



UNIVERSIDAD TÉCNICA
FEDERICO SANTA MARÍA

DEPARTAMENTO DE ELECTROTECNIA E INFORMÁTICA
INGENIERÍA EN INFORMÁTICA

Arquitectura de Software para CoreSafe: Plataforma Web y Móvil para la Asignación, Captura y Análisis del Programa de Registro Conductual (PRC) en INDEMIN

Felipe Gutiérrez Benítez

Felipe.gutierrezb@sansano.usm.cl

Dagoberto Cabrera Tapia
Profesor Guía

Diego Cáceres Solís
Profesor Correferente

Resumen: El presente trabajo de título se desarrolla en el contexto de las operaciones industriales y mineras de INDEMIN, donde el Programa de Registro Conductual (PRC) se gestiona de forma manual mediante registros en papel, generando pérdida de trazabilidad, duplicidad de esfuerzos y limitaciones en el análisis de datos. Ante esta situación, el objetivo de la tesina es diseñar e implementar la arquitectura de CoreSafe, una plataforma web y móvil orientada a la digitalización, asignación y análisis del PRC, con el objetivo de optimizar la captura de información conductual y fortalecer la gestión preventiva. La solución se basa en una arquitectura monolito modular con integración mediante API Gateway, y un enfoque offline-first que garantiza la disponibilidad del sistema en entornos de baja conectividad. El desarrollo de la aplicación involucró la participación de seis estudiantes, en distintas tareas de implementación, Sin embargo, la presente tesina se centra exclusivamente en el diseño arquitectónico del sistema, el cual fue concebido y desarrollado bajo un enfoque iterativo e incremental, aplicando principios de ingeniería de software orientada a la arquitectura. Los resultados esperados consideran la mejora en la eficiencia operativa, la trazabilidad de los registros conductuales y la generación de indicadores analíticos que faciliten la toma de decisiones y promuevan la cultura de seguridad en INDEMIN.

Palabras Clave: Arquitectura de software, digitalización, trazabilidad, seguridad laboral, CoreSafe.



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): Memoria o trabajo de título Tesis de Postgrado
Arquitectura de software para CORESAFE: Plataforma Web y Móvil para la asignación, captura y análisis del programa de

Título del trabajo: registro conductual (PCR) en INDEMIN

Nombre del candidato(a): Felipe Andrés Gutiérrez Benítez

Carrera / Grado: Ingeniería en Informática

Campus: Sede Viña del Mar **Departamento:** ELINF

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Dagoberto Cabrera Tapia, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por (**marcar una opción**):

6 meses 12 meses 2 años 3 años 5 años 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 22/01/2026

Firma:

Estudiante o Candidato(a):

Fecha: 22/01/2026

Firma:

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.



1 Introducción

1.1 Contexto y antecedentes

En el ámbito de las operaciones industriales y mineras, la seguridad laboral constituye un eje fundamental para garantizar la continuidad productiva y la protección de las personas. Debido a la complejidad de los procesos, la interacción entre equipos pesados, sustancias peligrosas y entornos de trabajo desafiantes, la prevención de accidentes se convierte en una prioridad estratégica tanto para las empresas como para los organismos reguladores.

A raíz de esta situación, diversos autores han identificado tres ámbitos con respecto a la seguridad en las organizaciones: *el ámbito teórico, orientado a explicar y pronosticar las conductas seguras e inseguras; el ámbito del diagnóstico, cuyo propósito es diseñar procedimientos que permitan una evaluación cualitativa y cuantitativa de la cultura de seguridad en la empresa; y el ámbito de intervención, que justifica a los dos anteriores y busca reducir la accidentabilidad y mejorar el bienestar de los trabajadores ([1], citando a Meliá, Ricarte y Arnedo, 1999, Campos, D. (2012).).*

Dentro de este marco, la empresa INDEMIN, dedicada a proveer soluciones integrales en operaciones industriales y mineras, siendo líder en el mercado de logística interna y movimiento de carga. Ha identificado que las observaciones de conducta en terreno han demostrado ser una herramienta clave para identificar comportamientos inseguros, promover prácticas seguras y anticipar riesgos antes de que se materialicen en incidentes o en accidentes. Sin embargo, en la mayoría de los casos, estos registros se realizan de manera manual, fragmentada y sin una adecuada sistematización, lo que genera dificultades en el seguimiento, análisis y retroalimentación oportuna hacia los trabajadores y las áreas responsables de la gestión de seguridad.

El desafío identificado surge entonces de la necesidad de digitalizar y centralizar los registros de observación de conducta en entornos industriales y mineros, con el fin de contar con información precisa, accesible y en tiempo real. La ausencia de herramientas tecnológicas integradas no solo limita la capacidad de detectar patrones de riesgo, sino que también dificulta la implementación de planes de acción efectivos y la mejora continua de la cultura preventiva dentro de las organizaciones. En este sentido *Bachche, Rydström y Bjelkemyr (2023) sostienen que la digitalización en los sistemas de seguridad industrial posibilita el monitoreo continuo y la detección temprana de riesgos, mientras que su ausencia restringe la comunicación y la capacidad preventiva de las organizaciones [2].*

1.2 Definición del problema

Actualmente, las observaciones conductuales se han consolidado como un método ampliamente utilizado en la gestión preventiva, dado que posibilitan la identificación temprana de desviaciones respecto de los procedimientos establecidos. Sin embargo, su aplicación en terreno presenta limitaciones significativas. En la mayoría de las organizaciones, incluyendo a INDEMIN, estos registros se realizan de forma manual, en documentos físicos y en formatos diferentes, lo que conlleva a:

1. **Registros manuales y dispersos:** Los registros y trata de data se realiza de manera manual y de manera rudimentaria con lápiz y papel. Y luego es traspasado a un Excel



2. **Fragmentación de la información:** Al existir múltiples formatos sin un estándar único de registro
3. **Dificultades en el análisis:** Los datos recabados no siempre se sistematizan, ni estandarizan. Lo cual no permite la detección de tendencias o patrones de riesgo
4. **Escasa retroalimentación oportuna:** Esto limita la capacidad de las áreas responsables para implementar medidas correctivas inmediatas
5. **Baja trazabilidad:** Se dificulta dar seguimiento al cumplimiento de los planes de acción derivados de las observaciones.

Estas limitaciones impactan directamente en la capacidad de INDEMIN para entregar a sus clientes información oportuna y de valor estratégico, debilitando la eficiencia de sus servicios de prevención y reduciendo el potencial de mejora continua en la seguridad laboral.

1.3 Descripción general de la propuesta de solución

La propuesta de solución se materializa en el diseño e implementación de CoreSafe, una plataforma web y móvil orientada a la asignación, captura y análisis del Programa de Registro Conductual (PRC) en faenas industriales y mineras.

El sistema busca digitalizar y centralizar los registros de observaciones conductuales, superando las limitaciones del método manual actualmente utilizado en INDEMIN. Para ello, se plantea una arquitectura tecnológica que integre los siguientes componentes:

1. Aplicación móvil: Permite a los supervisores y trabajadores en terreno registrar observaciones conductuales de manera rápida y offline, garantizando la continuidad operativa. Una vez disponible la conexión, los registros se sincronizan automáticamente con la plataforma central.
2. Plataforma Web: Orientada a la línea de mando, administradores y analistas, facilita la creación, asignación y gestión de PRC, además de proveer herramientas de visualización y generación de reportes que permitan el análisis comparativo de tendencias y el seguimiento de planes de acción
3. API corporativo de integración: CoreSafe se conectará con la infraestructura tecnológica de INDEMIN mediante una API centralizada, la cual permitirá sincronizar información relevante, como catálogo de maquinaria, ubicaciones de faenas y cuentas de usuarios. Esta integración se realizará de manera periódica y automatizada.
4. Módulo de análisis y reportabilidad: Incluye la generación de métricas e indicadores clave que permiten evaluar la evolución de las conductas observadas en el tiempo, identificando áreas de mejora y verificando la efectividad de los planes de intervención implementados.

1.4 Breve descripción de cómo aporta la arquitectura a la solución

La presente Tesina tiene como objetivo principal el diseño e implementación de la arquitectura de software que soporta la plataforma CoreSafe. Si bien la solución general planteada busca digitalizar y centralizar los registros del Programa de Registro Conductual (PRC), el aporte específico de este trabajo consiste en proveer la estructura tecnológica y metodológica que hace viable su desarrollo, operación y sostenibilidad en el tiempo.



Este aporte se manifiesta en los siguientes ejes

- Definición de la arquitectura de referencia en el cual se establecerá una estructura modular que separe responsabilidades entre los distintos servicios (Autenticación, gestión de formularios, asignación de observaciones, análisis y reportes), lo que incrementaría la posibilidad de incrementar la mantenibilidad y escalabilidad del sistema
- Integración con los sistemas corporativos de Indemin diseñando un esquema de integración mediante API, permitiendo la sincronización de forma periódica la información sobre la maquinaria, faenas y usuarios, garantizando la consistencia de los datos.
- Soporte a la operación de terreno, dentro de la propuesta se considera incorporar un enfoque offline-first para la aplicación móvil, lo cual asegura la captura de observaciones aún en condiciones de conectividad limitada, con procesos de sincronización confiables una vez restablecida la conexión.
- Gestión centralizada de usuarios y seguridad implementada con la integración de la API de Indemin, la cual agregará un modelo de autenticación federada y control de acceso basado en roles (RBAC), protegiendo la información y delimitando los permisos de acuerdo con las responsabilidades de cada actor del sistema.
- Capacidad de análisis y reportabilidad, en la cual la arquitectura contempla un módulo de reportes y métricas que facilitan la sistematización de los datos recabados, apoyando la detección de patrones de riesgo y la toma de decisiones estratégicas en materia de seguridad laboral.

1.5 Objetivos generales y específicos de la Tesina

Objetivo general

Diseñar e implementar la arquitectura de software de la plataforma CoreSafe, compuesta por un sistema web y móvil integrado, que permita la asignación, captura y análisis digital del Programa de Registro Conductual (PRC) en entornos industriales y mineros, asegurando la trazabilidad de la información, el soporte a la operación en terreno y la integración con los sistemas corporativos de INDEMIN.

Objetivos específicos

Para alcanzar el objetivo general, se plantean los siguientes objetivos específicos:

1. Analizar los procesos actuales de registro conductual en INDEMIN y los requerimientos funcionales y no funcionales que debe cubrir la plataforma CoreSafe
2. Diseñar una arquitectura de software modular que considere los servicios de autenticación, gestión de formularios, asignación de registros, análisis y reportabilidad

3. Implementar un prototipo funcional de la arquitectura propuesta, que contemple tanto la aplicación web como móvil con capacidad de operación offline-first.
4. Integrar la solución con los sistemas corporativos de INDEMIN mediante una API centralizada que asegure consistencia en la información de usuarios, faenas y maquinarias.
5. Evaluar la arquitectura implementada a través de pruebas funcionales y de rendimiento, verificando su alineación con los criterios de trazabilidad, eficiencia y seguridad.

1.6 Justificación del proyecto

La gestión preventiva en operaciones industriales y mineras exige contar con mecanismos que permitan anticipar riesgos, corregir desviaciones y garantizar la seguridad de los trabajadores. En este escenario, la digitalización de los registros conductuales se vuelve una necesidad estratégica, dado que posibilita transformar un proceso actualmente manual y fragmentado en un sistema integrado, trazable y eficiente.

El desarrollo de una plataforma tecnológica como CoreSafe se justifica con los siguientes aspectos:

1. **Optimización de recursos:** Al eliminar los registros manuales y la duplicidad de esfuerzos en la transcripción de datos, se reducen tiempos operativos y se optimiza el uso de los recursos humanos, lo que coincide con lo expuesto por *Sternad Zabukovšek, Jordan y Bobek (2023)*, quienes destacan que *la digitalización y automatización de procesos disminuye significativamente el tiempo y los costos asociados a la gestión manual de documentos [3]*
2. **Mejora la trazabilidad:** La centralización de la información en un repositorio digital permite un seguimiento oportuno de los planes de acción y facilita el control de las observaciones realizadas en terreno. De acuerdo con *Bachche, Rydström y Bjelkemyr (2023)*, *la integración digital favorece la recopilación continua de datos y el análisis predictivo, fortaleciendo la visibilidad sobre los riesgos y las oportunidades de mejora [2]*.
3. **Disponibilidad y acceso de datos:** El acceso a la información en tiempo real, tanto desde dispositivos móviles como desde la web, facilita la toma de decisiones basadas en evidencia y mejora la retroalimentación hacia los equipos operativos en terreno. Esta conectividad tecnológica respalda una gestión preventiva ágil y adaptable frente a cambios en las condiciones de trabajo o los riesgos detectados.
4. **Cumplimiento normativo:** La sistematización de los registros permite demostrar el cumplimiento de normativas de seguridad y protocolos de prevención frente a auditorías internas y externas

Justificación desde la perspectiva arquitectónica

Desde el punto de vista de la ingeniería de software, la justificación del proyecto radica en la necesidad de contar con una arquitectura tecnológica sólida y escalable que garantice la continuidad operacional y la integridad de los datos dentro del entorno corporativo de INDEMIN.

La arquitectura de CoreSafe se concibe como un componente esencial para asegurar la calidad y sostenibilidad del sistema, aportando en los siguientes aspectos:



1. Escalabilidad y mantenibilidad: Mediante un diseño modular basado en principios de separación de responsabilidades, se facilita la evolución del sistema sin comprometer su estabilidad, permitiendo la incorporación futura de nuevos módulos o funcionalidades.
2. Integración corporativa: La arquitectura incorpora una API centralizada que posibilita la interoperabilidad con los sistemas internos de INDEMIN (Checklist, SGO y RP), asegurando la consistencia de la información y la automatización de los procesos de sincronización.
3. Confiabilidad y operación en terreno: La adopción del enfoque offline-first en la aplicación móvil garantiza la continuidad de las operaciones en ambientes con conectividad limitada, resguardando la integridad de los datos hasta su sincronización.
4. Seguridad y control de accesos: Se implementan mecanismos de autenticación federada y control de roles (RBAC), los cuales protegen la información sensible y aseguran que cada usuario acceda únicamente a los recursos autorizados.
5. Sustentabilidad tecnológica: El uso de estándares abiertos, patrones de diseño reconocidos y prácticas DevOps promueve la reutilización de componentes, la trazabilidad del código y la eficiencia en la gestión del ciclo de vida del software.

En consecuencia, la arquitectura propuesta no solo soporta los objetivos funcionales de digitalizar el Programa de Registro Conductual, sino que también garantiza la calidad técnica, la integridad de la información y la escalabilidad del sistema, factores críticos para la continuidad operativa y la mejora continua de la gestión preventiva en INDEMIN.

1.7 Metodología

La metodología empleada en este trabajo de título se fundamenta en un enfoque de desarrollo iterativo e incremental, sustentando en las buenas prácticas de las metodologías ágiles, particularmente en la combinación Scrum-Kanban (Scrumban) para la planificación, seguimiento y control de avances. Este modelo permite abordar de forma progresiva la complejidad del proyecto, garantizando entregas funcionales en cada iteración y facilitando la incorporación de mejoras continuas basadas en la retroalimentación de los usuarios clave (observadores de conducta, línea de mando y administradores)

De manera complementaria, se integran principios y herramientas propias de la ingeniería de software orientada a la arquitectura, con el propósito de asegurar la calidad estructural, mantenibilidad y escalabilidad de la plataforma CoreSafe, tanto en su versión web como móvil. Y se realizaron pruebas sobre un entorno controlado compuesto por 5 usuarios, 3 formularios PRC y un escenario de sincronización con 56 observaciones.

Herramientas de la ingeniería de software aplicadas

1. Modelado de requisitos

Se aplicó un proceso sistemático de levantamiento y análisis de requisitos funcionales y no funcionales. Este modelo permitió capturar las necesidades de los distintos perfiles de usuario (observadores de conducta, línea de mando y administradores) y definir atributos de calidad relevantes para la arquitectura, tales como seguridad, trazabilidad, disponibilidad y soporte técnico. Los requisitos fueron representados mediante diagramas de caso de uso y descripciones formales específicas.

2. **Diseño arquitectónico**

El diseño se fundamenta en un enfoque de monolito modular, que permite estructurar la plataforma CoreSafe en módulos independientes, manteniendo una base de despliegue unificada. Esta decisión busca equilibrar la simplicidad operativa del modelo monolítico con los beneficios de escalabilidad, mantenibilidad y evolución propia de las arquitecturas distribuidas.

Se aplicaron patrones arquitectónicos de alta cohesión y bajo acoplamiento, y se adoptó el patrón offline-first en la aplicación móvil, garantizando la continuidad operativa en entornos sin conectividad. El diseño fue documentado mediante diagramas UML (componentes, despliegue, secuencia y clases), representando las interacciones, dependencias y flujos de comunicación entre los módulos del sistema.

3. **Validación y pruebas**

Se realizan pruebas en tres niveles

1. Pruebas funcionales, para verificar la correcta operación de los módulos de autenticación, asignación, captura de formularios y generación de reportes.
2. Pruebas de integración, enfocadas en validar la interoperabilidad entre CoreSafe y la API corporativa de INDEMIN.
3. Pruebas de rendimiento, orientadas a evaluar el desempeño en escenarios de sincronización offline/online, los tiempos de respuesta del sistema y la estabilidad general de la plataforma.

Estas pruebas permiten verificar el cumplimiento de los requisitos y la robustez de la arquitectura propuesta, asegurando su alineación con los objetivos del proyecto.

1.8 Breve descripción de la organización del informe en capítulos

El presente documento se encuentra estructurado en capítulos que orientan al lector a través del desarrollo del proyecto CoreSafe, abarcando desde su concepción inicial hasta la exposición de los resultados y las conclusiones. A continuación, se ofrece una descripción general del contenido de cada capítulo.

Capítulo 1: Introducción

Presenta los antecedentes, contexto y problemática que originan el desarrollo del proyecto. Se exponen los objetivos generales y específicos, la justificación de la propuesta, la metodología empleada y una síntesis de la organización del informe.

Capítulo 2: Marco Teórico y Conceptual

Desarrolla los fundamentos teóricos que sustentan el trabajo, abarcando los conceptos asociados al Programa de Registro Conductual (PRC), la cultura de seguridad y los principios de la digitalización aplicada a la gestión preventiva. Asimismo, se exponen los elementos técnicos y arquitectónicos relevantes para el diseño de la plataforma CoreSafe.

Capítulo 3: Diseño e Implementación de la Solución

Detalla el proceso de diseño arquitectónico, la definición de los componentes del sistema y las decisiones tecnológicas adoptadas. Se describen los patrones arquitectónicos utilizados, la integración con la API corporativa de INDEMIN y la aplicación del enfoque offline-first en la app móvil. Además, se abordan los procedimientos de implementación y las estrategias de validación.

Capítulo 4: Resultados y Evaluación



Presenta los resultados obtenidos a partir del desarrollo e implementación del sistema. Se analizan los indicadores de desempeño, la validación funcional y la contribución del sistema en términos de eficiencia, trazabilidad y mejora de la gestión preventiva.

Capítulo 5: Conclusiones y Trabajos Futuros

Reflexiona sobre los logros alcanzados, las limitaciones del proyecto y las oportunidades de mejora identificadas. Se proponen posibles líneas de evolución tecnológica y arquitectónica para futuras versiones de la plataforma. Finalmente, se incluyen los agradecimientos y la lista de referencias bibliográficas utilizadas.



2 Estado del arte y marco conceptual

2.1 Estado del arte

En los últimos años, la digitalización de los procesos de seguridad laboral ha tomado una alta relevancia en industrias de alta complejidad, como la minería, donde la prevención de riesgos resulta esencial para garantizar la continuidad operacional y el resguardo de las personas. La transformación digital en el ámbito de la gestión de seguridad y salud ocupacional ha permitido automatizar inspecciones, formularios y auditorías, mejorando la trazabilidad de los registros y la capacidad de respuesta ante eventos críticos.

Soluciones móviles existentes

De esta forma, se ha consolidado una categoría de plataformas EHS (Environment, Health & Safety) que digitalizan inspecciones checklist y reportes de incidentes desde dispositivos móviles. Entre las más reconocidas se destacan SafetyCulture (iAuditor) e Intelx, las cuales permiten la creación de formularios y flujos de aprobación, captura en terreno con evidencia, paneles de indicadores y conectores con suites corporativas. Estas soluciones han validado la viabilidad tecnológica del registro conductual en campo y la adopción de experiencias móviles (UX) adaptadas para auditores y supervisores.

No obstante, al ser productos horizontales, su cobertura del Programa de Registro Conductual (PRC) suele ser genérica, enfocándose principalmente en auditorías y no en métricas conductuales específicas —como el seguimiento de conductas seguras/riesgosas, barreras y planes de acción conductual—. En consecuencia, estos sistemas suelen depender de conectores genéricos o incurrir en altos costos de personalización para ajustarse a los requerimientos particulares de cada organización.

Arquitecturas móviles y enfoque offline-first

En entornos industriales y mineros, donde la conectividad es limitada o intermitente, ha cobrado relevancia la adopción de arquitecturas offline-first para garantizar la continuidad operativa y la integridad de los datos recolectados en terreno. Este enfoque técnico permite que las aplicaciones móviles almacenen y procesen la información localmente, sincronizándola posteriormente con los servidores corporativos cuando la conexión está disponible.

A nivel arquitectónico, esta estrategia requiere mecanismos de almacenamiento local transaccional, sincronización diferida bidireccional, control de versiones y políticas de resolución de conflictos entre datos locales y remotos. Asimismo, se complementa con medidas de seguridad avanzada —como cifrado de datos en reposo y en tránsito, autenticación basada en tokens y auditoría de cambios— que garantizan la confidencialidad y trazabilidad de la información incluso en condiciones adversas de red.

2.2 Marco conceptual

El presente apartado tiene por objetivo establecer el marco conceptual que sustenta el desarrollo de la plataforma CoreSafe, abordando los fundamentos teóricos, terminológico y técnicos que dan soporte a la solución propuesta en el contexto operativo de Indemin. Comprender estos conceptos resulta esencial para vincular el dominio organizacional del Programa de Registro Conductual (PRC) con los principios de la arquitectura de software, en tanto disciplina que orienta la estructuración, escalabilidad y mantenibilidad de sistemas complejos.

En este sentido, el marco conceptual se construye a partir de dos dimensiones complementarias.

Dominio de aplicación: Programa de Registro Conductual (PRC)

Esta dimensión se encuentra centrada en la gestión conductual y la seguridad laboral dentro de faenas industriales y mineras. En este dominio se definen los elementos claves del PRC, las observaciones de conducta, las conductas seguras y riesgosas, así como los procesos asociados a la trazabilidad y asignación de formularios en terrenos

Para comprender la propuesta de solución en esta tesina, es necesario definir los principales conceptos que se aborda para la aplicación CoreSafe en el contexto Indemin

- Programa de Registros Conductuales (PRC): Instrumento de gestión preventiva que permite programar y organizar la observación de conductas en faenas, a fin de identificar desviaciones y promover prácticas seguras.
- Observaciones de conducta: Las observaciones de conducta constituyen el núcleo operativo del PRC, al ser el instrumento mediante el cual se capturan los comportamientos observados en los trabajadores durante la ejecución de sus tareas. Estas observaciones se plasman en formularios estructurados que incluyen información sobre el contexto de la faena, el área, la maquinaria involucrada, el turno y los factores ambientales que puedan influir en la conducta. *Según Campos (2015) [1], las observaciones sistemáticas permiten obtener datos objetivos sobre las prácticas laborales, facilitando el análisis posterior de tendencias y la identificación de áreas críticas en materia de seguridad.*
- Conducta segura: Acción observable que contribuye a eliminar o minimizar la exposición a riesgos, garantizando el cumplimiento de los procedimientos establecidos
- Conducta riesgosa: Acción observable que incrementa la probabilidad de accidente, medida en escalas de criticidad.
- Trazabilidad: Capacidad de dar seguimiento completo a un dato desde su captura en terreno hasta su análisis y reporte final.
- Offline-first: Paradigma de diseño de aplicaciones móviles que prioriza el funcionamiento autónomo en ausencia de conectividad, asegurando la posterior sincronización de datos.

- Asignación: Un usuario/rol tiene una asignación de formularios a los que puede contestar definiendo su alcance a la faena en la cual trabaja y maquinaria que se encuentre disponible.

Aspectos técnicos y arquitectónicos

En este marco, la plataforma CoreSafe busca materializar tecnológicamente el proceso de digitalización del procedimiento manual que implicaban el uso de papel y su posterior transcripción en las hojas de cálculo. En ese sentido su diseño propone un sistema compuesto por una aplicación web y una aplicación móvil integradas a través de un API corporativo, que permite sincronizar la información proveniente de distintas faenas y dispositivos en tiempo real. Esta integración asegura la consistencia de los datos y facilita la interoperabilidad entre los distintos módulos del ecosistema digital de Indemin.

El diseño de la plataforma CoreSafe se fundamenta en los principios y buenas prácticas de la arquitectura de software, entendida como la estructura fundamental de un sistema compuesta por sus componentes, las relaciones entre ellos y las decisiones que orientan su evolución y mantenimiento (Bass, Clements & Kazman, 2021). La arquitectura no solo define la organización técnica del sistema, sino que actúa como un marco estratégico que asegura la alineación entre los objetivos funcionales del negocio y los requerimientos de calidad del producto.

De acuerdo con Gandhi, Richards y Ford (2024), la arquitectura de software puede comprenderse a partir de cuatro dimensiones fundamentales: las características arquitectónicas, las decisiones arquitectónicas, los componentes lógicos y el estilo arquitectónico. Estas dimensiones permiten analizar un sistema no solo desde su estructura técnica, sino también desde las decisiones y atributos de calidad que determinan su comportamiento y evolución. En la Figura 2 se ilustran estas dimensiones conceptuales, las cuales orientan el diseño arquitectónico de CoreSafe en su rol de plataforma integrada para la gestión conductual preventiva.

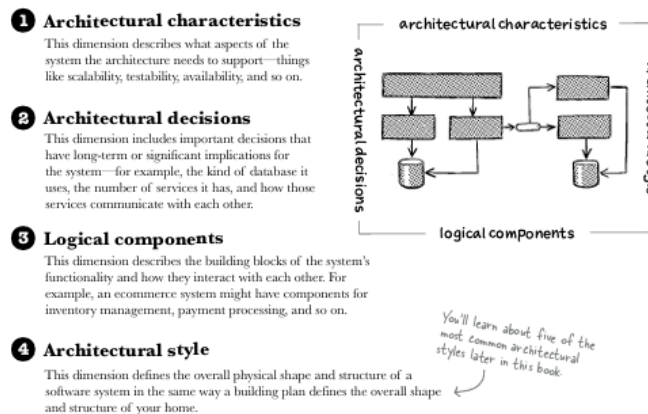


Figura 1. Dimensiones de la arquitectura de software

Fuente: Gandhi, R., Richards, M., & Ford, N. (2024). Head First Software Architecture: A Learner's Guide to Architectural Thinking. O'Reilly Media.

Desde esta perspectiva, la arquitectura de CoreSafe fue concebida bajo un enfoque modular y evolutivo, permitiendo la integración progresiva de funcionalidades sin comprometer la estabilidad del sistema. Esta decisión responde a la necesidad de garantizar escalabilidad, mantenibilidad y trazabilidad, atributos esenciales en entornos donde los datos de seguridad deben ser gestionados de forma continua, segura y verificable.

Arquitectura modular y estilo monolito modular

La adopción de un estilo arquitectónico monolito modular constituye una estrategia de diseño intermedia entre los sistemas monolíticos tradicionales y los basados en microservicios. Este enfoque permite organizar el código en módulos independientes que encapsulan dominios funcionales específicos, facilitando su mantenimiento y eventual desacoplamiento futuro (Garzas & Yilmaz, 2022) [7]. En el caso de CoreSafe, cada módulo responde a un dominio del PRC, como la gestión de observaciones, la administración de usuarios o la sincronización con la API corporativa, garantizando una separación lógica clara y una evolución controlada del sistema.

Según Newman (2020) [10], el **monolito modular** representa una alternativa eficaz para organizaciones que buscan altos niveles de paralelismo y simplicidad en el despliegue, evitando la complejidad operativa que implica la adopción prematura de microservicios. Este enfoque permite construir sistemas robustos y fácilmente mantenibles, siempre que los límites de los módulos estén bien definidos y alineados con el dominio del negocio, tal como se observa en la Figura 2.

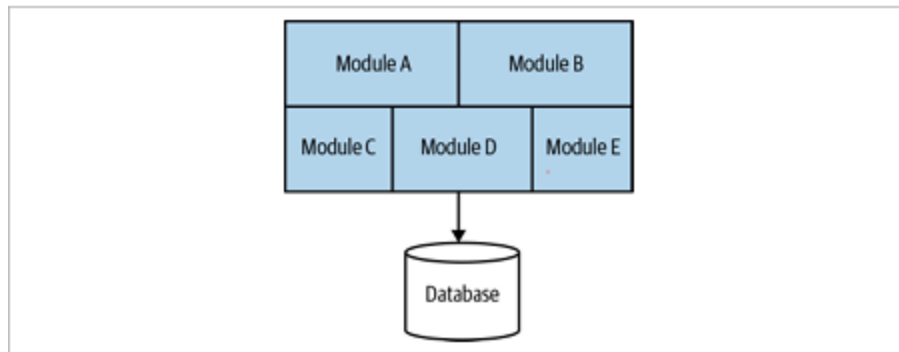


Figura 2. Representación de un monolito modular: el código interno se descompone en módulos

Fuente: Newman, S. (2020). Monolith to Microservices: Evolutionary Patterns to Transform Your Monolith. O'Reilly Media.

Esta decisión resulta particularmente adecuada para el entorno de INDEMIN, donde la madurez tecnológica y la necesidad de estabilidad operativa hacen que la modularidad interna sea prioritaria frente a la distribución total del sistema. De este modo, CoreSafe se concibe como un monolito modular con potencial evolutivo, preparado para futuras etapas de desacoplamiento sin comprometer la integridad funcional ni la trazabilidad de los datos.



Integración mediante API Gateway

La interoperabilidad con los sistemas corporativos de INDEMIN se logra mediante un API Gateway, componente que actúa como punto de entrada único para las solicitudes internas y externas del sistema. Este patrón arquitectónico facilita la gestión de seguridad, la autenticación de usuarios, el control de acceso y la trazabilidad de las operaciones.

En el contexto de CoreSafe, el API Gateway orquesta la comunicación entre la aplicación web, la aplicación móvil y los servicios corporativos, garantizando la integridad y consistencia de los datos. Además, permite incorporar validaciones y auditorías, lo que refuerza los principios de trazabilidad y seguridad de la información.

Patrón Offline-First y disponibilidad en entornos de baja conectividad

Uno de los desafíos más relevantes del entorno minero e industrial en que opera INDEMIN es la intermitencia de la conectividad. Para responder a esta condición, CoreSafe implementa el patrón de diseño offline-first, que prioriza la capacidad de la aplicación móvil de operar de manera autónoma sin requerir conexión permanente a internet.

Este enfoque mejora la disponibilidad del sistema, reduce la pérdida de datos y garantiza una experiencia continua, especialmente en escenarios de conectividad limitada. CoreSafe incorpora mecanismos de almacenamiento local y sincronización diferida, asegurando que las observaciones registradas en terreno se integren automáticamente al servidor central una vez restablecida la conexión.

Requerimientos no funcionales como guías arquitectónicas

Los requerimientos no funcionales constituyen un conjunto de atributos de calidad que orientan las decisiones estructurales y tecnológicas de la arquitectura de software. A diferencia de los requerimientos funcionales que describen lo que el sistema debe hacer, los no funcionales especifican cómo debe comportarse, definiendo niveles esperados de rendimiento, disponibilidad, seguridad, escalabilidad o mantenibilidad. Estos atributos representan criterios de éxito a largo plazo, pues determinan la sostenibilidad técnica, la eficiencia operativa y la capacidad de evolución del sistema.

En el contexto de la plataforma CoreSafe, los requerimientos no funcionales adquieren un papel fundamental, ya que la solución debe garantizar la disponibilidad del servicio incluso en entornos de conectividad limitada, la trazabilidad de cada registro conductual y la seguridad de los datos sensibles capturados en faenas industriales. De este modo, dichos atributos guían las decisiones arquitectónicas relacionadas con la adopción del patrón offline-first, la integración mediante API Gateway y la estructuración modular del sistema.

De acuerdo con Ford, Parsons, Kua y Sadalage (2024), *la arquitectura de software puede entenderse como la unión entre los requerimientos del dominio y las características arquitectónicas, conocidas también como nonfunctional requirements, system quality attributes o, de forma más general, "-ilities" (por ejemplo: scalability, reliability, maintainability, performance). Estas características constituyen capacidades críticas que determinan tanto el éxito inicial del proyecto como su mantenibilidad futura, actuando como los pilares que permiten la evolución continua de la solución.*

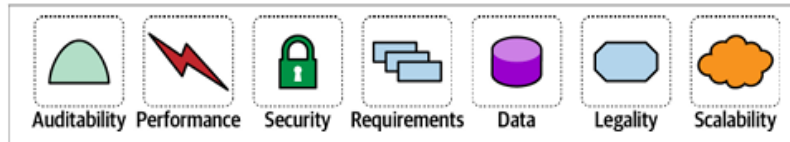


Figura 3. Alcance integral de la arquitectura de software: requerimientos y características arquitectónicas

Fuente: Ford, N., Parsons, R., Kua, P., & Sadalage, P. (2024). Building Evolutionary Architectures: Automated Software Governance (2nd ed.). O'Reilly Media.

En este enfoque, la arquitectura se concibe como un sistema evolutivo y gobernado por atributos de calidad, donde cada decisión técnica debe alinearse con mecanismos de validación y métricas que garanticen su cumplimiento en el tiempo. Para Ford et al. (2024) [9], *esta práctica —denominada automated software governance— permite evaluar continuamente la salud estructural del sistema, asegurando que los cambios futuros no comprometan los principios arquitectónicos esenciales.*

La integración de este enfoque en CoreSafe refuerza la necesidad de una gobernanza técnica sostenida, capaz de balancear la incorporación de nuevas funcionalidades con la preservación de los atributos de calidad definidos desde el diseño inicial. En consecuencia, los requerimientos no funcionales dejan de ser simples restricciones técnicas y pasan a constituir el núcleo de la arquitectura del sistema, orientando su diseño, implementación y evolución.

3 Desarrollo de la Solución

Se describirá el proceso de diseño, construcción e implementación de la plataforma CoreSafe, a partir de la arquitectura propuesta en los capítulos anteriores. Se desarrolla los componentes tecnológicos desarrollados, la estructura interna del sistema, los mecanismos de integración con los servicios con los servicios corporativos de INDEMIN y la estrategia utilizada para garantizar la disponibilidad, trazabilidad y confiabilidad del sistema tanto en su versión web como móvil.

Así mismo, se exponen los elementos técnicos que permitieron materializar un prototipo funcional, capaz de operar en entornos reales de faena industrial, donde las condiciones de conectividad, volumen de datos y exigencias de seguridad constituyen un desafío significativo para los sistemas de información.

3.1 Visión General de la Solución

CoreSafe se concibe como un ecosistema tecnológico destinado a digitalizar el PRC, integrando procesos actualmente manuales en una plataforma centralizada, trazable y disponible en terreno. La solución se compone de cuatro elementos principales:

1. Aplicación Web: Utilizada por administradores y línea de mando para la creación de plantillas, programación de PRC, visualización de métricas y gestión de usuarios.
2. Aplicación Móvil: Utilizada por los observadores de conducta en terreno para registrar observaciones en modalidad offline y online

3. Api Gateway corporativo: punto único de integración con el ecosistema digital de INDEMIN, desde donde se obtiene usuarios, faenas y maquinaria.
4. Base de datos centralizada: Repositorio único para formularios, plantillas, respuestas y métricas analíticas.

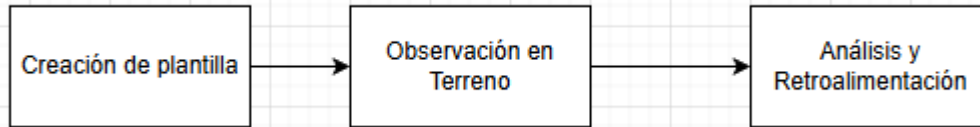


Figura 4. Diseño operacional del sistema

El diseño operacional del sistema se basa en un ciclo continuo compuesto por los 3 elementos señalados en la figura 4. Desde CoreSafe se establece la trazabilidad desde el momento en que se crea un formulario hasta su cierre analítico, garantizando integridad, consistencia y disponibilidad de la información en cada etapa.

3.2 Arquitectura de la Plataforma CoreSafe Servicio

La arquitectura de CoreSafe se propuso implementar siguiendo un enfoque monolito modular, sobre el cual se construye una estructura altamente cohesionada por dominios funcionales, pero expuesta mediante principios modernos de desacoplamiento y flexibilidad. El sistema no entrega interfaces visuales desde el servidor, por el contrario, opera bajo un paradigma headless, donde la totalidad de la lógica de negocio y acceso a datos se ofrece exclusivamente mediante una API REST estandarizada

De esta manera CoreSafe se organiza sobre tres niveles tecnológicos:

1. Capa de servicio (Monolito Modular Headless)	Aloja los cinco módulos internos (Autenticación, Gestión de Formularios, Asignación PRC, Captura Conductual, Analítica), cada uno expuesto mediante endpoints REST independientes.
2. Capa Backend for Frontend (BFF)	Una capa intermedia que optimiza el consumo de dichos módulos según el tipo de cliente (Web o Móvil), agregando, filtrando o transformando respuestas.
3. Capa de Cliente	Aplicaciones Web y Móvil construidas de manera independiente, las cuales consumen la API a través del BFF que les corresponde.

Tabla 1. Niveles tecnológicos CoreSafe

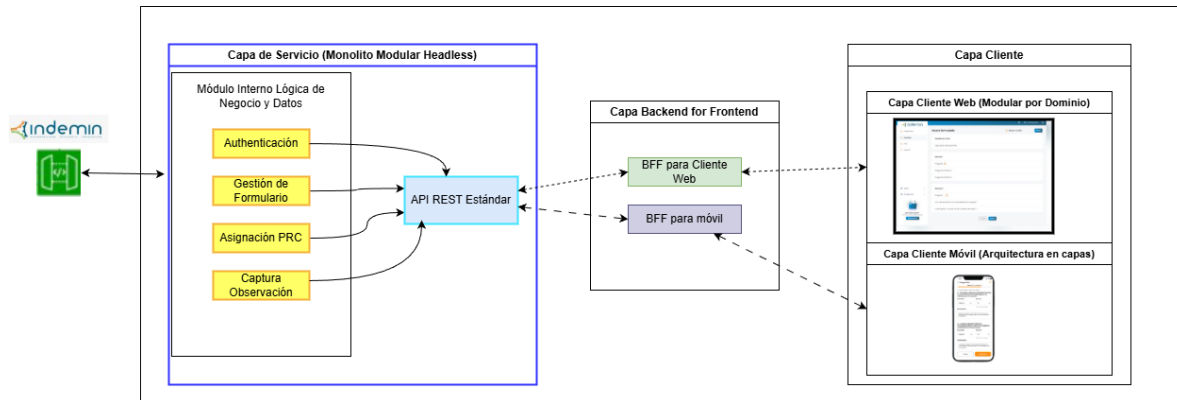


Figura 5. Diagrama Arquitectura General por Capa CoreSafe

Este diseño asegura separación completa entre la lógica central y las interfaces, permitiendo que la plataforma sea extensible hacia nuevos clientes (desktop, móvil y dashboards corporativos) sin modificar el backend ni afectar los módulos internos.

Arquitectura Lógica: Monolito Modular Headless

La arquitectura lógica de CoreSafe adopta tres principios fundamentales

A. Principio Headless (API REST como única interfaz)

- El monolito modular funciona exclusivamente como backend, eliminando cualquier renderización de vistas HTML o componentes UI tradicionales, siendo sus características las siguientes
- API REST se utiliza como contrato externo, esto quiere decir que todas las capacidades de los módulos internos son accesibles únicamente mediante endpoints RESTful, definidos de acuerdo con los verbos HTTP estándar (GET, POST, PUT, Delete)
- Mantiene una separación estricta de responsabilidades, ya que la UI no reside en el backend. La presentación (web y móvil) es responsabilidad de los clientes externos, lo que permite evolucionar la interfaz sin alterar la lógica del sistema.
- Cuenta con flexibilidad de interoperabilidad, al no depender de plantillas de servidor, el backend sirve como fuente de datos universal para múltiples tipos de cliente, como los anteriormente mencionados: web admin, app móvil offline-first, microservicios futuros, tableros analíticos

Este enfoque maximiza la reutilización y habilita un ecosistema escalable, alineado con arquitecturas modernas orientadas a servicios.

1. Módulo de Autenticación: Interacción con API corporativa de inicio de sesión, administración de tokens y gestión de roles (RBAC)

2. Módulo de Gestión de Formularios y plantillas: Creación de plantillas dinámicas, parametrización de preguntas y definición de indicadores conductuales.
3. Módulo de Asignación PRC: Vinculación entre faenas, maquinarias, observadores y formularios.
4. Módulo de Captura Observación: Registro de observaciones seguras y riesgosas, descripción de barreras y comentarios contextuales.
5. Módulo Analítico y de Métricas: Generación de indicadores, tendencia de conductas, criticidad y cumplimiento del PRC.

Cada Módulo opera con cohesión interna y bajo acoplamiento entre sí, siguiendo principios de orientación al dominio. Esta estructura da claridad conceptual y facilita la evolución futura del sistema.

B. Implementación del Monolito Modular con Laravel (MVC extendido por módulos)

Laravel al ser un framework define un marco de trabajo con un patrón arquitectónico definido como MVC, Modelo Vista Controlador. El cual junto con el estilo Headless y una estructuración modular basada en dominios. El patrón MVC se mantiene, pero reinterpretado

La reestructuración se logra de la siguiente manera. El rol dentro del monolito modular en el componente MVC Model, se representó entidades del dominio y encapsulando reglas de negocio y persistencia. Dentro del Controller (Controlador), expone endpoints REST del módulo, coordinando la lógica mediante services y retornando JSON. Y por último el componente View (Vista), fue eliminada del servidor y pasa a ser la respuesta JSON, ya que la responsabilidad de renderización recae en el cliente Web/Movil.

La consecuencia directa es un backend cohesionado, sin acoplamiento a framework visuales y optimizado para operar como una API escalable. Y la estructura de directorios refleja los cinco módulos principales, manteniendo la lógica organizada por dominio y facilitando mantenibilidad y trazabilidad.

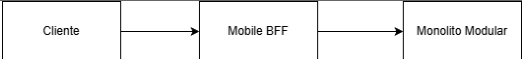
C. Patrón Backend for Frontend (BFF)

Para optimizar el consumo de la API según el tipo de cliente, se integró el patrón Backend for Frontend (BFF)

El BFF opera como una capa intermedia ubicada entre los clientes y el monolito modular.

Las funciones principales del BFF, es que en lugar de que un cliente móvil realice múltiples llamados a distintos módulos, el BFF combina las respuestas en una sola operación, ajusta los formatos de salida para optimizarlos según dispositivos y maneja la lógica específica de cada cliente, por ejemplo, la autenticación o filtros de paginación.

Los tipos de BFF definidos para CoreSafe son los siguientes:

Cliente	Capa BFF Asociada	Relación
App Móvil (offline-first)	Mobile BFF	 <pre> graph LR C[Cliente] --> MBFF[Mobile BFF] MBFF --> MM[Monolito Modular] </pre>



De esta manera se evita que la aplicación web o móvil deban conocer los detalles internos del backend o sus endpoints individuales, permitiendo obtener respuestas más optimizadas para el cliente

3.3 Arquitectura de la Aplicación Móvil CoreSafe

La aplicación móvil CoreSafe fue diseñada para operar en entornos exigentes y con conectividad limitada, por lo que su arquitectura se construyó bajo principios de separación de responsabilidades, modularidad y soporte nativo para modo offline. Su diseño se organiza siguiendo una arquitectura en capas, complementada con patrones de desarrollo que aseguran mantenimiento, escalabilidad y comportamiento predecible.

Arquitectura en Capas

La solución móvil adopta un modelo Layered Architecture, estructurado en cuatro niveles principales:

1) Capa de presentación

Implementada con Flutter, contiene las pantallas y componentes UI. Su responsabilidad se limita a la interacción con el usuario, delegando la lógica a capas inferiores.

2) Capa de Servicio

Orquesta la lógica de negocio y centraliza procesos como autenticación, comunicación con la API, acceso a datos y sincronización offline. Esta capa abstrae la complejidad del backend y de la persistencia local.

3) Capa de Modelos

Representa las entidades del dominio (usuarios, faenas, formularios, PRC, observaciones), facilitando la transferencia de datos entre capas mediante objetos.

4) Capa de Persistencia

Gestiona almacenamiento local mediante SQLite y SharedPreferences, garantizando disponibilidad de información en ausencia de conectividad.

Esta organización permite desacoplar la aplicación, facilitar pruebas y asegurar una evolución ordenada del código.

Patrones Arquitectónicos Aplicados

La implementación se apoya en varios patrones ampliamente reconocidos

- Service Layer: Encapsula la lógica de negocio y provee un punto de acceso único a operaciones críticas.
- Repository: Abstrae el origen de los datos (local vs remoto), permitiendo intercambiarlos sin afectar a la UI.
- Offline-First Prioriza el almacenamiento local y la sincronización diferida, asegurando operación continua en terreno.
- DTO y Factory Methods: facilitan la conversación entre objetos del dominio y estructuras JSON
- Singleton: garantiza consistencia en la gestión de sesiones, API y base de datos. Creando constantes reutilizables.

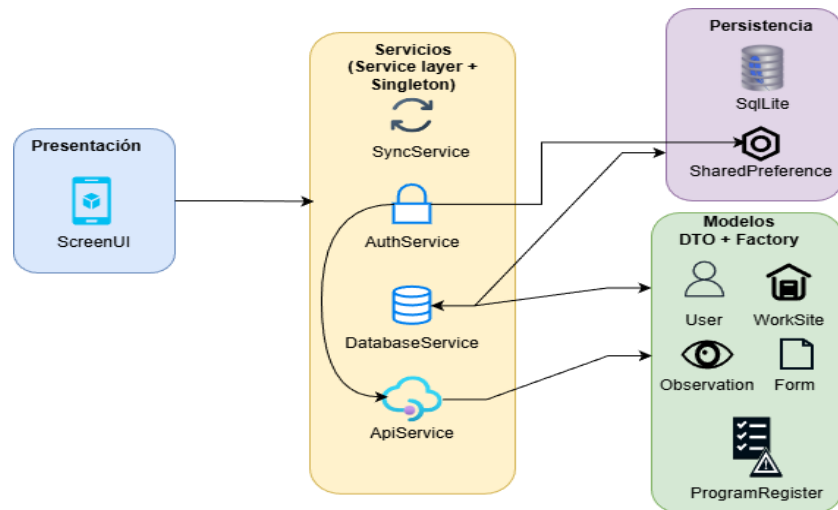


Figura 5. Diagrama Arquitectura Móvil

Comunicación con el Backend

La comunicación se realiza mediante la API REST del Monolito Modular o vía Mobile BFF, dependiendo del flujo requerido. La aplicación:

- Autentica mediante tokens seguro
- Maneja Headers automáticamente,
- Verifica la conectividad antes de sincronizar
- Reintenta operaciones pendientes al restablecer conexión

Este enfoque optimiza el uso de red y asegura robustez en zonas con conectividad intermitente.

Una de las decisiones más críticas tomadas a la hora de realizar esta arquitectura fue tomar un patrón arquitectónico first-offline. Ya que está les da prioridad a los datos locales, guardando y consultando los datos primeros en SQLite de esta manera permite a la aplicación funcionar sin conexión. Además, permite una sincronización Asíncrona, lo cual no bloquea operaciones del usuario. En el siguiente diagrama de secuencia se puede ver la diferencia entre una aplicación offline-first y una aplicación sin capacidad offline.

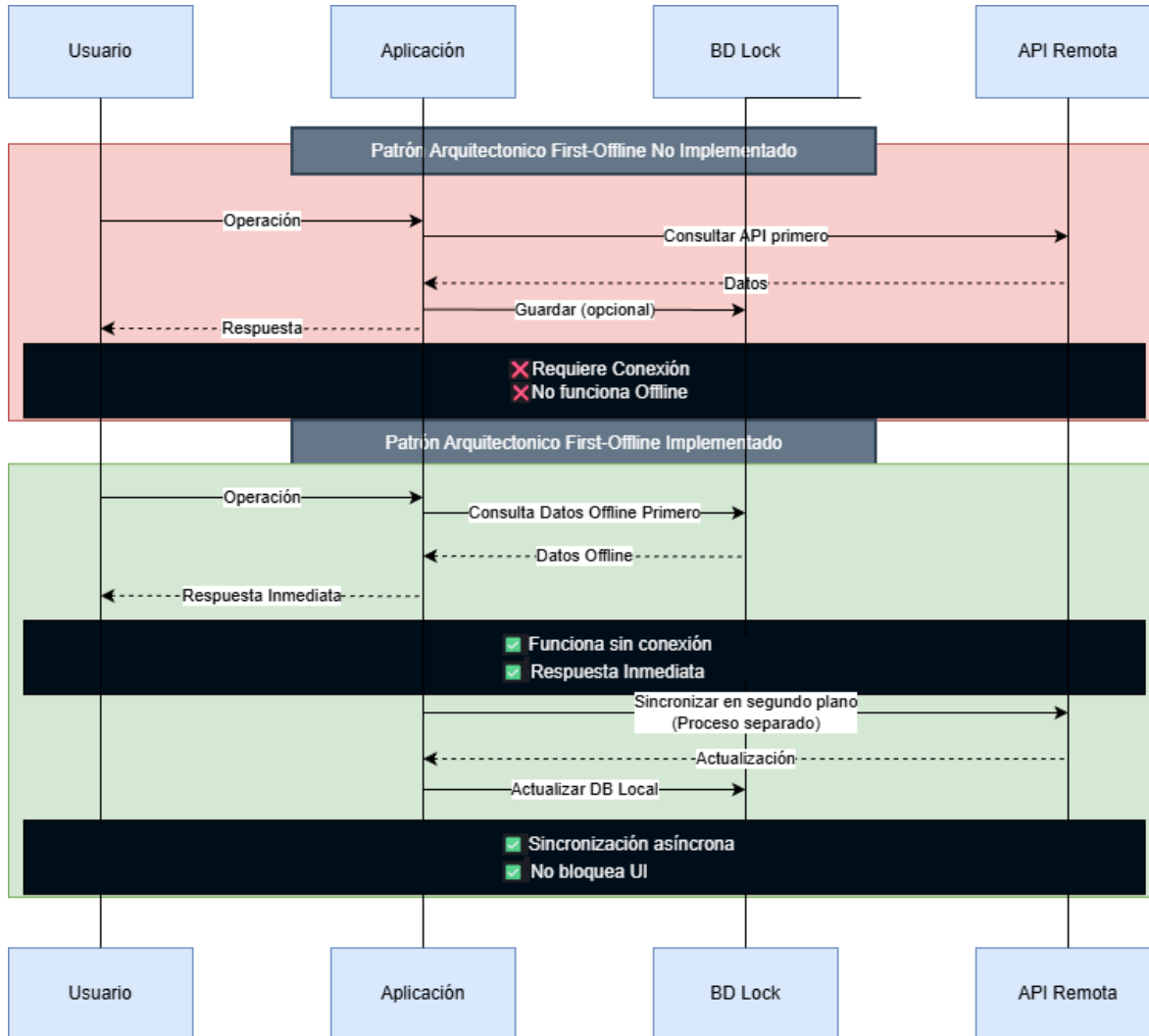


Figura 6. Diagrama de secuencia comparativo de arquitectura Online-First vs Offline-First

En síntesis, la arquitectura móvil descrita permite cumplir con el requisito crítico de operación en entornos de conectividad limitada, garantizando que el patrón offline-first se implemente de forma consistente con la API Headless del backend.

3.4 Arquitectura de la Aplicación Web CoreSafe

La aplicación web CoreSafe constituye el punto principal de administración del Programa de Registro Conductual (PRC). Su diseño se alinea con la arquitectura Headless del backend y con el modelo de modularidad por dominios previamente descritos en este. Para asegurar mantenibilidad, escalabilidad y organización clara de responsabilidades, la aplicación fue implementada utilizando Vue3, TypeScript, Pinia y Vue Router, siguiendo un enfoque modular y orientado al dominio.

Estilo Arquitectónico: Feature-Based Modular Architecture (Domain-Driven Design)

La aplicación adopta un estilo arquitectónico modular basado en features o dominios, similar al concepto monolito modular aplicado al frontend. Cada módulo agrupa los siguientes; Vistas, Componentes, Stores, Servicios y Tipos & utilidades. Los cuales corresponden a un dominio funcional del sistema, tales como Auth, PRC, Observaciones, Templates, Users.

Este enfoque de arquitectura de frontend se caracteriza por priorizar la alta cohesión interna de cada módulo, asegurando que los componentes relacionados a una funcionalidad específica permanezcan juntos y bien definidos. Simultáneamente, se logra una baja dependencia entre los diferentes dominios de negocio, lo que significa que los cambios en un área del sistema (como la gestión de clientes) tiene un impacto mínimo en otra (como el procesamiento de pagos). Esta clara separación y estructura no solo facilita enormemente el mantenimiento y la evolución del sistema CoreSafe a largo plazo, sino que también permite una escalabilidad orgánica donde la estructura del frontend puede crecer de manera natural y eficiente a medida que se incorporan o expanden los requisitos funcionales, reflejando así de manera directa y efectiva los dominios de negocio del sistema central.

Patrones Arquitectónicos Implementados

La aplicación web emplea múltiples patrones de diseños propios de Vue3 y del desarrollo moderno de frontend organizados por dominio, dentro de los cuales se enumerarán los principales

A. Composition API Pattern

La lógica reusable dentro de la aplicación se gestiona y encapsula eficientemente a través del uso de composables, los cuales actúan como funciones especializadas que permiten la composición flexible de comportamiento al agrupar la lógica de estado y reactividad de manera aislada y reusable. Esto no solo facilita la reutilización de funciones comunes a lo largo de diversos componentes, sino que también contribuye a un manejo más ordenado y limpio del estado local dentro de la aplicación, como se ejemplifica en composables específicos como `useAuth()`, `usePRC()`, o `useObservations()`.

B. State Management Pattern con Pinia

La gestión del estado global de la aplicación se implementa mediante la definición de stores específicos por cada dominio de negocio, lo que es fundamental para establecer una reactividad controlada a través de toda la aplicación, asegurando que las actualizaciones de datos sean predecibles y rastreables. Esta estrategia permite una clara separación entre el estado global y la lógica encapsulada en componentes o composables manteniendo la lógica de negocio lejos de la capa de datos shared, lo que a su vez garantiza una consistencia sólida y uniforme en los datos que se comporten entre los distintos módulos, con ejemplos dentro del proyecto son los siguientes stores como `auth.store.ts`, `prc.store.ts` y `observations.store.ts`

C. Service Layered Pattern

La capa de servicio abstrae la comunicación con la API REST del backend, proporcionando:

- Punto central de configuración de HTTP
- Reutilización en distintos módulos
- Mayor facilidad de pruebas
- Aislamiento frente a cambios en la API

En el proyecto se utilizaron `prc.service.ts`, `template.service.ts` y `user.service.ts`

D. Component-Based Architecture

En Vue los componentes se organizan en componentes específicos por dominio y componentes reutilizables, permitiendo separar la presentación de la lógica, de esta manera se fomenta una UI consistente.

E. Layered Architecture (Aplicada al Frontend)

La aplicación finalmente se estructura conceptualmente en cinco capas:

- 1- Capa de presentación (Views)
- 2- Capa de Lógica de Negocio (Composables)
- 3- Capa de Estado Global (Store)
- 4- Capa de Servicios (HTTP)
- 5- Capa de Datos Externa (API Backend)

Flujo de Datos y Comunicación

La comunicación sigue un flujo unidireccional, lo que simplifica el razonamiento sobre el estado y evita inconsistencias en la aplicación.

Capa/Componente	Rol Principal	Interacción con	Salida/Acción Típica
Vistas	Consumir composables y renderizar la interfaz de usuario (UI)	Composables	Eventos de usuarios, rendering de componentes
Composables	Encapsular lógica reutilizable y coordinar acciones	Stores y Servicios	Llamar a servicios o modificar Stores
Stores	Mantener y gestionar el estado global reactivo	Composables	Notificar cambios de estado a los composables (que actualizan las vistas).
Servicios	Ejecutar la lógica de negocio remota y realizar llamadas HTTP	Backend Headless /BFF	Realizar peticiones (GET,POST, etc.) a la API externa
API Externa	El Backend Headless/ BFF	Servicios	Responder con datos estructurados (JSON)

Este enfoque de flujo permite garantizar una claridad total en el recorrido de los datos y simplifica drásticamente los procesos de depuración al establecer una única fuente de verdad para el estado.

De esta manera, la arquitectura web modular complementa el backend Headless permitiendo que la gestión del PRC pueda evolucionar de forma independiente en la capa de presentación sin comprometer la lógica de negocio central.

3.5 Relación Entre la Arquitectura y los Atributos de Calidad

El diseño arquitectónico de CoreSafe no surge únicamente de la necesidad de digitalizar el PRC, sino también de la obligación de garantizar que la solución cumpla con atributos de calidad esenciales para su operación en entornos industriales y mineros. Cómo se ha ido narrando a lo largo del presente documento, la arquitectura seleccionada esta compuesta por un backend Headless basado en un monolito modular, capas BFF diferenciadas y aplicaciones Web y Móvil especializadas, las cuales se definieron específicamente para satisfacer requisitos de disponibilidad, mantenibilidad, rendimiento, escalabilidad y soporte multi-cliente. Resumido en una implementación como la mostrada en la siguiente imagen.

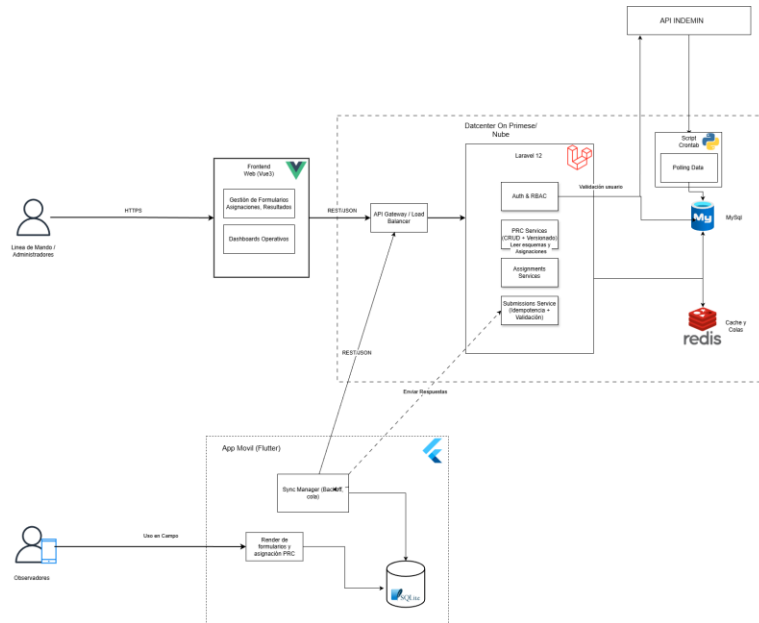


Figura 7. Diagrama de infraestructura CoreSafe

A continuación, se sintetizará cómo cada decisión arquitectónica se alinea con los atributos de calidad del sistema.

El patrón offline-first, aporta a la disponibilidad operacional. En la cual se implementa un almacenamiento local persistente, colas de sincronización y reconciliación diferida. Esto asegura que los observadores de conducta puedan capturar información incluso en condiciones de conectividad limitada o inexistente, uno de los desafíos característicos de las faenas mineras. En resumen, el patrón arquitectónico garantiza que haya continuidad del servicio en terreno, prevención de la pérdida de datos y experiencia de uso estable. Permitiendo que la disponibilidad no dependa completamente de la conectividad inalámbrica, sino de su propia capacidad para operar de manera autónoma

Con respecto al Monolito Modular, nos permite dar mantenibilidad y evolución controlada. Ya que cada dominio del PRC (Autenticación, plantillas, observaciones,



asignaciones, métricas) se organizan en módulos cohesionados y de bajo acoplamiento. Con este estilo arquitectónico se facilita la evolución del sistema sin afectar módulos no relacionados, permite realizar incorporaciones de nuevas funcionalidades de manera incremental, organiza claramente el código por dominios y reduce los costes de desarrollo y mantenimiento a largo plazo.

Con respecto al BFF nos entrega rendimiento y una mejor experiencia de usuario, ya que al diferenciar Web y Mobile permite que cada cliente consuma datos optimizados para sus necesidades, permitiendo generar una reducción de llamadas múltiples al backend, agregación y transformación de datos en un punto intermedio, respuestas más ligeras para los dispositivos móviles y una latencia percibida menor. De esta manera con el patrón BFF se mejora directamente la performance y la experiencia de usuario.

Y, por último, la razón para elegir arquitectura Headless (API REST) es que brinda escalabilidad y multi-cliente, dado que el backend adopta un enfoque Headless, donde toda la lógica de negocio se expone mediante API REST, no existe dependencias de frameworks para la interfaz y además permite que cualquier cliente externo pueda integrarse (Web, Móvil, Desktop, dashboards futuros). Y este desacoplamiento permite escalar horizontalmente mediante replicación de la API, agregar nuevos tipos de clientes sin alterar la lógica central y evolucionar de manera independiente el frontend respecto al backend

La convergencia de estas decisiones arquitectónicas asegura que CoreSafe cumpla los atributos de calidad definidos desde el levantamiento inicial de requerimientos. Cada patrón, estilo y estructura implementada responde a una necesidad explícita del sistema: operar en terreno, mantener consistencia, garantizar rendimiento, facilitar extensión futura y adaptarse a distintos canales de consumo.

Esta alineación entre arquitectura y atributos de calidad constituye uno de los principales aportes del proyecto, habilitando una plataforma sólida, sostenible y preparada para la evolución continua.

4 Resultados, Validación e Impacto de la Solución

El presente capítulo expone los resultados obtenidos tras la implementación del prototipo funcional de la plataforma CoreSafe, evaluando su comportamiento frente a los requerimientos previamente definidos y analizando el impacto generado en los procesos de registro conductual dentro del contexto operativo de INDEMIN. Se presenta además los indicadores derivados de las pruebas realizadas, la validación de atributos de calidad arquitectónicos y la contribución efectiva de la solución a la trazabilidad, disponibilidad y eficiencia de la gestión preventiva.

4.1 Resultados Generales de la implementación

La implementación de CoreSafe resultó en una plataforma coherente y completamente funcional al integrar varios componentes esenciales de la solución. El núcleo del sistema reside en un Backend Headless desarrollado en Laravel, el cual opera bajo un módulo de Monolito Modular y expone sus funcionalidades de manera exclusiva a través de una API REST bien definida. Para optimizar la experiencia del usuario, se incluyó una capa BFF, encargada de adaptar y consolidar el consumo de datos específicamente para las aplicaciones Web y Móvil.

Complementando el backend, se creó una aplicación Web con Vue3, la cual se estructura en módulos de dominio claros, adoptando patrones modernos como Composition API, el gestor de estados Pinia, una Service Layer dedicada y una arquitectura en capas bien definida. Además, la solución incorpora una Aplicación Móvil desarrollada en Flutter, que sigue una arquitectura en capa similar, pero con la característica clave de operar con un patrón offline-first y hacer uso de almacenamiento local para garantizar la funcionalidad incluso sin conexión.

Finalmente, la integración con los sistemas internos de INDEMIN se maneja a través de un API Gateway, permitiendo la sincronización corporativa continua de datos críticos como usuarios, faenas y maquinarias.

El prototipo funcional logra cubrir el ciclo completo del PRC indicado en la figura 4, página 14. Como resultado se demuestra que la plataforma resultante tiene viabilidad técnica y operativa, al lograr digitalizar un proceso que previamente dependía de papel, hojas de cálculo y transcripciones manuales.

4.2 Validación del Sistema Frente a los Requerimientos

Para poder realizar la validación, se ejecutó un escenario end-to-end donde un usuario observador inicia sesión, sincroniza su bandeja de PRC, registra una observación offline y se valida su aparición en el dashboard web tras recuperar conectividad. Y de igual manera se probó la administración de PRC desde la web

Validación de Requerimientos Funcionales

Los resultados de validación muestran que la arquitectura desarrollada satisface la mayoría de los requisitos funcionales definidos en la fase de análisis de los cuales se pueden destacar los siguientes para el prototipo de desarrollado:

Requisito Funcional	Resultado	Evidencia
Autenticación centralizada vía API corporativa	Cumplido	Integración con API Gateway
Administración de plantillas PRC	Cumplido	Frontend modular + API REST
Asignación de formularios a observadores	Cumplido	Módulo PRC en Web y API
Captura de observaciones en terreno	Cumplido	Mobile offline-first
Sincronización automática offline/online	Cumplido	Cola de operaciones + SyncService
Visualización y análisis inicial	Cumplido	Dashboard básico mediante Web BFF

Validación de Requerimientos No Funcionales

Los atributos de calidad arquitectónicos se evaluaron en relación con las necesidades de INDEMIN:

A. Disponibilidad

La aplicación Móvil opera sin conectividad gracias a almacenamiento local, sincronización diferida y patrón offline-first.



Resultado: Satisfactorio

B. Seguridad

- a. Backend Headless expuesto únicamente vía APIREST.
- b. Sincronización móvil eficiente incluso con múltiples observaciones

Resultado: Cumplido

C. Rendimiento

- a. Backend responde en promedio < 150ms en peticiones típicas probada desde Postman y dispositivo celular.
- b. Sincronización móvil eficiente incluso con múltiples observaciones.

Resultado: Cumplido.

D. Escalabilidad

- a. Modularidad por dominios en Backend y Web.
- b. Arquitectura en capas en Mobile.
- c. Patron BFF facilita incorporación de nuevos clientes o interfaces.

Resultado: Preparado para crecer a micro-servicio.

E. Mantenibilidad

- a. Backend modular por dominios.
- b. Web con Domain Driven Desing.
- c. Mobile con separación en capas y servicios reutilizables.

Resultado: Tener el proyecto bajo estos criterios arquitectónicos le permiten una alta mantenibilidad a futuro.

F. Trazabilidad

- a. Registros auditados de sincronizaciones.
- b. Identificación completa de origen, fecha, usuario y faena por cada PRC

Resultado: Aceptable.

4.3 Impacto Operacional en INDEMIN

La utilización de CoreSafe genera beneficios tangibles en la operación de seguridad y prevención de riesgos.

Eliminación del Registro Manual

La digitalización completa del proceso de registro marca un avance crucial al eliminar los formatos en papel y la subsecuente transcripción manual a Excel, lo cual era una fuente constante de inconsistencias y duplicidad de datos. Este cambio no solo reduce drásticamente los tiempos dedicados al proceso administrativo, sino que también eleva la calidad y fiabilidad del dato capturado, proporcionando una base más sólida para el análisis preventivo.

Operación continua en terreno

Un factor crítico para la operación en faenas es el diseño de la aplicación móvil con un patrón offline-first. Esta característica garantiza que el personal pueda capturar información crucial sin necesidad de conexión activa y asegurar que no haya pérdida de datos, ya que la información se sincroniza automáticamente en cuanto se restablece la



red. Esta capacidad de operación continua es fundamental en entornos mineros o industriales donde la conectividad es inherentemente intermitente.

Trazabilidad Completa del Proceso PRC

La arquitectura Robusta de CoreSafe permite alcanzar una trazabilidad completa del proceso del PRC. El sistema es capaz de reconstruir el contexto completo de cada registro, incluyendo quien realizó la observación, que maquinaria estuvo involucrada, qué conducta específica se registró y el momento preciso (cuándo y dónde). Esto permite mejorar las auditorías internas y asegurar el cumplimiento normativo de manera eficiente

Mejora en la Retroalimentación Preventiva

La digitalización garantiza que los registros de observaciones estén disponibles casi en tiempo real. Este acceso inmediato permite a los supervisores detectar tendencias riesgosas emergentes y les permite implementar acciones correctivas.

Integración Corporativa

Mediante el uso estratégico del API Gateway, CoreSafe ha logrado una integración corporativa uniforme con los sistemas internos de INDEMIN. Esto unifica la información sobre usuarios, faenas y maquinarias, eliminando la necesidad de administración paralela y redundante de datos.

4.4 Impacto Tecnológico y Arquitectónico

La implementación de CoreSafe ha logrado un impacto significativo que trasciende la mera resolución funcional del problema, elevando sustancialmente el nivel tecnológico del ecosistema de INDEMIN mediante la adaptación del paradigma Headless, el cual expone toda la lógica de negocio exclusivamente a través de una API REST, garantizando la independencia total entre frontend y backend para dar soporte a múltiples clientes (Web, Móvil Android y IOs) y permitir una evolución continua sin comprometer las integraciones existentes. Complementariamente, la integración de un BFF con segmentos específicos para web y móvil no solo reduce la carga sobre el backend principal, sino que simplifica las interfaces de datos y optimiza la experiencia del usuario final según el tipo de cliente. Este enfoque se refuerza con una arquitectura modular por dominios aplicada tanto en el backend como en las aplicaciones Web y Móvil, lo que aísla los dominios del PRC, facilita el mantenimiento, simplifica las pruebas y permite que cada módulo evolucione de manera completamente independiente. Finalmente, esta nueva arquitectura sienta las bases para futuras capacidades analíticas de alto nivel, pues al centralizar y estandarizar los datos y garantizar su trazabilidad completa, CoreSafe habilita la implementación futura de dashboards avanzados, analítica de tendencias y la aplicación de modelos de aprendizaje automático para la prevención de riesgos.

5 Conclusiones

5.1 Conclusiones Generales

El desarrollo de CoreSafe demuestra que es posible digitalizar de manera efectiva y arquitectónicamente sostenible el Programa de Registro Conductual (PRC), Reemplazando un proceso manual, fragmentado y carente de trazabilidad por una plataforma integrada, robusta y alineada con las necesidades operacionales de la industria minera y logística.

La arquitectura diseñada basada en un Monolito Modular Headless, complementado con una capa Backend for Frontend y clientes especializados Web y Móvil. Permitió construir un sistema coherente, extensible y preparado para operar en las condiciones reales de faena. La incorporación del patrón offline-first en la aplicación móvil y la integración con la API corporativa de INDEMIN garantizan continuidad operativa y consistencia de la información.

Los resultados demuestran que la solución desarrollada no solo cumple con los requerimientos funcionales y no funcionales establecidos, sino que mejora significativamente la trazabilidad, eficiencia y capacidad analítica del proceso preventivo.

5.2 Conclusiones por objetivo específico

1. Análisis del proceso y levantamiento de requisitos

Se logró caracterizar adecuadamente el flujo completo del PRC en INDEMIN, identificando brechas estructurales como la pérdida de trazabilidad, la dependencia del papel y la ausencia de retroalimentación inmediata. Este diagnóstico permitió orientar el diseño arquitectónico hacia atributos de disponibilidad, seguridad y consistencia de datos.

2. Diseño de una arquitectura modular y escalable

La arquitectura propuesta combina: Backend Headless modular, Capa BFF, Frontend Web modular por dominios, Cliente móvil con arquitectura en capas.

Este conjunto permitió desacoplar responsabilidades, fortalecer la mantenibilidad y habilitar el crecimiento futuro del sistema sin comprometer su estabilidad.

3. Implementación de un prototipo funcional

El prototipo demuestra la viabilidad técnica del diseño arquitectónico, integrando los componentes clave y validando su funcionamiento mediante pruebas funcionales, de integración y de rendimiento.

4. Integración con la API corporativa de INDEMIN

La plataforma mantiene consistencia con los sistemas internos de INDEMIN mediante sincronización automática de usuarios, faenas y maquinaria, eliminando duplicidad administrativa y fortaleciendo la gobernanza del sistema.

5. Evaluación de la arquitectura implementada

La validación confirmó que la solución satisface atributos de calidad esenciales:

- Disponibilidad
- Seguridad
- Escalabilidad
- Mantenibilidad
- Trazabilidad

Estos resultados confirman la pertinencia del enfoque arquitectónico adoptado.

5.3 Trabajos Futuros

Las capacidades y la arquitectura del sistema desarrollado sientan una base sólida para su evolución y expansión continua. Con el objetivo de potenciar significativamente el alcance, la inteligencia operativa y la sostenibilidad tecnológica de la solución, siempre y cuando esto se amerite, para ello se propone las siguientes estrategias

A. Ampliación y profundización del módulo analítico

La siguiente fase se centrará en transformar los datos recolectados en inteligencia procesable, trascendiendo la mera descripción para incorporar capacidades predictivas:

- Dashboards Interactivos y Personalizados: Desarrollo de tableros de control avanzados que permitan a los usuarios filtrar, segmentar y explorar dinámicamente los datos de seguridad y riesgo según sus necesidades operacionales (ejemplo, por área, supervisor, tipo de riesgo).
- Análisis de Tendencias Comparativas: Implementación de métricas de comparación cruzada que permitan evaluar el desempeño de seguridad entre faenas, maquinaria específica, turnos u unidades operativas para identificar las mejores prácticas y los focos de riesgo.
- Modelos Predictivos de Comportamiento Riesgoso (Machine Learning): Investigación e implementación de modelos de Machine Learning para predecir la probabilidad de incidentes futuros o identificar patrones de comportamiento riesgoso inminente, basándose en variables históricas y contextuales del sistema PRC.

B. Mejora de la arquitectura y funcionalidades del módulo móvil

La robustez y experiencia de usuario de la aplicación móvil son cruciales para la toma de datos en terreno. Se propone la siguiente optimización arquitectónica

- Migrar a un patrón de estado escalable que permite la gestión de estado avanzado, como utilización de providers, riverpod o bloc.
- Sincronización Incrementada con resolución avanzada de conflictos, permitiendo enviar cambios de manera incremental y aplicando una lógica robusta para la resolución automática de conflictos en la base de datos
- Incorporación de evidencia multimedia y geolocalización

C. Extensión del ecosistema arquitectónico y microservicios

Dado que la aplicación puede llegar a una cantidad mayor de usuarios, para asegurar la flexibilidad e integración empresarial a largo plazo, se plantea la siguiente evolución arquitectónica

- Exposición de nuevos microservicios, creando un desacoplamiento gradual y exposición de nuevas funcionalidades como microservicios independientes, derivados del monolito modular
- Integración con aplicaciones corporativas adicionales, estableciendo pipelines de datos y APIs para la integración con sistemas existentes como un SGO o un RP.



5.4 Reflexión Final

CoreSafe constituye una contribución significativa tanto a la práctica de la ingeniería de software como a la gestión preventiva en entornos industriales y mineros. El proyecto demuestra que una arquitectura cuidadosamente diseñada, sustentada por patrones modernos y orientada a la modularidad, puede transformar procesos críticos, garantizando trazabilidad, eficiencia y disponibilidad de la información.

La arquitectura propuesta no solo resuelve un problema actual, sino que establece un fundamento tecnológico sólido para el crecimiento futuro del sistema, permitiendo que INDEMIN avance hacia una gestión preventiva basada en datos, más inteligente, ágil y sostenible.

6 Referencias

- [1] Campos, D. (2012). Evaluación del Programa de Refuerzo Conductual Psicosocial (Informe Final). Asociación Chilena de Seguridad (ACHS), Departamento de Investigación y Desarrollo. Proyecto FUCYT P0131/2012. Disponible en:
https://www.achs.cl/docs/librariesprovider2/fucyt-proyectos/p0131_2012_dcamos_evaluacion-programa-de-refuerzo-conductual-psicosocial_informe-final_050115.pdf?sfvrsn=95f0cc3e_2
- [2] Bachche, S., Rydström, K., & Bjelkemyr, M. (2023). Digitalization and Safety. KTH Royal Institute of Technology. Disponible en:
<https://kth.diva-portal.org/smash/get/diva2:1825493/FULLTEXT01.pdf>
- [3] Sternad Zabukovšek, S., Jordan, S., & Bobek, S. (2023). *Managing Document Management Systems' Life Cycle in Relation to an Organization's Maturity for Digital Transformation*. *Sustainability*, 15(21), 15212. <https://doi.org/10.3390/su152115212>
- [4] Nivedhaa, N. (2024). *Software Architecture Evolution: Patterns, Trends, and Best Practices*. *International Journal of Computer Sciences and Engineering (IJCSE)*, 1(2), 1–14. Disponible en:
https://www.researchgate.net/publication/384019495_SOFTWARE_ARCHITECTURE_EVOLUTION_PATTERNS_TRENDS_AND_BEST_PRACTICES
- [5] Gandhi, R., Richards, M., & Ford, N. (2024). *Head First Software Architecture: A Learner's Guide to Architectural Thinking*. O'Reilly Media.
- [6] Bass, L., Clements, P., & Kazman, R. (2021). *Software Architecture in Practice* (4th ed.). Addison-Wesley.
- [7] Garzas, J., & Yilmaz, M. (2022). *Software Architecture: Principles, Patterns, and Practices*. Springer.
- [8] Newman, S. (2021). *Building Microservices: Designing Fine-Grained Systems* (2nd ed.). O'Reilly Media.
- [9] Ford, N., Parsons, R., Kua, P., & Sadalage, P. (2024). *Building evolutionary architectures: Automated software governance* (2nd ed.). O'Reilly Media.
- [10] Newman, S. (2020). *Monolith to microservices: Evolutionary patterns to transform your monolith*. O'Reilly Media.