



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Sistemas

**Desarrollo de un sistema web para la supervisión de
afiliaciones indebidas en una entidad de salud**

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el Título Profesional de Ingeniero de Sistemas

AUTOR

Albino Antonio DE LA CRUZ HUARINGA

ASESOR

Mg. César Augusto ALCÁNTARA LOAYZA

Lima, Perú

2026



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

De la Cruz, A. (2026). *Desarrollo de un sistema web para la supervisión de afiliaciones indebidas en una entidad de salud*. [Trabajo de suficiencia profesional, de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Sistemas]. Repositorio institucional Cybertesis UNMSM.

Metadatos complementarios

Datos de autor	
Nombres y apellidos	Albino Antonio De La Cruz Huaranga
Tipo de documento de identidad	DNI
Número de documento de identidad	09501175
URL de ORCID	https://orcid.org/0009-0008-1477-533X
Datos de asesor	
Nombres y apellidos	César Augusto Alcántara Loayza
Tipo de documento de identidad	DNI
Número de documento de identidad	09132297
URL de ORCID	https://orcid.org/0000-0003-3435-4555
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Augusto Parcemon Cortez Vásquez
Tipo de documento	DNI
Número de documento de identidad	08634618
Miembro del jurado 1	
Nombres y apellidos	Jorge Raúl Díaz Muñante
Tipo de documento	DNI
Número de documento de identidad	07216161
Datos de investigación	
Línea de investigación	No aplica
Grupo de investigación	No aplica
Agencia de financiamiento	Sin financiamiento

Ubicación geográfica de la investigación	País: Perú Departamento: Lima
	Provincia: Lima Distrito: Cercado de Lima Jr. Carlos Amezaga No. 375 Universidad Nacional Mayor de San Marcos Latitud: -12.0564232 Longitud: -77.0843327
Año o rango de años en que se realizó la investigación	2025
URL de disciplinas OCDE	2.02.04 -- Ingeniería de sistemas y comunicaciones https://purl.org/pe-repo/ocde/ford#2.02.04



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
Escuela Profesional de Ingeniería de Sistemas

Acta Virtual de Sustentación
del Trabajo de Suficiencia Profesional

Siendo las 19:00 horas del día 30 de marzo del año 2026, se reunieron virtualmente los docentes designados como Miembros de Jurado del Trabajo de Suficiencia Profesional, presidido por el Mg. Cortez Vásquez Augusto Parcemon (Presidente), Mg. Díaz Muñante Jorge Raúl (Miembro) y el Mg. Alcántara Loayza César Augusto (Miembro Asesor), usando la plataforma Meet (<http://meet.google.com/wcc-jfha-tms>), para la sustentación virtual del Trabajo de Suficiencia Profesional intitulado: **“DESARROLLO DE UN SISTEMA WEB PARA LA SUPERVISIÓN DE AFILIACIONES INDEBIDAS EN UNA ENTIDAD DE SALUD”**, por el Bachiller **De La Cruz Huaranga Albino Antonio**; para obtener el Título Profesional de Ingeniero de Sistemas.

Acto seguido de la exposición del Trabajo de Suficiencia Profesional, el Presidente invitó al Bachiller a dar las respuestas a las preguntas establecidas por los miembros del Jurado.

El Bachiller en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez a las observaciones y preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros del Jurado, el Bachiller obtuvo la nota de **17 (DIECISIETE)**.

A continuación el Presidente de Jurados el Mg. Cortez Vásquez Augusto Parcemon, declara al Bachiller **Ingeniero de Sistemas**.

Siendo las 19:55 horas, se levantó la sesión.

Presidente

Mg. Cortez Vásquez Augusto Parcemon

Miembro

Mg. Díaz Muñante Jorge Raúl

Miembro Asesor

Mg. Alcántara Loayza César Augusto

CERTIFICADO DE SIMILITUD

Yo, César Augusto Alcántara Loayza en mi condición de asesor acreditado mediante Resolución Decanal N° 001268-2025-D-FISI/UNMSM y OFICIO N° 002-PTTSP-FISI-2025 del Trabajo de Suficiencia Profesional, cuyo título es Desarrollo de un sistema web para la supervisión de afiliaciones indebidas en una entidad de salud, presentado por el bachiller De La Cruz Huaranga Albino Antonio para optar el Título Profesional de Ingeniero de Sistemas, CERTIFICO que se ha cumplido con lo establecido en la Directiva de Originalidad y de Similitud de Trabajos Académicos, de Investigación y Producción Intelectual. Según la revisión, análisis y evaluación mediante la plataforma PIDS, el documento evaluado cuenta con el 6% de similitud, nivel PERMITIDO para continuar con los trámites correspondientes y para su **publicación en el repositorio institucional**.

Se emite el presente certificado en cumplimiento de lo establecido en las normas vigentes, como uno de los requisitos para la obtención del **TITULO PROFESIONAL DE INGENIERO DE SISTEMAS**.

Lima, 4 de junio de 2026



Firma del Asesor:

Nombres y Apellidos del asesor: César Augusto Alcántara Loayza

DNI: 09132297

FICHA CATALOGRÁFICA

DESARROLLO DE UN SISTEMA WEB PARA LA SUPERVISIÓN DE AFILIACIONES INDEBIDAS EN UNA ENTIDAD DE SALUD

AUTOR: DE LA CRUZ HUARINGA ALBINO ANTONIO

ASESOR: ALCÁNTARA LOAYZA CÉSAR AUGUSTO

LIMA – PERÚ, 2025

TÍTULO PROFESIONAL: Título Profesional de Ingeniero de Sistemas.

ÁREA / PROGRAMA / LÍNEA DE INVESTIGACIÓN:

Ingenierías / Tecnología de Información y Comunicación / Ingeniería de Software

PREGRADO: Universidad Nacional Mayor de San Marcos - Facultad de Ingeniería de Sistemas e Informática - Escuela profesional de Ingeniería de Sistemas

Formato 21 x 29.7 cm

Páginas: xii, 96

DEDICATORIA

A mi madre, por su dedicación incansable y su apoyo constante, que fueron la fuerza fundamental para alcanzar cada meta y superar cada obstáculo durante este proceso. Su ejemplo de perseverancia ha sido mi inspiración diaria.

AGRADECIMIENTOS

Al Mg. César Augusto Alcántara Loayza por su orientación y apoyo, que fueron fundamentales para el logro de los objetivos de este trabajo.
Y por encima de todo doy gracias a Dios.

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

**Desarrollo de un sistema web para la supervisión de afiliaciones
indebidas en una entidad de salud**

Autor: De La Cruz Huaranga, Albino Antonio
Asesor: Alcántara Loayza, César Augusto
Título: Trabajo de Suficiencia Profesional para optar el Título Profesional de
Ingeniero de Sistemas
Fecha: Setiembre de 2025

RESUMEN

El presente trabajo de suficiencia profesional describe el desarrollo del Sistema de Supervisión de Afiliaciones Indebidas en Salud (SSAIS), orientado a apoyar y optimizar dicho proceso de supervisión en una entidad prestadora de servicios en salud.

En el contexto previo a la solución propuesta, implementación del SSAIS, el proceso se realizaba de forma predominante manual, la información se encontraba dispersa en distintas fuentes tanto internas como externas. Esta situación hacía más complejo el seguimiento de los casos, dificultaba la trazabilidad, y extendía los tiempos para la asignación y análisis de los expedientes.

A demás, la información de los afiliados con diagnósticos de alto costo, es decir, el universo de casos, no estaba disponible de manera inmediata, ya que para su obtención se dependía de la coordinación con el área técnica de base de datos, quien debía preparar la información de acuerdo a los requerimientos de cada solicitud. Dicha situación dificultaba contar con la información de manera oportuna ya que hay plazos que se deben cumplir.

El sistema optimiza la supervisión a través de sus distintas funcionalidades, permitiendo consultar de manera inmediata y organizada a los diagnósticos de alto costo, así como a la información de empleados, empleadores y datos asistenciales. Así mismo, el sistema ofrece opciones para la asignación de casos a los distintos roles y el seguimiento de estos dentro del proceso de supervisión, lo que contribuye a mejorar la trazabilidad de los expedientes.

En consecuencia, el SSAIS contribuye a optimizar el proceso de supervisión de afiliaciones indebidas, facilitando la integración de la información, la asignación y seguimiento de casos, mejorando la trazabilidad de los expedientes y la sostenibilidad financiera de la entidad.

Palabras clave: Sistema de Supervisión de Afiliaciones Indebidas en Salud, supervisión, afiliación indebida, seguimiento, trazabilidad.

**MAJOR NATIONAL UNIVERSITY OF SAN MARCOS
FACULTY OF SYSTEMS AND INFORMATICS ENGINEERING
PROFESSIONAL SCHOOL OF SYSTEMS ENGINEERING**

**Development of a web-based system for monitoring improper affiliations in a
healthcare organization**

Author: De la Cruz Huaranga, Albino Antonio

Advisor: Alcántara Loayza, César Augusto

Title: Professional Sufficiency Work to opt for the Professional Title of Systems
Engineer

Date: September 2025

ABSTRACT

This professional proficiency project describes the development of the System for the Supervision of Improper Health Insurance Enrollment (SSAIS), designed to support and optimize the supervision process at a healthcare provider.

Prior to the proposed solution, implementation of the SSAIS, the process was predominantly manual, with information scattered across various internal and external sources. This situation made it more complex to follow up on cases, hindered traceability, and extended the time required for the assignment and analysis of files.

In addition, information on members with high-cost diagnoses, i.e., the universe of cases, was not immediately available, as obtaining it depended on coordination with the technical database area, which had to prepare the information according to the requirements of each request. This situation made it difficult to obtain information in a timely manner, as there are deadlines that must be met.

The system optimizes supervision through its various functionalities, allowing for immediate and organized consultation of high-cost diagnoses, as well as information on employees, employers, and healthcare data. Likewise, the system offers options for

assigning cases to different roles and tracking them within the supervision process, which contributes to improving the traceability of files.

Consequently, the SSAIS helps optimize the process of monitoring improper enrollments, facilitating the integration of information, case assignment and follow-up, improving the traceability of files and the financial sustainability of the entity.

Keywords: System for Monitoring Improper Enrollments in Health, monitoring, improper enrollment, follow-up, traceability.

TABLA DE CONTENIDO

RESUMEN	v
ABSTRACT.....	vii
TABLA DE CONTENIDO	ix
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE TABLAS	xii
INTRODUCCIÓN	1
CAPÍTULO I – TRAYECTORIA PROFESIONAL	3
CAPÍTULO II – CONTEXTO EN EL QUE DESARROLLA LA EXPERIENCIA	19
2.1. Actividad que realiza la entidad prestadora de servicios de salud	19
2.2. Visión	19
2.3. Misión	20
2.4. Organización de la entidad prestadora de servicios de salud	20
2.5. Área Cargo Funciones Desempeñadas	21
2.6. Experiencia en la organización	22
CAPÍTULO III – ACTIVIDADES DESARROLLADAS	23
3.1. Situación Problemática	23
3.1.1. Casos internacionales	23
3.1.2. Casos nacionales.....	24
3.1.3. Definición del Problema	26
3.2. Solución	27
3.2.1. Objetivos	27
3.2.1.1. Objetivo general.....	27
3.2.1.2. Objetivos específicos.	28
3.2.2. Alcance.....	28
3.2.2.1. Alcance funcional.....	28
3.2.2.2. Alcance Técnico.....	29
3.2.2.3. Limitaciones y exclusiones.	30
3.2.3. Etapas y Metodología	31
3.2.3.1. Etapas.....	31

3.2.3.2.	Metodología.....	32
3.2.4.	<i>Fundamentos Utilizados</i>	35
3.2.5.	<i>Implementación de las Áreas, Procesos y Buenas Prácticas</i>	45
3.2.5.1.	Sprint 0: Planificación inicial y diseño técnico.....	45
3.2.5.2.	Sprint 1: Desarrollo de funcionalidades para la gestión de casos del Jefe USAPE	69
3.2.5.3.	Sprint 2: Desarrollo de funcionalidades para la gestión de casos del Analista USAPE.....	72
3.2.5.4.	Sprint 3: Desarrollo de funcionalidades para la gestión de casos del Supervisor... ..	75
3.2.5.5.	Sprint 4: Pruebas finales y despliegue.....	78
3.3.	Evaluación.....	79
3.3.1.	<i>Evaluación Económica/Evaluación Costo Beneficio</i>	79
3.3.2.	<i>Interpretación del Van y del Tir</i>	81
	Cálculo del VAN	81
	Cálculo del TIR.....	82
CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES.....		85
5.1.	Conclusiones.....	85
5.2.	Recomendaciones	86
5.3.	Fuentes de Información.....	87
5.4.	Glosario.....	90
ANEXOS		90

ÍNDICE DE FIGURAS

Figura 1 Organigrama de la entidad.....	20
Figura 2 Organigrama del scrum team.....	48
Figura 3 Diagrama Idef0 AS-IS	52
Figura 4 Diagrama BPMN AS-IS	55
Figura 5 Formatos Administrativos de supervisión	60
Figura 6 Diagrama Idef0 TO-BE	61
Figura 7 Diagrama BPMN TO-BE	62
Figura 8 Arquitectura del SSAIS	65
Figura 9 Herramienta de gestión de tareas.....	67
Figura 10 <i>Bandeja del Jefe USAPE</i>	70
Figura 11 <i>Funcionalidad Asignar caso a Analista</i>	71
Figura 12 Bandeja del analista USAPE	73
Figura 13 Funcionalidad analizar caso	73
Figura 14 Funcionalidad asignar caso a Supervisor.....	74
Figura 15 Bandeja del supervisor	76
Figura 16 Funcionalidad evaluar caso	77

ÍNDICE DE TABLAS

Tabla 1	Experiencia profesional.....	3
Tabla 2	Cursos realizados.....	18
Tabla 3	Casos de afiliaciones indebidas	24
Tabla 4	Stack tecnológico.....	29
Tabla 5	Distribución de sprints y propósito general	31
Tabla 6	Equipo de desarrollo.....	47
Tabla 7	Product Backlog inicial	49
Tabla 8	Requerimientos no funcionales	51
Tabla 9	Resumen activades clave del proceso AS-IS	56
Tabla 10	Resumen de problemas identificados en el proceso actual (AS-IS)	58
Tabla 11	Funcionalidades clave del proceso TO-BE.....	63
Tabla 12	Planificación temporal del proyecto	68
Tabla 13	Costos operativos del equipo de desarrollo	79
Tabla 14	Parámetros económicos para la evaluación financiera del proyecto.....	81
Tabla 15	Product Backlog Final	92
Tabla 16	Historias de usuario clave.....	95

INTRODUCCIÓN

La entidad prestadora de servicios de salud que de ahora en adelante la llamaremos MasSalud360¹, proporciona atención a los afiliados y a sus familiares en todo el territorio nacional, a través de una red que incluye centros de atención primaria, policlínicos y hospitales. En estos establecimientos, los beneficiarios pueden acceder a distintos tipos de atención, que comprenden consultas ambulatorias, emergencias, hospitalización, cirugías de alta complejidad y rehabilitación. Los afiliados incluyen trabajadores del sector público y privado, así como sus cónyuges, hijos menores de edad y otros familiares debidamente registrados.

La afiliación se realiza principalmente a través de las empresas, las cuales efectúan aportes periódicos al fondo que la entidad administra. El proceso de supervisión actual es fundamentalmente manual, presenta inconsistencias, como la permanencia de trabajadores que no tienen vínculo laboral, errores en el registro de condiciones médicas y la afiliación de dependientes con diagnósticos falsos de enfermedades crónicas, asimismo, se han detectado empresas “fantasma” creadas con el único fin de tener acceso indebido a beneficios y subsidios. Estas circunstancias evidencian la necesidad de fortalecer los mecanismos de control, con el propósito de garantizar la transparencia en la afiliación y asegurar que los servicios de salud lleguen efectivamente a quienes corresponda.

¹ Nombre ficticio usado para proteger la identidad y confidencialidad de la entidad, dado que la información tratada es de carácter sensible y no se tiene la autorización formal para hacer pública su denominación verdadera.

A continuación, se presenta la estructura general del trabajo, describiendo los capítulos que componen el informe y el propósito de cada uno dentro de la experiencia profesional.

En el capítulo I se aborda la trayectoria profesional, mostrando la experiencia adquirida y destacando las competencias, habilidades y conocimientos que sustentan el desarrollo del proyecto.

El capítulo II describe el contexto en el que se llevó a cabo la experiencia, incluyendo una reseña general de la institución y una descripción del área donde se realizaron las actividades, así como la experiencia obtenida a partir de las funciones desempeñadas.

El capítulo III detalla las actividades realizadas durante la experiencia, desde la identificación de la problemática y la propuesta de solución, hasta los objetivos, el alcance, la metodología aplicada, los fundamentos técnicos utilizados y la implementación organizada por sprints.

En el capítulo IV se presenta una reflexión crítica sobre la experiencia, analizando los aprendizajes y los retos enfrentados a lo largo del proceso.

Finalmente, en el capítulo V se incluyen las conclusiones y recomendaciones derivadas del trabajo, junto con las fuentes de información consultadas que respaldan el desarrollo del proyecto.

CAPÍTULO I – TRAYECTORIA PROFESIONAL

El autor cuenta con una amplia experiencia dentro del área de tecnologías de la información, principalmente en proyectos de desarrollo de software. Durante este periodo ha trabajado en actividades de análisis funcional, desarrollo, integración, y mantenimiento de procesos críticos en organizaciones públicas y privadas.

A lo largo de su carrera ha desempeñado distintos cargos evolucionando desde funciones técnicas de programación hacia responsabilidades de mayor alcance en el diseño y desarrollo de soluciones a nivel de frontend y backend.

Ha trabajado para empresas consultoras especializadas en desarrollo de software, participando en proyectos de sectores diversos como automotriz, hidrocarburos y telecomunicaciones. Entre sus experiencias más recientes, se ha desempeñado en ESALUD como programador backend y actualmente en SERFOR como analista desarrollador.

A continuación, se describen los detalles:

Tabla 1
Experiencia profesional

Organización y Periodo	Cargo y Funciones
SERFOR	Cargo: Analista Desarrollador
2025	Cliente: MIDAGRI - SERFOR
	Funciones:

-
- Reuniones con stakeholders y usuarios finales para levantamiento de requerimientos y definición de alcance.
 - Diseño y desarrollo de wireframes y mockups para presentación y aprobación por parte de los usuarios.
 - Análisis funcional detallado del proyecto, asegurando el cumplimiento de las especificaciones.
 - Elaboración de documentación técnica según formato de la entidad.
 - Desarrollo e implementación del backend y frontend del módulo, aplicando buenas prácticas y asegurando integración con el sistema SNIFFS.
 - Ejecución de pruebas funcionales para validar la correcta operación de las funcionalidades desarrolladas.

Tecnologías: Java, Angular, Spring Boot, JPA, SQL Server.

ESSALUD

2024 -2024

Cargo: Desarrollador Backend

Cliente: ESSALUD

Proyecto: SIPROVE

Funciones:

- Reuniones con stakeholders y usuarios finales para levantamiento de requerimientos y definición de alcance.
 - Diseño y desarrollo de wireframes y mockups para presentación y aprobación por parte de los usuarios.
 - Elaboración de documentación técnica, incluyendo manuales de usuario y cronogramas de actividades del proyecto.
 - Análisis funcional detallado del proyecto, asegurando el cumplimiento de las especificaciones.
 - Realización de pruebas funcionales para garantizar la calidad del software.
-

	- Reuniones específicas con los stakeholders y usuarios finales para validación funcional y pruebas de aceptación. - Coordinación y soporte al equipo de desarrollo mediante reuniones bajo metodología Scrum para seguimiento del avance y resolución de incidencias. Tecnologías: Java, Angular, Spring Boot, Microservicios, Hibernate, JPA, Oracle, Tomcat.
SES 2024 -2024	Cargo: Analista Programador Cliente: OTASS Proyecto: SIISS Funciones: - Participar en el análisis funcional de alto nivel del módulo contable de un ERP para el sector público, identificando funciones como comprobantes, pagos y conciliaciones, en base al TDR. - Participar en la definición las interfaces de usuario y flujos funcionales preliminares, considerando normativa SUNAT, MEF y SIAF. - Elaborar procedimientos y guías para incorporar prácticas de seguridad durante el desarrollo, con el objetivo de minimizar riesgos y cumplir con las normativas. Tecnologías: Java, Spring Boot, Angular, JPA, Spring Data, Spring REST, SQL Server, Microservicios.
INDRA 2021 -2023	Cargo: Analista Programador Cliente: Telefónica del Perú Proyecto: Monitoreo de Plataformas Plataformas: Altamira, ACS, Agendador, Troubleshoot, Apollo, IntraWay 2, Multiconsulta, TDE-TOA, NOKIA, GPON Expresse.

Funciones:

- Desarrollo de dashboards en Zabbix, Grafana, Power BI y Splunk para monitoreo y gestión operativa.
- Instalación y configuración de agentes Zabbix en múltiples plataformas.
- Automatización de tareas de monitoreo mediante scripts Bash.
- Configuración de métricas, KPIs y triggers personalizados en Zabbix.
- Implementación de monitoreo transaccional y pruebas funcionales.
- Migración y consolidación de infraestructura de monitoreo.
- Mantenimiento preventivo y gestión de respaldos y almacenamiento.
- Soporte y optimización de dashboards gerenciales y operativos

Principales aportes:

- Desarrollo de dashboards en Zabbix y Grafana, unificando visualizaciones de tráfico, rendimiento y estado de servicios e infraestructura en tiempo real.
 - Instalación y configuración de agentes Zabbix, pruebas de conectividad, ajustes de monitoreo para hardware, SO y servicios clave en múltiples plataformas.
 - Automatización mediante scripts Bash para monitoreo de tiempo de respuesta, uso de CPU, memoria, disco, tráfico y validación de procesos.
 - Configuración de métricas y KPIs personalizados en Zabbix, triggers y métricas de servicios críticos.
-

-
- Implementación de monitoreo transaccional para servicios, con pruebas de inserción, consulta y eliminación.
 - Migración a infraestructura de monitoreo centralizada en Zabbix y Grafana, consolidando la supervisión de todas las plataformas.
 - Mantenimiento preventivo, gestión del almacenamiento y respaldo de bases de datos de las plataformas, y ajustes periódicos para garantizar disponibilidad y estabilidad operativa.
 - Diseño e implementación de dashboards en Power BI para la gestión de eventos, visualizando alarmas activas, resueltas y métricas de desempeño de todas las plataformas monitoreadas.

Tecnologías: Bash Scripting, AWK, Power BI, Zabbix, Grafana, Splunk, PostgreSQL, PostgreSql, Oracle.

Proyecto: Gestión y Soporte Robotización

Principales aportes:

- Implementación y configuración del dashboard para Actualización de Facilidades Técnicas, con desarrollo de consultas SPL para visualización dinámica y en tiempo real de datos operativos.
 - Diseño y configuración del dashboard para Descuentos por Portabilidad, integrando consultas SPL y scripts complementarios para automatización del procesamiento y visualización de información.
 - Desarrollo y configuración del dashboard Gerencial Ecommerce Hogar, utilizando consultas SPL para facilitar la interpretación de indicadores clave en la toma de decisiones.
 - Creación y configuración del Dashboard de Monitoreo Triage TRI, aplicando consultas SPL para
-

visualización dinámica y actualizada de indicadores esenciales.

- Resolución de incidencias y mantenimiento continuo de los dashboards Gerencial Cobranzas y Monitoreo RPAs, asegurando funcionamiento óptimo y disponibilidad de información para la gestión operativa.

Tecnologías: Splunk

GLOBAL HITSS
2019 -2020

Cargo: Analista Programador

Cliente: Claro USA

Funciones:

- Desarrollo y mantenimiento de componentes backend, incluyendo lógica de negocio, validaciones y la integración con sistemas como Salesforce, Mulesoft y plataformas internas.
- Apoyo en el diseño y ajustes de interfaces y reportes utilizados por equipos funcionales, con foco en mejorar la experiencia del usuario.
- Participación activa en pruebas técnicas, resolución de incidencias y soporte durante despliegues para asegurar el funcionamiento estable del sistema.
- Coordinación con equipos funcionales y técnicos para pruebas e implementación.

Proyecto: EORDER

Principales aportes:

- Implementación de la integración con SFDC, permitiendo la recepción de pedidos vía Mulesoft, visualización en PDF y envío de confirmaciones automáticas.
 - Desarrollo de procedimientos almacenados y automatización del flujo de pedidos.
-

-
- Implementación de lógica de validación, colas de procesamiento y auditoría.
 - Modificación de reportes Jasper y creación de subreportes dinámicos.
 - Mejora de funcionalidades de creación manual, autopoblación y gestión de contactos.

Tecnologías: Java 8, Spring MVC, JSF 2, Primefaces, Hibernate, JPA, JasperReports, Apache POI, SLF4J, Oracle. Servidor de aplicaciones WildFly.

Proyecto: PMTOOL

Principales aportes:

- Desarrollo de lógica para registrar automáticamente órdenes de desconexión en tablas de cola con datos críticos del proceso.
- Diseño e implementación de la integración de procesos de desconexión entre plataformas comerciales, CRM y sistemas de provisión.
- Automatización de la selección de servicios activos para desconexión y gestión de fechas de facturación y penalidades.

Tecnologías: Java 8, Spring MVC, JSF 2, PrimeFaces 6.0, PrettyFaces, JSTL, JSP, Hibernate, JPA, JasperReports, Apache POI, SLF4J, Servicios web, Oracle. Servidor de aplicaciones JBoss.

Proyecto: SOM

Principales aportes:

- Implementación de la optimización del flujo de tareas y manejo de cancelaciones en proyectos del equipo de implementación.
-

-
- Desarrollo de notificaciones automáticas y mejoras en la gestión de notas, historial y exportación de datos.
 - Colaboración en mejoras de interfaz y funciones específicas para usuarios finales y equipos funcionales.

Tecnologías: Java 8, Spring MVC, JSF 2, PrimeFaces 6.0, PrettyFaces, JSTL, JSP, Hibernate, JPA, SLF4J, JasperReports, Aspose Words, Apache POI, Servicios web, Oracle. Servidor de aplicaciones Tomcat.

Proyecto: SOS

Principales aportes:

- Optimización y migración de estructuras de bases de datos para mejorar el manejo y control de información comercial.
- Desarrollo y ajuste de procedimientos almacenados y componentes backend para integrar Salesforce y otros sistemas externos.
- Actualización y mejora de reportes y funciones de exportación para ofrecer mejor visibilidad y control de datos.
- Implementación de interfaces y funcionalidades frontend según nuevos requerimientos.

Tecnologías: Java 8, Spring MVC, Spring Data, JSF 2, Primefaces, PrettyFaces, JSP, JSTL, Hibernate, JPA, Apache POI. Servidor de aplicaciones JBoss.

VASS PERÚ
2018 -2018

Cargo: Analista Programador

Cliente: Telefónica del Perú

Funciones:

- Desarrollar, actualizar e integrar servicios backend y bases de datos para asegurar el correcto
-

funcionamiento de aplicaciones web y el flujo de información.

- Análisis, desarrollo y optimización de procesos vinculados a la gestión de ventas y productos, trabajando en estrecha comunicación con los stakeholders para garantizar el cumplimiento de los requerimientos funcionales.
- Desarrollo, implementación, pruebas funcionales de y documentación técnica los sistemas.

Proyecto: Web Fija

Principales aportes:

- Desarrollo de funcionalidades web para visualizar pedidos y productos, facilitando el seguimiento de ventas.
- Optimización de servicios backend utilizados por la web pública, mejorando el rendimiento y reduciendo los tiempos de respuesta.
- Integración de datos desde el CMS para clientes con productos migrados asegurando coherencia en la información mostrada.
- Validación de números telefónicos durante procesos de venta y manejo de direcciones normalizadas fuera del distrito para mejorar la precisión.
- Generación de reportes con historial de pedidos e implementación de procesos batch para campañas comerciales mediante carga de archivos.

Tecnologías: Java 8, Spring Boot 1.4.7, servicios RESTful, JPA, Spring Data JPA, Apache POI, JasperReports, IBM Watson Java, Auth0 Java JWT, Microservicios, Gradle PostgreSQL, Angular, Bootstrap. Servidor de aplicaciones Jetty.

Proyecto: Convergente**Principales aportes:**

Proyecto desarrollado como evolución funcional de Venta Fija, enfocado en la paquetización de servicios móviles, fijos y equipos en una única oferta comercial.

- Implementación de mejoras y ajustes en el módulo de login existente para su integración en el nuevo flujo comercial de Movistar Total.
- Reutilización y adaptación de APIs existentes para autenticación, identificación de clientes, validación de líneas y generación de ofertas dentro del flujo comercial.
- Integración de microservicios con sistemas internos y externos como RENIEC para validación de identidad, +Simple y sistemas comerciales para recuperación de datos del cliente, ABD para portabilidad y scoring crediticio, y coordinación para la integración del Motor de Ofertas, asegurando la preparación del sistema para su incorporación futura.
- Desarrollo de lógica de negocio para validaciones comerciales: deudas, titularidad, cobertura técnica, condiciones de portabilidad y financiamiento de equipos.
- Automatización del proceso de venta paquetizada, incluyendo líneas móviles, fijas y adquisición de equipos, bajo reglas dinámicas de scoring.

Tecnologías: Java 8, Spring Boot 1.4.7, servicios RESTful, Hibernate, JPA, Spring Data JPA, Apache POI, JasperReports, IBM Watson Java, Auth0, Java JWT, SLF4J, Twilio, Bouncy Castle, Gradle PostgreSQL, Angular,

	Angular CDK, Angular Material, Bootstrap. Servidor de aplicaciones Jetty.
CONASTEC 2017 -2017	Cargo: Analista Programador Cliente: Atento Perú Funciones: Desarrollo e implementación de servicios API REST y migración de bases de datos de Oracle a SQL Server. Responsable del mantenimiento y evolución de proyectos del Sistema de Banca por Teléfono (SBT) y de administrador SBT Web. Implementación de procesos ETL con SSIS para consolidar y transformar datos en SQL Server. Tecnologías: Java, JSP, Html, Java Script, Servlets, Hibernate, Oracle, SQL Server, RESTFul.
TECNOCOM PERÚ 2013 -2016	Cargo: Especialista TI Cliente: Osinergmin Funciones: - Participación en el análisis, diseño, desarrollo y mantenimiento de soluciones orientadas a mejorar el control operativo del sector hidrocarburos, asegurando trazabilidad y cumplimiento normativo en los sistemas SCOP, SCOP GLP, NSCOP, SCOP Móvil, Seguridad PVO, Emergencias Mayores y PDJEE. - Levantamiento detallado y documentación de las incidencias reportadas por usuarios, con seguimiento constante hasta su resolución, en coordinación con el área usuaria. - Implementación de requerimientos funcionales solicitados por los usuarios, siempre orientados a optimizar procesos operativos y cumplir con las normativas vigentes.

-
- Ejecución de pruebas funcionales posteriores a cada implementación, con atención a las observaciones reportadas y corrección oportuna de los inconvenientes identificados.
 - Elaboración de documentación técnica completa, que incluye especificaciones, diseño detallado, diagramas y manuales de usuario para facilitar la adopción de las nuevas funcionalidades.

Proyecto: SCOP – Sistema de Control de Órdenes de Pedido

Principales aportes:

- Desarrollo e implementación exitosa del módulo “Cierre de Órdenes de Pedido Anticipadas” dentro del Módulo Orden de Pedido, optimizando la gestión de pedidos.
- Mantenimiento correctivo y evolutivo en diversos módulos, incluyendo Orden de Pedido, PNP, Mantenimiento, Reportes y Consultas, asegurando la continuidad operativa y mejora constante del sistema SCOP.
- Integración de visualización geográfica del agente comprador en las interfaces del agente mayorista, mejorando la trazabilidad y control.
- Mejora continua del sistema a través del análisis y resolución de ocurrencias reportadas, aumentando la satisfacción del área usuaria.
- Documentación clara y completa que facilitó la capacitación y soporte a usuarios finales.

Tecnologías: Java 6, JSP 2, JSTL, Servlets 2, JDBC y Oracle 11g. Servicios Web, iReport 5. Servidor de aplicaciones Tomcat 6.

Proyecto: SCOP GLP – Gas Licuado de Petróleo**Principales aportes:**

- Implementación de proceso para registrar consultas de usuarios controlados en órdenes de pedido GLP, optimizando el seguimiento y cumplimiento normativo.
- Implementación de mejoras en el módulo de generación de órdenes de pedido para pedidos simples a granel agilizando significativamente el proceso de ventas envasadas.
- Creación de procedimientos almacenados y se diseñaron estructuras de base de datos orientadas al control operativo y al soporte de análisis posteriores, contribuyendo a una supervisión más eficiente de las operaciones.

Tecnologías: Java 6, JSP 2, Servlets, Servicios web, JSTL, Apache POI, Struts Framework, iReport 5, JDBC y Oracle 11g. Servidor de aplicaciones Tomcat 6.

Proyecto: Emergencias Mayores**Principales aportes:**

- Creación y actualización, en el módulo de Registro, de los formatos de emergencia 1, 2, 4, 5, 7 y 9, que permiten el registro de accidentes graves o fatales, incidentes y situaciones de emergencia en el subsector hidrocarburos.
- Desarrollo e implementación de funciones para la generación y visualización de reportes relacionados.

Tecnologías: Java 6, Web, JQuery 1.4, jqGrid 4, Spring 3, Spring MVC 3, Spring Security 3.0, JPA 2/Hibernate4, iReport 5 y Oracle 11g. Servidor de aplicaciones Tomcat 7.

Proyecto: NSCOP**Principales aportes:**

- Desarrollo de funcionalidades para la emisión e impresión de Certificados de Conformidad dirigidos a Consumidores Directos y Redes de Distribución, en el módulo Certificados de Conformidad CD/RD, incluyendo consultas, mantenimiento de servicios web y desarrollo de páginas JSP.
- Implementación de funcionalidades para el registro de solicitudes de pre-importación, en el módulo Registro de Importaciones de GLP, abarcando la creación de consultas, desarrollo de páginas JSP y actualización de servicios web.

Tecnologías: Java 6, JSP, Spring 2.5, Struts 2, Hibernate 3, JasperReports 3, y Oracle 11g. Servidor de aplicaciones Tomcat 6.

Proyecto: PEDJEE**Principales aportes:**

- Desarrollo de funcionalidades para el registro en línea de Declaraciones Juradas en el sistema PDJEE, orientado a empresas con operaciones de exploración y explotación de hidrocarburos líquidos.
- Implementación de mejoras en el módulo de Inventario del PDJEE, con precarga de datos históricos para facilitar las declaraciones anuales.

Tecnologías: Java 6, JSP, JSTL, DisplayTag Framework, JQuery, Struts 1, Spring 3, Hibernate 3, JDBC, JasperReports 3.0 y Oracle 11g. Servidor de aplicaciones Tomcat 6.

Proyecto: PVO**Principales aportes:**

-
- Implementación de mejoras en el sistema de seguridad PVO, específicamente en el módulo de Mantenimientos, que abarca las secciones de Aplicaciones, Módulos, Páginas, Roles y Usuarios. Incluyó la reformulación del acceso al sistema y la ampliación de validaciones para optimizar el ingreso de datos en campos clave.

Tecnologías: Java 6, JQuery 1.8, JQuery-ui 1.8, Spring 3, JPA 2/Hibernate 3, Spring Core, Spring JDBC, Spring Security, Spring MVC, Web Services y Oracle 11g. Servidor de aplicaciones Tomcat 6, WebLogic.

Proyecto: SCOP MOVIL

Principales aportes:

- Análisis, diseño e implementación del módulo de Generación de Orden de Pedido para SCOP Móvil, enfocado en unidades operativas como grifos, estaciones de servicio y consumidores directos.
- Desarrollo de funcionalidades para el registro y cierre de órdenes de pedido de GLP.
- Implementación de opciones para registrar precios de GLP y cilindros.
- Actualización de información comercial vinculada a las operaciones del módulo.

Tecnologías: Java 6, JSP, JQuery 1.9, JQuery-ui 1.11, JQuery Mobile 1.3, Spring MVC 3, Hibernate 3.6, JPA 2, Oracle 11g. Servidor de aplicaciones Tomcat 7, WebLogic.

MDP CONSULTING

2012 -2013

Cargo: Programador

Ciente: Divemotor, Yanbal

Funciones: Análisis, diseño, implementación y pruebas funcionales de los sistemas para la gestión de citas y venta de repuestos.

	Responsable del mantenimiento correctivo y evolutivo de la plataforma extranet corporativa bajo líneas ágiles. Tecnologías: Java 6, Angular, Bootstrap, Spring Boot, Spring 3, JPA/Hibernate 4, JSF 2, Primefaces 3.4, Oracle 11g, servidor de aplicaciones JBoss.
--	--

Nota. Elaboración propia

Los cursos realizados por el autor se muestran en la Tabla 2:

Tabla 2

Cursos realizados

Institución	Curso
GALAXY TRAINING	Microservices Architecture: Fundamentals And Applications
GALAXY TRAINING	Full-Stack Java Developer: Spring Boot 3.0 y Angular 19
MITOCODE NETWORK	Docker Kubernetes Cloud
MITOCODE NETWORK	Java 11 Backend Developer
CIBERTEC	Java Advanced Developer
SISTEMAS UNI	Workshop: Oracle PL/SQL
SISTEMAS UNI	Introducción a SQL-Oracle
SISTEMAS UNI	MS SQL Server Implementación
SISTEMAS UNI	Modelamiento de Datos

Nota. Elaboración propia

CAPÍTULO II – CONTEXTO EN EL QUE DESARROLLA LA EXPERIENCIA

2.1. Actividad que realiza la entidad prestadora de servicios de salud

La entidad prestadora de servicios de salud MasSalud360 tiene como objetivo principal ofrecer a los asegurados y sus familiares acceso de manera oportuna y eficiente a los servicios de salud. La cobertura integral incluye atención médica primaria, consultas especializadas, hospitalización, cirugías, emergencias y programas de prevención y rehabilitación.

2.2. Visión

Ser una institución moderna y en constante mejora, enfocada en los asegurados, que garantice un acceso ético, oportuno y de calidad a la seguridad social en salud.

2.3. Misión

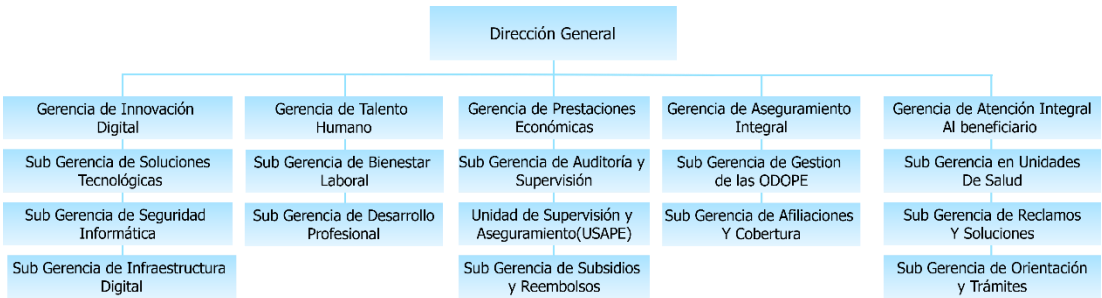
Proporcionar servicios de salud, económicos y sociales a los asegurados mediante una gestión innovadora y eficiente, asegurando la protección financiera y el acceso integral a las prestaciones, entre ellas pensiones por invalidez, subsidios por incapacidad temporal y por maternidad.

2.4. Organización de la entidad prestadora de servicios de salud

MasSalud360 está organizada en una dirección general donde se toman las decisiones estratégicas y se definen las políticas institucionales, mientras que las distintas gerencias y áreas de apoyo se encargan de la innovación digital, la gestión del talento humano, las prestaciones económicas, el aseguramiento integral y la atención al beneficiario, tal como se muestra en la figura 1.

Cada gerencia tiene subgerencias que realizan funciones específicas, como el desarrollo de soluciones tecnológicas, la administración de subsidios y pagos, la Unidad de Supervisión de Aseguramiento y Prestaciones Económicas (USAPE) y la coordinación de la atención en unidades de salud.

Figura 1
Organigrama de la entidad



Nota. Fuente: Elaboración propia

2.5. Área Cargo Funciones Desempeñadas

El área, cargo y funciones desempeñadas se detallan a continuación:

Área: Sub Gerencia de Auditoria y Supervisión.

Cargo: Analista funcional, responsable del análisis del proceso actual, diseño y de mejora del proceso de supervisión de afiliaciones indebidas.

Funciones: El trabajo se desarrolló siguiendo la metodología Scrum, pero adaptada a las necesidades de la entidad.

- Desempeñar funciones de analista funcional, participando en reuniones con el área usuaria y stakeholders para recopilar información con el fin de comprender el proceso de negocio, identificar, analizar y documentar los requerimientos del sistema.
- Elaborar cronogramas de trabajo que reflejen las fases del proyecto, entregables y tiempos estimados.
- Realizar seguimiento al cumplimiento de plazos y efectuar los ajustes necesarios para asegurar que las actividades del equipo estuvieran alineadas a los objetivos del proyecto.
- Participar en las ceremonias de la metodología Scrum, el Spring Planning, el Spring Review y la Sprint Retrospective, convocadas y facilitadas por el Scrum Master, aportando información funcional, avances del equipo y proporcionando retroalimentación de los requerimientos del área usuaria que también actuaba como Product Owner.
- Coordinar y facilitar las Daily Scrum, promoviendo la autoorganización del equipo, identificando avances, stoppers y próximos pasos.
- Preparar y conducir reuniones de planificación internas con el equipo de desarrollo, donde se explicaba el contenido de las tareas del Sprint Backlog elaboradas a partir de los requerimientos del Product Owner y por otro lado se coordinaba la distribución de responsabilidades tanto del frontend como del backend según las prioridades definidas por el Product Owner.
- Apoyar técnicamente al equipo de desarrollo en la implementación de mejoras y en la supervisión de procesos del sistema.
- Definir y ejecutar pruebas funcionales para validar los procesos implementados.

- Elaborar documentación técnica y manuales de usuario conforme a la metodología institucional.

2.6. Experiencia en la organización

- Coordinación efectiva con equipos multidisciplinarios, incluyendo médicos, administradores, ingenieros y gerentes.
- Desarrollo de habilidades blandas como comunicación, empatía y liderazgo colaborativo.
- Adaptación de metodologías ágiles a las dinámicas específicas del cliente.
- Fortalecimiento del enfoque funcional en entornos de aseguramiento de salud.

CAPÍTULO III – ACTIVIDADES DESARROLLADAS

3.1. Situación Problemática

Las afiliaciones indebidas son un problema de larga data y es un fenómeno global. Esta acción ilícita afecta los recursos económicos de los sistemas de la Seguridad Social, al tratarse de inscripciones que no cumplen con los requisitos legales para acceder a la cobertura, generando perjuicios económicos a la entidad.

3.1.1. Casos internacionales

En el contexto internacional, se han reportado casos de afiliaciones indebidas en los sistemas de seguridad social.

Tenemos el caso de Honduras, donde se creó una planilla fantasma de 22 personas que se hicieron pasar por médicos, para ello falsificaron documentos públicos, firmas y sellos. Este acto ilícito permitió la posibilidad no solo de percibir pagos de remuneración sino también recibir los beneficios que todo asegurado tiene, perjudicando las finanzas del Instituto Hondureño de Seguridad Social (IHSS) (Ministerio Público de Honduras, 2025).

Así mismo, en Uruguay, a través de una investigación interna que realizó el Banco de Previsión Social (BPS), encontró 300 afiliaciones realizadas de forma irregular y que

debido a esta afiliación sus hijos, cónyuges y concubinos sumarian un total de 500 afiliaciones (Montevideo portal, 2017).

Por otro lado, en España la Policía Nacional en coordinación con la Seguridad Social y la Dirección General de la Inspección del Trabajo, descubrieron una red empresarial conformada por diez empresas todas creadas por la misma persona. Estas empresas tenían como propósito cometer fraudes, cobrando a ciudadanos extranjeros para que estos obtengan contratos de trabajo falso y de este modo estén afiliados al sistema de Seguridad Social, accediendo así a las prestaciones y subsidios por desempleo (Ministerio de Inclusión, Seguridad Social y Migraciones, 2013).

Tabla 3
Casos de afiliaciones indebidas

País	Tipo de fraude	Descripción	Fuente
Honduras	Planillas fantasmas	Inclusión en planillas de personas ficticias.	Ministerio Público de Honduras, comunicado 2025.
Uruguay	Afiliaciones Fraudulentas	Empresas sin vinculo real con empleados.	Banco de Previsión Social (BPS). Declaración oficial 2017.
España	Empresa ficticia	Contratos falsos para afiliación.	Ministerio de Inclusión, 2013.

Nota. Elaboración propia

3.1.2. Casos nacionales

En el contexto nacional, a través de publicaciones oficiales del Seguro Social del Perú (EsSalud), se han documentado diversas modalidades de afiliaciones indebidas. Entre ellas se encuentran la suplantación de asegurados fallecidos, la declaración irregular de vínculos laborales y la inscripción de trabajadores en empresas sin actividad

económica o con RUC dado de baja que seguían declarando personal, así como otras prácticas registradas en informes oficiales (EsSalud, 2014; EsSalud, 2016).

Un caso particularmente relevante es la detección, en 2013 (publicado en 2014), de una presunta organización que utilizaba registros de asegurados fallecidos que aún permanecían activos en el sistema administrativo (EsSalud, 2014).

Posteriormente, en el 2015, y tras la aplicación de un reglamento orientado a verificar la condición del asegurado, la institución detectó 5,521 afiliaciones indebidas, lo que permitió evitar un gasto superior a 50 millones de soles en ese año. La identificación de estos casos se realizó mediante perfiles de riesgo, entendido como criterios o patrones que permiten identificar comportamientos o situaciones sospechosas de afiliaciones indebidas, para identificar a las personas que se filtran en el sistema, y el cruce de información con otras entidades públicas (EsSalud, 2016).

La problemática aún no ha sido superada. El Informe de Evaluación del Plan Estratégico Institucional 2020–2024, primer semestre 2024, informa sobre el seguimiento del indicador AEI 1.2, que mide la variación del gasto asistencial de personas atendidas en situación de afiliación indebida o fuera de su cobertura de salud. El informe muestra que la institución continúa evaluando el impacto económico de estas afiliaciones y señala la existencia de brechas operativas relacionadas con la integración y oportunidad de la información entre redes asistenciales y oficinas administrativas, lo que dificulta la consolidación y supervisión de los casos (EsSalud, 2024).

En 2025, la Contraloría General de la República informó que EsSalud realizó pagos indebidos a 101 servidores y pensionistas fallecidos entre 2020 y 2023, generando un perjuicio económico de S/ 963,303.55. Este hecho evidenció debilidades en los mecanismos de actualización y control de la información de los asegurados (Contraloría General de la República, 2025).

En el mismo periodo, mediante la Resolución de Gerencia Central N.º 30-GCSPE-ESSALUD-2025, EsSalud reorganizó las funciones relacionadas con la verificación de la

condición del asegurado a nivel nacional, mostrando que este tema sigue siendo una prioridad operativa (EsSalud, 2025).

3.1.3. Definición del Problema

En la entidad Massalud360, se ha identificado la persistencia de afiliaciones indebidas, tales como la incorporación de personas que no mantienen un vínculo laboral real, la declaración de cónyuges o concubinos como derechohabientes que no cumplen con los requisitos normativos, así como el registro de trabajadores ficticios mediante empresas inactivas o empleadores del hogar inexistentes. Estas prácticas irregulares permiten el acceso indebido a prestaciones de salud y subsidios económicos, afectando directamente la sostenibilidad financiera de la entidad.

Si bien Massalud360 cuenta con un proceso de supervisión del estado del asegurado, dicho proceso se ejecuta actualmente de manera predominantemente manual, con una alta dependencia de documentos físicos, consultas externas realizadas de forma no integrada y registros fragmentados de los hallazgos. Este proceso se apoya en un sistema heredado que funciona únicamente como repositorio de casos, sin capacidades de automatización, trazabilidad integral ni generación de información para la gestión.

Como consecuencia, el proceso presenta retrasos en la atención de los casos, riesgo de incumplimiento de los plazos normativos, deficiencias en la trazabilidad de las actividades y una asignación ineficiente de los casos a los verificadores. Asimismo, la entidad no dispone de mecanismos que permitan conocer en tiempo real la carga laboral de los verificadores, el número de casos resueltos por periodo ni la distribución geográfica de las afiliaciones irregulares, lo que limita la supervisión operativa y la toma de decisiones a nivel gerencial.

En este escenario, Massalud360 enfrenta serias dificultades para detectar oportunamente, gestionar y controlar de manera eficiente las afiliaciones indebidas, afectando tanto la eficiencia operativa como la sostenibilidad financiera institucional.

3.2. Solución

Para superar las limitaciones identificadas, se propone el desarrollo de un sistema web denominado SSAIS (Sistema de Supervisión de Afiliaciones Indebidas en Salud), orientado a automatizar, centralizar y optimizar el proceso de supervisión de la condición del asegurado en la entidad Massalud360.

SSAIS se fundamenta en el uso de un conjunto estructurado de expedientes de afiliados con diagnósticos de alto costo, obtenido a partir de consultas a la base de datos institucional, el cual constituye el insumo principal para el proceso de supervisión de la condición del asegurado.

A partir de este universo de casos, el sistema permite integrar la información del asegurado, del empleador y de los datos asistenciales, facilitando el análisis, seguimiento, control y trazabilidad del proceso de supervisión.

La solución incorpora una arquitectura basada en microservicios y módulos diferenciados por rol, que permiten a supervisores, analistas y jefes monitorear el estado de los casos en tiempo real, asignar la carga de trabajo de los supervisores, dar seguimiento al flujo de los casos y apoyar en la toma de decisiones, contribuyendo a optimizar y apoyar el proceso de supervisión de afiliaciones indebidas el cual presentaba una alta carga de actividades manuales.

3.2.1. Objetivos

3.2.1.1. Objetivo general.

Desarrollar un sistema web basado en una arquitectura de microservicios que permita dar soporte y optimizar el proceso de afiliaciones indebidas en la entidad Massalud360. El sistema centralizará y automatizará la información e incluirá mecanismos de trazabilidad y monitoreo para facilitar la supervisión operativa y la toma de decisiones gerenciales.

3.2.1.2. Objetivos específicos.

- Analizar el proceso actual de supervisión de la condición del asegurado en Massalud360, identificando sus principales limitaciones operativas y tecnológicas.
- Diseñar una arquitectura de microservicios que permita la centralización de la información, visualización, asignación y seguimiento de los casos seleccionados para supervisión derivados del universo de afiliados con diagnóstico de alto costo.
- Implementar un módulo para el acceso centralizado al universo de expedientes de afiliados con diagnósticos de alto costo, que es el insumo principal del proceso de supervisión.
- Implementar módulos diferenciados según los roles institucionales (Jefe USAPE, Analista USAPE y Supervisor) que faciliten la asignación, análisis y seguimiento de los casos.
- Implementar funcionalidades de monitoreo que permita a los roles responsables de la supervisión y de la gestión operativa visualizar, en tiempo real, la carga de casos asignados a los supervisores, así como el estado y avance de cada caso según su etapa dentro de dicho proceso.
- Garantizar la trazabilidad completa de las acciones realizadas durante el proceso de supervisión para fines de control y auditoría interna.

3.2.2. Alcance

3.2.2.1. Alcance funcional.

El sistema web SSAIS está orientado a centralizar y soportar el proceso de afiliaciones indebidas en Massalud360, a partir de información de atenciones asistenciales ya registradas en los sistemas institucionales, mediante la disponibilidad y centralización de expedientes de afiliados con diagnósticos de alto costo, los cuales constituyen el universo e insumo principal para la identificación de posibles afiliaciones indebidas. Sin embargo, no sustituye la supervisión ni toma decisiones sobre la condición de las afiliaciones.

El sistema permite:

- Disponer de manera directa y oportuna el universo de casos de afiliados con diagnóstico de alto costo, accesible desde las bandejas correspondientes a los diferentes roles institucionales.
- Visualizar y asignar los casos de supervisión entre los diferentes roles institucionales como Jefe USAPE, Analista USAPE y Supervisor, asegurando que cada expediente pueda ser revisado y analizado de manera estructurada.
- Registrar el inicio, desarrollo y cierre del proceso de supervisión por parte del supervisor, garantizando la trazabilidad de cada actividad realizada.
- Realizar el seguimiento del estado de cada caso durante todas las etapas del proceso de supervisión.
- Visualizar en tiempo real la carga de trabajo de los supervisores y el estado de los casos asignados.
- Consultar el historial de casos y actividades realizadas para fines de auditoría interna y control administrativo.

3.2.2.2. Alcance Técnico.

A continuación, en la tabla 2 se lista el Stack tecnológico usado.

Tabla 4
Stack tecnológico

Tecnología	Descripción
Arquitectura	Arquitectura de Microservicios para modularidad y mantenimiento independiente.
Java	Lenguaje de programación backend.
Angular	Framework para construir interfaces dinámicas.
Angular Material	Librería de componentes UI.

TypeScript	Lenguaje de programación frontend.
Spring Boot	Framework para crear aplicaciones java basadas en spring.
Spring Framework	Base para construir la lógica empresarial y servicios REST.
Hibernate/JPA	Framework ORM para gestión y persistencia de datos en base de datos relacionales.
Oracle Database	Base de datos principal para almacenamiento de datos.
API RESTFULL	Interfaz de comunicación entre microservicios y servicios externos.
Apache POI	Librería java para manipulación de archivos Word y otros formatos
Git	Control de versiones para gestión del código fuente

Nota. Elaboración propia

3.2.2.3. Limitaciones y exclusiones.

El sistema de supervisión de afiliaciones indebidas no automatiza todo el proceso, sino que facilita a los usuarios la revisión, análisis y seguimiento de cada caso. Las principales limitaciones y exclusiones son:

- Análisis y verificación final de cada caso: Esta etapa no está automatizada, ya que requiere juicio profesional, interpretación de documentos internos y externos (SUNAT, RENIEC) y la aplicación de criterios normativos. El sistema únicamente provee los posibles casos de supervisión y consolida la información relevante para su gestión, dejando la decisión final en manos de los usuarios.
- Detección automática de irregularidades: El sistema no incluye algoritmos de inteligencia artificial para la identificación de afiliaciones indebidas. Si bien la detección inicial se apoya en reglas y perfiles de riesgo, la validación final de cada caso depende del análisis humano y de la revisión profesional.
- Actividades de campo: Visitas domiciliarias y validación física de la información no se ejecutan de manera automatizada. El sistema únicamente permite registrar y dar seguimiento a estas actividades, asegurando trazabilidad y documentación del proceso.

- Alcance técnico: El desarrollo del sistema se ajustó a la arquitectura de microservicios existente en la organización. Por su parte, el despliegue en los entornos de producción y la administración de servidores y bases de datos son responsabilidad del área de infraestructura tecnológica, por lo que no forman parte del alcance de este desarrollo.

El sistema no reemplaza el análisis técnico ni la validación documental realizada por el personal verificador. La decisión final sobre la existencia de una afiliación indebida sigue siendo responsabilidad del área competente, lo que marca el límite funcional de la solución implementada.

3.2.3. Etapas y Metodología

El desarrollo del sistema se realizó usando la metodología ágil Scrum, permitiendo organizar el proyecto en etapas iterativas e incrementales, de modo que se lograra realizar entregas parciales y recibir retroalimentación por parte de los usuarios.

3.2.3.1. Etapas.

El proyecto se estructuró en sprints tal como se muestra en la siguiente tabla:

Tabla 5
Distribución de sprints y propósito general

Sprint	Duración	Propósito general
Sprint 0 – Planificación y definición	2 semanas	Reuniones con usuarios y stackholders, levantamiento de información, definición del

		product backlog, diseño preliminar de la arquitectura y esbozo de las interfaces.
Sprint 1 – Módulo Jefe USAPE	3 semanas	Implementación de las funcionales para el rol Jefe USAPE.
Sprint 2 - Módulo Analista USAPE	3 semanas	Implementación de las funcionales para el rol Analista USAPE.
Sprint 3 - Módulo Supervisor	3 semanas	Implementación de las funcionales para el rol Supervisor.
Sprint 4 – Pruebas finales y despliegue	1 semana	Pruebas de funcionalidad, ajustes finales y despliegue.
<i>Nota.</i> Elaboración propia		

3.2.3.2. Metodología.

El proyecto se llevó a cabo aplicando el marco de trabajo ágil Scrum, que permite gestionar proyectos de software de forma iterativa e incremental. Según Schwaber y Sutherland (2020), Scrum estructura el trabajo en ciclos breves denominados sprints, apoyándose en roles claramente definidos, artefactos específicos y ceremonias periódicas. Este enfoque facilita la entrega progresiva de resultados, la obtención constante de retroalimentación y la adaptación frente a cambios en los requerimientos.

A continuación, se describen los elementos principales según la guía de Scrum (Schwaber & Sutherland, 2020) implementados en el proyecto:

Sprints

Son ciclos de duración fija, generalmente de 2 a 4 semanas, durante los cuales se desarrolla un incremento funcional del sistema. Al finalizar cada sprint, se entrega un producto potencialmente utilizable y listo para validación.

Roles Scrum

- Product Owner: Es la persona que define qué se hace y en qué orden. Prioriza el trabajo según el valor que aporta, se encarga del *Product Backlog* y de validar los entregables.
- Scrum Master: Facilita la correcta aplicación del marco Scrum, supervisa los procesos y elimina impedimentos.
- Equipo de Desarrollo: Implementa, integra y prueba los módulos del sistema, trabajando de manera autónoma y colaborativa.

Artefactos Scrum

- Product Backlog: Lista priorizada de funcionalidades y requerimientos, estructurada en historias de usuario y agrupada en épicas según corresponda.
- Sprint Backlog: Conjunto de tareas seleccionadas del Product Backlog para desarrollarse durante un sprint.

Historias de Usuario y Épicas

- Historias de Usuario: Breves descripciones de funcionalidades desde la perspectiva del usuario final, que permiten priorizar y planificar el trabajo del equipo durante cada sprint.
- Épicas: Conjuntos de historias de usuario relacionadas que representan funcionalidades de mayor alcance y facilitan la organización estratégica del Product Backlog.

Ceremonias Scrum

- Planificación de Sprint (*Sprint Planning*): Selección de historias de usuario y definición de tareas para el sprint.

- Reunión diaria (*Daily Scrum*): Seguimiento del progreso y coordinación del equipo.
- Revisión de Sprint (*Sprint Review*): Evaluación de funcionalidades entregadas y retroalimentación de los usuarios.
- Retrospectiva de Sprint (*Sprint Retrospective*): Identificación de mejoras a implementar en los siguientes sprints.
- Incremento: Producto funcional al final del sprint, listo para evaluación y validación.

Estimación del trabajo en Scrum

La Guía de Scrum define el marco de trabajo a partir de roles, eventos y artefactos, pero no establece técnicas específicas ni unidades concretas para la estimación del esfuerzo (Schwaber & Sutherland, 2020). Por consiguiente, la forma de estimar queda a criterio del equipo, el cual debe seleccionar los métodos que resulten más adecuados según su experiencia y forma de trabajo, siempre en coherencia con los valores y principios de Scrum.

- Puntos de Historia (*History points*): Se utilizan como una medida de estimación relativa que permite dimensionar una Historia de Usuario a partir de aspectos como el esfuerzo involucrado, la complejidad técnica y la incertidumbre del trabajo, sin vincularlos directamente a una duración temporal específica (Cohn, 2005).
- La Escala Fibonacci: Técnica utilizada en Scrum para asignar puntos de historia a las tareas del backlog, permitiendo cuantificar de forma relativa el esfuerzo, la complejidad y el tamaño. Al escoger un número de la serie (1, 2, 3, 5, 8, ...) a una historia de usuario, el equipo no mide el tiempo exacto, sino que considera de manera conjunta la dificultad, complejidad y el grado de incertidumbre de la tarea (Cohn, 2005).
- Planning Poker: Es una técnica de estimación colaborativa en la que cada miembro del equipo asigna de manera individual un valor a cada Historia de Usuario usando una escala, como la Fibonacci. Después, las estimaciones se

comparten y discuten hasta que el equipo llegue a un consenso, lo que ayuda a reducir sesgos personales o jerárquicos en el proceso (Cohn, 2005).

Este enfoque permitió desarrollar el sistema web de manera progresiva, asegurando que cada módulo se construyera con trazabilidad, pudiera adaptarse a cambios en los requerimientos y cumpliera con los objetivos institucionales de soporte al proceso de afiliaciones indebidas.

3.2.4. Fundamentos Utilizados

Seguridad social

La seguridad social es un derecho de todas las personas que protege frente a problemas como enfermedad, invalidez, vejez, desempleo o fallecimiento. Su propósito es garantizar el bienestar y el acceso a servicios básicos (Gonzales Hunt & Paitán Martínez, 2017).

Cobertura de salud

Es el derecho legal a que se cubran o reembolsen los gastos médicos, normalmente a través de un contrato con una aseguradora, un plan de salud grupal vinculado al empleo, o programas gubernamentales.

Se refiere al derecho de que los gastos médicos sean pagados o reembolsados por un asegurador, un plan de salud colectivo o un programa público (Healthcare.gov, s.f.).

Seguro social

El seguro social es la manera concreta de hacer efectivo el derecho a la seguridad social. Proporciona cobertura principalmente a quienes tienen un trabajo formal y se

financia con aportes de trabajadores, empleadores y el Estado (Gonzales Hunt & Paitán Martínez, 2017).

Afiliación indebida

La afiliación indebida se entiende como la situación en la que una persona natural o una entidad empleadora registra a trabajadores en un sistema de seguridad social en salud sin cumplir con los requisitos previstos por la normativa. En la práctica, esto significa que se registran asegurados que no reúnen las condiciones legales para acceder a las prestaciones, lo que constituye un uso inapropiado del sistema (Consortio de Investigación Económica y Social [CIES], 2014; Ministerio de Trabajo y Promoción del Empleo, 2009).

Procedimiento de validación

El procedimiento de validación de afiliaciones indebidas busca identificar a los asegurados que no cumplen con los requisitos legales para acceder a los servicios del sistema de salud. De manera conceptual, este proceso consiste en revisar detalladamente la información de afiliados y de los empleadores, aplicar criterios de control establecidos y dar seguimiento a cada caso, garantizando que las irregularidades se detecten y se gestionen de manera ordenada y sistemática (EsSalud, 2015; CIES, 2014).

En el marco de la normativa vigente, estas acciones se respaldan en los mecanismos de verificación establecidos en el Texto Único Ordenado de la Ley N.º 27444 – Ley del Procedimiento Administrativo General, mediante los cuales las entidades públicas realizan acciones destinadas a constatar hechos, examinar documentación y comprobar la veracidad de la información declarada por los administrados (Ministerio de Justicia y Derechos Humanos, 2021).

A partir de este marco normativo, se puede inferir que la validación de afiliaciones constituye un proceso de control administrativo orientado a asegurar el cumplimiento de las condiciones requeridas para el aseguramiento. Su aplicación contribuye a reducir

inconsistencias en los registros, prevenir afiliaciones que no cumplan los criterios establecidos y fortalecer la confiabilidad de la información dentro del sistema de seguridad social.

Accesibilidad web

La accesibilidad web es importante para que cualquier persona pueda usar sistemas digitales, sin importar sus capacidades (W3C, 2018). Las Pautas de Accesibilidad para el Contenido Web (WCAG), elaboradas por el World Wide Web Consortium (W3C, 2018, 2023), son el estándar internacional más reconocido para que los sitios web funcionen correctamente y sean accesibles para personas con distintas discapacidades.

Estas pautas se organizan en cuatro principios fundamentales:

- **Perceptible:** Indica que la información y los elementos de la interfaz deben ser percibidos por todos los usuarios, sin importar sus capacidades sensoriales (W3C, 2018). Se pueden usar textos alternativos en imágenes, subtítulos en videos y mantener buen contraste entre texto y fondo. Así, personas con discapacidad visual o auditiva pueden acceder al contenido mediante tecnologías de apoyo, como lectores de pantalla o dispositivos auditivos.
- **Operable:** Se refiere a que la interfaz de usuario debe ser operativa para todos, permitiendo que cada persona use el método de entrada que le resulte más cómodo, como teclado, mouse, tacto, gestos o comandos de voz (W3C, 2018). Todas las funciones del sistema deben poder ejecutarse con estos métodos, de manera que cada usuario tenga acceso completo y flexible, sin importar el dispositivo o sus capacidades.
- **Comprensible:** Establece que la información y el funcionamiento del sistema deben ser claros y fáciles de entender (W3C, 2018). Esto significa usar un lenguaje sencillo, dar instrucciones claras, mostrar mensajes de error fáciles de interpretar y mantener una navegación coherente. De esta manera, usuarios con dificultades cognitivas, problemas de aprendizaje o bajo nivel de alfabetización digital pueden interactuar con el sistema más fácilmente, con menos esfuerzo y mayor claridad.

- Robusto: Señala que los contenidos y la interfaz del sistema deben ser interpretables de manera confiable por diferentes navegadores y tecnologías asistidas (W3C, 2018). Esto significa que se deben mantener estándares web, como HTML, CSS y ARIA, y usar correctamente las etiquetas y estructuras de la página. De esta manera, las tecnologías asistidas, como lectores de pantalla, pueden interpretar el contenido sin problemas, asegurando que todos los usuarios accedan a la información de forma efectiva.

En octubre de 2023, el W3C publicó la WCAG 2.2, incorporando nueve nuevos criterios adicionales de éxito para mejorar la accesibilidad de usuarios con distintas discapacidades. Entre estas mejoras se incluyen la visibilidad del enfoque de navegación, tamaños mínimos para la interacción táctil, soporte a gestos de arrastre y opciones de autenticación más accesibles. Estas actualizaciones buscan hacer la experiencia digital más inclusiva y asegurar que la información esté disponible de manera más efectiva y equitativa (W3C, 2023).

Los nueve criterios adicionales se agrupan dentro de cuatro directrices principales:

- Navegabilidad: Esta directriz busca que los usuarios puedan moverse y orientarse fácilmente dentro del contenido. Cuando un elemento recibe foco mediante navegación, por ejemplo, utilizando el teclado, debe ser claramente visible con un indicador visual perceptible, como un borde o contorno resaltado que ayude a identificar que elemento está activo (W3C, 2023).
- Modalidades de entrada: Esta directriz permite que los usuarios puedan interactuar con la interfaz utilizando distintos métodos de entrada, como el teclado, el mouse o la pantalla táctil. Para ello los elementos interactivos como botones, enlaces o controles de formulario deben tener un tamaño adecuado y ser fáciles de activar. Además, cuando una acción requiera arrastrar elementos, se debe ofrecer una alternativa más sencilla, como un clic o toque único, excepto en los casos en que el arrastre sea realmente indispensable (W3C, 2023).
- Comportamiento predecible: La interfaz debe funcionar de manera consistente y evitar cambios inesperados en el contexto, como redirecciones automáticas, apertura de nuevas ventanas sin aviso, cambios de foco que desorienten al usuario

o envío automático de formularios sin confirmación. Los elementos que cumplen la misma función en distintas páginas deben mantener el mismo nombre o etiqueta. Asimismo, los mecanismos de ayuda, como enlaces a soporte, contacto o centro de ayuda, deben ubicarse de forma consistente dentro del sistema (W3C, 2023).

- Asistencia de entrada: La interfaz debería ofrecer mecanismos que ayuden al usuario a prevenir, identificar y corregir errores, utilizando etiquetas claras, instrucciones fáciles de comprender y mensajes que orienten sobre cómo solucionarlos. Del mismo modo, en los procesos de autenticación es recomendable que estos sean accesibles y que no dependan únicamente de pruebas cognitivas complejas, favoreciendo así una experiencia más inclusiva y segura mediante mecanismos accesibles (W3C, 2023).

En términos funcionales, este principio implica reducir la carga cognitiva del usuario durante la interacción. En la práctica, puede observarse en mecanismos como el uso de códigos de verificación de un solo uso (OTP), que facilitan la autenticación sin exigir memorización compleja. En contraste, como ejemplos de pruebas de función cognitiva se pueden considerar acciones como recordar contraseñas extensas, transcribir caracteres mostrados en una imagen, resolver operaciones matemáticas o seleccionar imágenes específicas dentro de un rompecabezas visual, por ejemplo, en ciertos mecanismos tipo CAPTCHA. Por ello, cuando estos mecanismos se implementan, debe garantizarse la existencia de alternativas accesibles que mantengan tanto la seguridad como la facilidad de uso.

Modularidad

En un sistema web orientado a objetos, la modularidad se logra aplicando los principios SOLID, que guían la construcción de módulos independientes capaces de modificarse, probarse y reutilizarse sin afectar otras partes del sistema, reduciendo la dependencia entre componentes y manteniendo la cohesión interna de cada módulo (Martin, 2003).

Estos principios permiten que la aplicación sea más mantenible, flexible y fácil de evolucionar a lo largo del tiempo. A continuación, veamos cada uno de esos principios:

- Responsabilidad Única (SRP): Cada módulo o clase debe ocuparse de una única función o tarea. En un sistema web, esto significa que cada componente, como un controlador, un servicio o un componente de presentación, se centra en responsabilidades específicas, facilitando su comprensión, mantenimiento y pruebas (Martin, 2003).
- Abierto/Cerrado: Los módulos deben poder ampliarse con nuevas funcionalidades sin modificar el código existente. Esto permite añadir características como validaciones, filtros o reglas de negocio mediante herencia, composición o inyección de dependencias, manteniendo estable la aplicación (Martin, 2003).
- Sustitución de Liskov: Cualquier subclase debe poder reemplazar a su clase base sin generar fallos ni comportamientos inesperados. Así, componentes como servicios, controladores o interfaces pueden extenderse o sustituirse sin comprometer la funcionalidad general del sistema (Martin, 2003).
- Segregación de interfaces: Una clase no debe implementar métodos que no utiliza. Se recomienda crear interfaces pequeñas y específicas, de modo que cada módulo dependa únicamente de las operaciones que necesita, facilitando la comprensión y el mantenimiento de los servicios del sistema web (Martin, 2003).
- Inversión de dependencias: Los módulos deben depender de abstracciones (interfaces o clases abstractas) en lugar de implementaciones concretas, lo que permite realizar pruebas unitarias o de integración de manera aislada y favorece un diseño más flexible y desacoplado (Martin, 2003).

Seguridad Web

La seguridad en aplicaciones web es un aspecto fundamental para proteger la información y los servicios. Entre las referencias más reconocidas para el análisis de vulnerabilidades críticas se encuentra el OWASP Top 10:2021, que clasifica los riesgos más importantes y propone medidas de mitigación. A continuación, se detalla cada una de estas categorías:

- Pérdida de Control de Acceso: De acuerdo con OWASP Foundation (2021), para controlar esta vulnerabilidad es necesario trabajar desde el servidor. Es clave tener reglas claras de control de acceso, limitar el uso de CORS, asegurarse de que los

datos se manejen correctamente, usar tokens que expiren rápido y registrar cualquier intento fallido de acceso.

- Fallas Criptográficas: Cuando se manejan datos sensibles, como contraseñas, números de tarjetas de crédito o información personal, es fundamental prestar atención a cómo se envían o almacenan. Se recomienda utilizar siempre protocolos seguros, como HTTP, SMTP o SFTP, aplicar algoritmos y claves robustas, y evitar métodos de cifrado antiguos o funciones hash obsoletas. Para disminuir riesgos, los datos deben cifrarse tanto en tránsito como en reposo, emplear funciones confiables de derivación de contraseñas y no conservar información confidencial de forma innecesaria (OWASP Foundation, 2021).
- Inyección: Las aplicaciones pueden ser vulnerables a ataques de inyección si no se revisa cuidadosamente la información que ingresan los usuarios. Esto podría permitir que se ejecuten consultas inseguras o que se inserten datos maliciosos, poniendo en riesgo información sensible. Entre los tipos más frecuentes se incluyen SQL, NoSQL, comandos del sistema, ORM, LDAP y OGNL (OWASP, s.f.).

Para reducir estos riesgos, conviene separar siempre los datos de los comandos mediante APIs seguras o herramientas ORM parametrizadas. También es útil validar las entradas con listas blancas, escapar caracteres especiales en consultas dinámicas y limitar la cantidad de registros que se devuelven, así se disminuye la posibilidad de que la información se filtre (OWASP, s.f.).

- Diseño Inseguro: El diseño inseguro sucede cuando, desde el inicio del desarrollo, como en la arquitectura, los flujos de negocio o la definición de requisitos de seguridad, no se incluyen los controles necesarios. Esto puede dejar la aplicación expuesta a ataques, incluso si después la implementación se hace correctamente. Para evitar estos problemas, es recomendable seguir un ciclo de desarrollo seguro (S-SDLC). Entre las acciones sugeridas están modelar las amenazas, aplicar patrones de diseño seguros, separar capas y tenants, revisar los flujos críticos, controlar el uso de recursos y realizar pruebas unitarias e integradas que aseguren que la aplicación pueda resistir posibles ataques (OWASP, s.f.).
- Configuración de Seguridad Incorrecta: Una aplicación es vulnerable cuando tiene configuraciones inseguras, como permisos incorrectos, funciones

innecesarias activadas, contraseñas predeterminadas, manejo de errores que expone información sensible o software desactualizado.

Para evitar estos problemas, es recomendable aplicar procesos de hardening que sean automatizados y repetibles, mantener los entornos mínimos y consistentes, revisar y actualizar regularmente las configuraciones y los parches, segmentar la arquitectura de la aplicación, usar encabezados de seguridad y verificar de manera periódica que todas las configuraciones sean efectivas (OWASP, s.f.).

- Componentes Vulnerables y Desactualizados: Una aplicación corre riesgos cuando emplea componentes que están desactualizados o que no son seguros, como sistemas operativos, servidores, frameworks, bibliotecas y sus dependencias. También hay peligro si no se conoce la versión de estos componentes o si no se aplican los parches a tiempo.

Para evitar estos problemas, conviene llevar un inventario actualizado de todos los componentes, eliminar los que no se necesiten y utilizar únicamente fuentes oficiales. Además, es importante revisar de forma constante las posibles vulnerabilidades y aplicar las actualizaciones o mitigaciones necesarias durante todo el ciclo de vida de la aplicación (OWASP, s.f.).

- Fallas de Identificación y Autenticación: Una aplicación se vuelve vulnerable cuando permite el uso de contraseñas débiles o por defecto, guarda las credenciales de forma insegura, no implementa autenticación multi-factor, reutiliza o expone identificadores de sesión, o cuenta con procesos poco seguros para la recuperación de contraseñas. Este tipo de fallas facilita que usuarios no autorizados accedan a información confidencial o comprometan la integridad de la aplicación.

Para reducir estos riesgos, es recomendable implementar autenticación multi-factor, exigir contraseñas robustas siguiendo políticas modernas de complejidad y rotación, mostrar mensajes genéricos al recuperar credenciales, limitar los intentos de inicio de sesión y gestionar las sesiones de manera segura mediante identificadores aleatorios que expiren al cerrar sesión o después de periodos de inactividad (OWASP, s.f.).

- Fallas en el Software y en la Integridad de los Datos: Una aplicación es vulnerable cuando el software o los datos pueden ser alterados, por ejemplo, si se utilizan bibliotecas no confiables, los pipelines de CI/CD no están asegurados, las

actualizaciones no se verifican correctamente o los datos serializados pueden ser manipulados.

Para prevenir estos problemas, conviene usar firmas digitales para comprobar la integridad de los componentes, depender únicamente de repositorios confiables, analizar las dependencias con herramientas como OWASP Dependency Check, proteger los pipelines de CI/CD mediante controles de acceso y asegurarse de que los datos enviados estén cifrados o firmados electrónicamente (OWASP, s.f.).

- Fallas en el Registro y Monitoreo: Una aplicación corre riesgos cuando no dispone de registros claros, un monitoreo efectivo ni alertas, lo que dificulta darse cuenta de posibles brechas de seguridad, como fallos en los inicios de sesión, transacciones críticas o ataques en tiempo real.

Para reducir estos riesgos, es recomendable registrar los eventos más importantes incluyendo suficiente contexto, proteger los datos de los registros, garantizar que las transacciones clave puedan rastrearse, implementar alertas y mantener un monitoreo constante. Además, contar con un plan de respuesta y recuperación ayuda a reaccionar ante incidentes, y herramientas como ModSecurity o ELK pueden facilitar la correlación y supervisión de los registros de manera más eficiente (OWASP, s.f.).

- Falsificación de Solicitudes del Lado del Servidor (SSRF): Una aplicación es vulnerable a SSRF cuando solicita recursos remotos sin verificar correctamente las URLs proporcionadas por los usuarios, lo que permite que un atacante envíe solicitudes a destinos internos o protegidos.

Para prevenir este tipo de problemas, es recomendable validar y depurar todas las entradas, utilizar listas blancas de URLs, esquemas y puertos permitidos, desactivar redirecciones HTTP, segmentar las redes y aplicar firewalls restrictivos. Además, se debe proteger los servicios críticos frontales mediante cifrado o aislamiento según el nivel de riesgo (OWASP, s.f.).

Microservicio

Los microservicios son partes pequeñas de una aplicación que funcionan por su cuenta. Cada uno tiene su propio código y, normalmente, lo maneja un equipo pequeño,

así que se puede cambiar o mejorar sin afectar al resto de la aplicación. Además, cada microservicio se puede desplegar de manera independiente. A diferencia de los sistemas tradicionales, cada uno guarda sus propios datos y se comunica con los demás mediante APIs, sin que los otros vean cómo funciona por dentro. También se puede usar la tecnología que más convenga en cada servicio, según lo que necesite cada equipo (Microsoft, s.f.).

Arquitectura de Microservicios

La Arquitectura de Microservicios es una forma de crear aplicaciones que sean resilientes y fáciles de escalar. Son resilientes porque pueden seguir funcionando, aunque alguna parte falle, y fáciles de escalar porque se pueden expandir o reducir según la demanda sin afectar al resto de la aplicación. Además, cada servicio se puede actualizar o desplegar por separado.

Pasar a este tipo de arquitectura requiere pensar un poco diferente dado que, no basta con dividir la aplicación en partes más pequeñas. Cada servicio debe cumplir una función específica del negocio, con su propio almacenamiento y lógica, de manera que los equipos puedan desarrollarlo y probarlo de forma independiente. También es importante planear cómo se comunicarán los servicios, cómo se hará el monitoreo centralizado, cómo se desplegarán y cómo se gestionarán los errores, para que la aplicación siga funcionando bien mientras cada servicio evoluciona por su cuenta (Microsoft, s.f.-a).

Aunque esta arquitectura suele destacarse por su escalabilidad, resiliencia y la posibilidad de desplegar servicios de manera independiente, en la práctica también implica varios retos. No se trata solo de dividir una aplicación en partes más pequeñas, sino de gestionar correctamente la interacción entre ellas. Por ejemplo, cuando los servicios dependen unos de otros, la coordinación puede volverse compleja, especialmente durante el desarrollo y las pruebas. No siempre es sencillo reproducir escenarios donde intervienen varios servicios al mismo tiempo, y esto puede retrasar la detección de errores. Asimismo, si desde el inicio no se definen normas claras de comunicación, versionado y manejo de errores, el sistema puede perder coherencia con el tiempo. A ello se suma la latencia que

pueden generar las múltiples llamadas entre servicios y la dificultad de mantener la integridad de los datos en entornos distribuidos. Finalmente, este tipo de arquitectura exige un equipo con experiencia técnica y una cultura orientada a prácticas DevOps, automatización y monitoreo continuo. Sin estos elementos, los beneficios que promete el enfoque pueden verse reducidos por la complejidad operativa que introduce (Microsoft, s.f.-b).

3.2.5. Implementación de las Áreas, Procesos y Buenas Prácticas

El proyecto se ejecutó aplicando la metodología Scrum de acuerdo con el punto 3.2.3.1. donde se describen las etapas del proyecto.

Aunque Scrum propone una estructura estándar de sprints con entregables al final de cada iteración, en este proyecto se aplicó una versión adaptada de la metodología. Se incluyó un sprint inicial de planificación (Sprint 0) para recopilar información, definir el product backlog y diseñar de manera preliminar el sistema, así como un sprint final dedicado a realizar pruebas integrales, ajustes y desplegar el producto. Esta adaptación se llevó a cabo debido a la necesidad de tener mayor claridad sobre los elementos de análisis y diseño antes de avanzar hacia la construcción funcional del sistema, con el fin de reducir la probabilidad de retrabajos.

3.2.5.1. Sprint 0: Planificación inicial y diseño técnico.

En Scrum no existe oficialmente una fase llamada Sprint 0, aunque en la práctica profesional muchos equipos la usan como punto de partida. Su propósito es dejar listas las bases antes de iniciar los sprints, lo que incluye definir los roles de equipo, comprender la problemática del negocio, especificar los primeros requerimientos y preparar el entorno técnico de trabajo. Además, en esta etapa, se estableció una propuesta técnica preliminar del sistema y se delimitó el alcance funcional inicial.

Durante este sprint se definió el equipo de desarrollo y sus competencias. También se realizó el levantamiento y análisis inicial de requerimientos elaborando el backlog

inicial del proyecto. Así mismo estableció la planificación temporal mediante un diagrama de Gantt, se diseñó la arquitectura preliminar del sistema, y se eligió el stack tecnológico. Finalmente se analizó cómo funciona actualmente el proceso de supervisión (AS-IS) y se propuso mejoras al procedimiento de supervisión (TO-BE) para que sea más eficiente.

Scrum team del proyecto

El proyecto fue desarrollado aplicando la metodología Scrum, adaptada a las necesidades de la entidad. Para ello, se conformó un equipo multidisciplinario conformado por personal de la entidad y por profesionales externos, los cuales trabajaron de manera coordinada durante todo el desarrollo del proyecto.

A continuación, describimos los roles del equipo que participó en el proyecto, indicando sus responsabilidades y la manera en que contribuyeron al desarrollo el sistema.

Product Owner: El rol de Product Owner fue asumido por el Jefe de la Unidad de Supervisión de Aseguramiento y Prestaciones Económicas (USAPE), quien representó al área usuaria y fue responsable de definir las necesidades del negocio, priorizar los requerimientos y validar las funcionalidades desarrolladas.

Scrum Master: El rol de Scrum Master fue desempeñado por el Analista Programador de la Subgerencia de Auditoría, Supervisión de Aseguramiento y Prestaciones Económicas (SGASPE), quien coordinaba la comunicación entre el Product Owner, los stakeholders y el equipo de desarrollo. Además, actuó como facilitador de información y recursos, apoyando al equipo en la obtención de datos o contactos necesarios para la implementación del sistema y coordinando reuniones con el PO, stakeholders y el área usuaria.

Developers: Este equipo estaba conformado por un desarrollador backend, un desarrollador frontend, un analista funcional y un administrador de base de datos, quienes trabajaron de manera coordinada para cumplir con los objetivos del proyecto.

La siguiente tabla presenta la conformación del Scrum Team del proyecto, indicando los roles definidos, el cargo que desempeñaron dentro de la organización y sus principales responsabilidades según la metodología Scrum adaptada.

Tabla 6

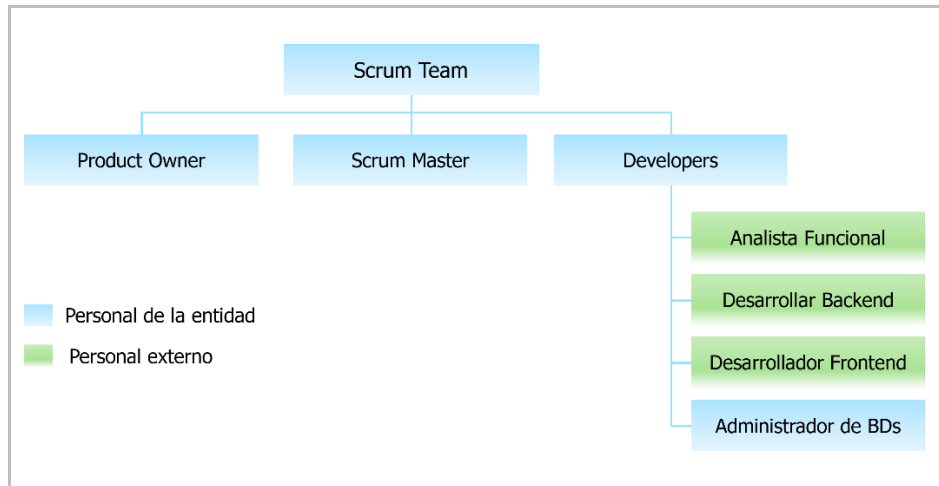
Equipo de desarrollo

Rol	Cargo en la organización	Descripción
Product Owner (PO)	Jefe de la Unidad de Supervisión de Aseguramiento y Prestaciones Económicas (USAPE)	Representar al área usuaria, definir las necesidades del negocio, priorizar los requerimientos y validar las funcionalidades desarrolladas.
Scrum Master	Analista Programador de la Sub Gerencia de Auditoria, Supervisión de Aseguramiento y Prestaciones Económicas (SGASPE)	Coordinar la comunicación entre PO, Stakeholders y equipo de desarrollo.
Developers	Equipo de desarrollo (Analista funcional, Desarrollador Backend, Desarrollador Frontend y Administrador de Base de Datos)	Levantar información, diseñar, desarrollar, implementar y probar las funcionalidades del sistema.

Nota. Elaboración propia

La Figura 2 ilustra la organización del Scrum Team del proyecto, destacando los roles involucrados y su relación dentro de la metodología Scrum adaptada.

Figura 2
Organigrama del scrum team



Nota. Elaboración propia

Levantamiento y análisis de requerimientos

Durante el presente sprint, como parte del levantamiento inicial de requerimientos del sistema SSAIS, se llevaron a cabo reuniones con el Product Owner (PO) para entender las necesidades del negocio. Asimismo, se realizaron entrevistas con el área usuaria, es decir, con los integrantes de la Unidad de Supervisión de Aseguramiento y Prestaciones Económicas (USAPE), para identificar las deficiencias del proceso actual y determinar los procesos susceptibles de mejora que el sistema podría optimizar. Por otro lado, se revisaron documentos internos y normas vigentes para saber cómo se hace actualmente y que se espera que haga el sistema. Con esta información se pudieron recopilar los primeros requerimientos funcionales del sistema, los cuales fueron documentados en el Product Backlog inicial mediante historias de usuario HU), priorizadas de acuerdo a la importancia del negocio y el impacto esperado en el proceso de supervisión.

Estas sesiones permitieron, además:

- Definir la Visión del sistema asegurando que las funcionalidades identificadas estén alineadas a los objetivos establecidos en la sección 3.4.1.

Visión del sistema: Convertirse en una herramienta que apoye de manera efectiva el proceso de supervisión de afiliaciones indebidas, permitiendo hacer un

seguimiento detallado de cada expediente y mantener su trazabilidad, de modo que el trabajo se realice correctamente y la entidad no sufra pérdidas económicas.

- Identificar a los actores principales en el proceso tales como Jefe USAPE, Analista USAPE y Supervisor.
- Identificar las funcionalidades más importantes, entre ellas:
 - Bandeja de casos Jefe USAPE
 - Bandeja de casos Analista USAPE
 - Bandeja de casos Supervisor

A partir de las actividades de recopilación, levantamiento y priorización de requerimientos con el Product Owner, se elaboró un Product Backlog inicial, donde se agrupó los requerimientos funcionales ya priorizados. Este backlog fue la base para organizar la planificación de los sprints posteriores, así como para el diseño de la arquitectura del sistema.

Tabla 7
Product Backlog inicial

Épica	Historia de usuario HU	Descripción	Prioridad
		Reuniones con stakeholders, levantamiento de información, definición del backlog	
Gestion de casos por el rol JEFE USAPE	HU-01 Listar casos	Visualizar lista general de casos	Alta
	HU-02 Buscar casos	Filtrar caso por diferentes criterios de búsqueda	Alta
	HU-03 Asignar casos	Asignar casos a Analista USAPE	Alta
	HU-04 Ver Trazabilidad	Permite visualizar el historial cronológico de las acciones ejecutadas sobre cada caso	Baja

Gestion de casos por el rol ANALISTA USAPE	HU-05	Generar alertas automáticas al detectar patrones sospechosos en los casos	Baja
	HU-06	Reasignar Casos a Analistas	Baja
	HU-07	Reporte dinámico de casos	Media
	HU-08	Listar casos asignados	Alta
	HU-09	Buscar casos	Alta
	HU-10	Asignar casos	Media
	HU-11	Analizar integridad de afiliación	Media
Gestion de casos por el rol SUPERVISOR	HU-12	Ver casos asignados	Alta
Pruebas funcionales, ajustes finales, validación integral y despliegue en producción			
Nota. Elaboración propia			

El Product Backlog final completo del proyecto, organizado por Épicas, Sprints, Historias de Usuario y Story Points, se presenta en el Anexo A.

Además de los requerimientos funcionales, en este sprint se identificaron los requerimientos no funcionales. Estos definen la calidad del sistema y sus restricciones y son importantes para que todo funcione correctamente, sea seguro y fácil de mantener.

Entre los más importantes tenemos:

Tabla 8
Requerimientos no funcionales

Criterio de calidad	Descripción
Disponibilidad	El sistema debe estar operativo el 99% de todo el tiempo.
Seguridad	Protección de los datos, control de accesos y encriptación de información sensible.
Rendimiento	El sistema debe responder de manera eficiente en menos de 3 segundos por operación bajo la carga esperada de hasta 1000 usuarios y las transacciones asociadas.
Escalabilidad	El sistema debe poder manejar hasta 1000 usuarios concurrentes y las transacciones generadas por ellos sin afectar su desempeño.
Mantenibilidad	Las actualizaciones, correcciones y nuevas funcionalidades deben poder implementarse en menos de 5 días hábiles sin afectar el funcionamiento del sistema.
Compatibilidad	Funciona correctamente con la infraestructura existente incluyendo la arquitectura de microservicios y otros sistemas relacionados.

Nota. Elaboración propia

Dado que la organización ya contaba con una arquitectura de microservicios, los criterios de calidad definidos ya se encontraban cubiertos por la infraestructura existente. El

desarrollo del sistema SSAIS se adaptó a este entorno, respetando las restricciones del ecosistema y facilitando su integración con los servicios y bases de datos disponibles. Por ello, los criterios de calidad se utilizaron como referencia para las decisiones de diseño y la priorización de funcionalidades, sin que fuera necesario modificar la arquitectura base.

Análisis del proceso actual (AS-IS)

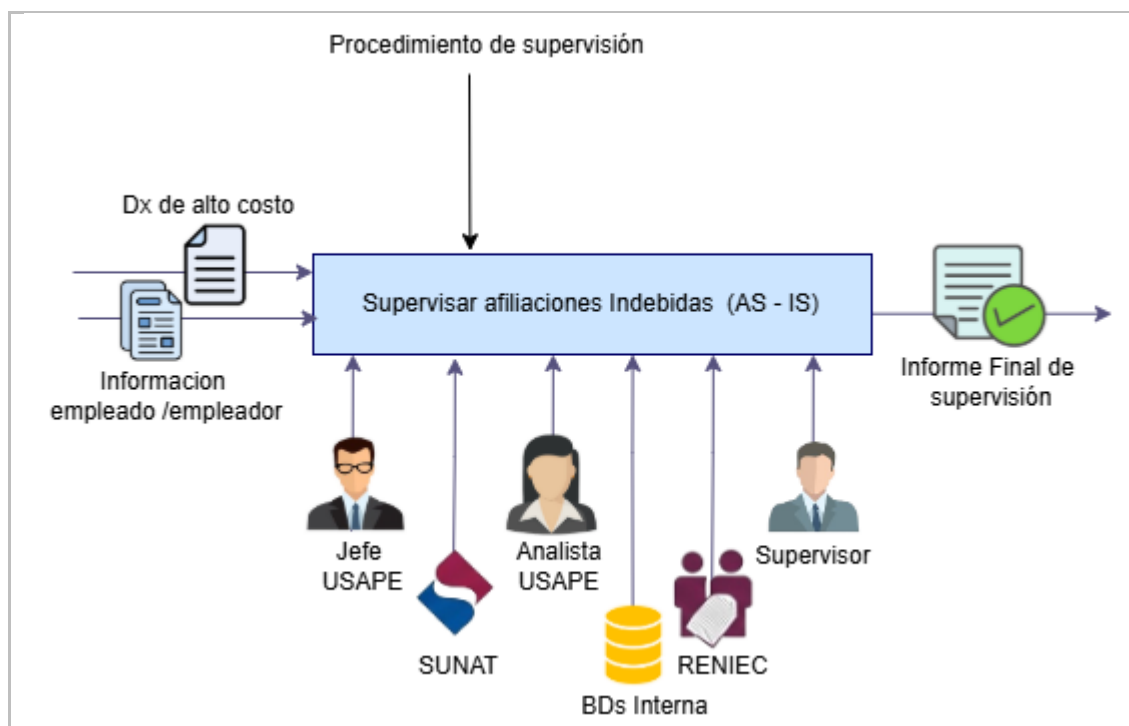
Descripción del proceso de supervisión

El siguiente diagrama muestra, a nivel general, el flujo de actividades, los actores y las fuentes de información del proceso. Sirve como base para analizar la situación actual y detectar oportunidades de mejora en la solución propuesta.

La Figura 3 muestra el diagrama IDEF0 del proceso actual:

Figura 3

Diagrama Idef0 AS-IS



Nota. Elaboración propia

El proceso de supervisión de afiliaciones indebidas las desarrolla tres actores principales: El Jefe de la Unidad de Supervisión de Aseguramiento y Prestaciones Económicas (USAPE), el Analista Supervisor y el Supervisor, quienes se encargan de coordinar, documentar y ejecutar todas las actividades de supervisión y seguimiento de los casos

El proceso comienza con la actividad Detectar posible inconsistencia, la cual puede originarse a partir de una denuncia de terceros o de oficio, es decir, por iniciativa propia de la entidad. En los casos iniciados de oficio, se envía un correo al personal técnico para que consulte las bases de datos internas para que se obtenga un conjunto de diagnósticos de alto costo de los afiliados, que luego serán analizados.

Cuando se identifica un posible caso, el Jefe USAPE realiza la actividad Solicitar información y documentación, mediante la cual solicita al Analista Supervisor, que recopile la información y la documentación del empleado y del empleador. Para esto se utilizan los canales institucionales disponibles, como los sistemas internos o entidades externas autorizadas, por ejemplo, SUNAT y RENIEC. La solicitud se realiza mediante medios tradicionales, como memorándums o correos electrónicos.

Posteriormente, el Analista Supervisor desarrolla la actividad Consultar información administrativa, recopilando y registrando los datos del empleado y empleador usando para ello fuentes tanto internas como externas, necesarios para analizar el caso.

Como parte del proceso, el Analista supervisor realiza la actividad Generar y remitir documentación, preparando la documentación del proceso de supervisión, como la Orden de Supervisión, la Carta de Supervisión, el Acta de Supervisión, el Formato de Cargo de Supervisión, el Acta de Notificación Fallida y el Acta de Impedimento. Todo esto se hace de manera manual, con impresión de documentos y manejo de archivos físicos. Una vez listos, los documentos se envían al Jefe USAPE.

Con la documentación disponible, el Jefe USAPE realiza la actividad Asignar supervisor responsable, designando el caso a un Supervisor, quien queda a cargo de la

supervisión y de recibir los documentos necesarios. Esta asignación se hace de manera manual, enviando los expedientes por correo electrónico.

El Supervisor, al Recepcionar casos asignados, revisa la documentación recibida y organiza las actividades de supervisión. A continuación, realiza la actividad Notificar inicio de supervisión, comunicando al empleador o al asegurado sobre el inicio de la supervisión, entregando la Orden de Supervisión y la Carta de Presentación en el domicilio indicado. Si no logra contactarlos, se lleva a cabo la actividad Registrar notificación fallida, dejando constancia del intento en el Acta de Notificación Fallida.

Seguidamente, el Supervisor desarrolla la actividad Realizar visita de supervisión en campo realiza la visita de supervisión en campo, para ello se desplaza ya sea en el centro de trabajo del asegurado o en el domicilio fiscal del empleador, para verificar la información, hacer entrevistas y recabar evidencia de manera presencial. Si durante la visita surgen impedimentos o demoras, como la negativa de acceso o la ausencia del responsable, el Supervisor realiza la actividad Registrar impedimento o demora registrando lo ocurrido en el Acta de Impedimento y coordina la Programación de nueva visita dentro de los plazos establecidos.

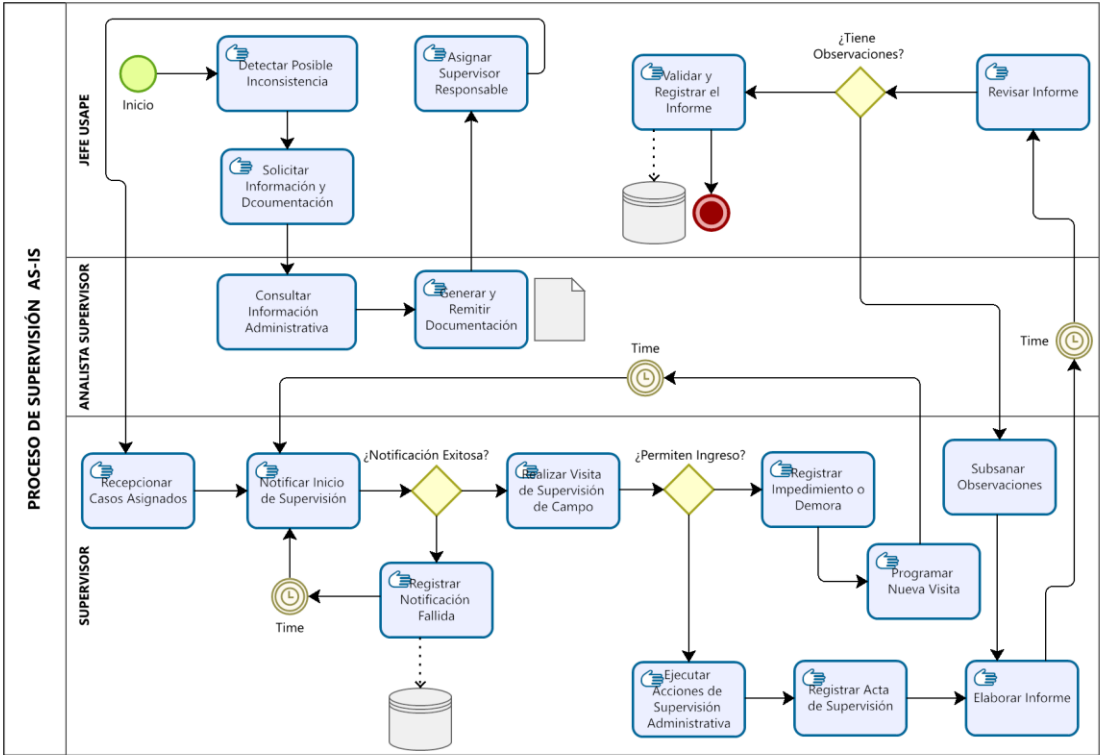
Durante la supervisión administrativa, el Supervisor realiza la actividad Ejecutar acciones de supervisión administrativa, para ello revisa la documentación disponible y constata los hechos. Como resultado, se realiza la actividad Registrar acta de supervisión, en el cual se registran todas las observaciones, declaraciones y hallazgos. También se incluyen las firmas correspondientes, o se deja constancia si alguien se niega a firmar.

Finalmente, el Supervisor realiza la actividad Elaborar informe, por el cual elabora un informe administrativo que resume los hallazgos y la evidencia recopilada, sin incluir recomendaciones ni sanciones. El informe se envía al Jefe USAPE, quien desarrolla la actividad Revisar informe, mediante el cual revisa para asegurarse de que esté completo y consistente. Si encuentra observaciones lo devuelve al Supervisor y este realiza la actividad Subsanan observaciones, realizando las correcciones necesarias. Una vez conforme, el Jefe USAPE realiza la actividad Validar y registrar el informe, registrándolo

como documento final en la base de datos interna, concluyendo el procedimiento de supervisión.

La Figura 4 presenta el diagrama BPMN correspondiente al proceso de supervisión de afiliaciones indebidas en su estado actual (AS-IS). En él se describen las principales actividades del proceso, los roles que intervienen y la secuencia de acciones que se siguen desde la identificación del caso hasta la elaboración del informe final de supervisión.

Figura 4
Diagrama BPMN AS-IS



Nota. Elaboración propia

Actividades realizadas para el análisis

Antes de desarrollar la solución propuesta, se analizó el proceso de supervisión de afiliaciones indebidas con el fin de entender cómo se llevaba a cabo en la práctica e

identificar ineficiencias y oportunidades de mejora. Para ello, se realizaron distintas actividades orientadas a conocer el funcionamiento del proceso en su estado actual, entre las cuales se incluyeron:

- Revisión de la documentación institucional y de la normativa interna, incluyendo resoluciones, leyes y regulaciones relevantes para el proceso de supervisión.
- Identificación de los roles y responsabilidades asignadas a cada uno de ellos.
- Se observó directamente el desarrollo de los flujos de trabajo en operación, prestando atención a la manera en cómo se realizan las tareas en la práctica, cómo se relacionan los distintos roles y cómo se transmite la información.

Actividades clave del proceso AS – IS

A continuación, se muestra un resumen de las actividades principales del proceso de supervisión. En este proceso, cada actividad está a cargo de roles específicos, quienes realizan las tareas indicadas, desde identificar inconsistencias hasta elaborar y registrar el informe final.

Tabla 9
Resumen actividades clave del proceso AS-IS

Actividad clave	Actor	Descripción
Detectar posibles inconsistencias	Jefe USAPE	El proceso se inicia por denuncia o de oficio mediante revisión manual de información interna.
Solicitar y recopilar información	Jefe USAPE /Analista Supervisor	El Jefe solicita al Analista información del afiliado y empleador y prepare la documentación de supervisión.
Notificación y visita de supervisión	Supervisor	El Supervisor notifica presencialmente el inicio del procedimiento, entrega Orden de Supervisión y recaba evidencia.

Registro de hallazgos	Supervisor	Los hallazgos, declaraciones se registran en actas (Acta de Supervisión, Acta de Impedimento, Acta de Notificación Fallida)
Cierre y registro del Informe	Jefe USAPE	Revisa, valida y registra el informe final en la base de datos interna.
<i>Nota.</i> Elaboración propia		

Problemas identificados

Como resultado de este análisis, se identificaron varios problemas que afectan la eficiencia, la trazabilidad y la confiabilidad del proceso:

- La obtención de los casos para supervisión se realizaba de manera manual, el área usuaria solicita la información al personal técnico por correo electrónico, quien realiza consultas directas en la base de datos institucional y entrega los resultados en archivos externos. Este procedimiento genera dependencia operativa, retrasa el acceso a la información y aumenta la probabilidad de errores al manipular los datos.
- La recopilación de la información tanto del empleador como del asegurado se realiza de manera manual, principalmente a través de consultas en portales web oficiales. Esta forma de trabajo incrementa el tiempo necesario para la atención de cada caso y hace que el proceso no se ejecute de la misma manera.
- La elaboración de los documentos requeridos para el trabajo de campo del verificador también se lleva a cabo de manera manual, generando reprocesos y aumentando la probabilidad de inconsistencias en la información registrada.
- La asignación de supervisores a los casos se hacía de forma manual por correo electrónico. En este proceso, los casos pasaban primero del Jefe de USAPE al Analista Supervisor y luego al Supervisor correspondiente, adjuntando los expedientes como archivos separados. Este método creaba dependencia operativa y complicaba el seguimiento de los casos, además de obligar a hacer controles manuales con reportes periódicos. Como resultado, no se podía tener una visión

general del avance de los casos ni información inmediata sobre la carga de trabajo de cada supervisor.

- La trazabilidad de las acciones y el seguimiento de los casos se realizaban a través de registros manuales en Excel. Esto hacía difícil reconstruir el historial completo de cada caso, limitaba la supervisión y complicaba la realización de auditorías.

A continuación, se muestra una tabla que resume los problemas identificados, indicando su descripción y el impacto que tienen en el proceso de supervisión de afiliaciones indebidas.

Tabla 10

Resumen de problemas identificados en el proceso actual (AS-IS)

Área/Proceso	Problema identificado	Impacto
Obtención de casos para supervisión	Generación manual de listado de casos solicitados por el área usuaria.	Retrasos en el acceso a la información y dependencia del área técnica.
Recopilación de información del empleador y afiliado	Búsqueda manual en fuentes internas y portales web oficiales.	Mayor tiempo de atención por caso y dificultad para estandarizar el proceso.
Asignación de supervisores	Envío de casos mediante correo electrónico con expedientes adjuntos.	Dificultad para controlar la carga de trabajo y seguimiento de casos.
Documentación para trabajo de campo	Elaboración manual de documentos y expedientes.	Reprocesos y riesgos de inconsistencias.
Seguimiento de casos	Reportes manuales en Excel no integrados al flujo operativo.	Información desactualizada y control limitado del avance de los casos.

Trazabilidad de casos	Registro manual de todas las acciones realizadas en Excel.	Imposibilidad de reconstruir el historial completo y dificultades en auditorias.
Generación de reportes de gestión	Elaboración manual de reportes operativos.	Dependencia del área técnica y demora en la toma de decisiones.

Nota. Elaboración propia

Propuesta de mejora del proceso (TO-BE)

A partir del análisis del proceso actual (AS IS) y los requerimientos funcionales identificados y detallados más adelante como HU, se diseñó un proceso futuro optimizado (TO BE) para la supervisión de afiliaciones indebidas. De esta manera se busca reducir la carga operativa manual, mejorar la trazabilidad de los casos, reforzar los controles y mecanismos de seguimiento, de manera que estos requerimientos funcionales apoyen al supervisor en la evaluación de la condición de afiliación, permitiéndole concentrarse en el análisis del caso y determinar si se trata de una afiliación indebida.

En el proceso TO-BE, los casos de supervisión se obtienen directamente desde el sistema en la bandeja respectiva de los diferentes actores, eliminando la necesidad de depender del área técnica para generar listados. El personal autorizado puede acceder a todos los casos mediante bandejas de trabajo automáticas, aplicando filtros por fechas, especialidad y unidad administrativa, lo que facilita identificar posibles inconsistencias de manera oportuna.

La información del afiliado y del empleador se concentra en una sola plataforma, integrando datos de fuentes internas y externas. Esto evita consultas repetidas y asegura que la información utilizada en el análisis del caso sea consistente.

Además, la elaboración de la documentación que se requiere para la supervisión se agiliza con la generación automática de los formatos administrativos del proceso de

Las figura 5 ilustra algunos ejemplos de estos formatos de supervisión, mostrando la Carta de Presentación y Orden de Supervisión, que se utilizan durante el procedimiento.


 "Decreto de la Igualdad de Oportunidades para Mujeres y Hombres"
 "Año del Bicentenario, de la consolidación de nuestra Independencia, y de la
 conmemoración de las heroicas batallas de Junín y Ayacucho"

Carta de Presentación

Magdalena del Mar, 8 de mayo del 2024

Carta N° 100-2025-SCA-00000 - 100-001

Tipo y Número de Documento: RUC. 20549029303

Domicilio: Calle Las Begonias 475- San Miguel

Apellidos y Nombres del Representante Legal: JUAN PEREZ

Tipo y Número de Documento de Identidad: 04537112

Presente. -

Por medio de la presente, nos dirigimos a ustedes con el fin de hacer referencia a nuestra **Orden de Supervisión N° 100-2025-SCA-00000 - 100-001**, de fecha **08/05/2024**, con la finalidad de presentar al personal supervisor que dará inicio al procedimiento de supervisión de la condición de asegurados.

Supervisor	Tipo y número del Documento de Identidad
Anthony Velazquez Vides	DNI: 43600123
Nickol Pérez Torres	DNI: 75136050

Si desea verificar la identidad en la persona designado, puede acceder a la siguiente dirección electrónica:

<http://www.massalud360.com.pe/agencias-y-oficinas-de-seguros> y/o comunicarse telefónicamente al número 1235000, anexos 1234, en el horario de 08:00 a 16:00 horas, con el fin de corroborar su identidad en las Unidades de Supervisión y Aseguramiento.

Firma y Sello
 Jefe de Unidad de Supervisión y Prestaciones Económicas


 "Decreto de la Igualdad de Oportunidades para Mujeres y Hombres"
 "Año del Bicentenario, de la consolidación de nuestra Independencia, y de la
 conmemoración de las heroicas batallas de Junín y Ayacucho"

Orden de Supervisión

Magdalena del Mar, 8 de mayo del 2024

Carta N° 100-2025-SCA-00000 - 100-001

Tipo y Número de Documento: RUC. 20549029303

Domicilio: Calle Las Begonias 475- San Miguel

Apellidos y Nombres del Representante Legal: JUAN PEREZ

Tipo y Número de Documento de Identidad: 04537112

Presente. -

Nos dirigimos a usted a fin de informarle que, en cumplimiento de lo dispuesto por la Ley N.° 19253, se procederá a iniciar el proceso de verificación correspondiente para el(los) siguiente(s) asegurado(s):

N°	Apellidos y nombres	DNI	Fecha de afiliación	Observaciones
1				
2				
3				

En virtud de lo establecido en nuestras normas internas y procedimientos de supervisión, solicitamos poner a disposición del personal encargado de la supervisión los siguientes documentos:

N°	Documentos Prioritarios	N°	Documentos Complementarios
1	Copia del contrato laboral vigente o último suscrito	1	Registros internos de funciones y responsabilidades
2	Comprobantes de pago o recibos de los últimos 6 meses	2	Estados de cuenta o registros de depósitos de CTS
3	Listado de asistencia o control horario	3	Inventarios, facturación o comprobantes de venta y compra

Firma y Sello
 Jefe de Unidad de Supervisión y Prestaciones Económicas

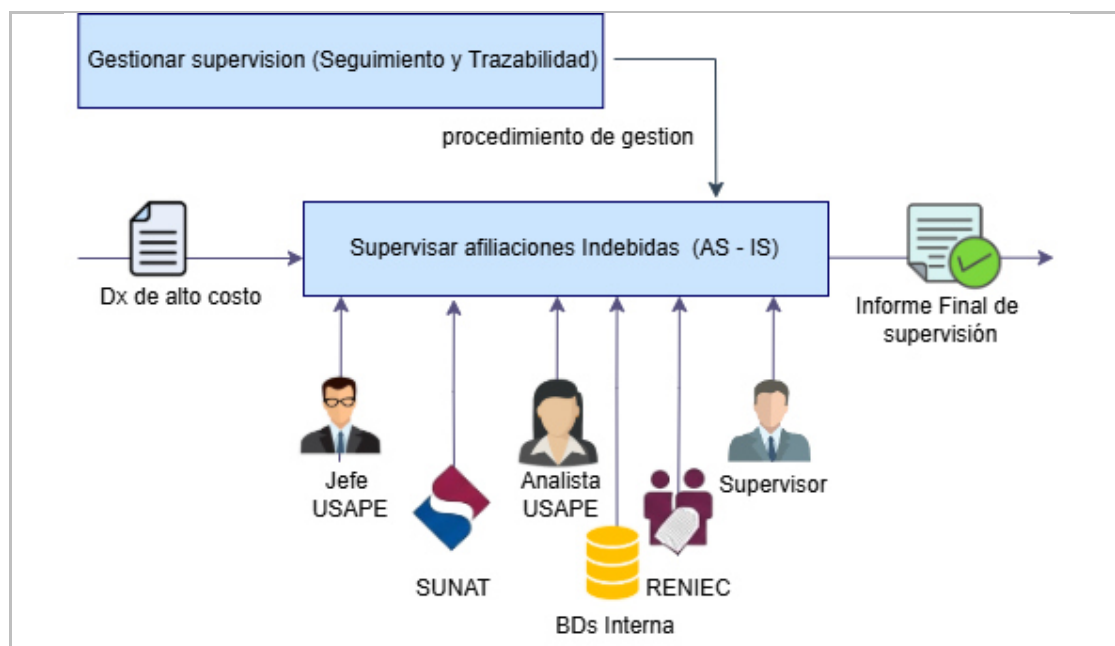
60

La asignación de casos se realiza directamente desde el sistema, lo que permite a los responsables distribuir la carga de trabajo de forma equilibrada y tener visibilidad en tiempo real del estado de cada caso. Además, durante la supervisión, el sistema guarda automáticamente las acciones realizadas, generando un historial cronológico que facilita el seguimiento y la trazabilidad.

Finalmente, el proceso TO-BE permite la visualización del historial de los casos y de su avance. Esto facilita que los distintos niveles jerárquicos consulten en tiempo real el estado de los casos, los responsables asignados y las principales acciones realizadas. Si bien no se generan reportes impresos de trazabilidad, la información se encuentra disponible directamente en el sistema para su consulta, lo que mejora la supervisión y apoya una toma de decisiones más eficiente.

La Figura 6 muestra el diagrama IDEF0 del proceso propuesto (TO-BE) de supervisión de afiliaciones indebidas. En él se describen las principales entradas, controles, mecanismos y salidas del proceso.

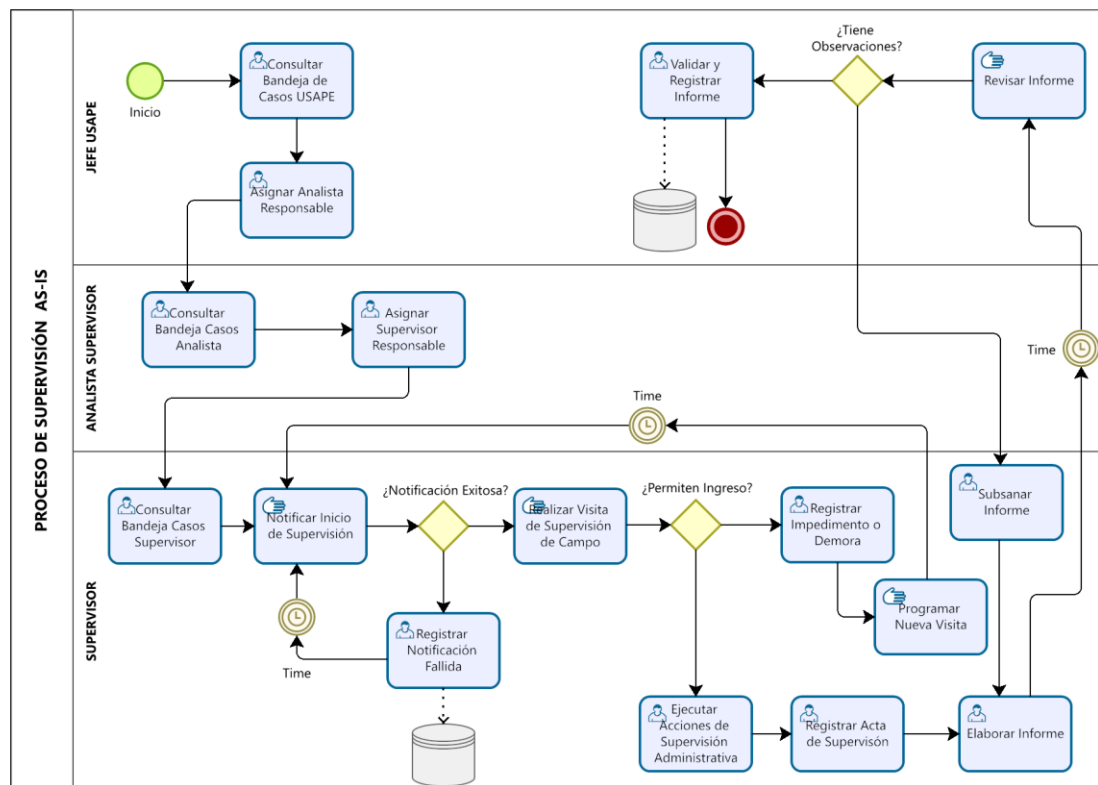
Figura 6
Diagrama Idef0 TO-BE



Nota. Elaboración propia

A continuación, se presenta el diagrama BPMN TO-BE del proceso de supervisión de afiliaciones indebidas, el cual muestra el proceso optimizado considerando las mejoras propuestas.

Figura 7
Diagrama BPMN TO-BE



Nota. Elaboración propia

La siguiente tabla muestra las actividades clave del proceso TO-BE, la funcionalidad desarrollada para cada una y su descripción correspondiente. En conjunto estas funcionalidades permiten apoyar en el proceso de supervisión de forma más eficiente y eficaz, permitiendo organizar y centralizar mejor la información, definir con mayor precisión las responsabilidades y dar un seguimiento adecuado de los casos.

Tabla 11*Funcionalidades clave del proceso TO-BE*

Actividad Clave	Funcionalidad desarrollada (HU)	Descripción
Detectar posibles inconsistencias	Bandejas de trabajo automatizadas	Los casos se obtienen directamente del sistema, eliminando la generación manual de listados y la dependencia del área técnica
Consultar información administrativa	Consulta centralizada de datos del empleado y empleador	Facilita el análisis integral del caso para la supervisión integrando fuentes internas y externos
Asignación supervisor responsable	Asignación sistematizada de supervisores	Los responsables asignan casos desde el sistema lo que permite una mejor distribución de la carga de trabajo y control del estado de los casos
Documentación y remitir documentación	Generación automática de formatos del proceso de supervisión con datos precargados	Reduce tareas manuales y errores administrativos
Seguimiento de casos	Registro automático de actividades del avance del proceso	El sistema registra las acciones realizadas en cada etapa del proceso lo cual permite saber el estado actual del caso
Trazabilidad	Registro y consulta del historial de casos	Registro automático del historial de actividades de cada caso, que permite el seguimiento y control oportuno del proceso.

Nota. Elaboración propia**Diseño de la arquitectura de la Solución**

El sistema SSAIS se diseñó e implementó usando la arquitectura de microservicios definida por la organización. Esta arquitectura define los lineamientos tecnológicos, estructurales y de integración que se deben seguir al desarrollar los sistemas institucionales.

En este proyecto, se buscó aplicar esta arquitectura de manera práctica, asegurando que el sistema se integrara correctamente a través de servicios REST y la adecuada separación de responsabilidades entre los componentes.

Con el enfoque de microservicios que se adoptó, la solución se organizó de forma lógica en estas capas:

- Capa de presentación: Se usa Angular y Angular Material para las aplicaciones frontend. Estas aplicaciones permiten que los usuarios interactúen con el sistema, hacen una validación inicial de los datos y se comunican con los servicios backend mediante API REST.
- Capa de servicios o lógica de negocio: Se compone de microservicios desarrollados en Java usando Spring Boot y Spring Framework. Estos microservicios aplican las reglas de negocio, manejan los flujos del sistema y ofrecen servicios REST que otros componentes pueden usar.
- Capa de persistencia de datos: Se usa Hibernate/JPA para mapear los objetos del sistema a la base de datos Oracle. Esta capa permite acceder y manejar la información de manera controlada, siguiendo un enfoque transaccional.
- Capa de integración: Se encarga de que los microservicios se comuniquen entre sí y con sistemas externos. Para esto, se usan servicios RESTful como la forma estándar de intercambiar información.

Esta separación en capas ayudó a mantener el sistema modular. También facilitó su mantenimiento, permitió escalarlo con facilidad y hacer mejoras de manera progresiva.

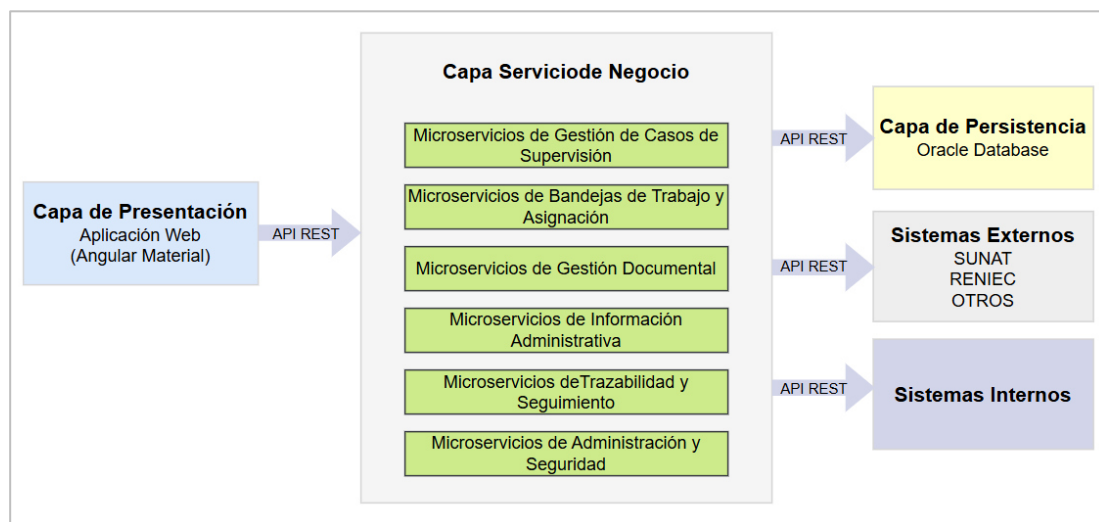
Microservicios principales del sistema

Dentro de la arquitectura definida, se identificaron los siguientes microservicios principales:

- Microservicio de gestión de casos: se encarga de registrar, dar seguimiento y actualizar los casos de supervisión.
- Microservicio de integración de información: obtiene y consolida datos de fuentes internas y externas.
- Microservicio de gestión documental: se ocupa de generar y almacenar los documentos relacionados con la supervisión.
- Microservicio de administración y seguridad: gestiona usuarios, roles y accesos al sistema.

La Figura 8 muestra el diagrama de la arquitectura de la solución SSAIS, donde se presentan las capas del sistema, los microservicios implementados y su integración con sistemas internos y externos mediante servicios REST.

Figura 8
Arquitectura del SSAIS



Nota. Elaboración propia

Decisiones técnicas y estrategias de implementación

La solución propuesta se basa en una arquitectura de microservicios que permite integrar información proveniente de distintas fuentes con el fin de automatizar el proceso de supervisión. Su implementación se estructura en tres componentes principales:

- Orquestación y Consolidación de Datos en Capa de Persistencia: La solución centraliza la información de distintas bases de datos sin necesidad de duplicarla. En lugar de hacer varias solicitudes desde el backend, se utiliza una consulta compleja y única en Oracle que une tablas de diferentes bases internas mediante Database Links, y vistas. Esta consulta reúne los datos del empleado, del empleador y la información asistencial, entregando al microservicio un conjunto de resultados listo para usar.

- **Estrategia de Optimización de Rendimiento:** Uno de los hallazgos críticos durante la fase de diseño y pruebas de concepto fue la demora en procesar aproximadamente 800k registros. Esto ocurría porque Oracle tenía que combinar datos de varias fuentes usando DB Links, lo que obligaba a mover grandes volúmenes de información entre nodos antes de completar los joins.
En lugar de cambiar el código del microservicio, se decidió reducir los datos desde la fuente. Aplicando un filtro de tres meses, Oracle pudo descartar gran parte de los registros, lo que disminuyó el tráfico entre servidores y aceleró notablemente el procesamiento de la consulta distribuida.
- **Automatización y Trazabilidad:** El sistema permite que la información fluya de manera continua dentro del flujo secuencial de trabajo. Los datos que llegan de las APIs se mapean directamente a los documentos de supervisión, lo que ayuda a evitar errores por transcripción manual. Además, cada cambio queda registrado con el usuario y la fecha correspondiente, de modo que todo el proceso de supervisión se pueda seguir y verificar en cualquier momento.

Selección de herramientas de gestión

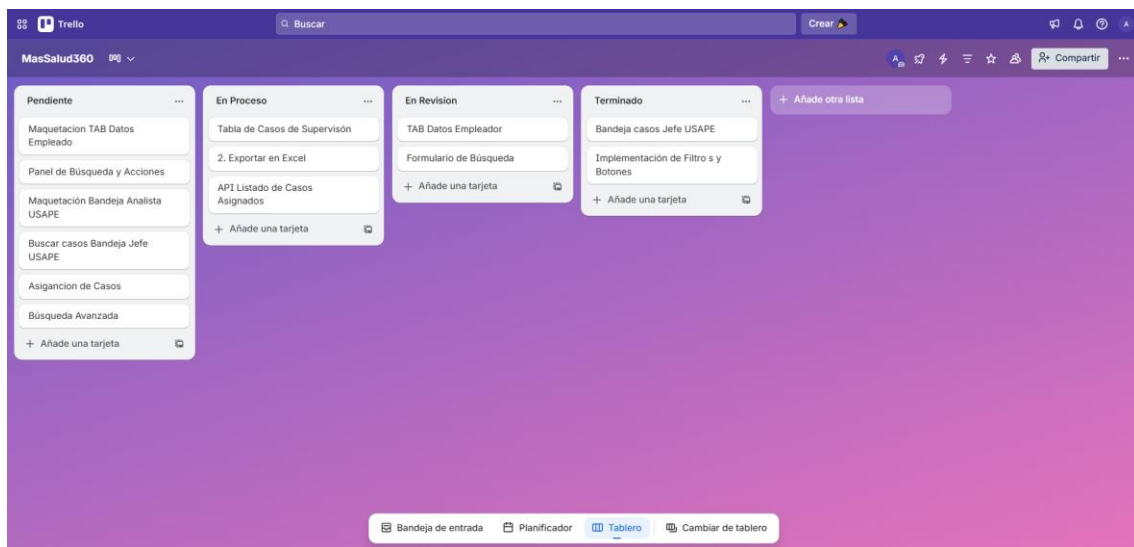
La organización del equipo y el seguimiento de las actividades se estructuraron de la siguiente manera:

- **Gestión de Tareas y Flujo de Trabajo:** Se decidió utilizar Trello como la herramienta principal para organizar el flujo de trabajo. Esto se eligió porque permite crear tableros Kanban, lo que facilitó ver de manera clara en qué estado se encontraba cada requerimiento (Backlog, En Proceso, Pruebas y Finalizado). Además, su flexibilidad ayudó a que los desarrolladores se adaptaran con mayor facilidad a los cambios en las reglas de negocio durante todo el ciclo del proyecto.
- **Gestión de versiones y repositorio:** De manera complementaria, se utilizó GitLab, basado en Git, para la gestión del código fuente. Su uso resultó clave para:
 - Mantener la integridad del código de los distintos microservicios, permitir que los equipos de Frontend y Backend trabajaran en paralelo y contar con un historial de cambios que ayudara a revertir funciones en caso de errores durante las pruebas de integración con las bases de datos Oracle.

- Sincronización del equipo (Daily Meetings): Para mantener al equipo alineado y evitar problemas, nos reuníamos todos los días a la misma hora por la mañana y duraban unos 15 minutos como máximo.

En estas reuniones se revisaban tres cosas principales: Qué tareas habíamos completado el día anterior, qué íbamos a hacer durante la jornada, y si había algún bloqueo o problema técnico.

Figura 9
Herramienta de gestión de tareas



Nota. Elaboración propia

Planificación temporal del proyecto

La planificación del proyecto se hizo para organizar el trabajo dentro del plazo de tres meses, usando la metodología Scrum adaptada al contexto de la institución.

Para elaborar la planificación del proyecto a alto nivel, se consideraron tres insumos principales: la experiencia y conocimientos técnicos del equipo de desarrollo en el stack tecnológico seleccionado, la experiencia del analista funcional como desarrollador, y el conocimiento de las necesidades del área usuaria y del Product Owner.

Con base en esta información, se definió la duración de cada sprint y se priorizaron las funcionalidades principales, denominadas “core”, como las bandejas para cada actor y las operaciones básicas asociadas. Esta planificación a un nivel general permitió organizar de manera estructurada el desarrollo del proyecto, garantizando que los recursos y capacidades del equipo se alinearan con los objetivos y requerimientos del negocio.

La Tabla 12 presenta la planificación temporal del proyecto SSAIS a un nivel general, indicando la duración de cada sprint, los objetivos principales y las funcionalidades a desarrollar. Este enfoque refleja la aplicación de Scrum adaptado, considerando la experiencia del equipo, el rol del analista funcional y las necesidades del área usuaria.

Tabla 12
Planificación temporal del proyecto

Spring	Duración	Objetivo general
0	2 semanas	Definir requerimientos y arquitectura inicial, mediante el análisis del proceso actual y la propuesta de mejora.
1	3 semanas	Implementar las funcionalidades principales del Jefe USAPE y la bandeja inicial del Analista USAPE.
2	3 semanas	Completar las funcionalidades del Analista USAPE y desarrollar la bandeja inicial del Supervisor.
3	3 semanas	Finalizar las funcionalidades del Supervisor y consolidar los incrementos de los sprints anteriores.
4	1 semana	Realizar pruebas integrales, validar todas las funcionalidades y desplegar el sistema en producción.

Nota. Elaboración propia

Al finalizar el Sprint 0, se realizó la planificación del Sprint 1, contando con la participación del Product Owner, el Scrum Master, el analista funcional y los miembros del equipo de desarrollo. Durante esta reunión se seleccionaron las historias de usuario que se incluirían en el sprint, se estimó el esfuerzo requerido y se estableció el objetivo a alcanzar.

3.2.5.2. *Sprint 1: Desarrollo de funcionalidades para la gestión de casos del Jefe USAPE*

Durante este sprint se trabajaron las funcionalidades principales del módulo, para el rol correspondiente enfocándose en la gestión de los casos registrados en el sistema. Las historias de usuario seleccionadas permitieron realizar tareas clave como ver, buscar, asignar y dar seguimiento al recorrido de los casos, así como generar reportes de diagnósticos de alto costo.

Para este Sprint se seleccionaron para su desarrollo las historias de usuario de mayor relevancia para el área usuaria, las cuales se detallan a continuación:

- HU-01: Listar casos: Permite visualizar el listado general de casos registrados, ordenados cronológicamente.
- HU-02: Buscar casos: Permite aplicar filtros por estado, fecha y tipo de caso para facilitar la localización de registros específicos.
- HU-03: Asignar casos: Permite asignar casos a la bandeja JEFE USAPE según criterios definidos por el usuario.
- HU-04: Ver trazabilidad: Permite consultar el historial de acciones realizadas sobre cada caso.
- HU-07: Reporte dinámico de casos: Permite imprimir reportes de casos y exportarlos en formato Excel para impresión o distribución según el rol del usuario.

La Figura 10 muestra el prototipo de la bandeja para el rol Jefe USAPE, donde se presentan las funcionalidades desarrolladas que permiten visualizar, buscar, asignar y dar seguimiento a los casos registrados, así como generar reportes de diagnóstico de alto costo.

Figura 10
Bandeja del Jefe USAPE

MasSalud360

Login

Bandeja

Historial

Reportes

Dashboards

Filtros de Búsqueda:

Seleccionar Fecha

Seleccionar USAPE

Buscar

Seleccionar

Cancelar

Exportar

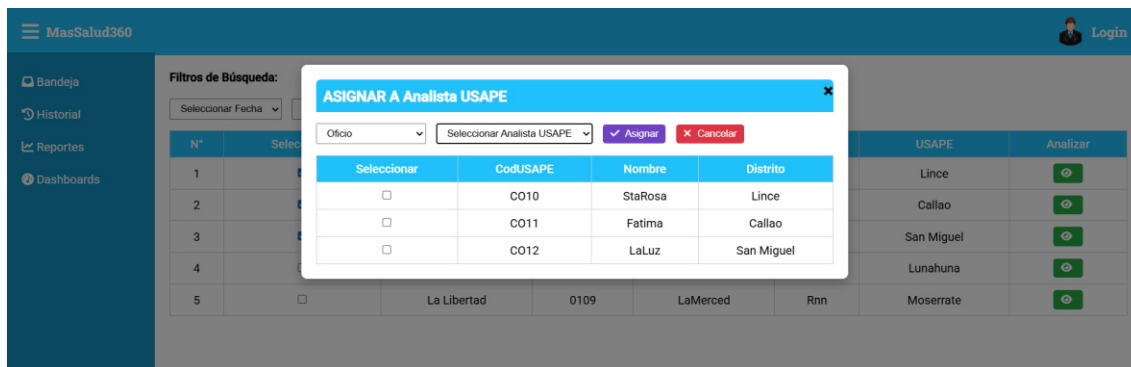
N°	Seleccionar	Región	Centro	Nombre	DxAc	USAPE	Analizar
1	☐	Lima	0102	StaRosa	Dal	Lince	
2	☐	Callao	0103	Fatima	Cnc	Callao	
3	☐	Lima	0105	LaLuz	Dbt	San Miguel	
4	☐	Ica	0107	SanMartin	Crz	Lunahuna	
5	☐	La Libertad	0109	LaMerced	Rnn	Moserrate	

Nota. Elaboración propia

La Figura 11 muestra el prototipo para la asignación de lotes de casos de potenciales de afiliaciones indebidas a los analistas.

Figura 11

Funcionalidad Asignar caso a Analista



Nota. Elaboración propia

Estas funcionalidades se desarrollaron siguiendo buenas prácticas de Scrum. El sprint comenzó con una sesión de planificación (Sprint Planning), donde se priorizaron las historias de usuario, se estimaron los puntos de historia y se definieron los entregables. Durante el desarrollo, se realizaron reuniones diarias (Daily Scrum) para coordinar el trabajo del equipo, identificar bloqueos y ajustar tareas según el avance.

Al finalizar, se llevó a cabo una revisión funcional (Sprint Review) con el Product Owner para confirmar que se cumplieran los criterios de aceptación. Finalmente, se hizo una retrospectiva (Sprint Retrospective) para identificar oportunidades de mejora en la dinámica de trabajo.

El Sprint terminó con la entrega de un módulo funcional que permite al usuario administrativo gestionar los casos de manera ágil desde la bandeja JEFE USAPE. Esta entrega se validó mediante pruebas funcionales y una revisión conjunta con el Product Owner, garantizando que se cumplieran los criterios de aceptación establecidos.

3.2.5.3. *Sprint 2: Desarrollo de funcionalidades para la gestión de casos del Analista USAPE.*

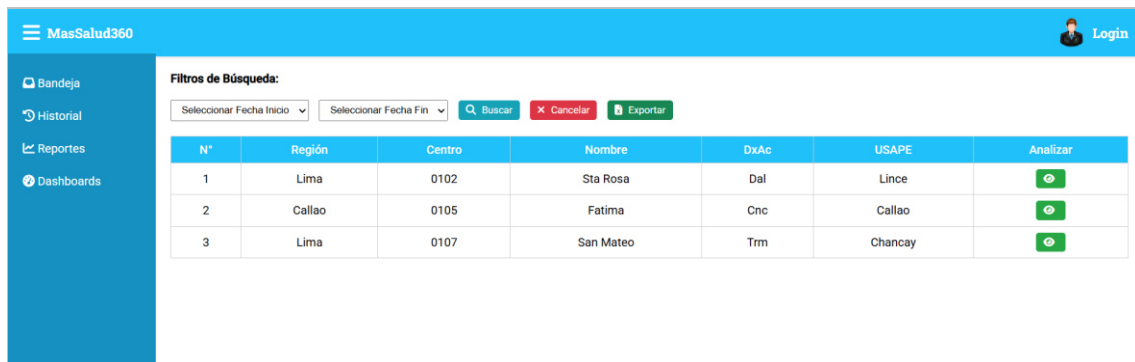
Durante este Sprint se trabajaron las funcionalidades para el rol Analista USAPE, centradas en manejar los casos que les asigna el rol Jefe USAPE. Esto permitió al usuario Analista USAPE ver los casos derivados, buscarlos con filtros, asignarlos al rol Supervisor y analizar cada expediente desde el punto de vista técnico.

Con la finalidad de dar continuidad al flujo de trabajo iniciado en el Sprint anterior, en este se centró en habilitar el entorno operativo del Analista USAPE. Para tal efecto se implementaron funcionalidades para la recepción de casos, búsqueda y el análisis centralizado de información, lo que permitió a este rol avanzar a hacia la siguiente etapa de asignación de Supervisor. Para ello se desarrollaron las siguientes Historias de usuario:

- HU-08: Listar casos asignados: Permitir al usuario Analista USAPE ver la lista completa de casos que le fueron asignados por el rol Jefe USAPE, organizados de manera cronológica.
- HU-09: Buscar casos: Facilita la localización de registros dentro de la bandeja mediante filtros por estado, fecha o tipo de caso.
- HU-10: Asignar casos: Permite derivar casos al rol Supervisor siguiendo criterios definidos por el usuario Analista USAPE, garantizando la trazabilidad del proceso.
- HU-11: Analizar integridad de afiliación: Permite visualizar de forma centralizada información consolidada de datos laborales y asistenciales, para sustentar la probabilidad de afiliación indebida.

La Figura 12 muestra el prototipo de la bandeja del Analista USAPE. En ella se presentan las funcionalidades desarrolladas para que pueda visualizar los casos asignados, incluyendo filtros de búsqueda, asignación al Supervisor y análisis de cada expediente

Figura 12
Bandeja del analista USAPE



Filtros de Búsqueda:

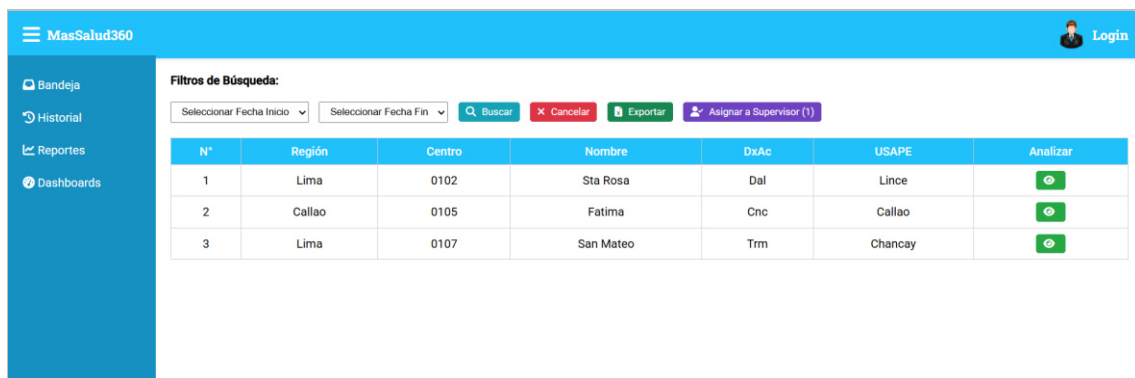
Seleccionar Fecha Inicio | Seleccionar Fecha Fin |

N°	Región	Centro	Nombre	DxAc	USAPE	Analizar
1	Lima	0102	Sta Rosa	Dal	Lince	Ojo
2	Callao	0105	Fatima	Cnc	Callao	Ojo
3	Lima	0107	San Mateo	Trm	Chancay	Ojo

Nota. Elaboración propia

La Figura 13 muestra el prototipo para la funcionalidad Analizar caso.

Figura 13
Funcionalidad analizar caso



Filtros de Búsqueda:

Seleccionar Fecha Inicio | Seleccionar Fecha Fin |

N°	Región	Centro	Nombre	DxAc	USAPE	Analizar
1	Lima	0102	Sta Rosa	Dal	Lince	Ojo
2	Callao	0105	Fatima	Cnc	Callao	Ojo
3	Lima	0107	San Mateo	Trm	Chancay	Ojo

Luego de pulsar sobre el botón con el icono del ojo, se muestra una ventana emergente:



ANALIZAR REGISTRO

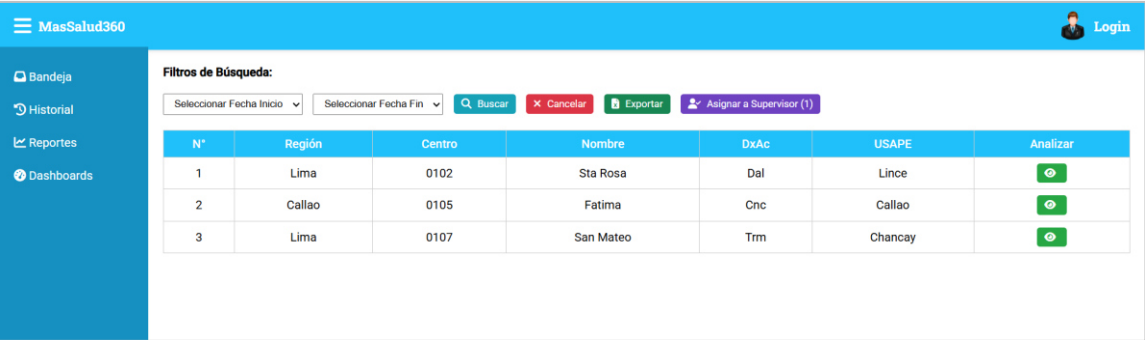
Empleado | Empleador | Atenciones

Datos del empleado

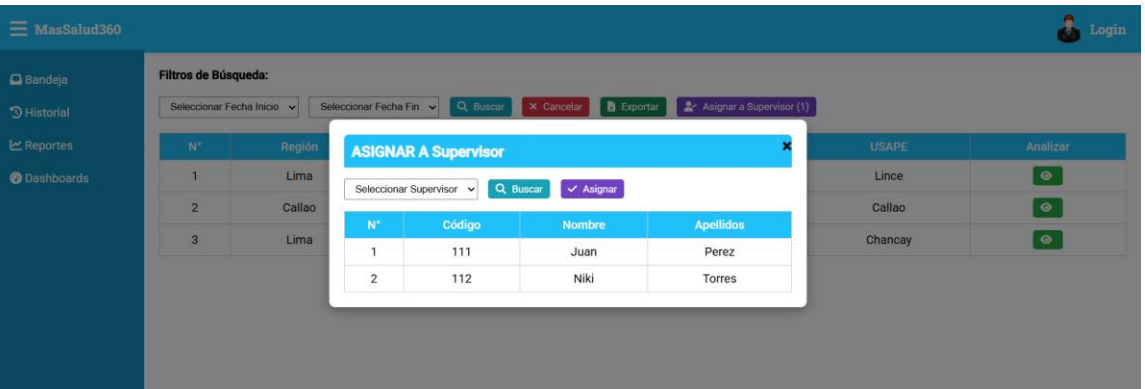
Nota. Elaboración propia

La Figura 14 muestra el prototipo para la funcionalidad Asignar a Supervisor.

Figura 14
Funcionalidad asignar caso a Supervisor



Luego de pulsar sobre el botón Asignar a Supervisor, se muestra una ventana emergente:



Nota. Elaboración propia

Estas funcionalidades se desarrollaron siguiendo buenas prácticas de Scrum. Todo comenzó con la planificación del sprint (Sprint Planning), donde se estimaron los puntos de historia y se definieron los entregables. Durante el desarrollo, el equipo se reunió diariamente (Daily Scrum) para coordinar avances y resolver cualquier bloqueo.

Al finalizar, se realizó una revisión funcional (Sprint Review) con el usuario clave para comprobar que se cumplieran los criterios de aceptación. Finalmente, se llevó a cabo una retrospectiva (Sprint Retrospective) para identificar oportunidades de mejora en el proceso de trabajo.

El sprint concluyó con un módulo funcional que permite al rol Analista USAPE gestionar los casos de manera ágil, fortaleciendo el flujo operativo entre los distintos perfiles del sistema y asegurando la trazabilidad de cada acción.

3.2.5.4. *Sprint 3: Desarrollo de funcionalidades para la gestión de casos del Supervisor.*

Durante el Sprint 3 se trabajaron las funcionalidades para el rol Supervisor, enfocándose en la validación técnica de los casos recibidos del rol Analista USAPE. Esto permitió que el Supervisor pudiera ver los casos asignados, revisar la información técnica, manejar los documentos según su estado y completar el proceso de supervisión con la carga del informe correspondiente.

En la ceremonia del Sprint Planning del Sprint 3, se definieron los requerimientos específicos para el rol Supervisor, los cuales no habían sido detallados en el Backlog inicial del Sprint 0 debido a que los procesos aún se encontraban en etapa de definición. En este contexto se agregaron las HU-12 a la HU-17 para cubrir el flujo de supervisión.

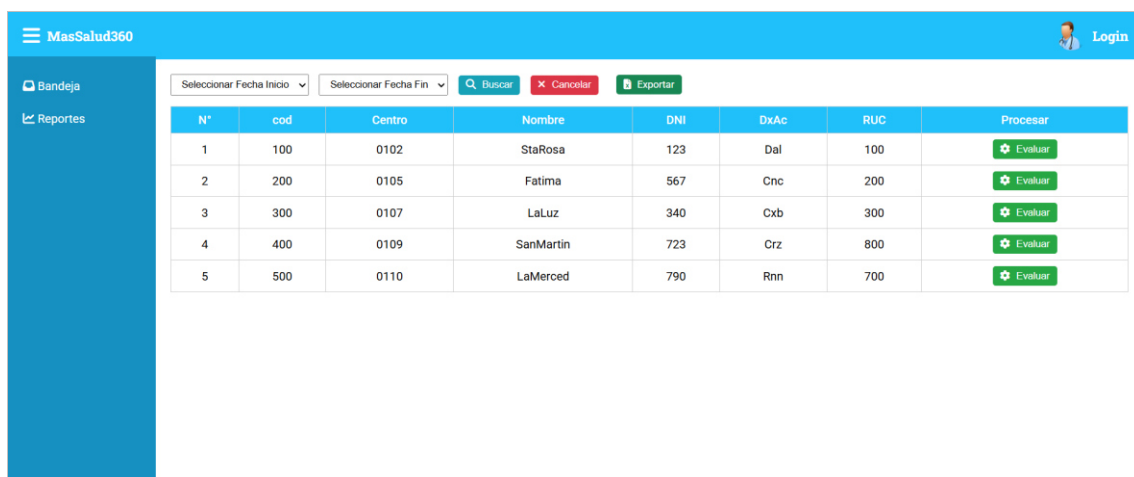
De esta manera a fin de completar el flujo operativo integral, este Sprint se enfocó en el desarrollo de funciones de evaluación técnica, gestión de documentos y formalización de cierre de casos; mediante la implementación de las siguientes historias de usuario:

- HU-12: Ver casos asignados: Permite al usuario Supervisor visualizar la lista general de casos que le fueron asignados por el rol Analista USAPE, ordenados por fecha y estado.

- HU-13: Buscar casos: Permite aplicar filtros por estado, tipo de caso y fecha de asignación para facilitar la localización de registros específicos.
- HU-14: Evaluar técnica de casos: Permite revisar el detalle del expediente, validar la información técnica y registrar observaciones.
- HU-15: Iniciar proceso de supervisión: Permite imprimir los formatos necesarios para la supervisión y cargar documentos relacionados al inicio del proceso.
- HU-16: Gestión de documentos: Permite cargar, adjuntar y visualizar documentos asociados a cada estado del caso durante el proceso de supervisión.
- HU-17: Finalizar proceso de supervisión: Permite registrar y cargar el informe técnico final para formalizar la culminación el cierre del caso y actualizar su estado.

La Figura 15 muestra el prototipo de la bandeja del Supervisor. En ella se presentan las funcionalidades desarrolladas para que pueda visualizar los casos, incluyendo filtros de búsqueda, evaluación, inicio de supervisión y manejo de documentos.

Figura 15
Bandeja del supervisor

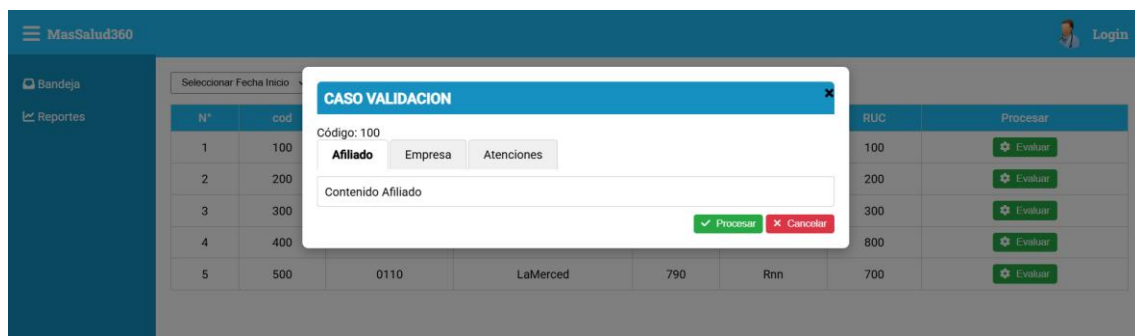


N°	cod	Centro	Nombre	DNI	DxAc	RUC	Procesar
1	100	0102	StaRosa	123	Dal	100	
2	200	0105	Fatima	567	Cnc	200	
3	300	0107	LaLuz	340	Cxb	300	
4	400	0109	SanMartin	723	Crz	800	
5	500	0110	LaMerced	790	Rnn	700	

Nota. Elaboración propia

La Figura 16 muestra el prototipo de la funcionalidad Evaluar, la cual presenta una ventana emergente que permite revisar la información y procesarla para dar inicio al trabajo en campo.

Figura 16
Funcionalidad evaluar caso



Nota. Elaboración propia

Estas funcionalidades se desarrollaron siguiendo las buenas prácticas de Scrum. El sprint comenzó con la planificación (Sprint Planning), donde se estimaron los puntos de historia y se definieron los entregables. Durante el desarrollo, se realizaron reuniones diarias (Daily Scrum) para revisar los avances y resolver bloqueos.

Al finalizar, se llevó a cabo una revisión funcional (Sprint Review) con el Product Owner, verificando que se cumplieran los criterios de aceptación.

Por último, se hizo una retrospectiva (Sprint Retrospective) para identificar oportunidades de mejora en el proceso de trabajo.

El sprint concluyó con un módulo completo que permite al rol Supervisor ejecutar el proceso técnico de supervisión de manera organizada, documentada y trazable, fortaleciendo el flujo operativo del sistema y asegurando la calidad de los casos procesados.

3.2.5.5. *Sprint 4: Pruebas finales y despliegue*

El Sprint 4 se planteó como una etapa para consolidar, validar y desplegar el sistema desarrollado. Aunque Scrum no contempla un sprint exclusivo para pruebas, en este caso se adaptó la metodología a las necesidades de la entidad, asegurando que el producto cumpliera con los estándares de calidad antes de su entrega formal.

Durante este sprint se realizaron las siguientes actividades:

- Ejecución de pruebas funcionales completas sobre los módulos desarrollados en los sprints anteriores.
- Validación cruzada de los flujos operativos entre los roles Jefe USAPE, Analista USAPE y Supervisor.
- Revisión de criterios de aceptación definidos por el Product Owner.
- Ajustes técnicos menores derivados de la retroalimentación obtenida.
- Preparación del entorno de producción y despliegue del sistema.

Estas actividades se llevaron a cabo siguiendo buenas prácticas de Scrum, con colaboración continua entre los miembros del equipo. Aunque en este sprint no se asignaron Story Points por tratarse de tareas técnicas no funcionales, el trabajo realizado fue fundamental para asegurar que el sistema fuera estable, trazable y completamente operativo.

El sprint terminó con la puesta en marcha del sistema en su entorno productivo, cumpliendo los objetivos funcionales establecidos y dejando la solución lista para su uso institucional.

3.3. Evaluación

3.3.1. Evaluación Económica/Evaluación Costo Beneficio

Inversión del proyecto

Para evaluar la viabilidad económica del proyecto, se calculó el costo del equipo técnico durante los tres meses de ejecución, considerando los honorarios mensuales de cada rol, infraestructura, equipamiento y el tiempo total estimado del proyecto.

Tabla 13

Costos operativos del equipo de desarrollo

Rol	Cantidad	Costo Mensual (s/.)	Meses	Costo Total(s/.)
Analista Programador	1	9000	3	27000
Analista Funcional	1	7000	3	21000
Desarrollador Backend	1	5000	3	15000
Desarrollador Frontend	1	5000	3	15000
Analista de Base de Datos	1	7000	3	21000
Oficina / espacio	1	1500	3	4500
Energía / Internet	1	300	3	900
Equipamiento (PC, software)	5	2500	3	7500
Total				111900

Nota. Elaboración propia

El costo total estimado para desarrollar el sistema es de S/ 111,900, este monto representa la inversión directa para implementar la solución tecnológica que optimiza el proceso de validación de afiliaciones irregulares.

Beneficios esperados

Entre los beneficios esperados se incluyen la reducción de pagos indebidos por afiliaciones no justificadas, disponibilidad inmediata de la información que es crítica para la supervisión, una mejor trazabilidad de los casos y un fortalecimiento del control institucional.

Dado que no se cuenta con información histórica oficial, los beneficios se estiman mediante proyecciones basadas en supuestos razonables, tal como se describe a continuación.

Justificación y origen de los valores de ahorro por caso

Dado que SSAIS es un sistema interno, y no se dispone de información oficial de costos de tratamientos o atenciones o atenciones indebidas, el valor de 5000 se considera una estimación razonable en razón de:

- Literatura especializada, datos de referencia o informes públicos sobre costos de atenciones de alto costo.
- Referencias de costos promedio de tratamientos de alto costo.

Este valor permite realizar un ejercicio de evaluación económica para demostrar la viabilidad financiera del proyecto mediante simulaciones de VAN y TIR.

3.3.2. Interpretación del Van y del Tir

Para analizar la rentabilidad del proyecto, se realizó una simulación financiera basada en el ahorro potencial que la entidad podría lograr al detectar afiliaciones irregulares asociadas a atenciones de alto costo. Estas incluyen enfermedades como cardiopatías, cáncer, diálisis, trasplantes, entre otros, cuyos tratamientos generan gastos significativos en medicamentos, cirugías y hospitalización.

Tabla 14

Parámetros económicos para la evaluación financiera del proyecto

Rol	Detalle	Costo Total(s/.)
Costo total del proyecto	s/. 111,900	Ver sección 3.3.1.
Tiempo de recuperación estimado	1 año	
Ahorro estimado por detección de casos irregulares	s/.600000 anual	10 casos por mes x 5000x12meses
Tasa de descuento usada	10% anual	Valor referencial
<i>Nota.</i> Elaboración propia		

Cálculo del VAN

$$VAN = \sum_{t=1}^n \frac{Flujo_t}{(1+r)^t} - inversion\ Inicial$$

$$VAN \frac{600000}{(1+0.10)} - 111900 = 433,554.55$$

Cálculo del TIR

$$0 = \sum_{t=1}^n \frac{Flujo_t}{(1 + TIR)^t} - inversion\ Inicial$$

$$0 = \sum_{t=1}^n \frac{600000}{(1 + TIR)^1} - 111900$$

Resolviendo:

$$1 + r = \frac{600000}{111900} = 8.333 \Rightarrow r \approx 4.362$$

Interpretación:

- El VAN positivo de S/ 433,554.55 muestra que el proyecto aporta un valor económico significativo a la aseguradora.
- La TIR del 436.2% indica que la inversión se recupera rápidamente y ofrece una rentabilidad elevada, gracias al efecto directo en la reducción de gastos por atenciones no justificadas.

CAPÍTULO IV – REFLEXIÓN CRÍTICA DE LA EXPERIENCIA

La experiencia profesional desarrollada en el proyecto SSAIS fue un proceso continuo de adaptación y de aprendizaje, principalmente porque hubo un cambio de contexto con respecto a mis experiencias laborales anteriores. Es decir, antes de ingresar a esta organización, mi trayectoria se había desarrollado en el sector privado donde trabajaba de forma remota como desarrollador de software con perfil técnico versátil. Esta experiencia en cambio se llevó a cabo en una entidad del sector público, con una cultura organizacional diferente y bajo una modalidad de trabajo híbrida, lo que implicó nuevos retos profesionales.

Si bien mi contrato inicial estaba orientado al desarrollo de software, al momento de incorporarme se me solicitó asumir el rol de analista funcional y que además estaría a cargo del equipo de desarrollo. Este cambio implicó mayor responsabilidad dado que tendría que coordinar tareas, actuar como intermediario entre el área usuaria y el equipo de desarrollo.

Si bien no era un rol completamente nuevo para mí, sí exigió una mayor capacidad de adaptación, ya que implicaba comprender procesos institucionales complejos y traducirlos en requerimientos claros y viables. La experiencia previa desempeñando distintos roles técnicos facilitó esta transición y me permitió abordar los desafíos de forma más efectiva.

Uno de los aspectos más relevantes de esta experiencia fue el fortalecimiento de mis habilidades blandas. La interacción directa y frecuente con el área usuaria y el Product

Owner, conformados por profesionales de diversas disciplinas, hizo necesario ajustar el lenguaje técnico del autor y preparar cuidadosamente cada reunión, anticipando preguntas y buscando consensos. A diferencia del trabajo remoto, la presencialidad obligó a una comunicación más directa y empática, lo que contribuyó a salir de la zona de confort y a mejorar la capacidad de escucha y negociación.

Durante el desarrollo del proyecto, surgieron cambios imprevistos, especialmente cerca del cierre de los sprints. Estas situaciones generaban cierta tensión en el equipo de desarrollo, por lo que fue necesario equilibrar las expectativas del área usuaria con lo que el equipo podía entregar realmente. Así que, como primera medida se negoció la priorización de los requerimientos, donde se comprometía a entregar los más críticos, mientras que los otros se aplazaban al siguiente sprint o en todo caso se acordaba una concesión operativa para el equipo.

Aunque el proyecto ya había sido iniciado con ciertos retrasos, esta experiencia me permitió aprender a optimizar la asignación de tareas, gestionar prioridades y mantener un flujo de trabajo eficiente, garantizando el cumplimiento de los objetivos dentro de los plazos previstos.

Esta experiencia también me permitió comprender la diferencia entre trabajar en el sector privado y el público, donde las culturas organizacionales son marcadamente distintas.

En este contexto, un consejo que recibí de alguien del propio entorno —“moverse según venga la ola”— resultó especialmente valioso. Me enseñó que la flexibilidad, la adaptabilidad y la capacidad de gestionar situaciones imprevistas son habilidades esenciales para integrarse y desenvolverse efectivamente en entornos públicos, especialmente para profesionales que provienen del sector privado.

CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- El sistema SSAIS, basado en una arquitectura de microservicios, centraliza la información de afiliados con diagnósticos de alto costo, permitiendo que los distintos roles institucionales Supervisor, Analista de USAPE y Jefe de USAPE accedan de manera eficiente a los datos relevantes para el proceso de supervisión.
- La bandeja de casos del Supervisor ayuda a analizar los casos asignados y hace que la supervisión sea más ágil. Con las herramientas del módulo, se reducen los tiempos de revisión y se facilita el seguimiento de cada caso, apoyando un proceso de supervisión más ordenado y eficiente.
- Los módulos del Analista de USAPE permiten evaluar los casos asignados y acceder a información detallada de los afiliados. También se pueden asignar casos a supervisores de manera organizada, asegurando que se realicen los análisis necesarios para identificar posibles afiliaciones indebidas.
- Los módulos del Jefe USAPE permiten supervisar todo el proceso. Se puede consultar el historial de casos, revisar el estado de cada supervisión y tomar decisiones gerenciales basadas en información confiable y trazable.

- En general, el sistema apoya a la entidad a realizar supervisiones más eficientes y confiables. Cada caso cuenta con seguimiento y trazabilidad completa, lo que facilita la toma de decisiones basadas en información precisa y oportuna.

5.2. Recomendaciones

Se recomienda implementar mejoras funcionales que refuercen el soporte al proceso de supervisión de afiliaciones indebidas, en particular en los siguientes aspectos:

- Se sugiere agregar métricas como el costo de medicinas, el número de atenciones y el gasto por afiliado. Estos datos permitirán generar alertas por altos costos, aplicar filtros por rangos y calcular un scoring de riesgo, facilitando la priorización de los casos con mayor impacto financiero o probabilidad de afiliación indebida.
- A partir de la información histórica y de los criterios del área usuaria, se sugiere definir patrones que faciliten la detección temprana de posibles casos.
- Crear dashboards ayudará a mostrar de manera clara el estado de los casos, los costos involucrados y el progreso del proceso de supervisión, facilitando la toma de decisiones y el seguimiento de los resultados.
- Implementar alertas automáticas que notifiquen sobre plazos y cambios de estado.
- Se recomienda evaluar el uso de IA para mejorar el proceso de supervisión a través de la detección automática de patrones sospechosos.

5.3. Fuentes de Información

Cohn, M. (2005). Agile estimating and planning. Upper Saddle River, NJ: Prentice Hall.

Consortio de Investigación Económica y Social. (2014). Glosario de términos sobre seguridad social en salud. Lima, Perú: CIES.

Contraloría General de la República. (2025). Informe de Auditoría N.º 033-2025-2-0251-AC: Pagos a servidores y pensionistas fallecidos durante los periodos 2020–2023. Lima, Perú: Autor. Recuperado de https://apps8.contraloria.gob.pe/SPIC/srvDownload/ViewPDF?CRES_CODIGO=2025CPO025100032&TIPOARCHIVO=ADJUNTO

EsSalud. (2014, 18 de agosto). EsSalud implementa Registro de Asegurados Fallecidos. Recuperado de <https://www.essalud.gob.pe/essalud-implementa-registro-de-asegurados-fallecidos/>

EsSalud. (2015). EsSalud: Historia y funciones. Lima, Perú. Recuperado de <https://www.essalud.gob.pe>

EsSalud. (2016, 13 de febrero). EsSalud detectó más de 5 mil afiliaciones indebidas en el 2015. Recuperado de <https://www.essalud.gob.pe/essalud-detecto-mas-de-5-mil-afiliaciones-indebidas-en-el-2015/>

EsSalud. (2025, 16 de octubre). Resolución de Gerencia Central N.º 30-GCSPE-ESSALUD-2025. Gerencia Central de Seguros y Prestaciones Económicas. Recuperado de <https://www.gob.pe/institucion/essalud/normas-legales/7400202-30-gcspe-essalud-2025>

Fielding, R. T. (2000). Architectural styles and the design of network-based software architectures (Tesis doctoral, University of California, Irvine). Recuperado de <https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>

Gonzales Hunt, C., & Paitán Martínez, J. (2017). El derecho a la seguridad social (1.^a ed.). Lima, Perú: Fondo Editorial PUCP. Recuperado de <https://repositorio.pucp.edu.pe/handle/20.500.12404/16611>

Harrington, H. J. (2016). Business process improvement: The breakthrough strategy for total quality, productivity, and competitiveness (3.^a ed.). New York, NY: McGraw-Hill.

Healthcare.gov. (s.f.). Health coverage. Recuperado el 23 de diciembre de 2025, de <https://www.healthcare.gov/glossary/health-coverage>

Martin, R. C. (2003). Agile software development: Principles, patterns, and practices. Upper Saddle River, NJ: Prentice Hall.

Microsoft. (s.f.). What is DevOps? Microsoft Learn. Recuperado de <https://learn.microsoft.com/es-es/devops/what-is-devops>

Microsoft. (s.f.-a). Microservices architecture style. Microsoft Learn. Recuperado de <https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/microservices>

Microsoft. (s.f.-b). Microservices architecture. Microsoft Learn. Recuperado de <https://learn.microsoft.com/es-es/dotnet/architecture/microservices/architect-microservice-container-applications/microservices-architecture>

Ministerio de Inclusión, Seguridad Social y Migraciones. (2013, 23 de julio). La Policía Nacional destapa un complejo entramado empresarial creado para defraudar a la Seguridad Social. Recuperado de <https://www.inclusion.gob.es/w/la-policia-nacional-destapa-un-complejo-entramado-empresarial-creado-para-defraudar-a-la-seguridad-social>

Ministerio de Justicia y Derechos Humanos. (2021). Texto Único Ordenado de la Ley N.º 27444 – Ley del Procedimiento Administrativo General. Perú. Recuperado de <https://www.gob.pe/institucion/minjus/informes-publicaciones/2040776-texto-unico->

ordenado-de-la-ley-n-27444-ley-del-procedimiento-administrativo-general-actualizado-al-13-de-julio-de-2021

Montevideo Portal. (2017, 9 de marzo). BPS tiene constancia de 500 afiliaciones fraudulentas al Fonasa. Recuperado de <https://www.montevideo.com.uy/Noticias/BPS-tiene-constancia-de-500-afiliaciones-fraudulentas-al-Fonasa-uc336961>

National Institute of Standards and Technology (NIST). (1993). Integration definition for function modeling (IDEF0). Recuperado de <https://www.idef.com>

OWASP Foundation. (2021). OWASP Top 10 – 2021: The ten most critical web application security risks. Recuperado de <https://owasp.org/Top10/>

Perú, Ministerio de Trabajo y Promoción del Empleo. (2009, 15 de enero). Decreto Supremo N.º 002-2009-TR que aprueba el Reglamento de la Ley N.º 29135. Diario Oficial El Peruano.

RENIEC (Registro Nacional de Identificación y Estado Civil). (s.f.). Recuperado de <https://www.reniec.gob.pe>

Schwaber, K., & Sutherland, J. (2020). La guía de Scrum: Guía definitiva de Scrum: Las reglas del juego (Versión noviembre 2020). Recuperado de <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish.pdf>

Seguro Social de Salud. (2024). Informe de Evaluación del Plan Estratégico Institucional (PEI) 2020–2024: Primer semestre 2024. Lima, Perú: Oficina Central de Planeamiento y Presupuesto. Recuperado de https://www.essalud.gob.pe/transparencia/inf_monitoreo_eval_planes_politicas/Inf_de_Eval_PEI_2020-2024_1er_semestre2024.pdf

SUNAT (Superintendencia Nacional de Administración Tributaria). (s.f.). Recuperado de <https://www.sunat.gob.pe>

TNH. (2025, 10 de febrero). Ejecutan capturas relacionadas a investigación sobre planillas fantasmas en el IHSS. Recuperado de <https://tnh.gob.hn/nacional/ejecutan-capturas-relacionadas-a-investigacion-sobre-planillas-fantasmas-en-el-ihss/>

World Wide Web Consortium (W3C). (2018). Web Content Accessibility Guidelines (WCAG) 2.1. Recuperado de <https://www.w3.org/TR/WCAG21>

World Wide Web Consortium (W3C). (2023). What's new in WCAG 2.2. Web Content Accessibility Guidelines. Recuperado de <https://www.w3.org/WAI/standards-guidelines/wcag/new-in-22/#guideline-25-input-modalities>

5.4. Glosario

- **API REST (Representational State Transfer)**

Es un estilo de arquitectura para crear servicios web que permite que distintos sistemas se comuniquen usando solicitudes HTTP. Con él, se pueden realizar operaciones como crear, leer, actualizar o eliminar datos (CRUD) de manera estandarizada, usando URLs y métodos como GET, POST, PUT y DELETE.

- **AS-IS**

Describe la situación actual de un proceso o sistema, tal como se encuentra hoy.

- **CORS (Cross-Origin Resource Sharing)**

Es un mecanismo de seguridad de los navegadores que controla si un recurso web, como una API, puede ser accedido desde otro dominio diferente al del servidor. Esto permite que distintos sitios se comuniquen de forma segura, evitando accesos no autorizados.

- **DevOps**

Se refiere a una cultura y a un conjunto de prácticas que buscan integrar los equipos de desarrollo de software con los de operaciones de tecnologías de la

información (TI). El objetivo es coordinar mejor el planeamiento, el desarrollo, la entrega y la operación de aplicaciones, fomentando la colaboración, la automatización y la mejora continua de la calidad. De esta manera, se mejora la comunicación y cooperación entre las áreas, evitando bloqueos y falta de coordinación, lo que permite entregar valor a los usuarios de manera más rápida y confiable (Microsoft, s.f.).

- **EsSalud (Seguro Social de Salud del Perú)**

Es la institución pública encargada de la administración del seguro social de salud en Perú, brindando cobertura médica a los afiliados y supervisando la correcta prestación de los servicios de salud.

- **IDEF0 (Integration Definition for Function Modeling)**

Es una técnica que ayuda a representar gráficamente cómo funcionan los procesos de un sistema. Permite ver claramente las entradas, salidas, controles y recursos de cada función, lo que facilita analizar y mejorar los procesos.

- **RENIEC (Registro Nacional de Identificación y Estado Civil)**

Es la entidad peruana responsable de la identificación de las personas y del registro de actos civiles como nacimientos, matrimonios y defunciones.

- **SUNAT (Superintendencia Nacional de Administración Tributaria)**

Es el organismo del Estado peruano encargado de la recaudación de impuestos y supervisión del cumplimiento tributario.

- **TO-BE**

Describe la situación futura deseada o propuesta para un proceso o sistema.

ANEXOS

ANEXO A

Product backlog del proyecto

La tabla 15 presenta el Product Backlog general del proyecto, mostrando las Épicas, Sprints e Historias de Usuario (HU), junto con su descripción y la estimación en Story Points. Permite ver de forma clara la planificación y el esfuerzo que requiere cada tarea.

Tabla 15

Product Backlog Final

Épica	Sprint	Historia de usuario HU	Descripción	Story Points
	Sprint 0		Reuniones con stakeholders, levantamiento de información, definición del backlog.	0
Gestion de casos por el rol Jefe USAPE	Sprint 1	HU-01 Listar casos	Visualizar lista general de casos.	5
		HU-02 Buscar casos	Filtrar caso por diferentes criterios de búsqueda.	3
		HU-03 Asignar casos	Asignar casos a Analista USAPE.	3
		HU-04 Ver Trazabilidad	Ver trazabilidad de casos.	5
		HU-07 Reporte dinámico de casos	Visualización, exportación e impresión de registros con columnas adaptadas según rol.	3
Gestion de casos por el rol Analista USAPE	Sprint 2	HU-08 Listar casos asignados	Visualizar lista general de casos asignados por el Jefe USAPE.	3
		HU-09 Buscar casos	Filtrar caso por diferentes criterios de búsqueda.	3

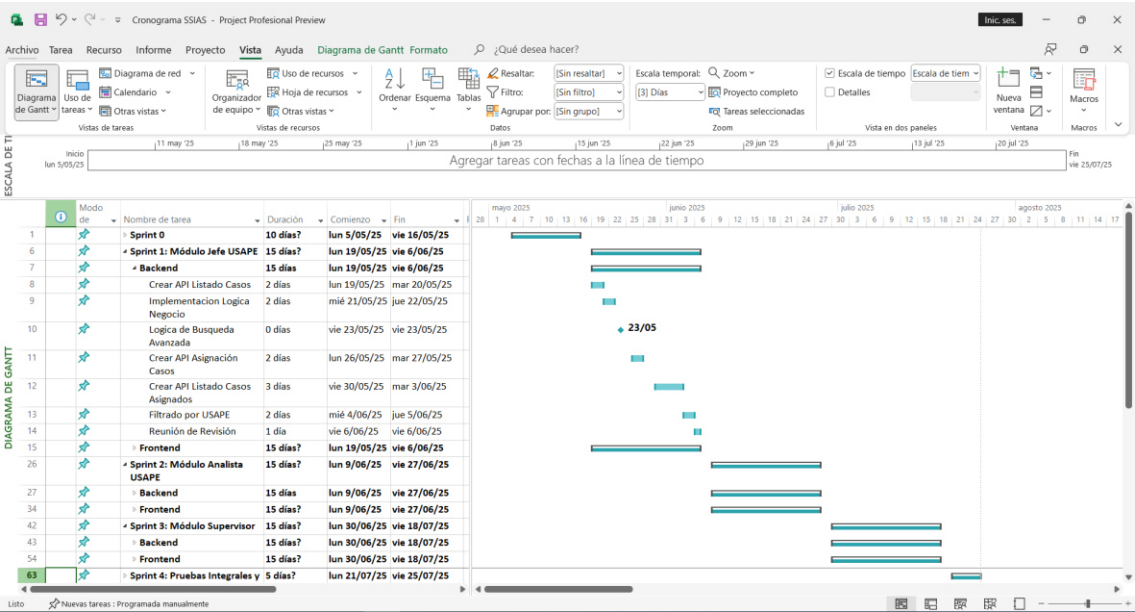
Gestion de casos por el rol Supervisor	Sprint 3	HU-10 Asignar casos	Asignar caso a Supervisor.	3
		HU-11 Analizar integridad de afiliación caso	Visualizar de forma centralizada información del empleado, empleador y datos asistenciales.	13
		HU-12 Ver casos asignados	Visualizar lista general de casos asignados por Analista USAPE.	3
		HU-13 Buscar caso	Filtrar caso por diferentes criterios de búsqueda.	3
		HU-14 Evaluación técnica de casos	Consultar la información consolidada de datos laborales y asistenciales.	2
		HU-15 Iniciar proceso Supervisión	Imprimir formatos necesarios y cargar los documentos relacionados.	8
		HU-16 Gestion de documentos	Cargar, adjuntar y visualizar documentos de cada estado del caso.	3
	Sprint 4	HU-17 Finalizar proceso Validación	Registrar y cargar informe final.	3
			Pruebas funcionales, ajustes finales, validación integral y despliegue en producción.	0

Nota. Elaboración propia

ANEXO B

Cronograma del proyecto SSAIS

La imagen siguiente muestra, de manera general, la planificación temporal inicial del proyecto SSAIS. El cronograma se organiza por Sprints, indicando la duración de cada uno y las principales tareas correspondientes al desarrollo Back-end y Front-end.



ANEXO C

Historias de usuario clave

La Tabla 8 muestra las historias de usuario clave, indicando el rol, el identificador de la HU, la funcionalidad y su descripción, ofreciendo una visión clara de los principales requerimientos del sistema.

Tabla 16

Historias de usuario clave

Rol	HU	Funcionalidad	Descripción
Jefe USAPE	HU-01	Listar casos	Permite acceder al listado general de casos registrados, ordenados de forma cronológica, con opciones de filtrado y exportación para facilitar su análisis y control.
	HU-03	Asignar casos	Permite asignar los casos de supervisión a los analistas, de acuerdo con criterios previamente definidos, asegurando una gestión clara, trazable y eficiente.
Analista USAPE	HU-08	Listar casos asignados	Permite visualizar de manera cronológica los casos asignados en su bandeja, con opciones de filtros de búsqueda y exportación de la información para facilitar su seguimiento y análisis.
	HU-10	Asignar casos	Permite asignar casos de Supervisión a los Supervisores, asegurando una adecuada distribución, trazabilidad y control del proceso de supervisión.
Supervisor	HU-12	Ver casos asignados	Permite al Supervisor gestionar los casos asignados, seleccionar aquellos

		que serán evaluados, con opciones de filtros de búsqueda y exportación de la información para facilitar su seguimiento y análisis.
HU-14	Evaluación técnica de casos	Permite analizar el detalle de un caso asignado mediante una vista con pestañas informativas que incluyen datos del asegurado, la empresa empleadora y datos asistenciales, para evaluar la información, y decidir su registro en estado, sin que ello implique el inicio del proceso de verificación.
HU-15	Iniciar proceso de supervisión	Permite iniciar formalmente el proceso de supervisión de un caso previamente evaluado, habilitando las herramientas necesarias para gestionar la supervisión.

Nota. Elaboración propia