# Solución de errores frecuentes de las plataformas de software de Ayudas Diagnósticas Suramericana S.A.: Implementación de mejoras para el manejo de errores del proceso de entrega de resultados a IPSA

Jose Franco Arroyave Cardona

Informe de práctica para optar al título de Ingeniero de Sistemas

Modalidad de Práctica

Práctica Empresarial

Asesores

Deisy Loaiza Berrío, Ingeniera de sistemas

Juan Pablo Palacio Guzmán, Ingeniero de sistemas

Universidad de Antioquia

Facultad de Ingeniería

Ingeniería de Sistemas

Medellín, Antioquia, Colombia

2025

| | |
|---|---|
| **Cita** | (Arroyave Cardona, 2025) |
| **Referencia**<br><br>**Estilo APA 7 (2020)** | Arroyave Cardona, J (2025). *Solución de errores frecuentes de las plataformas de software de Ayudas Diagnósticas Suramericana S.A.: Implementación de mejoras para el manejo de errores del proceso de entrega de resultados a IPSA* [Informe de práctica]. Universidad de Antioquia, Medellín, Colombia. |

creative commons CC BY NC SA

UNIVERSIDAD DE ANTIOQUIA
Vicerrectoría de Docencia

Sistema de Bibliotecas

Centro de Documentación Ingeniería (CENDOI)

**Repositorio Institucional:** http://bibliotecadigital.udea.edu.co

Universidad de Antioquia - www.udea.edu.co

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

**Tabla de contenido**

Resumen ..... 7

Abstract ..... 8

1. Introducción ..... 9

2. Objetivos ..... 9

2.1. Objetivo general ..... 9

2.2. Objetivos específicos ..... 10

3. Marco teórico ..... 10

3.1. Compañía Ayudas Diagnósticas ..... 10

3.2. Proceso de notificaciones: Mensajería Dinámica (MD) – Institución Prestadora de Salud @ (IPSA) ..... 12

3.3. Soporte a Incidentes ..... 13

4. Metodología ..... 14

5. Análisis de resultados ..... 15

6. Conclusiones y recomendaciones ..... 21

Referencias ..... 23

Anexos ..... 24

Anexo 1: Poster ..... 24

## Lista de tablas

**Tabla 1** Muestra de errores, agrupación por categorías y porcentajes asociados.....15

**Tabla 2** Comparación de mediciones previa y posterior a implementación.....21

## Lista de figuras

**Figura 1** Script SQL CREATE TABLE TSIR_CAMBIOSIDENTIFICACION.....16

**Figura 2** Implementación DTO TSIR_CAMBIOIDENTIFICACION .....17

**Figura 3** Implementación ListenerColaCambioIdentificacion.....18

**Figura 4** Implementación filtro para reenvío de órdenes IPSA.....19

**Figura 5** Implementación Función validadErroresIpsaReintentar .....19

## Siglas, acrónimos y abreviaturas

| | |
|---|---|
| **UdeA** | Universidad de Antioquia |
| **MD** | Mensajería Dinámica |
| **IPSA** | Institución Prestadora de Salud @ |
| **OPC** | Optimizador de Proceso Comerciales |
| **BH** | Beyond Health |
| **ITIL** | IT Infrastructure Library |
| **LIS** | Laboratory Integration System |
| **HIS** | Hospital Information System |
| **SQL** | Structured Query Language |
| **DevOps** | Development Operations |
| **DTO** | Data Transfer Object |
| **DAO** | Data Access Object |

## Resumen

Este informe presenta el proceso de estabilización y mejora en la notificación de resultados médicos entre las plataformas Mensajería Dinámica (MD) e IPSA en Ayudas Diagnósticas de Suramericana S.A. La infraestructura de notificación, basada en la tecnología de mensajería RabbitMQ, ha experimentado errores recurrentes que afectan la entrega oportuna de resultados a pacientes y prestadores de salud.

Para abordar esta problemática, se realizó un análisis detallado de los errores registrados en la plataforma, clasificándolos en errores técnicos y de negocio. Se establecieron estrategias para corregir los incidentes más críticos, priorizando aquellos con mayor impacto operativo. La implementación de soluciones técnicas permitió reducir significativamente la tasa de errores, optimizando la comunicación entre MD e IPSA y asegurando la correcta integración de resultados en la historia clínica de los pacientes.

El presente documento detalla la metodología aplicada, los hallazgos obtenidos y las estrategias implementadas, resaltando la importancia de una gestión eficiente de incidentes para garantizar la estabilidad del sistema y la mejora continua en los procesos tecnológicos.

*Palabras clave*: Mensajería Dinámica, RabbitMQ, notificación de resultados, errores técnicos, errores de negocio, Integración de sistemas.

## Abstract

This report presents the process of stabilizing and improving the notification of medical results between the Mensajería Dinámica (MD) and IPSA platforms in Ayudas Diagnósticas at Suramericana S.A. The notification infrastructure, based on RabbitMQ messaging technology, has experienced recurring errors affecting the timely delivery of results to patients and healthcare providers.

To address this issue, a detailed analysis of the recorded errors was conducted, classifying them into technical and business-related errors. Strategies were established to correct the most critical incidents, prioritizing those with the greatest operational impact. The implementation of technical solutions significantly reduced the error rate, optimizing communication between MD and IPSA and ensuring the correct integration of results into patients' medical records.

This document details the applied methodology, findings, and implemented strategies, highlighting the importance of efficient incident management to ensure system stability and continuous improvement in technological processes.

Keywords: Mensajería Dinámica, RabbitMQ, result notification, technical errors, business errors, system integration.

# 1. Introducción

La correcta integración de sistemas de información en el sector salud es fundamental para garantizar una atención eficiente y oportuna a los pacientes. En el contexto de Ayudas Diagnósticas de Suramericana S.A., la plataforma Mensajería Dinámica (MD) cumple un rol crucial en la gestión de notificaciones de resultados médicos, permitiendo la comunicación entre los laboratorios y el sistema de historia clínica IPSA.

A pesar de la importancia de este proceso, se han identificado múltiples fallos en la transmisión de resultados, lo que ha generado retrasos y errores en la disponibilidad de la información médica. Estos fallos pueden ser de origen técnico, relacionados con la infraestructura tecnológica, o de negocio, asociados a inconsistencias en la parametrización y configuración de los sistemas involucrados.

El presente proyecto busca mejorar la estabilidad del proceso de notificación de resultados, identificando las causas raíz de los errores y proponiendo soluciones técnicas que permitan reducir su recurrencia. A través de un enfoque estructurado de análisis y optimización, se espera fortalecer la comunicación entre MD e IPSA, asegurando la entrega precisa y confiable de los resultados médicos en el ecosistema digital de Ayudas Diagnósticas.

# 2. Objetivos

## 2.1. Objetivo general

Reducir la recurrencia de incidentes en las plataformas de Ayudas Diagnósticas de Suramericana S.A., mejorando la disponibilidad de los sistemas y la satisfacción de los usuarios finales.

### 2.2. Objetivos específicos

- Identificar las principales causas de los incidentes recurrentes en las plataformas de Ayudas Diagnósticas.
- Analizar soluciones tecnológicas en las mesas de trabajo, que aborden los incidentes más críticos y frecuentes.
- Implementar soluciones raíz empleando Java, Weblogic y SQL server, según se encuentre oportuno, de acuerdo con los casos identificados, incidiendo en los problemas técnicos que afectan la operación de las plataformas.
- Documentar los procedimientos empleados que dan solución a los incidentes, permitiendo una sistematización del proceso.

## 3. Marco teórico

### 3.1. Compañía Ayudas Diagnósticas

Ayudas Diagnósticas, como entidad prestadora de servicios, y parte de la compañía Suramericana S.A. (SURA) de seguros, se centra en la gestión de exámenes médicos de laboratorio e imagenología, encargándose del manejo de muestras biológicas, así como su procesamiento en equipos analizadores especializados, la toma de imágenes como radiografías y su disposición para los usuarios finales. Lo anterior, cumpliendo con las normativas legales y las reglas de negocio establecidas.

Para la gestión en Ayudas Diagnósticas, se cuenta con una amplia infraestructura de software y hardware interconectado que soporta la operación de la entidad, permitiendo dar respuesta oportuna a las necesidades de los usuarios a nivel nacional. Uno de los componentes más relevantes de la infraestructura es Mensajería Dinámica (MD), el cual nace de la necesidad de centralizar el flujo de información y la lógica del negocio para las diferentes integraciones con proveedores y usuarios.

En cuanto al flujo de información de la aplicación, las interacciones con MD comienzan con el ingreso de órdenes médicas que provienen de fuentes externas y contienen un conjunto de

exámenes o prestaciones que deben ser realizados a los pacientes, a estas órdenes se debe asociar una muestra biológica por medio de un código único; tanto las muestras, como las órdenes, son entregadas a los laboratorios para ser analizadas.

Actualmente, este software se integra con múltiples proveedores que permiten la comunicación entre los equipos de laboratorio y los sistemas de información, posterior a la realización de las pruebas, los resultados son notificados a Mensajería Dinámica y se distribuyen a los diferentes actores que puedan requerirlas, ya sean pacientes, médicos u otros sistemas de información.

Mensajería Dinámica, como solución de software, es un desarrollo propio de la compañía, diseñado, construido desde cero para satisfacer las necesidades específicas de SURA y Ayudas Diagnósticas. La aplicación utiliza *Java*, una plataforma informática orientada a objetos, creada por Sun Microsystems y adquirida posteriormente por Oracle. Esta tecnología proporciona un entorno robusto y ampliamente adoptado para el desarrollo de software empresarial (Oracle, s.f.). Además, la solución implementa el framework *Spring Boot*, una herramienta de desarrollo para aplicaciones Java enfocada en la velocidad, simplicidad y productividad. Este framework permite la creación de back-ends sostenibles y robustos a través de mejores prácticas de desarrollo de software, como la inyección de dependencias y configuraciones automáticas (Spring, s.f.).

En el lado del front-end, la aplicación utiliza *Angular*, un framework desarrollado por Google para crear interfaces de usuario dinámicas y altamente responsivas. Angular integra mejores prácticas para el desarrollo web, como modularidad y soporte para aplicaciones progresivas, facilitando la creación de experiencias de usuario óptimas (Google, s.f.).

Para el servidor de aplicaciones, se utiliza *WebLogic*, un servidor empresarial que gestiona el despliegue de aplicaciones Java en entornos distribuidos y asegura alta disponibilidad y escalabilidad (Oracle, s.f.). La base de datos emplea *SQL Server*, que ofrece características avanzadas de gestión y análisis de datos (Microsoft, s.f.).

Por último, en la integración con equipos de laboratorio, Mensajería Dinámica se conecta a middlewares especializados, componentes de software, que actúan como puente entre los analizadores físicos y la aplicación, garantizando una comunicación efectiva y precisa para el análisis de muestras biológicas.

En sistemas como el de Ayudas Diagnósticas, que requieren procesamiento en tiempo real de grandes volúmenes de datos provenientes de diferentes analizadores y sistemas, es fundamental el uso de software de alta concurrencia. Este tipo de software está diseñado para manejar múltiples procesos o solicitudes simultáneamente, lo cual es esencial para mantener la eficiencia y la disponibilidad de servicios en entornos con alto tráfico (Welsh & Culler, 2001). Para mejorar la concurrencia en aplicaciones, tecnologías como Spring Boot y microservicios son herramientas clave que permiten una respuesta ágil y eficiente en el manejo de grandes cantidades de datos y usuarios simultáneos.

El aplicativo MD hace parte de un ecosistema denominado Hospital Information System (HIS) o Laboratory Information System (LIS), el cual abarca la operación de salud de la compañía, donde se encuentran procesos como el agendamiento de cita, manejo de medicamento, historia clínica, entre otros dentro de este ecosistema. En este sentido, MD se integra con otros sistemas como Institución Prestadora de Salud @ (IPSA) el cual, es el encargado de almacenar la historia clínica de los pacientes.

### 3.2. Proceso de notificaciones: Mensajería Dinámica (MD) – Institución Prestadora de Salud @ (IPSA)

La comunicación entre los aplicativos de MD e IPSA está definida a través de la tecnología de *message broker* RabbitMQ, el cual sirve como intermediario para gestionar colas de mensajes, permitiendo la comunicación asíncrona entre las aplicaciones, en las cuales el emisor se encuentra desacoplado del receptor, ya que no requiere conocerlo, por lo que es una tecnología flexible y escalable (RabbitMQ, s.f.).

Para el proceso de notificación entre MD e IPSA se cuenta con dos colas: la primera, se encarga de notificar los resultados de exámenes médicos de MD a IPSA; la segunda, se utiliza para recibir un mensaje de confirmación, indicando que los resultados se recibieron satisfactoriamente, o si se presentó alguna novedad en el proceso, caso en el cual, se incluye un mensaje descriptivo del error facilitando su análisis.

Es de aclarar que, el aplicativo MD se encuentra en un proceso de renovación tecnológica, lo que implica la coexistencia de dos flujos de trabajo encargados de la notificación de resultados a IPSA, que manejan diferentes recursos en Base de Datos, así como formatos de mensajes y

responsabilidades. Actualmente, el flujo con más antigüedad soporta aproximadamente el 91% de la carga de trabajo y gradualmente se está realizando la migración de pruebas de laboratorio al nuevo flujo, lo cual resalta la importancia de abordar ambos escenarios al momento de intervenir el proceso de notificación.

En este sentido, el envío de resultados es susceptible a errores, los cuales pueden estar asociados a diferentes causas y deben ser intervenidos para mejorar la calidad del servicio que se brinda, tanto a las entidades prestadoras de salud, como a los pacientes, los cuales dependen de contar oportunamente con los resultados de exámenes médicos para una adecuada atención.

### 3.3. Soporte a Incidentes

La gestión de incidentes es un proceso crucial en cualquier ecosistema tecnológico que busca mantener la estabilidad y la continuidad operativa. En el caso de Mensajería Dinámica, la gestión se realiza mediante una plataforma de *Service Desk*, software especializado en gestión de solicitud o evento reportados por un usuario, en la cual, se documentan los casos, se adjuntan evidencias y se escalan a los equipos pertinentes. Según ITIL (IT Infrastructure Library), la gestión efectiva de incidentes debe incluir, tanto la respuesta oportuna, como el análisis de *Causas Raíz*, las cuales son el origen del inconveniente, para la implementación de soluciones preventivas (Iden & Eikebrokk, 2014).

En este caso, los incidentes recurrentes se agrupan y documentan como casos *Raizales* en la plataforma Azure DevOps, la cual brinda herramientas para manejar un enfoque colaborativo que integra desarrollo de software y operaciones para automatizar procesos, mejorar la calidad del software y acelerar la entrega continua (Bass et al., 2015).

Desde el rol de practicante y como aspirante al título de Ingeniero de sistemas, se puede contribuir significativamente, aplicando los conocimientos adquiridos en el proceso de formación académica. Con un enfoque en el desarrollo de software, se cuenta con conocimientos en programación orientada a objetos y experiencia en el lenguaje Java, pudiendo participar en la identificación de *Causas Raíz* de los incidentes y en la implementación de soluciones tecnológicas sostenibles.

De igual manera, gracias al manejo de bases de datos y lenguaje SQL, es posible comprender el modelo de datos y darle una gestión eficiente, pudiendo aportar en la mejora de la

funcionalidad y disponibilidad de las plataformas. Por su parte, el conocimiento en metodologías ágiles y DevOps permite adaptarse a las formas de trabajo, lo que facilita la colaboración eficaz en mesas de trabajo y documentación de procedimientos, asegurando un enfoque sistemático en la resolución de problemas técnicos.

## 4. Metodología

El proyecto se desarrolló con un enfoque cualitativo basado en el análisis de contenido y estudio de casos, permitiendo organizar la información generada en Mensajería Dinámica e identificar patrones de error recurrentes. A partir de estos hallazgos, se implementaron soluciones para mejorar la estabilidad del sistema.

El trabajo se estructuró en tres fases principales:

1. **Contextualización y Diagnóstico:**
    - Familiarización con los procesos internos de Ayudas Diagnósticas.
    - Conocimiento de la arquitectura de la aplicación y su integración con IPSA.
    - Identificación de incidentes recurrentes mediante el análisis de logs y reportes en la plataforma de service desk.
2. **Análisis y Priorización de Errores:**
    - Clasificación de errores: técnicos y de negocio.
    - Priorización de errores más críticos según su impacto en la operación.
    - Identificación de patrones comunes en la falla del sistema.
3. **Implementación de Soluciones y Validación:**
    - Desarrollo y despliegue de mejoras en la configuración del sistema.
    - Implementación de validaciones para mitigar errores recurrentes.
    - Evaluación del impacto de las mejoras mediante pruebas y monitoreo continuo.
    - Elaboración de informes y recomendaciones para la estabilización continua del proceso.

## 5. Análisis de resultados

El proceso de respuesta realizado por IPSA a través de la cola de mensajería Rabbit almacena el historial de mensajes en base de datos, utilizando diferentes tablas para los dos flujos que coexisten. Para el análisis de mensajes de error se abarcaron ambos escenarios y se consolidaron para su análisis. Los mensajes de error se agruparon en dos categorías: *Errores Técnicos*, asociados al funcionamiento de la aplicación y *Errores de Negocio*, asociados a inconsistencias en la configuración de los aplicativos.

Para el análisis de los errores se tomó como muestra la información correspondiente a los meses de diciembre del 2024 y enero del 2025, periodo en el cual se reportaron más de tres millones de resultados a IPSA, es decir, un promedio de 1,5 millones de resultados al mes. De esta muestra, se identificó que aproximadamente el 0,67% de los exámenes presentaron un mensaje de error en el proceso de notificación, lo cual a pesar de parecer un porcentaje no representativo, se tradujo en más de 10.000 notificaciones no exitosas a la aplicación de historia clínica, que, a su vez, representa en promedio a 8.800 pacientes con afectación en el servicio por mes.

Con lo anterior, la *Tabla 1* da cuenta de la cantidad de errores y su porcentaje asociado con respecto al total de notificaciones no exitosas:

**Tabla 1**
*Muestra de errores, agrupación por categorías y porcentajes asociados*

| Tipo de error | Cantidad | Porcentaje |
|---|---|---|
| Negocio | 12329 | 51,2% |
| Técnico | 11669 | 48,5% |
| no clasificado | 71 | 0,3% |
| Total | 24069 | 100,0% |

Fuente: Elaboración propia.

Con ello, se tiene que, para los *Errores Técnicos*, el 89.9% de ellos están comprendidos en dos casos:

1. Errores asociados a cambio de documento de identidad: el cual ocasiona que los resultados no puedan ser asociados a la orden en IPSA por inconsistencia en los datos del paciente, teniendo presente que, una orden puede permanecer varias semanas a la espera de resultados, por lo que puede presentarse que el paciente realice cambio en su documento de identidad en este periodo, ocasionando que la orden sea creada con un documento y al

cargar los resultados ya cuente con uno diferente (ello aplica, tanto para tipo de documento, como para número de documento). Este tipo de notificación agrupa el mayor volumen de *Errores Técnicos* con un 86,97%.

Desde el área *Beyond Health* (BH) se registran las actualizaciones de los pacientes de la compañía, incluyendo los cambios de documento, estos cambios se comunican a través del *message broker* RabbitMQ. Para dar solución, desde Mensajería Dinámica, se implementó una nueva funcionalidad que recibe los mensajes publicados en esta cola, con el fin de almacenarlos en una nueva tabla en base de datos para validar los posibles cambios que puedan afectar la entrega de resultados a IPSA y realizar el reenvío de las órdenes utilizado la nueva identificación.

Según la documentación de la cola se creó una nueva tabla en base de datos para almacenar los mensajes correspondientes a cambios de documento de identidad, estos incluyen el tipo y número de documento previo, tipo y número de documento nuevo; datos de auditoría como fecha del cambio, fecha de radicación, tipo de operación, origen del cambio y secuencia.

**Figura 1**

*Script SQL CREATE TABLE TSIR_CAMBIOSIDENTIFICACION*

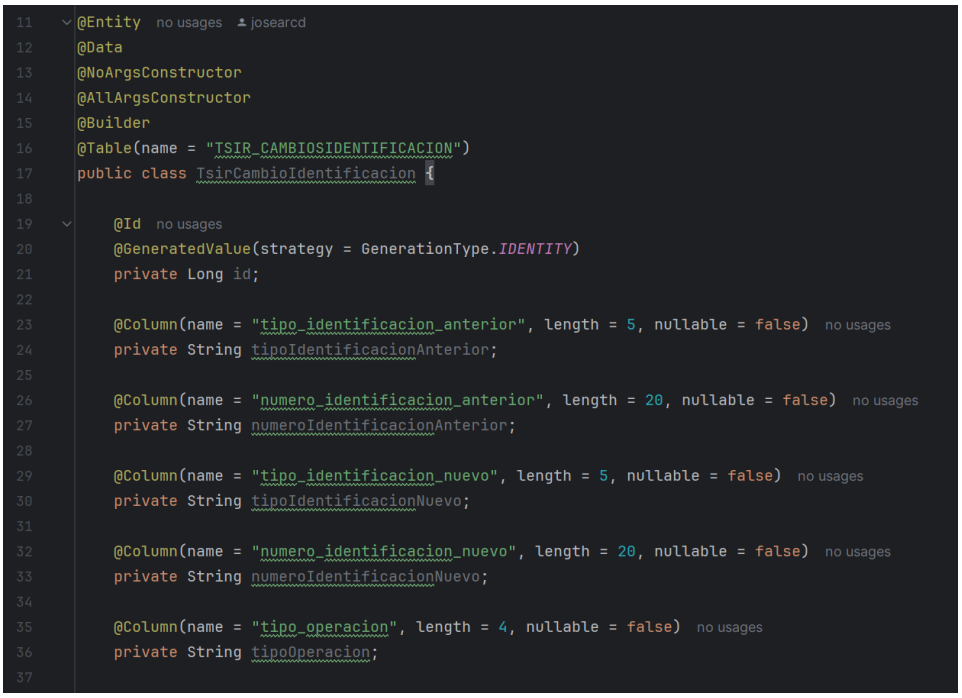
```sql
CREATE TABLE DB_DINAMICA_COMUN.dbo.TSIR_CAMBIOSIDENTIFICACION (
    id INT IDENTITY(1,1) PRIMARY KEY,
    tipo_identificacion_anterior VARCHAR(2),
    numero_identificacion_anterior VARCHAR(60),
    tipo_identificacion_nuevo VARCHAR(2),
    numero_identificacion_nuevo VARCHAR(60),
    tipo_operacion VARCHAR(2),
    secuencia VARCHAR(9),
    origen VARCHAR(5),
    fecha_cambio DATETIME,
    fecha_radicacion DATETIME
);
```

*Nota.* Base de Datos MD (SURA, 2025).

A su vez para facilitar el acceso y el manejo de los datos se utiliza el patrón de diseño *Data Access Object* (DAO), con el cual se busca facilitar el acceso a la base de datos a través de un *Data Transfer Object* (DTO), el cual se utiliza para mapear la información de la base de datos a un objeto el cual pueda interactuar en la lógica del negocio.

**Figura 2**

*Implementación DTO TSIR_CAMBIOIDENTIFICACION*



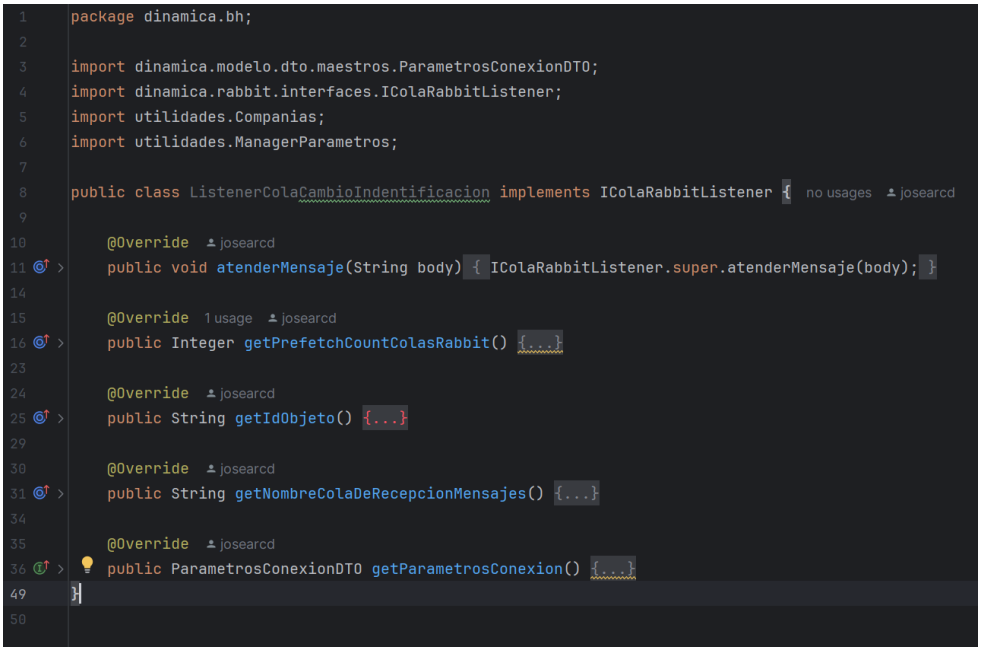
```java
@Entity  no usages  josearcd
@Data
@NoArgsConstructor
@AllArgsConstructor
@Builder
@Table(name = "TSIR_CAMBIOSIDENTIFICACION")
public class TsirCambioIdentificacion {

    @Id  no usages
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    @Column(name = "tipo_identificacion_anterior", length = 5, nullable = false)  no usages
    private String tipoIdentificacionAnterior;

    @Column(name = "numero_identificacion_anterior", length = 20, nullable = false)  no usages
    private String numeroIdentificacionAnterior;

    @Column(name = "tipo_identificacion_nuevo", length = 5, nullable = false)  no usages
    private String tipoIdentificacionNuevo;

    @Column(name = "numero_identificacion_nuevo", length = 20, nullable = false)  no usages
    private String numeroIdentificacionNuevo;

    @Column(name = "tipo_operacion", length = 4, nullable = false)  no usages
    private String tipoOperacion;

```

*Nota.* Fuente Repositorio MD (SURA, 2025).

Para poder recibir los mensajes publicados en el *bróker message*, se implementó la lógica encargada de escuchar y procesar la información que se reporta desde BH, utilizando una clase listener que implementa el framework IColaRabbitListener, el cual facilita el uso de esta funcionalidad, definiendo un marco de referencias que especifica las mínimas características que esta clase debe tener para integrarse con el software de MD.

**Figura 3**

*Implementación ListenerColaCambioIdentificacion*



```
package dinamica.bh;

import dinamica.modelo.dto.maestros.ParametrosConexionDTO;
import dinamica.rabbit.interfaces.IColaRabbitListener;
import utilidades.Companias;
import utilidades.ManagerParametros;

public class ListenerColaCambioIndentificacion implements IColaRabbitListener { no usages  josearcd

    @Override  josearcd
    public void atenderMensaje(String body) { IColaRabbitListener.super.atenderMensaje(body); }

    @Override  1 usage  josearcd
    public Integer getPrefetchCountColasRabbit() {...}

    @Override  josearcd
    public String getIdObjeto() {...}

    @Override  josearcd
    public String getNombreColaDeRecepcionMensajes() {...}

    @Override  josearcd
    public ParametrosConexionDTO getParametrosConexion() {...}
}
```

*Nota.* Fuente Repositorio MD (SURA, 2025).

Con la implementación de esta mejora se obtuvo que, los cambios de Registro Civil (RC) a Tarjeta de Identidad (TI) y de Tarjeta de Identidad a Cédula de Ciudadanía (CC), representan el 92% de los errores por cambio en el documento de identidad, luego de la implementación de la solución, los errores asociados no han presentado nuevas ocurrencias, dando como resultado una efectividad del 92% y hallando solución al 40,1% del total de los errores asociados a la notificación de resultados a IPSA.

2. Errores asociados a la disponibilidad del sistema: indican inconvenientes que no dependan de la estructura de los mensajes o los datos asociados, estos se ven afectados por problemas de comunicación, disponibilidad de servicios o infraestructura de las aplicaciones y representan el 2,97% de los errores técnicos.

   Para su intervención, se construyó un nuevo control, el cual, al momento de procesar los mensajes de respuesta, si el mensaje está dentro un listado específico, la notificación de resultado a IPSA será encolada nuevamente para su reenvío, limitando el número máximo de reenvíos con un parámetro y así también evitar procesos que se repitan indefinidamente. Dentro de la lógica que se encarga de procesar los mensajes enviados por IPSA, se incluyó una nueva condición que válida el contenido del mensaje, donde si el mensaje está dentro

un listado de errores conocidos, este debe será marcado para ser reenvía en la siguiente carga de notificación de resultados a IPSA, adicionalmente se considera un máximo de intentos posibles, donde al superarlos, el registro será marcado como un error.

**Figura 4**

*Implementación filtro para reenvío de órdenes IPSA*

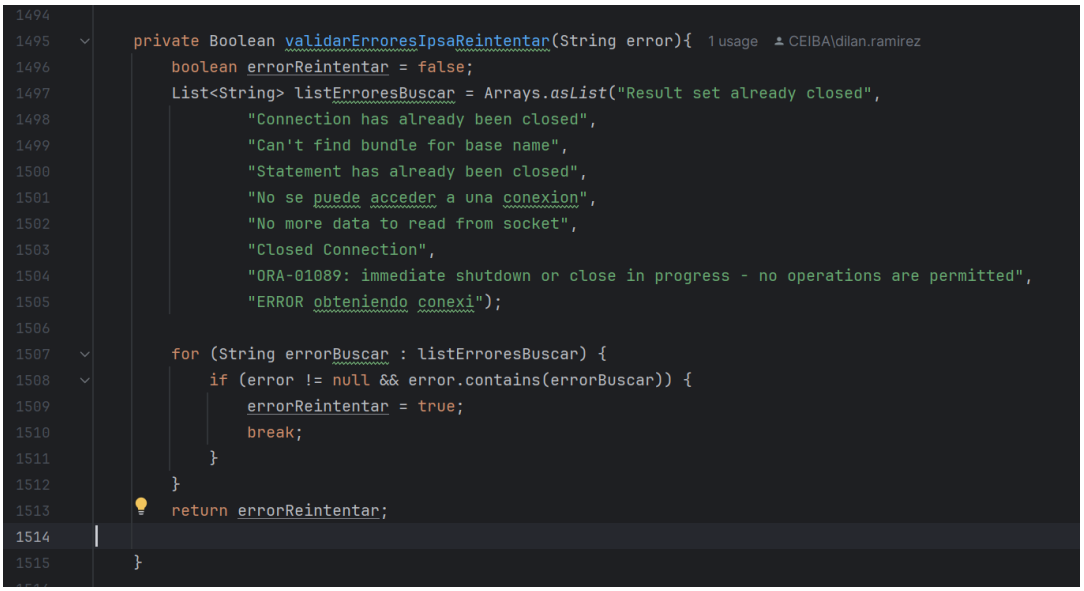


```
155            if ("N".equalsIgnoreCase(respIpsa.getStatus())) {
156                if(validarErroresIpsaReintentar(respIpsa.getMsgError()) && notif.getIntentos()<MAXIMOS_INTENTOS){
157                    notif.setFechaEnviado(null);
158                    notif.setEnviado("N");
159                }
160                else{
161                    notif.setEnviado("E");
162                }
163
```

*Nota.* Fuente Repositorio MD (SURA, 2025).

**Figura 5**

*Implementación Función validadErroresIpsaReintentar*



```
1494
1495    private Boolean validarErroresIpsaReintentar(String error){  1 usage  CEIBA\dilan.ramirez
1496        boolean errorReintentar = false;
1497        List<String> listErroresBuscar = Arrays.asList("Result set already closed",
1498                "Connection has already been closed",
1499                "Can't find bundle for base name",
1500                "Statement has already been closed",
1501                "No se puede acceder a una conexion",
1502                "No more data to read from socket",
1503                "Closed Connection",
1504                "ORA-01089: immediate shutdown or close in progress - no operations are permitted",
1505                "ERROR obteniendo conexi");
1506
1507        for (String errorBuscar : listErroresBuscar) {
1508            if (error != null && error.contains(errorBuscar)) {
1509                errorReintentar = true;
1510                break;
1511            }
1512        }
1513        return errorReintentar;
1514
1515    }
```

*Nota.* Fuente Repositorio MD (SURA, 2025).

La implementación del control en el reenvío de resultados mostró una efectividad del 98,3%, lo cual es un resultado satisfactorio para el escenario intervenido, dejando como remanente los casos que están asociados a caídas prolongadas en los servicios de la infraestructura o de la aplicación que generar una afectación mayor, en ocasiones asociadas a los mismos procesos de mantenimiento del sistema.

3. Otros errores técnicos: representan el 10,06% de los mensajes de error, estos no fueron intervenidos en el desarrollo de este proyecto, se planea asignar recursos para trabajar en estos casos en una segunda etapa, ya que requiere un análisis más detallado y un trabajo en conjunto con el personal de IPSA para identificar las particularidades técnicas que se presentan desde el sistema externo.

Para la clasificación de *Errores de Negocio*, se identificaron dos causas, las cuales se abordaron conjuntamente en mesas de trabajo con el área de Ayudas Diagnósticas encargada de la operación del negocio. En estas, se identificaron los responsables de validar la configuración de los diferentes sistemas y realizar las correcciones, para garantizar que se tenga coherencia en la comunicación, de tal forma que las automatizaciones e integraciones puedan ejecutarse sin novedad, garantizando la gestión de las órdenes, sus exámenes asociados, así como los resultados.

Los Errores de Negocio identificados se describen a continuación:

1. Errores asociados a configuración faltante: en los cuales, la parametrización del sistema está incompleta, debido a la alta complejidad del negocio y los constantes cambios asociados a las restricciones por tipos de convenios, instituciones asociadas y volumen en los tipos de exámenes disponibles.
2. Errores asociados a diferencias en los códigos de los servicios entre los sistemas: este se presenta por imprecisiones en digitación o de configuración para las diferentes instituciones que remiten exámenes para ser procesados por Ayudas Diagnósticas, donde al notificar resultados a IPSA, el destino no reconoce el código del examen y no es posible guardar en la historia clínica del paciente los resultados. Este error se genera desde la plataforma OPC, al momento de ingresar la orden.

Para medir la efectividad de las soluciones, se realizó una nueva medición de los errores registrados en la cola de respuestas en un periodo de 30 días, estos valores se compararon con las medias de errores por mes obtenidas de la primera muestra con la cual se realizó el análisis de los tipos de error. A continuación, se listan los valores obtenidos.

**Tabla 2**
Comparación de mediciones previa y posterior a implementación.

| Tipo de error | Errores x mes* | Porcentaje de error | Errores pos-implementación* | Porcentaje de efectividad* |
|---|---|---|---|---|
| Negocio | 6165 | 51,2% | 2179 | 65% |
| Técnico | 5835 | 48,5% | 981 | 83% |
| No clasificado | 36 | 0,3% | 41 | -15% |
| Total | 12035 | 100,0% | 3201 | 73% |

Fuente: Elaboración propia.
*Errores x mes: media de errores por mes antes de la implementación de las soluciones.

*Errores pos-implementación: errores registrados en un periodo de 30 días, luego de la implementación.

*Porcentaje de efectividad: la razón entre Errores x mes/Errores pos-implementación.

Según las mediciones, se obtuvo una efectividad general del 73%, un 83% para los errores asociados a factores técnicos y un 65% en errores del negocio. Para los errores no clasificados, no se realizaron acciones, adicionalmente su volumen es despreciable, por lo cual, su variación no afecta los resultados obtenidos.

## 6. Conclusiones y recomendaciones

El proceso de estabilización y mejora del sistema de notificación de resultados médicos en Mensajería Dinámica e IPSA permitió identificar y mitigar múltiples errores que afectaban la disponibilidad y precisión de la información. A partir de un análisis detallado de las incidencias, se estableció un plan de acción efectivo, priorizando las fallas con mayor impacto en la operación.

Entre los principales logros del proyecto se destacan:

- La clasificación estructurada de los errores en técnicos y de negocio, facilitando su identificación y tratamiento.
- La implementación de soluciones para los errores más críticos, como el cambio de documento de identidad y la disponibilidad del sistema, reduciendo la tasa de fallos en más del 90 %.
- La estabilización de la comunicación entre MD e IPSA mediante la actualización de procesos y mejoras en la gestión de colas de mensajería.

A pesar de estos avances, aún se requiere un refinamiento en las soluciones implementadas para aumentar su efectividad, particularmente en el manejo de cambios de documento, donde persisten variaciones del problema que no han sido completamente corregidas. Además, existen errores de menor frecuencia que requieren intervención. Se recomienda continuar con un monitoreo constante del sistema y desarrollar una segunda fase del proyecto enfocada en la mejora de la parametrización de los sistemas y la estandarización de códigos entre plataformas.

Desde el punto de vista tecnológico, este proyecto representó un desafío significativo, ya que involucró diversas tecnologías para la identificación y resolución de errores. La utilización de Java y el framework Spring Boot permitió implementar soluciones eficientes, aunque la versión utilizada de Java ya ha sido reemplazada por versiones más recientes. La integración con RabbitMQ fue clave para la comunicación entre los sistemas, destacando la importancia de los message brokers en arquitecturas distribuidas. Además, el uso de SQL Server optimizó la gestión de datos, garantizando la trazabilidad de los errores y su análisis detallado. Estas herramientas, en conjunto con WebLogic, proporcionaron un entorno robusto para la implementación de mejoras en la plataforma.

El éxito de este proyecto resalta la importancia de una gestión proactiva de incidentes y la necesidad de estrategias de mejora continua en la integración de sistemas tecnológicos en el sector salud.

Desde una perspectiva personal, el proyecto y el proceso de prácticas permitieron explorar un entorno laboral desafiante y experimentar los distintos retos asociados a una compañía de gran envergadura como Suramericana S.A. Más allá del aspecto técnico, se trabajó en conjunto con profesionales de diversas áreas, particularmente en el sector salud, resaltando la interdisciplinariedad y la transversalidad de los sistemas. Esto reafirma la importancia de ser flexible y tener un alto nivel de adaptabilidad en entornos tecnológicos complejos.

## Referencias

Bass, L., Weber, I., & Zhu, L. (2015). DevOps: A Software Architect's Perspective. Addison-Wesley Professional.

Google. (s.f.). *Documentación de Angular v5*. Angular.io. https://v5.angular.io/docs

Iden, J., & Eikebrokk, T. R. (2014). Using the ITIL process reference model for realizing IT governance: An empirical investigation. *Information Systems Management*, 31(1), 37-58. https://doi.org/10.1080/10580530.2014.854025

Microsoft. (s.f.). *SQL Server 2016*. Microsoft. https:// www.microsoft.com/ es-co/ sql-server/sql-server-2016

Spring. (s.f.). *¿Por qué Spring?* Spring.io. https://spring.io/why-spring

Oracle. (s.f.). ¿Qué es Java? Java.com. https:// www.java.com/ es/ download/ help/ whatis_java.html

Oracle. (s.f.). *WebLogic Server: Installers & Downloads*. Oracle. https:// www.oracle.com/middleware/technologies/weblogic-server-installers-downloads.html

Pivotal Software, Inc. (s.f.). RabbitMQ - Messaging that just works. RabbitMQ. https://www.rabbitmq.com/

Welsh, M., & Culler, D. E. (2001). *An Architecture for Highly Concurrent, Well-Conditioned Internet Services*. In Proceedings of the 8th Workshop on Hot Topics in Operating Systems (pp. 113-118). IEEE. https://doi.org/10.1109/HOTOS.2001.990068

# Anexos

## Anexo 1: Poster

Departamento de Ingeniería de Sistemas

### Solución de errores frecuentes de las plataformas de software de Ayudas Diagnósticas Suramericana S.A.

UNIVERSIDAD DE ANTIOQUIA

Facultad de Ingeniería

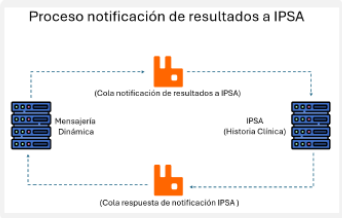
PRACTICANTE: Jose Franco Arroyave Cardona

ASESORES: Deisy Loaiza Berrío, Juan Pablo Palacio Guzmán

PROGRAMA: Ingeniería de Sistemas

Semestre de la práctica: 2024-2

El presente proyecto se enmarca en el proceso de prácticas profesionales del programa de Ingeniería de Sistemas de la Universidad de Antioquia.
En este sentido se desarrollo con base en la experiencia obtenida en la compañía Suramericana S.A. durante un periodo de seis meses. En los cuales se acompaño el área de Ayudas Diagnósticas desde el rol de Analista de Operaciones.




Ayudas Diagnósticas cuenta con el software Mensajería Dinámica, el cual se encarga de la gestión de órdenes que contienen un conjunto de exámenes médicos. Esta se integra con proveedores que permiten la comunicación entre los equipos analizadores de laboratorio y los sistemas de información.
Posteriormente, se encarga de la disposición de los resultados hacia los diferentes actores que puedan requerirlos, ya sean pacientes, médicos u otros sistemas de información.

### Ayudas Diagnosticas



### Resumen

Este proyecto aborda el proceso de estabilización y mejora en la notificación de resultados médicos entre las plataformas Mensajería Dinámica e IPSA para Ayudas Diagnósticas. La infraestructura de notificación, basada en la tecnología de mensajería RabbitMQ, experimenta errores recurrentes que afectan la entrega oportuna de resultados.
Para abordar esta problemática, se realizó un análisis detallado de los errores, clasificándolos en errores técnicos y de negocio. Se establecieron estrategias para abordar los incidentes, priorizando aquellos con mayor impacto. La implementación de soluciones técnicas redujo significativamente la tasa de errores, mejorando la comunicación entre las plataformas y la correcta disposición de resultados.

### Metodología

El proyecto se desarrolló con un enfoque cualitativo basado en el análisis de contenido y estudio de casos, permitiendo organizar la información generada en Mensajería Dinámica e identificar patrones de error recurrentes. A partir de estos hallazgos, se implementaron soluciones para mejorar la estabilidad del sistema.

### Resultados

Los mensajes de error se agruparon en dos categorías: Errores Técnicos y Errores de Negocio. Con un promedio de 1.5 millones de exámenes al mes, el 0,67% de estos presentaban errores al notificarse, más de 10.000 notificaciones fallidas a IPSA, que se traduce en 8.800 pacientes.
Errores Técnicos:

1. Errores por cambio de documento de identidad (86,97%): Para su intervención, se comenzó a capturar las notificaciones de cambio de documento realizadas por el área Beyond Health, las cuales se hacen a través de un message bróker, esto con el fin de validar si estos pudieron afectar la entrega de resultados a IPSA y realizar el reenvío de los resultados utilizado la nueva identificación.
2. Errores asociados a la disponibilidad del sistema (2,97%): Para este, se implementó un control, el cual, realiza el reenvío de las notificaciones para ciertos tipos de error que esta asociados a disponibilidad del sistema.

Errores de Negocio: Para su manejo se realizaron mesas de trabajo con el área de Ayudas Diagnósticas encargada de la operación del negocio. En estas, se identificaron los responsables de validar la configuración de los diferentes sistemas y realizar correcciones, para garantizar que se tenga coherencia en la parametrización que rigue la comunicación entre los diferentes aplicativos.

Más información sobre el proyecto




Comparación de mediciones previa y posterior a implementación.

| Tipo de error | Errores x mes | Porcentaje de error | Errores pos-implementación | Porcentaje de efectividad |
|---|---|---|---|---|
| Negocio | 6165 | 51,20% | 2179 | 65% |
| Técnico | 5835 | 48,50% | 981 | 83% |
| No clasificado | 36 | 0,30% | 41 | -15% |
| Total | 12035 | 100,00% | 3201 | 73% |

### Objetivos

Reducir la recurrencia de incidentes en las plataformas de Ayudas Diagnósticas de Suramericana S.A., mejorando la disponibilidad de los sistemas y la satisfacción de los usuarios finales.

- Identificar las principales causas de los incidentes recurrentes en las plataformas de Ayudas Diagnósticas.
- Analizar soluciones tecnológicas en las mesas de trabajo, que aborden los incidentes más críticos y frecuentes.
- Implementar soluciones raíz empleando Java, Weblogic y SQL server, según se encuentre oportuno, de acuerdo con los casos identificados, incidiendo en los problemas técnicos que afectan la operación de las plataformas.
- Documentar los procedimientos empleados que dan solución a los incidentes, permitiendo una sistematización del proceso.

### Conclusiones

- Se estructuraron errores en categorías técnicas y de negocio, facilitando su análisis y priorización, lo cual optimizó el tratamiento de incidencias con mayor impacto.
- La estabilización del sistema de notificación médica permitió identificar y reducir errores críticos, mejorando notablemente la precisión, disponibilidad y confiabilidad de la información entregada.
- La tasa de fallos disminuyó más del 90 % gracias a soluciones puntuales, como la mejora en validación de documentos y disponibilidad del sistema
- Esta experiencia permitió aplicar conocimientos en un entorno real, destacando el valor de la interdisciplinariedad, la adaptabilidad y la gestión proactiva de incidentes tecnológicos.

DATOS DE CONTACTO DEL AUTOR:

3105101970 +57 3105101970 Franco.arroyave@udea.edu.co https://www.linkedin.com/in/josefranco-ac/