

**Universidad de Concepción**  
**Departamento de Ingeniería Informática y Ciencias de la Computación**

**DIGITALIZACIÓN DE PROCESOS EN EMPRESA AVA MONTAJES: SEGURIDAD  
EN OPERACIONES**

Dazhi Enrique Feng Zong

Memoria presentada para la obtención del título de Ingeniero Civil Informático

Patrocinante: Marcela Varas C.  
Comisión Evaluadora: Geoffrey Hecht, Eduardo Méndez O.

Septiembre de 2024

# Índice

<b>Índice</b>	<b>2</b>
<b>Resumen</b>	<b>3</b>
<b>1. Introducción</b>	<b>4</b>
1.1. Antecedentes generales del problema	5
1.2. Solución propuesta y alcances	6
1.3. Objetivo general	6
1.4. Objetivos específicos	7
1.5. Metodología	7
1.6. Estructura del informe	9
<b>2. Descripción de la propuesta</b>	<b>10</b>
<b>3. Especificaciones y requerimientos del sistema</b>	<b>14</b>
<b>4. Funcionalidades principales</b>	<b>16</b>
<b>5. Arquitectura del sistema</b>	<b>17</b>
<b>6. Stack tecnológico</b>	<b>24</b>
<b>7. Diseño de la base de datos</b>	<b>32</b>
<b>8. Diagrama UML de casos de uso</b>	<b>37</b>
8.1. Documentación de casos de uso	37
<b>9. Desafíos en el desarrollo del sistema</b>	<b>44</b>
<b>10. Diseño de la aplicación móvil</b>	<b>49</b>
<b>11. Diseño de la aplicación web</b>	<b>53</b>
<b>12. Evaluación del sistema</b>	<b>60</b>
Resultados	60
<b>13. Conclusiones</b>	<b>63</b>
<b>14. Bibliografía</b>	<b>66</b>
<b>Anexos</b>	<b>69</b>
Historias de usuario por cada funcionalidad	69
Ejemplos de formularios de la empresa	74

# Resumen

El presente documento analiza y documenta la solución implementada para resolver la problemática de la ineficiencia y la falta de seguridad en la gestión de formularios de seguridad en AVA Montajes, una empresa chilena especializada en ingeniería, construcción y montaje industrial con un enfoque en el sector minero. La empresa, que actualmente realiza importantes proyectos para clientes como Codelco, BHP, Collahuasi y otras compañías mineras, está obligada por normativa y requisitos de seguridad a llenar formularios para registrar aspectos críticos de sus operaciones diarias. Sin embargo, la utilización de formularios en papel presenta problemas, tales como pérdida, deterioro y dificultad para recuperar la información, además de ineficiencias en la gestión del proceso.

La digitalización de los formularios se presenta como la solución para optimizar la eficiencia y la seguridad del proceso. Esta solución incluye la creación de formularios digitales que pueden ser accedidos y completados sin conexión a internet desde dispositivos móviles, como tablets y teléfonos. Las ventajas de esta solución incluyen la mejora en la legibilidad, la integridad y perdurabilidad de los datos, la facilidad de búsqueda y la posibilidad de adjuntar documentos de manera más eficiente. Además, la digitalización completa del proceso facilita la recolección de datos para análisis posteriores, permitiendo la generación de informes, reportes y estadísticas para análisis posteriores.

Para implementar esta solución, se propone el desarrollo de una aplicación web que permita a los administradores crear y publicar formularios digitales, así como una aplicación móvil que permita a los trabajadores completar estos formularios en terreno. La aplicación móvil debe ofrecer funcionalidades para funcionar en terreno, como validación de identidad, operación sin conexión a internet, y la capacidad de adjuntar fotos y dibujar firmas digitales.

El objetivo de esta memoria, en términos de implementación y desarrollo, es la creación de una aplicación móvil destinada a la visualización y respuesta de formularios digitales, así como la finalización de la aplicación web previamente desarrollada. Esta finalización incluye la integración de todas las funcionalidades restantes para la conexión con la aplicación móvil.

En cuanto al contenido del documento, se proporcionará un análisis y documentación completa del sistema, que incluye la descripción del problema, la solución propuesta, las especificaciones y requerimientos del sistema desarrollado, las funcionalidades principales, los diagramas de arquitectura del sistema y UML, el stack tecnológico utilizado, el diseño de la base de datos y de las aplicaciones, así como la evaluación del sistema realizada por los usuarios.

# 1. Introducción

Ingeniería AVA Montajes es una empresa chilena con más de 24 años de experiencia en el mercado, especializada en ingeniería, construcción y montaje industrial, enfocada en la industria minera tanto del ámbito público como privado del norte del país. Sus principales servicios incluyen:

- **Ingeniería:** Diseño, estudio, desarrollo y gestión de proyectos de ingeniería.
- **Construcción:** Obras civiles, montaje electromecánico, piping y estructuras metálicas.
- **Montajes:** Montaje de equipos industriales, maquinaria y piping.
- **EPC:** Ejecución de proyectos bajo la modalidad EPC (Engineering, Procurement and Construction), donde se encargan de todo el ciclo del proyecto, desde la ingeniería hasta la construcción y entrega final.
- **Renta de equipos:** Alquiler de maquinaria y equipos para la industria minera e industrial.
- **Mantenimiento:** Mantención preventiva y correctiva de instalaciones industriales.

Actualmente, la empresa está involucrada en varios proyectos con diferentes clientes, incluyendo:

- **CODELCO:** La empresa está trabajando en varios proyectos para diferentes divisiones de CODELCO, tales como Chuquicamata, Ministro Hales, Radomiro Tomic y Andina.
- **BHP**
  - Proyecto Spence: Construcción, montaje, y mantenimiento en las instalaciones de la mina Spence.
- **COMPAÑÍA MINERA DOÑA INÉS DE COLLAHUASI**
  - Proyecto Quinto Molino: Excavación, construcción de cimientos, cajón de pulpa y losas de fundación.
  - Proyecto Collahuasi: Diseño, adquisición, construcción y puesta en marcha de un tercer Espesador de Relaves.
  - Proyecto Collahuasi 14: Construcción de un By Pass.
- **BHP MINERA ESCONDIDA**
  - Proyecto Minera Escondida: Reemplazo de dos alimentadores de tensión.
- **SIERRA GORDA**
  - Proyecto Sierra Gorda: Construcción e instalación de tuberías, montaje de colectores de polvo, filtros y cañerías.

Para garantizar la seguridad de sus trabajadores y cumplir con las regulaciones legales, la empresa debe implementar un estricto control en sus procesos diarios. Una parte esencial de estas operaciones son los formularios de seguridad, que se utilizan en todas las actividades que realiza la empresa como excavaciones, soldaduras, construcciones, etc. Estos formularios incluyen aspectos importantes como el estado del equipo y/o las herramientas, la condición del trabajador, la identificación de riesgos y otros elementos relevantes. Actualmente, la empresa completa todos los formularios en papel; sin embargo, esta práctica resulta poco eficiente, especialmente en entornos mineros donde el ambiente puede dificultar su completado, gestión y recuperación.

Por lo tanto, se hace evidente la necesidad de digitalizar estos formularios para mejorar su eficiencia, optimizar su gestión y facilitar la toma de decisiones más certeras. La digitalización implica crear formularios en formato digital y completarlos utilizando dispositivos móviles como tablets y celulares.

En esta memoria, se presenta un análisis y documentación detallada sobre la solución desarrollada para digitalizar estos procesos, facilitando la creación, respuesta, almacenamiento y recuperación de los formularios de manera digital, mejorando así, la eficiencia y seguridad en las operaciones.

## 1.1. Antecedentes generales del problema

Para desarrollar el sistema que necesita la empresa, se identificaron tres antecedentes principales luego de analizar sus necesidades y especificaciones:

### 1. Digitalización del proceso de diseño, creación y respuesta de formularios

Para digitalizar completamente el proceso de formularios, es necesario poder diseñar y crear formularios digitales personalizables a partir de los formularios en formato físico. Estos formularios físicos de la empresa son muy estrictos y tienen distintos formatos bien definidos con varios elementos. Posteriormente a la creación, estos formularios deben poder ser respondidos de manera digital, digitalizando todo el proceso: creación, respuesta, almacenamiento, corrección, firmas, etc. El objetivo es eliminar la necesidad de realizar cualquier tarea en formato físico en todas las etapas del proceso.

### 2. Funcionamiento sin internet y sincronización de datos

Otro punto importante es que los formularios se completan en entornos mineros en el norte, donde es poco probable que haya internet. Por esta razón, se decidió desarrollar una aplicación móvil que guarde los datos localmente. Estos datos deben poder ser enviados al servidor y recibir datos del servidor cuando haya conexión a internet. Es importante que haya sincronización de datos cada vez que haya internet disponible y que la aplicación funcione completamente sin necesidad de conexión. La creación de formularios puede hacerse en una oficina con conexión a internet y luego cargar estos datos a la aplicación móvil. Pero, la respuesta y el almacenamiento del formulario digital deben poder realizarse sin conexión a internet, y que, al recuperar la conexión, se sincronicen con el servidor para evitar la pérdida de datos en caso de accidentes con la tablet.

### 3. Escalabilidad a nivel de la empresa

Actualmente, este sistema está planeado y diseñado para dos departamentos, pero hay que tener en cuenta que toda la empresa utiliza formularios. Es muy probable que se necesite expandir y escalar estas aplicaciones de formularios digitales para que funcionen con casi todos los departamentos de la empresa, e incluso con los formularios de los clientes. Por lo tanto, es importante que el diseño de estas aplicaciones permita su escalabilidad sin necesitar de muchos cambios o un gran rediseño para adaptarse a las necesidades de otros departamentos.

## 1.2. Solución propuesta y alcances

La solución propuesta es digitalizar completamente el proceso de formularios, lo que incluye el diseño y la creación de formularios digitales, la respuesta a estos formularios y su almacenamiento en el servidor.

Para ello, se decidió crear una aplicación web para la creación de formularios, ya que estos deben ser diseñados por un cargo administrativo del propio departamento, como un jefe de departamento, y no el equipo de TI. Una aplicación web con una interfaz les permitirá y facilitará la creación del formulario, definiendo la estructura, el formato, las preguntas, etc. Dado que los formularios tienen varios elementos irregulares, se optó por desarrollar una aplicación personalizada para la empresa en lugar de adquirir software externo y adaptarse a él.

Para la respuesta de los formularios, considerando que los formularios físicos se responden en terreno sin acceso a internet, se decidió desarrollar una aplicación móvil que emule este proceso. Esta aplicación, diseñada para tablets, permite renderizar el formulario creado en la aplicación web para que pueda ser respondido y firmado sin necesidad de conexión a internet, guardando el formulario en el dispositivo y sincronizándolo con el servidor cuando se recupere la conexión.

Además, todos los formularios y sus respuestas deben almacenarse en el servidor, permitiendo una fácil búsqueda y recuperación de cualquier formulario. Aprovechando la digitalización, también se pretende desarrollar una sección que recopile los datos de los formularios respondidos para generar estadísticas, reportes y gráficos.

Este sistema está diseñado para ser flexible y adaptable, de modo que pueda ser utilizado por toda la empresa en el futuro. Por lo tanto, el diseño de la aplicación debe ser flexible y fácilmente adaptable para futuras necesidades.

Actualmente, la aplicación web ya está funcionando para la creación de formularios, pero falta completar la conexión con la aplicación móvil, además del apartado de estadísticas y gráficos. En cuanto a la aplicación móvil, se cuenta con un prototipo con funciones básicas y varias vistas listas. El objetivo es rediseñar algunas partes para que se conecten con la base de datos y la aplicación web, trayendo los formularios del servidor para renderizarlos y permitir su respuesta, y luego enviarlos al servidor para su almacenamiento permanente.

Todo esto está diseñado para un departamento de la empresa, pero con la intención de ser flexible y adaptable para que pueda expandirse y usarse en la mayoría de los formularios de todos los departamentos de la empresa.

## 1.3. Objetivo general

Optimizar el proceso de seguridad durante las operaciones en faena, mejorando la eficiencia y seguridad en la gestión de formularios mediante la digitalización.

## 1.4. Objetivos específicos

Se cuenta con un prototipo de la aplicación móvil que incluye algunas funciones básicas y una aplicación web casi completa, pero no adaptada para hacer la conexión con la aplicación móvil. En este contexto, los objetivos principales del proyecto son los siguientes:

1. Rediseñar y completar el desarrollo de la aplicación móvil e integrarla con la aplicación web, agregando todas las funcionalidades pendientes.
2. Implementar funcionalidades en la aplicación web que permitan visualizar los datos generados por la aplicación móvil, lo cual podría incluir gráficos, estadísticas, tablas comparativas, informes y otros elementos relevantes, según lo solicitado por la empresa. Además, implementar funciones restantes para que funcione con la aplicación móvil, como la creación y asignación de usuarios, la sincronización web y móvil, la funcionalidad de firmar, etc.
3. Realizar un análisis y documentar el software, incluyendo aspectos como el problema, soluciones, requisitos funcionales y no funcionales, evaluación de tecnologías, evaluación de alternativas, decisiones importantes tomadas durante el proceso, desafíos, arquitectura del sistema, diagramas, consideraciones, etc.
4. Investigar e implementar, si es posible, medidas para garantizar que el software sea utilizable y cumpla con los requisitos de la empresa, incluyendo cumplimiento legal, seguridad y otros aspectos relevantes.

Para cada objetivo se cumplió lo siguiente:

1. La aplicación móvil está casi completa y es utilizable para testear en terreno real. Solo resta agregar y arreglar algunas pequeñas funcionalidades, pero el flujo principal está completo.
2. Los objetivos para la aplicación web se cumplieron parcialmente. Al iniciar el desarrollo, quedaban pendientes varias funcionalidades por implementar. Más adelante, se logró implementar las funcionalidades esenciales para que la aplicación web funcione con la aplicación móvil y sea utilizable por los altos cargos. Sin embargo, lo que no se logró implementar fue la vista de dashboard con gráficos, estadísticas y reportes que muestren la reportabilidad de los formularios.
3. Se ha realizado un análisis y documentación de los aspectos más importantes del sistema en esta memoria.
4. Este objetivo era opcional y se dejó para el final del proyecto, luego de terminar el desarrollo del sistema. Como el desarrollo no ha terminado, este objetivo tampoco se cumplió. La investigación del cumplimiento legal, como la validación de las firmas, se está realizando por parte de la empresa con un abogado laboral.

## 1.5. Metodología

La metodología para desarrollar el sistema fue la siguiente:

## **Reunión inicial con clientes y usuarios**

Antes que todo, lo primero que se hizo es la reunión con el cliente, quien solicita y explica sus necesidades y detalles. En esta etapa se analizan los requerimientos del cliente, lo que necesita, la viabilidad del proyecto, el tiempo estimado para el primer prototipo, etc. Se llega a un consenso con el cliente sobre el alcance del sistema y sus funcionalidades, y se programa la fecha de la próxima reunión con el primer prototipo.

Luego, internamente con el equipo de desarrollo de TI, se realiza una reunión para analizar y discutir los puntos tratados en la reunión con el cliente. Se determina qué se va a hacer, cómo se va a hacer, el tiempo requerido, el diseño, el alcance, posibles problemas, etc.

## **Diseño del sistema**

Posteriormente, se diseña el sistema. Para la aplicación web, se busca que sea lo más amigable e intuitiva posible, considerando que los usuarios no son muy tecnológicos. Se diseñó algo similar a Google Forms, pero más simplificado, evitando muchos clics o elementos que causen confusión. La aplicación web fue diseñada en colaboración con Pablo Zapata Schifferli, alumno de la carrera, y con la opinión del ingeniero líder de desarrollos de la empresa, Pablo Peña Medina.

En el caso de la aplicación móvil, el diseño ya estaba casi completo, por lo que solo fue necesario imitar el estilo de lo diseñado, como los botones, las páginas, la tipografía, los colores, los contenedores, etc. y realizar algunos ajustes.

El enfoque principal en el diseño del sistema fue la simplicidad, claridad e intuición para el usuario, para que pueda usarlo sin complicaciones. El diseño propuesto se valida en una pequeña reunión con el equipo de desarrollo de TI. También se diseña la base de datos inicial del sistema y, una vez validado todo, se comienza el desarrollo.

## **Desarrollo del sistema**

Para el desarrollo del sistema, se utilizó principalmente la documentación de los frameworks, Laravel [2] y Flutter [3] para las consultas iniciales. Para cosas más específicas y puntuales, se recurrió a Google, ChatGPT, Gemini y Stack Overflow. En casos en los que estas herramientas no ofrecían una solución, se consultó con el ingeniero supervisor, Pablo Peña.

El uso de chatbots de inteligencia artificial resultó muy útil para acelerar el desarrollo, proporcionando respuestas rápidamente a consultas simples, correcciones de sintaxis y detección de errores. Sin embargo, para problemas más complejos, tuvo una efectividad más limitada.

Además, se consultaron algunos cursos en Udemy para implementar funcionalidades específicas. Por ejemplo, en uno de los capítulos se explicaba cómo implementar la captura de fotos o la selección desde la galería, una característica necesaria para la aplicación móvil.

Después de implementar una característica, se realiza una prueba manual para verificar que no rompa ni cause errores en otras partes del sistema. No se implementaron pruebas automatizadas con ningún framework.

### **Reuniones de revisión**

Cada semana se realiza una reunión con el equipo de desarrollo de TI para revisar, corregir, discutir y planificar las tareas de la semana siguiente, además de resolver dudas.

Cada cierto tiempo, dependiendo de la disponibilidad de los clientes, se realiza una reunión con ellos donde se hace una demostración completa del sistema, mostrando todas las funcionalidades actuales. El cliente y los usuarios dan feedback sobre lo que opinan, qué se necesita corregir o mejorar, y se discute la viabilidad y el tiempo requerido para implementar los cambios. Al final de la reunión se acuerda lo nuevo que tendrá el sistema para la próxima reunión.

### **Prueba del sistema desarrollado**

Finalmente, cuando el sistema cuenta con todas las funcionalidades esenciales, sin errores críticos, se despliega para que los usuarios (personal en faena) lo prueben. Mientras tanto, se sigue trabajando en mejoras y se recibe feedback para ajustar y mejorar el sistema basado en su uso real.

## **1.6. Estructura del informe**

A continuación, en esta memoria se presenta lo siguiente:

1. Contextualización y descripción detallada del problema actual relacionado con el completado y almacenamiento de los formularios en el norte.
2. Solución propuesta y las ventajas que trae esta.
3. Especificaciones y requerimientos del sistema desarrollado, identificados a partir de la problemática y algunos solicitados por la empresa.
4. Funcionalidades principales del sistema.
5. Arquitectura del sistema, representada mediante diagramas de los tres primeros niveles del modelo C4.
6. Stack tecnológico del sistema: Flutter, Laravel y MySQL, explicación de la elección del stack y alternativas consideradas o alternativas a considerar para migrar.
7. Diseño de la base de datos, explicando las tablas principales.
8. Diagrama UML de casos de uso del sistema, incluyendo la documentación de cada caso de uso principal.
9. Desafíos principales al desarrollar el sistema.
10. Diseño de la aplicación móvil.
11. Diseño de la aplicación web.
12. Evaluación del sistema mediante el test de usabilidad SUS, aplicado a trabajadores de la empresa.
13. Conclusiones.
14. Referencias.
15. Anexos, incluyendo un listado de historias de usuario para cada funcionalidad principal.

## 2. Descripción de la propuesta

### **Problemática: Ineficiencia e inseguridad en los procesos en faena**

La empresa AVA Montajes, debido a la naturaleza de su actividad y su estrecha relación con la industria minera, se encuentra obligada, tanto por razones legales como por la seguridad de sus trabajadores, a implementar un riguroso control en todos sus procesos.

En el desarrollo de las actividades diarias se sigue un proceso riguroso que comienza con el llenado de formularios de seguridad. Estos formularios incluyen aspectos fundamentales como el estado del equipo y/o herramientas que se utilizaran, la condición del trabajador, el entorno de trabajo, identificación de riesgos, entre otros aspectos relevantes (ejemplos de formularios pueden consultarse en la sección de anexos). Es común que un trabajador deba completar múltiples formularios durante el transcurso de un día laboral, llegando fácilmente a una decena de ellos. Una vez completado, un supervisor (generalmente el jefe del área) debe aprobar el formulario mediante su firma. Solo después de este paso se permite iniciar la tarea correspondiente.

Los formularios que se completan son esenciales en varias actividades operativas. Sin la aprobación de un formulario por parte de un supervisor, ciertas actividades no pueden llevarse a cabo. Su importancia aumenta aún más en situaciones de accidentes, ya que se convierten en un respaldo legal para la empresa. Por lo tanto, es muy importante que los formularios sean debidamente completados, almacenados y protegidos contra daños o extravíos.

Sin embargo, el uso de formularios en papel no resulta muy eficiente, especialmente en áreas como el norte del país, donde las condiciones climáticas pueden afectar la integridad de los documentos. Además, en estos entornos mineros en terreno, los formularios no se completan en una oficina organizada, donde todo permanece limpio y ordenado. Los formularios físicos están expuestos a arrugarse, extraviarse o dañarse, lo que afecta la legibilidad de la información registrada. Además, la gestión de estos formularios en formato físico dificulta su búsqueda y recuperación, ya que tienden a acumularse en gran cantidad, dificultando la localización de uno específico cuando se necesita.

Como se puede observar, una parte del proceso que resulta muy ineficiente y con falta de optimización es la gestión de formularios. Esta falta de optimización genera algunas ineficiencias en el proceso. Por ejemplo, si un formulario se completa incorrectamente o con errores, el supervisor que lo revisa puede corregirlo inmediatamente. Pero, en el peor de los casos si el formulario está muy mal completado o con una letra no legible, se debe rellenar un formulario completamente nuevo y llevarlo nuevamente al supervisor para su aprobación. Dado que se completan múltiples formularios a lo largo del día, estos pequeños retrasos se acumulan y resultan en una pérdida de tiempo e ineficiencia en el proceso completo.

Sin embargo, los problemas de seguridad son aún más preocupantes. Si los formularios completados no se almacenan adecuadamente, es fácil que se extravíen, arruguen, dañen, manchen, etc. Además, hay un riesgo de que alguien sobrescriba, modifique o raye la

información ingresada. Por último, otra ineficiencia importante es la dificultad para buscar un formulario cuando se necesita, considerando que son cientos; esto implica revisar uno por uno hasta encontrar el requerido.

Aunque la gestión de formularios puede parecer menos relevante en comparación con otras fases del proceso, sigue siendo una parte esencial que se encuentra descuidada y poco optimizada a pesar de ser altamente mejorable. Es fundamental garantizar el almacenamiento adecuado de los formularios para evitar pérdidas o daños, especialmente si se requiere buscar alguno en el futuro.

### **Solución: digitalización de formularios**

Como se mencionó anteriormente, una parte del proceso que claramente presenta deficiencias es la gestión de formularios. Aunque este descuido no sea crítico, la ineficiencia y la falta de seguridad en esta área afectan a toda la operación. Por suerte, esta situación puede ser mejorada mediante la siguiente solución: la digitalización.

La digitalización implica la creación de formularios en formato digital y su completado a través de dispositivos como tablets o teléfonos. Este enfoque ofrece diversas ventajas, entre las que se destacan:

1. **Facilidad de completado:** Todos los campos del formulario se completan de manera ordenada y sistemática, evitando omisiones y errores comunes asociados al llenado manual. Además, algunos campos, como el proyecto, la fecha, el área, entre otros, no necesitan ser completados manualmente, ya que la información se encuentra disponible y solo requiere ser cargada.
2. **Legibilidad garantizada:** La información registrada en los formularios digitales es clara y legible, eliminando el problema de la ilegibilidad de la escritura manual.
3. **Integridad y perdurabilidad:** Los formularios digitales no corren el riesgo de extraviarse, arrugarse o deteriorarse, asegurando la integridad de los datos a lo largo del tiempo.
4. **Facilidad de búsqueda:** La gestión y búsqueda de formularios se simplifica bastante mediante sistemas de almacenamiento y organización digital, permitiendo acceder rápidamente a la información necesaria.
5. **Integridad y seguridad:** La digitalización asegura que los formularios completados no puedan ser manipulados o alterados, a diferencia de los formularios llenados a mano. La información registrada en un formulario digital permanece inalterable, ya que no puede ser borrada o modificada.
6. **Disponibilidad (casi) inmediata de la información:** Con acceso a internet, los formularios se cargarán en la base de datos, lo que permite que la información esté disponible al instante para todos los usuarios autorizados.
7. **Adjuntar documentos fácilmente:** En comparación con los formatos físicos, adjuntar documentos es mucho más fácil digitalmente. No hay que buscar el documento físicamente y es especialmente útil para fotos recién tomadas, ya que pueden ser añadidas inmediatamente, algo que no sería posible físicamente.

Estas ventajas varían según la parte del proceso que se convierta en digital:

- **Digitalización del formulario respondido:** Garantiza la seguridad de la información, evitando su pérdida o deterioro.
- **Digitalización del proceso de completado del formulario:** Mejora la legibilidad, lo que aumenta la eficiencia y la rapidez del proceso. Con la eliminación de formularios ilegibles, se evita la necesidad de rechazar los documentos por esa razón y, en caso de correcciones, no es necesario volver a llenar todo el formulario como suele ocurrir con el papel.
- **Digitalización completa del proceso:** Aumenta significativamente la eficiencia al simplificar el flujo de trabajo: completar el formulario, firmarlo y guardarlo automáticamente. Además, la búsqueda y recuperación del formulario se vuelven mucho más sencillas. No hay necesidad de buscar dónde almacenar y recuperar, y el formulario será completamente legible y sin errores de llenado.

Evidentemente, digitalizar el proceso completo de formularios es la mejor opción, debido a sus numerosas ventajas. Además de optimizar la eficiencia, la seguridad de la información y la velocidad del proceso, beneficiando principalmente a los trabajadores y supervisores, esto también resulta sumamente beneficioso para los directivos y gerentes. Al digitalizar los formularios completados, se facilita la recolección de datos para su posterior análisis, lo que permite la generación de informes y evaluaciones. Estos datos pueden ayudar a detectar tendencias, identificar patrones y áreas de mejora en el proceso.

### ¿Cómo digitalizar?

Actualmente, la única etapa del proceso de formularios que está digitalizada ocurre después de completar el formulario. Una vez que el formulario ha sido aprobado y firmado, se captura una fotografía o se escanea y se guarda de forma digital en Google Drive. Aunque esta parte sea la más importante y ya esté digitalizada, el proceso sigue siendo muy ineficiente y tedioso. Completar el formulario de manera rápida y legible, que sea fácil de revisar y que se pueda guardar de forma segura y recuperar fácilmente, así como adjuntar rápidamente una foto al formulario, son áreas que se pueden mejorar significativamente.

Para digitalizar este proceso por completo, es necesario crear formularios en formato digital que puedan ser completados en dispositivos tecnológicos adecuados, como tablets, teléfonos o computadoras.

La idea inicial de la empresa era que el equipo de TI creará los formularios digitales, subiendo la información de la plantilla del formulario a la base de datos. Esto incluiría el título, el código del formulario, las preguntas, el tipo de respuesta para cada pregunta, las opciones de respuesta, etc. Luego, una aplicación móvil, a través de una API, recibiría todos estos datos y renderizaría el formulario en la interfaz de la aplicación móvil, permitiendo a los usuarios responderlo.

Sin embargo, debido a la considerable cantidad de formularios y su naturaleza cambiante, este enfoque resulta inviable. Dado que son numerosos formularios y se actualizan con frecuencia, es poco práctico que el departamento de TI, compuesto por dos personas, se encargue de

cargarlos y crearlos. Obviamente, asignar la tarea de cargar los formularios a la base de datos a una persona de cada departamento tampoco es una opción. Además de resultar poco amigable y demasiado complejo para ellos, otorgar acceso a tantas personas a la base de datos conlleva riesgos de seguridad, errores y latencia.

Por lo tanto, se necesita una solución que permita a los departamentos diseñar sus propios formularios de manera sencilla y flexible, basándose en los formularios en papel existentes. Surgió la idea de desarrollar una aplicación web donde usuarios con el rol de administrador puedan crear los formularios digitales según las necesidades específicas de cada departamento o proyecto. Esta herramienta permitirá definir el formato y los elementos de los formularios, como tablas, checklists, imágenes, preguntas, etc.

Una vez que el administrador haya finalizado la creación del formulario según sus necesidades, podrá publicarlo. Una vez publicado, el formulario no podrá ser modificado y estará disponible para que otros usuarios puedan responder directamente desde la aplicación. Este proceso es similar a la funcionalidad ofrecida por plataformas como Google Forms, donde un usuario crea un formulario personalizado y lo publica para que otros respondan. Pero, en este caso, la aplicación se adapta completamente a las necesidades y requerimientos específicos de la empresa.

Una vez creados los formularios en la aplicación web, además, es necesario que puedan ser completados a través de una aplicación móvil, considerando que los trabajadores en terreno no tienen acceso a un computador. Esta aplicación móvil deberá permitir visualizar y responder los formularios, además de contar con funcionalidades esenciales para trabajar en terreno como: la validación de la identidad del usuario, la capacidad de funcionar sin conexión a internet, adjuntar fotos tomadas con la cámara y la posibilidad de firmas digitales.

En esta memoria, el enfoque principal estará dirigido al desarrollo y funcionamiento de la aplicación móvil, así como en completar su integración con la aplicación web. La creación de formularios mediante la aplicación web ya ha sido completada; ahora se deben implementar las demás funcionalidades y finalizar el desarrollo completo de la aplicación móvil, incluyendo todas las características necesarias.

Finalmente, se requiere agregar características esenciales al software completo. Esto incluye aspectos relacionados con la seguridad del software, validación de identidad y la protección de la privacidad de los datos, así como garantizar el cumplimiento legal de la solución implementada.

A continuación, en este documento se detalla el análisis y documentación detallada del software, que incluye aspectos como los requisitos, especificaciones, funcionalidades principales, la elección de la tecnología utilizada, alternativas consideradas, arquitectura del sistema, diagramas, entre otros aspectos relevantes.

### 3. Especificaciones y requerimientos del sistema

#### Objetivo

Como se mencionó anteriormente, el objetivo es el siguiente: Optimizar el proceso de seguridad durante las operaciones en faena, mejorando la eficiencia y seguridad en la gestión de formularios mediante la digitalización.

#### Necesidades

Se puede identificar las siguientes necesidades principales:

- **Creación de formularios digitales:** Se necesita poder crear formularios digitales basados en los formularios originales. Esta creación debe ser fácil para el usuario que lo va a crear y debe permitir varias opciones de personalización para adaptarse a los formularios originales.
- **Aplicación móvil:** Se requiere una aplicación móvil diseñada para tablets que contenga todos los formularios digitalizados necesarios para los trabajadores y que estos puedan completarlos.
- **Funcionamiento offline:** Dado que la conexión en el área norte es mala, la aplicación debe funcionar sin necesidad de conexión a internet. Los usuarios deben poder visualizar y responder los formularios incluso en ausencia de conexión.
- **Validación y firma de formularios:** Una vez que se complete un formulario, debe ser posible validarlo y firmarlo por las personas correspondientes. Se debe garantizar que la firma sea realizada por el usuario autorizado.
- **Almacenamiento seguro y accesible:** Todo lo ingresado en los formularios, incluyendo imágenes, firmas y otros datos relacionados, debe ser guardado de manera segura y fácilmente accesible para su búsqueda y recuperación en caso de necesidad.
- **Sincronización de datos:** Considerando el funcionamiento offline, se necesita un mecanismo para sincronizar los datos cuando haya conexión a internet. Esto implica, por ejemplo, la capacidad de actualizar la aplicación con nuevos formularios desde la base de datos y enviar los datos ingresados al servidor una vez recuperada la conexión.
- **Reportabilidad:** Se requiere una aplicación web que utilice los datos recolectados para presentarlos de manera clara y amigable, utilizando herramientas como estadísticas y gráficos para su análisis.
- **Diseño y funcionamiento intuitivo y simple:** Dado el perfil de los usuarios, que en su mayoría son trabajadores de mayor edad y no están familiarizados con tecnología avanzada, se requiere un diseño y funcionamiento de la aplicación móvil que sea fácil de entender y utilizar, sin complicaciones innecesarias.
- **Seguridad de los datos:** Es fundamental que los datos almacenados estén protegidos contra accesos no autorizados y no sean modificables una vez guardados en la base de datos, asegurando la privacidad y confidencialidad de la información.

## **Tipos de usuario**

Se requieren e identifican los siguientes tipos de usuario:

- **Trabajador en terreno:** Este usuario utiliza la aplicación móvil para buscar y completar los formularios necesarios para la actividad. Al terminar de completar uno, debe firmarlo.
- **Supervisor en terreno:** Es responsable de revisar y firmar los formularios respondidos por los trabajadores bajo su supervisión, validando lo completado y autorizando las actividades correspondientes.
- **Jefe/Gerente:** Se encarga de crear los formularios digitales y los usuarios, además tiene acceso con privilegios en la aplicación web, donde puede acceder a informes y estadísticas detalladas generadas a partir de los datos recopilados de los formularios digitales.

## 4. Funcionalidades principales

Teniendo las necesidades y los tipos de usuario, podemos definir las funcionalidades principales del sistema y las historias de usuario para cada funcionalidad y tipo de usuario (en los anexos):

1. **Creación formulario digital:** Los formularios digitales serán creados utilizando una aplicación web desde una computadora, por un jefe de departamento u otro cargo similar.
2. **Responder formulario digitalmente:** El formulario es principalmente completado por el trabajador en terreno, utilizando una tablet, sin requerir conexión a internet, aunque ocasionalmente otros tipos de usuarios pueden completarlo. Todos los usuarios tienen la capacidad de responder a los formularios, pero la responsabilidad es de los trabajadores.
3. **Firmar digitalmente:** Para validar el formulario, se requiere la firma del trabajador que lo completó y la firma de un supervisor que revise y tome conocimiento.
4. **Reportabilidad:** El acceso al dashboard con reportes, gráficos, estadísticas, etc., está restringido a los jefes o gerentes y solo se puede realizar desde la aplicación web utilizando una computadora.
5. **Almacenamiento de los formularios** El almacenamiento de formularios es automático para cualquier persona que los complete, lo que les permite buscar y acceder fácilmente a sus formularios respondidos. Además, los empleados con cargos superiores tendrán la capacidad de buscar y acceder a los formularios completados por ellos mismos o por los empleados de menor cargo bajo su supervisión.
6. **Creación y asignación de usuarios:** Se requiere una interfaz que permita al jefe, gerente o supervisor crear un usuario en el sistema en caso de que no esté registrado en la base de datos. Además, debe ser posible asignarle un supervisor y los trabajadores que va a supervisar.
7. **Requisitos no funcionales:** Se definen otras características del sistema:
  - **Disponibilidad:** La aplicación móvil debe ser capaz de funcionar sin conexión a internet, permitiendo a los trabajadores completar formularios con conectividad limitada o nula.
  - **Seguridad:** Se debe proteger los datos almacenados en la aplicación, garantizando que solo los usuarios autorizados tengan acceso a la información y que los datos sean inalterables o muy difíciles de alterar una vez guardados.
  - **Sincronización:** La aplicación móvil debe tener un método de sincronización que permita actualizar los formularios y enviar los datos completados al servidor cuando haya conexión a internet.
  - **Usabilidad:** Tanto la aplicación móvil como la aplicación web deben ser intuitivas y fáciles de usar, con interfaces claras y navegación sencilla, especialmente diseñadas para usuarios que pueden no estar familiarizados con tecnología avanzada.
  - **Rendimiento:** Tanto la aplicación web como la aplicación móvil deben ser rápidas y eficientes en su funcionamiento, incluso en dispositivos y conexiones de baja capacidad.
  - **Compatibilidad:** La aplicación móvil debe ser compatible con distintos dispositivos (teléfonos y tablets) y sistemas operativos (Android y iOS).

# 5. Arquitectura del sistema

La arquitectura del sistema se representa mediante los siguientes diagramas C4 de los primeros tres niveles:

### Contexto

Como se puede observar en la figura 1, el sistema se compone de dos sistemas de software: el sistema interno de AVA Montajes, encargado de la gestión de formularios digitales, y un sistema externo basado en la nube llamado Bunny.net, utilizado para almacenar archivos sin la necesidad de mantenerlos en el sistema local. Además, se encuentran los tres tipos de usuarios mencionados anteriormente, cada uno con su función principal.

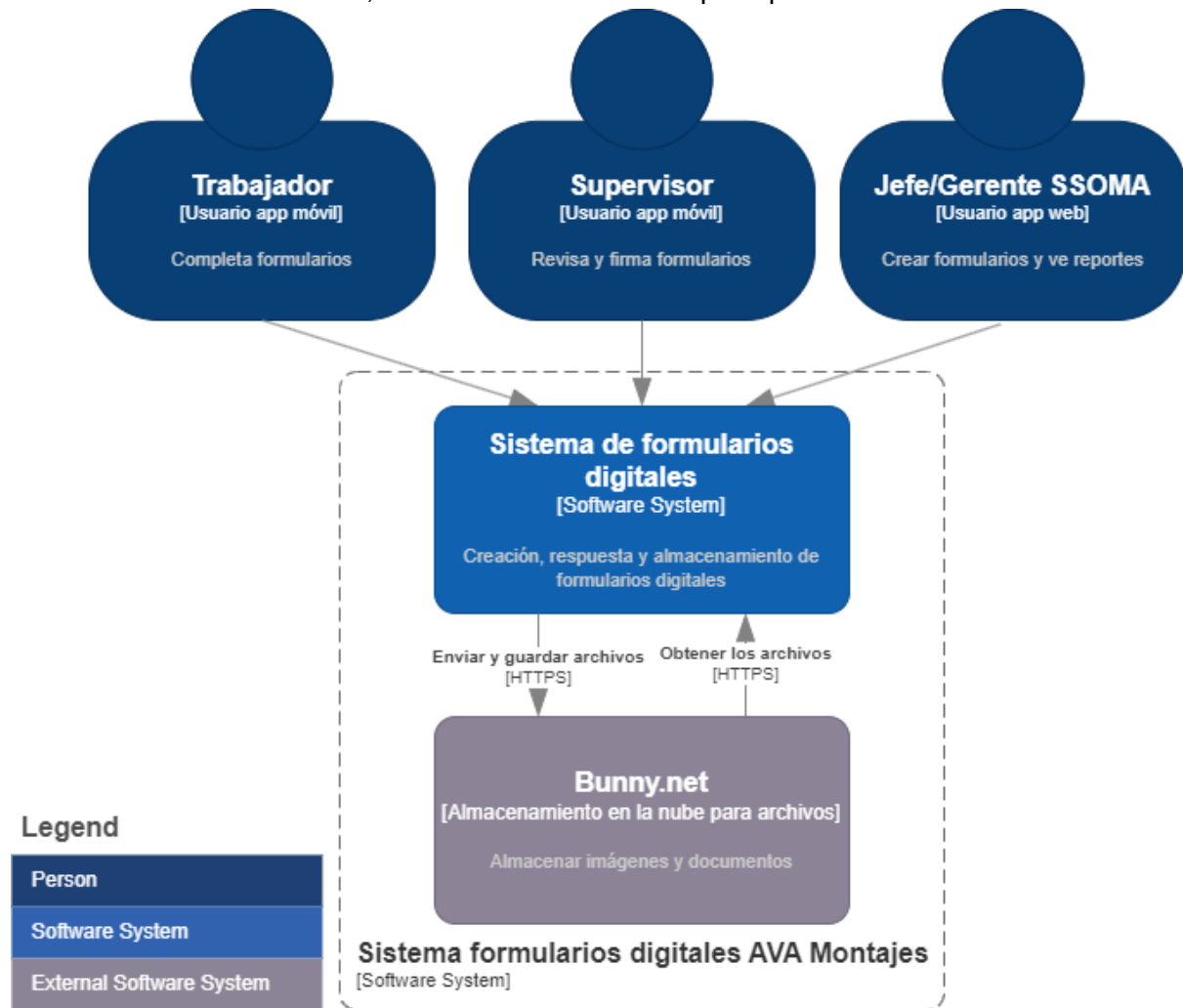


Figura 1. Diagrama de contexto del sistema

## Contenedores

El sistema se compone de cinco partes principales: aplicación web, aplicación móvil, API, servidor y el almacenamiento en la nube, como se muestra en la figura 2.

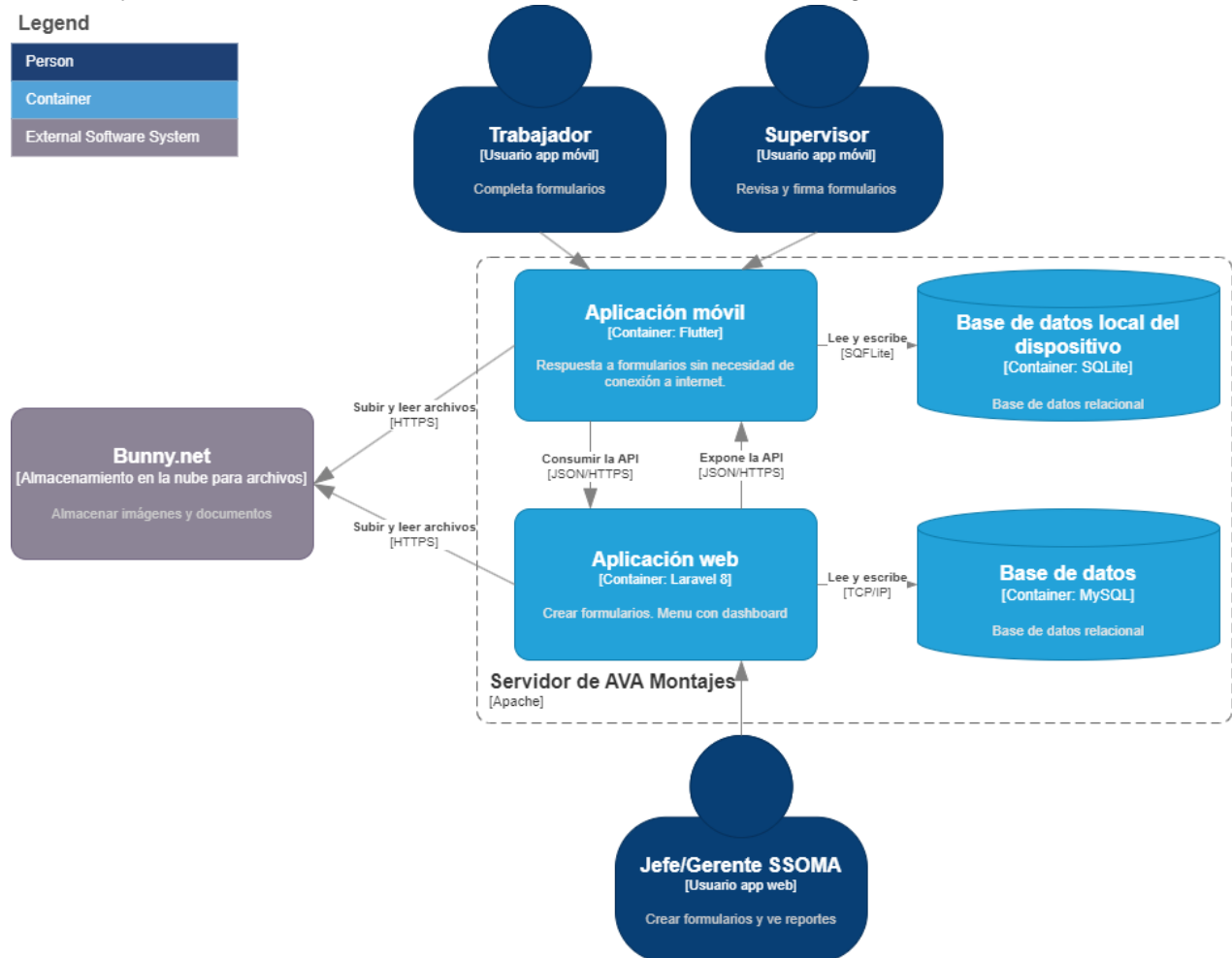


Figura 2. Diagrama de contenedores del sistema

### Aplicación móvil

Funcionalidades principales:

- Respuesta a formularios sin necesidad de conexión a internet.
- Sincronizar datos con el servidor cuando haya conexión disponible.
- Firma con validación de identidad.

Tecnologías:

- Flutter: Framework para el desarrollo de aplicaciones móviles, compatible con Android e iOS.
- SQLite: Se implementará una base de datos local para almacenar formularios sin conexión y sincronizarlos posteriormente.

### Aplicación web

Funcionalidades principales:

- Crear y firmar formularios

- Creación y asignación de usuarios.
- Menú con dashboard para reportabilidad.

Tecnologías:

- Laravel 8: Framework para el desarrollo del frontend y backend.
- Livewire: Utilizado en el frontend para facilitar la creación de vistas.
- AdminLTE (Bootstrap 4): Framework CSS.

## **API**

Funcionalidades principales:

- Facilita la comunicación entre la aplicación web y la aplicación móvil.
- Intercambio de datos entre la aplicación móvil y el servidor.

Tecnología:

- Laravel 8: La API se desarrollará dentro de la aplicación web.

## **Servidor**

Funcionalidades principales:

- Alojamiento y gestión de la aplicación web, la API y otros servicios.
- Gestión de la base de datos

Tecnología:

- Servidor Apache para alojar el sitio web.
- La base de datos del servidor será gestionada a través de MySQL
- Para la administración del servidor, se utiliza cPanel.

## **Almacenamiento en la nube para archivos**

Funcionalidades principales:

- Almacenar imágenes y documentos (por ejemplo, PDF).
- Permite la subida y acceso a estos archivos desde la aplicación web y la aplicación móvil, evitando cargar los archivos en la base de datos.

Tecnología:

- Bunny.net.

## Componentes aplicación móvil

El diagrama detallado del componente de la aplicación móvil se puede ver en la figura 3:

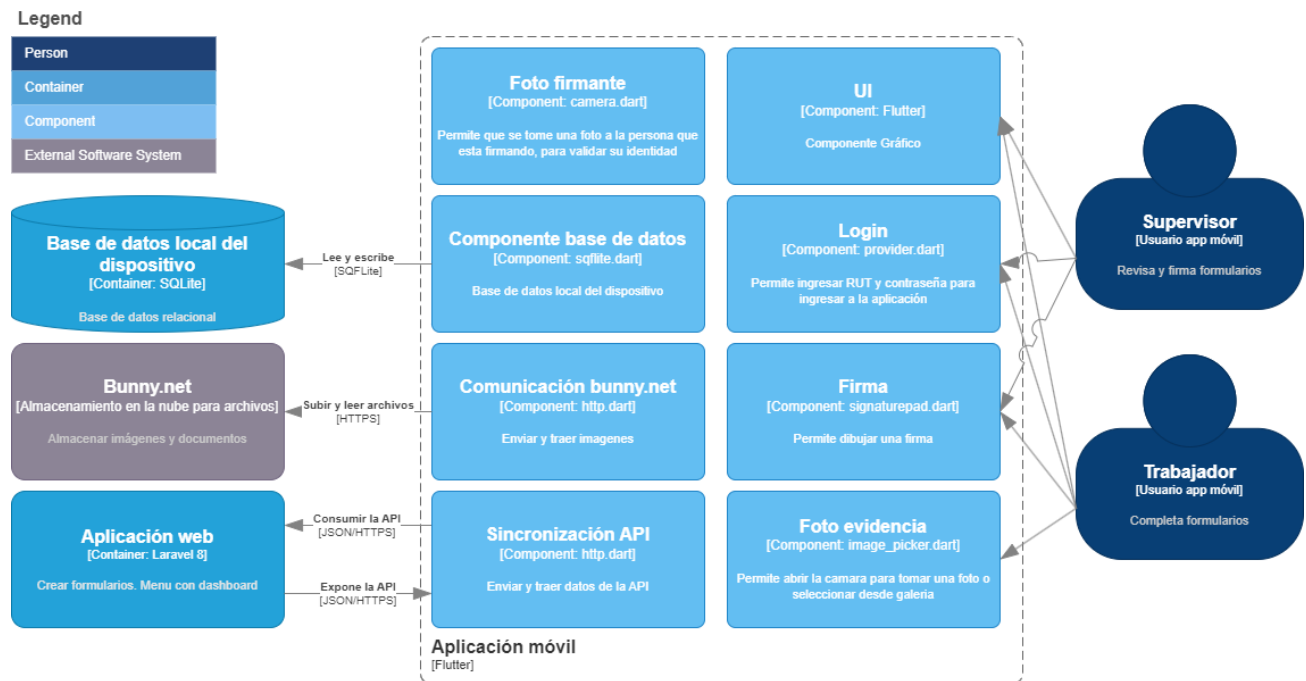


Figura 3. Diagrama de componente aplicación móvil

- Los usuarios interactúan con la aplicación móvil a través del componente de interfaz de usuario (UI) de la aplicación.
- Se tiene un componente para la interacción con la API y otro componente para la interacción con bunny.net. Ambos componentes utilizan el paquete http.dart para llevar a cabo comunicaciones HTTP.
- Se tiene otro componente para la comunicación con la base de datos SQLite del dispositivo, utilizando el paquete sqflite.dart
- Para gestionar el inicio de sesión y mantener la sesión activa, se tiene un componente que utiliza el paquete provider.dart.
- Para la etapa de firmas, se utilizan dos componentes distintos: uno permite la firma digital, mientras que el otro toma una foto del usuario que firma (similar al software de recursos humanos Buk, que toma una foto cuando alguien registra su asistencia). Para ello, se emplean los paquetes signaturepad.dart y camera.dart respectivamente.
- Además, hay otro componente para tomar fotografías para la evidencia o seleccionar imágenes desde la galería. Para esto, se utiliza el paquete image\_picker.dart.

## Componentes aplicación web

El diagrama detallado del componente de la aplicación web se puede ver en la figura 4:

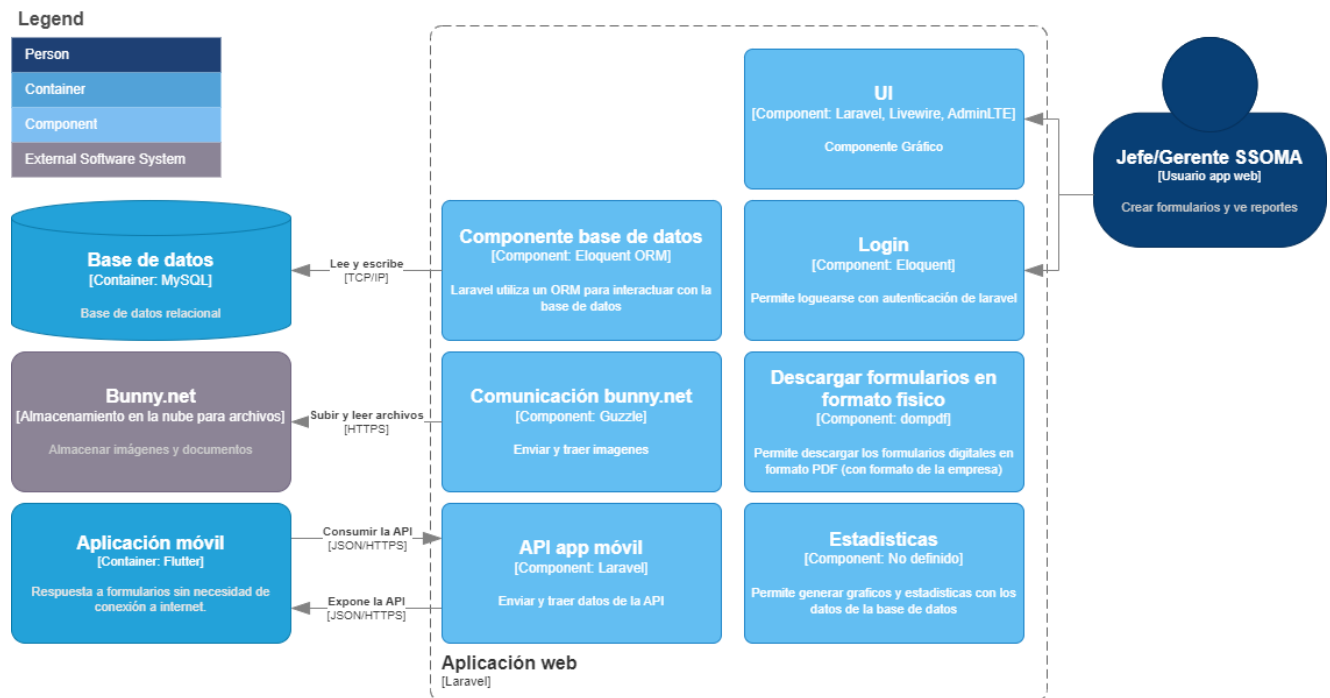


Figura 4. Diagrama de componente aplicación web

- Los usuarios interactúan con la aplicación web mediante su interfaz de usuario (UI). Para el desarrollo de los componentes de la aplicación web, se utilizó Livewire, que facilita la creación de componentes en PHP. Además, se utilizó AdminLTE, un framework CSS basado en Bootstrap 4, para estilizar la aplicación.
- El componente que expone la API para la aplicación móvil se implementó dentro de la aplicación web con Laravel.
- Para la comunicación con bunny.net, se utilizó la biblioteca Guzzle, que permite realizar solicitudes HTTP. Estas comunicaciones se realizan a través de HTTPS.
- Un componente importante es el encargado de interactuar con la base de datos del servidor. En este caso, se utiliza Eloquent, el ORM de Laravel, que simplifica la interacción con la base de datos MySQL. Estas comunicaciones se realizan mediante el protocolo TCP/IP.
- El componente de inicio de sesión también se desarrolla mediante Eloquent, aprovechando la autenticación integrada en Laravel. Laravel mantiene automáticamente la sesión activa mediante cookies.
- Un componente esencial es la capacidad de generar formularios en formato físico en PDF. Para ello, se utiliza la biblioteca dompdf, que permite generar archivos PDF a partir de HTML y CSS.
- El componente de estadísticas para generar un dashboard con gráficos, estadísticas y reportes de los formularios respondidos no ha sido implementado aún. Se planea utilizar Chart.js integrado con AdminLTE para esta funcionalidad.

## Componentes base de datos servidor

El diagrama detallado del componente de la base de datos del servidor se puede ver en la figura 5:

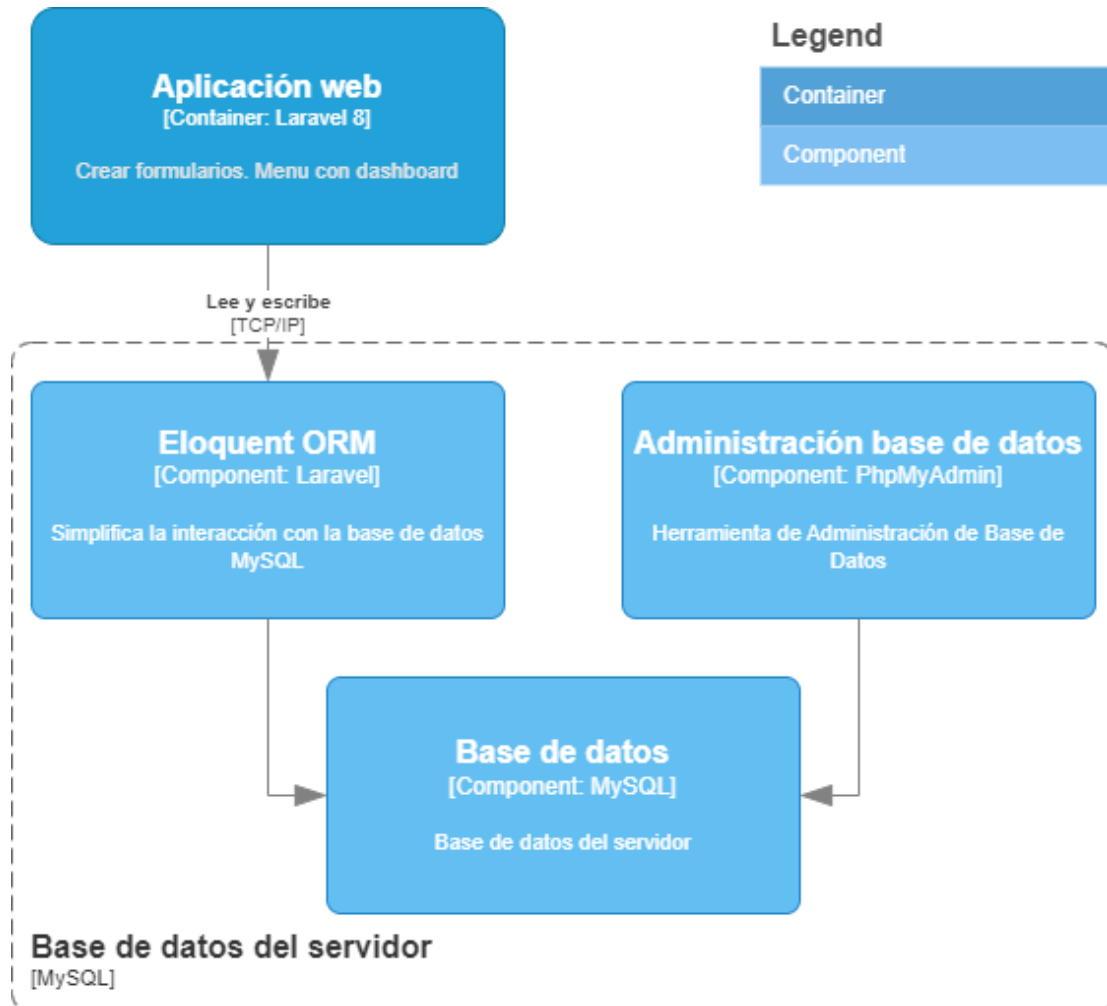


Figura 5. Diagrama de componente bases de datos del servidor

- La interacción entre la aplicación web y la base de datos se realiza a través del componente Eloquent ORM. Eloquent ORM establece comunicación con la aplicación utilizando TCP/IP para enviar y recibir datos de la base de datos del servidor MySQL.
- Para administrar la base de datos MySQL, se utiliza *phpMyAdmin*. Esta herramienta proporciona una interfaz para administrar y gestionar la base de datos, facilitando tareas como la creación de tablas, inserción y eliminación de datos, la importación y exportación de datos, etc.

## Componentes base de datos aplicación móvil

El diagrama detallado del componente de la base de datos de la aplicación móvil se puede ver en la figura 6:

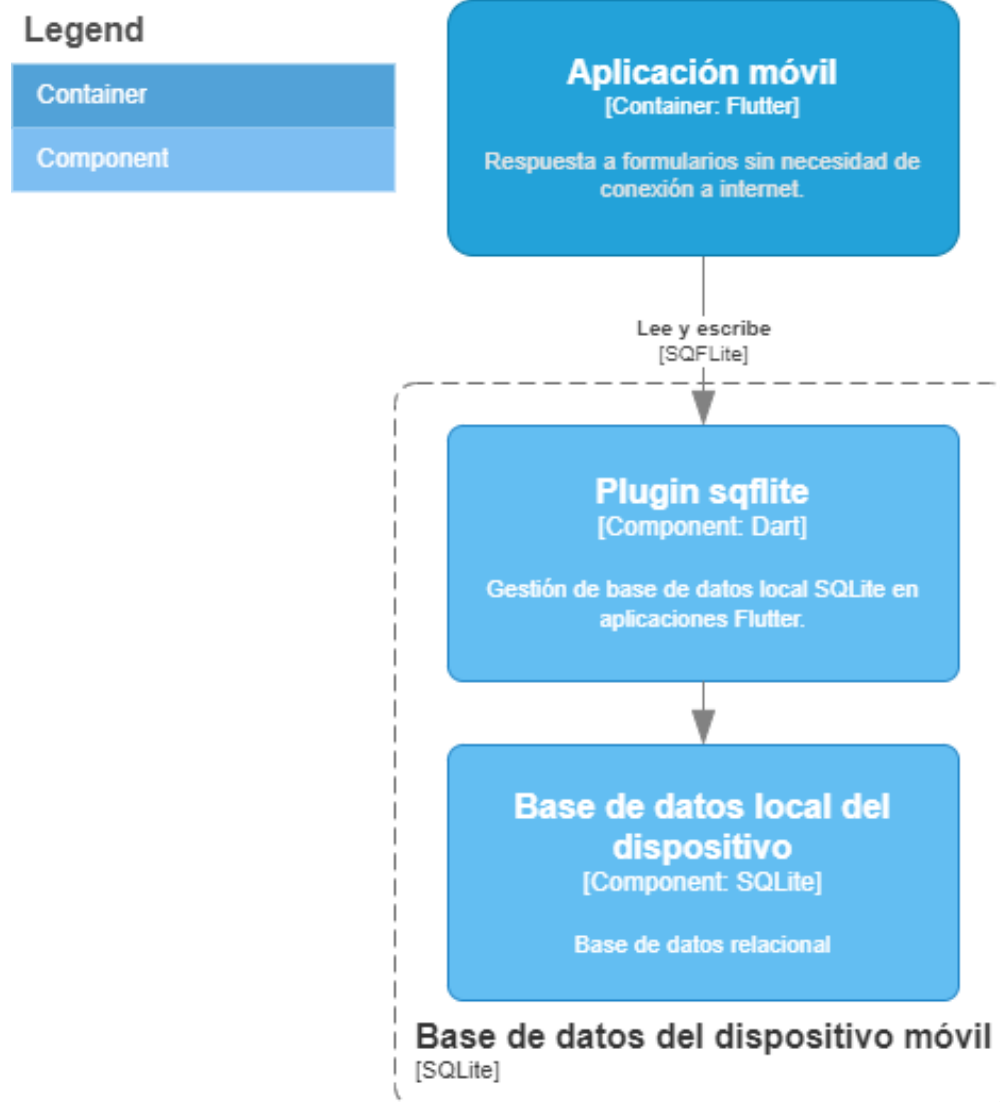


Figura 6. Diagrama de componente base de datos aplicación móvil

- Para la interacción entre la aplicación móvil y la base de datos del dispositivo, se tiene un componente que utiliza sqflite. Sqflite simplifica y facilita esta interacción al proporcionar una interfaz para trabajar con bases de datos SQLite en aplicaciones móviles.

## 6. Stack tecnológico

En esta sección, se detallará el stack tecnológico utilizado y se justificará la elección de cada tecnología, detallando sus ventajas y desventajas. Las alternativas a la tecnología actual se extraen de *Get On Board* [5], una plataforma que ofrece empleos en la industria tecnológica y proporciona estadísticas sobre las tecnologías más ofrecidas y demandadas en el mercado laboral de América Latina. Por lo tanto, las alternativas mencionadas son algunas de las más populares y necesitadas en el mercado, lo que garantiza la disponibilidad de personal para mantener la aplicación.

### Aplicación móvil: Flutter

Flutter se seleccionó como framework para la aplicación móvil debido a su capacidad para crear aplicaciones tanto para Android como para iOS. Además, cuenta con una variedad de bibliotecas y widgets que facilitan el desarrollo de aplicaciones móviles en comparación con otras opciones.

#### Ventajas:

- **Desarrollo multiplataforma:** Flutter permite crear aplicaciones móviles para Android e iOS.
- **Facilidad de uso:** Flutter tiene una gran cantidad de widgets que simplifican la creación de interfaces de usuario (UI) y componentes.
- **Desarrollo más rápido:** Gracias a su variedad de widgets y bibliotecas predefinidas, se acelera el proceso de desarrollo al eliminar la necesidad de implementar ciertas funcionalidades desde cero. Además, cuenta con hot reload que permite una recarga rápida de la aplicación, lo que significa que los cambios realizados se pueden visualizar casi instantáneamente.

#### Desventajas:

- **Personalización de widgets:** Aunque los widgets predefinidos facilitan el desarrollo inicial, la personalización puede resultar mucho más compleja.
- **Peso de la aplicación:** Las aplicaciones Flutter pueden tener un tamaño de archivo más grande en comparación con otras soluciones.
- **Rendimiento en comparación con el desarrollo nativo:** Flutter puede tener un rendimiento ligeramente inferior en comparación con las aplicaciones desarrolladas nativamente para cada plataforma (Android e iOS).

#### Alternativas:

- **React Native:** Framework basado en React muy popular y ampliamente utilizado en el desarrollo móvil. Aunque tiene un rendimiento ligeramente inferior a Flutter, es una muy buena alternativa si se está familiarizado con React y se busca reutilizar código o componentes de la aplicación web.
- **Ionic:** Un framework más antiguo que utiliza tecnologías web estándar como HTML, CSS y JavaScript, así como frameworks de frontend como Angular. Aunque es una opción válida para el desarrollo de aplicaciones móviles, su rendimiento puede ser inferior en

comparación con Flutter y React Native. Además, hay que mencionar que existe una versión de pago con características adicionales y soporte premium.

#### **Otras alternativas no recomendadas:**

- **Kotlin Native, Java, Swift y Objective-C:** No son alternativas viables para el desarrollo multiplataforma, dado que cada una está limitada a un único sistema operativo (solo Android o iOS).

Se recomienda continuar utilizando Flutter, ya que ofrece un rendimiento óptimo, es fácil de aprender para aquellos que no tienen experiencia previa en desarrollo frontend web y cuenta con una amplia variedad de widgets que simplifican y aceleran el proceso de desarrollo. React Native puede ser una buena opción si se utiliza React en la aplicación web, ya que permite aprovechar los conocimientos existentes y la reutilización de código y componentes entre la aplicación web y móvil. Ionic, en comparación, se considera una opción inferior debido a su rendimiento y popularidad decreciente en comparación con Flutter y React Native.

#### **Aplicación web: Laravel y Livewire**

Se eligió Laravel como el framework principal para la aplicación web por varias razones. Cuando se empezó el desarrollo de software en la empresa, se contaba inicialmente con un único desarrollador encargado de todos los proyectos. Para simplificar y agilizar el desarrollo, se optó por utilizar un framework fullstack completo en lugar de dividir el desarrollo entre frameworks de frontend y backend separados.

Se optó por Laravel debido a su facilidad de aprendizaje, su amplio conjunto de características y paquetes, además de su popular utilización en numerosas empresas pequeñas. A esto se suma la gran comunidad de desarrollo que tiene y su clara documentación.

Por otro lado, la decisión de utilizar Livewire como framework frontend se eligió por la misma razón de simplificar y acelerar el desarrollo. Livewire se integra sin dificultad con Laravel y utiliza el mismo lenguaje y sintaxis, lo que elimina la necesidad de aprender otro lenguaje para utilizar un framework de frontend distinto.

#### **Ventajas:**

- **Simplicidad y facilidad de uso:** Laravel y Livewire son simples y fáciles de utilizar, aprender e instalar.
- **Bajo consumo de recursos:** Laravel y Livewire son livianos en términos de recursos.
- **Funcionalidades integradas:** Laravel incluye funciones como autenticación, almacenamiento en caché y enrutamiento, lo que acelera el proceso de desarrollo al proporcionar estas herramientas integradas.
- **Integración fácil de Livewire:** Livewire se integra perfectamente con Laravel, teniendo el mismo lenguaje y sintaxis. Esto elimina la necesidad de aprender otro framework y lenguaje.
- **Eliminación de la necesidad de crear una API:** Livewire elimina la necesidad de comunicación a través de API entre un framework frontend y el backend.

- **Simplificación del desarrollo:** Al combinar frontend y backend, se simplifica y acelera el desarrollo al eliminar la necesidad de manejar ambos separadamente.

#### **Desventajas:**

- **Menor escalabilidad:** La combinación de frontend y backend puede limitar la escalabilidad de la aplicación, especialmente en proyectos grandes y complejos.
- **Menor flexibilidad:** Laravel, a pesar de su simpleza, puede ser considerado poco flexible en comparación con otros frameworks. Además, Livewire puede ofrecer menos flexibilidad en comparación con separar el frontend y el backend.
- **Rendimiento:** El lenguaje PHP, aunque ampliamente utilizado, a menudo es considerado como antiguo y no destacado por su rendimiento. Además, Livewire puede experimentar un rendimiento más lento en comparación con un framework de frontend real.
- **Menos funcionalidades que un framework de frontend independiente:** Livewire puede tener menos características y funcionalidades en comparación con frameworks de frontend independientes, lo que puede limitar en implementar ciertas funciones.
- **Comunidad más pequeña y menor documentación:** La comunidad de Livewire es mucho menor en comparación con otros frameworks y tecnologías, lo que puede dificultar en buscar soluciones o recursos en línea.

#### **Alternativas:**

En cuanto a los frameworks **frontend**, tenemos tres opciones principales:

- **React:** Es el framework frontend más popular y conocido. No es muy difícil de aprender y ofrece un muy buen rendimiento en las aplicaciones.
- **AngularJS:** Considerado el más difícil de los tres para aprender, pero es altamente escalable y mantenible, lo que lo hace popular e ideal para aplicaciones a gran escala.
- **Vue.js:** Destaca por ser el más fácil de aprender entre los tres. Ofrece flexibilidad, aunque cuenta con una comunidad más pequeña y menos bibliotecas en comparación con los otros dos.

Para el desarrollo del **backend**, contamos con las siguientes alternativas. Algunas de estas opciones son también fullstack, por lo que es posible que no sea necesario utilizar un framework de frontend o utilizar un framework más simple y liviano similar a Livewire:

- **Express.js:** Framework de backend bastante rápido, sencillo, flexible y fácil de aprender. Además, es muy ligero, ya que no incluye una gran cantidad de paquetes predefinidos, solo se instalan los que se necesitan. El desarrollo con Express.js suele ser más rápido, gracias a la libertad de JavaScript al escribir código, sin embargo, esta flexibilidad puede resultar en una mayor dificultad para el mantenimiento del código y limitar la escalabilidad, ya que permite varios estilos de escribir código. Express.js utiliza JavaScript o TypeScript, los cuales son lenguajes comunes en el desarrollo frontend, facilitando la integración con estos frameworks.
- **Java Spring Boot:** Diseñado para aplicaciones empresariales y desarrollo a gran escala, Spring Boot es un framework altamente escalable, flexible y seguro. Es adecuado para aplicaciones grandes y complejas, así como para proyectos empresariales que requieren escalabilidad y seguridad, ya que tiene varias características integradas, como seguridad,

gestión de transacciones y soporte de bases de datos. Sin embargo, tiene una gran curva de aprendizaje, especialmente porque utiliza Java. Además, dado que es un framework más antiguo, puede resultar más difícil de mantener debido a las sucesivas actualizaciones. y todas las características que posee.

- **Ruby on Rails:** Al igual que Laravel, posee varias características y funcionalidades, aunque su curva de aprendizaje es más pronunciada y su flexibilidad más limitada en comparación con Laravel. Sin embargo, es más robusto y puede ofrecer un rendimiento superior. Es importante mencionar que utiliza el lenguaje Ruby, que ha disminuido en popularidad y ya no es tan ampliamente utilizado como antes.
- **Django:** Un framework que posee varias funcionalidades al igual que Laravel, se destaca por su capacidad para desarrollar sitios web a gran escala, enfocados en la seguridad y el rendimiento, especialmente en campos como la ciencia y la tecnología. Sin embargo, es menos sencillo y más difícil de aprender, a pesar de que Python, el lenguaje en el que está basado, es conocido por ser más fácil y popular. En comparación, Laravel suele ser la elección preferida por pequeñas empresas, pymes y para los que son nuevos en el desarrollo web.
- **NestJS:** NestJS es un framework basado en Node.js, similar a Express. En comparación con Express, NestJS es más reciente y está diseñado para aplicaciones más grandes, lo que lo hace más mantenible y escalable. Sin embargo, su aprendizaje puede resultar más difícil, ya que es más estricto y robusto. Además, a diferencia de Express, NestJS es menos flexible y más pesado en recursos.
- **Symfony:** Symfony, la otra alternativa fullstack con PHP, se distingue por su mayor flexibilidad, escalabilidad y robustez en comparación con la simpleza de Laravel. Está diseñado para aplicaciones empresariales complejas que requieren de personalización y escalabilidad. Tiene una estructura modular y opciones de configuración que lo convierten en una buena opción para proyectos con requisitos específicos. Sin embargo, hay que considerar que, Laravel al ser más popular y preferido, es probable que tenga una documentación más extensa y una comunidad de desarrolladores más amplia que Symfony.

#### **Otras alternativas no recomendadas:**

- **.NET:** Similar a Spring Boot en varios aspectos, pero tiene funcionalidades de pago y depende mucho de un ecosistema Windows/Microsoft. A pesar de esto, sigue siendo uno de los frameworks más utilizados por numerosas empresas.
- **Flask:** Flask, siendo un microframework, viene con funcionalidades limitadas en comparación con otros frameworks, lo que podría no ser adecuado para proyectos complejos.

Dado que las aplicaciones que se están desarrollando son para uso interno dentro de la empresa y es muy poco probable que sean utilizadas por millones de personas, además de contar con un equipo de desarrollo pequeño, se requieren frameworks que sean simples, fáciles de aprender y que permitan un desarrollo rápido, en lugar de un framework complicado y potente que soporte una gran cantidad de usuarios con tiempos de respuesta rápidos. Entre todas las alternativas, Express.js es el framework que mejor se ajusta a estos requisitos.

Express.js es un framework de backend ligero, simple y fácil de aprender. Como utiliza el lenguaje JavaScript/TypeScript, que es el mismo utilizado por todos los frameworks de frontend, facilita la mantenibilidad, integración y el aprendizaje.

Entre los frameworks de frontend recomendados para usar con Express.js, React es el más popular y recomendado. Vue.js puede ser una buena opción si se busca simplicidad, pero Express.js con React es una combinación muy popular que cuenta con una amplia documentación. Además, si se opta por React, para el desarrollo de aplicaciones móviles se puede considerar utilizar React Native, el framework móvil más popular.

Sin embargo, Laravel sigue siendo un buen framework que cumple los requisitos del equipo TI de la empresa, y es ampliamente utilizado por varias empresas, por lo que no es necesario migrar a otro framework como Express.js a menos que sea absolutamente necesario. Se deben considerar los costos de migración, el tiempo necesario para llevarla a cabo, los problemas y errores que van a surgir, rehacer todas las aplicaciones en otro lenguaje y la integración con el servidor actual. Laravel actualmente cumple con todas las necesidades de la empresa, y está funcionando correctamente, por lo que no hay necesidad de cambiarlo.

Una posible mejora inmediata, o quizás la primera en considerar, podría ser el framework de frontend utilizado actualmente: Livewire. Es cierto que es simple, utiliza la misma sintaxis que Laravel y se integra rápidamente y sin dificultades, pero puede limitar las capacidades del frontend. Si en el futuro se requiere un frontend dedicado para un mayor control, una mejor mantenibilidad, mayor escalabilidad, reusabilidad, la adición de funcionalidades más complejas, un mejor rendimiento, una mejor organización al separar el frontend y el backend, y una comunidad y documentación mucho más amplia, podría considerarse el uso de Vue.js o React.

Generalmente, Laravel se utiliza con Vue.js, ya que Vue.js es más simple y se integra fácilmente con Laravel. React siempre es una buena opción, pero tiene una curva de aprendizaje mayor. Aunque Angular también es una opción viable, se suele utilizar en aplicaciones más complejas y requiere más tiempo para aprender y desarrollar en comparación con Vue.js o React.

### **Base de datos: MySQL**

MySQL es uno de los sistemas de gestión de bases de datos más populares que utiliza SQL. Se eligió MySQL porque viene incluido con cPanel y XAMPP, lo que evita la necesidad de integrar otro sistema de base de datos en cPanel. Además, es la única opción para trabajar en XAMPP. Es común utilizar MySQL con Laravel, especialmente en entornos LAMP (Linux, Apache, MySQL y PHP).

### **Ventajas:**

- **Organización y claridad:** Al ser SQL, todo está organizado de manera clara, facilitando la comprensión y gestión.
- **Buen rendimiento:** Gracias a su naturaleza SQL, MySQL tiene un buen rendimiento al realizar operaciones de lectura y escritura en conjuntos de datos pequeños y medianos.

- **Seguridad robusta:** Debido a su estructura integrada y sistema de almacenamiento de datos, las bases de datos SQL ofrecen una mayor seguridad y facilitan la implementación de medidas de protección de datos.
- **Facilidad de aprendizaje:** La sintaxis simple y la complejidad moderada hacen que MySQL sea fácil de aprender.
- **Popularidad y amplio uso:** MySQL es uno de los sistemas de gestión de bases de datos más populares y ampliamente utilizados en el mundo, lo que garantiza una amplia comunidad de usuarios y recursos de soporte disponibles.
- **Gratuito:** MySQL está disponible de forma gratuita.
- **Compatibilidad multiplataforma:** MySQL es compatible con varios sistemas operativos, incluyendo Windows, Linux, MacOS, IRIX, así como con plataformas de alojamiento en la nube como Amazon, Microsoft y otros.

#### **Desventajas:**

- **Escalabilidad horizontal:** Debido a su naturaleza SQL, la escalabilidad horizontal entre múltiples servidores puede ser complicada de implementar en MySQL.
- **Desafíos de escalabilidad:** MySQL no fue diseñado con la escalabilidad en mente, lo que puede complicar su implementación en comparación con otras opciones.
- **Soporte parcial de código abierto:** Aunque MySQL cuenta con una parte de código abierto, la mayoría está bajo la licencia de Oracle, y se debe pagar por el soporte.
- **Cumplimiento limitado con los estándares de SQL:** SQL tiene estándares específicos que MySQL no sigue completamente. MySQL no tiene soporte para algunas características estándar de SQL, mientras que tiene extensiones y características distintas que no coinciden con los estándares de SQL. Puede ser un problema cuando se necesite migrar a otras bases de datos.

#### **Alternativas:**

Se recomienda seguir utilizando una base de datos relacional SQL, ya que proporcionan mayor seguridad y migrar a una base de datos no relacional podría resultar costoso y difícil. Generalmente, las bases de datos NoSQL se utilizan más para análisis de datos o soluciones simples, mientras que las empresas suelen utilizar bases de datos relacionales para proyectos más complejos.

- **PostgreSQL:** Es una de las bases de datos más populares, al igual que MySQL, con varias características similares. Sin embargo, PostgreSQL destaca por su gran escalabilidad vertical, soporte para muchos tipos de datos, incluyendo NoSQL, mayor variedad de funciones, soporte para características más avanzadas y es de código abierto. Tiene problemas con la documentación poco consistente y la falta de herramientas de revisión que muestren el estado actual de una base de datos. Es una elección ideal para análisis de datos y almacenamiento de datos, siendo especialmente popular entre instituciones financieras y sistemas de telecomunicaciones.

#### **Otras alternativas no recomendadas:**

- **MongoDB:** Un sistema de gestión de bases de datos gratuito y de código abierto, no relacional, conocido por su simplicidad y velocidad. Ofrece una buena compatibilidad

tanto con SQL como con NoSQL, y es escalable horizontalmente. Sin embargo, tiene un alto consumo de memoria y la ausencia de operaciones de unión (joins), lo que resulta en una mayor utilización y desperdicio de la memoria. Además, dado que está diseñado para operaciones rápidas de datos y es NoSQL, es menos seguro en comparación con los sistemas de gestión de bases de datos relacionales.

- **Oracle:** Un sistema de gestión de bases de datos relacionales destacado entre otras bases de datos SQL. Cuenta con una amplia variedad de funcionalidades útiles e innovadoras, una documentación extensa, un buen servicio al cliente y una gran capacidad. Sin embargo, es costoso, requiere de una muy buena infraestructura y es difícil de aprender, incluso se necesita personal certificado por Oracle para su instalación y mantenimiento.
- **Microsoft SQL Server:** La solución de Microsoft, recomendada para entornos que utilizan el ecosistema de Windows o productos de Microsoft. Aunque es costosa, ofrece varias funcionalidades y se integra muy bien con otros productos de Microsoft.
- **Redis:** Un almacén de estructuras de datos en memoria de código abierto y NoSQL, a veces utilizado como caché. Se utiliza en aplicaciones de IoT o arquitecturas de microservicios.
- **Elasticsearch:** Un sistema de gestión de bases de datos orientado a documentos con un procesamiento rápido de datos. Sin embargo, sólo admite el formato de documento JSON y no tiene herramientas de reportabilidad. Es una buena opción para bases de datos enormes que utilizan motores de búsqueda.
- **Cassandra:** Un sistema de gestión de bases de datos no relacional basado en Java, con una alta seguridad y escritura rápida. Sin embargo, su lectura puede ser lenta y requiere muchos recursos. Se utiliza en aplicaciones que procesan grandes volúmenes de información, como data centers, análisis en tiempo real o data streaming constante, como aplicaciones meteorológicas.

Dado que se continuará desarrollando con Laravel y XAMPP, la mejor y única opción es seguir utilizando MySQL. Esta base de datos es la elección óptima para los desarrollos de la empresa debido a su simplicidad, facilidad de aprendizaje, buen rendimiento y compatibilidad con varias plataformas.

En caso de migrar de Laravel y MySQL en el futuro, se recomienda seguir utilizando un sistema de gestión de bases de datos relacional, ya que es más seguro y la migración es más sencilla en comparación con migrar a un sistema no relacional. Además, los beneficios que se obtienen con una base de datos relacional son mayores para los tipos de desarrollos de la empresa, especialmente para desarrollos internos, en comparación con una base de datos no relacional.

Entre las opciones para sistemas de gestión de bases de datos relacionales se encuentran MySQL, PostgreSQL, Oracle y SQL Server. Oracle y SQL Server son sistemas de pago; Oracle, además, es bastante avanzado, contando con varias funcionalidades, configuraciones y opciones, requiriendo un mayor conocimiento para administrarlo. Por otro lado, SQL Server es una buena opción si se utilizará un ecosistema de Microsoft (Windows, .NET, Azure, etc.). Por lo

tanto, las únicas opciones viables (y gratuitas, compatibles con cPanel) son MySQL o PostgreSQL.

Entre estas dos, MySQL es más simple y cubre todas las necesidades de la empresa. Sin embargo, si se requieren más funcionalidades o características, se podría considerar PostgreSQL, aunque tiene una curva de aprendizaje mayor.





Figura 7.b. Diagrama de la base de datos del sistema, parte usuarios

El diagrama de la base de datos se puede ver en las figuras 7.a y 7.b. A continuación, se presenta una explicación de su funcionamiento:

#### **Formato formulario:**

- La tabla de **formularios** contiene información única sobre cada formulario, como su ID, título, subtítulo, código y la especialidad a la que pertenece.
- Cada formulario tiene varias **secciones**, y a su vez, estas **secciones** contienen **subsecciones**.
- Una **subsección** puede ser de tipo *PREGUNTAS*, *CHECKLIST*, *IMAGEN*, o *HEADER*, y contiene una lista de preguntas.
- Dado que muchos enunciados de las preguntas se repiten, no se almacena directamente la pregunta dentro de la subsección, sino que se crea la tabla **tiene\_pregunta** para asociar la **subsección** con la **pregunta**. La tabla de **preguntas** incluye el enunciado y el tipo de cada pregunta, que indica el tipo de respuesta esperada (como texto, selección única, fecha, número, etc.). La tabla **tiene\_pregunta** asocia la **subsección** con la **pregunta** mediante sus respectivos IDs.
- Si la subsección es tipo *IMAGEN*, puede tener solamente una imagen, si es tipo *CHECKLIST*, tiene una **checklist**, que consiste en una tabla simple con 2 o 3 columnas: la primera columna para las preguntas, la segunda para completar con la respuesta y, opcionalmente, la tercera columna para observaciones. Esto permite representar las preguntas como una tabla o checklist.

#### **Respuesta formularios:**

- Para responder una pregunta, es necesario asociar una respuesta. Dado que las respuestas también pueden repetirse, se crea la tabla **respuesta**, que almacena el enunciado de cada respuesta. A su vez, se crea la tabla **pregunta\_tiene\_respuesta**, que relaciona el ID de la respuesta con el ID de la pregunta.
- Dado que anteriormente se definió que las preguntas se asociarían a través de la tabla **tiene\_pregunta**, entonces se asocia el ID de la **respuesta** con el ID de **tiene\_pregunta**. Además, necesitamos el ID del formulario respondido para saber a qué formulario pertenece esa respuesta.
- Se crea la tabla **formularios\_respondidos**, que contiene el ID del formulario original, junto con otros campos como la fecha de completado, el estado (completado, revisado, etc.) y si tiene observaciones.

#### **Firma:**

- La tabla **firmas** contiene el título o nombre de las firmas requeridas.
- La tabla **formulario\_tiene\_firma** asocia el **formulario** con las **firmas** requeridas.
- La tabla **firmantes** almacena la firma de cada firmante, junto con su foto, ID y fecha de la firma. Además, se asocia con el **formulario respondido** y con la tabla **formulario\_tiene\_firma** para especificar la firma correspondiente en el formulario.

#### **Formularios y usuarios pueden pertenecer a varias especialidades y proyectos:**

- Dado que un formulario y un usuario pueden pertenecer a varias especialidades y proyectos, se crean cuatro tablas adicionales. Estas tablas almacenan el ID del usuario o formulario y el ID de la especialidad o proyecto, estableciendo la relación entre ambos.

### **Usuario supervisa usuario:**

- La tabla `usuario_supervisa_usuario` establece la relación entre el supervisor y el trabajador supervisado. La relación se realiza mediante el ID del supervisor y el ID de la persona supervisada.

### **Versiones:**

- La tabla `versiones` almacena la fecha de la última actualización de una tabla. Cuando la aplicación realiza la sincronización para obtener datos, verifica si la fecha que tiene es anterior a la fecha de esta tabla. Si es así, actualiza los datos; de lo contrario, no realiza ninguna acción para no cargar datos innecesariamente.

Durante la práctica, en conjunto con Pablo Zapata Schifferli, alumno de la carrera, y Pablo Peña Medina, ingeniero líder de los desarrollos de la empresa, se diseñó la estructura base de la base de datos del sistema, cubriendo aproximadamente el 70% de su totalidad.

Este diseño incluyó elementos esenciales como la tabla de **formularios**, que se relaciona con la tabla de **secciones**, **subsecciones** y las tablas de **checklist**, **imagen** y **tiene\_pregunta**. Se realizó una optimización adicional creando la tabla de **preguntas**, encargada de almacenar el enunciado de las mismas, así como las tablas **pregunta\_tiene\_respuesta** y **respuesta**. Para almacenar las firmas de forma flexible, se crearon las tablas **firmas**, que almacena el título o nombre de la firma, **formulario\_tiene\_firma**, que relaciona el formulario con las firmas necesarias, y finalmente la tabla **firmantes**, vinculada con `formulario_respondido`.

El diseño inicial solo incluía algunos campos esenciales como los identificadores (IDs), nombres o títulos de los formularios y secciones, orden y fechas. A medida que se fueron creando los formularios digitales, se añadieron más campos para permitir características específicas. Por ejemplo, se creó el campo "borde" para indicar si una sección tiene borde, lo mismo para notas, observaciones y otros elementos similares. Si un formulario requería una característica opcional, se agregaba un campo para almacenarla, para que la aplicación pudiera renderizar esa característica.

Durante las reuniones de demostración con los clientes, se solicitaron diversas funcionalidades adicionales para el sistema, lo que requirió la creación de nuevos campos y tablas en la base de datos. Entre las solicitudes, se incluyó la capacidad de tomar y almacenar fotos como evidencia para respuestas a preguntas, lo que llevó a la creación de un campo en `pregunta_tiene_respuesta` para almacenar el enlace a la imagen. También se pidió la posibilidad de adjuntar evidencia correctiva a las respuestas, lo que requirió otro campo adicional. Además, se implementó la opción de marcar un formulario como crítico si contenía respuestas incorrectas, junto con otras funcionalidades solicitadas.

Por último, se crearon tablas adicionales para gestionar la clasificación de formularios y usuarios. Aunque usuarios y formularios suelen estar asociados a un proyecto y una especialidad específicos, en ocasiones pueden pertenecer a varios proyectos o incluso a todos. También es posible que haya cambios de proyecto, lo que hace necesario crear estas tablas para almacenar la información tanto del proyecto actual como de proyectos anteriores. Se está considerando la

posibilidad de almacenar fechas de cambio de proyecto y otros datos adicionales en el futuro, lo cual será parte del trabajo pendiente.

## 8. Diagrama UML de casos de uso

El diagrama UML de casos de uso del sistema se puede ver en la siguiente imagen (figura 8):

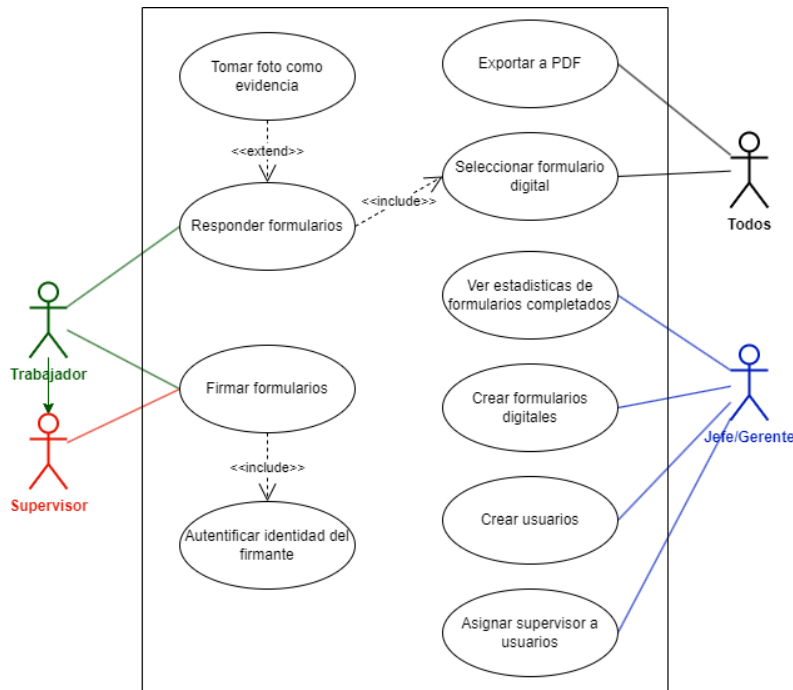


Figura 8. Diagrama UML de casos de uso del sistema

### 8.1. Documentación de casos de uso

Caso de Uso: **Crear formularios digitales.**

Historias de usuario asociadas: U.S.1.1 hasta U.S.1.10

Actor Principal: Jefe/Gerente.

Personal involucrado:

- Jefe/Gerente: Quiere crear formularios digitales utilizando la aplicación web, ya sea a partir de formularios físicos o las plantillas en Excel.

Precondiciones:

- El jefe/gerente ya ha ingresado a la aplicación web y se ha autenticado previamente con sus credenciales.
- El jefe/gerente está en el apartado de formularios y presionó el botón de crear formulario nuevo.

Post Condiciones:

- Se crea un formulario digital que está disponible para ser respondido por los usuarios correspondientes.

Escenario principal de éxito:

1. El sistema solicita los datos iniciales del formulario, como especialidad, título, subtítulo y código.
2. Con los datos ingresados, el sistema crea el formulario con el encabezado, incluyendo preguntas iniciales predefinidas (editables) y una sección para las firmas.

3. El usuario tiene la posibilidad de editar el formulario, agregando, modificando o eliminando preguntas.
4. Para ello, el usuario puede crear secciones que contengan una o varias subsecciones, cada una con un tipo específico (lista de preguntas, tabla simple o imagen).
  - a. La lista de preguntas incluye preguntas de igual o distintos tipos, como texto, número, fecha o selección única.
  - b. Las tablas simples son de 2 o 3 columnas, permitiendo al usuario ingresar respuestas y observaciones.
  - c. Las imágenes pueden ser subidas para complementar las instrucciones del formulario.
5. Además, se puede agregar notas a las secciones y subsecciones, así como habilitar un campo para escribir observaciones por parte de los usuarios.
6. Todos estos elementos, secciones, subsecciones y preguntas son editables y pueden ser modificados en cualquier momento.
7. En cualquier momento se puede exportar el formulario a PDF, para ver cómo va quedando el formulario en formato físico.
8. Cualquier cambio se almacena automáticamente en la base de datos, permitiendo al usuario retomar el proceso en cualquier momento.
9. El jefe/gerente al terminar de crear y editar el formulario, puede publicar el formulario, lo que impide su edición a menos que sea despublicado.
10. Una vez publicado, el formulario está disponible para que otros usuarios lo visualicen y completen. No se puede despublicar si hay formularios respondidos de ese formato.

Escenarios alternativos:

1' Error al ingresar datos: Si se detecta un dato incompleto, el sistema muestra un mensaje de error junto al campo correspondiente.

8' Error sin conexión a internet: Si no tiene conexión a internet, no se guardarán los cambios hechos y se perderán.

Caso de Uso: **Ver estadísticas de formularios completados.**

Historias de usuario asociadas: U.S.4.1 hasta U.S.4.3

Actor Principal: Jefe/Gerente.

Personal involucrado:

- Jefe/Gerente: Quiere ver estadísticas, gráficos e informes de los formularios respondidos utilizando la aplicación web.

Precondiciones:

- El jefe/gerente ya ha ingresado a la aplicación web y se ha autenticado previamente con sus credenciales.

Post Condiciones:

- El jefe/gerente ve un dashboard con estadísticas y gráficos de formularios respondidos, además tiene la opción de exportar la información en formato PDF o Excel.

Escenario principal de éxito:

1. El jefe o gerente está en un menú o página principal y selecciona el apartado de dashboard.
2. Le aparecerán estadísticas y gráficos, todo filtrable, dependiendo de lo que necesita.

3. Puede seleccionar y exportar la información que necesita.

Escenarios alternativos:

2' Error sin conexión a internet: Si no tiene conexión a internet, no le aparecerá la información.

**Caso de Uso: Crear usuarios.**

Historias de usuario asociadas: U.S.6.5 hasta U.S.6.7

Actor Principal: Jefe, gerente o supervisor.

Personal involucrado:

- Jefe, gerente o supervisor: Quiere crear un usuario que no encontró al buscar en el listado de trabajadores.

Precondiciones:

- El jefe, gerente o supervisor ya ha ingresado a la aplicación web y se ha autenticado previamente con sus credenciales.
- El jefe, gerente o supervisor está en el menú con el listado de usuarios y ya ha realizado la búsqueda del trabajador por su RUT o nombre sin encontrarlo.

Post Condiciones:

- Se crea un usuario con la información ingresada exitosamente.

Escenario principal de éxito:

1. Se presiona el botón "Crear usuario".
2. Aparecerá un modal en el que se debe completar la información del trabajador seleccionando su especialidad, proyecto, tipo de usuario (supervisor o trabajador), fecha de nacimiento y seis campos que debe completar: nombre, apellido, cargo, email, RUT y teléfono.
3. Si toda la información se ingresa correctamente y el usuario no ha sido creado anteriormente, se crea el usuario.

Escenarios alternativos:

2' Error al ingresar datos, datos incompletos: Especialidad, proyecto, nombre, apellido, RUT, fecha de nacimiento y tipo de usuario son obligatorios.

2'' Error al ingresar datos, errores de formato: El RUT debe ser un RUT chileno válido, el teléfono debe contener solo números y tener 9 dígitos, la fecha de nacimiento debe indicar que el usuario es mayor de 18 años.

2''' Error al ingresar datos, ya está en la base de datos: Email y RUT deben ser únicos y no deben existir en la base de datos previamente.

**Caso de Uso: Asignar supervisor a usuarios.**

Historias de usuario asociadas: U.S.6.8 hasta U.S.6.11

Actor Principal: Jefe/Gerente.

Personal involucrado:

- Jefe/Gerente: Quiere asignar supervisores al usuario seleccionado o asignar trabajadores a un supervisor.

Precondiciones:

- El jefe/gerente ha ingresado a la aplicación web y se ha autenticado previamente con sus credenciales.

- El jefe/gerente ha seleccionado previamente al trabajador y está viendo su perfil con toda su información.

Post Condiciones:

- El jefe/gerente asigna exitosamente los supervisores o trabajadores al usuario.

Escenario principal de éxito:

1. El jefe o gerente está en el perfil del usuario y hace clic en "Agregar supervisor" o "Agregar trabajadores supervisados".
2. Aparece un listado con todos los supervisores o trabajadores disponibles, mostrando su nombre, RUT y un botón al lado que dice "Agregar supervisor/trabajador".
3. Al presionar el botón, se agrega el supervisor o trabajador al usuario y se quita de esta lista.

Escenarios alternativos:

1' Error sin conexión a internet: Si no hay conexión a internet, nada funcionará, los datos no se cargarán y los cambios no se guardarán.

Caso de Uso: **Seleccionar formulario digital.**

Historias de usuario asociadas: U.S.2.9, U.S.2.10, U.S.2.13, U.S.5.5

Actor Principal: Usuario que tiene acceso y desea acceder a un formulario, ya sea sin responder o respondido, para responderlo o visualizarlo. Puede ser un trabajador, supervisor, jefe, gerente, etc.

Personal involucrado:

- Usuario que tiene acceso y desea acceder a un formulario: Quiere seleccionar un formulario para acceder a él y realizar alguna acción.

Precondiciones:

- El usuario ya ha ingresado al sistema (aplicación web o móvil) y se ha autenticado previamente con sus credenciales.

Post Condiciones:

- El usuario ha accedido al formulario y ahora está visualizando el formulario.

Escenario principal de éxito:

1. El usuario está en un menú o página principal con un listado de formularios filtrables.
2. El usuario selecciona el formulario que desea.
3. El sistema redirecciona a la página con la vista del formulario.

Escenarios alternativos:

3' Error sin conexión a internet: Si no hay conexión a internet, no se redirigirá a la página del formulario en caso de estar utilizando la aplicación web.

Caso de Uso: **Exportar a PDF.**

Historias de usuario asociadas: U.S.1.10, U.S.2.4, U.S.2.15, U.S.5.4

Actor Principal: Cualquier usuario con acceso al sistema que haya seleccionado un formulario.

Personal involucrado:

- Usuario que ha seleccionado un formulario: Quiere ver el formulario en formato PDF, ya sea para tener una vista previa o descargarlo.

Precondiciones:

- El usuario ya ha seleccionado un formulario anteriormente, puede ser un formato de formulario o un formulario respondido.

Post Condiciones:

- El usuario puede ver el formulario en formato PDF.

Escenario principal de éxito:

1. El usuario está en la vista del formulario.
2. El usuario hace click a un botón que diga PDF.
3. Se mostrará la vista PDF del formulario y un icono para descargar el formulario opcionalmente.

Escenarios alternativos:

3' Error sin conexión a internet: Si no hay conexión a internet, no se mostrará la versión PDF del formulario. En caso de la aplicación móvil, si se ha ingresado a ver la versión PDF del formulario anteriormente, mostrará la última versión accedida anteriormente.

Caso de Uso: **Responder formularios.**

Historias de usuario asociadas: U.S.2.1 hasta U.S.2.9

Actor Principal: Trabajador en terreno.

Personal involucrado:

- Trabajador: Quiere utilizar la aplicación móvil para responder el formulario seleccionado.

Precondiciones:

- El trabajador ya ha seleccionado previamente el formulario que quiere responder.

Post Condiciones:

- El trabajador completa y responde exitosamente el formulario.

Escenario principal de éxito:

1. El trabajador accede al formulario, el cual está dividido en varias páginas, cada una representando una subsección del formulario.
2. Para pasar a la siguiente página, el trabajador debe responder todas las preguntas de la página (subsección) actual, ya que todas las preguntas son obligatorias.
3. En cualquier momento, el trabajador puede volver atrás para revisar alguna pregunta o salir del formulario (en este caso, no se guardará el formulario).
4. Al llegar a la última página y terminar el formulario, aparecerá una confirmación y, al confirmar, el trabajador deberá firmar.
5. Luego de firmar, el formulario se guarda con la firma en el dispositivo móvil localmente. Si hay conexión a internet, se envía al servidor.

Escenarios alternativos:

2' Error de campos vacíos: Si hay un campo incompleto, no podrá pasar a la siguiente página.

Caso de Uso: **Tomar foto como evidencia.**

Historias de usuario asociadas: U.S.2.6, U.S.2.14

Actor Principal: Trabajador en terreno que está respondiendo un formulario con la aplicación móvil y encuentra una falla.

Personal involucrado:

- Trabajador: Quiere tomar una foto de alguna falla o defecto del vehículo o herramienta, para evidenciar la falla.

Precondiciones:

- El trabajador está respondiendo un formulario en la aplicación móvil, específicamente una checklist.
- El trabajador ha marcado una respuesta como incorrecta.

Post Condiciones:

- El trabajador ha tomado una foto exitosamente de la falla o defecto.

Escenario principal de éxito:

1. El trabajador responde una pregunta de manera incorrecta en la checklist y se le presenta un diálogo para completar con una observación y una imagen.
2. Puede elegir por tomar una foto en el momento o seleccionar una desde la galería del dispositivo.
3. Después de capturar o seleccionar la imagen, esta se muestra en el diálogo.
4. Al finalizar de responder el formulario, la imagen asociada a la respuesta se guarda junto con el formulario.

Caso de Uso: **Firmar formularios.**

Historias de usuario asociadas: U.S.3.1 hasta U.S.3.5

Actor Principal: Usuario firmante de la aplicación móvil (trabajador o supervisor).

Personal involucrado:

- Trabajador en terreno: Quiere firmar el formulario que acaba de completar para garantizar que fue él quien lo completó.
- Supervisor: Quiere firmar el formulario completado por el trabajador para validar y aprobar el formulario respondido.

Precondiciones:

- Trabajador: El trabajador ha completado el formulario y ha confirmado que quiere entregarlo.
- Supervisor: El trabajador ha firmado el formulario.

Post Condiciones:

- El usuario firmante logra firmar exitosamente el formulario.

Escenario principal de éxito:

1. Luego de completar el formulario, al trabajador le aparece una ventana para dibujar y firmar.
2. Al firmar, el formulario se envía con la firma al servidor.
3. Si el formulario contiene una pregunta crítica mal respondida (es decir, hay un fallo importante), aparecerá una solicitud para que firme el supervisor. El trabajador deberá pasarle la tablet al supervisor para que firme.
  - a. Al supervisor le aparece una ventana con las preguntas críticas mal respondidas y puede firmar, autenticándose con su RUT, contraseña y el panel de dibujo para firmar.
4. En caso contrario, el supervisor puede firmar de tres formas: accediendo a los formularios en la tablet del trabajador y autenticándose, desde su propio dispositivo móvil, o desde la aplicación web. Estas dos últimas opciones requieren conexión a internet y que el formulario firmado por el trabajador haya sido sincronizado previamente.
5. La firma del supervisor se sincroniza con el servidor si hay conexión a internet.

Escenarios alternativos:

2' Error sin conexión a internet: Si no hay conexión a internet, el formulario no se enviará al servidor.

3.a', 4' Error de credenciales incorrectas: Debe reintentar hasta ingresar las credenciales correctas.

5' Error sin conexión a internet: Si no hay conexión a internet, la firma no se sincronizará con el servidor.

5'' Sincronización total: Si el formulario no ha sido sincronizado previamente cuando firmó el trabajador, ahora se enviará al servidor junto con las dos firmas.

Caso de Uso: **Autenticar la identidad del firmante.**

Historias de usuario asociadas: U.S.3.6 hasta U.S.3.8

Actor Principal: Usuario firmante del formulario respondido, ya sea trabajador o supervisor.

Personal involucrado:

- Usuario que firmó el formulario: Al firmar, se debe validar la identidad del usuario.

Precondiciones:

- El usuario ya firmó (dibujó su firma en caso de la aplicación móvil).

Post Condiciones:

- Se autentica la identidad del usuario y se almacena de alguna forma esa autenticación.

Escenario principal de éxito:

1. El usuario firma y presiona enviar.
2. Se toma una foto de la persona que está firmando con la cámara de forma silenciosa (similar a Buk).
3. La foto se asocia a la firma y queda almacenada en el dispositivo móvil.

Escenarios alternativos:

1' Caso tipo de usuario supervisor: Si el supervisor está firmando, debe autenticarse ingresando su RUT y contraseña antes de enviar.

2' Firma del supervisor desde aplicación web: Si el supervisor firma desde una computadora, no se podrá tomar una foto de él. En este caso, se envía un código de la firma a su correo personal.

3' Firma del supervisor desde aplicación web: Si el supervisor firma desde una computadora, no se podrá tomar una foto de él. Por lo tanto, se crea y asocia un código a la firma.

## 9. Desafíos en el desarrollo del sistema

### Digitalización de formularios

Para la digitalización del proceso de formularios, se decidió realizar un desarrollo interno en la empresa en lugar de adquirir y utilizar un software externo. ¿Por qué no utilizar plataformas como Google Forms, Microsoft Forms u otras soluciones similares?

- **Limitaciones en la creación de preguntas:** Estas plataformas no permiten crear algunos tipos de preguntas que la empresa necesita. Aunque pueden crear preguntas simples con respuestas breves, no son capaces para preguntas más complejas, como, por ejemplo, tablas con más de 20 columnas. Esto hace inviable la creación de formularios específicos para la empresa.
- **Cumplimiento del formato de la empresa:** Los formularios de la empresa tienen un formato estricto y definido que no se puede alterar, además, es exigido por algunos clientes. A menos que se pueda transformar el formulario digital de estas plataformas al formato deseado por la empresa, su uso es inviable.
- **Requerimientos específicos de la empresa:** La empresa requiere características específicas que no están presentes en todas estas plataformas, como preguntas con dos respuestas (respuesta y observación), la opción de firmar, la capacidad de marcar algunas preguntas como críticas con un comportamiento distinto, autocompletar respuestas para preguntas comunes y generales, funcionamiento offline, entre otros.
- **Necesidades futuras:** Algunos softwares pueden cumplir con los requerimientos actuales de la empresa, pero ¿qué pasa si en el futuro se necesitan otras características? Es poco probable que estos softwares implementen rápidamente lo que la empresa requiere. En cambio, con un desarrollo propio, los cambios o nuevas funcionalidades pueden implementarse en pocos días o semanas.

Aunque estas plataformas son útiles y ofrecen diversas herramientas, no son suficientes para las necesidades de la empresa. Además, la empresa tendría que adaptar el proceso al software. Si el software no tiene alguna función o característica necesaria, la empresa tendría que cambiar parte de sus procesos para ajustarse a las limitaciones del software, y en el peor de los casos, eliminar alguna parte.

Otro punto importante es que estos softwares ofrecen una digitalización parcial del proceso, manejando solo la creación y respuesta de formularios. No incluyen otras partes, como la firma, la revisión y corrección del formulario, la generación de gráficos, estadísticas y reportes basados en los formularios respondidos, o la conversión del formulario a físico (PDF), entre otros.

### Creación de formularios

Dado que no es factible utilizar un software externo para digitalizar formularios, se decidió desarrollar un software interno. Inicialmente, se pensó que el departamento de TI podría crear los formularios debido a que se pensaba que eran pocos, simples y similares. Sin embargo, esta idea fue descartada al descubrir que había numerosos formularios con formatos variados y cambiantes. Por lo tanto, se necesitaba una solución para permitir que algunas personas de cada departamento crearan los formularios. En colaboración con Pablo Zapata Schifferli, alumno de la

carrera, se desarrolló una aplicación web para la creación de formularios durante la práctica profesional en febrero-marzo 2024.

¿En qué consiste la aplicación?

En primer lugar, se necesita el formulario. Un formulario tiene varias preguntas, pero nos dimos cuenta de que, si simplemente asociamos las preguntas, el formulario no tendrá el formato adecuado. Así surgió la sección, que sería un rectángulo horizontal completo en el formulario. La sección contendrá un listado de preguntas y se puede elegir si viene con borde o no.

Luego surgió otro problema: ¿qué pasa si no queremos que la sección ocupe todo el espacio horizontal y solo la mitad? ¿O si queremos tener dos secciones dentro de una sección? Entonces, se decidió que la sección contendría subsecciones, y las subsecciones contendrían el listado de preguntas. La sección tendrá un tipo donde se definirá el número y formato de estas subsecciones (después de analizar varios formularios, se decidieron los formatos disponibles).

El formato está listo: el formulario tendrá secciones, estas secciones serán de distintos tipos y tendrán diferentes subsecciones, y las subsecciones contendrán un listado de preguntas. Pero ¿qué pasa si queremos una tabla o una checklist? Se decidieron crear dos nuevos elementos: la tabla y la checklist, que contendrán un listado de preguntas. La tabla y la checklist tendrán atributos como el número de columnas, celdas, título de la columna, tipo de respuesta, etc. Entonces, una subsección puede tener opcionalmente una tabla o checklist y luego la lista de preguntas. Hay un caso especial donde, si se quiere ingresar una imagen como ayuda o información, se deberá elegir en el tipo de subsección.

En el caso de las preguntas del listado de preguntas, estas pueden ser de distintos tipos como respuesta breve, larga, solo número, tipo fecha, selección única, etc. También tienen algunos campos opcionales como unidad de medida, crítico, respuesta favorable, etc.

Para las firmas necesarias, se asocian las firmas con el formulario. Para un formulario respondido, se crea un formulario respondido con un nuevo ID que también tiene referencia al ID del formulario original sin responder para el formato. Las respuestas se asocian con una pregunta y el formulario respondido, y las firmas se asocian al formulario respondido.

Finalmente, se realizó una optimización para evitar la repetición innecesaria de preguntas y respuestas en la base de datos. En lugar de guardar la pregunta directamente cada vez que se repite, se creó una tabla para almacenar todas las características de la pregunta (como el ID de la subsección, si es crítica, si tiene unidad de medida, etc.). La pregunta y su tipo de respuesta se guardan en otra tabla. Entonces, la tabla que contiene las características de la pregunta referencia a la tabla con la pregunta mediante un ID, para evitar la repetición de preguntas. Lo mismo aplica para las respuestas.

### **Funcionamiento offline**

Para resolver el problema de la conectividad, se decidió crear una aplicación móvil que pueda utilizarse sin conexión a internet. Aunque hacer que la aplicación web sea responsive permitiría su uso en dispositivos móviles, no solucionaría el problema de la falta de conexión. Sin embargo, esta solución no elimina completamente la necesidad de internet, ya que se necesitará para cargar y enviar datos al servidor.

Surge un nuevo problema: cuándo realizar la sincronización. La sincronización para obtener datos, como los formatos de los formularios, es fácil. Se puede sincronizar, por ejemplo, por la mañana antes de iniciar las actividades o al finalizar la jornada, ya que no se crearán o actualizarán formularios durante las actividades. Para gestionar esto, se implementó una tabla de versiones con la fecha del último formulario publicado. Si la fecha en el dispositivo es diferente (anterior), se realizará la sincronización; de lo contrario, no se hará nada, para no descargar datos innecesarios.

El problema es sincronizar datos enviados desde la aplicación móvil, como los formularios respondidos y las firmas. La solución fue que, al finalizar la respuesta de un formulario, firmar, o realizar cualquier acción que modifique la base de datos, la aplicación intente sincronizar inmediatamente si hay conexión a internet. Si no hay conexión, se implementó una función que revisa periódicamente (cada hora) si hay internet disponible y, de ser así, sincroniza todos los datos pendientes.

Pero, esto puede presentar un problema: si se está sincronizando algo y se ejecuta la función de sincronización periódica al mismo tiempo, la sincronización se realizaría dos veces. Para solucionar este problema, se implementó una tabla llamada "lock" (candado) que guarda el formulario que se está sincronizando. Primero, se revisa si el formulario está en la tabla. Si está, significa que se está sincronizando actualmente; si no, se agrega el formulario a la tabla, "bloqueándolo" y evitando que otro método sincronice el formulario, ya que ya está en proceso de sincronización. Al finalizar la sincronización, se elimina el formulario de la tabla, terminando así el proceso de sincronización de ese formulario.

### **Respuesta de formulario**

Para renderizar el formulario en la aplicación móvil, la base de datos local del dispositivo utiliza un modelo similar a la del servidor. Durante la sincronización, los datos se descargan del servidor y se almacenan en esta base de datos local. Sin embargo, renderizar el formulario de la misma manera que en la aplicación web no es recomendable debido a los distintos tamaños de los dispositivos. Por lo tanto, la aplicación móvil renderiza el formulario por páginas, cada una correspondiente a una subsección con las preguntas. Cada pregunta debe ser respondida para pasar a la siguiente página (subsección) hasta llegar a la última página.

Un desafío fue qué hacer cuando se completa el formulario y no hay conexión a internet para enviarlo al servidor, o si el supervisor no tiene internet para revisarlo. Para solucionar esto, se implementó un mecanismo similar al método tradicional sin digitalización: el supervisor deberá ir a firmar en el dispositivo móvil del trabajador (o el trabajador llevará su dispositivo móvil al

supervisor), es como llevar a firmar un formulario en papel. La firma se guardará en el dispositivo y se sincronizará cuando haya conexión a internet.

Para validar la identidad de quien está firmando, se toman las siguientes medidas. Como el dispositivo es del trabajador y debe estar autenticado, cuando él firma, se toma una foto silenciosa con la cámara (similar a Buk). En el caso del supervisor, deberá ingresar sus credenciales antes de poder firmar, y también se tomará una foto de él.

Otro desafío importante es cómo manejar preguntas críticas (urgentes) mal respondidas. Por ejemplo, ¿qué pasa si se reporta un neumático fallando? La idea es notificar al supervisor, pero no siempre hay conexión internet. La solución implementada es que, si el trabajador responde incorrectamente a una pregunta crítica, su dispositivo móvil se bloqueará. Esto significa que no podrá seguir utilizando la aplicación para responder otros formularios hasta que lleve el dispositivo al supervisor para que lo desbloquee o si el supervisor tiene internet, firma desde otro lugar y lo desbloquea. De este modo, se asegura que el trabajador notificará al supervisor y el supervisor será informado de cualquier problema urgente.

### **Sincronización de móvil a web**

La sincronización de datos desde la aplicación móvil a web no fue muy difícil. Simplemente se implementó una API que recibe los datos de la aplicación móvil, los cuales son procesados y almacenados en la base de datos del servidor por la aplicación web. Hay que señalar que las imágenes no se guardan directamente en la base de datos, sino que se almacenan en bunny.net con un nombre específico, y ese nombre se registra en la base de datos. Renderizar un formulario respondido en la aplicación web tampoco fue complicado, solo se tuvo que agregar las respuestas a los campos que tenían preguntas.

Para notificar al supervisor sobre preguntas mal respondidas y algo en mal estado, se implementó una función que envía correos electrónicos a todos los supervisores del trabajador, detallando los problemas y los elementos en mal estado.

### **Asignación de usuarios**

Un aspecto que no se planificó ni se tuvo muy en cuenta fue la asignación de supervisores a los trabajadores. Este proceso debe ser gestionado por el departamento que utilizará la aplicación y no por el departamento de TI. Por lo tanto, se creó una interfaz que permite a los cargos más altos, como jefes o gerentes, visualizar todos los usuarios y asignarles supervisores y trabajadores.

Esta interfaz incluye un listado completo de usuarios y permite crear nuevos usuarios si no están en el sistema. Al seleccionar un usuario, se muestra toda su información detallada, tanto personal como laboral. Además, se presenta un listado de sus supervisores, los trabajadores que supervisa, los formularios que ha respondido y los formularios respondidos por sus trabajadores supervisados.

Los usuarios de mayor rango tienen la capacidad de modificar la información de los supervisores y trabajadores asignados al usuario que están visualizando, pudiendo agregar o quitar supervisores y trabajadores según sea necesario.

### **Diseño responsive de la aplicación móvil**

Para hacer la aplicación responsive, se siguió un proceso simple: se obtuvo el ancho del dispositivo móvil actual y se dividió por el ancho de la tablet con la que se diseñó la aplicación (600 píxeles). ¿Por qué solo el ancho y no el alto también? Porque el ancho es lo importante; el alto no es relevante ya que los usuarios pueden desplazarse hacia abajo. La división proporciona un factor que se utiliza para multiplicar los tamaños de letra, los tamaños de los contenedores y los tamaños de los widgets, reduciendo el tamaño de todos estos elementos y haciendo la aplicación responsive.

En algunos casos, hay filas con dos o más elementos que, aunque se reduzcan de tamaño, no caben en una sola fila. En estos casos, se dividen los elementos en dos filas distintas. El diseño responsive se probó emulando distintos dispositivos de diferentes tamaños.

### **Seguridad**

Es importante mencionar a la empresa que el sistema debe ser seguro y que la información ingresada en el servidor no puede ser alterada. Los formularios pueden servir como respaldo legal, por lo que se deben implementar medidas de seguridad para garantizar la inalterabilidad de los datos ingresados.

### **Validación de firma**

Finalmente, es importante investigar la validez legal de la firma digital. Aunque se ha implementado la captura de la foto del firmante, esto puede no ser suficiente para validar la firma legalmente, lo que podría afectar gran parte del proceso de gestión de formularios digitalizados.

# 10. Diseño de la aplicación móvil

Se diseña la aplicación móvil con las siguientes pantallas/vistas principales:

## Pantalla de inicio de sesión

La pantalla inicial al abrir la aplicación se puede observar en la figura 9 y debe contar con las siguientes funcionalidades:

- En el primer inicio de la aplicación, el usuario debe seleccionar el proyecto y la especialidad correspondiente.
- Se deben cargar los datos del servidor relacionados con ese proyecto y especialidad para evitar cargar información innecesaria de otros proyectos y especialidades.
- La sincronización debe traer solo los datos nuevos del servidor, evitando cargar información previamente sincronizada y almacenada en la base de datos del dispositivo.
- Después de la sincronización, se mostrará la vista de inicio de sesión, la cual se puede ver en la figura 10 y que incluirá campos para ingresar el RUT del usuario y su contraseña.
- Se debe incluir un botón para cambiar de proyecto o especialidad y volver a sincronizar, cargando la información correspondiente al nuevo proyecto o especialidad seleccionado.
- También se deben agregar botones para restablecer la contraseña y acceder a la información de contacto o soporte.

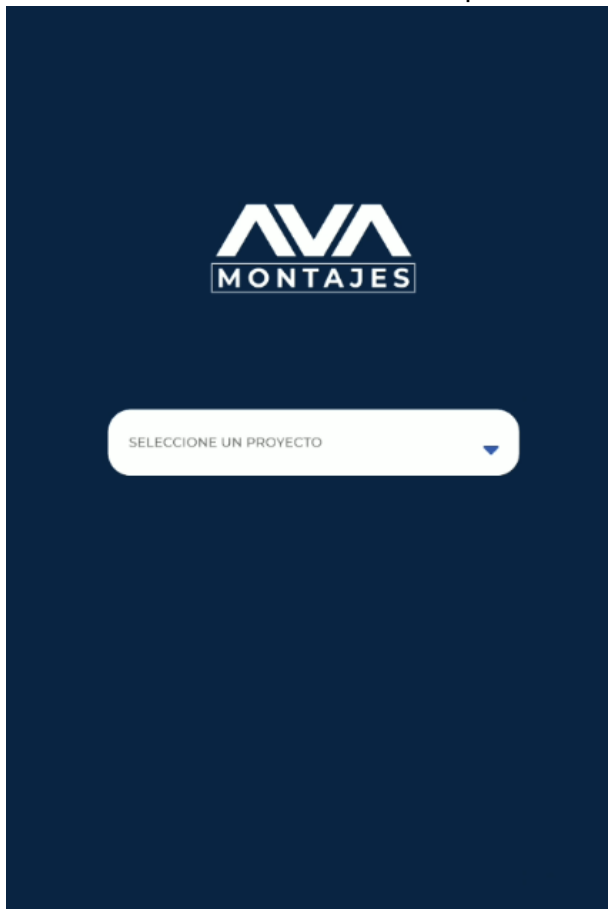


Figura 9. Pantalla de selección del proyecto

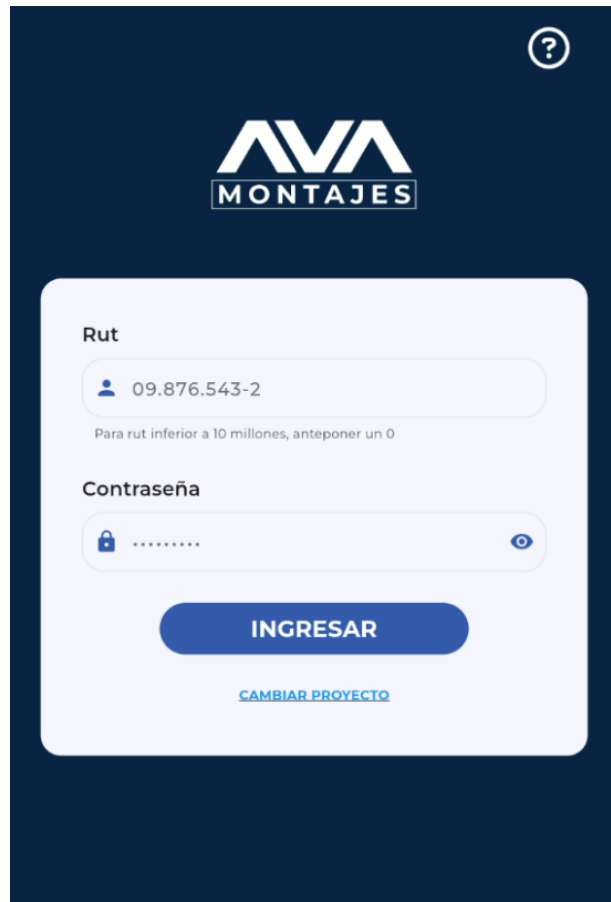


Figura 10. Pantalla de inicio de sesión

## Pantalla inicio

Esta pantalla, mostrada en la figura 11, permite acceder a todas las demás pantallas de la aplicación e incluye:

- Información básica del empleado e informaciones varias.
- Un botón para acceder a la pantalla con el listado de formularios a responder y botones para acceder a distintos formularios respondidos, dependiendo de su estado:
  - Formulario respondido completo, que no requiere ninguna acción adicional.
  - Formulario con falta de evidencia correctiva.
  - Formulario con falta de firma del supervisor o trabajador.
- Una barra lateral con funcionalidades básicas como cambio de contraseña, información de contacto y ayuda, cerrar sesión y sincronización, la cual se puede ver en la figura 12.



Figura 11. Pantalla de inicio

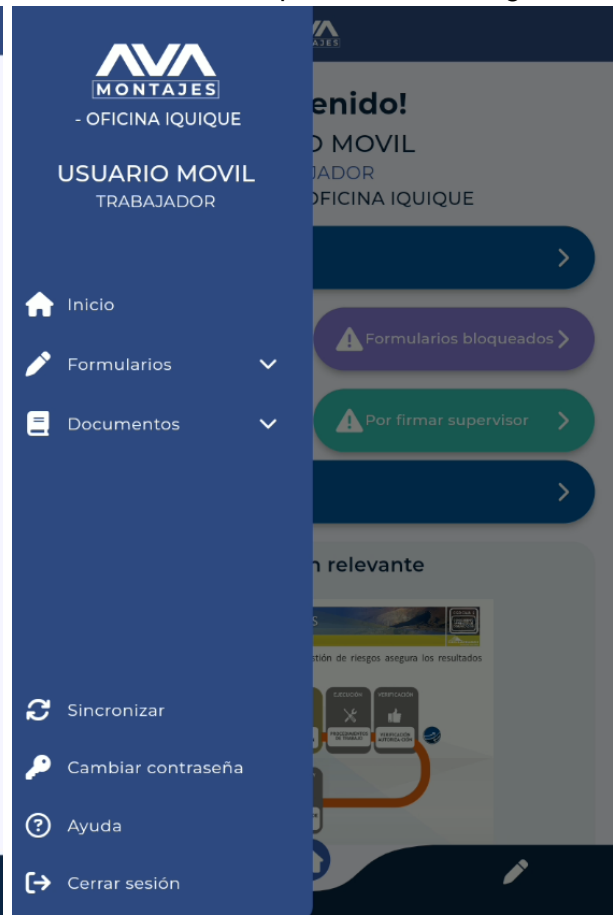


Figura 12. Barra lateral con distintos botones

## Pantalla formularios por responder

Esta pantalla muestra los formularios disponibles que el usuario puede responder, dependiendo del proyecto y especialidad a los que está asignado, así como su cargo en la empresa.

- Aparece un listado de formularios con el código del formulario, el nombre y un botón para la vista PDF del formulario, como se ve en la figura 13.
- Al apretar el botón de PDF, se muestra la vista PDF del formulario, como se ve en la figura 14.
- Al seleccionar un formulario, se redirige a su vista, que contiene todas las preguntas que el usuario debe responder.
  - El formulario se divide en diferentes páginas para cada subsección. En la figura 15 se puede ver, por ejemplo, una subsección (página) que contiene una checklist.
  - El usuario debe poder utilizar la cámara para tomar fotos como evidencia o seleccionar imágenes de la galería al responder preguntas tipo checklist.
- Una vez que el usuario ha completado el formulario, debe firmarlo digitalmente. El formulario se guarda en la base de datos local del dispositivo y, si hay conexión a internet disponible, se envía al servidor.
  - En caso de que alguna pregunta crítica del formulario no haya sido respondida correctamente, por ejemplo, si se identifica alguna parte esencial de un vehículo en malas condiciones, se bloqueará la aplicación móvil (mostrando un modal no cerrable). Esto impedirá al trabajador realizar cualquier acción hasta que notifique al supervisor. El supervisor debe tomar conocimiento de todas las preguntas mal respondidas y firmar el formulario lo más pronto posible, desbloqueando el dispositivo.



Figura 13. Listado de formularios por responder

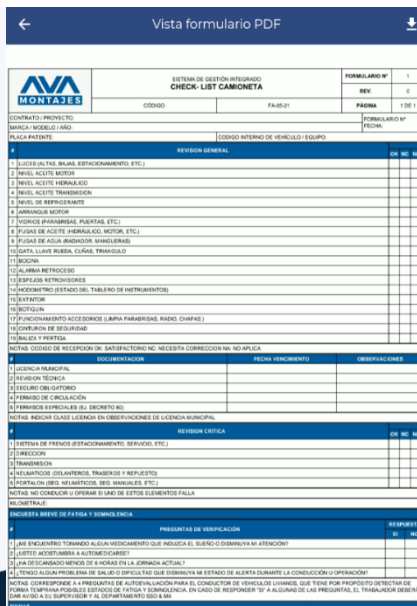


Figura 14. Vista PDF del formulario a responder



Figura 15. Subsección (página) del formulario que contiene una checklist

## Pantalla formularios respondidos

Esta pantalla muestra un listado de los formularios respondidos por el usuario o, si es supervisor, muestra todos sus formularios y los de sus trabajadores.

- Al igual que en el listado de formularios por responder, se muestra el código y el título del formulario, el botón de vista previa en PDF, y ahora también se muestra quién respondió el formulario, la fecha de respuesta y el estado del formulario (completo, con alguna firma faltante o con evidencia correctiva faltante). Las tarjetas del listado vienen en distintos colores dependiendo del estado y están separadas por niveles de fecha de respuesta del formulario (hoy, hace 1 día, hace 2 días, etc.), como se ve en la figura 16.
- Se deben mostrar los dos tipos de formularios respondidos: los que se han enviado al servidor y los que solo están almacenados localmente. Al seleccionar un formulario no sincronizado, si hay conexión a internet, se debe enviar al servidor.
- El formulario renderizado no es tan similar al formato del formulario que se tiene que responder, ya que se renderiza la pregunta y la respuesta en una tarjeta, independientemente del tipo de pregunta (selección única o entrada de texto), como se puede ver en la figura 17.
  - Si al responder el formulario se tomó alguna foto de algo en mal estado, entonces el formulario estará incompleto hasta que se tome una foto del problema corregido como evidencia de la acción correctiva. Después, tanto el trabajador como el supervisor deben firmar nuevamente.
  - Si al seleccionar un formulario no ha sido firmado por un supervisor, el supervisor puede firmar el formulario si el usuario actual le entrega el dispositivo.
- En la figura 18 se puede apreciar una sección con preguntas críticas (borde amarillo), el botón para enviar la evidencia correctiva al terminar de subir toda, y las firmas del formulario. Si no hay firmas, no se mostrarán o se mostrará un botón para firmar dependiendo del estado del formulario.

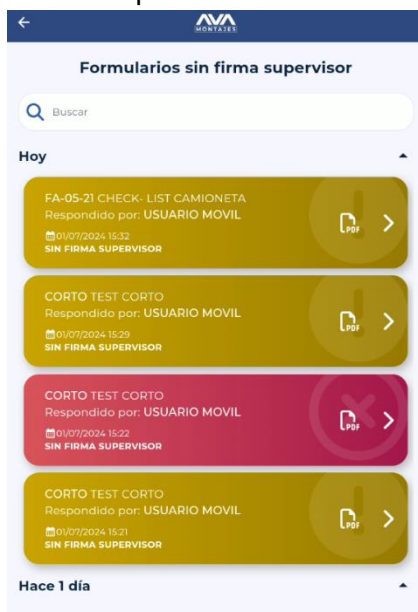


Figura 16. Listado de formularios respondidos y sin firma supervisor

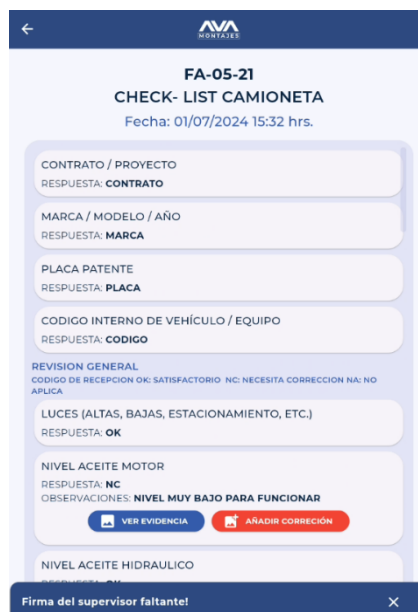


Figura 17. Vista parcial superior de un formulario respondido

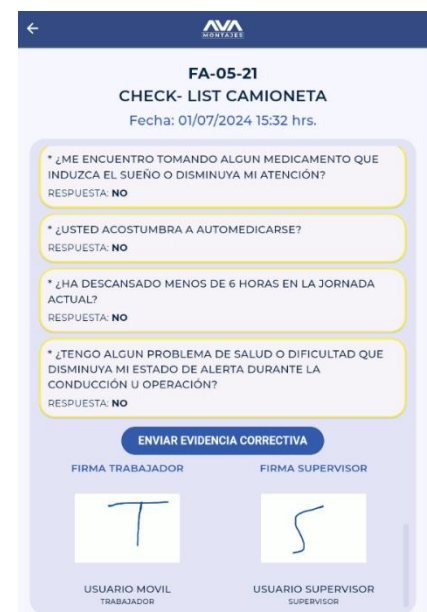


Figura 18. Vista parcial inferior de un formulario respondido

# 11. Diseño de la aplicación web

La aplicación web se diseñó con tres vistas principales:

## Formato/Creación de formularios:

- Esta vista está disponible solo para gerentes y jefes de departamentos. Contiene una tabla con un listado de todos los formatos de formularios de distintas especialidades, mostrando el título, subtítulo, código y estado de los formularios, como se puede ver en la figura 19.
- Hay un botón que, al hacer clic, permite crear un formulario. Al presionarlo, se abre un modal en el que se debe rellenar con la información del proyecto, especialidad, título, subtítulo y código, como se muestra en la figura 20.
- Al seleccionar un formulario de la vista, se redirige a la página que muestra el formulario.
- Esta página contiene un encabezado con el título, subtítulo y código del formulario, así como preguntas predefinidas que están presentes en casi todos los formularios, como se muestra en la figura 21. Todo esto (título, subtítulo, código y preguntas) se puede editar.
- Se incluye una sección que contiene las subsecciones, y estas subsecciones tendrán las preguntas. La figura 22 muestra el modal de creación de sección. Una subsección puede ser de tipo:
  - Tabla simple: Una tabla con 2 o 3 columnas, ideal para checklist.
  - Listado de preguntas: Preguntas de cualquier tipo (selección única, selección múltiple, respuesta breve, respuesta extensa, solo número, fecha, etc.).
  - Imagen: Una imagen para proporcionar ayuda.
- La tabla simple tiene 2 o 3 columnas con los títulos de los encabezados modificables. Generalmente, se utiliza para crear checklist con selección única de opciones definidas, pero también puede ser utilizada como tabla normal donde el usuario responderá con texto. El creador del formulario debe ingresar el enunciado de cada pregunta (la pregunta de cada fila) y seleccionar el tipo de respuesta de la tabla. Si es tipo checklist, además debe completar con distintas propiedades de la pregunta como si es crítico, si la respuesta requiere evidencia y la respuesta correcta de la pregunta. Además, se puede adjuntar una imagen de referencia a la pregunta. La figura 23 muestra dos tablas simples: una tipo checklist con selección única y otra tabla en la que se rellena con texto. La figura 24 muestra el modal de edición de la tabla simple checklist.
- El listado de preguntas permite escribir el enunciado de la pregunta y seleccionar el tipo de respuesta. Todas las preguntas están ordenadas por fecha de creación. También se puede adjuntar una imagen de ayuda a la pregunta y adjuntar imágenes a las opciones de selección única si es de tipo selección única (radio). Es similar a la tabla simple, pero sin el formato de tabla. La figura 25 muestra el modal de edición del listado de preguntas con imágenes adjuntas.
- Una subsección tipo imagen permite subir una imagen.
- Al terminar de editar el formulario, este se puede publicar para permitir respuestas, lo que lo volverá ineditable. También se puede despublicar para poder volver a editar, siempre

y cuando no haya sido respondido por ningún trabajador. Si ya ha sido respondido, no se podrá despublicar.

- También está la opción de ver la versión PDF en todo momento para ver cómo va quedando en formato físico. La figura 26 muestra el formato PDF del formulario solicitado por la empresa.

#	Título	Subtítulo	Código	Estado
1	CHECK- LIST CAMIONETA	SISTEMA DE GESTIÓN INTEGRADO	FA-05-21	Publicado
2	INSPECCIÓN DIARIA MANITOU	SISTEMA DE GESTIÓN INTEGRADO	FA-05-21H	Publicado
3	INSPECCIÓN CORTA	PRUEBA CORTA	FA-CORTO	Publicado
4	CHECKLIST INCOMPLETO	INCOMPLETO	-	No publicado

Figura 19. Tabla por especialidad de formularios

Figura 20. Modal de creación de formulario

Figura 22. Modal creación sección

Figura 21. Ejemplo del encabezado de un formulario

#	DOCUMENTACION	FECHA VENCIMIENTO	OBSERVACIONES
1	LICENCIA MUNICIPAL		
2	REVISION TÉCNICA		
3	SEGURO OBLIGATORIO		
4	PERMISO DE CIRCULACIÓN		
5	PERMISOS ESPECIALES (EJ. DECRETO 80)		

+ AGREGAR PREGUNTA

Notas: INDICAR CLASE LICENCIA EN OBSERVACIONES DE LICENCIA MUNICIPAL

#	REVISION CRITICA	OK	NC	N/A
1	SISTEMA DE FRENOS (ESTACIONAMIENTO, SERVICIO, ETC.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	DIRECCION	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	TRANSMISION	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	NEUMATICOS (DELANTEROS, TRASEROS Y REPUESTO)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	PORTALON (SEG. NEUMÁTICOS, SEG. MANUALES, ETC.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

+ AGREGAR PREGUNTA

Notas: NO CONDUCIR U OPERAR SI UNO DE ESTOS ELEMENTOS FALLA

Figura 23. Ejemplo de dos tablas simples, una con respuesta breve y otra tipo checklist

### EDITAR TABLA SIMPLE

**TITULO DE LA COLUMNA 1:**

**TITULO DE LA COLUMNA 2:**

**SELECCIONE EL TIPO DE RESPUESTA PARA LA COLUMNA 2:**

**OPCIONES DE RESPUESTA**

OK  ✗

NC  ✗

N/A  ✗

**INCLUIR CAMPO OBSERVACIÓN?:**  SI  NO

**ENUNCIADOS COLUMNA "REVISION CRITICA"**

	¿ES CRÍTICA?	¿CON EVIDENCIA?	RESP. CORRECTA
1. SISTEMA DE FRENOS (ESTACIONAMIENTO, SERVICIO, ETC.)	<input checked="" type="radio"/> SI <input type="radio"/> NO	<input type="radio"/> SI <input type="radio"/> NO	OK <input type="text"/>
2. DIRECCION	<input checked="" type="radio"/> SI <input type="radio"/> NO	<input type="radio"/> SI <input type="radio"/> NO	OK <input type="text"/>
3. TRANSMISION	<input checked="" type="radio"/> SI <input type="radio"/> NO	<input type="radio"/> SI <input type="radio"/> NO	OK <input type="text"/>
4. NEUMATICOS (DELANTEROS, TRASEROS Y REPUESTO)	<input checked="" type="radio"/> SI <input type="radio"/> NO	<input type="radio"/> SI <input type="radio"/> NO	OK <input type="text"/>
5. PORTALON (SEG. NEUMÁTICOS, SEG. MANUALES, ETC.)	<input checked="" type="radio"/> SI <input type="radio"/> NO	<input type="radio"/> SI <input type="radio"/> NO	OK <input type="text"/>

+

**Vista previa**

#	REVISION CRITICA	OK	NC	N/A
		<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

TERMINAR EDICIÓN

Figura 24. Modal de edición de tabla simple

**EDITAR PREGUNTA**

Para editar una pregunta, seleccione la pregunta deseada en la columna izquierda. En la columna derecha, encontrará toda la información de la pregunta para su edición, incluyendo el enunciado y el tipo de respuesta. Para guardar los cambios, presionar el botón "Guardar pregunta". También tiene la opción de eliminar la pregunta seleccionada o crear una nueva pregunta.

**1. VEHICULO**

**2. PRUEBA 1**


**3. PRUEBA 2**

NUEVA PREGUNTA +

**ENUNCIADO DE LA PREGUNTA:**



VEHICULO



AGREGAR IMAGEN A LA PREGUNTA QUITAR IMAGEN





**SELECCIONE EL TIPO DE RESPUESTA:**

Selección única

FURGON  

CAMIONETA  

BUS  

ELIMINAR PREGUNTA GUARDAR PREGUNTA

Figura 25. Modal de edición listado de preguntas


		SISTEMA DE GESTIÓN INTEGRADO <b>CHECK- LIST CAMIONETA</b>		FORMULARIO N°	1			
				REV.	0			
		CÓDIGO	FA-05-21	PÁGINA	1 DE 1			
CONTRATO / PROYECTO:				FORMULARIO N°				
MARCA / MODELO / AÑO:				FECHA:				
PLACA PATENTE:		CODIGO INTERNO DE VEHÍCULO / EQUIPO:						
REVISION GENERAL					OK	NC	N/A	
#								
1	LUCES (ALTAS, BAJAS, ESTACIONAMIENTO, ETC.)							
2	NIVEL ACEITE MOTOR							
3	NIVEL ACEITE HIDRAULICO							
4	NIVEL ACEITE TRANSMISION							
5	NIVEL DE REFRIGERANTE							
6	ARRANQUE MOTOR							
7	VIDRIOS (PARABRISAS, PUERTAS, ETC.)							
8	FUGAS DE ACEITE (HIDRÁULICO, MOTOR, ETC.)							
9	FUGAS DE AGUA (RADIADOR, MANGUERAS)							
10	GATA, LLAVE RUEDA, CUÑAS, TRIANGULO							
11	BOCINA							
12	ALARMA RETROCESO							
13	ESPEJOS RETROVISORES							
14	HODOMETRO (ESTADO DEL TABLERO DE INSTRUMENTOS)							
15	EXTINTOR							
16	BOTIQUIN							
17	FUNCIONAMIENTO ACCESORIOS (LIMPIA PARABRISAS, RADIO, CHAPAS.)							
18	CINTURON DE SEGURIDAD							
19	BALIZA Y PERTIGA							
NOTAS: CODIGO DE RECEPCION OK: SATISFACTORIO NC: NECESITA CORRECCION NA: NO APLICA								
#	DOCUMENTACION	FECHA VENCIMIENTO	OBSERVACIONES					
1	LICENCIA MUNICIPAL							
2	REVISION TÉCNICA							
3	SEGURO OBLIGATORIO							
4	PERMISO DE CIRCULACIÓN							
5	PERMISOS ESPECIALES (EJ. DECRETO 80)							
NOTAS: INDICAR CLASE LICENCIA EN OBSERVACIONES DE LICENCIA MUNICIPAL								
REVISION CRITICA					OK	NC	N/A	
#								
1	SISTEMA DE FRENOS (ESTACIONAMIENTO, SERVICIO, ETC.)							
2	DIRECCION							
3	TRANSMISION							
4	NEUMATICOS (DELANTEROS, TRASEROS Y REPUESTO)							
5	PORTALON (SEG. NEUMÁTICOS, SEG. MANUALES, ETC.)							
NOTAS: NO CONDUCIR U OPERAR SI UNO DE ESTOS ELEMENTOS FALLA								
KILOMETRAJE:								

Figura 26. Ejemplo vista PDF de un formulario

## Formularios respondidos

- Esta vista es similar a la de formatos de formularios, pero muestra los formularios que ya han sido respondidos. Dependiendo del cargo, las opciones son:
  - Gerentes y jefes pueden ver todos los formularios respondidos.
  - Supervisores pueden ver los formularios respondidos de sus trabajadores.
  - Trabajadores pueden ver solo sus propios formularios respondidos.
- Al elegir uno, se redirige a la página del formulario, que aparecerá completado con las respuestas, como se ve en la figura 27.
- Además, si el formulario está firmado, se mostrarán las firmas. Si no está firmado y el usuario tiene un alto cargo (supervisor y superior), podrá firmar digitalmente utilizando su contraseña.
- La versión PDF del formulario también incluirá las respuestas y las firmas, como se muestra en la figura 28.

2	DIRECCION	X	
3	TRANSMISION	X	
4	NEUMATICOS (DELANTEROS, TRASEROS Y REPUESTO)	X	
5	PORTALON (SEG. NEUMATICOS, SEG. MANUALES, ETC.)	X	

NOTAS: NO CONDUCIR U OPERAR SI UNO DE ESTOS ELEMENTOS FALLA

KILOMETRAJE: 834

ENCUESTA BREVE DE FATIGA Y SOMNOLENCIA			
#	PREGUNTAS DE VERIFICACION	RESPUESTA	
		SI	NO
1	¿HE ENCUENTRO TOMANDO ALGUN MEDICAMENTO QUE INDECA EL SUEÑO O DIMINUYA MI ATENCION?	X	
2	¿JUSTO ACOSTUMBRADO A AUTOMEDICARME?	X	
3	¿HE DESCANSADO MENOS DE 8 HORAS EN LA JORNADA ACTUAL?	X	
4	¿TENGO ALGUN PROBLEMA DE SALUD O DIFICULTAD QUE DIMINUYA MI ESTADO DE ALERTA DURANTE LA CONDUCCION U OPERACION?	X	

NOTAS: CORRESPONDE A 4 PREGUNTAS DE AUTOEVALUACION PARA EL CONDUCTOR DE VEHICULOS ENVIADOS, QUE TIENE POR PROPOSITO DETECTAR DE FORMA TEMPRANA POSIBLES ESTADOS DE FATIGA Y SOMNOLENCIA. EN CASO DE RESPONDER "SI" A ALGUNAS DE LAS PREGUNTAS, EL TRABAJADOR DEBERA DAR AVISO A SU SUPERVISOR Y AL DEPARTAMENTO SSO & MA

Observaciones generales

TRABAJADOR		SUPERVISOR	
Nombre:	USUARIO MOVIL	Nombre:	USUARIO SUPERVISOR
Fecha firma:	15:32 01/07/2024	Fecha firma:	15:37 01/07/2024

Código de validación de la firma: 102201

Figura 27. Ejemplo formulario respondido y firmado, versión web

1	SALIDA Y PERTIGA	X	
---	------------------	---	--

NOTAS: LOGRO DE RECEPCION OK, SATISFACCION NC, NECESITA CORRECCION NI, NO APLICA

#	DOCUMENTACION	FECHA VENCIMIENTO	OBSERVACIONES
1	LICENCIA MUNICIPAL	01-07-2024	LICENCIA
2	REVISION TECNICA	01-07-2024	REVISION
3	SEGURO OBLIGATORIO	01-07-2024	SEGURO
4	PERMISO DE CIRCULACION	01-07-2024	PERMISO
5	PERMISO ESPECIAL ES (F. DECRETO BS)	01-07-2024	PERMISO ESPECIAL

NOTAS: VERIFICAR CLASE LICENCIA EN OBSERVACIONES DE LICENCIA MUNICIPAL

#	REVISION CRITICA	OK	NC	RIA
1	SISTEMA DE FRENS (ESTACIONAMIENTO, SERVIDO, ETC.)	X		
2	DIRECCION	X		
3	TRANSMISION	X		
4	NEUMATICOS (DELANTEROS, TRASEROS Y REPUESTO)	X		
5	PORTALON (SEG. NEUMATICOS, SEG. MANUALES, ETC.)	X		

NOTAS: NO CONDUCIR U OPERAR SI UNO DE ESTOS ELEMENTOS FALLA

KILOMETRAJE: 834

ENCUESTA BREVE DE FATIGA Y SOMNOLENCIA			
#	PREGUNTAS DE VERIFICACION	RESPUESTA	
		SI	NO
1	¿HE ENCUENTRO TOMANDO ALGUN MEDICAMENTO QUE INDECA EL SUEÑO O DIMINUYA MI ATENCION?	X	
2	¿JUSTO ACOSTUMBRADO A AUTOMEDICARME?	X	
3	¿HE DESCANSADO MENOS DE 8 HORAS EN LA JORNADA ACTUAL?	X	
4	¿TENGO ALGUN PROBLEMA DE SALUD O DIFICULTAD QUE DIMINUYA MI ESTADO DE ALERTA DURANTE LA CONDUCCION U OPERACION?	X	

NOTAS: CORRESPONDE A 4 PREGUNTAS DE AUTOEVALUACION PARA EL CONDUCTOR DE VEHICULOS ENVIADOS, QUE TIENE POR PROPOSITO DETECTAR DE FORMA TEMPRANA POSIBLES ESTADOS DE FATIGA Y SOMNOLENCIA. EN CASO DE RESPONDER "SI" A ALGUNAS DE LAS PREGUNTAS, EL TRABAJADOR DEBERA DAR AVISO A SU SUPERVISOR Y AL DEPARTAMENTO SSO & MA

FIRMAS

TRABAJADOR		SUPERVISOR	
USUARIO MOVIL	T	USUARIO SUPERVISOR	S

Figura 28. Ejemplo formulario respondido y firmado, versión PDF

## Usuarios

- Similar a las vistas anteriores, esta vista muestra una lista de usuarios clasificados por especialidad. Esta vista es accesible solo para altos cargos. Se muestra una tabla con información de los usuarios, incluyendo su RUT, nombre, cargo, rol, proyecto al que pertenecen, número de supervisores y número de trabajadores que supervisan, como se ve en la figura 29.
- Se incluye un buscador/filtro por RUT y nombre para buscar un usuario en específico. Si el usuario no se encuentra, se puede crear uno nuevo haciendo clic en el botón "CREAR USUARIO". Al hacerlo, se abrirá un modal (figura 30) que solicita los siguientes datos: proyecto, especialidad, nombre, apellido, cargo, email, RUT, teléfono, fecha de nacimiento y tipo de usuario.
- Al seleccionar un usuario, se redirige a la página de ese usuario, como se ve en la figura 31, donde se muestra su información personal (nombre, RUT, fecha de nacimiento, etc.), información del cargo (especialidad, proyecto, cargo, email, etc.) y tres tablas:
  - La primera tabla, más pequeña, lista los supervisores actuales y pasados del usuario.

- La segunda tabla muestra los formularios respondidos por el usuario. Si el usuario es supervisor o tiene un cargo superior, también se muestra los formularios respondidos por los usuarios bajo su supervisión.
- La última tabla, visible solo para supervisores o cargos superiores (no trabajadores), muestra una lista de los usuarios bajo su supervisión.
- Hay dos botones: "Agregar supervisores" y "Agregar trabajadores para supervisar". Al hacer clic en cualquiera de ellos, se abre un modal (figura 31) que muestra un listado de usuarios. Esta función está disponible solo para gerentes y jefes.
  - Si se selecciona "Agregar supervisores", aparecerá una lista de supervisores disponibles para asignar al usuario actual.
  - Si se selecciona "Agregar trabajadores para supervisar", aparecerá una lista de trabajadores para seleccionar y asignar bajo la supervisión del usuario actual.

#	Rut	Nombre	Cargo	Rol	Proyecto	Supervisores	Trabajadores
1	12345678-0	USUARIO SUPERVISOR	SUPERVISOR	SUPERVISOR	OFICINA IQUIQUE	0 supervisores	1 supervisando
2	12345678-9	USUARIO MOVIL	TRABAJADOR	TRABAJADOR	OFICINA IQUIQUE	1 supervisores	

Figura 29. Tabla por especialidad de usuarios

### CREACIÓN USUARIO

**PROYECTO** **ESPECIALIDAD**

SELECCIONE UN PROYECTO SELECCIONE UNA ESPECIALIDAD

**NOMBRE** **APELLIDO**

**CARGO** **EMAIL**

nombre@ava.cl

**RUT** **TELEFONO**

12.345.678-9

**FECHA DE NACIMIENTO** **TIPO DE USUARIO EN APP MÓVIL**

MM/DD/YYYY SELECCIONE EL TIPO DEL USUARIO

### ASIGNACIÓN SUPERVISOR A USUARIO

#	RUT	NOMBRE	
1	12345678-9	USUARIO MOVIL	<input type="button" value="ASIGNAR TRABAJADOR"/>
2	51615615-5	SUPERVISOR DEL SUPERVISOR	<input type="button" value="ASIGNAR TRABAJADOR"/>

Figura 32. Modal de asignación de supervisor a usuario

Figura 30. Modal de creación de usuario

INFORMACIÓN PERSONAL

**NOMBRE:** USUARIO SUPERVISOR  
**RUT:** 12345678-0  
**FECHA DE NACIMIENTO:** 31/12/1969  
**TELEFONO:** 123456781

MIS FORMULARIOS RESPONDIDOS

FORMULARIOS RESPONDIDOS POR TRABAJADORES

#	TÍTULO	SUBTÍTULO	CÓDIGO	ESTADO	FECHA DE RESPUESTA
1	INSPECCIÓN CORTA	PRUEBA CORTA	FA-CORTO	COMPLETO	24/06/2024 14:52
2	INSPECCIÓN CORTA	PRUEBA CORTA	FA-CORTO	COMPLETO	24/06/2024 15:30
3	INSPECCIÓN CORTA	PRUEBA CORTA	FA-CORTO	COMPLETO	01/07/2024 15:21
4	INSPECCIÓN CORTA	PRUEBA CORTA	FA-CORTO	RESPONDIDO	01/07/2024 15:22
5	INSPECCIÓN CORTA	PRUEBA CORTA	FA-CORTO	COMPLETO	01/07/2024 15:29
6	CHECK- LIST CAMIONETA	SISTEMA DE GESTIÓN INTEGRADO	FA-05-21	RESPONDIDO	01/07/2024 15:32
7	INSPECCIÓN CORTA	PRUEBA CORTA	FA-CORTO	COMPLETO	09/07/2024 17:03

INFORMACIÓN CARGO

**ESPECIALIDAD:** ESTRUCTURAS  
**PROYECTO:** OFICINA IQUIQUE  
**EMAIL:** DFENG2019@UDEC.CL  
**CARGO:** SUPERVISOR  
**ROL APP MÓVIL:** SUPERVISOR  
**ESTADO:** ACTIVO

SUPERVISORES:

RUT	NOMBRE	SUPERVISIÓN
51615615-5	SUPERVISOR DEL SUPERVISOR	ACTIVA

[AGREGAR SUPERVISOR\(ES\)](#)

TRABAJADORES SUPERVISADOS ACTUALMENTE

TRABAJADORES SUPERVISADOS ANTIGUAMENTE

#	RUT	NOMBRE	CARGO
1	12345678-9	USUARIO MOVIL	TRABAJADOR

[AGREGAR TRABAJADORES PARA SUPERVISAR](#)

Figura 31. Vista de la página de un usuario

## 12. Evaluación del sistema

Para evaluar el sistema, se aplicó el test de usabilidad SUS (System Usability Scale). La escala SUS fue inventada por John Brooke en 1986 como un método rápido y simple para medir la usabilidad de sistemas [20]. En 2013, John Brooke hizo una retrospectiva de su método, describiendo su desarrollo y el impacto que ha tenido en la industria de la usabilidad. Durante esta retrospectiva, se discutió cómo SUS se ha convertido en un estándar para la evaluación de usabilidad, utilizado en miles de evaluaciones en todo el mundo [21].

El SUS se utilizó durante varios años sin una validación formal hasta que se validó en 2013. Esta validación se llevó a cabo a través de numerosos estudios y análisis que confirmaron la fiabilidad y la utilidad del SUS en distintos contextos y para diferentes tipos de sistemas [22].

La escala consta de un cuestionario de 10 preguntas, que deben ser respondidas en una escala del 1 al 5, donde 1 es "totalmente en desacuerdo" y 5 es "totalmente de acuerdo". Este cuestionario incluye preguntas sobre distintos aspectos de la usabilidad, como la facilidad de uso, la complejidad percibida, y la confianza del usuario en el sistema.

Para calcular la puntuación SUS, primero se suman las contribuciones de cada ítem. La contribución de cada ítem varía de 0 a 4. Para los ítems 1, 3, 5, 7 y 9, la contribución es la posición en la escala menos 1. Para los ítems 2, 4, 6, 8 y 10, la contribución es 5 menos la posición en la escala. Se multiplica la suma de las puntuaciones por 2.5 para obtener el valor total de SUS. Las puntuaciones SUS tienen un rango de 0 a 100, mientras más cercano al 100, mejor.

El cuestionario realizado es el siguiente, con preguntas y selección única del 1 al 5.

Tabla 1. Cuestionario SUS para el sistema de formularios digitales

Pregunta	1	2	3	4	5
Creo que me gustaría utilizar la aplicación con frecuencia.					
Encontré la aplicación innecesariamente compleja.					
Pensé que la aplicación era fácil de usar.					
Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación.					
Descubrí que las diversas funciones de la aplicación estaban bien integradas.					
Pensé que había demasiada inconsistencia en la aplicación.					
Me imagino que la mayoría de la gente aprendería a utilizar la aplicación muy rápidamente.					
Encontré la aplicación muy complicada de usar.					
Me sentí muy seguro al utilizar la aplicación.					
Necesitaba aprender muchas cosas antes de poder empezar a utilizar la aplicación.					

## Resultados

Se aplicó el test de usabilidad para la aplicación web, la cual incluye únicamente la creación de formularios digitales. Este test fue respondido por dos personas de alto cargo. En cuanto a la aplicación móvil, que permite responder formularios digitales, firmar, tomar fotos como evidencia, entre otras funciones, el test fue respondido por tres personas.

Tabla 2. Resultados SUS para la creación de formularios en la aplicación web

Pregunta	Nota	SUS
Creo que me gustaría utilizar la aplicación con frecuencia.	4.5	3.5
Encontré la aplicación innecesariamente compleja.	2	3
Pensé que la aplicación era fácil de usar.	4.5	3.5
Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación.	1	4
Descubrí que las diversas funciones de la aplicación estaban bien integradas.	5	4
Pensé que había demasiada inconsistencia en la aplicación.	1	4
Me imagino que la mayoría de la gente aprendería a utilizar la aplicación muy rápidamente.	4.5	3.5
Encontré la aplicación muy complicada de usar.	1	4
Me sentí muy seguro al utilizar la aplicación.	4.5	3.5
Necesitaba aprender muchas cosas antes de poder empezar a utilizar la aplicación.	2.5	2.5

Tabla 3. Resultados SUS para la respuesta de formularios en la aplicación móvil

Pregunta	Nota	SUS
Creo que me gustaría utilizar la aplicación con frecuencia.	4.3	3.3
Encontré la aplicación innecesariamente compleja.	1	4
Pensé que la aplicación era fácil de usar.	4.33	3.33
Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación.	1	4
Descubrí que las diversas funciones de la aplicación estaban bien integradas.	4.34	3.34
Pensé que había demasiada inconsistencia en la aplicación.	1.66	3.34
Me imagino que la mayoría de la gente aprendería a utilizar la aplicación muy rápidamente.	5	4
Encontré la aplicación muy complicada de usar.	1	4
Me sentí muy seguro al utilizar la aplicación.	5	4
Necesitaba aprender muchas cosas antes de poder empezar a utilizar la aplicación.	1.33	3.67

Donde “Nota” indica el promedio de la respuesta de esa pregunta de todas las encuestas respondidas y “SUS” el puntaje SUS que se le da a la respuesta (restar -1 si la pregunta es impar o 5 - nota si la pregunta es par).

Por lo tanto, para la **aplicación web**, el puntaje sería:

$$3.5 + 3 + 3.5 + 4 + 4 + 4 + 3.5 + 4 + 3.5 + 2.5) \times 2.5 = \mathbf{88.75}$$

Y para la **aplicación móvil**:

$$(3.3 + 4 + 3.33 + 4 + 3.34 + 3.34 + 4 + 4 + 4 + 3.67) \times 2.5 = \mathbf{92.45}$$

Como se puede observar en los resultados, la aplicación web obtuvo un puntaje de **88.75**, mientras que la aplicación móvil alcanzó un puntaje de **92.45**. Ambos puntajes, cercanos al 100, indican que las aplicaciones son intuitivas, simples y fáciles de usar, sin causar complicaciones a los usuarios. Sin embargo todavía existe margen para mejorar y hacerlas aún más intuitivas y fáciles de usar.

Se recibieron comentarios sobre la aplicación web, señalando que ciertas funcionalidades en los modales pueden ser confusas y poco claras al principio. También se mencionó que no se podían replicar partes del formulario físico en el formato digital (algunas partes no se podían, otras sí, pero se necesitaba mayor creatividad y experiencia con la aplicación).

En cuanto a la aplicación móvil, se destacó que es intuitiva y fácil de seguir (dado que su principal función es responder formularios, lo que reduce la posibilidad de confundirse). Sin embargo, se señaló la necesidad de mejorar el menú, que puede resultar confuso para los usuarios nuevos. También se sugirió hacer más clara la pantalla que muestra el formulario respondido, para que indique de manera más clara el estado y las acciones necesarias, como firmar, subir evidencia correctiva, o si no es necesario hacer nada.

## 13. Conclusiones

Ingeniería AVA Montajes, a pesar de no ser una empresa de desarrollo o software, puede beneficiarse enormemente de las tecnologías de la información. Se identificó el problema de los formularios físicos en papel y se propuso e implementó una solución de digitalización que resuelve este desafío.

La digitalización de los formularios no solo garantiza su almacenamiento seguro a lo largo del tiempo, sino que también ofrece múltiples ventajas adicionales. Entre estas ventajas se incluyen la facilidad en el completado de los formularios, la mejora en la claridad y legibilidad, la inalterabilidad de los datos y la capacidad para recopilar información de los formularios respondidos. Esta información puede extraerse para obtener estadísticas y realizar análisis. De este modo, no solo se soluciona el problema principal de la pérdida y daño de los formularios, sino que la digitalización añade un valor significativo al proceso.

Además, la solución implementada no se limita a un único departamento de la empresa. Al estar diseñada para digitalizar todos los formularios de la empresa, esta solución permite su expansión y adaptación a otros departamentos con facilidad.

La solución permite diseñar y crear formularios digitales personalizables, desde la configuración de secciones, tamaño y número de preguntas, hasta los tipos de preguntas y respuestas que incluirá. También ofrece la posibilidad de agregar imágenes y modificar los elementos creados, permitiendo replicar la estructura de un formulario físico. Una vez creado el formulario, puede ser publicado para que todos los trabajadores autorizados puedan responderlo. Los trabajadores acceden al formulario a través de la aplicación móvil, donde se renderiza el formulario para seleccionar y responder cada pregunta.

Es importante destacar que la aplicación móvil puede operar sin conexión a internet, aunque se requiere conectividad en algún momento antes o después de la jornada laboral para cargar nuevos formatos de formulario o sincronizar los formularios respondidos con el servidor. Además de estas capacidades básicas, la solución incluye funcionalidades solicitadas por la empresa, como la firma digital de formularios, la exportación de formularios a PDF con el formato corporativo establecido y la opción de tomar fotos como evidencia de problemas detectados.

En la aplicación web, los altos cargos tienen acceso privilegiado para crear formularios, revisar y firmar los formularios respondidos, gestionar usuarios, y próximamente, visualizar un dashboard con estadísticas y gráficos detallados sobre los formularios. Además, tienen acceso a las mismas funcionalidades que los trabajadores tienen para responder formularios en la aplicación móvil.

Uno de los desafíos más importantes surgió al inicio del diseño de la solución: cómo digitalizar un formulario, cómo asociar preguntas manteniendo el formato requerido, y cómo hacerlo lo suficientemente flexible para admitir diferentes formatos. Además, se consideraron aspectos como permitir la respuesta, la firma electrónica y la inclusión de imágenes. Se analizaron

múltiples formularios para desarrollar la solución actual, teniendo en mente cada uno de estos puntos.

Otro desafío importante surgió cuando se solicitó que la solución funcionara sin conexión a internet. Para esto, se desarrolló una aplicación móvil que pudiera ser utilizada en faena sin necesidad de conectividad. El diseño e implementación de esta funcionalidad no fue muy complicado, pero sí resultó tedioso, ya que requería considerar qué ocurriría en ausencia de internet en algunas funcionalidades y cómo manejar estos casos. Se decidió implementar mecanismos de sincronización para el envío y recepción de datos, asegurando que la aplicación funcione tanto en presencia como en ausencia de conexión a internet.

Las aplicaciones desarrolladas están listas y funcionales, con todas las funcionalidades principales necesarias, y pueden adaptarse fácilmente para otros departamentos de la empresa. Sin embargo, aún quedan pendientes algunas funcionalidades no esenciales y optimizaciones por implementar que harán que la aplicación sea aún más completa y eficiente.

Por ejemplo, se podrían implementar mejoras que permitan a los usuarios con los roles adecuados modificar ciertos datos, eliminando la necesidad de que el departamento de TI realice estas modificaciones. Además, se podría mejorar aún más la interfaz para hacerla más intuitiva y accesible. También, se podrían considerar mejoras adicionales menos críticas, como la inclusión del perfil del usuario dentro de la aplicación móvil, incluyendo datos detallados como su especialidad, área de trabajo y otros datos relevantes. Finalmente, se deben añadir funcionalidades solicitadas por la empresa, como la capacidad de subir documentos PDF, para que puedan ser descargados y visualizados en la tablet sin conexión a internet.

Dependiendo de la complejidad de las solicitudes adicionales, se necesitará más tiempo para terminar la aplicación. Sin embargo, si las solicitudes no son muy complicadas o totalmente diferentes de lo ya desarrollado, se estima que se necesitarían entre una y dos semanas para implementar cada nueva característica. Por lo tanto, completar la aplicación móvil con todos los detalles podría tomar alrededor de un mes. Después de finalizarla, será necesario mantener la aplicación, corregir errores y agregar nuevas funcionalidades según surjan las necesidades.

En cuanto a los costos, será necesario contar con un desarrollador con experiencia en Flutter para el desarrollo móvil, y en Laravel para la parte web, además del costo de la licencia de desarrollador de Apple para poder distribuir la aplicación en dispositivos iOS, que es de 100 dólares anuales.

Todo esto es necesario para completar la aplicación de formularios móviles para el departamento de SSOMA. Adaptar la solución para otros departamentos implicará otros costos y tiempos adicionales.

En resumen, aunque AVA Montajes es una empresa de montajes de minería y no de desarrollo o tecnología, puede beneficiarse enormemente de la informática, al igual que otros sectores no tecnológicos como hospitales, pequeños negocios, instituciones educativas, entre otros. La

solución propuesta no solo resuelve el problema básico de los formularios físicos, sino que también ofrece beneficios adicionales, como la reportabilidad, que serían imposibles de lograr con formularios en papel. Esta memoria demuestra que la implementación de soluciones digitales puede mejorar significativamente la eficiencia y seguridad operativa, incluso en industrias tan específicas como la minería.

## 14. Bibliografía

- [1] «AVA Montajes - AVA MONTAJES», Ava Montajes. <https://www.ava.cl/> (accedido 21 de marzo de 2024).
- [2] «Laravel - the PHP framework for web artisans». <https://laravel.com/docs/8.x/> (accedido 3 de marzo 2024).
- [3] «Flutter documentation». <https://docs.flutter.dev/> (accedido 3 de marzo 2024).
- [4] «Laravel - the PHP framework for web artisans». <https://laravel.com/docs/8.x/eloquent> (accedido 13 de abril de 2024).
- [5] «Get on Board Insights - Real-time tech job market trends in Latin America». <https://insights.getonbrd.com/reports/supply-vs-demand> (accedido 23 de abril de 2024).
- [6] R. Wandzel, «Flutter vs React Native vs Ionic [Developer’s perspective]», ULAM LABS, 29 de febrero de 2024. <https://www.ulam.io/blog/flutter-vs-react-native-vs-ionic> (accedido 23 de abril de 2024).
- [7] S. Martin, «React Native vs. Flutter vs. Ionic - Sophia Martin - Medium», Medium, 23 de septiembre de 2020. [En línea]. Disponible en: <https://sophiamartin121.medium.com/react-native-vs-flutter-vs-ionic-46d3350f96ee> (accedido 23 de abril de 2024).
- [8] Nautilus Techlabs, «Ionic vs. React Native vs. Flutter: What’s Best for 2023? | Medium», Medium, 6 de octubre de 2023. [En línea]. Disponible en: <https://medium.com/@naulustechlabs/ionic-vs-react-native-vs-flutter-whats-best-for-2023-a8ae061d5542> (accedido 23 de abril de 2024).
- [9] K. De Silva, «React vs. Angular vs. Vue: A Comparative Analysis - Kalana De Silva - Medium», Medium, 9 de enero de 2024. [En línea]. Disponible en: <https://medium.com/@kalanamalshan98/react-vs-angular-vs-vue-a-comparative-analysis-828159a32c37> (accedido 26 de abril de 2024).
- [10] M. Sevim, «Angular vs React vs Vue: Which is better? - Commencis - Medium», Medium, 27 de enero de 2023. [En línea]. Disponible en: <https://medium.com/commencis/angular-vs-react-vs-vue-which-is-the-better-1b6c88b3825c> (accedido 26 de abril de 2024).
- [11] J. Taylor, «“Laravel vs Java Spring: Which One to Choose for Your Next Web Development Project?”», Medium, 10 de abril de 2023. [En línea]. Disponible en: <https://medium.com/@jamestaylorsoftware/laravel-vs-java-spring-which-one-to-choose-for-your-next-web-development-project-43b60bfacb1f> (accedido 27 de abril de 2024).

- [12] AJawhar, «Spring VS .NET. Which one is better? - AJawhar - Medium», Medium, 10 de enero de 2023. [En línea]. Disponible en: <https://medium.com/@alaajawhar123/spring-vs-net-which-one-is-better-b49d7fd559f7> (accedido 27 de abril de 2024).
- [13] S. Asadi, «Express.js vs Java Spring vs PHP Laravel - Sajad Asadi - Medium», Medium, 18 de enero de 2019. [En línea]. Disponible en: <https://medium.com/@lvlr.xaus/express-js-vs-java-spring-vs-php-laravel-fe74a68828b3> (accedido 27 de abril de 2024).
- [14] Sphinx, «NestJS vs. Express: The Ultimate Comparison of Node Frameworks», Medium, 29 de noviembre de 2023. [En línea]. Disponible en: <https://medium.com/@sphinxshivraj/nestjs-vs-express-the-ultimate-comparison-of-node-frameworks-948b9cc9d3a1> (accedido 28 de abril de 2024).
- [15] D. eXperience Hub, «Laravel vs Django: Which is Better for Large Application Development?», 5 de octubre de 2023. <https://www.linkedin.com/pulse/laravel-vs-django-which-better-large-application-development/> (accedido 30 de abril de 2024).
- [16] @thewilsonwings, «Django vs Laravel which is best for web development», Medium, 15 de diciembre de 2023. [En línea]. Disponible en: <https://medium.com/@info.wilsonwings/django-vs-laravel-which-is-best-for-web-development-56238b959093> (accedido 30 de abril de 2024).
- [17] A. Sheldon, «Symfony vs Laravel: Which PHP Framework to choose? - Geek culture - medium», Medium, 14 de mayo de 2023. [En línea]. Disponible en: <https://medium.com/geekculture/symfony-vs-laravel-which-php-framework-to-choose-8dba240d200c> (accedido 1 de mayo de 2024).
- [18] J. Taylor, «Laravel vs Ruby On Rails: Which Is the Best Framework?», Medium, 13 de abril de 2023. [En línea]. Disponible en: <https://medium.com/@jamestaylorsoftware/laravel-vs-ruby-on-rails-which-is-the-best-framework-d0d9f60b4e42> (accedido 1 de mayo de 2024).
- [19] Editor, «Comparing Database Management Systems: MySQL, PostgreSQL, MSSQL Server, MongoDB, Elasticsearch, and others», AltexSoft, 15 de mayo de 2023. <https://www.altexsoft.com/blog/comparing-database-management-systems-mysql-postgresql-mssql-server-mongodb-elasticsearch-and-others/> (accedido 3 de mayo de 2024).
- [20] Brooke, John. (1995). SUS: A quick and dirty usability scale. Usability Eval. Ind.. 189.
- [21] Brooke, John. (2013). SUS: a retrospective. Journal of Usability Studies. 8. 29-40.

- [22] S. C. Peres, T. Pham, y R. Phillips, «Validation of the System Usability Scale (SUS)», Proceedings Of The Human Factors And Ergonomics Society Annual Meeting/Proceedings Of The Human Factors And Ergonomics Society ... Annual Meeting, vol. 57, n.o 1, pp. 192-196, sep. 2013, doi: 10.1177/1541931213571043.

# Anexos

## Historias de usuario por cada funcionalidad

### 1. Creación formulario digital

#### Usuario: Jefe de departamento

- **U.S.1.1:** Como jefe de departamento, quiero poder crear formularios digitales para que mis empleados puedan completarlos.
- **U.S.1.2:** Como jefe de departamento, quiero primero elegir la especialidad a la que pertenece el formulario para clasificarlo.
- **U.S.1.3:** Como jefe de departamento, quiero asignarle un título, subtítulo y un código al formulario para crear un formulario único e identificable.
- **U.S.1.4:** Como jefe de departamento, quiero tener la capacidad de editar cada sección del formulario digital que estoy creando, incluyendo la adición, edición y eliminación de secciones, preguntas, títulos, notas, tablas, etc., para crear un formulario digital que sea lo más similar al original.
- **U.S.1.5:** Como jefe de departamento, quiero poder agregar algunas propiedades a las preguntas, como por ejemplo si es una pregunta crítica, cuál es su respuesta favorable o si debe ir con evidencia para crear un formulario más completo y personalizado.
- **U.S.1.6:** Como jefe de departamento, quiero poder agregar imágenes de ayuda o referencia a las secciones o preguntas al formulario para que el trabajador tenga una ayuda visual.
- **U.S.1.7:** Como jefe de departamento, quiero poder publicar el formulario creado, para que no se puedan realizar más cambios y permitir que sea accesible para que los trabajadores lo completen.
- **U.S.1.8:** Como jefe de departamento, quiero poder ver todos los formularios digitales creados para identificar si me falta crear alguno o seleccionar uno para visualizarlo.
- **U.S.1.9:** Como jefe de departamento, quiero poder despublicar un formulario para poder editarlo y corregirlo, en caso de haber encontrado un error.
- **U.S.1.10:** Como jefe de departamento, quiero poder ver la versión PDF del formulario digital en todo momento para revisar visualmente cómo está quedando y asegurarme de que se ajuste a mis requisitos.

### 2. Responder formulario digitalmente

#### Usuario: Trabajador

- **U.S.2.1:** Como trabajador en terreno, quiero una aplicación móvil que me permita completar los formularios de manera fácil y rápida, para poder registrar la información necesaria y cumplir con los protocolos de seguridad antes de iniciar una actividad.
- **U.S.2.2:** Como trabajador en terreno, quiero que el diseño de la aplicación móvil sea simple, con elementos visuales claros y de tamaño adecuado, utilizando colores

apropiados, tipografía clara y los elementos bien organizados, para evitar distracciones y confusiones que podrían dificultar la completación de los formularios.

- **U.S.2.3:** Como trabajador en terreno, quiero que la aplicación móvil sea intuitiva, fácil de usar con un flujo claro de lo que se debe hacer, incluyendo instrucciones claras y sencillas, para poder completar los formularios correctamente sin dificultades ni confusiones.
- **U.S.2.4:** Como trabajador en terreno, quiero que el formulario a responder tenga una vista previa en PDF para visualizar globalmente lo que se tendrá que responder y asegurarme de que estaré completando el formulario indicado.
- **U.S.2.5:** Como trabajador, quiero que el campo de respuesta sea de acorde a la pregunta, por ejemplo, si se pregunta por algún valor numérico, en la respuesta solo debo poder seleccionar y escribir números, lo mismo para fechas, solo texto, etc. para facilitar el llenado del formulario.
- **U.S.2.6:** Como trabajador en terreno, quiero poder adjuntar fotos a los formularios para evidenciar cualquier problema o situación relevante.
- **U.S.2.7:** Como trabajador, quiero saber cuáles preguntas son críticas para tomar una decisión más informada.
- **U.S.2.8:** Como trabajador, quiero que se valide la completitud de todos los campos antes de enviar el formulario, para no enviar un formulario con información incompleta.
- **U.S.2.9:** Como trabajador, quiero poder ver el listado de formularios que me corresponde responder para seleccionar alguno y completarlo.
- **U.S.2.10:** Como trabajador, quiero poder ver el listado de formularios he respondido para seleccionar alguno y revisar su información, incluyendo las distintas secciones, subsecciones, preguntas, respuestas, observaciones, imágenes, evidencia, firma, etc.
- **U.S.2.11:** Como trabajador, quiero poder filtrar los listados de formularios, para facilitar la búsqueda de un formulario en específico.
- **U.S.2.12:** Como trabajador, quiero que el listado de formularios completados tenga información del estado, firmas faltantes, si necesita evidencia, fecha de respuesta para tener la información crítica faltante del formulario rápidamente.
- **U.S.2.13:** Como trabajador, al seleccionar un formulario respondido, quiero que se me indique lo faltante o lo que debo hacer, como firmar o ingresar evidencia correctiva, para realizar las correcciones necesarias.
- **U.S.2.14:** Como trabajador o supervisor, quiero poder ingresar evidencia correctiva a un formulario respondido cuando se haya solucionado el problema de alguna pregunta con evidencia ingresada.
- **U.S.2.15:** Como trabajador o supervisor, quiero poder ver y descargar el formulario respondido en formato PDF para imprimirlo y tener una copia.

### **3. Firmar digitalmente**

#### **Usuario: Trabajador**

- **U.S.3.1:** Como trabajador en terreno, quiero poder firmar el formulario completado para que el supervisor pueda revisarlo y aprobarlo.

### **Usuario: Supervisor**

- **U.S.3.2:** Como supervisor en terreno, quiero poder acceder a los formularios digitales completados por los trabajadores desde la aplicación móvil, para revisarlos y poder firmarlos.
- **U.S.3.3:** Como supervisor, quiero que se me notifique cuando se haya completado un formulario con alguna pregunta crítica importante mal respondida para enterarme rápidamente sobre alguna falla o problema.
- **U.S.3.4:** Como supervisor en terreno, quiero poder firmar digitalmente sin internet los formularios completados por los trabajadores desde sus tablets, con el fin de aprobarlos para garantizar el cumplimiento de los protocolos de seguridad antes de iniciar la actividad.
- **U.S.3.5:** Como supervisor, quiero poder firmar digitalmente con internet los formularios completados por los trabajadores desde mi tablet o computador, en caso de no encontrarme cerca para ir a firmar a la tablet del trabajador.
- **U.S.3.6:** Como supervisor, quiero que se solicite mi contraseña al firmar para validar mi identidad y asegurar que otra persona no firme.

### **Usuario: Firmante**

- **U.S.3.7:** Como firmante, quiero que se valide mi identidad al firmar, para garantizar la autenticidad de mi firma y prevenir problemas legales.
- **U.S.3.8:** Como firmante, quiero poder firmar nuevamente el formulario respondido si se ha subido evidencia correctiva, para tomar conocimiento de que se ha solucionado el problema.

## **4. Reportabilidad**

### **Usuario: Jefe/Gerente**

- **U.S.4.1:** Como jefe o gerente, quiero tener reportes en forma de estadísticas, gráficos, u otros formatos, basados en los datos recopilados de los formularios digitales, para realizar análisis y tomar decisiones más informadas.
- **U.S.4.2:** Como jefe o gerente, quiero poder filtrar los datos, para elegir la información que necesito ver.
- **U.S.4.3:** Como jefe o gerente, quiero poder descargar y exportar los reportes, para imprimirlos y tener una copia física.

## **5. Almacenamiento de los formularios**

### **Usuario: Todos**

- **U.S.5.1:** Como empleado, necesito poder encontrar rápidamente un formulario completado cuando lo esté buscando, para poder acceder a la información que contiene.
- **U.S.5.2:** Como empleado, quiero poder filtrar los formularios respondidos por fecha de respuesta, título, código, estado, etc. para poder encontrar rápidamente el formulario necesitado.

- **U.S.5.3:** Como empleado, quiero que el formulario me muestre toda la información, incluyendo las preguntas, respuestas, imágenes, observaciones, firmas, evidencia adjunta, etc.
- **U.S.5.4:** Como empleado, quiero poder exportar el formulario en formato PDF, para poder imprimirlo y tener una copia física.

**Usuario: Todos con algún cargo superior (jefes, gerentes, supervisores, etc.)**

- **U.S.5.5:** Como usuario, quiero acceder a los formularios de los trabajadores bajo mi supervisión para poder seleccionar uno y revisar la información que contiene.

## 6. Creación y asignación de usuarios

**Usuario: Gerente, jefe y supervisor**

- **U.S.6.1:** Como jefe, gerente o supervisor, quiero poder buscar un trabajador por su RUT o nombre para saber si existe y acceder a su perfil.
- **U.S.6.2:** Como jefe, gerente o supervisor, quiero un listado de todos los trabajadores, con información básica como el RUT, nombre, cargo, proyecto, etc., para seleccionar alguno y ver/editar su información.
- **U.S.6.3:** Como jefe, gerente o supervisor, quiero que el listado de todos los trabajadores sea acorde a mi cargo. Por ejemplo, si soy gerente, me aparecerán todos los trabajadores de todos los cargos, pero si soy supervisor, solo me deberían aparecer mis trabajadores. No deberían aparecer gerentes, jefes o cualquier persona con mayor cargo para impedir el acceso no autorizado.
- **U.S.6.4:** Como jefe, gerente o supervisor, quiero poder filtrar la lista de trabajadores por su información, como RUT, nombre, cargo, proyecto, estado, etc., para facilitar la búsqueda de algún trabajador o contar cuántos trabajadores cumplen ciertos filtros, por ejemplo, ver todos los trabajadores sin supervisores.
- **U.S.6.5:** Como jefe, gerente o supervisor, quiero poder crear un usuario si no está en el listado, para agregarlo al sistema y que pueda usar la aplicación móvil.
- **U.S.6.6:** Como jefe, gerente o supervisor, quiero que se me solicite la información del trabajador al crear el formulario, como especialidad, proyecto, nombre, apellido, RUT, email, cargo, teléfono, fecha de nacimiento y tipo de usuario (trabajador/supervisor), para crear el usuario con esa información y asignarle el tipo de usuario que será, otorgándole distintas funcionalidades y permisos a ciertas vistas.
- **U.S.6.7:** Como jefe, gerente o supervisor, quiero que se valide la información que estoy ingresando al crear el usuario. Por ejemplo, la especialidad, el proyecto, el nombre, el apellido, el RUT, la fecha de nacimiento y el tipo de usuario son obligatorios. El RUT debe ser un RUT válido y no repetido, la fecha de nacimiento debe ser mayor a 18 años, el teléfono debe tener solo números y 9 dígitos, etc., para que notifique en caso de estar ingresando mal la información y corregirla.
- **U.S.6.8:** Como jefe, gerente o supervisor, quiero poder seleccionar un trabajador para acceder a la página del trabajador y ver toda su información.

- **U.S.6.9:** Como jefe, gerente o supervisor, quiero poder ver su información personal, como nombre, apellido, RUT, etc., su información del cargo, como especialidad, proyecto, cargo, rol, estado, supervisores y sus formularios respondidos.
- **U.S.6.10:** Como jefe, gerente o supervisor, si el trabajador es supervisor, además, quiero poder ver los formularios respondidos por sus trabajadores y un listado de los trabajadores que está supervisando.

**Usuario: Jefe/Gerente**

- **U.S.6.11:** Como jefe o gerente, quiero tener la capacidad de editar la información de un usuario seleccionado. Esto incluye su información personal, detalles de su cargo, la capacidad de agregar o editar su supervisor, y gestionar los trabajadores supervisados en caso de que el usuario sea un supervisor, para asignar y ajustar la asignación de trabajadores y supervisores, así como corregir cualquier información incorrectamente ingresada.

**7. Requisitos no funcionales**

**Usuario: Todos**

- **U.S.7.1:** Como usuario, quiero tener acceso exclusivo a la información que me corresponde, y que otros no puedan ver mis datos a menos que cuenten con acceso autorizado, para proteger la información y prevenir accesos no autorizados.
- **U.S.7.2:** Como usuario, quiero que la aplicación sea simple, intuitiva, clara y fácil de usar, para realizar mis tareas sin perderme ni confundirme.
- **U.S.7.3:** Como usuario, quiero que la aplicación tenga una respuesta rápida y fluida, con tiempos de carga mínimos, para realizar mis tareas sin perder tiempo, concentración ni paciencia mientras espero.

**Usuario: Usuarios aplicación móvil (trabajador y supervisor)**

- **U.S.7.4:** Como usuario de la aplicación móvil, quiero poder utilizarla (iniciar sesión, visualizar formularios, responder y firmar) incluso cuando no haya acceso a internet, para completar mis funciones asignadas sin depender de la conexión a internet y sin retrasar el proceso esperando que haya conexión.
- **U.S.7.5:** Como usuario de la aplicación móvil, quiero poder utilizar la aplicación tanto en un celular como en una tablet, ya sea en dispositivos Android o iOS, para realizar mis funciones.
- **U.S.7.6:** Como usuario de la aplicación móvil, quiero que la aplicación se sincronice automáticamente cuando haya conexión a internet, para enviar y traer la información más reciente del servidor y mantener la aplicación actualizada.
- **U.S.7.7:** Como usuario de la aplicación móvil, quiero que mi sesión se cierre automáticamente después de unos minutos de inactividad para evitar que otro trabajador acceda a la aplicación con mi cuenta si la dejo abierta.

## Ejemplos de formularios de la empresa


	SISTEMA DE GESTIÓN INTEGRADO		REVISIÓN	7
	CHECK- LIST CAMIONETA		FECHA	18-4-2023
	CÓDIGO	FA-05-21	PÁGINA	1 de 1
CONTRATO / PROYECTO				
MARCA / MODELO / AÑO		FECHA		
PLACA PATENTE		CODIGO INTERNO DE VEHÍCULO / EQUIPO		
<b>CODIGO DE RECEPCION OK: SATISFACTORIO NC: NECESITA CORRECCION NA: NO APLICA</b>				
<b>REVISION GENERAL</b>	<b>OK</b>	<b>NC</b>	<b>NA</b>	
LUCES (Altas, bajas, estacionamiento, etc.)				
NIVEL ACEITE MOTOR				
NIVEL ACEITE HIDRAULICO				
NIVEL ACEITE TRANSMISION				
NIVEL DE REFRIGERANTE				
ARRANQUE MOTOR				
VIDRIOS (Parabrisas, puertas, etc.)				
FUGAS DE ACEITE (hidráulico, motor, etc.)				
FUGAS DE AGUA (Radiador, mangueras)				
GATA, LLAVE RUEDA, CUÑAS, TRIANGULO				
BOCINA				
ALARMA RETROCESO				
ESPEJOS RETROVISORES				
HODOMETRO (Estado del Tablero de Instrumentos)				
EXTINTOR				
BOTIQUIN				
FUNCIONAMIENTO ACCESORIOS (Limpia parabrisas, radio, Chapas.)				
CINTURON DE SEGURIDAD				
BALIZA Y PERTIGA				
<b>DOCUMENTACION</b>	<b>FECHA VENCIMIENTO</b>	<b>OBSEVACIONES</b>		
LICENCIA MUNICIPAL		<b>CLASE LICENCIA</b>		
REVISION TÉCNICA				
SEGURO OBLIGATORIO				
PERMISO DE CIRCULACIÓN				
PERMISOS ESPECIALES (Ej. Decreto 80)				


Figura 32.a. Formulario típico del departamento SSOMA de la empresa. La parte superior del formulario contiene el encabezado, con los datos generales y preguntas comunes a la mayoría de los formularios.

REVISION CRITICA (no conducir u operar si uno de estos elementos falla)		OK	NC	NA
SISTEMA DE FRENOS (estacionamiento, servicio, etc.)				
DIRECCION				
TRANSMISION				
NEUMATICOS (Delanteros, traseros y repuesto)				
PORTALON (Seg. Neumáticos, Seg. Manuales, Etc.)				
KILOMETRAJE:			REVISADO POR SUPERVISOR A CARGO (NOMBRE COMPLETO Y FIRMA)	
OBSERVACIONES				



<b>ENCUESTA BREVE DE FATIGA Y SOMNOLENCIA</b>		
<p>Corresponde a 4 preguntas de autoevaluación para el conductor de vehículos livianos, que tiene por propósito detectar de forma temprana posibles estados de Fatiga y Somnolencia.  <b>En caso de responder "SI" a algunas de las preguntas, el trabajador deberá dar aviso a su supervisor y al departamento SSO &amp; MA</b></p>		
<b>PREGUNTAS DE VERIFICACIÓN</b>		<b>RESPUESTA</b>
		SI NO
¿Me encuentro tomando algún medicamento que induzca el sueño o disminuya mi atención?		
¿Usted acostumbra a automedicarse?		
¿Ha descansado menos de 6 horas en la jornada actual?		
¿Tengo algún problema de salud o dificultad que disminuya mi estado de alerta durante la conducción u Operación?		
<b>DATOS DEL CONDUCTOR Y JEFE EQUIPOS Y MAQUINARIAS</b>		
NOMBRE Y FIRMA DE CONDUCTOR / OPERADOR		
NOMBRE Y FIRMA DE JEFE DE EQUIPOS Y MÁQUINARIAS		FECHA:

Figura 32.b. Formulario típico y básico del departamento SSOMA de la empresa. La parte inferior incluye un espacio destinado para la firma y toma de conocimiento del formulario.

	SISTEMA DE GESTIÓN INTEGRADO		Revisión	4
	<b>RECEPCION EQUIPOS RODANTES</b>		Fecha	19-10-2020
	CÓDIGO	FA-05-03	Página	1 de 1

FECHA		TIPO EQUIPO	
NOMBRE		PLACA PATENTE	
RUT		MARCA Y MODELO	
CARGO		KM/HOR	
OBRA O FAENA		NIVEL DE COMBUSTIBLE	R <input type="checkbox"/> 1/4 <input type="checkbox"/> 1/2 <input type="checkbox"/> 3/4 <input type="checkbox"/>

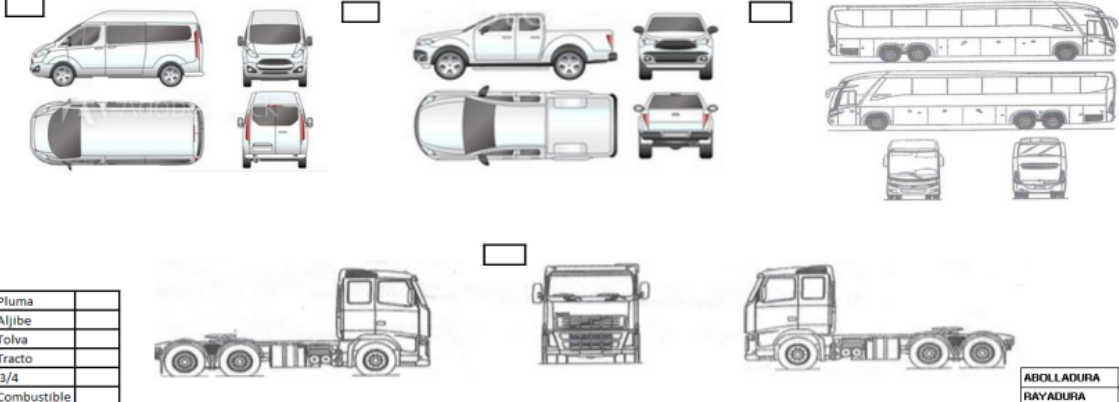
N°	INVENTARIO	ESTADO		
1.	MOTOR	B	M	N/A
1.1	Radio			
1.2	Antena			
1.3	Encendedor			
1.4	Botiquín			
1.5	Extintor			
1.6	Herramientas			
1.7	Kit de emergencia			
1.8	Rueda de repuesto			
1.9	Parabrisas y vidrios en general			
1.10	Estado de patentes			

N°	INVENTARIO	SI	NO	N/A
2	DOCUMENTOS VIGENTES			
2.1	Permiso de Circulación			
2.2	Revisión Técnica y gases			
2.3	Seguro Obligatorio			
3	KIT MINERO			
3.1	Pértiga			
3.2	Baliza			
3.3	Traba tuercas			
3.4	Cuñas			
3.5	Cintas reflejantes			

Pluma	
Aljibe	
Tolva	
Tracto	
3/4	
Combustible	



ABOLLADURA	O
RAYADURA	X

TRABAJO A REALIZAR	
1.-	
2.-	
3.-	
4.-	
5.-	
6.-	
7.-	
8.-	
9.-	
10	

INGRESA A TALLER	RECEPCION TALLER
NOMBRE:	NOMBRE:
CARGO:	CARGO:
FIRMA:	FIRMA:

Figura 33. Formulario con selección de imagen.

	SISTEMA DE GESTIÓN INTEGRADO		REVISIÓN	4
	INSPECCIÓN PLACA COMPACTADORA		FECHA	19-10-2020
	CÓDIGO	FA-05-21Z	PÁGINA	1 de 1
LUGAR DE LA INSPECCIÓN				FECHA
NOMBRE RESPONSABLE DEL ÁREA				
<b>CHEQUEAR EL ESTADO DE OPERATIVIDAD (B=BUENO / M=MALO)</b>				
Nº	ITEM A REVISAR	ESTADO		
1.	<b>EQUIPO Y COMPONENTES</b>	B	M	<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;">CODIGO:</div>  <div style="display: flex; justify-content: space-around; margin-top: 10px;">    </div> <div style="display: flex; justify-content: space-around; margin-top: 10px;">    </div>
1.1	¿Estado del Motor?			
1.2	¿Estado del nivel de aceite de Motor?			
1.3	¿Estado de aletas de refrigeración?			
1.4	¿Estado del filtro de aire?			
1.5	¿Estado del estanque de combustible (presenta fugas)?			
1.6	¿Estado del nivel de aceite hidráulico?			
1.7	¿Estado del Embrague?			
1.8	¿Estado protección de polea?			
1.9	¿Estado de Batería (se encuentra en condiciones y protegida)?			
1.10	¿Estado de barra de mando?			
1.11	¿Estado fuelle inferior?			
1.12	¿Estado de mandos direccionales?			
1.13	¿Estado marcha de avance?			
1.14	¿Estado marcha de retroceso?			
1.15	¿Estado cubierta protectora?			
1.16	¿Estado del acelerador?			
1.17	¿Estado piola de acelerador?			
1.18	¿Estado de Corta corriente?			
2.	<b>OPERACIÓN</b>	B	M	
2.1	¿El operador del equipo está capacitado? (Conoce el Instructivo)			
2.2	¿El operador cuenta con protección auditiva y respiratoria?			
2.3	¿cuenta con guantes apropiados?			
2.4	¿El área de operación esta exenta de condiciones desfavorables para su operación?			
2.5	¿El lugar de almacenamiento es adecuado y se encuentra despejado?			
2.7	¿La superficie de trabajo es adecuada para su operación?			

Figura 34. Formulario con imagen de referencia.