



Implementación de Plataforma de monitoreo para respiradores industriales

Memoria presentada para la obtención del título de Ingeniero Civil Informático

POR

MATIAS NORAMBUENA FUENTEALBA

Profesores patrocinantes: Pedro Pinacho Davidson
Pablo Aqueveque Navarro

Profesional supervisor: Johannes Bunster Darras

Agradecimientos

Quiero expresar mi más profundo agradecimiento a mi familia, por estar siempre a mi lado, brindándome su apoyo incondicional incluso en los momentos más difíciles. Su compañía, cariño y confianza fueron fundamentales para que pudiera completar esta etapa de mi vida.

A mi abuelita, que, aunque no esté físicamente, siempre me ha acompañado con su amor y su recuerdo presente en cada paso que doy. Su ejemplo de fortaleza y ternura ha sido una inspiración constante en este camino.

A mis amigos, gracias por acompañarme durante todos estos años de carrera, por las risas, los desafíos compartidos y por estar presentes en cada momento importante. En especial, a quienes estuvieron conmigo en la recta final de este proceso, su apoyo y ánimo marcaron una gran diferencia.

Agradezco también a los profesores y profesionales que me guiaron en el desarrollo de esta memoria, en particular a los profesores Pedro Pinacho y Pablo Aqueveque, y al profesional supervisor Johannes Bunster, por su orientación, disposición y enseñanzas a lo largo del proyecto.

No puedo dejar de mencionar al equipo de fútbol de la carrera, quienes fueron parte importante de mi vida universitaria. Más que compañeros de equipo, se convirtieron en una familia, con quienes compartí no solo la pasión por el deporte, sino también innumerables momentos de compañerismo, esfuerzo y superación.

Gracias a todos quienes, de una u otra forma, me acompañaron en este camino. Este logro es también de ustedes.

Resumen

Implementación de una Plataforma Web para el Monitoreo de Respiradores Industriales Usando Angular.

Las cifras alarmantes sobre accidentes laborales y enfermedades respiratorias derivadas de condiciones adversas en entornos industriales, como minería y construcción, evidencian la necesidad de implementar nuevas tecnologías para el monitoreo de la calidad del aire. Este proyecto, desarrollado en el marco del proyecto FONDEF ID24I10487, presenta el desarrollo de una plataforma web diseñada para la gestión y monitoreo en tiempo real de las condiciones ambientales y de los respiradores industriales, con el fin de detectar tempranamente situaciones que puedan afectar la salud de los trabajadores.

La propuesta se llevó a cabo utilizando un enfoque iterativo e incremental, lo que permitió ajustes continuos durante el desarrollo. La plataforma, desarrollada con Angular como framework principal, destaca por su capacidad de ofrecer mediciones precisas en tiempo real, almacenar datos históricos y facilitar el análisis predictivo. Estas funcionalidades representan una mejora significativa respecto a los sistemas de monitoreo tradicionales, que suelen depender de registros manuales y carecen de inmediatez en la detección de riesgos.

Como resultado, esta solución tecnológica no solo optimiza la seguridad laboral, sino que también mejora la toma de decisiones a nivel operativo, al ofrecer información centralizada y accesible.

Palabras claves: Monitoreo en tiempo real, respiradores industriales, plataforma web, Angular, calidad del aire, seguridad laboral, tecnologías avanzadas.

Índice

Resumen.....	3
1. Introducción	6
1.1. Antecedentes generales	6
1.2. Solución propuesta y alcances	7
1.3. Objetivo general.....	8
1.4. Objetivos específicos	9
1.5. Metodología	9
1.6. Estructura del informe	9
2. Marco teórico.....	11
2.1. Monitoreo de condiciones laborales en entornos industriales de alto riesgo	11
2.2. Framework de Desarrollo de plataforma Web: Angular	12
2.3. Comunicación entre frontend y backend mediante API REST.....	13
2.4. Uso de MySQL para el modelado y almacenamiento de datos de monitoreo	14
2.5. Beneficios del monitoreo en tiempo real en la prevención de riesgos laborales	14
2.6. Mediciones y Umbrales en el monitoreo Industrial	15
2.6.1. Mediciones en la Industria y su Aplicación en Respiradores	15
2.6.2. Integración de dispositivos de medición, hardware y comunicación	16
2.6.3 Comunicación y flujo de datos	17
2.6.4. Concepto de Umbrales en Sistemas de Monitoreo.....	18
2.6.5. Relación entre Mediciones, Umbrales y Alertas	19
3. Desarrollo.....	20
3.1 Análisis	20
3.1.1 Limitaciones de los métodos tradicionales	20
3.1.2 Introducción al sistema	20
3.1.3 Casos de uso principales.....	21
3.1.4 Requerimientos funcionales	21
3.1.5 Requerimientos no funcionales	23
3.1.6 Trazabilidad y Validación	24
3.1.7 Comparación con sistemas existentes	25
3.1.8 Impacto esperado.....	26

3.1.9 Retos en la implementación.....	26
3.2 Diseño y arquitectura.....	28
3.2.1 Arquitectura del sistema.....	28
3.2.2 Modelado de datos	29
3.2.3 Diseño de la API.....	31
3.2.4 Interfaz de usuario	33
3.2.5 Diseño de seguridad.....	36
3.3 Implementación.....	37
3.3.1 Desarrollo de la interfaz web	38
3.3.2 Integración de la API.....	39
3.3.3 Elaboración del manual de usuario	41
4. Evaluación.....	43
5. Discusión.....	48
5.1. Ventajas obtenidas.....	48
5.2. Dificultades enfrentadas.....	48
5.3. Lecciones aprendidas.....	49
6. Conclusión	51
7. Referencias.....	52
8. Anexos	54

1. Introducción

1.1. Antecedentes generales

El aumento en la preocupación por la salud ocupacional y la seguridad laboral se encuentra respaldado por datos alarmantes. Según la Organización Internacional del Trabajo (ILO), cada año, alrededor de 2 millones de personas pierden la vida debido a situaciones ocurridas en su lugar de trabajo, sin contar los 160 millones que padecen enfermedades laborales y los 270 millones de accidentes laborales que ocurren anualmente. Entre las causas de estas tragedias, un porcentaje significativo se atribuye a enfermedades respiratorias y cánceres originados por la exposición a sustancias peligrosas, como sílice y otros polvos.

En entornos laborales industriales, la protección de los trabajadores es un desafío constante. Las actividades que generan exposición a agentes contaminantes, como la minería, la construcción y otras industrias pesadas, requieren medidas de seguridad avanzadas para mitigar el impacto en la salud de los empleados. En particular, los respiradores industriales desempeñan un papel crucial como equipo de protección personal (EPP). Sin embargo, la falta de un monitoreo efectivo y en tiempo real limita la capacidad de prevenir y actuar rápidamente ante condiciones adversas que podrían amenazar la integridad de los trabajadores.

Sumados a lo dicho, los costos asociados a la falta de medidas adecuadas de seguridad laboral no solo impactan en la salud de los trabajadores, sino también en la economía global, con pérdidas estimadas en miles de millones de dólares debido a bajas laborales, tratamientos médicos y disminución de la productividad. Por lo tanto, abordar este problema mediante soluciones tecnológicas innovadoras se ha convertido en una prioridad en sectores industriales clave.

1.2. Solución propuesta y alcances

La implementación de una plataforma de monitoreo para respiradores industriales es el enfoque principal de este proyecto de título, con el objetivo de garantizar la salud y seguridad de los trabajadores en entornos laborales desafiantes. Aunque los respiradores industriales son herramientas cruciales de protección, actualmente carecen de la tecnología necesaria para proporcionar datos en tiempo real sobre la calidad del aire, el estado de los filtros o el ajuste adecuado del equipo. Por esta razón, este proyecto se centra en desarrollar una plataforma web que permita el monitoreo continuo y la evaluación efectiva del rendimiento de estos dispositivos.

El proyecto Fondef IAD I+D Sistema de Monitoreo de Condiciones de Respiradores Industriales (ID24110487) aborda este desafío a través de una solución que incluye un sistema de monitoreo portátil y adaptable, capaz de detectar la respiración y otras condiciones de los trabajadores sin interferir en sus actividades diarias. En este contexto, el desarrollo de la plataforma web es el núcleo de este trabajo de título, ya que se enfoca en ofrecer acceso a información detallada sobre las condiciones laborales en tiempo real, generando reportes y alertas automatizadas que permiten a supervisores y administradores tomar decisiones rápidas y efectivas para mitigar riesgos.

La plataforma web está diseñada para proporcionar una interfaz intuitiva y modular que centraliza la visualización del estado de los trabajadores y las alertas capturadas. Las funcionalidades principales incluyen la gestión de alertas sobre el nivel de batería, el estado de los filtros y el ajuste (fitting) del respirador. Estos datos, recopilados y procesados en tiempo real, aseguran que cualquier anomalía pueda ser detectada y gestionada de manera oportuna, mejorando la seguridad y reduciendo el riesgo de incidentes.

Además de la supervisión en tiempo real, la plataforma permite la recopilación y almacenamiento de datos históricos, proporcionando una base para análisis que facilita la identificación de patrones y tendencias que podrían indicar la necesidad de mantenimiento o reemplazo del equipo. De esta manera, se optimiza la planificación y se fortalece la capacidad de respuesta ante potenciales emergencias, asegurando un entorno de trabajo más seguro y eficiente.

Con estas capacidades, la plataforma de monitoreo de respiradores industriales se establece como una herramienta esencial para la gestión integral de la seguridad en entornos industriales, contribuyendo al bienestar de los trabajadores y a la mejora continua de las prácticas de seguridad laboral.

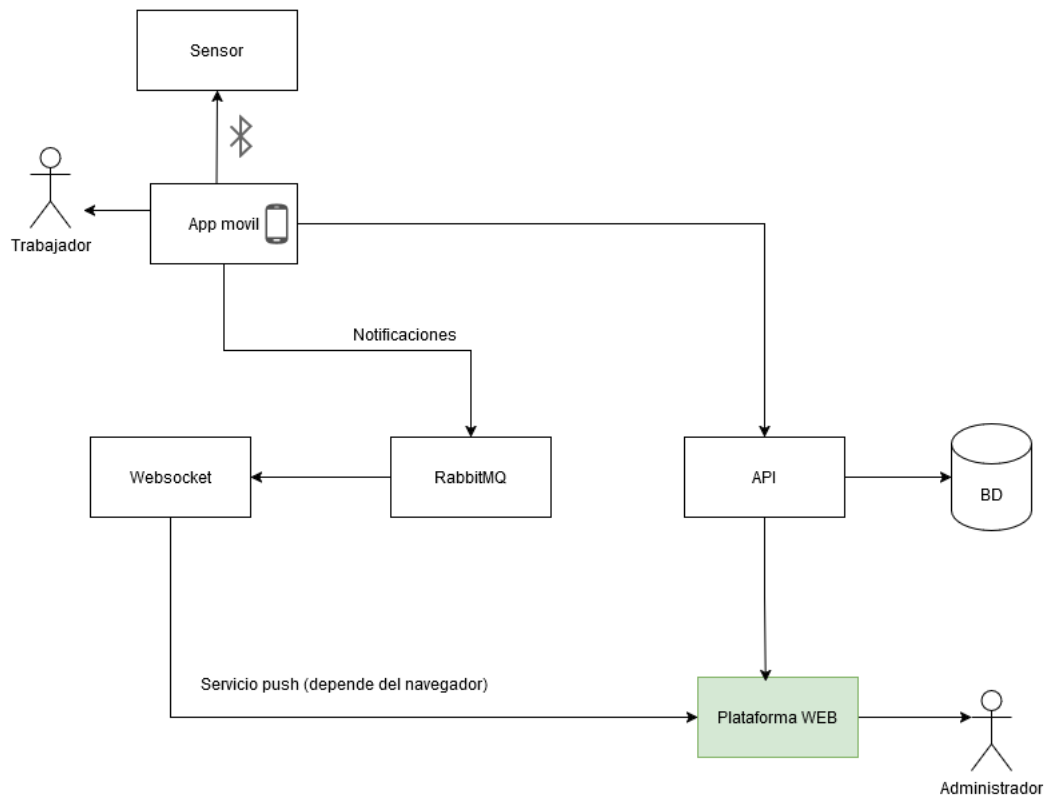


Imagen 1: Arquitectura del sistema

1.3. Objetivo general

Desarrollar una plataforma web para el monitoreo de la calidad del aire y el control de las condiciones de los respiradores en entornos industriales, facilitando la identificación temprana de anomalías y la generación de alertas ante posibles riesgos para la salud de los trabajadores.

1.4. Objetivos específicos

- **Desarrollar la plataforma web:** Implementar una interfaz que permita la visualización y gestión de los datos capturados, ofreciendo a los usuarios una forma intuitiva de analizar la información sobre la calidad del aire y otros factores medidos.
- **Integrar la API en el frontend de la plataforma web:** Configurar la comunicación entre la plataforma web y la API preexistente, asegurando una interacción fluida mediante peticiones HTTP y el manejo adecuado de las respuestas y errores.
- **Documentar la arquitectura de software:** Elaborar un documento detallado que describa las vistas de la arquitectura de software y de red, y cómo se integran los componentes de la solución.
- **Crear un manual de usuario:** Desarrollar un manual que explique cómo los diferentes tipos de usuarios pueden interactuar con la plataforma, detallando la gestión de usuarios, la visualización de datos y el manejo de alertas.

1.5. Metodología

El desarrollo de este proyecto se llevó a cabo mediante un enfoque iterativo e incremental. Durante el proceso, se realizaron reuniones semanales para evaluar los avances y refinar los requerimientos funcionales y no funcionales. Este enfoque permitió revisiones constantes del progreso y la incorporación de mejoras en cada fase del desarrollo, asegurando así una solución eficiente y ajustada a las necesidades detectadas.

1.6. Estructura del informe

Capítulo 1 “Introducción”: Presenta los antecedentes del problema, la solución propuesta y los objetivos del proyecto, además de la metodología utilizada en el desarrollo.

Capítulo 2 “Marco Teórico”: Proporciona el contexto y los fundamentos teóricos relacionados con el monitoreo industrial, las tecnologías utilizadas en la plataforma, la integración de la API y los beneficios del monitoreo en tiempo real.

Capítulo 3 "Desarrollo": Describe el análisis de la solución, incluyendo las limitaciones de los métodos tradicionales, los requerimientos funcionales y no funcionales, así como la comparación con otros sistemas existentes. También detalla el diseño y arquitectura del sistema, el modelado de datos, la implementación de la API y la interfaz de usuario.

Capítulo 4 "Evaluación": Presenta las pruebas realizadas sobre la plataforma, evaluando su funcionalidad, facilidad de uso y rendimiento. Además, analiza el feedback recibido de los usuarios para determinar mejoras potenciales.

Capítulo 5 "Discusión ": Examina los resultados obtenidos, destacando las ventajas de la plataforma, las dificultades enfrentadas en el desarrollo y las lecciones aprendidas a lo largo del proceso.

Capítulo 6 "Conclusión ": Resume los logros alcanzados en el desarrollo de la plataforma, la importancia de su implementación en el contexto industrial y las oportunidades de mejora para futuras versiones.

Capítulo 7 "Referencias": Incluye las fuentes bibliográficas y documentales utilizadas a lo largo del desarrollo del informe.

Capítulo 8 "Anexos": Contiene documentación complementaria, como el diagrama relacional de la base de datos, e imágenes de la interfaz.

2. Marco teórico

2.1. Monitoreo de condiciones laborales en entornos industriales de alto riesgo

El monitoreo industrial de trabajadores es esencial en la industria moderna para garantizar la salud y seguridad laboral, especialmente en entornos peligrosos. Este proceso implica la recopilación y análisis de datos sobre variables físicas y ambientales que afectan directamente a los trabajadores, como temperatura, presión, humedad y contaminación. Según la Organización Internacional del Trabajo (OIT), un entorno de trabajo seguro y saludable es un principio y derecho fundamental, y la implementación de sistemas de gestión de seguridad y salud en el trabajo es crucial para prevenir accidentes y enfermedades laborales [1].

Con los avances tecnológicos, se han desarrollado dispositivos que se integran en los respiradores industriales para medir en tiempo real variables críticas y otros parámetros relevantes. Esta integración ha demostrado ser eficaz en sectores como la minería y la construcción, donde el uso de sensores y monitoreo remoto proporciona información en tiempo real sobre variables clave, permitiendo una respuesta rápida a las condiciones cambiantes en el entorno laboral [2].

La incorporación de tecnologías de monitoreo en tiempo real no solo mejora la capacidad de respuesta ante situaciones de riesgo, sino que también permite un análisis preventivo. La recopilación continua de datos facilita la identificación de patrones que pueden predecir fallos o condiciones inseguras antes de que ocurran incidentes graves. Esto se traduce en una reducción de los accidentes laborales y en una mejora del bienestar general de los trabajadores, lo que es especialmente importante en industrias donde las condiciones ambientales son a menudo extremas y cambiantes [3].

Además, la adopción de tecnologías de monitoreo se ha vinculado con mejoras en la eficiencia operativa y la reducción de costos relacionados con bajas laborales. Por ejemplo, la implementación de sistemas de monitoreo para equipos mineros ayuda a

garantizar que las máquinas funcionen de manera óptima, lo que contribuye a reducir el tiempo de inactividad y los costos de mantenimiento [4].

En resumen, el monitoreo industrial mediante tecnologías avanzadas es una herramienta fundamental para garantizar la seguridad y salud de los trabajadores, optimizar las operaciones y reducir costos en industrias de alto riesgo.

2.2. Framework de Desarrollo de plataforma Web: Angular

Angular es un framework de desarrollo de aplicaciones web creado y mantenido por Google. Es conocido por su capacidad para construir aplicaciones de una sola página (Single Page Applications) con una estructura bien definida y mantenible. Angular utiliza TypeScript, un superconjunto de JavaScript, que proporciona características avanzadas como la tipificación estática, mejorando la calidad del código y facilitando el mantenimiento a largo plazo. Entre sus componentes clave se incluyen módulos, servicios y directivas, que permiten una construcción modular y reutilizable de la interfaz de usuario.

Según un artículo de Kinsta [5], Angular permite reducir el tiempo de codificación y crear aplicaciones de alto rendimiento. Además, su uso de TypeScript permite detectar y corregir errores en etapas tempranas del desarrollo, aumentando la estabilidad del código.

Angular Material es una biblioteca de componentes de interfaz de usuario que implementa las especificaciones de Material Design de Google. Ofrece una colección de componentes preconstruidos que facilitan la creación de aplicaciones web modernas, responsivas y visualmente atractivas. Además, Angular Material prioriza los estándares de accesibilidad, lo que simplifica para los desarrolladores la creación de aplicaciones accesibles para usuarios con diversas necesidades [6].

Angular fue seleccionado en comparación con otros frameworks como React o Vue.js debido a su robustez para aplicaciones de gran escala, su ecosistema integrado y sus herramientas avanzadas para *testing* y optimización. Un análisis comparativo de frameworks publicado en BrowserStack confirma que Angular es un *framework* completo, mientras que React es solo una biblioteca de user interface (UI) y Vue un *framework* progresivo, lo que lo hace ideal para proyectos empresariales con necesidades de escalabilidad y modularidad [7]

2.3. Comunicación entre frontend y backend mediante API REST

La integración de una API REST es esencial para garantizar la comunicación eficiente entre el frontend de la plataforma web y la base de datos. REST (*Representational State Transfer*), definido por Roy Fielding en el año 2000, es un estilo arquitectónico ampliamente utilizado en sistemas distribuidos debido a su escalabilidad y compatibilidad con los protocolos web existentes, como HTTP [8].

La API REST organiza sus funcionalidades en módulos o "recursos", cada uno relacionado con una parte específica del sistema, como la gestión de trabajadores, mediciones ambientales, alertas y jornadas de trabajo. Este enfoque modular permite un mantenimiento más sencillo y asegura un rendimiento óptimo al manejar grandes volúmenes de solicitudes simultáneas. Además, REST se basa en operaciones CRUD (Crear, Leer, Actualizar, Eliminar), utilizando métodos HTTP estándar como GET, POST, PUT y DELETE, lo que simplifica su implementación y compatibilidad con múltiples plataformas [9].

En comparación con GraphQL, REST fue seleccionado debido a su simplicidad y eficiencia para manejar estructuras de datos bien definidas y jerárquicas. Mientras que GraphQL permite consultas más flexibles y precisas, REST es más adecuado para proyectos donde las relaciones entre datos son claras y el rendimiento es una prioridad. Según un análisis de Amazon Web Services, REST es ideal para aplicaciones con

interacciones predecibles y bien estructuradas, mientras que GraphQL se recomienda en entornos donde la flexibilidad en la consulta de datos es crítica [10].

2.4. Uso de MySQL para el modelado y almacenamiento de datos de monitoreo

MySQL fue seleccionada como la base de datos relacional para esta plataforma debido a su fiabilidad, alto rendimiento y facilidad de uso. Según Google Cloud, MySQL es rápido, fiable, escalable y fácil de usar, lo que lo convierte en una opción preferida para aplicaciones web que manejan grandes volúmenes de datos de manera eficiente [11].

En la evaluación de alternativas, PostgreSQL destacó por sus capacidades analíticas avanzadas; sin embargo, MySQL fue preferida por su mayor rendimiento en operaciones de lectura, simplicidad en la configuración y amplia disponibilidad de recursos técnicos. El ranking de DB-Engines muestra que MySQL ocupa una posición superior en popularidad en comparación con PostgreSQL, lo que indica una mayor adopción en aplicaciones web con modelos de datos estructurados y relaciones claras [12].

2.5. Beneficios del monitoreo en tiempo real en la prevención de riesgos laborales

El monitoreo en tiempo real es una innovación clave en la gestión de la seguridad laboral, ya que permite la detección inmediata de problemas, reduciendo significativamente el tiempo de reacción ante emergencias y mejorando la eficiencia operativa. Un estudio publicado en la Global Journal of Research in Science and Technology destaca que la implementación de paneles de monitoreo en tiempo real en operaciones industriales mejora el desempeño en seguridad y asegura el cumplimiento normativo, permitiendo una gestión proactiva de los riesgos [13].

Además, los datos en tiempo real son fundamentales para optimizar los procesos de mantenimiento preventivo, asegurando la continuidad de las operaciones. Según un informe de Deloitte, la implementación de sistemas de mantenimiento predictivo puede reducir los costos de mantenimiento en un 5-10% y aumentar la disponibilidad de los equipos en un 10-20% [14].

Existen diversas soluciones en el mercado que aplican monitoreo en tiempo real para la seguridad industrial. Un ejemplo de ello es el sistema de **Honeywell Connected Worker** [15], que permite la supervisión remota de trabajadores en entornos de alto riesgo, proporcionando alertas en tiempo real sobre condiciones peligrosas y facilitando una respuesta rápida ante emergencias. Esta tecnología mejora la seguridad y reduce los tiempos de reacción en situaciones críticas, optimizando la gestión de riesgos en la industria.

2.6 Mediciones y Umbrales en el monitoreo Industrial

El uso de sensores en entornos industriales ha permitido la monitorización en tiempo real de variables críticas para la seguridad de los trabajadores. En el caso de los respiradores industriales, estos sensores pueden integrarse como dispositivos complementarios para medir parámetros ambientales y operativos, contribuyendo a la detección temprana de riesgos. Por ejemplo, los sensores de presión y temperatura monitorean las condiciones del aire suministrado, asegurando que se mantengan dentro de los rangos seguros establecidos [16].

2.6.1. Mediciones en la Industria y su Aplicación en Respiradores

Los sensores industriales desempeñan un papel clave en la prevención de riesgos al medir variables como:

- **Frecuencia respiratoria:** Permite identificar esfuerzos respiratorios inusuales.
- **Nivel de ajuste:** Evalúa el sellado del respirador en el rostro del trabajador.
- **Presión en los filtros:** Indicador del estado del filtro del respirador.
- **Estado de batería:** Asegura el correcto funcionamiento del sensor durante la jornada.

Su aplicación en entornos como la minería o la construcción ha demostrado reducir incidentes relacionados con la exposición a contaminantes. Por ejemplo, el uso de detectores portátiles de gases múltiples permite monitorear simultáneamente hasta seis gases, alertando a los trabajadores sobre la presencia de sustancias peligrosas y mejorando la seguridad en el lugar de trabajo [17].

2.6.2. Integración de dispositivos de medición, hardware y comunicación

El sistema de monitoreo de respiradores industriales se basa en una arquitectura que permite la captura y transmisión de datos en tiempo real mediante sensores integrados en los dispositivos utilizados por los trabajadores. Para lograr esto, es fundamental comprender la interacción entre los sensores, la placa base, la comunicación con la infraestructura digital y las características técnicas del sistema.

Integración del Sensor con el Respirador

Los sensores están físicamente integrados en los respiradores y se encargan de medir variables críticas como la frecuencia respiratoria, presión, humedad y nivel de batería. Estos datos son fundamentales para detectar condiciones anómalas y prevenir riesgos en los trabajadores expuestos a entornos industriales exigentes.

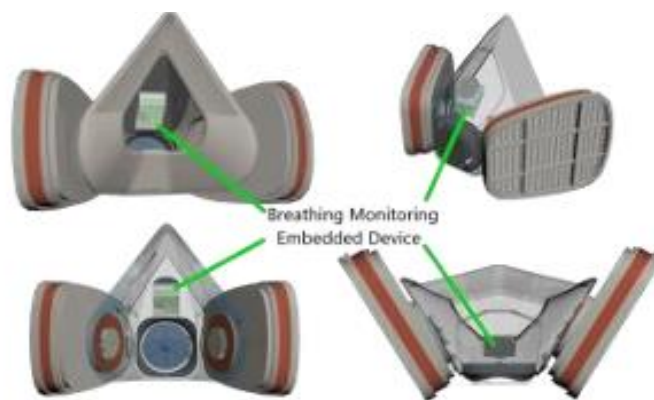


Imagen 2: Ubicación del sensor dentro del respirador

Características Técnicas del Hardware

El sistema cuenta con una **placa base** que gestiona la recolección de datos y su procesamiento inicial antes de transmitirlos. A continuación, se presentan los principales componentes del hardware:

- **Sensores principales:**
 - Sensor de presión: Detecta cambios en la resistencia del filtro.
 - Sensor de nivel de ajuste: Detecta si el respirador está correctamente sellado.
 - Sensor de batería: Monitorea la carga del dispositivo.

- **Placa base:** Procesa la información recibida de los sensores y la prepara para su transmisión.
- **Fuente de energía:** La autonomía del dispositivo es clave para su operatividad. La duración de la batería depende del uso y la eficiencia en la gestión del consumo de energía.

2.6.3 Comunicación y flujo de datos

Los datos generados por los sensores son enviados a una aplicación móvil mediante comunicación **Bluetooth**. Posteriormente, esta aplicación móvil transmite la información a la infraestructura central del sistema a través de un protocolo de mensajería asíncrona (**RabbitMQ** y **WebSockets**), lo que permite una actualización en tiempo real en la plataforma web.

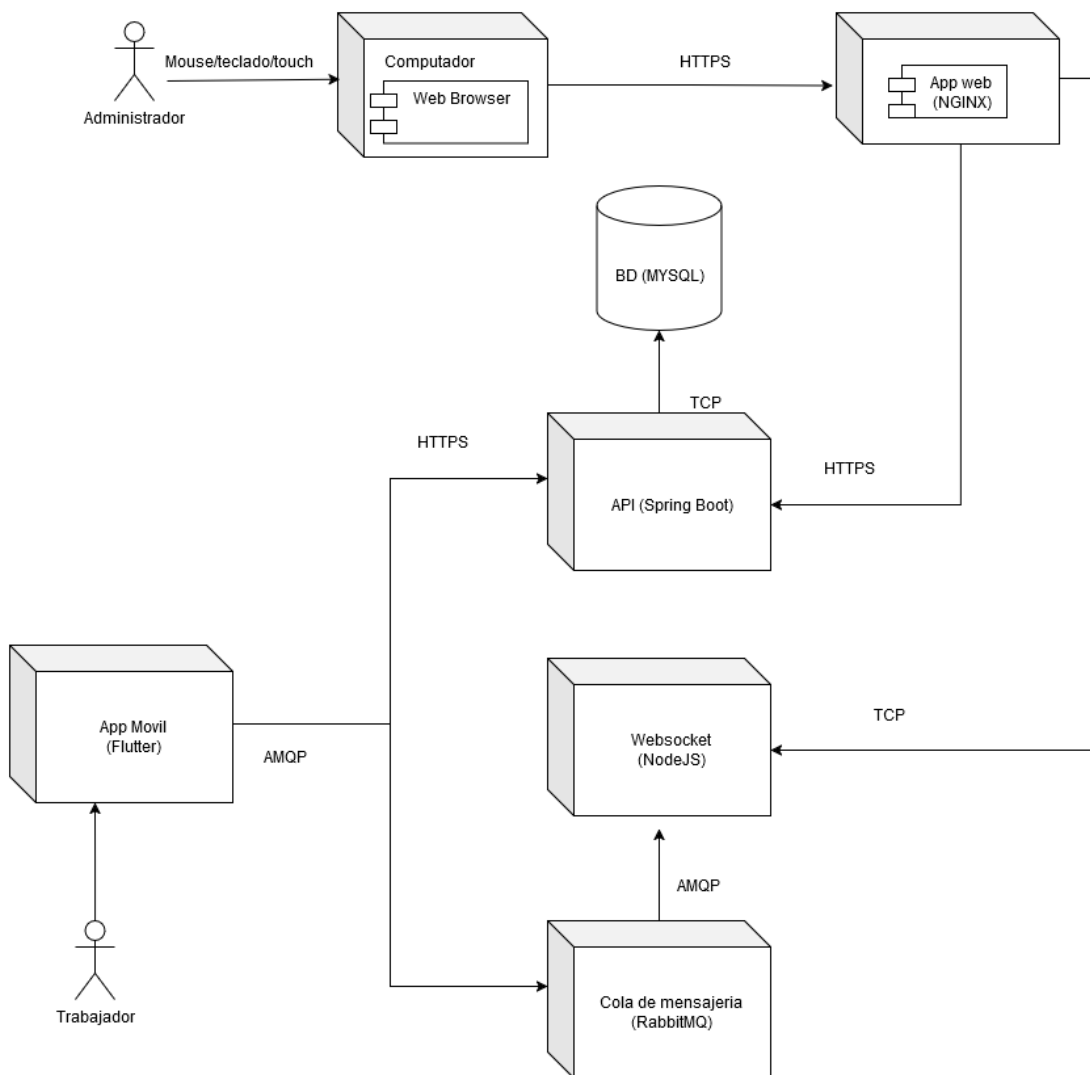


Imagen 3: Diagrama de despliegue

Seguridad y Fiabilidad en la Comunicación

Para garantizar la integridad y protección de los datos durante la transmisión, el sistema implementa diversas medidas de seguridad:

- **Uso de HTTPS** para proteger las comunicaciones entre dispositivos y servidores.
- **Firewall institucional** que restringe accesos no autorizados y protege la infraestructura.
- **Manejo eficiente de datos en RabbitMQ**, asegurando que las mediciones críticas sean procesadas con prioridad.

2.6.4. Concepto de Umbrales en Sistemas de Monitoreo

Los umbrales permiten clasificar los datos en rangos que determinan cuándo una condición es segura o riesgosa. Se dividen en:

- **Umbral bajo:** Condición normal.
- **Umbral medio (advertencia):** Posible anomalía sin riesgo inmediato.
- **Umbral alto (crítico):** Situación de riesgo que requiere acción inmediata.

Estos valores se establecen siguiendo normativas como las de la Administración de Seguridad y Salud Ocupacional (OSHA), que proporciona directrices para gestionar la calidad del aire en el entorno laboral, incluyendo controles de temperatura, humedad y ventilación [18]. En Chile, la Norma DS 594 regula las condiciones sanitarias y ambientales básicas en los lugares de trabajo, estableciendo límites permisibles para la exposición a contaminantes y otros agentes ambientales [19].

Ejemplo de umbrales definidos:

Variable	Rango Normal	Advertencia	Crítico
Frecuencia Respiratoria	12 - 18 rpm	18 - 20 rpm	> 20 rpm o < 12 rpm
Nivel de Ajuste	0 - 102000	102000 - 103000	> 103000
Presión en Filtros (Medio)	> 100500 Pa	100500 - 100350 Pa	< 100350 Pa

Estado de Batería (%)	> 20%	10 - 20%	< 5%
------------------------------	-------	----------	------

2.6.5. Relación entre Mediciones, Umbrales y Alertas

Los sensores generan alertas cuando detectan valores fuera de los umbrales establecidos. Algunos ejemplos incluyen:

- **Alerta de batería baja:** Indica necesidad de recarga.
- **Alerta de filtro obstruido:** Señala la necesidad de reemplazo.
- **Alerta de respiración anormal:** Posible fatiga o condición ambiental adversa.
- **Alerta de ajuste incorrecto:** Indica que el respirador no está sellando adecuadamente.

La implementación de estos sistemas de monitoreo y alerta contribuye a una respuesta rápida ante condiciones peligrosas, mejorando la seguridad y reduciendo el riesgo de incidentes en entornos industriales.

3. Desarrollo

3.1 Análisis

3.1.1 Limitaciones de los métodos tradicionales

El monitoreo de respiradores y filtros en entornos industriales, en su forma convencional, presenta diversas limitaciones que dificultan garantizar la seguridad de los trabajadores. Entre estas se encuentran:

- **Falta de recolección de datos en tiempo real:** Los métodos tradicionales dependen de inspecciones periódicas, lo que no permite detectar cambios rápidos en las condiciones ambientales o el estado del equipo, aumentando el riesgo de incidentes.
- **Dependencia de la intervención humana:** La supervisión manual es propensa a errores debido a factores como la fatiga, falta de capacitación o subestimación de riesgos. Esto puede generar retrasos en la detección de fallas críticas.
- **Ausencia de almacenamiento histórico confiable:** Los registros manuales dificultan el análisis de tendencias a largo plazo, limitando la planificación de mantenimiento y la implementación de estrategias preventivas.

Estas deficiencias subrayan la necesidad de una solución tecnológica que automatice la recolección y el análisis de datos, permitiendo una respuesta más eficiente y oportuna.

3.1.2 Introducción al sistema

La Plataforma de Monitoreo de Respiradores Industriales es una herramienta diseñada para supervisar y gestionar el uso de respiradores en entornos industriales de alto riesgo. Se enfoca en dos tipos de usuarios:

- **Administrador:** Responsable de la gestión del sistema, configuraciones y usuarios.
- **Supervisor:** Encargado de monitorear a los trabajadores bajo su supervisión.

Definiciones clave:

- **Trabajador:** Usuario que opera en entornos industriales y cuyo respirador está siendo monitoreado.

- **Supervisor:** Usuario que tiene acceso a la plataforma para monitorear a los trabajadores de su cuadrilla y recibir alertas sobre su estado.
- **Administrador:** Usuario con acceso total al sistema, encargado de la configuración de parámetros, gestión de trabajadores y supervisores, y mantenimiento general de la plataforma.
- **Cuadrilla:** Grupo de trabajadores asignados a un supervisor.
- **Alerta:** Notificación generada cuando una condición monitoreada supera los umbrales de seguridad establecidos. Puede ser de tipo batería baja, ajuste deficiente del respirador, saturación del filtro, entre otras.
- **Umbral:** Límites configurables que determinan cuándo una condición es normal, de advertencia o crítica.

3.1.3 Casos de uso principales

Los siguientes casos de uso representan las interacciones clave del sistema:

1. **Inicio de Sesión:** El usuario ingresa con su RUT y contraseña.
2. **Visualización del Estado de la Cuadrilla:** Supervisores pueden monitorear el estado de sus trabajadores y recibir alertas en tiempo real.
3. **Gestión de Umbrales:** El administrador puede configurar los umbrales críticos para cada trabajador.
4. **Gestión de Trabajadores:** Solo los administradores pueden agregar, editar o desactivar trabajadores.
5. **Gestión de Dispositivos:** Administración de filtros y respiradores por parte del administrador.
6. **Historial de Alertas:** Supervisores y administradores pueden consultar el registro de alertas por tipo y fecha.

3.1.4 Requerimientos funcionales

Los siguientes requerimientos funcionales describen las características y funcionalidades que debe cumplir la plataforma para garantizar su correcto funcionamiento. Cada requerimiento está identificado con un código único para facilitar su trazabilidad y referencia.

Gestión de Usuarios

ID	Requerimiento	Descripción
RF-001	Autenticación de usuarios	La plataforma debe permitir la autenticación de usuarios mediante RUT y contraseña con validación segura.
RF-002	Definición de roles	El sistema debe asignar permisos según el rol del usuario (Administrador o Supervisor).
RF-003	Restricción de accesos	Los supervisores solo pueden acceder a información relacionada con las cuadrillas y alertas, mientras que los administradores tienen acceso total a todas las secciones de la plataforma.
RF-004	Gestión de trabajadores	Los administradores pueden agregar nuevos trabajadores y editar los existentes en la plataforma.

Monitoreo en Tiempo Real

ID	Requerimiento	Descripción
RF-005	Estado de cuadrilla	Supervisores y administradores deben poder visualizar en tiempo real el estado de los trabajadores.
RF-006	Registro de alertas	La plataforma debe registrar automáticamente alertas de filtro, batería, ajuste y respiración anormal.
RF-007	Visualización de alertas	Las alertas deben mostrarse con iconografía y colores según criticidad.

Gestión de Datos y Configuración

ID	Requerimiento	Descripción
RF-008	Historial de alertas	La plataforma debe permitir consultar un historial de alertas con filtro por fecha, trabajador y tipo de alerta.

RF-009	Configuración de umbrales	El administrador debe poder definir umbrales personalizados para alertas de respiración, ajuste, batería y filtro.
RF-010	Gestión de filtros y respiradores	Solo los administradores pueden registrar, editar y eliminar filtros y respiradores en la plataforma.
RF-011	Gestión de roles y ubicaciones	Solo los administradores pueden definir nuevos roles y ubicaciones para los trabajadores.

Interfaz y Usabilidad

ID	Requerimiento	Descripción
RF-012	Interfaz accesible	La plataforma debe ser intuitiva, con indicadores visuales claros y colores semáforo para alertas.
RF-013	Integración con Help Desk	Se debe permitir la creación de tickets de soporte mediante un enlace a una plataforma externa.

3.1.5 Requerimientos no funcionales

Los requerimientos no funcionales especifican las características de calidad y desempeño que la plataforma debe cumplir.

Seguridad

ID	Requerimiento	Descripción
RNF-001	Cifrado de datos	Todas las contraseñas y credenciales deben almacenarse de forma segura usando técnicas de cifrado.
RNF-002	Comunicación segura	Toda la comunicación debe realizarse sobre HTTPS para garantizar la privacidad de los datos.
RNF-003	Gestión de sesiones	Se debe implementar autenticación con JWT y restringir accesos según roles.

Escalabilidad y Rendimiento

ID	Requerimiento	Descripción
RNF-004	Manejo de usuarios concurrentes	La plataforma debe ser capaz de gestionar múltiples usuarios sin degradación en el rendimiento.

Compatibilidad y Mantenimiento

ID	Requerimiento	Descripción
RNF-005	Compatibilidad con navegadores	La plataforma debe ser compatible con las últimas versiones de Chrome, Firefox y Edge.
RNF-006	Modularidad	La arquitectura debe permitir la adición de nuevas funcionalidades sin afectar el sistema existente.

3.1.6 Trazabilidad y Validación

Cada requerimiento funcional y no funcional debe estar asociado con una estrategia de validación para garantizar su correcta implementación. A continuación, se presenta la matriz de trazabilidad:

Requerimiento	Método de Validación
RF-001	Prueba de inicio de sesión con usuarios registrados y no registrados.
RF-002	Prueba de asignación de roles y permisos en diferentes escenarios.
RF-003	Simulación de restricciones de acceso según el tipo de usuario.
RF-004	Prueba de creación y edición de trabajadores.
RF-005	Simulación de múltiples sensores enviando datos en tiempo real.
RF-006	Pruebas de generación de alertas en distintas condiciones.
RF-007	Evaluación de la interfaz de usuario con colores e iconografía.
RF-008	Verificación de filtrado en el historial de alertas.

RF-009	Configuración y prueba de umbrales con diferentes parámetros.
RF-010	Prueba de registro, edición y eliminación de filtros y respiradores.
RF-011	Prueba de definición de roles y ubicaciones en la plataforma.
RF-012	Evaluación de la usabilidad de la interfaz y accesibilidad.
RF-013	Validación del enlace a la plataforma de Help Desk.
RNF-001	Revisión del almacenamiento de contraseñas cifradas.
RNF-002	Validación de la seguridad en la comunicación con HTTPS.
RNF-003	Prueba de autenticación y expiración de sesiones con JWT.
RNF-004	Simulación de carga con múltiples usuarios concurrentes.
RNF-005	Prueba de compatibilidad en distintos navegadores.
RNF-006	Validación de la modularidad con pruebas de ampliación de funciones.

3.1.7 Comparación con sistemas existentes

La propuesta de la plataforma supera las limitaciones de los sistemas actuales. A continuación, se detalla una comparación:

Característica	Métodos Tradicionales	Plataforma Propuesta
Recolección de datos	Manual	Automatizada
Monitoreo en tiempo real	No	Si
Gestión de alertas	Manual	Avanzada y personalizada
Análisis histórico	Limitado	Completo

Tabla 1: Tabla de comparación con métodos tradicionales

Esta comparación destaca cómo la solución propuesta integra tecnologías modernas para ofrecer una herramienta integral.

3.1.8 Impacto esperado

La implementación de la plataforma tiene el potencial de generar un impacto significativo en la seguridad laboral y la eficiencia operativa:

- **Reducción de riesgos:** La detección temprana de fallos permite prevenir incidentes graves, mejorando la seguridad laboral y la confianza de los trabajadores.
- **Optimización de recursos:** El análisis de datos históricos ayuda a planificar el uso de equipos, reducir costos operativos y mejorar la eficiencia de las operaciones.
- **Toma de decisiones basada en datos:** Los supervisores y administradores pueden implementar medidas correctivas de manera más efectiva gracias a la información confiable y procesable que proporciona la plataforma.

3.1.9 Retos en la implementación

Aunque la plataforma propuesta aborda las limitaciones de los métodos tradicionales, también presenta desafíos que deben considerarse:

1. Adopción Tecnológica

Uno de los retos principales ha sido garantizar que los usuarios, especialmente supervisores y administradores, adopten la plataforma sin dificultades. Para abordar esto, se ha desarrollado un manual de usuario detallado enfocado en el uso eficiente de las funcionalidades clave.

2. Integración de la API y manejo de datos

La integración de la API es fundamental para el funcionamiento de la plataforma. Para garantizar un correcto manejo de los datos, se han implementado validaciones en el frontend que verifican la integridad de la información antes de enviarla a la API. Asimismo, se han desarrollado pruebas para evaluar la correcta recuperación y visualización de los datos provenientes del backend, asegurando que los tiempos de respuesta sean adecuados y que la información se presente de manera estructurada y clara en la interfaz.

3. Diseño de interfaz intuitiva

La plataforma debe ser clara y accesible para todos los usuarios. Para lograrlo, se ha utilizado Angular Material y se han definido iconografías que mejoran la experiencia visual. Además, se han realizado pruebas de usabilidad con diferentes perfiles de usuario para ajustar elementos según la retroalimentación recibida, priorizando la accesibilidad y facilidad de navegación.

4. Seguridad y control de accesos

El manejo de información sensible requiere medidas estrictas de seguridad. Se ha implementado autenticación con JWT, cifrado de contraseñas y uso de HTTPS para la comunicación segura. Además, se han aplicado restricciones de acceso según roles para garantizar que solo los administradores puedan gestionar configuraciones críticas.

5. Mantenimiento y escalabilidad

Para asegurar que la plataforma sea sostenible en el tiempo, se ha estructurado el código en módulos independientes, permitiendo que futuras expansiones no afecten el rendimiento. También se han definido métricas de rendimiento para evaluar posibles mejoras en la infraestructura.

Con estas estrategias, la implementación de la plataforma se encamina a ofrecer una solución robusta, segura y fácil de usar.

3.2 Diseño y arquitectura

3.2.1 Arquitectura del sistema

La arquitectura del sistema se basa en un modelo de tres capas, lo que permite una estructura organizada y escalable. Estas capas incluyen la capa de presentación (frontend), la capa lógica de negocios (backend) y la capa de datos, cada una con funciones específicas dentro del funcionamiento de la plataforma.

La capa de presentación es la parte central de este proyecto de título y corresponde a la plataforma web, desarrollada con Angular. Su diseño está enfocado en la usabilidad y en la facilidad de acceso a la información, permitiendo a supervisores y administradores gestionar alertas, trabajadores y parámetros de seguridad de manera intuitiva. Se implementó un sistema de navegación basado en rutas, lo que facilita la transición fluida entre secciones sin necesidad de recargar la página. Además, la interfaz aprovecha Angular Material para garantizar un diseño moderno y consistente, con componentes visuales adaptables a distintos dispositivos.

La capa lógica de negocios, implementada con Spring Boot, se encarga de procesar la información y manejar la comunicación entre la plataforma web y la base de datos. Su función principal es recibir solicitudes del frontend, validar los datos y ejecutar las operaciones necesarias antes de devolver una respuesta estructurada. La API REST que expone este backend está diseñada para gestionar usuarios, alertas, jornadas de trabajo y configuraciones del sistema.

La capa de datos, basada en MySQL, almacena la información relevante de la plataforma, incluyendo datos de trabajadores, jornadas laborales, mediciones ambientales e inerciales y alertas generadas. Su estructura permite organizar la información de manera eficiente, garantizando que las consultas sean rápidas y seguras.

El sistema sigue un flujo de comunicación donde la plataforma web envía solicitudes a la API, la cual procesa la información y la almacena o la recupera desde la base de datos,

devolviendo una respuesta para ser presentada al usuario en la interfaz. Esta separación modular facilita la escalabilidad del sistema y permite futuras mejoras sin afectar la estabilidad de la plataforma.

3.2.2 Modelado de datos

El modelo de datos del sistema ha sido diseñado para gestionar eficientemente la información capturada por la plataforma, asegurando la integridad y accesibilidad de los datos almacenados. Se ha optado por una base de datos relacional en MySQL, debido a su estabilidad, escalabilidad y compatibilidad con sistemas transaccionales.

La estructura del modelo de datos se basa en múltiples entidades interconectadas, donde cada una desempeña un papel clave en la operatividad de la plataforma. Se han definido claves primarias y foráneas para garantizar la consistencia en la información almacenada y facilitar la optimización de consultas.

Principales entidades y relaciones

El diseño del modelo de datos contempla las siguientes entidades principales:

- **Trabajador:** Registra la información de los usuarios, incluyendo credenciales de acceso, datos personales y roles dentro del sistema.
- **Jornada de trabajo:** Almacena los periodos de actividad de cada trabajador, vinculando dispositivos y condiciones ambientales.
- **Mediciones ambientales e inerciales:** Registra los datos capturados en tiempo real por los sensores, como temperatura, humedad, presión y aceleración.
- **Alertas:** Contiene eventos generados automáticamente cuando se superan los umbrales de seguridad definidos.

3.2.3 Diseño de la API

La API REST desarrollada para la plataforma de monitoreo de respiradores industriales actúa como el intermediario entre la base de datos y la interfaz de usuario. Su arquitectura sigue principios **RESTful**, lo que permite una comunicación eficiente entre el backend y el frontend, facilitando la gestión de datos en tiempo real.

Principales funcionalidades de la API

La API proporciona endpoints organizados en diferentes categorías, según los datos que gestionan:

Autenticación y gestión de usuarios

- **Login y autenticación:** Proporciona un **token de sesión** al usuario autenticado, permitiendo acceso a las funcionalidades según su rol. Se emplea **JWT** para manejar la autenticación.
- **Gestión de usuarios:**
 - Crear usuarios: POST /trabajador/create-user/
 - Obtener lista de usuarios: GET /trabajador/
 - Consultar información de un usuario: GET /trabajador/{id}/
 - Actualizar información de usuario: POST /trabajador/create-user/

Gestión de jornadas de trabajo

- Permite a los supervisores y administradores administrar las jornadas activas y finalizadas:
 - Obtener todas las jornadas: GET /jornada-trabajo/
 - Obtener una jornada específica: GET /jornada-trabajo/{id}/
 - Obtener jornadas activas: GET /jornada-trabajo/usuario/{rut}/

Registro y consulta de mediciones

- Captura de datos ambientales y de movimiento del trabajador:
 - Obtener mediciones ambientales por jornada: GET /mediciones-ambientales/jornada/{jornadaId}/
 - Obtener últimas mediciones por trabajador: GET /mediciones-ambientales/last/{rut}/
 - Obtener mediciones inerciales: GET /mediciones-inerciales/

Gestión de alertas y notificaciones

- Registra y administra alertas generadas automáticamente según los umbrales de seguridad establecidos:
 - Obtener todas las alertas: GET /solicitudes/
 - Obtener alertas por RUT: GET /solicitudes?rut={rut}/
 - Obtener detalle de una alerta: GET /solicitudes/{id}/
 - Obtener alertas de Firebase: GET /alertas-firebase/
 - Crear alerta en Firebase: POST /alertas-firebase/
 - Eliminar alertas en Firebase: DELETE /alertas-firebase/{id}/

Gestión de dispositivos (Respiradores y Filtros)

- Permite el registro y administración de los dispositivos de protección:
 - Obtener filtros disponibles: GET /tipo-filtro/
 - Obtener imagen de un filtro: GET /tipo-filtro/{id}/images/
 - Agregar nuevo filtro: POST /tipo-filtro/
 - Actualizar filtro: POST /tipo-filtro/
 - Eliminar filtro: DELETE /tipo-filtro/{id}/
 - Obtener respiradores disponibles: GET /tipo-respirador/
 - Obtener imagen de un respirador: GET /tipo-respirador/{id}/images/
 - Agregar nuevo respirador: POST /tipo-respirador/
 - Actualizar respirador: POST /tipo-respirador/
 - Eliminar respirador: DELETE /tipo-respirador/{id}/

Gestión de configuraciones y umbrales

- Permite la configuración de umbrales para alertas de respiración, ajuste y batería:
 - Guardar configuración de umbrales: POST /alertas-umbrales/
 - Obtener última configuración de umbrales por trabajador: GET /alertas-umbrales/trabajador/last/{rut}/

Gestión de ubicaciones y roles

- Administra los lugares de trabajo y los roles asignados a cada usuario:
 - Obtener todas las ubicaciones: GET /ubicacion/
 - Obtener una ubicación específica: GET /ubicacion/{id}/
 - Agregar ubicación: POST /ubicacion/

- Actualizar ubicación: POST /ubicacion/
- Eliminar ubicación: DELETE /ubicacion/{id}/
- Obtener todos los roles: GET /rol/
- Obtener un rol específico: GET /rol/{id}/
- Agregar rol: POST /rol/
- Actualizar rol: POST /rol/
- Eliminar rol: DELETE /rol/{id}/

Estrategia de Seguridad y Autenticación

Para garantizar la seguridad de los datos y el acceso restringido según los permisos de cada usuario, la API implementa:

- **Autenticación con JSON Web Tokens (JWT):** Para validar sesiones de usuario de manera segura.
- **Protección de endpoints con withCredentials: true:** Asegurando que solo usuarios autenticados puedan acceder a información confidencial.
- **Encriptación de contraseñas:** Se emplea **bcrypt** para almacenar credenciales de forma segura en la base de datos.
- **Restricción de acceso según roles:** Cada usuario tiene permisos definidos (Administrador, Supervisor) que limitan sus acciones dentro de la plataforma.

El diseño de la API garantiza una comunicación eficiente entre los diferentes componentes del sistema, asegurando que los datos capturados por los sensores sean almacenados y gestionados de forma segura. La estructura modular permite una fácil escalabilidad, facilitando futuras mejoras o integraciones con nuevas tecnologías.

3.2.4 Interfaz de usuario

La interfaz de usuario de la plataforma fue desarrollada con Angular, siguiendo un enfoque modular para garantizar una navegación intuitiva y una experiencia de usuario optimizada. La arquitectura del frontend se basa en la separación de componentes, asegurando reutilización, escalabilidad y mantenibilidad.

Arquitectura de la Interfaz de Usuario

La interfaz está organizada en componentes independientes que interactúan con la API REST para recuperar y actualizar datos en tiempo real. Se emplea Angular Material para proporcionar una apariencia moderna y accesible, mejorando la usabilidad para supervisores y administradores.

El frontend se estructura en módulos principales que reflejan las funcionalidades clave de la plataforma:

1. Módulo de autenticación

- Inicio de sesión con credenciales de usuario.
- Validación y gestión de sesiones mediante JWT.
- Redirección a la pantalla principal tras autenticación exitosa.

2. Módulo de monitoreo y supervisión

- Resumen: Vista principal con métricas sobre trabajadores activos, respiradores disponibles y alertas registradas.
- Estado de cuadrilla: Presenta en detalle el estado de los trabajadores asignados a cada supervisor.
- Historial de alertas: Permite filtrar alertas por tipo y estado.

3. Módulo de gestión de datos

- Gestión de trabajadores: Creación, edición y eliminación de trabajadores en la plataforma.
- Gestión de umbrales: Configuración de valores críticos para la detección de alertas en los respiradores.
- Gestión de dispositivos: Administración de filtros y respiradores utilizados en la plataforma.

4. **Módulo de configuración y soporte**

- Roles y ubicaciones: Asignación de permisos y configuración de las ubicaciones laborales.
- Sección de ayuda: Preguntas frecuentes y sistema de tickets para resolución de problemas.

Flujo de Datos y Comunicación con la API

El frontend se comunica con la API REST a través del servicio `data.service.ts`, que gestiona todas las peticiones HTTP. La estructura sigue el patrón de diseño basado en servicios, centralizando la obtención y envío de datos.

- RxJS y Observables: Permiten manejar datos reactivos, asegurando que la interfaz se actualice dinámicamente sin necesidad de recargas de página.
- Manejo de errores: Se implementa interceptores para capturar errores en las respuestas de la API, mejorando la experiencia del usuario.
- Autenticación y permisos: Se emplea Guards de Angular para restringir el acceso a ciertas vistas según el rol del usuario.

Diseño Modular y Escalabilidad

La arquitectura modular del frontend facilita la adición de nuevas funcionalidades sin afectar el rendimiento del sistema. Cada módulo está encapsulado, lo que permite su actualización o expansión sin interferir en otros componentes.

- Separación de responsabilidades: Cada funcionalidad se implementa en un componente independiente, lo que mejora la mantenibilidad del código.

Experiencia del Usuario y Accesibilidad

Para mejorar la interacción con la plataforma, se han incorporado diversas estrategias de diseño:

- Angular Material: Se usan componentes visuales predefinidos para proporcionar una interfaz consistente e intuitiva.

- Simplicidad en la navegación: Se organiza la información en menús laterales y vistas tabulares para facilitar el acceso a las funciones principales

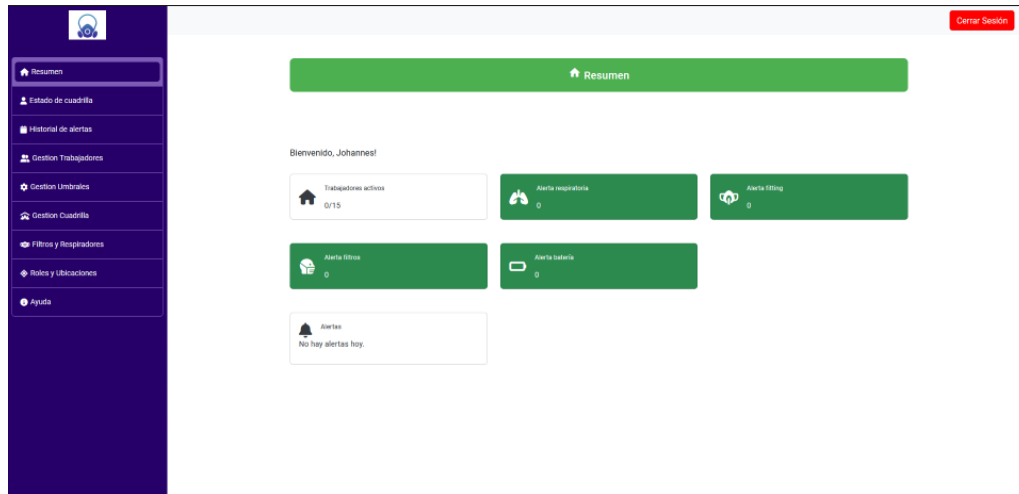


Imagen 4: Captura de pantalla del dashboard principal de la plataforma

3.2.5 Diseño de seguridad

La seguridad es un aspecto clave en la plataforma, ya que se maneja información sensible relacionada con la salud y el desempeño laboral de los trabajadores. Se han implementado diversas estrategias para garantizar la protección de los datos, el control de acceso y la detección de actividades sospechosas.

1. Seguridad en la Comunicación

Para proteger la transmisión de datos entre el frontend y el backend, se han implementado las siguientes medidas:

- Cifrado HTTPS: Toda la comunicación entre los distintos módulos del sistema se realiza a través del protocolo HTTPS, asegurando la privacidad e integridad de los datos transmitidos.
- Uso de Headers de Seguridad: Se configuran cabeceras HTTP como Strict-Transport-Security para prevenir accesos no seguros.

2. Seguridad en la Autenticación y Autorización

La autenticación y gestión de permisos en la plataforma están diseñadas para garantizar que solo los usuarios autorizados puedan acceder a funciones específicas:

Autenticación basada en cookies:

- Al iniciar sesión, el servidor genera un token de acceso (JWT) que se almacena en una cookie segura (httpOnly y SameSite=strict).
- En cada solicitud, la cookie es enviada automáticamente en los headers para validar la identidad del usuario sin exponer el token en el frontend.
- Este enfoque protege contra ataques de robo de tokens y reduce vulnerabilidades de seguridad.

Gestión de roles y permisos:

Se han definido roles específicos con distintos niveles de acceso:

- **Administrador:** Gestiona trabajadores, cuadrillas, filtros y respiradores, y configura umbrales.
- **Supervisor:** Puede visualizar información relevante de los trabajadores, pero sin modificar configuraciones globales ni gestionar dispositivos.

La plataforma incorpora medidas de seguridad robustas para proteger la comunicación, controlar el acceso y monitorear actividades sospechosas.

Gracias a la autenticación basada en cookies seguras (httpOnly, SameSite=strict), HTTPS y un sistema de roles bien definido, se garantiza un entorno seguro y eficiente para los usuarios.

3.3 Implementación

La implementación de la plataforma web para el monitoreo de respiradores industriales se basó en un enfoque modular, asegurando escalabilidad, seguridad y eficiencia en la integración con la API y la base de datos. Esta sección detalla los principales aspectos del desarrollo, centrándose en la arquitectura de la interfaz web, la comunicación con

el backend y la estructuración del código para garantizar mantenibilidad y expansibilidad.

3.3.1 Desarrollo de la interfaz web

El frontend de la plataforma fue desarrollado en **Angular**, adoptando el patrón de arquitectura basado en componentes, lo que permite una organización clara y reutilizable del código.

Estructura Modular del Frontend

El desarrollo de la interfaz se estructuró en módulos independientes que encapsulan funcionalidades específicas. Entre los principales módulos se encuentran:

- **Módulo de autenticación:** Controla el acceso de usuarios mediante autenticación JWT, validando credenciales y asignando permisos según el tipo de usuario.
- **Módulo de monitoreo:** Permite a supervisores y administradores visualizar el estado de los trabajadores en tiempo real y recibir alertas basadas en datos de sensores.
- **Módulo de gestión de datos:** Administra información sobre trabajadores, dispositivos (filtros y respiradores), roles y ubicaciones.
- **Módulo de configuración:** Proporciona opciones avanzadas para la personalización de umbrales de alertas y parámetros del sistema.

Comunicación con la API

El frontend interactúa con la API REST mediante un **servicio centralizado en Angular**, que gestiona la comunicación con el backend a través de peticiones HTTP (GET, POST, DELETE).

Para garantizar una comunicación fluida y eficiente con la API, se implementaron:

- **Manejo de peticiones asíncronas con RxJS:** Uso de **Observables** para actualizar dinámicamente la interfaz sin necesidad de recargar la página.
- **Autenticación y control de permisos:** Implementación de AuthGuards para restringir el acceso a secciones de la plataforma según el rol del usuario.
- **Manejo de errores:** Se desarrolló un **interceptor HTTP** para capturar respuestas erróneas y proporcionar mensajes detallados al usuario.



Imagen 5: Árbol de navegación web. Para una mejor visualización, se encuentra disponible en el **Anexo [2]** en tamaño completo.

3.3.2 Integración de la API

La plataforma web se integra con una API externa, la cual proporciona los servicios necesarios para la autenticación de usuarios, la consulta de información sobre trabajadores y dispositivos, el monitoreo de alertas y la recuperación de datos operativos en tiempo real. Dado que esta API no fue desarrollada dentro del presente proyecto, su integración se basó en la documentación proporcionada, la cual especifica los endpoints disponibles, los parámetros requeridos y los formatos de respuesta.

Consumo de Endpoints desde el Frontend

Para la comunicación con la API, se implementó un servicio centralizado en Angular que gestiona las solicitudes HTTP, asegurando una interacción eficiente entre la interfaz de usuario y el sistema de datos. Se utilizaron los métodos **GET** y **POST**, ya que no se realizaron modificaciones directas ni eliminaciones de datos para evitar la generación de información aislada o la pérdida de referencias entre registros relacionados.

El flujo de consumo de datos se desarrolló de la siguiente manera:

1. **Autenticación:** Los usuarios deben iniciar sesión en la plataforma, tras lo cual reciben un token de seguridad que es enviado en cada solicitud posterior para validar el acceso.
2. **Consulta de datos:** La plataforma obtiene información relevante sobre trabajadores, jornadas activas, alertas y dispositivos mediante los endpoints documentados.
3. **Registro de nuevas entradas:** En los casos en que se requiere almacenar información, como la creación de nuevos registros en el sistema, se utilizan solicitudes POST.
4. **Procesamiento de respuestas:** Los datos obtenidos de la API son transformados y presentados en la interfaz de usuario, asegurando que la información sea clara y utilizable para supervisores y administradores.
5. **Manejo de errores:** Se implementaron mecanismos para detectar errores en las respuestas de la API, permitiendo mostrar mensajes informativos a los usuarios y evitando interrupciones en la operatividad de la plataforma.

Pruebas de Integración con Postman

Para validar la correcta integración con la API, se realizaron pruebas utilizando **Postman**, lo que permitió verificar la estructura de las solicitudes y respuestas antes de implementarlas en la plataforma.

Las pruebas incluyeron:

- **Autenticación:** Validación del proceso de inicio de sesión y persistencia del token de seguridad.
- **Consulta de datos:** Verificación de la correcta recepción de información sobre trabajadores, dispositivos y alertas.
- **Registro de nuevos datos:** Prueba del almacenamiento de información en los módulos que lo requieren, asegurando la coherencia entre registros.

A través de estas pruebas, se ajustó el manejo de datos en el frontend para garantizar que las interacciones con la API se realizaran de manera eficiente y que la información mostrada en la plataforma estuviera alineada con los valores esperados.

Con esta integración, la plataforma permite a supervisores y administradores acceder en tiempo real a datos críticos, facilitando la gestión operativa y la toma de decisiones basada en información confiable y actualizada.

3.3.3 Elaboración del manual de usuario

El manual de usuario tiene como objetivo proporcionar una guía clara sobre el uso de la plataforma, asegurando que los administradores y supervisores puedan operar eficientemente cada una de sus funcionalidades.

Metodología de Diseño

Para garantizar su efectividad, el manual se diseñó bajo los siguientes principios:

1. **Claridad y accesibilidad:** Se empleó un lenguaje técnico pero comprensible para usuarios con distintos niveles de experiencia.
2. **Organización modular:** Se estructuró en secciones alineadas con las funcionalidades de la plataforma.
3. **Soporte visual:** Se incorporaron capturas de pantalla y ejemplos prácticos para ilustrar cada procedimiento.
4. **Enfoque práctico:** Se incluyeron pasos detallados para cada acción, asegurando una rápida comprensión.

Contenido del Manual

El documento se organiza en las siguientes secciones:

- **Introducción:** Explicación general del propósito de la plataforma.
- **Interfaz de Usuario:** Descripción de la estructura del sistema y acceso a funcionalidades.
- **Guía de Uso:** Explicación detallada de cada módulo, con instrucciones paso a paso.
- **Gestión de Usuarios y Dispositivos:** Procedimientos para agregar, editar y eliminar trabajadores y dispositivos.
- **Historial de Alertas:** Uso del sistema de alertas y su clasificación.
- **Configuración de Parámetros:** Definición de umbrales y ajustes en el sistema.
- **Soporte y Preguntas Frecuentes:** Resolución de problemas comunes.

El manual ha sido validado por un investigador del proyecto, garantizando su precisión y utilidad.

4. Evaluación

El sistema será evaluado mediante dos métodos reconocidos de usabilidad para determinar su puntuación:

1. **Heurísticas de Jakob Nielsen:** Evaluación basada en 10 principios de usabilidad para identificar problemas en la interfaz.
2. **System Usability Scale (SUS):** Encuesta estandarizada que permite medir la percepción del usuario sobre la facilidad de uso de la plataforma.

Los resultados de estas evaluaciones permitirán determinar el nivel de usabilidad del sistema y generar mejoras en futuras versiones.

Participantes en la Evaluación

La evaluación de la plataforma fue realizada por los siguientes miembros clave del equipo de desarrollo:

- Johannes Bunster: Desarrollador del equipo, encargado de la integración y las funcionalidades del sistema.
- Daniela Pinto: Responsable del desarrollo de la aplicación móvil vinculada a la plataforma.
- Francisco Pastene: Miembro del equipo y líder del proyecto, aportando una visión integral sobre los procesos.
- Aníbal Morales: Encargado de la parte de integración de sistemas, contribuyendo a la validación de la operatividad del sistema.

Estos participantes entregaron retroalimentación basada en su experiencia técnica y en su interacción directa con la plataforma.

Heurísticas de Jakob Nielsen

Se aplicó una evaluación basada en los 10 principios de usabilidad de Jakob Nielsen, utilizando una escala de 1 a 5, donde:

- 1: Problema menor, no afecta la usabilidad.
- 2: Problema leve, se puede corregir fácilmente.
- 3: Problema moderado, afecta la experiencia de usuario.
- 4: Problema severo, impide la correcta interacción.
- 5: Problema crítico, requiere corrección inmediata.

Heurística	Puntaje (1-5)
Visibilidad del estado del sistema	3
Relación entre el sistema y el mundo real	2
Control y libertad del usuario	2.25
Consistencia y estándares	2.25
Prevención de errores	3.25
Reconocimiento en lugar de recuerdo	2.25
Flexibilidad y eficiencia de uso	2.5
Diseño estético y minimalista	3
Reconocimiento, diagnóstico y recuperación de errores	3.5
Ayuda y documentación	3

Análisis de Resultados:

- Las heurísticas mejor evaluadas fueron 'Relación entre el sistema y el mundo real' y 'Reconocimiento en lugar de recuerdo', lo que indica que estos aspectos presentan menos problemas y están bien implementados en la plataforma.

- Las heurísticas con puntajes más altos, como 'Reconocimiento, diagnóstico y recuperación de errores' y 'Prevención de errores', evidencian áreas donde se requiere optimización para facilitar la interacción del usuario y minimizar errores.

System Usability Scale (SUS)

Se presentó una encuesta con 10 preguntas, puntuadas del 1 (totalmente en desacuerdo) al 5 (totalmente de acuerdo).

1. Me gustaría usar esta plataforma frecuentemente.
2. Considero que la plataforma es fácil de usar.
3. Creo que necesitaría soporte técnico para utilizar la plataforma.
4. Encontré que las funciones de la plataforma están bien integradas.
5. Creo que hay demasiada inconsistencia en la plataforma.
6. Pienso que los usuarios aprenderán a usar la plataforma rápidamente.
7. Encontré la plataforma confusa.
8. Me siento seguro usando la plataforma.
9. Necesité aprender muchas cosas antes de usar la plataforma.
10. Estoy satisfecho con la experiencia de uso.

Cálculo del puntaje SUS:

$(\text{Suma de respuestas impares} - 5) + (25 - \text{suma de respuestas pares}) \times 2.5$

Escala de 0 a 100:

- 80-100: Excelente
- 68-79: Bueno
- 50-67: Aceptable
- <50: Necesita mejoras

Se aplicó la encuesta estándar del SUS con 10 preguntas. El cálculo final del SUS se realizó utilizando la fórmula establecida, obteniendo un puntaje promedio de **52.5**, lo que sitúa a la plataforma en un rango **aceptable** dentro de la escala estándar de usabilidad.

Visualización del Resultado del SUS:

Según la escala de aceptabilidad del SUS, un puntaje entre 50 y 67 se considera aceptable, lo que indica que la plataforma en su estado actual cumple con los criterios básicos de usabilidad, aunque aún existe margen de mejora para optimizar la experiencia de usuario.

Relación entre los Métodos de Evaluación:

Ambos métodos identifican áreas de mejora, especialmente en la prevención y recuperación de errores, pero también destacan fortalezas como la claridad en la interacción y la relación intuitiva con el entorno. Estos resultados refuerzan la necesidad de iteraciones para optimizar la experiencia de usuario y fortalecer aspectos clave del diseño.

De acuerdo con los hallazgos, se sugiere considerar futuras mejoras orientadas a la simplificación de flujos críticos dentro de la plataforma, reduciendo los pasos necesarios para completar tareas recurrentes. Asimismo, se recomienda fortalecer los mecanismos de retroalimentación visual y auditiva, asegurando que las respuestas del sistema ante acciones del usuario sean claras, inmediatas y comprensibles. Esto facilitará una experiencia más fluida y reducirá posibles errores de navegación.

Otra área que considerar es la mejora en la personalización de la interfaz, permitiendo que los usuarios puedan ajustar ciertas configuraciones según sus preferencias, lo que contribuirá a una mayor adaptabilidad y satisfacción en el uso continuo. Además, sería pertinente incorporar evaluaciones periódicas de usabilidad con usuarios reales, enfocadas en validar los cambios implementados y detectar nuevas oportunidades de optimización.

Finalmente, se plantea reforzar las estrategias de accesibilidad para garantizar que la plataforma sea inclusiva y cumpla con estándares internacionales, considerando, por

ejemplo, la mejora en el contraste visual, opciones de navegación asistida y la simplificación de los textos informativos.

Estas acciones contribuirán a una mejora continua del sistema, promoviendo una experiencia de usuario más eficiente, intuitiva y adaptable a distintos contextos de uso.

5. Discusión

El desarrollo de la plataforma de monitoreo para respiradores industriales representa un avance significativo en la gestión y protección de la salud ocupacional en entornos de alto riesgo. Durante el proceso de implementación, se lograron integrar múltiples tecnologías para garantizar una solución funcional y efectiva. Sin embargo, también se presentaron desafíos que requirieron soluciones específicas.

5.1. Ventajas obtenidas

El uso de una plataforma web para la gestión de alertas y monitoreo en tiempo real ha proporcionado diversas ventajas, entre las que destacan:

- **Centralización y accesibilidad de datos:** La plataforma permite la visualización y gestión de información de manera estructurada, garantizando que supervisores y administradores accedan a datos actualizados desde cualquier dispositivo con conexión a la plataforma.
- **Monitoreo proactivo:** La generación de alertas automáticas permite detectar condiciones críticas como baja batería, fallas en el ajuste del respirador o saturación del filtro, lo que facilita una respuesta rápida y efectiva.
- **Gestión eficiente de alertas:** La plataforma proporciona herramientas de filtrado y búsqueda, lo que agiliza la identificación de eventos críticos sin necesidad de revisar manualmente grandes volúmenes de información.
- **Base sólida para futuras mejoras:** Los resultados obtenidos en las evaluaciones, en la escala SUS y en las heurísticas de Nielsen, validan la funcionalidad de la plataforma, resaltando que es una buena base para futuras optimizaciones.

5.2. Dificultades enfrentadas

El desarrollo de la plataforma no estuvo exento de desafíos. Algunos de los más relevantes incluyeron:

- **Integración de tecnologías:** Integrar múltiples tecnologías, como Angular para el frontend, un API para el backend y MySQL para la base de datos, fue un proceso complejo que requirió de un enfoque coordinado y detallado para asegurar que todos los componentes funcionaran de manera armónica. Este desafío fue superado mediante la implementación de pruebas de integración y revisiones constantes del código.
- **Gestión de errores y seguridad:** Manejar de forma adecuada los errores en las peticiones HTTP y garantizar la seguridad de los datos fueron aspectos críticos que demandaron una atención especial. Se implementaron prácticas de seguridad como la encriptación de contraseñas con bcrypt y el uso de JWT para la autenticación de usuarios, lo que fortaleció la protección de la información.
- **Percepción de usabilidad:** Si bien la evaluación SUS arrojó un promedio de 52.5, considerado aceptable, se identificó la necesidad de optimizar la experiencia para usuarios con menor experiencia técnica.
- **Trabajo en equipo e integración personal:** Aunque el equipo siempre mostró una buena disposición para resolver dudas y ayudar en el proceso de integración, a nivel personal resultó desafiante adaptarse a la dinámica de trabajo y al ritmo de desarrollo del proyecto. Esta situación requirió un esfuerzo adicional en términos de comunicación y colaboración para alinearse con los objetivos y estándares del equipo.

5.3. Lecciones aprendidas

El proceso de desarrollo dejó aprendizajes clave, no solo en el ámbito técnico, sino también en la gestión del trabajo en equipo y la adaptabilidad en entornos de desarrollo colaborativo. Algunas de las lecciones más relevantes fueron:

- **Planificación estructurada:** La integración de múltiples tecnologías requiere una fase de diseño bien definida desde el inicio, lo que facilita la implementación y reduce el margen de errores en etapas avanzadas.
- **Adaptabilidad y mejora continua:** La iteración y la retroalimentación constante permitieron realizar ajustes y optimizaciones progresivas, demostrando que la flexibilidad es clave en proyectos de software.

- **Importancia de la comunicación y el trabajo en equipo:** Aunque hubo dificultades en la integración personal al equipo, fue fundamental desarrollar habilidades de comunicación efectiva y buscar activamente la colaboración con otros integrantes para alinearse con el flujo de trabajo. La buena disposición del equipo fue un factor clave para superar este desafío y garantizar el éxito del proyecto.
- **Desarrollo de autonomía y autogestión:** En un entorno de trabajo colaborativo, resulta esencial aprender a tomar decisiones de manera autónoma y gestionar de forma eficiente el tiempo y las tareas asignadas para mantener el ritmo del proyecto.

6. Conclusión

El desarrollo de la plataforma de monitoreo para respiradores industriales ha cumplido con los objetivos planteados, proporcionando una herramienta funcional para la gestión de alertas y la supervisión de las condiciones laborales. La integración tecnológica logró establecer una solución robusta y efectiva, mientras que las evaluaciones aplicadas ofrecieron una visión clara sobre el estado actual de la plataforma.

El puntaje promedio de 52.5 en la escala SUS ubicó la plataforma en el rango aceptable, y los resultados obtenidos en las heurísticas de Nielsen permitieron identificar áreas de fortaleza, como la accesibilidad de datos y la gestión eficiente de alertas, así como áreas donde es necesario optimizar la experiencia de usuario.

Estos resultados validan que la plataforma es funcional en su versión actual y constituye una base sólida para futuras mejoras. En versiones posteriores se recomienda enfocar los esfuerzos en optimizar los procesos de recuperación de errores, la navegación intuitiva y la accesibilidad general del sistema.

En conclusión, la plataforma desarrollada facilita la toma de decisiones informadas por parte de supervisores y administradores, y fortalece la prevención de riesgos laborales. Su funcionalidad actual contribuye significativamente a la seguridad de los trabajadores en entornos industriales, y su diseño flexible la posiciona como una solución adaptable y mejorable en el futuro.

7. Referencias

En el desarrollo de este proyecto se consultaron y utilizaron diversas fuentes bibliográficas y documentación técnica que sirvieron de apoyo tanto para el marco teórico como para la implementación de la plataforma. Las referencias más relevantes son las siguientes:

- [1] Organización Internacional del Trabajo (OIT). *Seguridad y Salud en el Trabajo: Un derecho fundamental*. Disponible en: <https://www.ilo.org/es/temas-y-sectores/seguridad-y-salud-en-el-trabajo>
- [2] SOAINT. *Automatización en la Minería: Clave para la eficiencia y la seguridad*. Disponible en: <https://soaint.com/blog/sector-minero-automatizacion-eficiencia>
- [3] Hexagon. *Mine Monitoring Solutions*. Disponible en: <https://hexagon.com/es/solutions/mine-monitoring>
- [4] Ferrit. *Los sistemas de monitoreo para equipos mineros ayudan a garantizar que las máquinas funcionen mejor*. Disponible en: <https://www.ferrit.cz/es/actualidades/los-sistemas-de-monitoreo-para-equipos-mineros-ayudan-a-garantizar-que-las-maquinas-funcionen-mejor>
- [5] Kinsta. *Angular vs React: Una Comparación En Profundidad*. Disponible en: <https://kinsta.com/es/blog/angular-vs-react/>
- [6] Angular Minds. *The Best Choice for Your Project: Angular Material vs Bootstrap*. Disponible en: <https://www.angularminds.com/blog/the-best-choice-for-your-project-angular-material-vs-bootstrap>
- [7] BrowserStack. *Angular vs React vs Vue: Una Comparación Completa*. Disponible en: <https://www.browserstack.com/guide/angular-vs-react-vs-vue>
- [8] MuleSoft. *What is REST API design?*. Disponible en: <https://www.mulesoft.com/api/rest/what-is-rest-api-design>
- [9] ELA Innovation. *What is a REST API and why is it useful?*. Disponible en: <https://elainnovation.com/en/what-is-a-rest-api-and-why-is-it-useful/>
- [10] Amazon Web Services. *GraphQL vs REST API - Difference Between API Design Architectures*. Disponible en: <https://aws.amazon.com/compare/the-difference-between-graphql-and-rest/>
- [11] Google Cloud. *MySQL*. Disponible en: <https://cloud.google.com/mysql?hl=es>

[12] DB-Engines. *DB-Engines Ranking*. Disponible en: <https://db-engines.com/en/ranking>

[13] Akano, O. A., Hanson, E., Nwakile, C., & Esiri, A. E. (2024). *Designing real-time safety monitoring dashboards for industrial operations: A data-driven approach*. *Global Journal of Research in Science and Technology*, 2(2), 1–9. Disponible en: https://www.researchgate.net/publication/385214020_Designing_real-time_safety_monitoring_dashboards_for_industrial_operations_A_data-driven_approach

[14] Deloitte. *Predictive Maintenance and The Smart Factory*. Disponible en: <https://www2.deloitte.com/us/en/pages/operations/articles/predictive-maintenance-and-the-smart-factory.html>

[15] Honeywell. *Enabling the Connected Worker of the Future*. Disponible en: <https://www.honeywellforge.ai/us/en/whitepaper/enabling-the-connected-worker-of-the-future>

[16] TE Connectivity. *Sensores para el cuidado de las vías respiratorias*. Disponible en: <https://www.te.com/es/industries/medical-technologies/medical-products-components/medical-sensors/respiratory-care.html>

[17] MSA Safety. *Detectores Portátiles*. Disponible en: <https://mx.msasafety.com/Detectores-Port%C3%A1tiles/c/114?locale=es>

[18] Sanalife. *¿Qué es el estándar de OSHA para la calidad del aire?*. Disponible en: <https://www.sanalifewellness.com/es/blog/what-is-the-osh-standard-for-air-quality>

[19] Instituto de Seguridad del Trabajo (IST). *Decreto Supremo N° 594: Reglamento sobre Condiciones Sanitarias y Ambientales Básicas en los Lugares de Trabajo*. Disponible en: <https://www.ist.cl/wp-content/uploads/2016/08/DECRETO-SUPREMO-594-ACTUALIZADO-2019.pdf>

8. Anexos

Los anexos presentan materiales complementarios que detallan y respaldan el desarrollo y diseño de la plataforma web de monitoreo de respiradores industriales. A continuación, se listan los elementos incluidos:

[1] Diagrama Relacional de la base de datos

- El diagrama muestra la estructura de la base de datos, incluyendo las principales tablas como **JORNADA_TRABAJO**, **TRABAJADOR**, **MEDICIONES_AMBIENTALES**, junto con sus relaciones.
- Representa las claves primarias y foráneas, asegurando la integridad referencial y la correcta gestión de los datos en la plataforma.

[https://drive.google.com/drive/folders/1e2USQFTsZeGMrbl_MLwFO98M56h4HiYV?usp=drive link](https://drive.google.com/drive/folders/1e2USQFTsZeGMrbl_MLwFO98M56h4HiYV?usp=drive_link)

[2] Árbol de navegación web

- El árbol de navegación representa la estructura jerárquica de la plataforma web, detallando las principales secciones y subsecciones disponibles para los diferentes tipos de usuarios.
- Facilita la comprensión del flujo de navegación, permitiendo identificar cómo se accede a cada funcionalidad dentro del sistema.

[https://drive.google.com/file/d/1-JYgLEy8wc3l_Hld2GOz-RH-C3YxEvrF/view?usp=drive link](https://drive.google.com/file/d/1-JYgLEy8wc3l_Hld2GOz-RH-C3YxEvrF/view?usp=drive_link)

[3] Capturas de pantalla de la interfaz de la plataforma

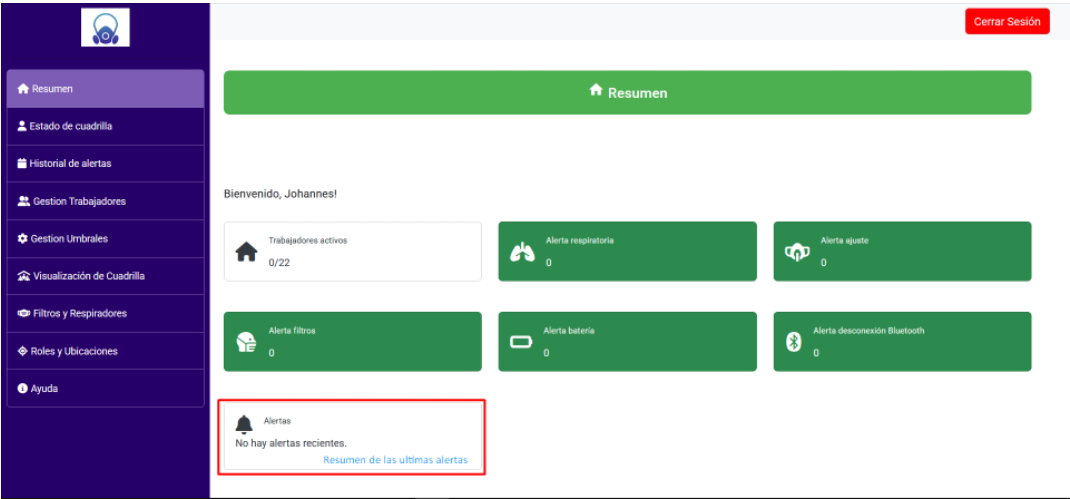


Figura 3.1: Sección resumen

The screenshot shows the 'Estado de cuadrilla' section of the platform. The sidebar is the same as in Figure 3.1. The main content area has a light gray header with a 'Cerrar Sesión' button. Below the header is a green bar with a team icon and the text 'Estado de cuadrilla'. The main content area displays a table with the following data:

RUT	Nombre	Alerta Ajuste	Alerta Filtros	Alerta Respiratoria	Alerta Batería	Conexión Sensor	Conexión Internet
19808621-5	Johannes Bunster Darras					0	
20994770-6	Pau Gatica Salgado					0	
7917258-8	Lucía Henríquez Valajoy					0	
14374430-2	Cesar Inostroza Robles					0	
20651373-1	Paula Inostroza Pallaluel					0	
19855441-3	Bastian Moraga Ramirez					0	
20161887-8	Valeria Moraga Ramirez					0	
15179437-6	Andrés Morales Montecinos					0	
9208849-9	Jorge Nitalos Pastene					0	
19823174-6	Melissa Norambuena Fuentesalba					0	
20818830-3	German Norambuena Fuentesalba					0	
21726530-4	Joaquín Norambuena Fuentesalba					0	
9620205-9	German Norambuena Lagos					0	
18808038-3	Daniel Pastene Pinto					0	
18817472-8	Francisco Pastene Henríquez					0	
8653928-4	Ivan Pastene Riera					0	
19091866-1	Romina Pinto Salgado					0	
11947796-4	Escarlet Ramirez Molina					0	
1881743-6	Prueba Rut inventado					0	
18817473-6	Prueba Rut inventado2					0	
18430511-9	Jaime San Martín Torres					0	
18413238-9	Martin Westermeyer Gandonneau					0	

Figura 3.2: Sección estado de cuadrilla

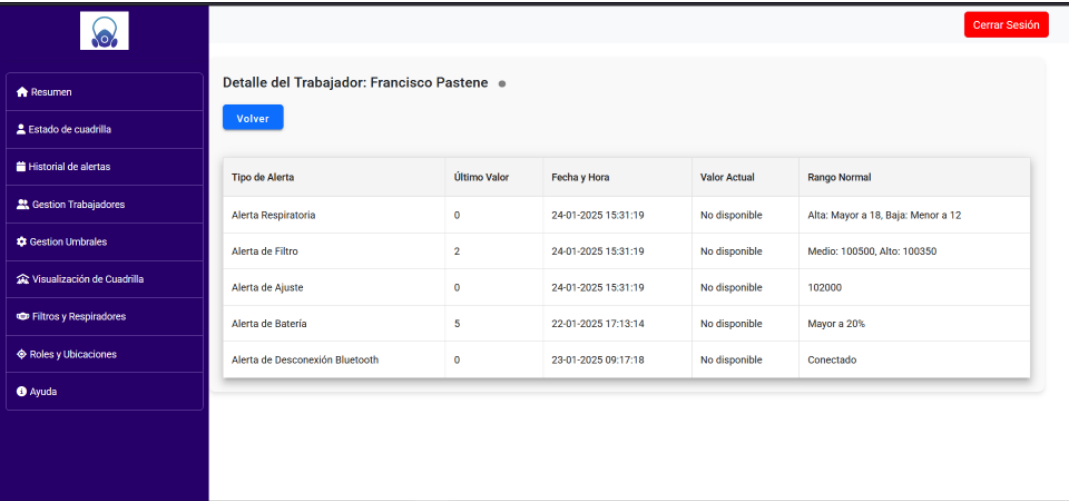


Figura 3.3: Detalles del trabajador

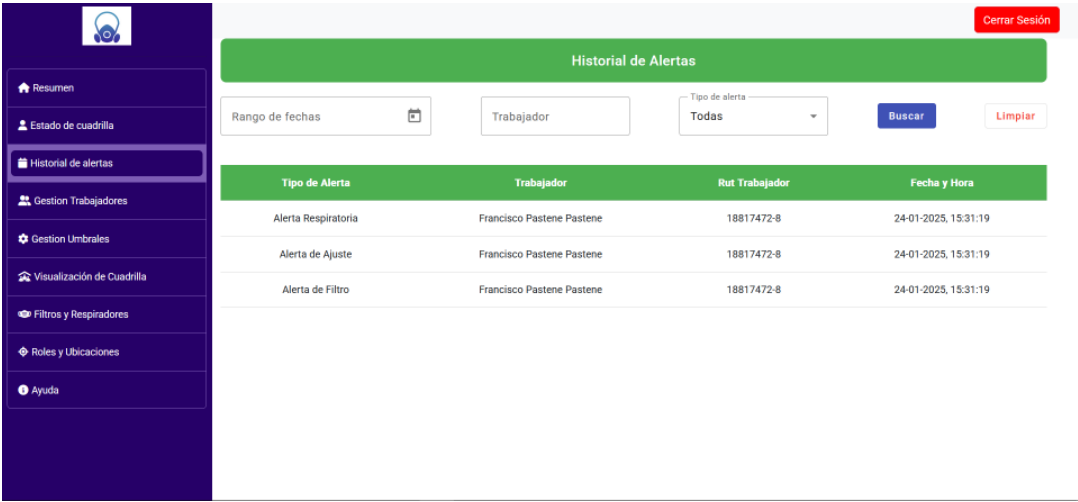


Figura 3.4: Sección historial de alertas

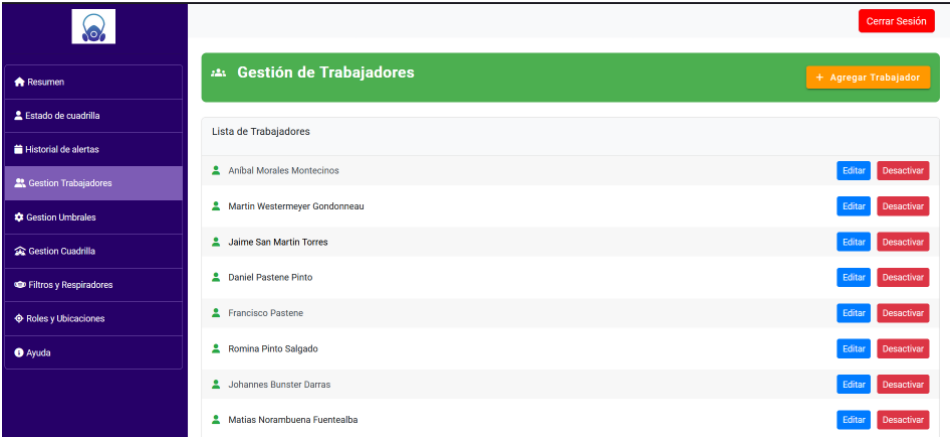


Figura 3.5: Sección gestión de trabajadores

Agregar Trabajador

Agregar Trabajador

RUT*

Nombres*

Primer Apellido*

Segundo Apellido*

Correo Electrónico*

Clave*

Cargo*

Agregar

Cancelar

Figura 3.6: Formulario para agregar trabajador

Editar Trabajador

 Editar Trabajador

RUT*

19823174-6

Nombres*

Matias

Primer Apellido*

Norambuena

Segundo Apellido*

Fuentealba

Correo Electrónico*

matias.norambuena23@gmail.com

Cargo*

Trabajador

Actualizar

Cancelar

Figura 3.7: Formulario para editar trabajador



Resumen

Estado de cuadrilla

Historial de alertas

Gestion Trabajadores

Gestion Umbrales

Gestion Cuadrilla

Filtros y Respiradores

Roles y Ubicaciones

Ayuda

Configurar Umbrales de Alertas

Seleccione un Trabajador Activo

Umbrales Respiratorios

Umbra1 Respiratorio Alto*

18

Umbra1 Respiratorio Bajo*

12

Umbrales de Ajuste

Umbra1 de Ajuste*

102000

Umbrales de Filtros

Umbra1 Filtro Alto*

100500

Umbra1 Filtro Medio*

100350

Umbrales de Bateria

Umbra1 Bateria Alto*

20

Umbra1 Bateria Medio*

10

Umbra1 Bateria Bajo*

5

Guardar Umbrales

Figura 3.8: Gestión de umbrales

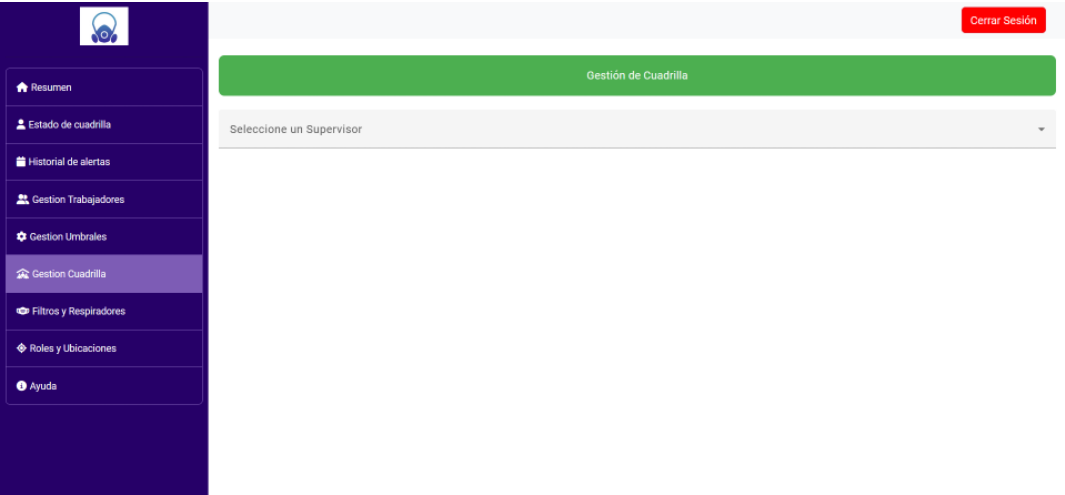


Figura 3.9: Sección gestión de cuadrilla



Figura 3.10: Supervisor seleccionado y trabajadores asignados

Resumen

Estado de cuadrilla

Historial de alertas

Gestion Trabajadores

Gestion Umbrales

Gestion Cuadrilla

Filtros y Respiradores

Roles y Ubicaciones

Ayuda

Gestión de Filtros y Respiradores

Filtros Disponibles

Agregar Filtro

Marca	Modelo	Fecha de Homologación	Editar	Estado
3M	2071-pp95	21-08-2024		Deshabilitado
3M	2078-pp95	09-10-2024		Habilitado
3M	2097-P100	14-10-2024		Habilitado
3M	60923	11-10-2024		Habilitado
3M	6001	09-11-2024		Habilitado
3M	2096-P100	14-11-2024		Habilitado
3M	2096-P100V5	14-11-2024		Habilitado

Figura 3.11: Sección de filtros disponibles

Agregar Filtro

Marca*

Modelo*

Fecha de Homologación*
dd - mm - aaaa

Descripción

Imagen
Examinar... Ningún archivo seleccionado.

Agregar

Cancelar

Figura 3.12: Formulario para agregar filtro

Resumen

Estado de cuadrilla

Historial de alertas

Gestion Trabajadores

Gestion Umbrales

Gestion Cuadrilla

Filtros y Respiradores

Roles y Ubicaciones

Ayuda

Respiradores Disponibles

Agregar Respirador

Marca	Modelo	Fecha de Homologación	Editar	Estado
3M	52P71	21-08-2024		Habilitado
3M	5203	09-10-2024		Habilitado
3M	7502-37082	09-10-2024		Habilitado
3M	6200-07025	11-10-2024		Habilitado
3M	6300-07026	29-10-2024		Habilitado
3M	6291-07002	14-11-2024		Habilitado

Figura 3.13: Sección de respiradores disponibles

Agregar Respirador

Marca*

Fecha de Homologación*
dd - mm - aaaa

Modelo*

Descripción

Imagen

Ningún archivo seleccionado.

Figura 3.14: Formulario para agregar respirador

Gestión de Roles y Ubicaciones			
Roles Disponibles			
Agregar Rol			
Nombre	Descripción	Editar	Estado
ROL	encargado cementos		Habilitado
ROL 2	nuevo4		Habilitado
ROL 3	Operador Grúa Horquilla		Habilitado
ROL 4	Mantenedor de Estantes		Habilitado
ROL 5	Técnico eléctrico		Habilitado
ROL 6	Nuevo Rol		Habilitado
C4i	oficina c4i		Deshabilitado
Ingeniero de proyectos	Ingeniero 1		Habilitado

Figura 3.15: Sección de roles disponibles

Agregar Rol

Guardar

Cancelar

Figura 3.16: Formulario agregar rol

🔊

- 🏠 Resumen
- 👤 Estado de cuadrilla
- 📅 Historial de alertas
- 👥 Gestion Trabajadores
- 🏠 Gestion Umbrales
- 🏠 Gestion Cuadrilla
- 🔊 Filtros y Respiradores
- 🔧 Roles y Ubicaciones
- 📖 Ayuda

ROL	Nombre	Estado
ROL 6	Nuevo Rol	✔ Habilitado
C4i	oficina c4i	✔ Habilitado
Ingeniero de proyectos	Ingeniero 1	✖ Deshabilitado
		✔ Habilitado

Ubicaciones Disponibles

Agregar Ubicación

Nombre	Edificio	Piso	Descripción	Editar	Estado
Cuadra1	Ed1	2	1200	✏	✔ Habilitado
ubicacion2	1	0	descripcion	✏	✔ Habilitado
ubicacion1	1	0	descripcion	✏	✔ Habilitado
Sistemas	Edificio Sistemas	0	Informatica	✏	✔ Habilitado
ubicacion3	Ed3	3	nuevo	✏	✔ Habilitado
C4i	oficina	0	oficina C4i	✏	✔ Habilitado

Figura 3.17: Sección de ubicaciones disponibles

Agregar Ubicación

0
▼

Guardar

Cancelar

Figura 3.18: Formulario agregar ubicación

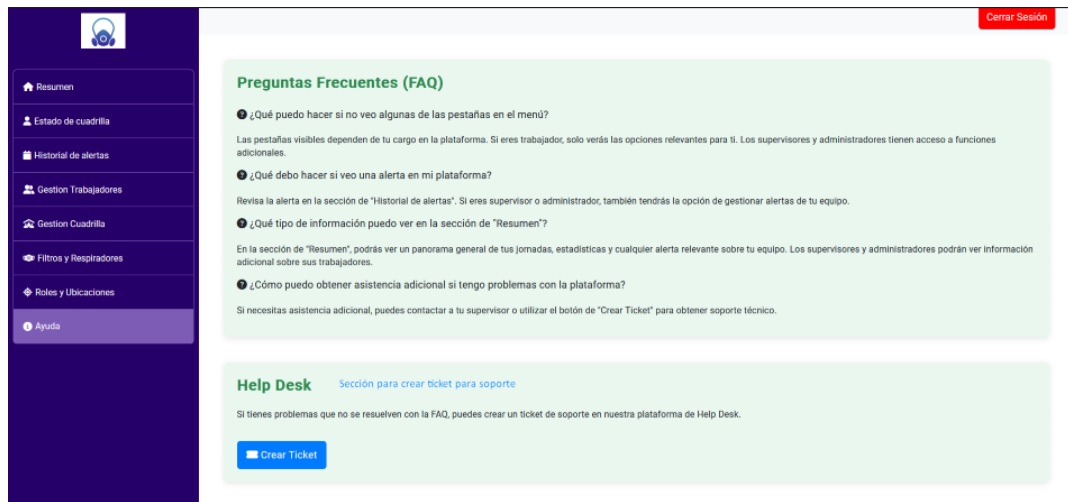


Figura 3.19: Sección de ayuda y preguntas frecuentes