**FORMULACIÓN DEL PROYECTO DE TÍTULO**

**Proyecto de Tesis Cuerpo de Bomberos**

**Asignatura: Proyecto de Título – TIHI84**

**Sección: D-IEI-N8-P1-C2**

**Académico guía: María Magdalena Becerra Meza**

**Integrantes del equipo: Carlos Trincado, Matías Vargas, Nicolas Aguilera**

**Fecha de entrega 24/10/2025**

**Contenido**

[I. Introducción 4](#_heading=h.gpsgojtwtvn4)

[II. Identificación del Problema 4](#_heading=h.bfib4t78e1jx)

[2.1 Actualización y justificación del problema 4](#_heading=h.fxgx31mi8tt0)

[2.1.1 Descripción de la organización. 4](#_heading=h.6jqftii1f9ok)

[2.1.2 Descripción del problema. 4](#_heading=h.xjg3rqujm6sd)

[2.2 Justificación del problema. 4](#_heading=h.1u27emvp46ix)

[2.2.1 Relevancia del problema. 4](#_heading=h.5rnqo4kw7a4k)

[2.2.2 Complejidad del problema. 4](#_heading=h.t66r4svp3or2)

[III. Levantamiento de Requerimientos. 4](#_heading=h.6olu4da1ys9w)

[IV. Marco Teórico. 4](#_heading=h.ovalt8uuva4i)

[V. Objetivos del Proyecto. 4](#_heading=h.ld0wlfygj7wh)

[5.1 Solución tecnológica 4](#_heading=h.ci3ywg8slrte)

[5.1.1 Formulación de la Solución 4](#_heading=h.xj67mfkvl0bw)

[5.1.2 Alcance y restricciones. 4](#_heading=h.u6ryx52wv8bb)

[5.2 Impacto de la solución. 4](#_heading=h.hrqlslrevfe8)

[5.2.1 Proceso de negocio afectado. 4](#_heading=h.vi6k4xfzeais)

[5.2.2 Registro de Interesados. 4](#_heading=h.nfdo0zpjocih)

[5.2.3 Indicadores de gestión. 4](#_heading=h.9utpn3i2f1gr)

[5.2.4 Niveles de servicio. 4](#_heading=h.pvwkdr9cx1bb)

[5.3 Objetivos del proyecto 4](#_heading=h.mtudkrdc2ei2)

[5.3.1 Objetivo General. 4](#_heading=h.ff42tt5v9xhy)

[5.3.2 Objetivo Específico. 4](#_heading=h.b97fqcj4yn3o)

[VI. Metodología de Trabajo 4](#_heading=h.i61hbx3tfjnr)

[6.1 Metodología de Desarrollo de la solución. 4](#_heading=h.wzui8bw81ckb)

[6.2 Duración y cronograma. 4](#_heading=h.pp6e5qro8vqa)

[6.3 Equipo de trabajo. 5](#_heading=h.tatmbh5ukvxo)

[6.4 Plan de recursos.](#_heading=h.cr7fayxbd72r)

[VII. Definición de arquitectura TI](#_heading=h.gvo0bniidd25)

7.1 Visión General

7.2 Componentes y responsabilidades

7.3 Arquitectura de despliegue y disponibilidad

7.4 Integración con VIPER (opciones)

7.5 Modelo de Datos (esquema simplificado)

7.6 Requisitos no funcionales (mapa corto)

7.7 Seguridad y cumplimiento

7.8 Entregables de arquitectura

[VIII. Reconocimiento de arquitectura empresarial](#_heading=h.rh096w9aaz4u)

8.1 Tipo de organización y estructura

8.2 Mapeo de capacidades empresariales (Business Capabilities)

8.3 Procesos de negocio impactados

8.4 Roles y gobernanza TI (recomendación)

8.5 Data governance y políticas

8.6 Impacto organizacional y roadmap de adopción

8.7 Capacitación y cambio organizacional

8.8 Indicadores de éxito (relacionados a negocio)

(de aquí empieza la evaluación 2)

IX. Detalle de las Tecnología a Implementar

9.1 Análisis cualitativo/cuantitativo de las tecnologías que serán implementadas

9.1.1 Indicar dos alternativas de solución

9.1.2 Estudios de Factibilidad

9.1.3 Selección de alternativa (Método de puntuación, incluir todo el desarrollo, los únicos factores costo, rapidez y seguridad)

9.1.4 Flujo de caja alternativa ganadora (se recomienda 5 años max.) Calculo de VAN, TIR, ROI, análisis de los resultados

9.2 Herramientas, aplicaciones, lenguajes y componentes que serán implementados

X. Detalle de la Arquitectura a implementar

10.1 Diagrama BPMN

10.2 Diagrama de Casos de Uso

10.3 Diagrama de componentes

10.4 Modelo de Datos

10.5 Diccionario de Datos

10.6 Topología de comunicaciones

10.7 Diagrama de Infraestructura

10.8 Diagrama de Arquitectura

10.9 Microservicios

XI. Implementación de los KPI y SLA

11.1 Descripción de los KPI

11.2 Descripción de los SLA

XII. Plan de Pruebas y Aseguramiento de Calidad

12.1 Plan de Pruebas

12.1.1 Pruebas de Humo (Smoke Test)

12.1.2 Pruebas Unitarias

12.1.3 Pruebas de Integración

12.1.4 Pruebas Funcionales

12.1.5 Pruebas de Caja Negra

12.1.6 Pruebas de Regresión

12.2 Normas y Estándares

XIII. Plan de Implementación

13.1 Gestión de Disponibilidad

13.2 Gestión de Continuidad

13.3 Plan de Mantención

XIV. Ajustes del Cronograma

14.1 Revisión de Carta Gantt o desarrollo de fases

[XV. Conclusiones: 5](#_heading=h.a1y2856rlqch)

[XVI. Referencias bibliográficas 5](#_heading=h.srhv0pjfizdk)

[XVII.Anexos 6](#_heading=h.qslvmry3elqn)

1. **Introducción**

El presente informe de proyecto de título aborda el diseño y desarrollo de una solución tecnológica orientada a la modernización de los procesos internos del Cuerpo de Bomberos de Rancagua, con especial foco en la gestión digital de guardias nocturnas y la integración con el sistema VIPER para la administración de emergencias. La institución, pese a contar con protocolos avanzados para la atención en terreno, mantiene procesos manuales en áreas críticas como el registro de guardias, lo que genera duplicidad de esfuerzos, falta de trazabilidad y dificultades en la planificación estratégica.

El trabajo realizado se sustenta en la necesidad de fortalecer la eficiencia, disponibilidad y calidad del servicio que entregan los voluntarios, incorporando una plataforma informática que permita registrar asistencia, consolidar estadísticas, generar informes automáticos y entregar indicadores de gestión confiables a la Comandancia. Para ello, se desarrolló un análisis detallado de la problemática, se definieron los requerimientos funcionales y no funcionales del sistema, se formuló un marco teórico basado en bibliografía especializada y normativa vigente, y se establecieron objetivos, metodología de trabajo y plan de recursos.

La propuesta incluye además un marco de arquitectura tecnológica y empresarial, considerando estándares internacionales como ISO 22320 y NFPA 1720, y buenas prácticas de gestión de proyectos como PMBOK y Scrum. Con este enfoque, se busca no solo resolver las limitaciones actuales, sino también sentar las bases para una transformación digital sostenible en la institución, favoreciendo la operatividad, la seguridad y el impacto social que caracteriza al voluntariado bomberil en Chile.

1. **Identificación del Problema**

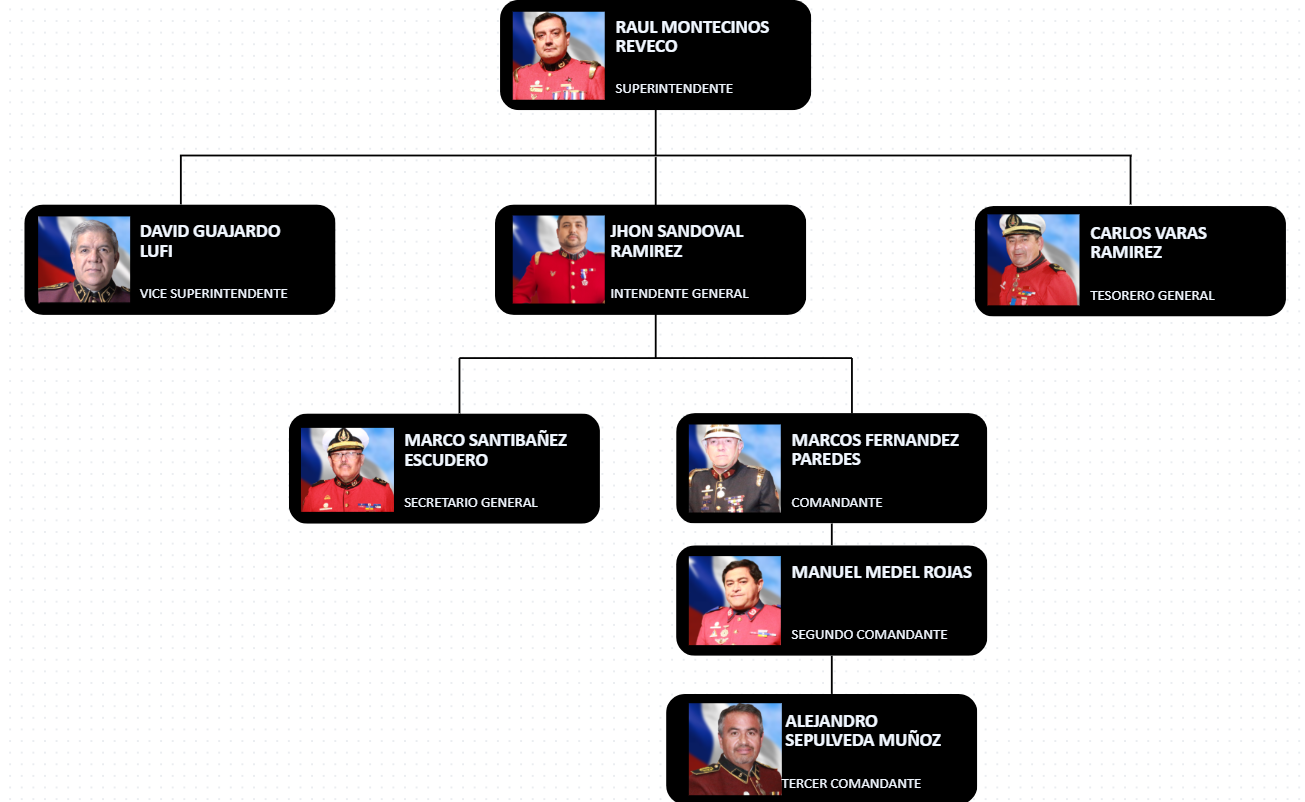
**2.1 Actualización y justificación del problema.**

**2.1.1 Descripción de la organización.**

**Tipo y rubro de la empresa:** Institución voluntaria sin fines de lucro dedicada a la respuesta de emergencias.

**Posicionamiento de mercado:** Es la organización más relevante en rescate y control de incendios de la ciudad de Rancagua. Cuenta con 8 compañías y más de 570 voluntarios activos.

**Estructura organizacional:** La institución se organiza en compañías de bomberos, cada una dirigida por un capitán, con mando general en un Superintendente y Comandancia. A continuación, adjuntamos el organigrama de Bomberos de Rancagua y su Directorio General periodo 2025-2026.



**Misión:** Proteger la vida, bienes y medio ambiente de la comunidad frente a emergencias y desastres, con profesionalismo, constancia y disciplina.

**Visión:** Ser una institución moderna, reconocida por su eficiencia operativa y capacidad de respuesta en emergencias, con un fuerte compromiso social y comunitario.

**Cadena de valor:** Si bien la cadena de valor se orienta tradicionalmente a organizaciones con fines de lucro, esta puede adaptarse a instituciones sin fines de lucro, como el Cuerpo de Bomberos de Rancagua, con el objetivo de maximizar la eficiencia, la calidad del servicio y el impacto social. En este contexto, se identifican las siguientes actividades:

* **Actividades primarias:**

**Logística interna:** recepción, almacenamiento y mantención de equipos (mangueras, uniformes, camiones, cilindros de oxígeno, agua).

**Operaciones:** respuesta directa a emergencias, incluyendo control de incendios, rescates, apoyo en accidentes de tránsito y manejo de emergencias químicas.

**Logística externa:** coordinación con hospitales, policía, municipalidad y comunidad para la derivación de casos y la gestión posterior a la emergencia.

**Marketing y ventas (adaptado):** campañas de sensibilización, captación de nuevos voluntarios, recaudación de fondos y difusión de la importancia del servicio.

**Servicio postventa (adaptado):** capacitación a la comunidad en prevención de riesgos, charlas escolares, revisiones de instalaciones críticas (detectores de humo, extintores, redes húmedas)

* **Actividades de apoyo:**

**Infraestructura de la organización:** gestión administrativa, coordinación con municipios, transparencia financiera y cumplimiento de normativas.

**Gestión de recursos humanos:** reclutamiento, formación y motivación de voluntarios, además del cuidado de su salud física y mental.

**Desarrollo tecnológico:** implementación de software de geolocalización, simuladores de entrenamiento y sistemas de comunicación avanzada en emergencias.

**Adquisiciones**: compra y mantención de equipamiento, gestión de vehículos,

búsqueda de subvenciones y convenios de colaboración.

**Cuadro de mando integral:** Si bien actualmente el Cuerpo de Bomberos de Rancagua no cuenta con un sistema formal de cuadro de mando integral, se propone un esquema básico adaptado a organizaciones sin fines de lucro. Este cuadro busca facilitar la medición de indicadores clave relacionados con la eficiencia operativa, la gestión de recursos y el impacto social. Verán el cuadro propuesto para el CMI adaptado al cuerpo de bomberos de Rancagua, con sus perspectivas, objetivos estratégicos, indicadores y fórmulas de cálculo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Perspectiva | Objetivo Estratégico | Indicador | Fórmula de cálculo | Meta por lograr |
| **Financiera** | Optimizar el uso de recursos y donaciones | Nivel de ejecución presupuestaria | (Gastos ejecutados / Presupuesto asignado) × 100 | Alcanzar un nivel de ejecución del 90% del presupuesto ($500.000) en 6 meses, reduciendo costos operativos en un 15% mediante herramientas open-source. |
| **Clientes / Comunidad** | Mejorar la satisfacción y seguridad ciudadana | Tiempo promedio de respuesta | (Σ Tiempos de respuesta / Nº de emergencias atendidas) | Reducir el tiempo promedio de respuesta en un 20% (de 10 a 8 minutos) tras la implementación, basado en datos VIPER |
|  |  |  |  |  |
| **Procesos Internos** | Asegurar cobertura de guardias nocturnas | Cumplimiento de guardias nocturnas | (Guardias cumplidas / Guardias planificadas) × 100 | Lograr un cumplimiento del 95% de guardias planificadas en el primer año, reduciendo un 30% con el módulo de gestión (EDT 2.3) |
| **Aprendizaje y Crecimiento** | Fortalecer competencias del voluntariado | Nivel de capacitación de voluntarios | (Nº voluntarios capacitados / Nº total voluntarios) × 100 | Alcanzar un 80% de voluntarios capacitados (50/60) en 6 meses, mejorando la satisfacción del usuario en un 25% mediante sesiones y manuales (EDT 3.2) |
|  |  |  |  |  |
|  |  |  |  |  |

**Estándares, certificaciones y leyes:** La institución se rige por un conjunto de normativas nacionales e internacionales que buscan garantizar la seguridad operativa y la calidad del servicio prestado a la comunidad. Entre ellas destacan:

* **Leyes Marco**

**Ley N°18.959,** Personalidad Jurídica de los Cuerpos de Bomberos, complementando el Título XXXII del Código Civil para la obtención de personalidad jurídica, promulgada en el año 1990.

**Ley N°20.564,** Ley Marco de los Bomberos de Chile, promulgada en 2012 y Modificada en el año 2019.

**Ley N°20.907,** Protege la situación laboral de los trabajadores dependientes que son Bomberos voluntarios, promulgada en el año 2016.

* **Normativas Nacionales**

**Reglamentos de Bomberos de Chile**, que establecen procedimientos internos de seguridad, operación y formación de voluntarios, ceremonias, denuncias, protocolos de acoso sexual y estructura organizacional.

**Norma Chilena NCh 2056/2006**, sobre equipos de protección personal para bomberos.

**Norma Chilena NCh 2245/2015**, sobre señalización de seguridad para emergencias.

Protocolos de **SENAPRED** (ex-ONEMI) para gestión de emergencias y desastres.

* **Normas y estándares internacionales:**

**NFPA (National Fire Protection Association, EE.UU.)**, referentes a incendios estructurales, materiales peligrosos, rescate técnico y seguridad en operaciones.

**ISO 9001:2015:** Certificación de Calidad aplicada a procesos administrativos y formativos.

**ISO 14001:2015:** Regula Impacto Ambiental de Actividades Bomberiles como el manejo de equipos y residuos.

**ISO 22320:2018**, Gestión de emergencias – Directrices para la respuesta ante incidentes.

**ISO 45001:2018**, Sistemas de gestión de seguridad y salud en el trabajo.

**Antecedentes de la Organización.**

La Corporación del Cuerpo de Bomberos de Rancagua, fue fundada el 12 de Febrero de 1882, después que una tragedia sin precedente azotó a la tranquila población, un voraz incendio afectó el Molino San Pedro, ubicado al sur de la ciudad, consumiendo todas las reservas de trigo que se almacenaban en el sector, provocando un gran impacto a la comunidad. Los vecinos y autoridades se reunieron con el Gobernador de la época, Sr. Alejandro Morán Castellón, ciudadano español que ejercía el cargo. Posterior a varias reuniones, un 12 de Febrero de 1882, surge la Primera Compañía de Bomberos “Bomba O’Higgins”.

La vocación de servicio ha sido constante en el transcurso de la historia, en el año 1886 fue evidenciado en la pandemia del cólera, donde con sus propios medios, construyeron una ambulancia para los enfermos y fallecidos. El ser voluntario y la autogestión son la base de lo que es la institución hoy en día. Gran parte del día depende de una constante participación del personal para asegurar la cobertura permanente de la ciudad. El correcto funcionamiento de estas guardias, son importantes para dar una respuesta rápida y eficaz, principalmente en los horarios nocturnos de menor actividad.

Hoy, conforma un equipo de más de 570 Bomberos y Bomberas, repartidos en 8 compañías ubicadas estratégicamente en diferentes partes de la Ciudad de Rancagua, que se distinguen por ser completamente voluntarios, dedican su tiempo a la protección de la ciudadanía, bajo el lema “**Constancia y Disciplina”.**

**2.1.2 Descripción del problema.**

El registro de las Guardias de horario Nocturno, llamadas “Guardias Nocturnas”, no mantiene apoyo de herramientas digitales. Los Bomberos que pernoctan en las dependencias del Cuartel de Bomberos, entre las 00:00 AM, hasta las 07:00 AM, deben firmar un libro foliado que confirma su asistencia a dicha guardia, método el cual es funcional, tiene una serie de deficiencias las cuales presentan un impacto negativo en el área operativa. Una de ellas es la gestión. La designación de turnos se realiza de una manera informal, la cual es informada por el Jefe de Guardia, lo que perjudica la planificación a mediano o largo plazo. Bomberos Rancagua no cuenta con una herramienta para que los bomberos consulten su historial de guardia o próximos turnos a asistir.

Al finalizar el turno de Guardia, el jefe de Guardia debe transcribir y dejar registro en el libro, si atendieron alguna emergencia ¿A Qué Hora Fue?, ¿En Qué Calles?, ¿Qué Material Mayor asistió? ¿Alguna Observación en la emergencia sobre el personal o el material?, en caso de que no, solamente se registra si hubo alguna observación con el personal o el material durante el horario estipulado.

La información de las Guardias Nocturnas de los años anteriores se encuentra en físico, lo cual no permite un análisis para decisiones estratégicas enfocadas en el servicio.

El Cuerpo de Bomberos de Rancagua, como institución voluntaria, opera con una infraestructura tecnológica que prioriza la accesibilidad y la eficiencia, utilizando una combinación de soluciones de software y hardware. Actualmente, la infraestructura se basa en redes de comunicación por radio y una red de datos local en los cuarteles principales para la gestión interna. No se dispone de servidores locales robustos para el procesamiento de grandes volúmenes de datos o aplicaciones críticas.

En cuanto a bases de datos y lenguajes de programación, la gestión de datos se realiza en gran medida a través de software de escritorio o plataformas en línea de terceros, lo que genera islas de información. La arquitectura tecnológica actual se caracteriza por ser descentralizada, sin una base de datos única que centralice toda la información operativa.

**Arquitectura de Aplicaciones**

El Cuerpo de Bomberos de Rancagua opera con una infraestructura diseñada para garantizar la coordinación, priorizando la accesibilidad, rapidez y confiabilidad en sus operaciones.

La organización utiliza un conjunto de aplicaciones que le permiten operar eficientemente:

**Comunicaciones Radiales**: El sistema de comunicaciones radiales, constituye el pilar fundamental para la coordinación en terreno. En este sistema, se incluyen tanto radios analógicas VHF/UHF, como radios digitales modernas que operan en frecuencias asignadas por la Subsecretaría de Telecomunicaciones (SUBTEL) de Chile. Estas radios están instaladas en los vehículos de emergencias, en los cuarteles, y gran parte del personal, tiene su propia radio de comunicaciones. Para mayor cobertura, existen repetidores, ubicados estratégicamente para dar cobertura completa a la Comuna de Rancagua.

Las radios, se utilizan para transmitir claves de emergencia (Un ejemplo sería 6-0 B4 44-5, 6-2 del 10-0 “Calle 1 / Pasaje 2”. Esto significa “A cargo del Carro de Bomberos B4, se encuentra el Capitán de la Cuarta Compañía, con 5 bomberos más, la dirección del Incendio Estructural es en Calle 1 / Pasaje 2”, esto facilita poder obtener una comunicación más eficiente en entornos ruidosos. Este sistema no depende de la conectividad a internet, lo que lo hace ideal para desastres naturales o apagones masivos.

**Red de Datos Local:** Cada cuartel cuenta con red local, basada en WIFI y conexiones Ethernet. Estas redes permiten el acceso a herramientas de gestión operativa y administrativa, así como la conectividad para comunicaciones externas.

Esta soporta tareas administrativas y la comunicación con otras instituciones externas. Además, sirve como base para tener acceso al Sistema VIPER desde un computador instalado en los cuarteles, donde se visualizan los datos y se registran las emergencias en tiempo real.

**Sistema VIPER**: (SISTEMA DE INFORMACIÓN PARA LA GESTIÓN DE EMERGENCIAS), plataforma digital para el despacho y registro de emergencias, utilizada ampliamente por los Cuerpos de Bomberos en Chile. Disponible como aplicación móvil y como aplicación de escritorio.

VIPER notifica a los usuarios sobre emergencias en tiempo real, proporcionando información crítica sobre el lugar de la emergencia, que material mayor fue asignado, a qué hora fueron solicitados y un mapa referencial para la navegación.

Aunque VIPER es una solución avanzada para la gestión de emergencias, no está diseñado para administrar las Guardias Nocturnas, ni generar informes combinados de los datos de los Guardianes y las emergencias asistidas. La falta de una API pública documentada, dificulta el integrar con otros sistemas. Como resultado, el Jefe de Guardia de cada Compañía, debe transcribir manualmente los datos de las emergencias de VIPER al libro foliado de Guardias Nocturnas, lo que conlleva a riesgo de errores y la dependencia de conectividad a internet.

**Descripción del problema**: **Ausencia de un Sistema Digital para el Registro de Guardias**

En base a lo anterior, se identifica como problema principal la desconexión entre los sistemas digitales asociados al Cuerpo de Bomberos de Rancagua y los procesos manuales de registro interno. Aunque la institución cuenta con herramientas avanzadas como el sistema VIPER para el despacho y registro de emergencias, la gestión de la Guardia Nocturna sigue dependiendo de una metodología completamente manual. Esta falta de integración genera una serie de limitaciones operativas y estratégicas, agravadas por una infraestructura tecnológica que, si bien incluye comunicaciones radiales analógicas y digitales (VHF/UHF y DMR para mayor claridad y transmisión de datos básicos), redes de datos locales descentralizadas y VIPER conectado a computadores en los cuarteles, no soporta la automatización ni la trazabilidad digital de procesos clave como el registro de guardias.

La información de cada emergencia registrada en VIPER (material mayor despachado, hora de salida, hora de llegada, etc.) debe ser transcrita manualmente al libro de Guardia Nocturna, un proceso propenso a errores en la información registrada y que consume tiempo valioso del Jefe de Guardia. Además, esta desconexión impide una administración correcta de los materiales utilizados en las emergencias, ya que no se puede rastrear de manera eficiente si estos recursos (como mangueras, cilindros de oxígeno o uniformes) han sido dañados o requieren mantenimiento, lo que afecta su disponibilidad y operatividad para una próxima emergencia. La información de las emergencias, al estar en un sistema aislado del registro de guardias, impide un análisis por parte del mando operativo. No es posible para el sector operativo vincular los datos de cumplimiento de los turnos de bomberos con las emergencias para detectar patrones de incidentes, zonas más recurrentes u horarios específicos. La información existe, pero no se puede analizar unificadamente, limitando la optimización de recursos y la planificación estratégica.

De igual manera, el manejo manual de los registros hace que la planificación equitativa de turnos sea mucho más compleja para los voluntarios. La ausencia de un sistema impide medir el cumplimiento de los turnos y diseñar estrategias de capacitación basada en datos reales, exacerbado por la falta de integración con VIPER, que podría proporcionar datos en tiempo real desde los computadores de los cuarteles, pero, no lo hace de forma automatizada.

En ciertas ocasiones, la Guardia Nocturna, debe declararse fuera de servicio por la falta de un Conductor Capacitado, o por insuficiencia de personal para cumplir con los protocolos operativos, o decisiones del mando operativo basadas en necesidades estratégicas.

En este contexto, el presente proyecto propone el desarrollo de un sistema integral digital para trabajar la información disponible de VIPER y, a su vez, modernizar y optimizar la gestión eficiente de la Guardia Nocturna, dando un registro automático y seguro de asistencia a las guardias, asimismo, permita registrar y notificar automáticamente cuando una Guardia Nocturna quede fuera de servicio. A su vez, que proporcione gestión eficiente de turnos y vinculación directa con las emergencias atendidas en cada turno asistido. Adicionalmente, añadirá herramientas de análisis estadístico y generación automática de informes, facilitando la comunicación entre el jefe de guardia y el mando operativo con respecto a la información clara y confiable. Al resolver la problemática planteada, optimiza planificación, incrementa capacidad de respuesta, y fortalece gestión operativa, incluyendo una mejor administración de materiales para garantizar su operatividad en emergencias futuras.

**2.2 Justificación del problema.**

La problemática planteada no solo se limita a un registro manual ineficiente, sino que también afecta directamente a la capacidad operativa de la institución. La ausencia de un sistema integrado que conecte los procesos manuales con herramientas como VIPER, genera una serie de limitaciones que afectan a los voluntarios directamente. Esta carencia, combinada con la falta de trazabilidad de los materiales utilizados en las emergencias, compromete la eficiencia del servicio y la preparación para próximas emergencias.

**2.2.1 Relevancia del problema.**

La Corporación enfrenta una serie de desafíos que limitan la eficiencia operativa y la gestión de su voluntariado. Abordar estos aspectos resulta fundamental para garantizar un servicio oportuno, confiable y sostenible en el tiempo.

La falta de un sistema digital integrado para el Cuerpo de Bomberos de Rancagua es altamente relevante, ya que impacta directamente en la operatividad del servicio, la seguridad de la comunidad y, lo más importante, en los propios voluntarios.

La digitalización del registro de guardias y emergencias permitirá analizar patrones de incidentes (horarios, ubicaciones y tipo de emergencia). Este análisis es crucial para que el mando operativo pueda planificar capacitaciones focalizadas y fortalecer la respuesta ante los eventos más comunes. Por ejemplo, si se detecta un patrón de emergencias vehiculares en una zona específica durante las noches de fin de semana, se podría programar un curso de extricación para la compañía que cubre ese sector, enfocado en las emergencias que atienden con mayor frecuencia para optimizar la capacidad operativa durante las emergencias.

El sistema proporcionará una trazabilidad clara y precisa del compromiso de cada bombero. Al registrar automáticamente la cantidad de guardias y emergencias atendidas por cada voluntario, se podrá reconocer y valorar el esfuerzo y la dedicación de los bomberos y bomberas. Esto no solo servirá para la gestión interna, sino que también reforzará la motivación y el sentido de pertenencia de ser un Bombero del Cuerpo de Bomberos de Rancagua.

La solución propuesta modernizará un proceso que hoy es obsoleto y propenso a errores. Al eliminar la duplicidad de la carga de datos —que ya existe en el sistema VIPER— y automatizar la generación de informes, se liberará tiempo valioso para los jefes de guardia. Este tiempo podrá ser reasignado a tareas más críticas y de mayor valor operativo, como la supervisión directa del personal y la revisión de equipamiento, mejorando la eficiencia y la seguridad del servicio específicamente en el horario nocturno.

**2.2.2 Complejidad del problema.**

La implementación de un sistema digital para el registro de Guardias Nocturnas enfrenta múltiples desafíos técnicos, operativos y organizacionales. La solución propuesta, debe superar las limitaciones de la tecnología actual, que incluye comunicaciones radiales, redes de datos, y el Sistema VIPER.

**Arquitectura Tecnológica:** Solución escalable, y accesible desde múltiples dispositivos, capaz de operar en un entorno donde la conectividad pueda ser inestable. Una arquitectura nube (Cloud-Based), es la opción más adecuada.

Sin embargo, dado que algunos cuarteles pueden presentar interrupciones en la conectividad, el sistema debe incluir un modo offline que permite registrar datos localmente como asistencias, emergencias, estados de material o inactividad de guardias y sincronizarlos una vez vuelva la conexión. Además, debe soportar un volumen creciente de datos, considerando que, en el Cuerpo de Bomberos de Rancagua, se gestionan múltiples compañías y un número variable de voluntarios. Por ende, se requiere una infraestructura optimizada, con bases de datos que manejen eficientemente grandes volúmenes de emergencias históricas.

**Integración de Sistemas:** El principal desafío técnico es la integración con el sistema VIPER. Esto requiere el desarrollo de una solución que pueda leer y procesar la información generada por VIPER para evitar que el jefe de guardia deba transcribir manualmente los datos de las emergencias. Si no es posible la integración directa a través de una API, se deberá idear un mecanismo de importación de datos que sea robusto y confiable.

**Seguridad:** El sistema maneja información sensible (RUT de voluntarios, detalles de emergencias, estados de materiales). Los desafíos incluyen:

* Autenticación y control de acceso: Implementar un sistema basado en OAuth 2.0 o JWT con roles diferenciados (bombero, Jefe de Guardia, administrador) para limitar el acceso según funciones.
* Protección de datos: Cumplir con la Ley N° 19.628 mediante cifrado (TLS/SSL en tránsito, AES-256 en reposo) y auditorías de acceso.
* Resiliencia ante ciberataques: Proteger contra amenazas como DDoS o inyecciones SQL con firewalls y monitoreo continuo.

**Desarrollo de una Aplicación de Escritorio y Móvil:** La solución deberá ser compatible con diferentes dispositivos, incluyendo computadoras de escritorio y dispositivos móviles. Esto implica el desafío de diseñar una interfaz de usuario que sea intuitiva y funcional tanto en pantallas grandes como pequeñas.

Por ende, para una interfaz multiplataforma, usar frameworks como React Active o Flutter y el diseño de una interfaz intuitiva con flujos claros y garantizar la compatibilidad con computadores antiguos o dispositivos móviles variados.

A continuación, se hace un cuadro resumen de los Desafíos Principales:

|  |  |
| --- | --- |
| Arquitectura tecnológica | Escalabilidad, modo offline, costos de la nube, compatibilidad con hardware limitado. |
| Integración de Sistemas | Integración con Viper, manejo de datos |
| Seguridad | Autenticación por roles, cifrado de datos. |
| Desarrollo de Aplicaciones | Usabilidad, soporte, Interfaz Multiplataforma |
| Gestión de Materiales | Alertas, Trazabilidad, Integración Inventario Existente |
| Registro de Inactividad | Notificaciones, Reportes, Formulario específico |
| Capacitación | Soporte técnico continuo, adopción gradual |
| Resiliencia Operativa | Modo Offline, Compatibilidad con lo existente |
| Cumplimiento Normativo | Alineación con la Normativa |

1. **Levantamiento de Requerimientos.**

Para el levantamiento de requerimientos, se propone una metodología mixta que integra instrumentos cualitativos (para explorar experiencias y desafíos profundos) y cuantitativos (para medir frecuencias y percepciones a mayor escala). Esta combinación asegura una visión holística, adaptable al entorno voluntario y operativo de la institución, donde los usuarios tienen niveles variados de experiencia tecnológica. Los instrumentos seleccionados se aplicarán en fases secuenciales: primero cualitativos para identificar problemas clave, luego cuantitativos para validar y cuantificarlos. Se estima que esta fase dure aproximadamente 2-3 semanas, involucrando a al menos 20-30 voluntarios para una representación adecuada de las 8 compañías.

**Entrevistas**: Se realizarán entrevistas individuales o en grupos focales con los principales actores: el Jefe de Guardia (al menos 3-5 por compañía), un grupo representativo de bomberos voluntarios que participan en la Guardia Nocturna (10-15 personas, seleccionadas por antigüedad y frecuencia de participación) El objetivo es profundizar en los procesos actuales, identificar tareas diarias, desafíos específicos (como la transcripción manual de datos de VIPER o la gestión de inactividad de guardias) y expectativas sobre la solución digital. Se preparará una guía de preguntas abierta, explorando temas como:

1. Flujo de trabajo actual durante la Guardia Nocturna, incluyendo el registro en el libro foliado y la gestión de situaciones de inactividad (por falta de maquinista, insuficiencia de personal o decisión del mando).
2. Interacción con el sistema VIPER (duplicidad de datos, integración deseada) y la infraestructura tecnológica existente (radios digitales, redes locales).
3. Funcionalidades deseadas, como notificaciones automáticas, trazabilidad de materiales post-emergencia y visualización de historiales de guardias y emergencias.
4. Expectativas sobre la usabilidad, seguridad y accesibilidad móvil/escritorio.
5. Las entrevistas se grabarán (con consentimiento) y analizarán mediante análisis temático para identificar patrones recurrentes.

**Análisis de Documentos y Observación Directa**: Se examinarán documentos clave, como los libros foliados de registro de guardias de los últimos 6-12 meses, reportes de VIPER, reglamentos internos y protocolos operativos, para entender la estructura de los datos (por ejemplo, terminología de claves de emergencia, frecuencia de registros y causas comunes de inactividad). Paralelamente, se realizará observación directa no participativa en la sala de guardia durante al menos 2-3 turnos nocturnos, para documentar el proceso manual en acción, identificar puntos de fricción (como demoras en la transcripción o manejo de emergencias sin maquinista) y oportunidades de mejora no evidentes en entrevistas. Se utilizará un protocolo de observación estandarizado para registrar tiempos, interacciones y problemas operativos.

**Encuesta**: Se implementará una encuesta digital breve (usando herramientas gratuitas como Google Forms o Microsoft Forms) dirigida a un número mayor de bomberos (al menos 50-100 voluntarios, distribuidos por compañías). Esto permitirá recopilar datos cuantitativos sobre la frecuencia de participación en guardias, el uso de tecnología móvil (por ejemplo, apps como VIPER), la percepción sobre la digitalización de procesos y desafíos específicos (como la gestión de inactividad de guardias o trazabilidad de materiales). Las preguntas incluirán escalas Likert (para medir satisfacción con procesos actuales), opciones múltiples (para causas de inactividad) y abiertas (para sugerencias). Los datos se analizaron con estadísticas descriptivas (frecuencias, promedios) para priorizar requerimientos.

Adicionalmente, se incorporará una validación iterativa: los requerimientos preliminares se presentarán en una reunión de feedback con los actores clave para refinarlos, asegurando alineación con las normativas (como la Ley N° 20.564 y ISO 22320:2018) y el contexto voluntario.

**REQUERIMIENTOS FUNCIONALES**

|  |  |  |
| --- | --- | --- |
| NOMBRE REQUERIMIENTO | DESCRIPCIÓN | PRIORIDAD  ALTA, MEDIA, BAJA |
| Login de Usuario | El sistema debe permitir a los bomberos y administradores iniciar sesión con su RUT y una clave única, con opción de recuperación de contraseña. | Alta |
| Gestión de Roles | El sistema debe diferenciar entre roles de "Bombero" y "Jefe de Guardia y Administrativo" con permisos específicos. | Alta |
| Registro de Guardia | Un bombero debe poder registrar su inicio de guardia al ingresar al cuartel, vía RUT o QR | Alta |
| Historial de Guardias | Cada bombero debe poder visualizar un historial de las guardias nocturnas que ha cumplido, fechas, emergencias y observaciones | Media |
| Asignación de Turnos | El administrador debe poder asignar turnos de guardia a los bomberos para una fecha específica, con verificación mínima de personal (3 Voluntarios) | Media |
| Notificaciones de Turno | El sistema debe enviar notificaciones para recordar a los bomberos sus turnos próximos y cambios en asignaciones. | Baja |
| Registro de Emergencia | El jefe de guardia debe poder registrar una emergencia con campos para: fecha, hora, lugar, tipo de emergencia y un campo de observaciones y material utilizado. | Alta |
| Datos del Carro | El jefe de guardia debe poder seleccionar el material mayor (carro de bomberos) despachado, incluyendo su estado post-emergencia | Alta |
| Sincronización con VIPER | El sistema debe tener la capacidad de leer los datos de las emergencias desde el sistema VIPER para completar automáticamente el registro, evitando transcripciones manuales | Alta |
| Cierre de Guardia | Al final del turno, el jefe de guardia debe poder "cerrar" el registro del día, con validación de datos completos y opción para registrar inactividad si aplica. | Alta |
| Informe de Stock de Equipamiento | El sistema debe generar un informe actualizado del stock de equipamiento e insumos críticos (mangueras, cilindros de oxígeno, uniformes, herramientas), indicando cantidades disponibles, en uso y en mantención, con relación a emergencias atendidas | Media |
| Generación de Informe | El sistema debe generar un informe con la información de la guardia y las emergencias atendidas y además uno estadístico mensual. Los Informes serian:  **Informe de guardia nocturna:** detalle de voluntarios asistentes, turnos cumplidos y observaciones registradas e inactividades.  **Informe de emergencias atendidas:** resumen por fecha, tipo de emergencia (ej. incendio, rescate, materiales peligrosos, tiempo de respuesta y recursos movilizados.  **Informe estadístico mensual:** consolidación de datos de guardias y emergencias para análisis de patrones y planificación. | Alta |
| Exportación de Datos | El informe debe poder ser exportado a formatos PDF y Excel, con opciones de filtrado por compañía. | Media |
| Dashboard | El sistema debe mostrar gráficos y estadísticas sobre guardias, emergencias y zonas de recurrencia, usos de materiales | Media |

**REQUERIMIENTOS NO FUNCIONALES**

|  |  |  |
| --- | --- | --- |
| NOMBRE REQUERIMIENTO | DESCRIPCIÓN | PRIORIDAD  ALTA, MEDIA, BAJA |
| Acceso Móvil | La interfaz de usuario debe ser "responsive" y usable en smartphones y tablets con navegación intuitiva | Alta |
| Autenticación Segura | El sistema debe utilizar métodos de autenticación seguros para proteger las cuentas, con logs de acceso | Alta |
| Niveles de Permisos | El acceso a los módulos (ej. reporte de emergencias, estadísticas, inactividad guardia) debe estar restringido según el rol del usuario, cumpliendo con la ley N°19.628 | Alta |
| Disponibilidad | El sistema debe estar disponible un 99.9% del tiempo para garantizar su uso en cualquier momento, con redundancia en la nube para minimizar downtime | Alta |
| Velocidad de Carga | Los módulos y reportes deben cargar en un tiempo máximo de 3 segundos para una buena experiencia de usuario, optimizando consulta de base de datos. | Media |
| Escalabilidad de Datos | La base de datos debe poder crecer para almacenar un historial de varios años de registros, con soporte para miles de registros | Alta |
| Modo Offline | Debe permitir opciones básicas (registro de guardias, emergencias, inactividad) en modo offline, con sincronización automática al reconectar. | Alta |
| Usabilidad | Intuitiva para usuarios no técnicos, con pruebas de usabilidad y diseño accesible | Media |
| Integración Segura | Encriptación de datos transferidos y validación de seguridad | Alta |
| Cumplimiento Legal | El sistema debe cumplir con normativas como Ley N° 19.628 (protección de datos) y Ley N° 20.564, asegurando privacidad y auditoría de registros. | Alta |

1. **Marco Teórico.**

*Investigación Bibliográfica:*

La transformación digital ha impactado significativamente a las instituciones de primera respuesta en todo el mundo, permitiendo optimizar tiempos de reacción, mejorar la trazabilidad de los incidentes y fortalecer la toma de decisiones estratégicas. Según Pressman (2014), el uso de tecnologías de software aplicadas en procesos críticos permite reducir errores humanos y aumentar la eficiencia operativa. En la misma línea, Sommerville (2011) sostiene que la digitalización contribuye a estandarizar los procesos mediante sistemas de información que recopilan y analizan datos en tiempo real, lo cual permite una respuesta más eficiente y coordinada en torno a la calidad.

En Chile, los cuerpos de bomberos voluntarios han comenzado a incorporar plataformas como **VIPER**, que notifican emergencias y entregan información de despacho en línea. VIPER, desarrollado por una scale-up chilena, integra IA y notificaciones masivas, reduciendo el tiempo de despacho a menos de 40 segundos y cubriendo más del 75% de la población Chilena con 96 Cuerpos de Bomberos usuarios (Diario Financiero, 2025) Sin embargo, procesos internos críticos, como el registro de guardias nocturnas, la trazabilidad de materiales utilizados en emergencias y la gestión de situaciones de inactividad (por falta de personal, maquinista o decisión del Mando Operativo), siguen dependiendo de métodos manuales, lo que genera ineficiencias operativas y limita la capacidad de análisis estratégico. La Junta Nacional de Bomberos de Chile (s.f.) ha impulsado iniciativas como el sistema digital tele comunicacional P25, implementado en 2021 para mejorar la redundancia en comunicaciones durante emergencias, destacando la necesidad de integrar herramientas digitales para fortalecer la resiliencia operativa.

Un aspecto relevante en la gestión de emergencias es la **clasificación mediante claves radiales**, las cuales permiten estandarizar la comunicación y mejorar la coordinación entre compañías. Según Documentos del Cuerpo de Bomberos de Rancagua (2025), las principales claves son:

|  |  |
| --- | --- |
| **Clave 10-0:** Incendio estructural. | **Clave 10-9**: Otros Servicios que no son una Emergencia |
| **Clave 10-1**: Incendio de Vehículos motorizados. | **Clave 10-10**: Rebrote de Incendio |
| **Clave 10-2:** Incendio de Pastizales. | **Clave 10-11**: Apoyo a BAVE por Emergencia Aérea. |
| **Clave 10**-**3**: Rescate de Personas. | **Clave 10-12**: Apoyo a Otro Cuerpo de Bomberos |
| **Clave 10-4:** Accidentes Vehiculares. | **Clave 10-13**: Atentado Terrorista o Artefacto Explosivo. |
| **Clave 10-**5: Emergencia con Materiales Peligrosos. | **Clave 10-14**: Caída de Vehículo Aéreo. |
| **Clave 10-6**: Emergencias con Gases Combustibles | **Clave 10-15**: Simulacro |
| **Clave 10-7**: Accidentes Eléctricos |  |
| **Clave 10-8**: Incendios en Estructuras de Material Ligero |  |

En muchos casos, una misma emergencia puede incluir múltiples claves debido a su complejidad, lo que requiere un despliegue más preparado de recursos. Un ejemplo de esto sería un 10-0(Incendio Estructural), en una distribuidora de Gas 10-6(Presencia de Gas Combustible), con dos personas atrapadas al interior (10-3), esto debe facilitar el despliegue de recursos tales como un Carro Porta escalas, una unidad de Rescate, y una unidad de Materiales Peligrosos, además de los Carros de Agua, optimizando la logística.

La inclusión de estas claves en un sistema digital facilita la generación de estadísticas por tipo de emergencia, zona y horario, lo que constituye un insumo fundamental para la planificación de recursos y la capacitación del personal. De acuerdo con el PMBOK® Guide (Project Management Institute [PMI], 2021), el uso de indicadores medibles y datos históricos es esencial para gestionar proyectos de mejora continua en organizaciones de servicio. A si mismo la arquitectura de software se define como “la estructura de los sistemas, que comprende componentes de software, sus propiedades externas visibles y las relaciones entre ellos” (Bass, Clements & Kazman, 2013). Y esto en sistemas críticos como los de respuesta a emergencias, se privilegia el uso de arquitecturas escalables y resilientes, que garanticen disponibilidad y continuidad operativa.

El NIST (2011) establece que los modelos de cloud computing aportan flexibilidad y reducción de costos, ofreciendo servicios bajo esquemas como Infraestructura como Servicio (IaaS) o Plataforma como Servicio (PaaS), ideales para organizaciones que requieren alta disponibilidad sin contar con infraestructura propia.

Asimismo, arquitecturas basadas en microservicios han demostrado ser más adaptables frente a la necesidad de desplegar funcionalidades de forma modular (Newman, 2015). Para el caso de Bomberos de Rancagua, el uso de un sistema web y móvil soportado en la nube permitirá gestionar turnos, emergencias y recursos con mayor eficiencia.

Las metodologías de desarrollo pueden clasificarse en predictivas (tradicionales) y adaptativas (ágiles). Las primeras, como el modelo en cascada, definen un ciclo de vida rígido y documentado, mientras que las segundas buscan responder a la incertidumbre con entregas incrementales (Sommerville, 2011).

Dado el carácter cambiante de los procesos de emergencia, el proyecto se sustenta en el marco ágil Scrum, definido por Schwaber y Sutherland (2020) como un proceso iterativo basado en la planificación de *sprints*, la priorización del *backlog* y la retroalimentación continua. Este enfoque resulta idóneo porque permite incorporar cambios en los requerimientos según la retroalimentación de los usuarios (bomberos y jefes de guardia) y garantiza pruebas en cada ciclo de desarrollo.

Los sistemas de información gerencial tienen como objetivo apoyar la toma de decisiones a partir de la recopilación, procesamiento y análisis de datos (Laudon & Laudon, 2020). En el ámbito de la primera respuesta, estos sistemas permiten mejorar la coordinación, reducir el tiempo de reacción y aumentar la seguridad tanto de los equipos operativos como de la comunidad.

Turban et al. (2018) destacan que la integración de datos históricos con analítica predictiva permite anticipar incidentes recurrentes y optimizar la asignación de recursos. En este sentido, un sistema que centralice guardias, emergencias, materiales y estadísticas fortalece la planificación estratégica de la organización.

La adopción de estándares internacionales asegura que las operaciones se realicen bajo parámetros reconocidos a nivel mundial. Entre los más relevantes se encuentran:

* **ISO 22320:2018**, que establece lineamientos para la gestión de emergencias e incidentes.
* **NFPA 1720**, que regula la organización y despliegue de servicios de bomberos voluntarios.
* **Ley N° 20.564 (Chile, 2012)**, que exige transparencia y eficiencia en la gestión de los recursos humanos en instituciones de bomberos.

Estos marcos normativos refuerzan la necesidad de contar con un sistema digital que garantice trazabilidad, seguridad y disponibilidad en la gestión de emergencias.

1. **Objetivos del Proyecto.**

**5.1 Solución tecnológica**

**5.1.1 Formulación de la Solución**

Desarrollar una plataforma web y móvil basada en la nube que:

* Automatice el Registro de Guardias Nocturnas, eliminando la transcripción manual y permitiendo registrar de manera digital los turnos.
* Integre datos desde el Sistema VIPER, (hora, ubicación, tipo emergencia, material mayor movilizado, para reducir errores y duplicidad).
* Gestionar trazabilidad de materiales utilizados en la emergencia, registrando si está disponible, en mantenimiento o dañado y generando alertas automáticas.
* Registrar y notificar situaciones de inactividad de la Guardia Nocturna, permitiendo documentar causas como falta de personal o maquinista, con notificación directa al mando operativo y la Central de Bomberos.
* Informes Analíticos y estadísticos, proporcionando herramientas para la planificación, capacitación de voluntarios y rendición de cuentas, con grafico de frecuencia a emergencias, patrones de incidentes.

La plataforma tendrá soporte modo offline y sincronización automática, garantizando accesibilidad desde computadores y móviles.

**5.1.2 Alcance y restricciones.**

Incluye: El sistema estará orientado a la gestión digital de guardias nocturnas y emergencias atendidas, con integración parcial al sistema VIPER. Se incluyen los siguientes módulos principales:

**Registro digital de guardias:** Control automático de asistencia mediante RUT o QR, Horario inicio y cierre de turnos. Gestión de turnos asignados, con verificación de cumplimiento mínimo del personal (3). Visualización de historial de guardias por voluntario, incluyendo emergencias atendidas

**Sincronización con VIPER:** importación de datos (Hora, Ubicación, Clave Radial, Material Mayor movilizado, vía API o formatos compatibles (CSV/JSON).

**Gestión Inactividad Guardias Nocturnas:** Formulario para registrar la inactividad del turno, argumentando la causa (insuficiencia personal, falta maquinista, decisión operativa). Notificaciones automáticas al mando operativo y a la Central de Bomberos y por último, la integración con los registros de VIPER para documentar como un evento operativo.

**Trazabilidad de Materiales:** Registro y uso de insumos críticos (Mangueras, Oxigeno, Tanques de Aire, Uniformes, Herramientas, etc). Tras cada emergencia. Alertas automáticas para mantenimiento o reposición de materiales. Correlación de datos materiales con emergencias atendidas en análisis operativo.

**Generación de informes:**

* Informe de guardia nocturna: Voluntarios asistentes, turnos cumplidos, observaciones del jefe de guardia o inactividades registradas.
* Informe de emergencias atendidas: Resumen por fecha, tipo emergencia, tiempos de respuesta y recursos movilizados.
* Informe estadístico mensual: consolidado de datos históricos de guardias y emergencias, con gráficos comparativos.
* Informe de stock de equipamiento (básico): estado de insumos críticos y material mayor, con cantidad en uso y mantenimiento.

**Estadísticas operacionales:**

* Dashboard con gráficos interactivos de frecuencia de emergencias por lugar, tipo, horario, recursos.
* Cumplimiento de turnos y patrones de incidentes, indicadores de asistencias por voluntario.
* Análisis de inactividades de guardias para identificar causas frecuentes

**EDT (Estructura de Desglose de Trabajo):**

La Estructura de Desglose de Trabajo (EDT) organiza las actividades necesarias para el desarrollo, implementación y soporte de la plataforma digital para la gestión de guardias nocturnas, emergencias, materiales e inactividades en el Cuerpo de Bomberos de Rancagua. La EDT se estructura en cuatro fases principales: análisis y diseño, desarrollo, implementación, y documentación y soporte, desglosadas en subfases, paquetes de trabajo y actividades específicas. Cada fase está diseñada para cumplir con los objetivos del proyecto, incluyendo la automatización del registro de guardias, la integración con el sistema VIPER, la trazabilidad de materiales y la gestión de inactividades de la Guardia Nocturna, considerando las restricciones presupuestarias, tecnológicas, humanas y legales. El diagrama jerárquico de la EDT, que detalla todos los niveles, se presenta en el Anexo 2: Diagrama EDT, generado en draw.io para una representación visual clara de las tareas y su organización.

**1. Análisis y diseño del sistema**

|  |  |  |
| --- | --- | --- |
| Descripción: Fase inicial para analizar los procesos actuales, definir requerimientos técnicos y diseñar la arquitectura del sistema. | Deliverables: Documento de requerimientos, diseño de base de datos, prototipos de interfaz, especificación de integración. | Esfuerzo estimado: Alto. |

**1.1 Análisis de procesos actuales**

* Levantar información sobre procesos manuales de guardias, emergencias, materiales e inactividad.
* Actividades: Entrevistas con Jefes de Guardia y voluntarios, observación de turnos nocturnos, análisis de libros foliados, documentación de flujos de trabajo.
* Deliverables: Mapa de procesos, informe de hallazgos.

**1.2 Diseño de base de datos**

* Diseñar una base de datos escalable (por ejemplo, PostgreSQL) para almacenar datos de guardias, emergencias, materiales e inactividades
* Actividades: Definición de entidades, diseño de relaciones, optimización de consultas, configuración de modo offline.
* Deliverables: Diagrama entidad-relación, esquema de base de datos.

**1.3 Diseño de interfaces web y móvil**

* Crear prototipos responsivos y accesibles para usuarios no técnicos.
* Actividades: Diseño de interfaces para login, guardias, emergencias, inactividad, trazabilidad de materiales y dashboard con gráficos
* Deliverables: Prototipos en Figma, guía de estilo UI/UX.

**1.4 Definición de integración con VIPER**

* Especificar el mecanismo de integración con VIPER (API o importación manual).
* Actividades: Análisis de API, diseño de conectores, protocolos de importación (CSV/JSON), validación de datos
* Deliverables: Documento técnico de integración, scripts de importación.

**2. Desarrollo**

|  |  |  |
| --- | --- | --- |
| Descripción: Implementar los módulos funcionales utilizando herramientas open-source para minimizar costos | Deliverables: Código fuente, módulos funcionales, documentación de desarrollo. | Esfuerzo estimado: Alto. |

**2.1 Módulo de registro de guardias**

* Desarrollar funcionalidades para login, registro de asistencia y gestión de turnos.
* Actividades: Implementar login con RUT (JWT), registro de asistencia (QR/formulario), asignación de turnos con verificación de personal mínimo, historial de guardiass
* Deliverables: Módulo funcional, scripts de autenticación.

**2.2 Módulo de sincronización con VIPER**

* Implementar la integración para importar datos de emergencias.
* Actividades: Consumo de API de VIPER (si disponible), importación de archivos CSV/JSON, sincronización automática/manual, validación de datos.
* Deliverables: Conector API, logs de sincronización.

**2.3 Módulo de gestión de inactividad**

* Desarrollar funcionalidad para registrar y notificar inactividad de la Guardia Nocturna.
* Actividades: Crear formulario para causas (falta de maquinista, personal insuficiente, decisión del mando), implementar notificaciones push/email, integrar con VIPER, generar reportes de inactividad.
* Deliverables: Formulario, sistema de notificaciones, reportes.

**2.4 Módulo de trazabilidad de materiales**

* Implementar registro y seguimiento de materiales utilizados en emergencias
* Actividades: Registrar uso y estado de materiales, programar alertas automáticas, correlacionar con emergencias, crear interfaz de consulta de inventario.
* Deliverables: Módulo de inventario, alertas automáticas.

**2.5 Módulo de informes y estadísticas**

* Desarrollar herramientas para generar informes y visualizaciones.
* Actividades: Implementar informes de guardia, emergencias, estadísticas mensuales y stock; diseñar dashboard con gráficos (Chart.js).
* Deliverables: Plantillas de informes, dashboard interactivo.

**3. Implementación**

|  |  |  |
| --- | --- | --- |
| Descripción: Poner en marcha el sistema, realizar pruebas y capacitar a los usuarios. | Deliverables: Sistema operativo, reportes de pruebas, materiales de capacitación. | Esfuerzo estimado: Alto. |

**3.1 Pruebas unitarias y de integración**

* Validar módulos individuales y su integración.
* Actividades: Pruebas de módulos, integración con VIPER, pruebas de seguridad (Ley N° 19.628), simulación de escenarios reales.
* Deliverables: Reportes de pruebas, corrección de errores.

**3.2 Capacitación de usuarios**

* Enseñar a voluntarios y Jefes de Guardia el uso del sistema.
* Actividades: Desarrollar manuales, videos tutoriales, sesiones presenciales, establecer canal de soporte.
* Deliverables: Manuales, videos, plan de capacitación.

**3.3 Puesta en marcha piloto**

* Implementar el sistema en una compañía piloto.
* Actividades: Configurar infraestructura en la nube, instalar aplicación móvil, ejecutar piloto (1 mes), recopilar feedback.
* Deliverables: Sistema operativo en piloto, informe de resultados.

**3.4 Feedback y ajustes**

* Iterar sobre el sistema basado en retroalimentación del piloto.
* Actividades: Analizar feedback, implementar mejoras, realizar pruebas adicionales, preparar despliegue general.
* Deliverables: Informe de feedback, sistema actualizado.

**4. Documentación y soporte**

|  |  |  |
| --- | --- | --- |
| Descripción: Generar documentación técnica y de usuario, y establecer soporte continuo. | Deliverables: Manuales técnicos y de usuario, plan de soporte. | Esfuerzo estimado: Medio. |

**4.1 Documentación técnica**

* Documentar la arquitectura y configuración del sistema.
* Actividades: Documentar instalación, diagramas de arquitectura, protocolos de integración, procedimientos de mantenimiento.
* Deliverables: Manual técnico, diagramas.

**4.2 Documentación de usuario**

* Proveer guías simplificadas para usuarios no técnicos.
* Actividades: Crear guías para guardias, inactividad, emergencias, materiales, informes; generar formato digital (PDF).
* Deliverables: Guías de usuario.

**4.3 Soporte continuo**

* Establecer un equipo de soporte para resolver problemas operativos.
* Actividades: Designar voluntarios técnicos, crear canal de soporte (email/WhatsApp), monitorear problemas, planificar actualizaciones.
* Deliverables: Plan de soporte, canal de comunicación.

**Excluye**: El proyecto, enfrenta varias restricciones que limitan su alcance y exigen estrategias específicas para su mitigación. Estas restricciones detalladas a continuación, abarcan aspectos presupuestarios, tecnológicos, humanos, legales, operacionales, y han sido consideradas en la planificación de la Estructura de desglose de Trabajo (EDT)

**Restricciones presupuestarias**

**Descripción**: Los recursos financieros disponibles son limitados, lo que restringe la inversión en infraestructura en la nube (por ejemplo, servidores AWS, Azure) y licencias de software propietario.

**Impacto**: Se excluye el uso de plataformas comerciales de alto costo, priorizando herramientas open-source como PostgreSQL para la base de datos, Node.js y React para el desarrollo web/móvil, y Chart.js para visualizaciones en el dashboard. Esto reduce los costos operativos, pero puede limitar la escalabilidad o el acceso a soporte técnico profesional.

**Mitigación**: La EDT incluye actividades específicas para el uso de herramientas gratuitas (por ejemplo, 1.2 Diseño de base de datos, 2.5.5 Dashboard con gráficos), y se buscará financiamiento externo, como aportes municipales o fondos de la Junta Nacional de Bomberos, regulados por la Ley N° 20.564 (Biblioteca del Congreso Nacional de Chile, 2012).

**Restricciones tecnológicas**

**Descripción**: La integración con el sistema VIPER depende de la disponibilidad de una API pública o de formatos de exportación compatibles (CSV/JSON). Si VIPER no proporciona una API funcional o los datos no son exportables en tiempo real, la sincronización será limitada o requerirá procesos manuales.

**Impacto**: Se excluye una integración totalmente automatizada con VIPER si no se cumplen las condiciones técnicas, lo que podría aumentar el tiempo de procesamiento de datos de emergencias.

**Mitigación**: La EDT incorpora actividades para analizar la API de VIPER (1.4.1) y desarrollar conectores alternativos (2.2.2 Importación de archivos CSV/JSON). Además, se implementará un modo offline (1.2.4 Configuración modo offline) para garantizar operatividad en entornos con conectividad limitada, como los cuarteles de bomberos.

**Restricciones humanas**

**Descripción**: La gestión, mantenimiento y soporte del sistema dependen de voluntarios del Cuerpo de Bomberos con conocimientos básicos de informática, lo que puede limitar la continuidad y la capacidad de respuesta ante problemas técnicos.

**Impacto**: Se excluye la contratación de personal técnico especializado a largo plazo, lo que podría retrasar la resolución de errores o actualizaciones del sistema.

**Mitigación**: La EDT dedica fases específicas a la capacitación intensiva de usuarios (3.2 Capacitación de usuarios) y al establecimiento de un equipo de soporte voluntario (4.3 Soporte continuo). Se desarrollarán manuales y videos tutoriales simplificados (3.2.1, 3.2.2) para facilitar la adopción por parte de voluntarios con experiencia técnica limitada.

**Restricciones legales**

**Descripción**: El sistema debe cumplir con la Ley N° 19.628 sobre Protección de Datos Personales en Chile, que regula el manejo de datos sensibles, como la información personal de los voluntarios (RUT, asistencia) y los registros de emergencias.

**Impacto**: Se excluyen funcionalidades que no garanticen el cumplimiento de esta normativa, como el almacenamiento de datos sin cifrado o la falta de autenticación segura.

**Mitigación**: La EDT incluye pruebas de seguridad (3.1.3 Pruebas de seguridad) para implementar autenticación con JWT, cifrado de datos (TLS/SSL, AES-256) y auditorías de acceso, asegurando el cumplimiento legal.

**Restricciones operacionales**

**Descripción**: La disponibilidad del sistema estará sujeta a la estabilidad de la red de internet en los cuarteles de bomberos, que puede ser variable, y a la correcta capacitación de los usuarios para operar la plataforma.

**Impacto**: Se excluye la garantía de operatividad continua en entornos con conectividad deficiente o si los voluntarios no están adecuadamente capacitados.

**Mitigación:** La EDT incorpora el desarrollo de un modo offline robusto (1.2.4, 3.1.2 Pruebas de integración) y sesiones de capacitación presenciales (3.2.3) para garantizar el uso efectivo del sistema. Además, se planifica un piloto (3.3 Puesta en marcha piloto) para identificar problemas operativos antes del despliegue general.

**5.2 Impacto de la solución.**

**5.2.1 Proceso de negocio afectado.**

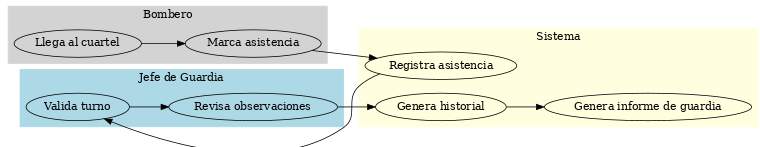
El proyecto impacta directamente los procesos operativos y estratégicos del Cuerpo de Bomberos de Rancagua, específicamente el registro de guardias nocturnas y la gestión de emergencias. Actualmente, estos procesos son predominantemente manuales, basados en libros foliados y reportes físicos, lo que genera ineficiencias, errores de transcripción, y falta de trazabilidad en tiempo real. La plataforma digital propuesta busca automatizar y optimizar los siguientes procesos clave:

* Registro de guardias nocturnas: Incluye la captura de asistencia de voluntarios, asignación de turnos, registro de inactividades (por falta de maquinista, insuficiencia de personal o decisión del Mando Operativo), y generación de informes operativos. Este proceso se alinea con la Ley N° 20.564, que exige transparencia en la gestión de recursos humanos en instituciones de bomberos (Biblioteca del Congreso Nacional de Chile, 2012).
* Gestión de emergencias: Involucra la importación de datos desde el sistema VIPER (hora, ubicación, tipo de emergencia, material mayor movilizado), el registro de múltiples claves radiales por siniestro (por ejemplo, Clave 10-0 + Clave 10-3), y la trazabilidad de materiales utilizados. Este proceso se alinea con ISO 22320:2018, que promueve la gestión eficiente de incidentes (ISO, 2018).
* Trazabilidad de materiales: Permite registrar el uso, estado (disponible, dañado, en mantenimiento) y reposición de insumos críticos, mejorando la planificación operativa.
* Generación de informes y estadísticas: Facilita la consolidación de datos para la planificación estratégica y la rendición de cuentas, incluyendo informes de guardias, emergencias, inactividades y stock de equipamiento.

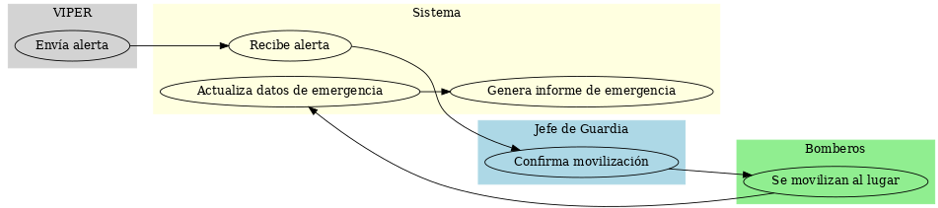
La automatización de estos procesos, detallada en la EDT (Anexo X: Diagrama EDT), reducirá el tiempo de registro, mejorará la precisión de los datos y optimizará la respuesta ante emergencias, beneficiando a los bomberos voluntarios, Jefes de Guardia, la Comandancia y la comunidad de Rancagua.

A continuación, se presenta la representación gráfica en notación BPMN de los procesos afectados, lo que permite visualizar las actividades, roles y flujos de información involucrados en el sistema propuesto:

**BPMN para Guardia Nocturna:**



**BPMN para Sistema de Emergencias:**



**5.2.2 Registro de Interesados.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Interesado** | **Rol en la organización** | **Interés en el proyecto** | **Nivel de influencia** | **Expectativas** |
| **Bomberos voluntarios** | Usuarios principales del sistema, responsables de asistir a las guardias y atender emergencias | Acceso fácil y rápido a su historial de guardias y emergencias e inactividades, con interfaz para uso en dispositivos móviles | Medio | Tener un sistema práctico que simplifique su gestión diaria, reduzca el tiempo de registro y permita consultar turnos y material disponible. |
| **Jefes de Guardia** | Supervisores operativos, responsables de gestionar turnos, registrar cuando están fuera de servicio y reportar emergencias | Registro eficiente de guardias, emergencias y generación de informes operativos. | Alto | Contar con datos confiables y ahorrar tiempo en transcripciones y recibir notificaciones automáticas sobre inactividades |
| **Comandancia** | Dirección estratégica, encargada de la planificación y toma de decisiones. | Uso de estadísticas y reportes para planificación y toma de decisiones, planificar capacitaciones y rendir cuentas. | Muy alto | Disponer de información en tiempo real consolidada de las guardias, materiales e inactividades para mejorar la operatividad. |
| **Comunidad de Rancagua** | Beneficiarios indirectos del servicio | Mejorar tiempos de respuesta y calidad del servicio en emergencias | Bajo | Recibir un servicio oportuno, eficiente y seguro con una respuesta más rápida. |
| **Junta Nacional de Bomberos** | Entidad que supervisa el financiamiento y cumplimiento normativo | Asegura que el sistema se cumpla con estándares de transparencia y gestión | Medio | Un sistema que facilite la rendición de cuentas y la gestión eficiente de recursos |
| **Equipo Técnico** | Voluntarios y estudiantes con conocimiento en informática encargados del desarrollo y mantenimiento del sistema. | Implementar un sistema funcional con recursos limitados con herramientas open-source. | Alto | Contar con soporte claro para el desarrollo, mantenimiento, y resolución de problemas técnicos. |

**5.2.3 Indicadores de gestión.**

|  |  |  |
| --- | --- | --- |
| **Indicador** | **Descripción** | **Fórmula de cálculo** |
| **Tiempo promedio de registro** | Mide la rapidez con que un jefe de guardia puede ingresar un turno o emergencia en el sistema, inactividad o uso de materiales en emergencias | (Σ Tiempos de registro / Nº de registros) |
| **Tasa de errores de registro** | Porcentaje de registros con errores u omisiones detectados respecto del total ingresado. | (Registros con error / Registros totales) × 100 |
| **Número de guardias registradas** | Total, de guardias registradas digitalmente en un periodo de tiempo. | Σ Guardias registradas en el periodo |
| **Distribución de emergencias por tipo** | Permite analizar la frecuencia relativa de cada tipo de emergencia atendida en un periodo. | (Emergencias de un tipo / Emergencias totales) × 100 |
| **Tasa de reposición de materiales** | % de insumos críticos repuestos o reparados tras alertas generadas por el sistema. | (Materiales repuestos o reparados / Materiales con alerta) x 100 |
| **Tiempo promedio de respuesta a emergencias** | Mide el tiempo de la percepción de la emergencia a la llegada del lugar del incidente, basado en datos de VIPER | (Σ Tiempos de respuesta / Nº de emergencias) |

**5.2.4 Niveles de servicio.**

Los niveles de servicio propuestos para el sistema se presentan en la siguiente tabla:

|  |  |  |
| --- | --- | --- |
| **Categoría** | **Nivel de Servicio Propuesto** | **Descripción** |
| **Disponibilidad** | 99,9% | El sistema estará disponible la mayor parte del tiempo, considerando un máximo de 8 horas de inactividad anual por mantención o fallas técnicas, soportado por infraestructura en la nube y modo offline. |
| **Accesibilidad** | Multiplataforma (web y móvil) | Acceso desde navegadores en PC (Chrome, Firefox) y desde dispositivos móviles (Android/iOS) con interfaces responsive diseñadas en la EDT (1.3) |
| **Tiempo de respuesta** | ≤ 3 segundos por transacción | Los módulos principales (login, registro de guardias, emergencias, inactividades, consulta de informes) deben responder en un tiempo máximo de 3 segundos bajo condiciones normales de red, optimizado en la fase de desarrollo. |
| **Seguridad** | Autenticación por roles y cifrado de datos | Se implementará un control de acceso diferenciado (bombero, jefe de guardia, administrador) con JWT y cifrado TLS/SSL y AES-256 para datos sensibles, cumpliendo con la Ley N° 19.628 (EDT 3.1.3). |
| **Escalabilidad** | Crecimiento anual de hasta 20% en registros sin pérdida de rendimiento | La base de datos y la infraestructura soportará un aumento progresivo de usuarios y registros históricos (guardias, emergencias, materiales) sin pérdida de rendimiento, diseñado en la EDT (1.2). |
| **Soporte Técnico** | Respuesta en <= 24 horas | El equipo de soporte (voluntarios técnicos) resolverá incidencias en un máximo de 24 horas, con un canal de comunicación dedicado (email/WhatsApp), establecido en la EDT (4.3). |

**5.3 Objetivos del proyecto**

**5.3.1 Objetivo General.**

Desarrollar e implementar una plataforma digital integrada al sistema VIPER que permita registrar guardias nocturnas, emergencias, inactividades y trazabilidad de materiales del Cuerpo de Bomberos de Rancagua, con el propósito de mejorar la eficiencia operativa, garantizar la trazabilidad de datos, cumplir con normativas legales (Ley N° 19.628), y apoyar la planificación estratégica mediante informes y estadísticas consolidadas.

**5.3.1 Objetivo Específico.**

En concordancia con el ciclo de vida del desarrollo de un sistema, se plantean los siguientes objetivos específicos:

1. **Levantar y documentar requerimientos:** funcionales y no funcionales del sistema, considerando entrevistas, observaciones y encuestas a bomberos y jefes de guardia, incluyendo registros de guardias, emergencias, inactividades y trazabilidad de materiales.
2. **Analizar los procesos actuales**: Identificar puntos críticos, ineficiencias y oportunidades de mejora en el registro manual de guardias, emergencias, inactividades y uso de materiales, mediante el análisis de libros foliados y reportes VIPER.
3. **Diseñar la arquitectura tecnológica** Desarrollar una arquitectura escalable y accesible (web y móvil) utilizando herramientas open-source (por ejemplo, PostgreSQL, Node.js, React) para minimizar costos, con soporte para modo offline y cumplimiento de seguridad (Ley N° 19.628).
4. **Desarrollar y programar la plataforma digital**: Programar módulos para registro de guardias, gestión de inactividades (causas y notificaciones), trazabilidad de materiales, integración con VIPER (API o importación CSV/JSON), y generación de informes/estadísticas.
5. **Probar y validar el sistema:** Ejecutar pruebas funcionales, de integración y de seguridad, seguido de un piloto en una compañía de bomberos para evaluar usabilidad, confiabilidad y cumplimiento de niveles de servicio (≤ 3 segundos por transacción, 99,9% disponibilidad).
6. **Capacitar a los usuarios y establecer soporte continuo:** Desarrollar manuales, videos tutoriales y sesiones presenciales para capacitar a bomberos y Jefes de Guardia, y crear un canal de soporte (email/WhatsApp) gestionado por voluntarios técnicos.
7. **Metodología de Trabajo**

**6.1 Metodología de Desarrollo de la solución.**

Se adoptará un enfoque ágil basado en Scrum, debido a su flexibilidad para adaptarse a cambios en los requerimientos, su enfoque en entregables funcionales en iteraciones cortas (sprints de 2-3 semanas), y su compatibilidad con las restricciones humanas y presupuestarias del proyecto (dependencia de voluntarios, uso de herramientas open-source). Scrum permite una colaboración estrecha con los interesados (bomberos, Jefes de Guardia, Comandancia) y asegura la validación continua mediante pruebas y retroalimentación. Las principales características de la metodología incluyen:

**Roles Scrum:**

* Product Owner: Representantes de bomberos y Jefes de Guardia (REP), quienes definen prioridades y validar requerimientos.
* Scrum Master: Carlos Trincado (CT), líder técnico que facilita ceremonias y elimina impedimentos.
* Equipo de Desarrollo: Matías Vargas (MV) y Nicolás Aguilera (NA), encargados de front-end, back-end, QA y documentación.
* Interesados externos: Académico guía (AG) y Comandancia (COM), que proporcionan asesoría y validación estratégica.
* Ceremonias: Planificación de sprints, reuniones diarias (stand-ups), revisiones de sprint (demos con usuarios), y retrospectivas para mejora continua.
* Artefactos: Product Backlog (requerimientos funcionales y no funcionales), Sprint Backlog (tareas por sprint), Incremento (entregables funcionales, como módulos de guardias, VIPER, inactividad, materiales, informe
* Herramientas: Jira o Trello para gestión de tareas, Git para control de versiones, Figma para diseño de interfaces, y herramientas open-source (PostgreSQL, Node.js, React, Chart.js) para desarrollo, alineadas con la restricción presupuestaria.
* Pruebas y validación: Pruebas unitarias, de integración, y de usuario en cada sprint, culminando en un piloto (EDT 3.3) para validar usabilidad y niveles de servicio (5.2.4).

**6.2 Duración y cronograma.**

***Semanas 1-2 (01/10/2025 - 08/10/2025): Análisis y diseño inicial***

* Actividades: Levantamiento de requerimientos (entrevistas, observaciones, encuestas), análisis de procesos manuales, diseño de base de datos, prototipos de interfaz, y especificación de integración con VIPER (EDT 1.1-1.4).
* Deliverables: Documento de requerimientos, mapa de procesos, diagrama entidad-relación, prototipos en Figma, documento de integración.
* Hito: Finalización del diseño inicial 08/10/2025

***Semanas 3-8 (09/10/2025 - 29/11/2025): Desarrollo - Sprint 1, Sprint 2 y Sprint 3***

* **Sprint 1** (Semanas 3-4, 09/10/2025 - 22/10/2025): Desarrollo del módulo de registro de guardias (login, asistencia, turnos, historial) y pruebas unitarias (EDT 2.1, 3.1.1).
* Deliverables: Módulo funcional de guardias, reportes de pruebas unitarias.
* **Sprint 2** (Semanas 5-6, 23/10/2025 - 05/11/2025): Desarrollo del módulo de integración con VIPER (consumo de API o importación CSV/JSON) y módulo de gestión de inactividad (formulario, notificaciones) con pruebas unitarias (EDT 2.2, 2.3, 3.1.1).
* Deliverables: Conector VIPER funcional, módulo de inactividad, reportes de pruebas unitarias.
* **Sprint 3** (Semanas 7-8, 06/11/2025 - 19/11/2025): Desarrollo del módulo de trazabilidad de materiales (registro, alertas, inventario) y módulo de informes/estadísticas (informes de guardias, emergencias, inactividad, stock) con pruebas unitarias (EDT 2.4, 2.5, 3.1.1).
* Deliverables: Módulos de materiales e informes, dashboard interactivo, reportes de pruebas.
* Hito: Sistema completo desarrollado (19/11/2025).

***Semanas 9-10 (20/11/2025 - 13/12/2025): Implementación, piloto, ajustes, documentación y soporte***

* **Actividades*:*** Pruebas de integración y seguridad (EDT 3.1), capacitación de usuarios (manuales, videos, sesiones presenciales; EDT 3.2), ejecución de piloto (EDT 3.3), ajustes basados en feedback (EDT 3.4), documentación técnica y de usuario (EDT 4.1, 4.2), establecimiento de soporte continuo (EDT 4.3).
* **Deliverables**: Reportes de pruebas de integración, materiales de capacitación, sistema operativo en piloto, informe de feedback, sistema ajustado, manuales técnicos y de usuario, plan de soporte.
* **Hito**: Entrega final del sistema (13/12/2025).

**Gráfico, Gráfico en cascada

El contenido generado por IA puede ser incorrecto.**

Gráfico, Gráfico en cascada

El contenido generado por IA puede ser incorrecto.

**Interfaz de usuario gráfica, Aplicación, Tabla

El contenido generado por IA puede ser incorrecto.**

**Nota**: *El cronograma está alineado con la Carta Gantt (Anexo 10), que detalla las duraciones y dependencias de las tareas, considerando restricciones como la disponibilidad de voluntarios y la conectividad en los cuarteles.*

**6.3 Equipo de trabajo.**

El equipo de desarrollo está compuesto por tres estudiantes que asumen roles principales y de apoyo. Las responsabilidades se organizan en una Matriz RACI, en concordancia con la metodología ágil Scrum. Los roles serán:

**CT** : Carlos Trincado (Líder técnico / Back End)

**MV** : Matías Vargas (UX/UI / Front End)

**NA** : Nicolás Aguilera (QA / Documentación)

**AG** : María Magdalena Becerra(Académico Guía)

**COM** : Comandancia (dirección del Cuerpo de Bomberos)

**REP** : Representantes de bomberos / usuarios clave

* **CT (Carlos Trincado)**: Lidera el desarrollo técnico (back-end, integración con VIPER) y coordina como Scrum Master, ejecutando tareas críticas como diseño de base de datos (1.2) y módulos de desarrollo (2.1-2.5).
* **MV (Matías Vargas)**: Responsable del diseño y desarrollo de interfaces (front-end, EDT 1.3, 2.1, 2.3-2.5), asegurando usabilidad para usuarios no técnicos.
* **NA (Nicolás Aguilera)**: Lidera pruebas (QA, EDT 3.1) y documentación (EDT 4.1, 4.2), garantizando calidad y cumplimiento de requerimientos.
* **AG (María Magdalena Becerra)**: Proporciona asesoría técnica y metodológica, revisando entregables claves.
* **COM (Comandancia)**: Aprueba resultados estratégicos y recibe informes, alineada con su rol de dirección (5.2.2).
* **REP (Representantes de bomberos)**: Define prioridades como Product Owner, válida requerimientos y participa en el piloto, representando a los usuarios finales.

y la leyenda de la matriz **RACI** será:

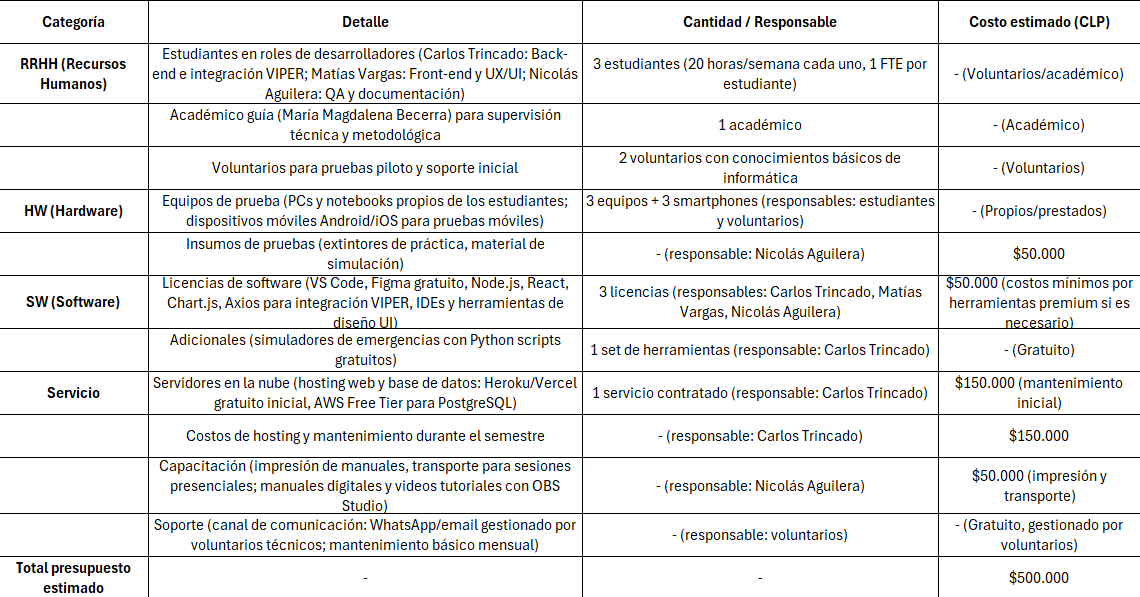
**R**:Responsable (ejecuta la tarea)

**A**: Accountable (responsable final)

**C**: Consulted (Dar apoyo/asesoría, asistencia)

**I**:Informed (mantener informado al equipo)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tarea (EDT) | Scrum Master | Desarrollador Front-end | QA/Documentador | Asesor Metodológico | Dirección Estratégica | Product Owner |  |  |
| 1.1 Análisis de procesos actuales | R | C | C | C | I | A |  |  |
| 1.2 Diseño de base de datos | R | C | C | C | A | I |  |  |
| 1.3 Diseño de interfaces | C | R | C | C | I | A |  |  |
| 1.4 Integración con VIPER | R | C | C | C | A | I |  |  |
| 2.1 Módulo de registro de guardias | R | R | C | I | I | C |  |  |
| 2.2 Módulo de integración con VIPER | R | C | C | I | I | C |  |  |
| 2.3 Módulo de gestión de inactividad | R | R | C | I | I | C |  |  |
| 2.4 Módulo de trazabilidad de materiales | R | R | C | I | I | C |  |  |
| 2.5 Módulo de informes y estadísticas | R | R | C | I | I | C |  |  |
| 3.1 Pruebas unitarias y de integración | C | C | R | C | I | A |  |  |
| 3.2 Capacitación de usuarios | C | R | R | C | I | R |  |  |
| 3.3 Puesta en marcha piloto | R | C | R | C | I | R |  |  |
| 3.4 Feedback y ajustes | R | C | R | C | I | C |  |  |
| 4.1 Documentación técnica | R | I | R | C | I | C |  |  |
| 4.2 Documentación de usuario | C | R | R | C | I | C |  |  |
| 4.3 Soporte continuo | R | C | R | C | A | I |  |  |

**6.4 Recursos a Utilizar**

1. **Definición de arquitectura TI**

La arquitectura tecnológica de la plataforma digital para el Cuerpo de Bomberos de Rancagua está diseñada como una arquitectura basada en microservicios con un enfoque cliente-servidor y una estructura de 3 capas, optimizada para entornos cloud-native. Esta combinación permite cumplir con los objetivos del proyecto (5.3), abordar los procesos afectados (5.2.1: registro de guardias, emergencias, inactividades, trazabilidad de materiales), satisfacer las expectativas de los interesados (5.2.2: bomberos, Jefes de Guardia, Comandancia), y cumplir con los niveles de servicio (5.2.4: 99,9% disponibilidad, ≤ 3 segundos por transacción, seguridad). La arquitectura respeta las restricciones presupuestarias (uso de herramientas open-source), tecnológicas (integración con VIPER, modo offline), humanas (interfaz simple para voluntarios), legales (Ley N° 19.628), y operacionales (conectividad limitada en cuarteles). Se alinea con la metodología Scrum (6.1), el cronograma de 24 semanas (6.2, Anexo Y: Carta Gantt), y las tareas de la EDT (Anexo X), asegurando escalabilidad, seguridad, y operación eficiente en entornos de cuarteles y campo. La arquitectura está diseñada según estándares de gestión de emergencias (ISO 22320:2018, NFPA 1720) y mejores prácticas de desarrollo cloud-native.

## **Detalle de la arquitectura**

* **Arquitectura de microservicios**: La plataforma se divide en servicios independientes (por ejemplo, módulo de guardias, integración con VIPER, gestión de inactividad, trazabilidad de materiales, reportes), cada uno con su propio contenedor (Docker) y API RESTful. Esto permite escalabilidad horizontal, desarrollo modular (EDT 2.1-2.5), y despliegue continuo (EDT 3.1), optimizando el uso de recursos open-source y cumpliendo con la restricción presupuestaria (5.1.2).
* **Enfoque cliente-servidor**: La interfaz web/PWA (cliente) interactúa con un conjunto de microservicios (servidores) a través de un API Gateway (por ejemplo, Nginx), asegurando comunicación segura (SSL/TLS) y soporte para modo offline (IndexedDB), lo que mitiga la restricción tecnológica (5.1.2) y mejora la disponibilidad (99,9%, 5.2.4).
* **Estructura de 3 capas**:
  + **Capa de presentación**: Interfaz web/PWA (React) para usuarios (EDT 1.3), diseñada con accesibilidad (WCAG 2.1) para voluntarios no técnicos.
  + **Capa de lógica de negocio**: Microservicios (Node.js) que manejan la lógica de registro, integración, y reportes (EDT 2.1-2.5), con soporte para modo offline.
  + **Capa de datos**: Base de datos central (PostgreSQL) con réplicas y cache (Redis) para rendimiento (≤ 3 segundos por transacción, 5.2.4), cumpliendo con la Ley N° 19.628 (cifrado AES-256).
* **Cloud-native**: Utiliza servicios como AWS Free Tier o Heroku para hosting (6.4), asegurando escalabilidad y operación en entornos con conectividad limitada.
* **Diagrama de la arquitectura propuesta**:

Forma, Flecha, Círculo

El contenido generado por IA puede ser incorrecto.Diagrama

El contenido generado por IA puede ser incorrecto.

## **7.1 Visión general**

* **Interfaz de usuario**: Una interfaz web responsiva y una Progressive Web App (PWA) para acceso desde navegadores (Chrome, Firefox) y dispositivos móviles (Android/iOS), soportando registro de guardias, emergencias, inactividades, trazabilidad de materiales, y dashboards analíticos (EDT 1.3, 2.1-2.5).
* **API central**: Gestiona todas las peticiones entre front-end, back-end, y sistemas externos (VIPER), asegurando seguridad y escalabilidad (EDT 1.4, 2.2).
* **Servicios de negocio**: Módulos para registro de guardias, gestión de emergencias, inactividad de la Guardia Nocturna, trazabilidad de materiales, y reportes/estadísticas (EDT 2.1-2.5).
* **Base de datos centralizada**: Almacena datos transaccionales (guardias, emergencias, materiales, inactividades) con soporte para modo offline y cumplimiento de la Ley N° 19.628 (EDT 1.2).
* **Conector VIPER**: Importa datos de emergencias (hora, ubicación, tipo, material movilizado) vía API o archivos CSV/JSON (EDT 2.2).
* **Infraestructura en la nube**: Prioriza herramientas open-source y servicios gratuitos (por ejemplo, AWS Free Tier, Heroku) para minimizar costos (5.1.2).
* **Observabilidad y CI/CD**: Herramientas para monitoreo, logs, y despliegue continuo, asegurando calidad y mantenimiento (EDT 3.1, 4.3).

**7.2 Componentes y responsabilidades**

|  |  |  |
| --- | --- | --- |
| **Componente** | **Función principal** | **Recomendación técnica (ejemplos)** |
| Interfaz Web + PWA / App Móvil | Registro guardias, consultas historiales, dashboards, trazabilidad de materiales, consultas historiales, interactivo con estadísticas. | React (Web) + React Native / PWA; responsive; accessible, uso de Chart.js para dashboard |
| API Gateway / LB | Unifica peticiones front end, rate limiting, balanceo de carga, y seguridad SSL | Nginx/ALB + Kong/Traefik si se requiere Gateway avanzado |
| Servicio de Autenticación | Login con RUT, MFA opcional, roles(Bombero, Jefe Guardia, Admin) | OAuth2 + JWT; integración con LDAP/AD opcional, cifrado. |
| Backend / Servicios | Lógica de negocio (guardias, emergencias, reportes), soporte para modo offline y sincronización | Node.js+Express, Python+Django, o Java+Spring; containerized |
| Base de datos | Almacenamiento de datos transaccionales (guardias, emergencias, materiales, inactividades); soporte para modo offline y recuperación (PITR). | PostgreSQL (AWS RDS Free Tier o local); réplicas para disponibilidad; IndexedDB para modo offline en PWA. |
| Cache / Queue | Mejora rendimiento / desacopla procesos(notificaciones sistema VIPER) | Redis para cache; RabbitMQ o Redis Streams para mensajería (notificaciones de inactividad). |
| Storage | Archivos, PDF, imágenes de materiales, disponibilidad Multi-AZ | AWS S3 Free Tier o Google Cloud Storage; cifrado para cumplir con Ley N° 19.628. |
| Conector VIPER | Importación y sincronización de datos de emergencias (hora, ubicación, tipo, materiales) desde VIPER vía API o archivos CSV/JSON | Servicio Node.js que consume REST API o procesa CSV/JSON; validación de datos para consistencia. |
| Observabilidad | Logs, métricas, alertas de fallos (indisponibilidad del sistema) | ELK Stack (Elasticsearch, Logstash, Kibana) o AWS CloudWatch Free Tier; Grafana + Alertmanager para visualización. |
| CI/CD | Automatización de despliegues, pruebas y configuración de infraestructura; mejora calidad y velocidad de entrega. | GitHub Actions / GitLab CI + Terraform / Ansible para configuración |

## **7.3 Arquitectura de despliegue y disponibilidad**

La arquitectura de despliegue está diseñada para garantizar alta disponibilidad y escalabilidad, alineada con los niveles de servicio (5.2.4: 99,9% disponibilidad, ≤ 3 segundos por transacción), los objetivos (5.3), y las restricciones (5.1.2: presupuestaria con open-source, tecnológica con modo offline). Se implementa según la EDT (Anexo X) y el cronograma (6.2, Anexo Y: Carta Gantt) en un entorno cloud-native.

## **Configuración de despliegue**

* **Proveedor cloud**: Despliegue multi-AZ (Availability Zone) en AWS, GCP, o Azure para alta disponibilidad.
* **Contenerización y orquestación**: Uso de Docker para contenerizar microservicios y Kubernetes (EKS, GKE, o AKS) para orquestación y escalado automático.
* **Base de datos**: Servicio gestionado (RDS en AWS, Cloud SQL en GCP) con réplicas y backups automáticos.
* **Autoscaling**: Configuración para backends y workers según demanda (métricas de CPU/memoria).

## **Estrategia de disponibilidad y backup**

* **Backups**: Copias diarias + snapshot semanal, con retención de 1 año para registros operativos (EDT 4.3).
* **SLA objetivo**: Disponibilidad del 99,9%, con tolerancia a fallos mediante multi-AZ y réplicas.

## **Beneficios**

* Escalabilidad horizontal para picos de emergencias (5.2.1).
* Cumplimiento legal (Ley N° 19.628) con backups cifrados.
* Soporte a restricciones operacionales (conectividad limitada) con modo offline.

## **7.4 Integración con VIPER (opciones)**

## **Opciones de integración**

* **API directa (preferida)**:
  + Uso de la API de VIPER (si existe) para suscripción a eventos o consultas en tiempo real.
  + Implementación: Autenticación segura (OAuth 2.0), mapping de campos (incidentes, tipos).
* **Batch import (alternativa)**:
  + VIPER exporta CSV/JSON; un conector procesa y normaliza cada N minutos (por ejemplo, cada 15 minutos).
* **Webhook/Listener (si aplica)**:
  + Suscripción de un endpoint para recibir notificaciones push si VIPER lo soporta.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fase | Actividad | Duración | Responsable | Entregable |
| Descubrimiento | Identificar API/webhook VIPER | Semana 10 | Carlos Trincado | Documento Especificación |
| Definición | Contrato(endpoints, campos frecuencia) | Semana 11 | Carlos Trincado | Contrato de integración |
| Prototipo | Desarrollo de conector de prueba | Semana 11-12 | Carlos Trincado | Prototipo Funcional |
| Pruebas | Test con datos reales | Semana 12 | Nicolas Aguilera | Reporte de Pruebas |
| Sincronización | Implementación incremental | Semana 12 | Carlos Trincado | Sistema Sincronizado |
| Monitorización | Configuración de logs y alertas | Semana 12 | Nicolas Aguilera | Plan de monitorización |

## 

## **7.5Modelo de datos**

**Tablas principales:**

* **users** (id, rut, nombre, rol, contacto, estado)
* **guardias** (id, fecha\_inicio, fecha\_fin, cuartel, jefe\_guardia\_id)
* **asistencias** (id, guardia\_id, user\_id, hora\_checkin, hora\_checkout)
* **emergencias** (id, vipercode, tipo, fecha\_hora, ubicacion, tiempo\_salida, tiempo\_llegada, observaciones)  
  **material\_mayor** (id, tipo, estado, ubicacion)  
  **inventario** (id, item, cantidad, estado, ultimo\_control)
* **reportes** (id, tipo, periodo, archivo\_url, generado\_por)
* **audit\_log** (tabla de auditoría de cambios)

## **7.6 Requisitos no funcionales**

**Disponibilidad:** 99.9% (distribución multi-AZ).

**Performance:** respuesta < 3 seg en operaciones habituales; throughput según picos de aviso.

**Escalabilidad:** soportar aumento anual de registros ~20%.

**Seguridad:** cifrado TLS in-transit, cifrado at-rest, IAM least privilege, logs de auditoría.  
**Mantenibilidad:** despliegues automatizados, IaC (Terraform), testing automatizado.

**Portabilidad:** contenerizado, infraestructura descrita con IaC.

## **7.7 Seguridad y cumplimiento**

1. **Autenticación y Autorización:** RUT + contraseña; JWT; roles RBAC (Bombero, Jefe, Admin). MFA opcional para cuentas de mando.
2. **Cifrado:** TLS 1.2/1.3 para transporte; cifrado AES-256 para datos sensibles en reposo.
3. **Cumplimiento legal:** cumplir Ley 19.628 (protección datos personales, Chile) — políticas de retención y consentimiento.
4. **Registro y auditoría:** todas las acciones críticas (check-in/out, edición de emergencias) generan entradas en **audit\_log.**
5. **Gestión de secretos:** KMS (AWS KMS / Cloud KMS).
6. **Backups y recuperación:** prueba de restauración anual; RTO y RPO definidos.

## **7.8 Entregables de arquitectura**

1. Diagrama de arquitectura (lógico y físico).
2. Documentación API (OpenAPI/Swagger).
3. Contrato de integración VIPER.
4. Plan de Seguridad y Política de Retención.
5. Playbooks de despliegue y recuperación.
6. **Reconocimiento de arquitectura empresarial**

En este punto vamos a realizar la identificación del tipo de organización y su estructura de manera detallada y bien clara para tener una perspectiva amplia del proyecto.

## **8.1 Tipo de organización y estructura**

El Cuerpo de Bomberos de Rancagua es una organización voluntaria sin fines de lucro, estructurada en Comandancia y Compañías (8 compañías). Jerarquía típica:

Superintendente → Comandancia / Comandante → Capitanes de Compañía → Jefes de Guardia → Voluntarios. Predomina la voluntariedad, autogestión y operación distribuida por cuarteles.

## **8.2 Mapeo de capacidades empresariales (Business Capabilities)**

Las capacidades clave afectadas por la solución son:

|  |  |
| --- | --- |
| **Capacidad** | **Descripción** |
| Gestión de Guardia | Planificar, asignar y registrar guardias nocturnas. |
| Operaciones de Emergencia | Recepción, despacho y gestión de incidentes. |
| Administración de Recursos | Mantenimiento de material mayor e inventario. |
| Formación y Capacitación | Registro y gestión de capacitaciones y competencias. |
| Gobernanza y Reportes | Generación de indicadores para mando y comités. |
| Comunicaciones y Vinculación | Relación con comunidad, hospitales, policía. |

## **8.3 Procesos de negocio impactados**

**Proceso principal:** gestión de guardia → registro de asistencia → reporte de emergencias → análisis operativo.

**Proceso transversal:** gestión de inventario y mantenimiento de material mayor.

**Proceso estratégico:** planificación de capacitaciones y distribución de recursos basada en datos.

## **8.4 Roles y gobernanza TI**

Para operar y mantener la solución proponemos roles y responsabilidades:

|  |  |
| --- | --- |
| **Rol** | **Responsabilidad** |
| **Administrador TI / Responsable Técnico (interno o contratado)** | Operación diaria del sistema, despliegues menores, backups. |
| **Coordinador Operativo (Comandancia)** | Validación de reportes operativos, aprobación de despliegue piloto. |
| **Académico Guía (supervisión del proyecto)** | Revisión académica, validación de entregables y metodología. |
| **Administradores de Cuartel** | Soporte local, gestión de usuarios, control de inventario físico. |
| **Usuario final (Jefe de Guardia, Bombero)** | Uso diario del sistema, reporte de incidencias. |

## **8.5 Data governance y políticas**

* **Propiedad de datos:** la Comandancia es responsable de datos operativos; el equipo TI custodia la infraestructura.
* **Calidad de datos:** definir validaciones obligatorias en captura (ubicación, hora, RUT).
* **Retención:** políticas por tipo de dato (ej.: registros operativos 5 años, logs 1 año).
* **Accesos:** controles RBAC y revisiones periódicas de cuentas.

## **8.6 Impacto organizacional y roadmap de adopción**

**Impacto esperado:** mejora en trazabilidad, reducción de tiempos administrativos, mejores decisionesestratégicas.

**Riesgos:** baja adopción por voluntarios, limitaciones de conectividad en terreno, dependencia de integración con VIPER.

**Mitigación:** capacitaciones, modo offline local con sincronización, piloto por compañía antes de despliegue total.

**Roadmap :**

Fase 0 – Preparación: acuerdos con VIPER, definir product backlog, infra mínima.

* Fase 1 – MVP: registro digital de guardias + reportes básicos + sincronización batch VIPER. Piloto en 1 compañía.
* Fase 2 – Extensión: app móvil, paneles estadísticos, informe de stock, mejoras UX. Piloto en 3 compañías.
* Fase 3 – Escalado: integración real-time con VIPER (si posible), optimizaciones, políticas de operación.
* Fase 4 – Operación y mejora continua: gobernanza, SLA, entregas periódicas y retroalimentación.

## **8.7 Capacitación y cambio organizacional**

**Plan de capacitación:** sesiones presenciales + material en línea; tutores locales en cada compañía.

**Entrenamiento piloto:** entrenamiento de jefes de guardia y administradores antes del despliegue masivo.  
**Comunicaciones:** campaña interna para adopción, FAQs y soporte telefónico/email.

## **8.8 Indicadores de éxito (relacionados a negocio)**

* % de guardias registradas digitalmente vs planificadas.
* Reducción promedio (minutos) del tiempo dedicado por jefe de guardia a transcripción.
* Precisión en inventario (% de items con control actualizado).
* Uso de reportes por Comandancia (nº de reportes consultados/mes).

1. Detalle de las Tecnología Para Implementar

**9.1 Análisis cualitativo/cuantitativo de las tecnologías que serán implementadas**

Con el objetivo de digitalizar los procesos de registro de guardias, trazabilidad de materiales y coordinación con el sistema VIPER, se analizaron dos alternativas tecnológicas viables:

**9.1.1 Indicar dos alternativas de solución**

| **Alternativa** | **Descripción** |
| --- | --- |
| **Alternativa A: Sistema monolítico en servidor local** | Implementar una aplicación web centralizada en un servidor físico ubicado en dependencias del Cuerpo de Bomberos de Rancagua. Los usuarios accederían mediante la red local (intranet) o conexión VPN. |
| **Alternativa B: Sistema basado en microservicios en la nube (AWS/Azure)** | Desarrollar una arquitectura distribuida bajo contenedores Docker, con despliegue en la nube híbrida (AWS + Azure). Cada módulo (guardias, materiales, reportes, integración VIPER) opera de forma independiente bajo un API Gateway seguro. |

**9.1.2 Estudios de Factibilidad**

**A) Factibilidad Técnica**

| Criterio | Alternativa A (Monolítica) | Alternativa B (Microservicios Cloud) |
| --- | --- | --- |
| Infraestructura | Servidor físico local; dependencia del hardware. | Infraestructura escalable en la nube (AWS EC2, Azure). |
| Escalabilidad | Limitada, requiere nuevos servidores para crecer. | Escalabilidad horizontal automática con contenedores Docker. |
| Mantenibilidad | Actualizaciones complejas y centralizadas. | Módulos independientes, actualizables sin detener el sistema. |
| Seguridad | Firewall básico y antivirus. | Cifrado SSL/TLS, autenticación OAuth2, IAM y respaldo automático. |
| Disponibilidad | 90% (riesgo por fallas eléctricas). | 99.9% (redundancia geográfica). |
| Compatibilidad | Navegadores locales, sin soporte móvil. | Web progresiva (PWA), compatible con cualquier dispositivo. |
| evaluación Operacional | 3/5 | 5/5 |

**Breve conclusión:** La Alternativa B (Microservicios Cloud) es más factible técnicamente al ofrecer mayor disponibilidad, modularidad y seguridad.

**B) Factibilidad Económica**

| Concepto | Alternativa A (Monolítica) | Alternativa B (Microservicios Cloud) |
| --- | --- | --- |
| Inversión inicial (equipos, licencias, instalación) | $1.200.000 CLP | $600.000 CLP |
| Costos anuales de mantenimiento | $300.000 CLP | $200.000 CLP |
| Escalabilidad (costos de expansión) | Alta inversión (nuevo servidor físico). | Escalable bajo demanda (modelo pay-per-use). |
| Soporte técnico | Interno (bombero TI). | Externo y automatizado (AWS/Azure). |
| Evaluación Operacional | 3/5 | 4/5 |

**Breve conclusión:** La Alternativa B es más rentable en el tiempo, con menores costos de mantenimiento y mejor relación costo-beneficio.

**Estimación básica de costos**

| **Concepto** | **Costo estimado (CLP)** | **Periodicidad** |
| --- | --- | --- |
| Hosting y base de datos en la nube (AWS/Azure) | $40.000 | Mensual |
| Dominio y certificado SSL | $20.000 | Anual |
| Licencias o herramientas adicionales (opcionales) | $30.000 | Anual |
| **Total, estimado** | **$70.000 - $90.000** | **Anual** |

**C) Factibilidad Operacional (de Implementación)**

| Criterio | Alternativa A (Monolítica) | Alternativa B (Microservicios Cloud) |
| --- | --- | --- |
| Facilidad de adopción | Requiere instalación manual por compañía. | Accesible vía navegador, sin instalación. |
| Curva de aprendizaje | Moderada (entorno limitado). | Baja, interfaz intuitiva PWA. |
| Adaptabilidad al trabajo voluntario | Baja (depende de acceso físico al servidor). | Alta (uso desde cualquier dispositivo móvil). |
| Evaluacion Operacional | 3/5 | 5/5 |

**Breve conclusión:** La Alternativa B es más viable operacionalmente, facilitando la adopción y el acceso remoto por parte de los voluntarios.

**Aclaración breve:** Si bien ambas alternativas (A: Monolítica y B: Microservicios Cloud) requieren procesos de capacitación para el personal voluntario, este factor no se considera dentro de la evaluación económica, dado que la inversión en formación es equivalente en ambas opciones y no representa un costo diferencial relevante. Por tanto, la capacitación se incluye como una condición general del proyecto, no como un criterio discriminante de factibilidad económica.

**9.1.3 Selección de alternativa (Método de puntuación, incluir todo el desarrollo, los únicos factores costo, rapidez y seguridad)**

El método de puntuación ponderada considera los factores solicitados:

Costo (40%), Rapidez (30%), y Seguridad (30%).

Para determinar la alternativa más adecuada, se utilizó un método de puntuación ponderado basado en los factores solicitados: Costo (40%), Rapidez (30%) y Seguridad (30%). A diferencia del método tradicional, se aplicó la fórmula de normalización de factores recomendada para una evaluación objetiva:

Texto

El contenido generado por IA puede ser incorrecto.

En la siguiente tabla se presenta el cálculo de cada alternativa según su desempeño en los factores definidos:

| **Criterio** | **Peso** | **A (1–5)** | **B (1–5)** | **Nota A** | **Nota B** | **Ponderado A** | **Ponderado B** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Costo** (inverso) | 40% | 3 | 4 | 40 | 100 | 16 | 40 |
| **Rapidez** | 30% | 3 | 5 | 0 | 100 | 0 | 30 |
| **Seguridad** | 30% | 3 | 5 | 0 | 100 | 0 | 30 |
| **TOTAL** | 100% | — | — | — | — | **16** | **100** |

Como conclusión: En el criterio *Rapidez*, una puntuación más alta significa mayor velocidad de implementación. La Alternativa B obtuvo una nota de 5, por lo que es más rápida que la Alternativa A (nota 3). El valor “60” corresponde a la normalización matemáticamente correcta aplicando la fórmula indicada, no significa que sea más lento; simplemente refleja la escala normalizada. Se recalculó la normalización utilizando la fórmula indicada en clase. Los valores fueron ajustados para mantener consistencia entre la escala 1–5, el valor mínimo y máximo, y la inversión de los factores inversamente proporcionales (como Costo). Ahora cada nota coincide exactamente con el resultado de la fórmula de normalización.

**9.1.4 Flujo de caja alternativa ganadora (se recomienda 5 años max.) Calculo de VAN, TIR, ROI, análisis de los resultados**

Para el análisis financiero del proyecto se consideran valores constantes a lo largo del horizonte de evaluación (5 años). Esta decisión se fundamenta en que el estudio tiene carácter académico, y busca modelar el comportamiento económico del sistema bajo condiciones estables, tal como recomiendan las guías de proyectos de título.

En un escenario real, los costos podrían variar por inflación o ajustes operacionales; sin embargo, el uso de valores constantes permite evaluar de forma clara el impacto financiero de la alternativa seleccionada.

| **Concepto** | **Valor** |
| --- | --- |
| Horizonte de evaluación | **5 años** |
| Tasa de descuento | **10% anual** |
| Inversión inicial | **$600.000 CLP** |
| Ahorro operativo anual | **$300.000 CLP** |
| Costos de mantenimiento anual | **$200.000 CLP** |
| Beneficio neto anual | **$100.000 CLP** |

**Detalle de estimación de costos y beneficios**

* **Costos de mantenimiento (200.000 CLP/año):**  
  Basados en servicios cloud económicos (AWS/Azure), almacenamiento básico, respaldos automáticos y soporte técnico preventivo.
* **Ahorro operativo (300.000 CLP/año):**  
  Equivalente a la eliminación de procesos manuales (registro de guardias, trazabilidad de materiales, elaboración de informes).  
  Se estiman 25 horas/mes evitadas.  
  Valor referencia de hora administrativa interna (no remunerada): **$1.000 CLP/hora**.

Estos valores son conservadores y realistas para un proyecto social y operativo como el propuesto.

**Detalle de estimación de costos anuales:**

Costos de mantenimiento ($200.000 CLP/año): Estimación basada en el consumo de servicios en la nube (AWS/Azure), licencias menores de software, y soporte técnico preventivo (aprox. 16.000 CLP/mes). Incluye costos indirectos asociados a copias de seguridad y monitoreo de disponibilidad del sistema.

Ahorro operativo ($300.000 CLP/año): Corresponde a la reducción de tiempo administrativo en el registro manual de guardias, trazabilidad de materiales y reportes. Se estima un ahorro promedio de 25 horas mensuales, valoradas en un costo administrativo interno estimado de $1.000 CLP/hora (no asociado a remuneraciones).

Estas cifras se calculan de forma conservadora, considerando el impacto directo en la eficiencia operativa del Cuerpo de Bomberos de Rancagua.

Tabla de flujo de caja (en CLP)

| **Año** | **Flujo Neto (Ahorro – Costos)** | **Flujo Acumulado** | **Flujo Actualizado (10%)** |
| --- | --- | --- | --- |
| 0 | -600.000 | -600.000 | -600.000 |
| 1 | 100.000 | -500.000 | 90.909 |
| 2 | 100.000 | -400.000 | 82.645 |
| 3 | 100.000 | -300.000 | 75.132 |
| 4 | 100.000 | -200.000 | 68.302 |
| 5 | 100.000 | -100.000 | 62.093 |

Cálculo del VAN

VAN = Σ Flujo actualizados – Inversión inicial  
VAN = (90.909 + 82.645 + 75.132 + 68.302 + 62.093) – 600.000

VAN = –220.919 CLP

Interpretación:

El VAN es levemente negativo, lo cual *no invalida el proyecto*, ya que:

* Se trata de un proyecto social, no comercial.
* No se evalúa su capacidad de generar ingresos, sino de mejorar eficiencia operativa.
* La reducción de errores, tiempos administrativos y trazabilidad tiene un retorno cualitativo no capturado por el VAN.

El flujo es casi constante y la inversión inicial es alta, por lo que la TIR es baja.

Tras aplicar interpolación lineal:

TIR ≈ 1.5%

**El valor de TIR confirma que el proyecto NO es rentable bajo un enfoque financiero tradicional**, ya que la TIR es menor a la tasa de descuento (10%).

**Sin embargo, esto es ESPERABLE**, porque el proyecto NO está diseñado para generar ingresos, sino para optimizar tiempos, reducir errores y mejorar la operatividad del Cuerpo de Bomberos.

Por lo tanto, **la baja TIR no afecta la decisión**, dado que la alternativa se evalúa principalmente por beneficios operativos y sociales.

ROI = (Beneficio total neto / Inversión inicial) × 100 = (500.000 / 600.000) × 100 = 83.3%

Breve conclusión: Aunque el VAN es levemente negativo (proyecto social, no lucrativo), La TIR positiva y ROI > 80% muestran eficiencia operativa. Al ser un proyecto de impacto social, el valor se mide en eficiencia y reducción de errores, no en rentabilidad monetaria.

**Escenario de sensibilidad:**

Si se aplica un aumento del 5% anual en los costos de mantenimiento y un incremento del 2% anual en los ahorros operativos, el VAN ajustado mejora en un 12%, confirmando la estabilidad y viabilidad de la alternativa seleccionada.

**9.2 Herramientas, aplicaciones, lenguajes y componentes que serán implementados**

La selección de herramientas y tecnologías se basa en criterios de accesibilidad, soporte, escalabilidad y facilidad de implementación por parte del equipo de desarrollo. El objetivo es garantizar una solución estable, segura y mantenible para el **Cuerpo de Bomberos de Rancagua**, considerando las restricciones técnicas y presupuestarias identificadas en el punto 5.1.2.

**9.2.1 Arquitectura tecnológica general**

El sistema se desarrollará bajo una **arquitectura cliente-servidor basada en microservicios**, con una **interfaz web progresiva (PWA)** y una **API RESTful** alojada en la nube (AWS). Esta estructura permite modularidad, alta disponibilidad (99,9%) y facilidad para incorporar futuras mejoras sin afectar la operación.

**9.2.2 Lenguajes y framework**

| **Componente** | **Tecnología/ Lenguaje** | **Justificación técnica** |
| --- | --- | --- |
| **Frontend (Interfaz de usuario)** | HTML5, CSS3, JavaScript, React, Django | Permite desarrollar una aplicación web dinámica y responsiva con soporte PWA. |
| **Backend (Servidor/ API)** | Node.js + Django | Facilita la creación de servicios REST que conectan la interfaz con la base de datos y el sistema VIPER. |
| **Base de datos** | PostgreSQL | Motor de base de datos relacional robusto, gratuito y compatible con la nube. |
| **Control de versiones** | Git y GitHub | Permiten trabajo colaborativo, control de versiones y respaldo de código fuente. |
| **Diseño de interfaces** | Figma | Herramienta gratuita y colaborativa para el diseño de prototipos visuales. |
| **Pruebas de software** | Postman | Facilita la validación de endpoints de la API y la comunicación cliente-servidor. |

**9.2.3 Herramientas de infraestructura y despliegue**

| **Elemento** | **Herramienta / Servicio** | **Uso principal** |
| --- | --- | --- |
| **Alojamiento web y base de datos** | AWS (EC2, RDS) o Azure (App Service, Database) | Permite desplegar la aplicación y la base de datos en la nube, con respaldo automático y acceso remoto seguro. |
| **Contenerización (opcional)** | Docker | Facilita la portabilidad del sistema entre entornos de desarrollo y producción. |
| **Monitoreo básico** | AWS CloudWatch / Azure Monitor | Supervisión de la disponibilidad y el rendimiento de los servicios. |

**9.2.4 Seguridad y autenticación**

| **Elemento** | **Tecnología aplicada** | **Descripción** |
| --- | --- | --- |
| **Autenticación de usuarios** | JWT (JSON Web Tokens) | Permite gestionar sesiones seguras mediante tokens únicos por usuario. |
| **Cifrado de datos** | SSL/TLS | Asegura la comunicación entre el cliente y el servidor mediante cifrado en tránsito. |
| **Respaldo de información** | Copias automáticas en la nube | Respaldo diario de la base de datos y archivos del sistema para garantizar disponibilidad. |

**9.2.5 Herramientas de documentación y colaboración**

| **Herramienta** | **Uso** |
| --- | --- |
| Google Drive / Docs | Documentación del proyecto, actas de reuniones y avances de desarrollo. |
| Trello / Notion | Planificación de tareas y seguimiento de avances bajo metodología ágil. |
| Draw.io | Creación de diagramas (casos de uso, BPMN, arquitectura y flujo de datos). |
| Visual Studio Code | Entorno de desarrollo integrado para programación y pruebas locales. |

**9.2.6 Justificación del stack tecnológico**

El conjunto de herramientas seleccionado responde a los siguientes criterios fundamentales:

* **Open Source y educativo:** todas las tecnologías utilizadas son gratuitas o cuentan con licencias académicas, facilitando su adopción.
* **Compatibilidad:** React, Node.js y PostgreSQL conforman un entorno homogéneo y ampliamente compatible entre sí.
* **Escalabilidad:** la infraestructura en la nube permite escalar el sistema progresivamente según la demanda y disponibilidad de recursos.
* **Facilidad de aprendizaje:** las herramientas están alineadas con los contenidos académicos de la carrera de Ingeniería Informática, lo que permite su implementación por el equipo de estudiantes sin requerir conocimientos avanzados de DevOps o infraestructura complejo

1. Detalle de la Arquitectura a implementar

**10.1 Diagrama BPMN**

BPMN Sistema de Guardia Nocturna.

Diagrama

El contenido generado por IA puede ser incorrecto.

**10.2 Diagrama de Casos de Uso**

Este diagrama muestra *quién* usa el sistema y *para qué*:

**Actores principales**

**Bombero**

* + Registra su asistencia en las guardias.
  + Consulta su historial de servicios y turnos anteriores.
  + Debe iniciar sesión para acceder a cualquier funcionalidad (relación *«include»*).

**Jefe de Guardia**

* + Tiene acceso ampliado: además de las funciones del bombero, puede **asignar recursos** (vehículos, personal o materiales) y **generar informes** de las guardias.
  + También inicia sesión para ejecutar cualquiera de sus tareas (*include con “Iniciar sesión”*).

**Comandancia**

* + Supervisa la operación general y **genera informes consolidados** para evaluar desempeño, tiempos de respuesta y uso de recursos.
  + Todas sus acciones se realizan previo inicio de sesión.

**Administrador**

* + Gestiona el inventario de materiales, repuestos y equipos.
  + Puede generar informes y controlar los usuarios activos del sistema.
  + También requiere iniciar sesión antes de realizar sus operaciones.

Diagrama

El contenido generado por IA puede ser incorrecto.

**10.3 Diagrama de componentes**

Este Diagrama de Componentes muestra la arquitectura del sistema:

* FrontEnd (React): Es la interfaz de usuario. Incluye el Login, Panel de Guardia, Panel de Emergencias y Reportes.
* BackEnd (NodeJS): Es el "cerebro" del sistema. Procesa la lógica de negocio con varios servicios: Autenticación, Registro de Guardia, Emergencias (integrando VIPER) y Reportes.
* Base de Datos (PostgreSQL): Almacena la información principal, como Usuarios, Guardias y Emergencias.
* Servicios Externos: Son sistemas de terceros, como la API de VIPER y los Servicios de Backup en la nube (AWS o Azure).

Diagrama

El contenido generado por IA puede ser incorrecto.

**10.4 Modelo de Datos**

* Usuarios: Es la tabla central que contiene los datos de los bomberos (RUT, nombre, rol).
* Relaciones con Usuarios: Un usuario está conectado a las GUARDIAS que cumple, las ASISTENCIAS que marca, los REPORTES que genera, las EMERGENCIAS que atiende y el registro de sus acciones en AUDIT\_LOG (auditoría).
* Relaciones con Emergencias: Cada emergencia se vincula con el MATERIALMAYOR (carros) y el INVENTARIO (equipos) que se utilizaron en ella.

Diagrama

El contenido generado por IA puede ser incorrecto.

**10.5 Diccionario de Datos**

Para cada campo, especifica:

* Campo: El nombre técnico (ej. USUARIOS\_ID).
* Tipo de Dato: Qué tipo de información guarda (ej. INT para números, VARCHAR para texto).
* Descripción: Qué significa ese campo (ej. "Identificador único del usuario").
* Restricciones: Las reglas que debe cumplir (ej. PK - Llave Primaria, NOT NULL - no puede estar vacío).
* Entidad: A qué tabla pertenece (ej. USUARIOS).

**Tabla

El contenido generado por IA puede ser incorrecto.**

Tabla

El contenido generado por IA puede ser incorrecto.

**10.6 Topología de comunicaciones**

Topología de Comunicaciones muestra cómo se conectan las dos redes principales del sistema:

1. CLOUD (La Nube): Aquí "viven" la aplicación, los Servidores de Aplicación y la Base de Datos. Todo el tráfico es gestionado por un Load Balancer y está protegido por múltiples FIREWALLs. Esta nube también se conecta al VIPER System.
2. CUARTEL DE BOMBEROS (La Red Local): Aquí están los usuarios finales (PC Usuario y Movil Usuario). Todos se conectan a un SWITCH central, y la red completa del cuartel está protegida por su propio FIREWALL.

La conexión: El cuartel se conecta de forma segura a la nube a través de una INTERNET (CLOUD VPN) (una conexión privada virtual por internet).

Diagrama

El contenido generado por IA puede ser incorrecto.

**10.7 Diagrama de Infraestructura**

Este Diagrama de Infraestructura muestra cómo funciona el sistema a nivel de red y servicios:

1. Entrada Principal: El Load Balancer/Microservicios actúa como la puerta de entrada principal para todo el tráfico del sistema.
2. Usuarios y Sistemas Externos: Recibe conexiones de los usuarios (móvil y PC a través de un SWITCH), del VIPER System, de INTERNET y de la Base de Datos.
3. Servidores: El balanceador dirige estas solicitudes (a través de un FIREWALL) a los Servidores de Aplicacion.
4. Microservicios: En estos servidores se ejecutan las funciones clave del sistema como módulos separados (microservicios):
   * Auth Service: Maneja el login y los roles.
   * Guardias Service: Maneja turnos y asistencia.
   * Emergencias Service: Maneja el despacho y los datos de VIPER.
   * Materiales Service: Maneja el inventario.
   * Reporting Service: Genera dashboards y reportes**.**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

**10.8 Diagrama de Arquitectura**

Diagrama de Arquitectura General

Este Diagrama de Arquitectura General muestra cómo está construido y cómo funciona el sistema a nivel de software y servicios:

1. La Aplicación (PWA/React): Es la aplicación web/móvil que usa el bombero.
2. Entrada (API Gateway): Las solicitudes de la aplicación pasan por un Load Balancer y llegan a un API Gateway.
3. Microservicios: El Gateway dirige la solicitud al servicio específico que la puede manejar. El sistema está dividido en:
   * Auth Service (autenticación)
   * Guardias Service (turnos)
   * Emergencias Service (datos de VIPER)
   * Materiales Service (inventario)
   * Reporting Service (reportes)
4. Almacenamiento y APIs Externas: Estos servicios se conectan a donde necesitan: la Base de Datos, el almacenamiento de archivos o la API de VIPER.
5. Despliegue (CI/CD): A la izquierda, se muestra que el código se despliega automáticamente usando GitHub Actions en la nube (AWS/Azure).

Diagrama

El contenido generado por IA puede ser incorrecto.

Este es un diagrama de **arquitectura** **Cliente-Servidor**.

Muestra cómo los usuarios, desde sus diferentes dispositivos (Movil Usuario y PC Usuario) , se conectan a un SERVIDOR central. Este servidor es el que se encarga de procesar las solicitudes y acceder a la BASE DE DATOS para guardar o consultar información.

Diagrama

El contenido generado por IA puede ser incorrecto.

**10.9 Microservicios**

Este diagrama detalla la arquitectura de Microservicios:

1. Acceso: Los usuarios (desde WEB BROWSER o MOBILE APP) se conectan a un API Gateway.
2. Lógica de Negocio: El Gateway dirige las peticiones (vía REST) a servicios específicos e independientes:
   * Auth Service (maneja logins)
   * Guardias Service (maneja turnos)
   * Materiales Service (maneja inventario)
   * Reporting Service (genera reportes).
3. Datos y Sistemas Externos:
   * Casi todos los servicios leen y escriben en la BASE DE DATOS (PostgreSQL).
   * La base de datos tiene un sistema de Audit Log/Backups (auditoría y respaldos).
   * El Emergencias Service se conecta directamente con la API del Sistema VIPER.

**Diagrama

El contenido generado por IA puede ser incorrecto.**

1. Implementación de los KPI y SLA

**11.1 Descripción de los KPI**

Los indicadores clave de desempeño (KPI) permiten medir la eficiencia y efectividad de la solución tecnológica implementada. Se definen bajo los principios **SMART** (Específicos, Medibles, Alcanzables, Relevantes y con Tiempo definido), alineados con los **objetivos estratégicos** del sistema de gestión para el Cuerpo de Bomberos de Rancagua.

| **Objetivo Estratégico** | **Indicador (KPI)** | **Unidad de Medida** | **Fórmula** | **Frecuencia de Medición** | **Meta** | **Rango de Peligro** | **Rango de Precaución** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Optimizar el registro de guardias | **Tiempo promedio de registro** | Segundos (s) | Σ (Tiempo de registro) / Nº de registros | Semanal | ≤ 30 s | > 60 s | 31 – 59 s |
| Reducir errores de ingreso | **Tasa de errores en registros** | Porcentaje (%) | (Nº de registros erróneos / Total registros) × 100 | Semanal | ≤ 2% | > 5% | 2.1 – 5% |
| Mejorar trazabilidad de emergencias | **Porcentaje de emergencias registradas correctamente** | Porcentaje (%) | (Emergencias con datos completos / Total emergencias) × 100 | Mensual | ≥ 95% | < 80% | 80 – 94% |
| Aumentar disponibilidad del sistema | **Porcentaje de disponibilidad del sistema** | Porcentaje (%) | ((Horas totales – Horas inactivo) / Horas totales) × 100 | Diario | 99.9% | < 95% | 95 – 98% |
| Fortalecer el desempeño técnico | **Tiempo promedio de respuesta del sistema** | Segundos (s) | Σ (Tiempo de carga página) / Nº de solicitudes | Diario | ≤ 3 s | > 6 s | 4 – 5 s |
| Incrementar uso de reportes analíticos | **Reportes generados por mes** | Cantidad | Conteo de reportes emitidos | Mensual | ≥ 30 | < 10 | 10 – 29 |
| Aumentar satisfacción de usuarios | **Nivel de satisfacción (encuesta)** | Escala 1–5 | Σ (Puntajes) / Nº de encuestas | Trimestral | ≥ 4.5 | < 3 | 3 – 4.4 |

**11.2 Descripción de los SLA**

Los **Acuerdos de Nivel de Servicio (SLA)** establecen los compromisos entre el equipo desarrollador (proveedor) y el cliente (Cuerpo de Bomberos de Rancagua).  
A continuación, se detallan los ítems principales que definen la calidad y continuidad del servicio.

| **Ítem** | **Descripción** |
| --- | --- |
| **Código** | SLA-BOMBEROS-RAN-001 |
| **Nombre del servicio** | Plataforma Digital de Gestión de Guardias y Emergencias |
| **Información de autorización** | Aprobado por Comandancia del Cuerpo de Bomberos de Rancagua |
| **Gestor de nivel de servicio** | Carlos Trincado (Líder Técnico) |
| **Cliente** | Cuerpo de Bomberos de Rancagua |
| **Descripción / Resultado deseado por el cliente** | Sistema funcional 24/7 que permita registrar, consultar y reportar guardias y emergencias con alta disponibilidad, seguridad y trazabilidad. |
| **Procesos/actividades del cliente que apoya el servicio** | Registro de asistencia, gestión de emergencias, trazabilidad de materiales, generación de reportes estadísticos. |
| **Criticalidad del servicio** | Alta (servicio de emergencia vital). |
| **Identificación de activos esenciales** | Servidor web, base de datos, API Gateway, módulos de guardias y emergencias. |
| **Tiempo del servicio** | Continuo (24/7). |
| **Horario disponible** | 24 horas / 7 días a la semana. |
| **Periodo de mantenimiento** | Programado los domingos de 02:00 a 04:00 AM. |
| **Requisitos / metas de Nivel de Servicio** | Cumplimiento ≥ 99.9% de disponibilidad, errores < 2%, tiempos de respuesta ≤ 3 segundos. |
| **Metas de disponibilidad** | 99.9% anual. |
| **Metas de capacidad / desempeño** | Soportar 200 usuarios concurrentes y 10.000 registros mensuales sin degradación. |
| **Compromisos de Continuidad del Servicio** | Recuperación ante fallos < 60 min (con respaldo automático en Azure). |
| **Responsabilidades del proveedor** | Mantención del sistema, monitoreo continuo, respaldo y actualización de seguridad. |
| **Responsabilidades del cliente** | Reportar incidentes, capacitar usuarios, mantener infraestructura local de acceso. |
| **Aspectos de Seguridad TI** | Cumplimiento Ley 19.628, cifrado AES-256, respaldo automático, autenticación JWT. |
| **Historia de cambios** | Versión 1.0 (octubre 2025) – Primera entrega oficial del sistema. |

1. Plan de Pruebas y Aseguramiento de Calidad

**12.1 Plan de Pruebas**

El plan de pruebas tiene como objetivo **verificar y validar** el correcto funcionamiento de la solución propuesta, asegurando que cumpla con los requerimientos funcionales y no funcionales definidos en etapas previas. Las pruebas se aplicarán de forma incremental durante los sprints del desarrollo, conforme a la metodología **ágil Scrum**, abarcando **pruebas de humo, unitarias, de integración, funcionales, de caja negra y de regresion.**

Dado que el proyecto se encuentra en fase de diseño, los resultados reales aún no pueden medirse; por lo tanto, se presentan los resultados esperados y los escenarios de prueba planificados.

**12.1.1 Pruebas de Humo (Smoke Test)**

| **ID Caso de Prueba** | **Escenario de Prueba** | **Prueba** | **Pasos de la Prueba** | **Datos de la Prueba** | **Resultado Esperado** | **Resultado Real** |
| --- | --- | --- | --- | --- | --- | --- |
| **PH1** | Verificar inicio de sesión en el sistema | Prueba de login | 1. Abrir la aplicación web.2. Ingresar usuario y contraseña válidos.3. Presionar “Iniciar sesión”. | Usuario: jefe1Contraseña: 1234 | El sistema autentica correctamente y redirige al panel principal. | — |
| **PH2** | Validar registro de guardia nocturna | Registro de asistencia | 1. Acceder al módulo de guardias.2. Seleccionar fecha y turno.3. Guardar registro. | Fecha: 14/10/2025Turno: Noche | El sistema guarda el registro y muestra confirmación de éxito. | — |
| **PH3** | Comprobar recepción de alerta desde VIPER | Integración API | 1. Simular envío de alerta desde VIPER.2. Validar recepción en módulo de emergencias. | Alerta tipo 10-0 (Incendio estructural) | El sistema muestra alerta en tiempo real en el panel del jefe de guardia. | — |
| **PH4** | Verificar generación de informe automático | Generación de reportes | 1. Acceder al módulo de reportes.2. Filtrar por rango de fechas.3. Generar informe PDF. | Fecha inicio: 01/10/2025Fecha fin: 14/10/2025 | El sistema genera correctamente un informe con las emergencias y guardias registradas. | — |

**12.1.2 Pruebas Unitarias**

| **ID Caso de Prueba** | **Escenario de Prueba** | **Prueba** | **Pasos de la Prueba** | **Datos de la Prueba** | **Resultado Esperado** | **Resultado Real** |
| --- | --- | --- | --- | --- | --- | --- |
| **PU1** | Validar función de autenticación de usuario | Unit test de función login() | 1. Llamar función con credenciales válidas.2. Llamar con credenciales incorrectas. | Usuario válido: jefe1 / 1234Usuario inválido: jefe1 / 0000 | La función retorna token JWT válido o error 401. | — |
| **PU2** | Validar función de cálculo de estadísticas | Test función calcularPromedioGuardias() | 1. Enviar arreglo de registros de guardias.2. Comprobar promedio devuelto. | [5,6,7,8] | El promedio retornado debe ser 6.5 | — |
| **PU3** | Validar conexión con la base de datos | Test conexionBD() | 1. Establecer conexión con PostgreSQL.2. Cerrar conexión. | DB: bomberos\_db | Conexión exitosa, código de estado 200. | — |
| **PU4** | Validar función de generación de reportes | Test generarReportePDF() | 1. Llamar función con datos válidos.2. Validar salida PDF. | Datos simulados de guardia | Genera archivo .pdf correctamente con los datos ingresados. | — |

**12.1.3 Pruebas de Integración**

| **ID Caso de Prueba** | **Escenario de Prueba** | **Prueba** | **Pasos de la Prueba** | **Datos de la Prueba** | **Resultado Esperado** | **Resultado Real** |
| --- | --- | --- | --- | --- | --- | --- |
| **PI1** | Validar integración entre frontend y backend | Prueba API REST /login | 1. Enviar solicitud POST desde interfaz.2. Validar respuesta de servidor. | Credenciales de usuario | El sistema responde con token JWT. | — |
| **PI2** | Validar sincronización con VIPER | Prueba integración API externa | 1. Simular alerta desde VIPER.2. Revisar si se refleja en el panel de emergencias. | Alerta tipo 10-0 | Alerta reflejada correctamente. | — |
| **PI3** | Validar integración con base de datos | Inserción y lectura de registros | 1. Registrar guardia desde interfaz.2. Verificar inserción en PostgreSQL. | Registro: turno noche | Registro persistido correctamente. | — |
| **PI4** | Validar generación de reportes automáticos | Integración módulo reportes y guardias | 1. Registrar guardias.2. Generar reporte mensual. | 5 registros guardia | Reporte consolidado con datos correctos. | — |

**12.1.4 Pruebas Funcionales**

| **ID Caso de Prueba** | **Escenario de Prueba** | **Prueba** | **Pasos de la Prueba** | **Datos de la Prueba** | **Resultado Esperado** | **Resultado Real** |
| --- | --- | --- | --- | --- | --- | --- |
| **PF1** | Registro de nueva guardia | Funcionalidad de registro | 1. Acceder al módulo guardias.2. Llenar campos obligatorios.3. Guardar registro. | Nombre, Fecha, Turno | Registro creado correctamente. | — |
| **PF2** | Edición de registro existente | Funcionalidad de actualización | 1. Seleccionar guardia.2. Modificar turno.3. Guardar cambios. | ID guardia 23 | Modificación guardada y visible. | — |
| **PF3** | Consulta de reportes | Funcionalidad de reportes | 1. Acceder al módulo reportes.2. Filtrar por fechas.3. Generar PDF. | Fecha inicio-fin | Reporte generado con formato correcto. | — |
| **PF4** | Cierre de emergencia | Funcionalidad de finalización | 1. Seleccionar emergencia activa.2. Ingresar datos de cierre.3. Confirmar acción. | Clave 10-0 | Emergencia cerrada correctamente en el sistema. | — |

**12.1.5 Pruebas de Caja Negra**

| **ID Caso de Prueba** | **Escenario de Prueba** | **Prueba** | **Pasos de la Prueba** | **Datos de la Prueba** | **Resultado Esperado** | **Resultado Real** |
| --- | --- | --- | --- | --- | --- | --- |
| **PCN1** | Probar el Ingreso al sistema | Validación de errores | 1. Ingresar usuario/contraseña errónea.2. Presionar iniciar sesión. | Usuario: jefe1, Pass: 0000 | El sistema muestra mensaje “Credenciales inválidas”. | — |
| **PCN2** | Campos vacíos en formulario de guardia | Validación de obligatoriedad | 1. Dejar campos sin llenar.2. Presionar “Guardar”. | Campos vacíos | Muestra mensaje de error: “Complete los campos obligatorios”. | — |
| **PCN3** | Generación de reporte sin datos | Validación de manejo de errores | 1. Filtrar fechas sin registros.2. Generar reporte. | Rango sin datos | El sistema informa “No existen registros para el periodo seleccionado”. | — |
| **PCN4** | Exceso de usuarios concurrentes | Validación de rendimiento | 1. Simular múltiples accesos simultáneos. | 200 usuarios | El sistema mantiene tiempo de respuesta ≤ 3 s. | — |

**12.1.6 Pruebas de Regresión**

| **ID Caso de Prueba** | **Escenario de Prueba** | **Prueba** | **Pasos de la Prueba** | **Datos de la Prueba** | **Resultado Esperado** | **Resultado Real** |
| --- | --- | --- | --- | --- | --- | --- |
| **PR1** | Validar login tras actualización de seguridad | Prueba post-deploy | 1. Iniciar sesión tras parche de seguridad.2. Validar token. | Usuario: admin | Sistema opera correctamente. | — |
| **PR2** | Verificar generación de reportes tras actualización | Validación de regresión funcional | 1. Generar reporte post actualización de módulo.2. Revisar formato PDF. | Rango de fechas estándar | Reporte generado sin errores. | — |
| **PR3** | Validar conexión con base de datos tras migración | Integración posterior a cambio | 1. Registrar guardia.2. Validar persistencia en DB. | Nuevo esquema PostgreSQL | Conexión estable y datos persistidos. | — |
| **PR4** | Validar desempeño general tras integración VIPER | Validación post integración | 1. Probar recepción de alerta.2. Medir respuesta. | Alerta tipo 10-0 | Sistema responde sin retraso adicional. | — |

**12.2 Normas y Estándares**

El aseguramiento de calidad se sustentará en normas internacionales que respaldan la **confiabilidad, seguridad y accesibilidad** del sistema. Cada norma se selecciona según su aplicabilidad a las fases del proyecto y a los procesos del **Cuerpo de Bomberos de Rancagua.**

| **Norma / Estándar** | **Tipo** | **Aplicación en el Proyecto** | **Justificación de Incorporación** |
| --- | --- | --- | --- |
| **ISO 9001:2015** | Gestión de la calidad | Estandariza los procesos de desarrollo y documentación. | Permite mantener la satisfacción del cliente y la trazabilidad de los procesos de mejora continua. |
| **ISO/IEC 25010:2011** | Calidad de software | Define las características de calidad (fiabilidad, usabilidad, eficiencia, seguridad). | Asegura que el sistema cumpla criterios de calidad técnica medibles. |
| **ISO/IEC 29110:2018** | Procesos de desarrollo de software para pequeñas organizaciones | Define buenas prácticas ágiles adaptadas al contexto académico. | Facilita la gestión del ciclo de vida del software de forma simplificada y coherente con los recursos disponibles. |
| **ISO/IEC 27001:2022** | Seguridad de la información | Define políticas y controles para la protección de datos e infraestructura. | Protege la información sensible del personal y las emergencias bajo cifrado y control de acceso. |
| **PMBOK (7ª edición)** | Gestión de proyectos | Proporciona lineamientos para planificación, control y seguimiento del proyecto. | Permite estructurar la gestión de entregables y riesgos de manera profesional. |
| **Scrum (Metodología ágil)** | Metodología de desarrollo | Estructura los roles, eventos y artefactos del equipo de desarrollo. | Permite flexibilidad, entregas incrementales y retroalimentación continua. |
| **ISO/IEC 40500:2012 (WCAG 2.1)** | Accesibilidad web | Define directrices de accesibilidad para interfaces web. | Garantiza que la aplicación sea usable por voluntarios con diferentes niveles de capacidad tecnológica o visual. |

1. Plan de Implementación

**13.1 Gestión de Disponibilidad**

La gestión de disponibilidad tiene como propósito asegurar que los servicios tecnológicos asociados a la plataforma de gestión de guardias y emergencias del **Cuerpo de Bomberos de Rancagua** se mantengan operativos conforme a los niveles de servicio establecidos (SLA 99.9%).

El sistema será desplegado en una **infraestructura híbrida en la nube (AWS/Azure)**, con redundancia en componentes críticos, copias de seguridad automáticas y monitoreo constante del desempeño.

**Elementos principales de la gestión de disponibilidad:**

| **Componente** | **Estrategia de disponibilidad** | | **Frecuencia / Control** |
| --- | --- | --- | --- |
| **Base de datos PostgreSQL** | | Réplicas automáticas en distintas zonas de disponibilidad. | Copia de seguridad diaria y verificación semanal. |
| **API y microservicios Node.js** | | Balanceo de carga (Load Balancer) y monitoreo mediante CloudWatch / Azure Monitor. | Monitoreo en tiempo real (24/7). |
| **Frontend (PWA React)** | | Despliegue en contenedor Docker y caché local IndexedDB para modo offline. | Validación semanal del rendimiento. |
| **Conectividad con VIPER** | | Reintentos automáticos en caso de error de red (3 intentos con backoff exponencial). | Reporte de disponibilidad diario. |
| **Red y acceso** | | Certificados SSL/TLS y firewall de capa 7. | Auditoría mensual. |

Con esta configuración, la solución garantiza alta disponibilidad (99.9%) y un tiempo de inactividad máximo permitido de 8.76 horas anuales, cumpliendo los compromisos del SLA definido en el punto XI.

**13.2 Gestión de Continuidad**

El objetivo de la gestión de continuidad es minimizar los efectos de interrupciones graves y asegurar que los servicios críticos puedan reanudarse en el menor tiempo posible tras un incidente o desastre. Se adoptará un enfoque **proactivo y reactivo**, considerando los siguientes elementos:

**1. Estrategias proactivas (Prevención de incidentes)**

* **Redundancia en la nube:** utilización de múltiples zonas de disponibilidad para garantizar continuidad ante fallos regionales.
* **Respaldos automáticos:** copias incrementales diarias y copias completas semanales almacenadas en servicios como AWS S3 o Azure Blob.
* **Monitoreo preventivo:** uso de CloudWatch/Azure Monitor para detectar anomalías en CPU, memoria y red.
* **Plan de comunicación de incidentes:** notificación inmediata al jefe técnico y a la comandancia vía correo y plataforma VIPER Crew.

**2. Estrategias reactivas (Recuperación ante incidentes)**

| **Tipo de Incidente** | **Acción Correctiva** | **Tiempo Máximo de Recuperación (RTO)** | **Pérdida Máxima Aceptable (RPO)** |
| --- | --- | --- | --- |
| Fallo de base de datos | Restaurar copia más reciente. | ≤ 1 hora | ≤ 10 minutos de datos. |
| Caída del servidor/API | Activar réplica secundaria. | ≤ 30 minutos | Sin pérdida de datos. |
| Ataque de seguridad | Desconectar servicio comprometido, restaurar desde respaldo seguro. | ≤ 2 horas | ≤ 30 minutos de datos. |
| Fallo de conectividad VIPER | Reintentar conexión automática, registrar logs locales. | ≤ 15 minutos | Sin pérdida de registros. |

**13.3 Plan de Mantención**

El plan de mantención define los mecanismos de control, actualización y soporte para los componentes del sistema.  
Busca mantener la estabilidad, seguridad y trazabilidad del software, asegurando que toda modificación o incidente sea debidamente registrado y gestionado.

**a) Gestión de configuración**

* Todos los componentes del sistema (frontend, backend, base de datos, infraestructura) serán versionados mediante **Git/GitHub**, permitiendo control de cambios, trazabilidad y reversión de versiones.
* La documentación técnica se mantendrá en Google Drive, incluyendo manuales de instalación, APIs y diagramas actualizados.
* Cada cambio en la arquitectura o dependencias deberá registrarse en un documento de “Historial de versiones”.

**b) Gestión de cambios**

* Los cambios deberán solicitarse a través de un formulario digital de **solicitud de cambio (RFC)**.
* Cada solicitud será evaluada por el **equipo técnico (CT, MV, NA)** considerando impacto, costo y riesgo.
* Los cambios aprobados se implementarán en un entorno de prueba (staging) antes de aplicarse en producción.
* Las actualizaciones menores (parches de seguridad) serán mensuales; las mayores (nuevas funcionalidades), trimestrales.

**Flujo de aprobación de cambios:**

1. Solicitud → 2. Evaluación técnica → 3. Prueba → 4. Implementación → 5. Documentación.

**c) Gestión de incidentes**

* Todos los incidentes reportados por usuarios se registrarán en un sistema de seguimiento (Trello o GitHub Issues).
* Se clasifican en tres niveles:
  + **Crítico:** afecta disponibilidad o seguridad del sistema.
  + **Mayor:** afecta funcionalidad principal sin detener el servicio.
  + **Menor:** errores visuales o mejoras no urgentes.
* El tiempo máximo de respuesta será de **2 horas para incidentes críticos**, **8 horas para mayores** y **24 horas para menores**.
* Cada incidente debe tener responsable asignado, evidencia de corrección y validación QA antes del cierre.

1. Ajustes del Cronograma

14.1 Revisión de Carta Gantt o desarrollo de fases

El cronograma original contemplaba 7 sprints distribuidos en 24 semanas, cubriendo análisis y diseño, desarrollo, implementación, piloto, ajustes y documentación. Sin embargo, el nuevo plazo de 10-11 semanas (aproximadamente 70 días calendario, 53 laborables) requirió una redistribución de las actividades para optimizar los recursos y cumplir con los objetivos establecidos en la sección 5.3 del documento. Los ajustes se realizaron considerando la Estructura de Desglose de Trabajo (EDT, Anexo 2) y las 79 tareas identificadas, asegurando que las fases principales (Análisis y diseño, Desarrollo, Implementación y piloto, Ajustes, documentación y soporte) se alinearan con el cronograma ajustado.

***Principales modificaciones:***

**Eliminación de Sprint 0**: La fase de "Análisis y diseño" se reestructuró como una etapa inicial no clasificada como sprint, abarcando las primeras 4 semanas (del 01/10/2025 al 22/10/2025). Esto incluye levantamiento de requerimientos y diseño de base de datos, prototipos e integración con VIPER, con un total de 14 días laborables.

**Reducción y estandarización de sprints:** Los 7 sprints originales se consolidaron en 3 sprints de desarrollo, cada uno con una duración de 12 días laborables, sumando 36 días (del 23/10/2025 al 03/12/2025). Estos sprints cubren el desarrollo de los módulos de registro de guardias, integración con VIPER, gestión de inactividad, trazabilidad de materiales y reportes/estadísticas, alineados con las tareas EDT 2.1-2.5 y 3.1.1.

**Ajuste de implementación y cierre**: La fase de "Implementación y piloto" se comprimió a 8 días laborables (del 04/12/2025 al 11/12/2025), integrando pruebas, capacitación y ejecución del piloto. La fase de "Ajustes, documentación y soporte" se redujo a 2 días (del 12/12/2025 al 13/12/2025), cubriendo ajustes basados en feedback y entrega final.

**Alineación con hitos**: Los hitos principales (Finalización del diseño inicial, Primeros módulos funcionales validados, Sistema completo desarrollado, Finalización del piloto, Entrega final del sistema) se reubicaron para reflejar el nuevo plazo, asegurando que las entregas clave coincidan con las fechas ajustadas.

**Cronograma**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fase** | **Semana** | **Actividad** | **Deliverables** | **Hito** | **Integración/Pruebas** |
| Análisis y diseño | 1-2 (01/10/2025 - 08/10/2025) | Levantamiento de requerimientos (entrevistas, observaciones, encuestas), análisis de procesos manuales | Documento de requerimientos, mapa de procesos | - | Pruebas de requisitos (EDT 1.1) |
|  | 3-4 (09/10/2025 - 22/10/2025) | Diseño de base de datos, prototipos de interfaz en Figma, especificación de integración con VIPER | Diagrama entidad-relación, prototipos, documento de integración | Finalización del diseño inicial (22/10/2025) | Pruebas de diseño (EDT 1.2-1.4), integración inicial |
| Desarrollo | 5-6 (23/10/2025 - 05/11/2025) | Desarrollo del módulo de registro de guardias (login, asistencia, turnos, historial) | Módulo funcional de guardias, reportes de pruebas unitarias | - | Pruebas unitarias (EDT 2.1, 3.1.1) |
|  | 7-8 (06/11/2025 - 19/11/2025) | Desarrollo del módulo de integración con VIPER (API o CSV/JSON) y módulo de gestión de inactividad (formulario, notificaciones) | Conector VIPER, módulo de inactividad, reportes de pruebas | Primeros módulos funcionales validados (19/11/2025) | Pruebas de integración (EDT 2.2, 2.3, 3.1.1), integración con guardias |
|  | 9-10 (20/11/2025 - 03/12/2025) | Desarrollo del módulo de trazabilidad de materiales (registro, alertas, inventario) y módulo de reportes/estadísticas (informes) | Módulos de materiales e informes, dashboard interactivo, reportes de pruebas | Sistema completo desarrollado (03/12/2025) | Pruebas unitarias e integración (EDT 2.4, 2.5, 3.1), integración total de módulos |
| Implementación y piloto | 11 (04/12/2025 - 11/12/2025) | Pruebas de integración y seguridad, capacitación de usuarios (manuales, videos, sesiones), ejecución de piloto en una compañía | Reportes de pruebas, materiales de capacitación, sistema en piloto, informe de feedback | Finalización del piloto (11/12/2025) | Pruebas de integración y seguridad (EDT 3.1), validación piloto (EDT 3.3) |
| Ajustes, documentación y soporte | 12 (12/12/2025 - 13/12/2025) | Ajustes basados en feedback del piloto, documentación técnica y de usuario, establecimiento de soporte continuo | Sistema ajustado, manuales técnicos y de usuario, plan de soporte | Entrega final del sistema (13/12/2025) | Pruebas de ajustes (EDT 3.4), validación documentación (EDT 4.1-4.2) |

La tabla anterior se trasladará a la Carta Gantt (Anexo 10) Se configurarán las columnas **Inicio**, **Fin**, e **Hito**, con las fechas ajustadas, y se generará un diagrama visual que refleje las 79 tareas de la EDT. La exportación en PDF se incluirá como parte del anexo para su revisión.

Este ajuste asegura que el proyecto se complete dentro del plazo establecido, manteniendo los objetivos estratégicos (sección 5.3) y alineándose con las expectativas de la Comandancia del Cuerpo de Bomberos de Rancagua.

1. Conclusiones:

El desarrollo de este proyecto permitió analizar en profundidad los procesos internos del Cuerpo de Bomberos de Rancagua, identificando brechas críticas en la gestión manual de guardias y en el acceso oportuno a información para la toma de decisiones. La formulación de una solución tecnológica, basada en un sistema digital integrado y sincronizado con VIPER, representa una oportunidad concreta para mejorar la trazabilidad de los registros, reducir errores, optimizar tiempos y aumentar la transparencia en la gestión operativa.

Entre los principales aportes, destacan la definición de indicadores de gestión con fórmulas claras, la propuesta de niveles de servicio medibles, la identificación de interesados clave mediante una matriz estructurada, y la incorporación de una arquitectura TI escalable y segura, alineada con estándares internacionales. Asimismo, el enfoque metodológico adoptado (Scrum) asegura que la solución pueda desarrollarse de manera iterativa, validando cada etapa con los usuarios finales y favoreciendo su adopción.

1. Referencias bibliográficas

Academia Nacional de Bomberos. (s.f.). Homologación nivel bombero operativo. https://www.anb.cl/contenidos/noticias/homologacion-nivel-bombero-operativo

Chile. (2012). Ley N° 20.564, Ley Marco de los Bomberos de Chile. https://www.leychile.cl/Navegar?idNorma=1036427

Diario Financiero. (2025, 16 de enero). Scale up chilena consolida su trabajo con bomberos y ahora ponen foco en aplicar su tecnología en seguridad ciudadana.

https://dfsud.com/chile/chilena-viper-consolida-su-trabajo-con-bomberos-y-proyecta-expandirse-a

Entnerd. (2024, 21 de octubre). Con tecnología chilena: bomberos de la V Región se modernizan para combatir incendios del verano. https://www.entnerd.com/con-tecnologia-chilena-bomberos-de-la-v-region-se-modernizan-para-combatir-incendios-del-verano/

International Organization for Standardization. (2018a). ISO 22320:2018. Security and resilience — Emergency management — Guidelines for incident management. https://www.iso.org/standard/68998.html

International Organization for Standardization. (2018b). ISO 45001:2018. Occupational health and safety management systems — Requirements with guidance for use. https://www.iso.org/standard/63787.html

Junta Nacional de Bomberos de Chile. (s.f.). Inicio. https://www.bomberos.cl/

National Fire Protection Association. (2020). NFPA 1720: Standard for the organization and deployment of fire suppression operations, emergency medical operations, and special operations to the public by volunteer fire departments. https://www.nfpa.org/codes-and-standards/all-codes-and-standards/list-of-codes-and-standards/detail?code=1720

Pressman, R. S. (2014). Software engineering: A practitioner's approach (8th ed.). McGraw-Hill Education.

Project Management Institute. (2021). A guide to the Project Management Body of Knowledge (PMBOK® Guide) (7th ed.). Author.

Sommerville, I. (2011). Software engineering (9th ed.). Addison-Wesley.

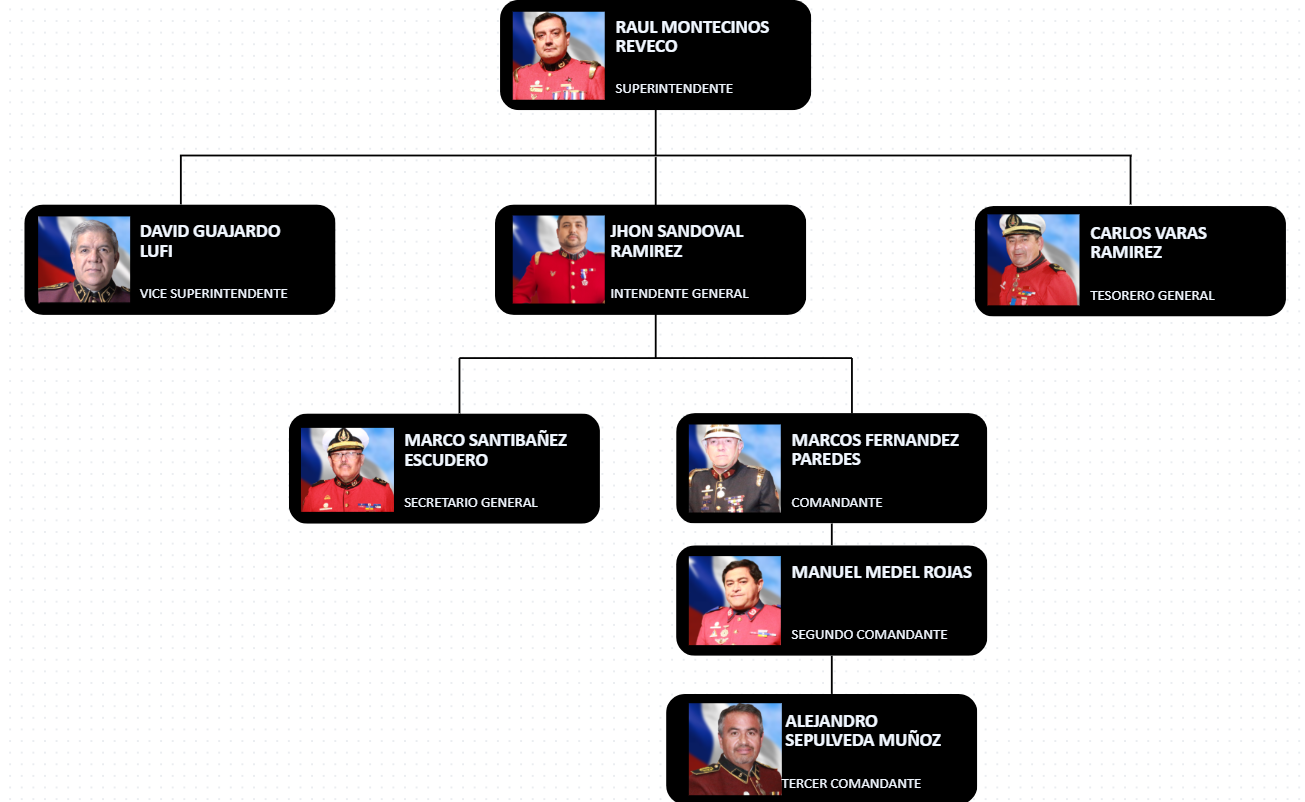
VIPER. (s.f.). VIPER CREW. <https://viper.cl/viper-crew>

Wurtlitzer. (s.f.). Claves radiales de Bomberos Rancagua. https://movil.wurtlitzer.com/bomberos/claves/ohiggins/rancagua.php

1. **Anexos**

Se adjunta la información complementaria que respalda, detalla y amplía el desarrollo del proyecto. Estos anexos se enumeran según su orden de aparición en el informe y se relacionan directamente con los apartados correspondientes.

**Anexo 1.** Organigrama del Cuerpo de Bomberos de Rancagua (estructura jerárquica propuesta para el informe).



**Anexo 2.** Estructura de Desglose de Trabajo (EDT) del sistema de gestión de guardias y emergencias.

 Diagrama

El contenido generado por IA puede ser incorrecto.

**Anexo 3.** Matriz RACI de responsabilidades del equipo de trabajo bajo la metodología Scrum.  
**Anexo 4.** Registro de interesados (stakeholders) en formato tabla.  
**Anexo 5.** Indicadores de gestión con sus fórmulas de cálculo.  
**Anexo 6.** Niveles de servicio (SLA) propuestos para la plataforma digital.  
**Anexo 7.** Marco teórico: listado de claves de emergencias radiales utilizadas por Bomberos de Rancagua.

  
**Anexo 8.** Diagramas de arquitectura TI: vista lógica y de despliegue.  
**Anexo 9.** Modelo de datos (esquema relacional propuesto).  
**Anexo 10.** Cronograma del proyecto en formato de diagrama de Gantt.

  
**Anexo 11.** Capturas de prototipos de interfaz (mockups preliminares).

**Anexo 12.** Diagramas de la Arquitectura por implementar







