 2 a 5 minutos por documento.

Esta optimización del 99% en los tiempos de respuesta garantiza que las unidades de transporte salgan de las instalaciones con documentación autorizada en tiempo real, eliminando riesgos de sanciones en controles de carretera.

2) CAPACIDAD DE PROCESAMIENTO Y ESCALABILIDAD

El sistema fue sometido a pruebas de carga para verificar el comportamiento del autoescalado horizontal de Kubernetes frente al flujo real de datos del ERP.

- **Carga Real Promedio:** Actualmente, el flujo de operación normal del Grupo Industrial Graiman inyecta al sistema un promedio de 2 a 3 documentos por minuto.
- **Capacidad Base y Escalabilidad:** Las pruebas de estrés demostraron que, incluso en su configuración base (número mínimo de pods activos), el sistema es capaz de procesar hasta 50 documentos por minuto sin degradación del servicio, siendo el mayor limitante la respuesta del SRI.
- **Escalabilidad Teórica:** Gracias al diseño desacoplado y al uso de colas con bloqueo distribuido, el sistema demostró ser horizontalmente escalable, permitiendo instanciar nuevos workers de forma dinámica ante picos de demanda.

3) EFICIENCIA EN EL CONSUMO DE RECURSOS

Uno de los logros técnicos más destacados es la eficiencia del código desarrollado en Python y la optimización de los contenedores Docker. El monitoreo en Kubernetes arrojó métricas de consumo extremadamente bajas para una operación empresarial, lo que se traduce en costos de nube mínimos:

- **Núcleo de Automatización:** En operación promedio, cada instancia consume apenas 60m de CPU (60 milicores, equivalente al 6% de un núcleo) y 100MB de memoria RAM.
- **Backend de Gestión:** El servicio API mantiene un consumo promedio de 60m de CPU y 150MB de memoria RAM.

Estos valores demuestran que la arquitectura es altamente ligera, permitiendo ejecutar toda la infraestructura crítica del sistema con una fracción de los recursos

4) ESTABILIDAD Y TOLERANCIA A FALLOS

Durante el periodo de evaluación, la lógica de negocio implementada en el núcleo del sistema demostró una estabilidad del 100%, no registrándose errores de código, fugas de memoria o caídas de servicios por excepciones no controladas.

Los incidentes registrados se categorizaron en dos tipos externos, gestionados exitosamente por los mecanismos de resiliencia:

1. **Indisponibilidad del SRI:** Ante caídas de los servicios web tributarios, el sistema activó automáticamente sus políticas de reintentos con backoff exponencial, procesando los documentos una vez restablecido el servicio
2. **Errores de Calidad de Datos:** Se detectaron casos de información incompleta desde el ERP, es decir, error de usuario. El sistema identificó estas anomalías, detuvo el proceso específico y generó las notificaciones y alertas correspondientes para su corrección inmediata.

XIV. CRONOGRAMA

Ejecución incremental con iteraciones cortas. Empezamos por análisis y riesgos normativos, pasamos al diseño e implementación cloud-native en AKS, levantamos el portal de trazabilidad y cerramos con pruebas de desempeño, resiliencia, seguridad y la evaluación antes/después.

A) OBJETIVOS

O1. Analizar los procesos actuales de generación, emisión y autorización de documentos electrónicos en GIG para identificar requerimientos funcionales.

No.	Actividad
1	Relevamiento de procesos actuales en GIG
2	Revisión normativa SRI v2.32
3	Identificación de requerimientos funcionales y no funcionales
4	Matriz de actores y permisos

O2. Diseñar e implementar un sistema de gestión de documentos electrónicos cloud-native basado en una arquitectura modular y escalable mediante microservicios contenedorizados en Kubernetes, para la recolección, firmado, envío y almacenaje de documentos electrónicos a los servicios web del SRI conforme la versión 2.32.

No.	Actividad
1	Diseño arquitectónico modular
2	Desarrollo del sistema de automatización escalable con todas sus etapas respectivas
3	Contenerización y despliegue en Kubernetes
4	Observabilidad técnica básica

O3. Desarrollar un módulo web de trazabilidad y monitoreo, empleando un backend REST con FastAPI y un frontend en React, que permita visualizar el estado de cada documento enviado, generar alertas en caso de demoras o errores, y proveer funcionalidades de auditoría, garantizando así control y transparencia en todo el proceso de facturación electrónica.

No.	Actividad
1	Diseño de API de monitoreo en FastAPI
2	Construcción del frontend en Next.js

3	Integración con métricas y alertas
4	Control de acceso por roles

O4. Evaluar el desempeño del sistema implementado mediante pruebas piloto y métricas de rendimiento, comparando el tiempo promedio de autorización, la tasa de errores o rechazos corregidos automáticamente, y el impacto en la reducción de costos operativos.

No.	Actividad
1	Definición del plan piloto
2	Medición de tiempo promedio por etapa, recursos utilizados y tasa de errores
3	Análisis de ahorro operativo

B) CALENDARIO

Objetivo	Actividad	Fecha Inicio	Fecha Fin	Horas por Responsable	Responsable
O1	Act.1	11/11/2025	12/11/2025	10	Juan Malo
O1	Act.2	12/11/2025	13/11/2025	10	Juan Malo
O1	Act.3	14/11/2025	15/11/2025	7	Juan Malo
O1	Act.4	16/11/2025	16/11/2025	4	Juan Malo
O2	Act.1	17/11/2025	23/11/2025	30	Juan Malo
O2	Act.2	24/11/2025	10/12/2025	80	Juan Malo
O2	Act.3	11/12/2025	15/12/2025	25	Juan Malo
O2	Act.4	16/11/2025	17/12/2025	10	Juan Malo
O3	Act.1	18/12/2025	22/12/2025	20	Juan Malo
O3	Act.2	23/12/2025	28/12/2025	20	Juan Malo
O3	Act.3	29/12/2025	30/12/2025	8	Juan Malo
O3	Act.4	02/01/2026	02/01/2026	4	Juan Malo
O4	Act.1	03/01/2026	04/01/2026	8	Juan Malo
O4	Act.2	05/01/2026	06/01/2026	10	Juan Malo
O4	Act.3	07/01/2026	07/01/2026	4	Juan Malo
Total				250h	

XV. PRESUPUESTO

A continuación se presenta el presupuesto del proyecto. Esta sección tiene el objetivo de mostrar cuánto costará realmente poner en marcha la solución cloud-native.

Denominación	Cantidad	Costo Unitario	Costo Total
1. Bienes			
Computadora	1	1,000.00	1,000.00
2. Cloud			
Pago de la nube de Azure	12	350.00	4200.00
3. Personal			
Estudiante	250h	20.00	5,000.00
4. Servicios			
Transporte	2 meses	30.00	60.00
Internet	2 meses	30.00	60.00
Total			10,320.00

XVI. CONCLUSIONES

Tras finalizar el diseño, desarrollo e implementación del sistema Cloud-Native para la gestión de Guías de Remisión en el Grupo Industrial Graiman, se exponen las siguientes conclusiones derivadas de los objetivos planteados y los resultados obtenidos:

- **Cumplimiento Normativo y Autonomía Tecnológica:** El análisis de los procesos previos evidenció que la dependencia de proveedores externos representaba un riesgo crítico para el cumplimiento de la normativa de inmediatez del SRI estipulada para 2026. Con la implementación de la nueva solución, se logró la total autonomía tecnológica del GIG, garantizando no solo el cumplimiento legal, sino también la capacidad de adaptar el sistema ágilmente ante futuros cambios regulatorios sin costos de licenciamiento de terceros.
- **Eficiencia de la Arquitectura Cloud-Native:** Los resultados confirman que la adopción de microservicios bajo la orquestación de Kubernetes representa la solución más adecuada para entornos fiscales de alta demanda. La estrategia de segmentar el flujo operativo en fases autónomas aseguró una ejecución estable y una gestión óptima de los recursos tecnológicos.
- **Impacto en la Operatividad Logística:** La optimización de los tiempos de autorización fue el logro más significativo del proyecto. Se redujo el ciclo de vida del documento de un rango de 8 a 24 horas a un promedio de 2 a 5 minutos con el nuevo sistema. Esta reducción del 99% en la latencia tiene un impacto directo en la cadena logística, eliminando los tiempos de espera de los transportistas y mitigando el riesgo de sanciones durante los controles de tránsito.
- **Control y Trazabilidad Integral:** El desarrollo del módulo web de gestión solucionó la falta de visibilidad que existía. La implementación de paneles de monitoreo en tiempo real y sistemas de alertas automáticas ha permitido pasar de un modelo reactivo a uno proactivo. Ahora, el equipo contable puede identificar y corregir errores de emisión en cuestión de minutos, garantizando la integridad de la información tributaria reportada.
- **Escalabilidad y Resiliencia:** Las pruebas de carga confirmaron que el sistema posee una escalabilidad horizontal elástica gracias al uso de patrones de diseño como colas con bloqueo distribuido y autoescalado. El sistema demostró "inmunidad" ante caídas del servicio del SRI mediante mecanismos de reintento automático, asegurando que ningún documento se pierda o quede en un estado inconsistente, independientemente de la estabilidad de los servicios externos.

XVII. RECOMENDACIONES

Con base en la experiencia adquirida durante el desarrollo y la puesta en marcha del sistema, y considerando el potencial de la arquitectura Cloud-Native implementada, se plantean las siguientes recomendaciones para garantizar la evolución y sostenibilidad de la plataforma:

- **Expansión Horizontal a Otros Documentos Tributarios:** Dada la naturaleza modular del sistema y el uso de patrones de diseño extensibles, se recomienda migrar gradualmente el resto de documentos electrónicos del grupo a esta plataforma. Centralizar toda la emisión tributaria en este ecosistema unificaría la gestión tecnológica, eliminaría la deuda técnica y aprovecharía la eficiencia de costos ya instalada.
- **Explotación de Datos y Analítica Predictiva:** El sistema almacena una gran cantidad de telemetría de negocio. Se sugiere implementar un módulo de Business Intelligence o modelos de Machine Learning básicos que permitan predecir picos de facturación para pre-escalar la infraestructura, o detectar anomalías antes de enviarlas al SRI, actuando como un auditor fiscal preventivo automatizado.
- **Evolución hacia metodologías GitOps:** Aunque el sistema cuenta con un pipeline CI/CD robusto en Azure Pipelines, se recomienda evolucionar la gestión de la infraestructura hacia un enfoque GitOps. Esto permitiría que el estado del clúster de Kubernetes se sincronice automáticamente con el repositorio de código, facilitando rollbacks instantáneos ante fallos en despliegues y mejorando la auditoría de cambios en la configuración.
- **Política de Rotación de Credenciales:** A pesar de contar con cifrado robusto, se aconseja establecer una política automatizada para la rotación periódica de las llaves de cifrado maestras y los secrets del clúster. Esto minimiza la superficie de ataque y asegura que, en el caso hipotético de una filtración de credenciales antiguas, la integridad del sistema actual no se vea comprometida.

XVIII. USO DE IA GENERATIVA (GENAI)

GenAI ha facilitado la redacción y el refinamiento del lenguaje, pero el desarrollo de ideas, argumentos, resultados y conclusiones académicas/científicas se han realizado por el autor. Todo el contenido generado por GenAI ha sido revisado y editado minuciosamente por el autor, quienes asumen toda la responsabilidad por el contenido final de esta publicación.

XIV. REFERENCIAS BIBLIOGRÁFICAS

- ETSI. (2024, julio 23). Electronic signatures and trust infrastructures (ESI); XAdES digital signatures; Part 1: Building blocks and XAdES baseline signatures (EN 319 132-1 V1.3.1). European Telecommunications Standards Institute. https://www.etsi.org/deliver/etsi_en/319100_319199/31913201/01.03.01_60/en_31913201v010301p.pdf
- The Kubernetes Authors. (2025, agosto 7). Kubernetes documentation. Cloud Native Computing Foundation. <https://kubernetes.io/docs/home/>
- Servicio de Rentas Internas del Ecuador. (2025, octubre 8). Ficha técnica comprobantes electrónicos. Esquema offline. Versión 2.32. Servicio de Rentas Internas. <https://www.sri.gob.ec/o/sri-portlet-biblioteca-alfresco-internet/descargar/29562323-2e76-42f5-abb6-cb7ac542c3c6/FICHA%20TE%CC%81CNICA%20COMPROBANTES%20ELECTRO%CC%81NICOS%20ESQUEMA%20OFFLINE%20Versio%CC%81n%202.32.pdf>

XX. ANEXOS

A) ÍNDICE DE FIGURAS

- **Figura 1:** Diagrama de secuencia del proceso ERP
- **Figura 2:** Diagrama de secuencia del proceso Adicional
- **Figura 3:** Diagrama de secuencia del proceso Base
- **Figura 4:** Diagrama de secuencia del proceso Firmado
- **Figura 5:** Diagrama de secuencia del proceso Envío/Auth
- **Figura 6:** Diagrama de secuencia del proceso PDF
- **Figura 7:** Diagrama de secuencia del proceso Notificación
- **Figura 8:** Arquitectura general de conexión del sistema.
- **Figura 9:** Arquitectura de Microservicios para Facturación Electrónica (SRI) en Azure.

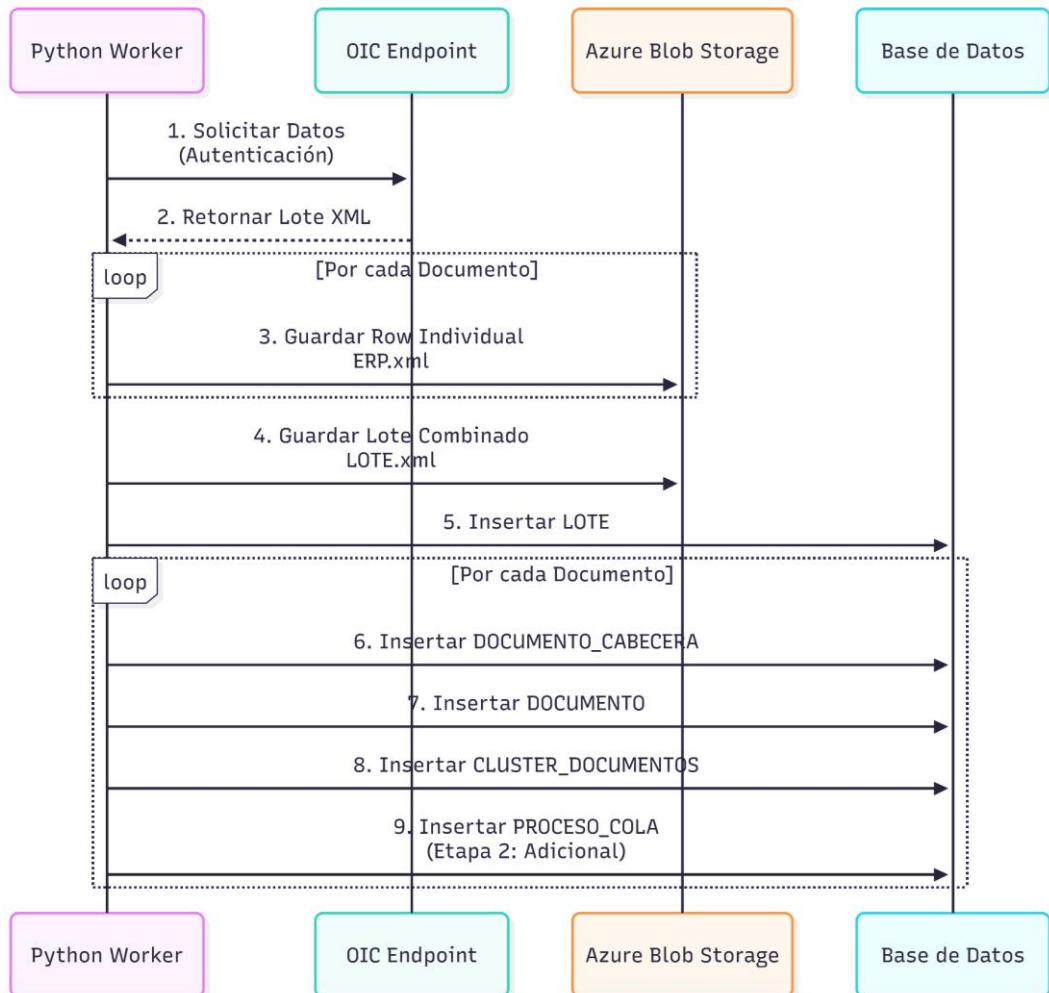


Figura 1. Diagrama de secuencia del proceso ERP

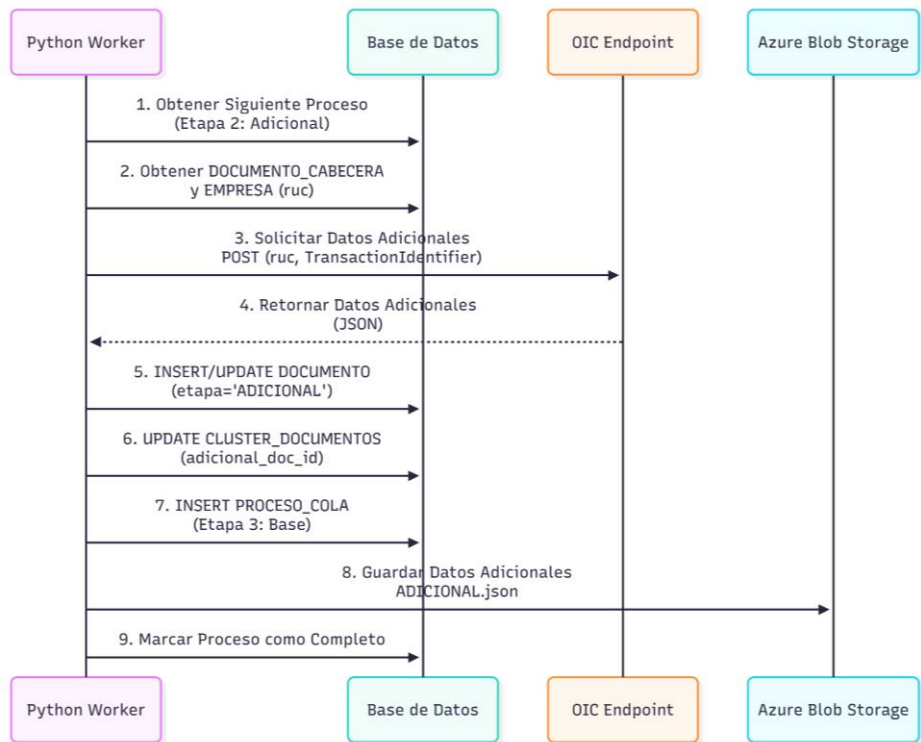


Figura 2. Diagrama de secuencia del proceso Adicional



Figura 3. Diagrama de secuencia del proceso Base

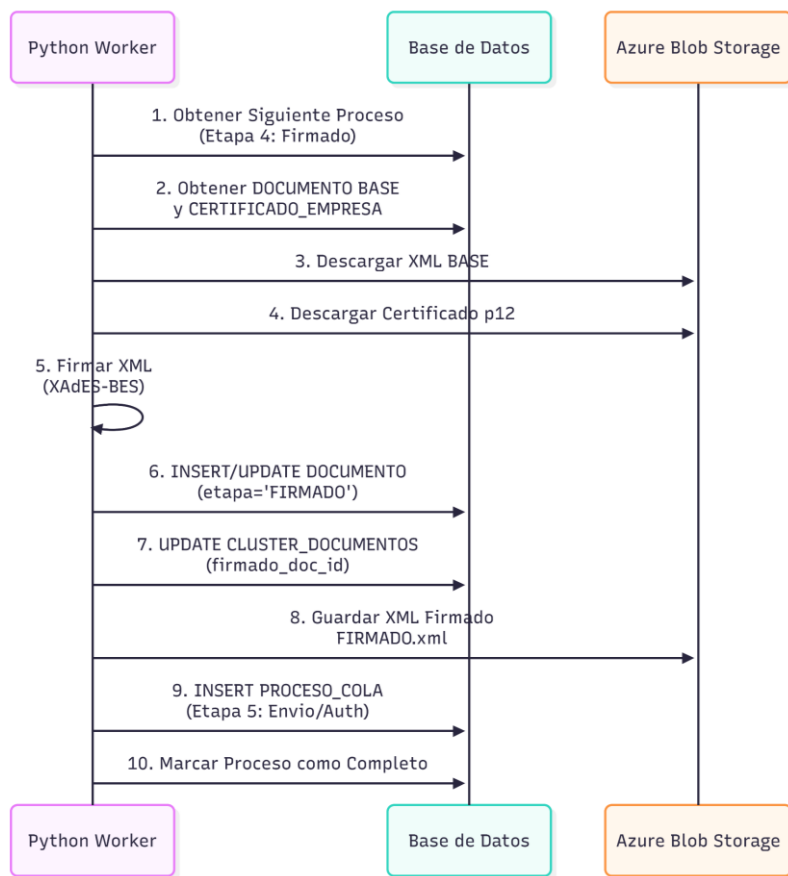


Figura 4. Diagrama de secuencia del proceso Firmado

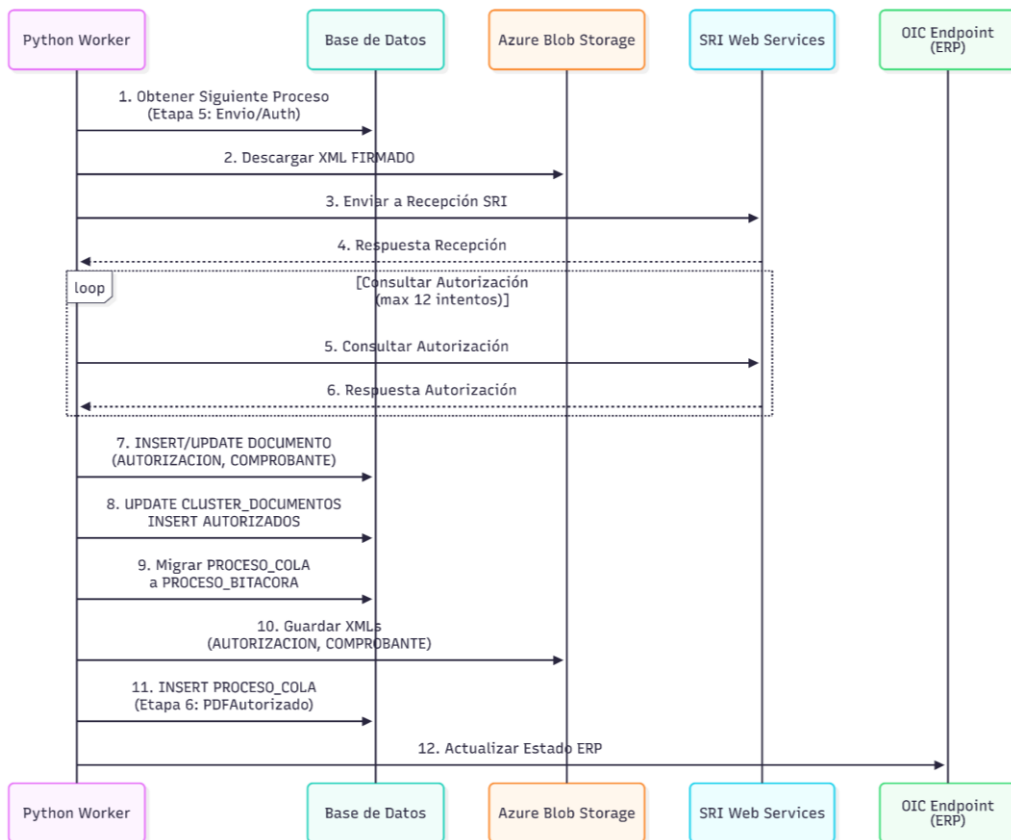


Figura 5. Diagrama de secuencia del proceso Envío/Auth

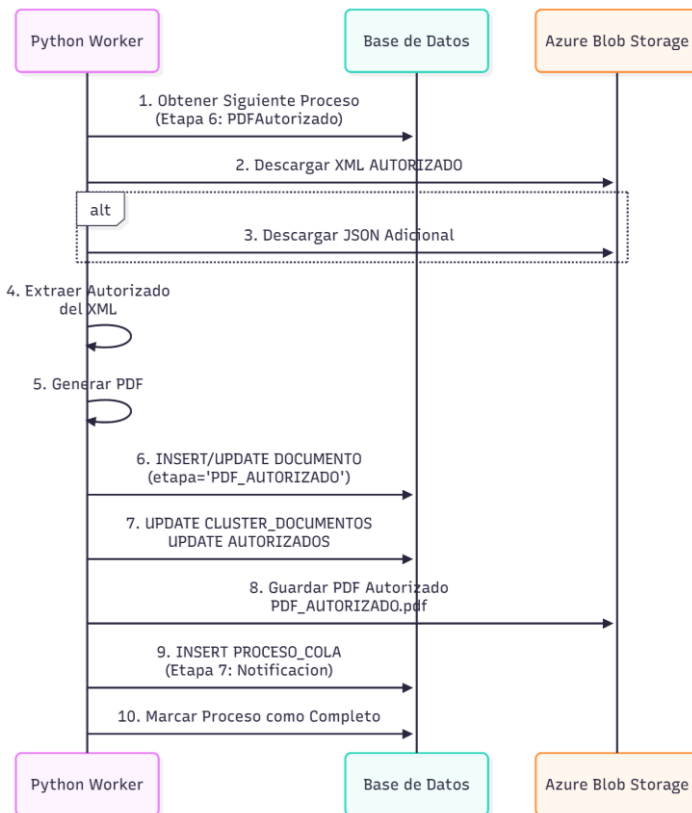


Figura 6. Diagrama de secuencia del proceso PDF

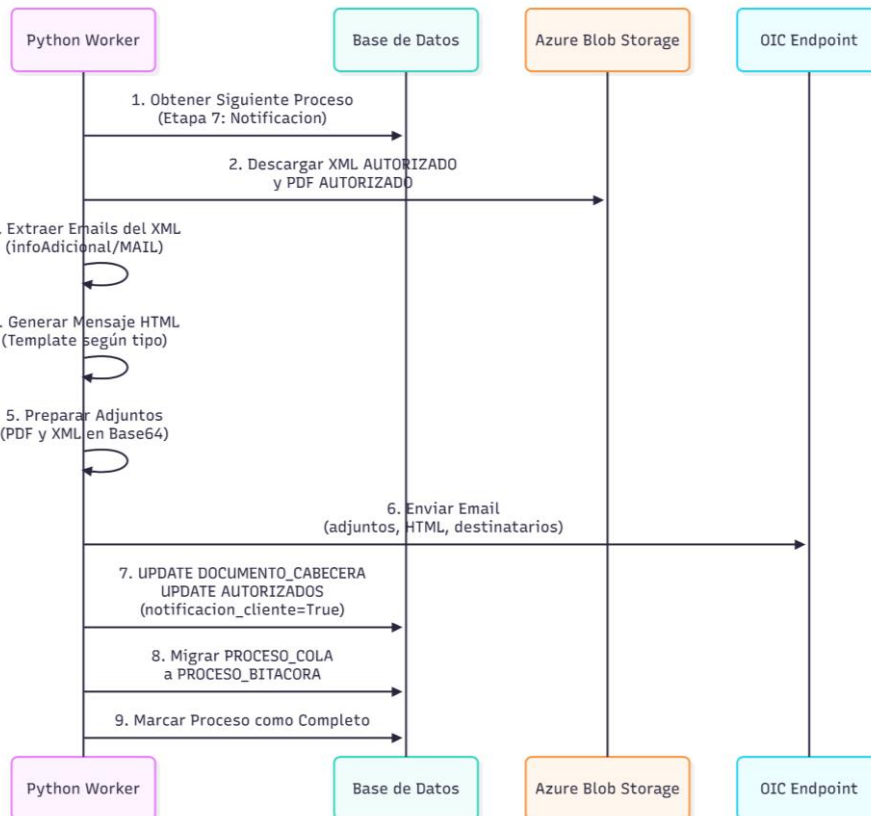


Figura 7. Diagrama de secuencia del proceso Notificación

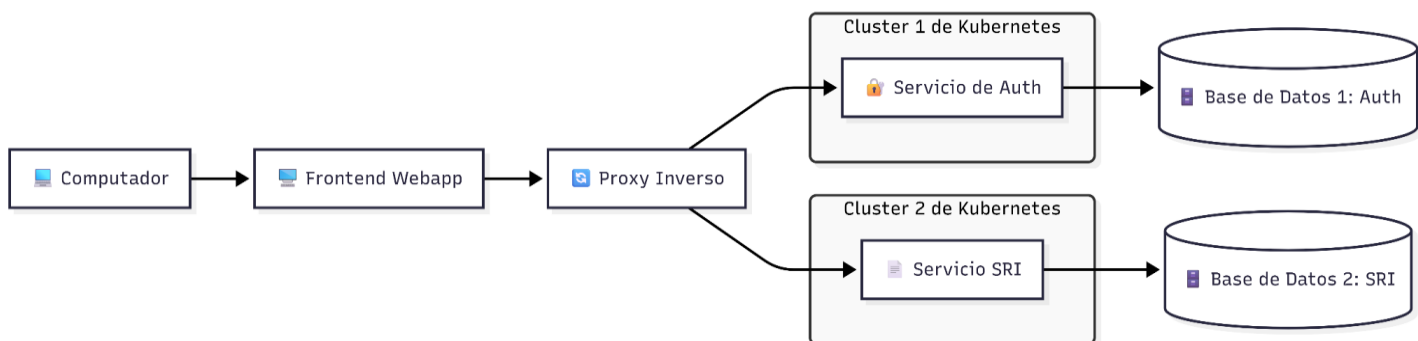


Figura 8. Arquitectura general de conexión del sistema.

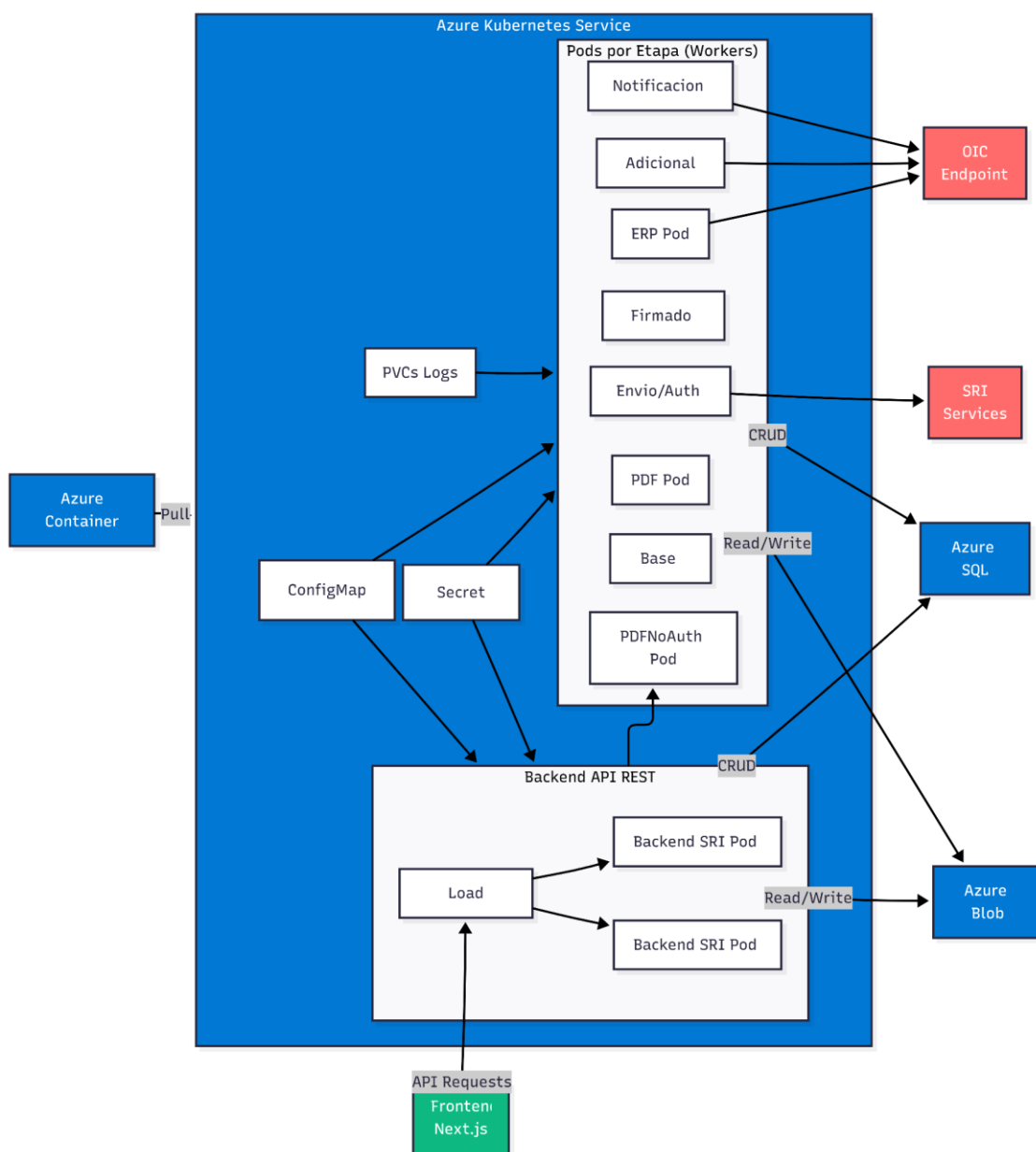


Figura 9. Arquitectura de Microservicios para Facturación Electrónica (SRI) en Azure.