

CAPÍTULO I

GENERALIDADES

I Generalidades

I.1 Planteamiento del problema

Una problemática identificada en la 2ª Compañía Santa Bárbara SAR Bolivia – La Paz es la falta de un sistema centralizado y automatizado que permita administrar de manera eficiente los procesos críticos de la institución. Actualmente, la información relacionada con emergencias, inventarios, donativos, capacitaciones, personal y credenciales se gestiona de manera manual, lo que ocasiona pérdida de datos, retrasos en la toma de decisiones y vulnerabilidad en la seguridad institucional.

Este problema se manifiesta a través de diversas causas interrelacionadas:

- La dispersión de información en documentos físicos y hojas de cálculo.
- La inexistencia de un mecanismo digital confiable para la verificación de credenciales, lo que incrementa el riesgo de suplantación.
- La falta de trazabilidad en el inventario y donativos, lo que genera duplicidad, pérdidas o asignación ineficiente de recursos.
- Procesos manuales en la elaboración de reportes e informes, que demandan tiempo y esfuerzo adicional.
- Limitaciones en la comunicación y coordinación de recursos durante emergencias.
- En base a lo mencionado, el problema central que aborda la presente investigación se puede definir de la siguiente manera:

Ausencia de un sistema informático integral que centralice, automatice y garantice la seguridad de la información operativa y administrativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz.

I.2 Marco Teórico

El presente capítulo desarrolla los fundamentos teóricos que sustentan el proyecto denominado “MEJORAMIENTO DE LA GESTION OPERATIVA PARA LA 2a CIA. DE BOMBEROS SANTA BARBARA SAR BOLIVIA-LA PAZ A TRAVEZ DE LAS TIC” el cual tiene como propósito optimizar la gestión administrativa, operativa y de recursos de la institución mediante la implementación de un sistema informático integral.

Este marco teórico se estructura en torno a dos dimensiones principales:

- La variable independiente, que abarca las teorías relacionadas con la gestión operativa y administrativa en organizaciones de emergencia, así como los principios de modernización institucional y gestión pública aplicados al contexto del voluntariado de rescate.
- La variable dependiente, que comprende los fundamentos de informática, ingeniería de software y tecnologías de la información y comunicación (TIC) aplicados al desarrollo e implementación del sistema propuesto.

I.2.1 Variable Independiente: Gestión Operativa y Administrativa en Organizaciones de Emergencia

I.2.1.1 Gestión Operativa en Cuerpos de Emergencia

La gestión operativa puede entenderse como el conjunto de procesos que permiten planificar, coordinar, ejecutar y controlar las actividades necesarias para garantizar la atención eficiente de emergencias, rescates y situaciones críticas. Según Chiavenato (2009), la gestión operativa implica el uso eficiente de recursos humanos, materiales y tecnológicos para alcanzar los objetivos estratégicos de una organización.

En el contexto de los cuerpos de bomberos voluntarios, esta gestión adquiere un valor especial, ya que no solo busca la eficacia en las intervenciones, sino también la seguridad, la trazabilidad y la coordinación en tiempo real. La gestión operativa en este tipo de instituciones requiere disponer de información precisa sobre la disponibilidad del personal, los recursos materiales y el estado de los equipos, así como un control constante de las actividades y capacitaciones de los bomberos activos.

En la actualidad, los procesos manuales y el uso de registros físicos generan dificultades en la recopilación y administración de datos, ocasionando pérdida de información, duplicidades y una disminución de la eficiencia operativa. Por ello, la implementación de herramientas digitales permite mejorar significativamente la capacidad de respuesta ante emergencias, al facilitar el acceso inmediato a información confiable y actualizada.

I.2.1.2 Gestión Administrativa Institucional

La gestión administrativa, de acuerdo con Fayol (1916), se basa en funciones esenciales como planificar, organizar, dirigir y controlar los recursos de una institución. En el caso de los bomberos voluntarios, esta gestión se ve limitada por la falta de mecanismos que permitan la centralización y automatización de la información institucional.

El modelo de gestión tradicional, sustentado en el manejo de documentos físicos y procedimientos manuales, resulta insuficiente ante las necesidades operativas actuales. Una administración eficiente requiere el uso de herramientas que posibiliten el control de inventarios, el seguimiento de donativos, la actualización de la información del personal y la trazabilidad de las intervenciones.

Implementar un sistema informático en este ámbito contribuye a:

- Reducir errores administrativos.
- Aumentar la transparencia institucional.
- Facilitar la planificación de recursos humanos y materiales.
- Fortalecer la rendición de cuentas hacia la ciudadanía y organismos de coordinación interinstitucional.

De esta manera, la gestión administrativa eficiente constituye la base sobre la cual se sustenta la gestión operativa moderna, haciendo imprescindible la incorporación de tecnologías de la información como medio de transformación institucional.

I.2.1.3 Modernización Institucional y Digitalización de Procesos

La transformación digital es el proceso mediante el cual las organizaciones adoptan tecnologías para optimizar sus procedimientos y mejorar su desempeño (Ferreira & Silva, 2012). En el contexto de los cuerpos de emergencia, esta transformación implica la creación de sistemas informáticos que centralicen información crítica y permitan la coordinación efectiva entre distintas áreas y actores operativos.

La modernización institucional no solo responde a una tendencia tecnológica, sino a una necesidad organizativa. Según la Comisión Económica para América Latina y el Caribe (CEPAL, 2020), la digitalización permite incrementar la eficiencia administrativa, reducir costos operativos y fortalecer la transparencia institucional.

El sistema desarrollado para la 2ª Cía. Santa Bárbara SAR responde a este paradigma, al proporcionar una herramienta que integra los distintos procesos de la compañía —emergencias, inventarios, capacitaciones, personal y credenciales— bajo una plataforma única, accesible y segura.

I.2.1.4 Seguridad Institucional y Verificación de Identidad

Uno de los principales problemas identificados en el ámbito operativo de las organizaciones de rescate es la falsificación de credenciales y la suplantación de identidad. La ausencia de mecanismos digitales verificables genera vulnerabilidades que afectan la seguridad institucional y la confianza ciudadana.

La incorporación de credenciales digitales con código QR representa una solución innovadora, que permite validar en tiempo real la autenticidad de un bombero operativo. Este mecanismo fortalece la transparencia y asegura que únicamente el personal acreditado participe en intervenciones y misiones de rescate.

I.2.2 Variable Dependiente: Ingeniería de Software y Tecnologías de la Información Aplicadas

I.2.2.1 Ingeniería de Software

La ingeniería de software, según Pressman (2014), es la aplicación sistemática de principios científicos y técnicos al diseño, desarrollo, mantenimiento y documentación de programas de computadora. Su propósito es garantizar la calidad, confiabilidad y sostenibilidad del software a través de metodologías estructuradas.

En este proyecto, la ingeniería de software se aplica bajo la metodología Rational Unified Process (RUP), que organiza el ciclo de vida del desarrollo en fases iterativas: Inicio, Elaboración, Construcción y Transición. Este enfoque permite reducir riesgos, validar requerimientos de forma progresiva y entregar un producto funcional en cada iteración.

I.2.2.2 Metodología RUP

El Rational Unified Process (RUP) es una metodología orientada a objetos que proporciona un marco estructurado para el desarrollo de software, basado en buenas prácticas y orientado al control de calidad (Jacobson, Booch & Rumbaugh, 1999).

Las principales características de RUP incluyen:

- Desarrollo iterativo e incremental.
- Enfoque en la gestión de riesgos.
- Documentación continua y verificable.
- Enfoque centrado en la arquitectura del sistema.

En el proyecto de la 2ª Cía. Santa Bárbara SAR, RUP se implementó para estructurar el trabajo en fases claramente definidas, asegurando que cada entregable (modelos UML, especificación IEEE 830, código fuente, base de datos y manuales) se valide en su contexto.

I.2.2.3 Modelado UML (Unified Modeling Language)

El Lenguaje Unificado de Modelado (UML) es un estándar internacional que permite representar de manera visual la estructura y comportamiento de un sistema. Su uso contribuye a mejorar la comunicación entre desarrolladores, usuarios y partes interesadas.

En el presente proyecto se elaboraron diversos diagramas UML:

- Casos de Uso del Negocio: representación de las interacciones entre las áreas institucionales.
- Casos de Uso del Sistema: descripción de los servicios funcionales del software.
- Diagramas de Actividades: flujos de trabajo y procesos automatizados.
- Diagramas de Secuencia: interacción temporal entre los componentes del sistema.
- Diagrama de Clases: estructura de datos y relaciones lógicas.

El modelado UML permitió visualizar el sistema antes de su implementación, facilitando la detección temprana de inconsistencias y la optimización del diseño.

I.2.2.4 Tecnologías de la Información y Comunicación (TIC)

Las TIC constituyen el conjunto de recursos, herramientas y equipos utilizados para procesar, almacenar y comunicar información. Según la UNESCO (2018), su integración en instituciones públicas y organizaciones sociales promueve la eficiencia, transparencia y sostenibilidad de los procesos.

En este proyecto, las TIC son la base de la digitalización institucional, mediante el uso de herramientas modernas y de código abierto como React, Node.js, Express, PostgreSQL y Tailwind CSS. Estas tecnologías permiten desarrollar sistemas web accesibles, escalables y seguros.

I.2.2.5 Arquitectura Cliente-Servidor

El sistema propuesto se basa en la arquitectura cliente-servidor, donde el cliente (navegador web) solicita servicios al servidor (backend), que procesa los datos y responde con información estructurada. Este modelo garantiza la centralización de datos, la integridad de la información y el control de accesos.

La implementación de esta arquitectura permite que los usuarios institucionales —comandante, jefes de área y personal operativo— accedan a la plataforma desde cualquier dispositivo con conexión a internet, manteniendo seguridad y consistencia en los datos.

I.2.2.6 Base de Datos Relacional y Normalización

El manejo de información en el sistema se sustenta en una base de datos relacional desarrollada en PostgreSQL, que asegura la integridad de los datos a través de relaciones entre tablas y el cumplimiento de reglas de normalización hasta la Cuarta Forma Normal (4FN).

Este enfoque elimina redundancias, mejora la eficiencia en las consultas y garantiza la consistencia de los registros relacionados con emergencias, personal, capacitaciones, inventario y credenciales.

I.2.2.7 Seguridad de la Información

La seguridad es un componente crítico en todo sistema informático. Se aplicaron principios de confidencialidad, integridad y disponibilidad (CID), complementados con la encriptación de contraseñas mediante Bcrypt y el control de accesos por roles.

Asimismo, se incorporó la verificación de credenciales mediante código QR, lo que asegura la autenticidad de los bomberos activos y fortalece la confianza institucional.

I.2.2.8 Transformación Digital en el Sector Voluntario

Finalmente, este proyecto se enmarca dentro del proceso de transformación digital de las organizaciones de rescate voluntario, promoviendo el uso de software como herramienta para la transparencia y eficiencia social.

El desarrollo de este sistema no solo contribuye a mejorar la gestión operativa de la 2ª Cía. Santa Bárbara SAR, sino que sienta las bases para la replicabilidad del modelo en otras compañías y organizaciones de respuesta a emergencias en Bolivia.

I.2.3 Conclusión del Marco Teórico

La integración de los fundamentos teóricos de ambas variables permite comprender el alcance del proyecto en su totalidad:

- Desde la variable independiente, se abordaron las bases de la gestión operativa y administrativa en instituciones de emergencia, la necesidad de modernización y la importancia de la verificación de identidad.

- Desde la variable dependiente, se analizaron los principios de la ingeniería de software, las metodologías de desarrollo (RUP), los modelos UML y las tecnologías utilizadas para materializar la solución informática.

La combinación de estos enfoques proporciona el sustento teórico necesario para demostrar que la aplicación de tecnologías de información adecuadas puede transformar positivamente la gestión institucional de los cuerpos de bomberos voluntarios, fortaleciendo su eficiencia, transparencia y capacidad de respuesta.

I.3 Estado del Arte

El presente apartado tiene por finalidad analizar el estado actual del problema abordado en esta investigación, tanto desde un enfoque teórico (conceptual) como desde una perspectiva tecnológica, con el propósito de identificar las soluciones existentes computacionales o manuales que se han implementado en contextos similares y de las cuales se puedan obtener referencias, aprendizajes o mejoras.

En este sentido, el Estado del Arte constituye un punto de referencia que permite determinar los avances alcanzados en la digitalización de los procesos de gestión operativa y administrativa en cuerpos de bomberos y organizaciones de emergencia, así como las limitaciones que aún persisten y que justifican el desarrollo del sistema propuesto para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz.

I.3.1 Estado del Arte Conceptual

Desde una perspectiva conceptual, la gestión operativa y administrativa en los cuerpos de bomberos y organizaciones SAR (Search and Rescue) continúa siendo un proceso complejo debido a la multiplicidad de áreas, actores y recursos involucrados.

A nivel teórico, este tipo de instituciones se rige por principios de coordinación, eficiencia y trazabilidad, que son fundamentales para la atención oportuna de emergencias y la seguridad de las operaciones.

Autores como Chiavenato (2009) y Fayol (1916) sostienen que una gestión efectiva requiere una planificación adecuada, organización funcional y control de procesos para asegurar el cumplimiento de los objetivos institucionales. En el caso de los cuerpos de emergencia, estos principios se traducen en la necesidad de mantener una administración que garantice la disponibilidad de recursos humanos y materiales, así como un flujo de información confiable para la toma de decisiones.

Sin embargo, en la práctica, la mayoría de las compañías de bomberos en Bolivia y en Latinoamérica continúan utilizando métodos manuales de registro y comunicación interna. El uso de planillas físicas, cuadernos o documentos dispersos dificulta la trazabilidad de los datos, la transparencia administrativa y el control logístico.

Otro aspecto crítico identificado es la falsificación de credenciales físicas. Estas tarjetas, aunque funcionales, son fácilmente replicables o deteriorables, lo que pone en riesgo la seguridad institucional y la credibilidad frente a la población. La verificación manual de identidad carece de

mecanismos de validación digital, lo que genera vulnerabilidades tanto en operativos conjuntos como en misiones de rescate.

El contexto conceptual actual, por tanto, se caracteriza por un modelo de gestión tradicional que, aunque cumple parcialmente con los objetivos operativos, resulta insuficiente frente a las demandas tecnológicas contemporáneas. La tendencia mundial hacia la transformación digital ha demostrado que las instituciones que integran las Tecnologías de la Información y Comunicación (TIC) en sus procesos administrativos logran una mayor eficiencia, seguridad y transparencia, especialmente en sectores de atención a emergencias.

De esta forma, el marco conceptual del problema evidencia la necesidad de una solución tecnológica integral que permita la automatización de procesos internos, la digitalización documental, la trazabilidad operativa y la implementación de mecanismos de verificación de identidad confiables.

I.3.2 Estado del Arte Tecnológico

Desde la perspectiva tecnológica, se observa un avance considerable en el desarrollo de sistemas de información aplicados a la gestión de emergencias y administración institucional, tanto a nivel internacional como regional. No obstante, la mayoría de estas soluciones presentan limitaciones al momento de ser adaptadas a las condiciones operativas y presupuestarias de los cuerpos de bomberos voluntarios bolivianos.

A continuación, se describen las principales experiencias identificadas:

I.3.2.1 Soluciones Internacionales

I.3.2.1.1 SICOPA – Chile (Sistema de Compañías de Bomberos)

Plataforma institucional desarrollada para la administración de recursos humanos, materiales y operativos. Permite registrar intervenciones, gestionar inventarios, controlar personal y generar credenciales digitales.

A pesar de su eficacia, SICOPA es un sistema propietario con licencias comerciales y requerimientos de infraestructura tecnológica avanzada, lo que dificulta su implementación en instituciones voluntarias con recursos limitados.

I.3.2.1.2 SIGER – Colombia (Sistema de Información para la Gestión del Riesgo)

Desarrollado por la Unidad Nacional de Gestión del Riesgo de Desastres. Está enfocado en la coordinación institucional a nivel nacional, integrando información de múltiples organismos de respuesta. Sin embargo, no contempla la gestión interna de las compañías de bomberos o unidades SAR locales.

I.3.2.1.3 FireHouse Software – Estados Unidos

Software comercial especializado en cuerpos de bomberos. Ofrece módulos para la administración de emergencias, personal, flota vehicular, reportes y estadísticas. Su principal

limitación es el alto costo de licenciamiento y la dependencia de servidores externos, lo que lo hace inviable para organizaciones sin financiación estatal.

I.3.2.2 Soluciones Nacionales

I.3.2.2.1 Plataforma SAR Bolivia (2021)

Iniciativa del Grupo Voluntario de Salvamento Bolivia (GVSB-SAR), orientada a registrar y centralizar información de los voluntarios activos a nivel nacional. Aunque representa un avance significativo en la organización institucional, su alcance es limitado, ya que no incluye módulos de inventario, capacitación ni gestión de emergencias.

I.3.2.2.2 Sistema de Emergencias 114 – Gobierno Autónomo Municipal de La Paz

Sistema municipal que centraliza llamadas de emergencia y coordina con la Policía, hospitales y unidades de rescate. Su enfoque está en la atención al ciudadano y la respuesta inmediata, más no en la administración interna de los recursos de cada compañía.

I.3.2.2.3 Registros Manuales y Planillas Físicas

En la mayoría de las compañías de bomberos del país, incluyendo la 2ª Cía. Santa Bárbara SAR, la información de emergencias, inventario y personal se gestiona en formato físico o mediante hojas de cálculo aisladas. Este método, aunque de bajo costo, presenta grandes desventajas en seguridad, trazabilidad y capacidad de análisis.

I.3.3 Cuadro Comparativo de Soluciones Existentes

Solución / País	Tipo de sistema	Características principales	Ventajas	Limitaciones
SICOPA (Chile)	Web Institucional	Gestión integral de personal, inventario, emergencias y credenciales.	Integración completa de módulos y reportes automatizados.	Sistema propietario y de alto costo. Requiere infraestructura avanzada.
SIGER (Colombia)	Plataforma Nacional	Coordinación interinstitucional y monitoreo del riesgo.	Gestión centralizada a nivel país.	No enfocado en la administración interna de unidades locales.
FireHouse (EE. UU.)	Comercial	Gestión de emergencias, personal, vehículos y estadísticas.	Sistema probado con soporte técnico.	Elevado costo y dependencia de servicios externos.
Plataforma SAR Bolivia	Interna / Nacional	Registro de voluntarios SAR.	Centralización de datos de personal activo.	No contempla gestión de recursos, inventario ni emergencias.
Sistema de Emergencias 114 (Bolivia)	Público / Municipal	Centralización de llamadas de emergencia.	Mejora la coordinación institucional.	No gestiona información operativa interna.
Procesos Manuales (Santa Bárbara SAR)	Físico / Local	Registros en papel y hojas de cálculo.	Bajo costo de implementación.	Riesgo de pérdida de datos, baja trazabilidad y sin control de autenticidad.

Tabla 1. Cuadro Comparativo de soluciones existentes
Fuente: Elaboración Propia.

I.3.4 Análisis Comparativo y Conclusiones

El análisis comparativo evidencia que, si bien existen esfuerzos significativos a nivel regional e internacional para digitalizar la gestión de emergencias, no hay una solución adaptada a las características y necesidades operativas de los cuerpos de bomberos voluntarios en Bolivia.

Las plataformas internacionales ofrecen funcionalidades avanzadas, pero resultan económicamente inviables o requieren infraestructura tecnológica que las instituciones locales no poseen. Por otro lado, las iniciativas nacionales se encuentran en fases iniciales o de alcance limitado, centrándose en el registro de personal o la atención ciudadana, sin abordar de manera integral la gestión administrativa interna.

Los procesos manuales, aún predominantes, limitan la trazabilidad de la información, generan redundancias y dificultan la toma de decisiones basada en datos. Esta situación justifica plenamente la creación de una plataforma web modular y escalable, adaptada a la realidad de los bomberos bolivianos, que permita centralizar y automatizar los procesos esenciales de su labor operativa.

Por lo tanto, el sistema propuesto en esta investigación representa una innovación tecnológica local, que no solo responde a las limitaciones actuales, sino que también constituye un modelo potencialmente replicable en otras compañías SAR del país.

I.4 Objetivos

I.4.1 Objetivo General

Mejorar la gestión operativa de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz.

I.4.2 Objetivos Específicos

- Sistema web para el mejoramiento de la gestión operativa de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz a través de las TIC.
- Capacitación al personal para el manejo óptimo del sistema web de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz.

I.5 Matriz del Marco lógico (MML)

La Matriz de Marco Lógico presenta una forma resumida de los aspectos más importantes del proyecto con la ayuda de los puntos anteriores. Esta matriz suministra la siguiente información mediante sus columnas:

- Resumen narrativo de los objetivos y las actividades.
- Indicadores, que serían los resultados específicos a alcanzar.
- Medios de verificación, que verifican los indicadores.
- Supuestos, son factores que están fuera del control de la gerencia del proyecto

Y mediante sus filas, la matriz suministra la siguiente información:

- Fin, es una descripción de la solución de los problemas de nivel superior y nacional.
- Propósito, describe el resultado al final del periodo de ejecución.
- Componentes, son las obras, estudios, capacitaciones específicas que necesitan ser desarrollador por el gerente del proyecto.
- Actividades, son aquellas que el gerente tiene que ejecutar para llevar a cabo el desarrollo de un componente.

Resumen Narrativo	Indicadores	Medios de verificación	Supuestos
Fin Contribuir a mejorar la administración operativa de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz.	A un año de haber implementado el sistema de gestión operativa en la compañía, en 2024, al menos un 80% del personal de la compañía expresa un grado de satisfacción por la eficiente gestión de la información.	Cuadro comparativo de encuestas realizadas en el año base y luego de un año, con referencia a la percepción del personal de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz sobre la eficiencia en la gestión administrativa.	La 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz mejora la condición tecnológica y administrativa necesaria para el correcto funcionamiento del sistema.
Objetivo General (Propósito) Mejorar la gestión operativa de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz.	Al finalizar el proyecto, se han automatizado al menos un 90% de los procesos inherente a la gestión operativa de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz. $AP = \frac{NRO\ DE\ PROCESOS\ AUTOMATIZADOS}{NRO\ TOTAL\ DE\ PROCESOS} * 100$ $AP = (9/10) * 100 = 90\%$ • Iniciar Sesión • Administrar Usuario • Administrar Rol • Administrar Vehículos • Administrar Personal • Administrar Capacitaciones • Administrar Donativos • Administrar Emergencias • Administrar Inventario • Administrar Credenciales	Carta de Conformidad que exprese satisfacción por parte de la comandante de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz.	Se dispone del tiempo, los recursos y el equipo necesario para optimizar la gestión operativa de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz.

Objetivos Específicos (Componentes) C1.- Sistema web para el mejoramiento de la gestión operativa de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz a través de las TIC.	Al finalizar el proyecto se ha completado el desarrollo del sistema web para la gestión operativa, de acuerdo a los requerimientos expresados bajo la norma IEEE 830.	Carta de conformidad del docente de taller III acerca del cumplimiento del sistema.	El personal de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz hace uso del sistema. La compañía cuenta con el equipamiento necesario para la implementación del proyecto														
C2.- Capacitación al personal para el manejo optimo del sistema, implementado.	Al finalizar el proyecto se ha realizado la capacitación de aproximadamente de 20 a 30 bomberos.	Carta de informe de capacitación adjuntando lista de los participantes avalada por comandancia la comandante de la 2a Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz.	El personal de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz asiste a la capacitación														
Actividades Componente 1: • Sistema Web • Etapa de inicio • Etapa de elaboración • Etapa de construcción • Etapa de pruebas Componente 2:	<table><tr><th colspan="2">Resumen presupuestario</th></tr><tr><th>Rubro</th><th>Total</th></tr><tr><td>COMPONENTE CAPACITACION, ASISTENCIA TECNICA Y ORGANIZACION</td><td>1098 bs.</td></tr><tr><td>COMPONENTE ADMINISTRACION</td><td>0 bs.</td></tr><tr><td>COMPONENTE CONSULTORIA (ESTUDIOS, INVESTIGACION Y CONSTRUCCINES)</td><td>35000 bs.</td></tr><tr><td>COMPONENTE INSUMOS (MATERIALES Y SUMINISTROS)</td><td>8950 bs.</td></tr><tr><td>COMPONENTE CONSULTORIA EQUIPAMINETO</td><td>12190 bs.</td></tr></table>	Resumen presupuestario		Rubro	Total	COMPONENTE CAPACITACION, ASISTENCIA TECNICA Y ORGANIZACION	1098 bs.	COMPONENTE ADMINISTRACION	0 bs.	COMPONENTE CONSULTORIA (ESTUDIOS, INVESTIGACION Y CONSTRUCCINES)	35000 bs.	COMPONENTE INSUMOS (MATERIALES Y SUMINISTROS)	8950 bs.	COMPONENTE CONSULTORIA EQUIPAMINETO	12190 bs.	Informe presupuestario del proyecto, avalado por el director.	Disponibilidad de los recursos humanos y herramientas necesarias para el desarrollo del proyecto.
Resumen presupuestario																	
Rubro	Total																
COMPONENTE CAPACITACION, ASISTENCIA TECNICA Y ORGANIZACION	1098 bs.																
COMPONENTE ADMINISTRACION	0 bs.																
COMPONENTE CONSULTORIA (ESTUDIOS, INVESTIGACION Y CONSTRUCCINES)	35000 bs.																
COMPONENTE INSUMOS (MATERIALES Y SUMINISTROS)	8950 bs.																
COMPONENTE CONSULTORIA EQUIPAMINETO	12190 bs.																

• Capacitación a miembros de la compañía • Definición de medios y estrategias • Capacitación del manejo del sistema Web.	(MAQUINARIA, EQUIPO Y VEHICULOS)			
	COMPONENTE OTROS ACTIVOS REALES (ESCRITORIOS, MESAS, ENTRE OTROS)	6220 bs.		
	Total, del Proyecto	63458 Bs		

Tabla 2. Matriz de Marco Lógico
Fuente: Elaboración Propia.

I.6 Cronograma de Actividades

Nº	Actividad	Nro. días	Fecha inicio	Fecha Finalización.	M 1	M2	M3	M4	M5	M6	M7	M8
1	Sistema informativo		17/02/2025	17/10/2025	X	X	X	X	X	X	X	X
2	Relevamiento de información según la norma IEEE830 Determinación de los requerimientos	28	17/02/2025	17/03/2025	X							
3	Análisis y diseño del sistema Diagramas de casos de uso Diagramas de Secuencia Diagramas de clases Diagramas de Actividades	61	18/03/2025	17/05/2025		X	X					
4	Programación y construcción del sistema Desarrollo del código de programación del sistema	123	18/05/2025	17/09/2025			X	X	X	X		
5	Pruebas y validación del sistema Pruebas de las funciones del sistema.	30	19/09/2025	17/10/2025						X	X	
6	Capacitación al personal Planificación para la capacitación. Elaboración del material para la capacitación. Desarrollo de la capacitación al personal	30	18/10/2025	17/11/2025							X	X
	TOTAL, DIAS	213										

Tabla 3. Cronograma de Actividades
Fuente: Elaboración Propia.

CAPÍTULO II

COMPONENTES

II Componentes

II.1 Componente I: Sistema web para el mejoramiento de la gestión operativa de la 2ª Compañía de Bomberos Santa Barbara SAR Bolivia- La Paz a través de las TIC

II.1.1 Marco Teórico

II.1.1.1 Antecedentes

Durante las últimas décadas, la transformación digital ha modificado profundamente la forma en que las instituciones públicas, privadas y voluntarias gestionan su información y procesos internos. En particular, las organizaciones dedicadas a la respuesta ante emergencias —como cuerpos de bomberos, unidades de rescate y grupos SAR (Search and Rescue)— han incorporado progresivamente las Tecnologías de la Información y Comunicación (TIC) como parte de su estructura operativa.

Según Pressman (2014), la automatización de procesos administrativos y operativos mediante sistemas informáticos contribuye significativamente a la eficiencia institucional, reducción de errores humanos, y mejora en la trazabilidad de la información crítica. En el contexto de gestión de emergencias, estas ventajas se traducen en una capacidad superior de respuesta, mayor control de recursos y una comunicación más efectiva entre los diferentes niveles jerárquicos.

A nivel internacional, países como Chile, México, Colombia y Perú han implementado sistemas digitales en sus cuerpos de bomberos que permiten el registro en tiempo real de emergencias, el control de inventario táctico, la asignación de personal operativo y la emisión de credenciales verificables para los miembros activos. Dichos sistemas representan un paso fundamental hacia la profesionalización y transparencia institucional.

En Bolivia, la adopción de soluciones tecnológicas en este ámbito aún es limitada. Si bien algunas entidades públicas y organizaciones de socorro han comenzado a digitalizar sus registros, muchas compañías de bomberos voluntarios todavía dependen de métodos manuales o de herramientas ofimáticas dispersas para la administración de su información. Este escenario provoca ineficiencias operativas, duplicación de datos y dificultades en la trazabilidad de la información institucional.

Ante esta realidad, la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz identifica la necesidad de modernizar su gestión operativa a través del desarrollo de un sistema web que centralice y automatice sus procesos clave: administración de personal, registro de emergencias, control de inventario, seguimiento de capacitaciones y emisión de credenciales digitales. Este proyecto busca fortalecer la institucionalidad, mejorar la coordinación interna y garantizar una atención más eficiente ante las emergencias que enfrenta la población.

II.1.1.2 Gestión Operativa Institucional

La gestión operativa institucional puede entenderse como el conjunto de actividades, procedimientos y mecanismos que permiten coordinar y ejecutar de manera eficiente los recursos humanos, materiales y tecnológicos de una organización. En el caso de las instituciones de

emergencia, esta gestión representa la base sobre la cual se planifican las acciones de respuesta, prevención y capacitación.

De acuerdo con Chiavenato (2017), la gestión operativa constituye el nivel táctico de la administración, donde los procesos se transforman en actividades concretas orientadas a cumplir los objetivos estratégicos de la institución. En organizaciones de respuesta ante emergencias, este nivel adquiere especial relevancia, ya que la efectividad operativa depende directamente de la precisión y oportunidad de la información.

Para una compañía de bomberos voluntarios, la gestión operativa abarca la planificación de guardias, la asignación de recursos a emergencias, el mantenimiento de vehículos y equipos, la administración de donativos, la capacitación del personal y el seguimiento de las actividades diarias. Cada una de estas áreas genera datos que deben ser registrados, validados y comunicados a distintos niveles jerárquicos, garantizando así la continuidad y la trazabilidad institucional.

En este sentido, el uso de sistemas informáticos integrados permite no solo optimizar los procesos administrativos, sino también convertir la información en un recurso estratégico para la toma de decisiones. Un sistema web de gestión operativa facilita la automatización de tareas repetitivas, la reducción del uso de papel, la eliminación de registros redundantes y la disponibilidad en tiempo real de datos confiables.

Además, el enfoque digital fortalece la transparencia institucional y permite mantener un registro histórico que puede ser auditado o consultado para fines administrativos, formativos o legales. En el caso de la 2ª Cía. Santa Bárbara SAR, esta gestión constituye el eje central para mejorar la eficiencia interna y la coordinación interinstitucional ante eventos de emergencia.

II.1.1.3 Función de la Gestión Operativa en la Institución

La función de la gestión operativa dentro de una institución de respuesta a emergencias consiste en planificar, coordinar, ejecutar y controlar las acciones que permiten alcanzar los objetivos institucionales de forma eficiente y segura. En la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia La Paz, esta función se traduce en garantizar que cada operación desde la atención a emergencias hasta la formación del personal se realice de manera organizada, documentada y con trazabilidad.

La gestión operativa cumple funciones específicas dentro de la estructura institucional, entre las que destacan:

- Planificación y control de recursos: incluye la asignación de equipos, vehículos y materiales para garantizar la disponibilidad inmediata ante una emergencia.
- Registro y seguimiento de emergencias: permite mantener un historial documentado de incidentes, identificando tiempos de respuesta, recursos utilizados y resultados obtenidos.
- Administración del personal: involucra el registro de datos, jerarquías, grados, capacitaciones y certificaciones, garantizando la profesionalización del cuerpo operativo.
- Gestión de inventarios y donativos: posibilita la trazabilidad del material táctico, donaciones recibidas, insumos médicos y equipamiento logístico.

- Capacitación continua: asegura que el personal operativo mantenga un alto nivel de preparación técnica y física mediante la planificación y seguimiento de programas formativos.

Estas funciones, cuando se ejecutan manualmente, suelen generar dispersión de información y retrasos administrativos. Por ello, la implementación de un sistema web institucional permitirá centralizar todos los procesos y garantizar la actualización constante de los datos.

Asimismo, la digitalización de la gestión operativa contribuye a mejorar la coordinación con otras entidades, como grupos SAR regionales, autoridades locales o unidades de salud, fortaleciendo el trabajo interinstitucional y la capacidad de respuesta de la compañía.

II.1.1.4 La 2ª Compañía de Bomberos Santa Barbara SAR Bolivia- La Paz

La 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz forma parte del Grupo Voluntario de Salvamento Bolivia SAR, organización sin fines de lucro dedicada a la atención de emergencias, rescates, incendios, apoyo humanitario y asistencia en desastres naturales. Su labor se rige por principios de solidaridad, disciplina, voluntariado y servicio a la comunidad.

La compañía tiene su sede en la ciudad de La Paz, y depende orgánicamente de la estructura nacional de Bolivia SAR, bajo la coordinación del Directorio Nacional y los Comandantes Departamentales. Está conformada por personal voluntario especializado en distintas áreas, tales como operaciones, logística, instrucción, medicina, finanzas y relaciones públicas, todas ellas integradas para garantizar la respuesta efectiva ante emergencias.

La misión de la compañía es proteger la vida, los bienes y el medio ambiente a través de acciones coordinadas de prevención, rescate y atención de emergencias. Su visión institucional se orienta a consolidarse como una unidad voluntaria de referencia nacional, reconocida por su profesionalismo, compromiso social y gestión eficiente.

Dentro de su estructura jerárquica, la 2ª Cía. Santa Bárbara SAR cuenta con un Comandante Departamental, jefaturas de secciones operativas y administrativas, y equipos de voluntarios distribuidos por turnos. Este esquema organizacional permite mantener la operatividad 24/7 y una coordinación efectiva entre las distintas áreas.

El trabajo voluntario de la compañía, aunque altamente comprometido, enfrenta desafíos asociados a la falta de herramientas tecnológicas integradas, la dispersión de información y la dependencia de procesos manuales. Por ello, el desarrollo del presente sistema web busca fortalecer la gestión institucional mediante la automatización de registros, la digitalización de documentos, el control de inventario, la generación de reportes y la emisión de credenciales verificables, contribuyendo así al fortalecimiento organizacional y a la transparencia operativa.

La implementación de esta solución tecnológica permitirá a la compañía no solo mejorar su desempeño interno, sino también servir como modelo para otras unidades SAR y cuerpos de bomberos del país, promoviendo una cultura institucional basada en la eficiencia, la trazabilidad y la innovación.

II.1.1.5 Metodología de Desarrollo

Este componente se construye siguiendo un enfoque iterativo–incremental, con fuerte énfasis en trazabilidad, gestión de riesgos y documentación progresiva. Como marco de proceso se adopta RUP (Rational Unified Process) y, como lenguaje de modelado, UML (Unified Modeling Language). Ambos se integran para transformar requerimientos institucionales en artefactos verificables (modelos, código y pruebas) hasta alcanzar un producto listo para operación.

II.1.1.5.1 Metodología RUP (Rational Unified Process)

II.1.1.5.1.1 Visión general

RUP es un proceso disciplinado para ingeniería de software que organiza el trabajo en cuatro fases (Inicio, Elaboración, Construcción y Transición) y ocho disciplinas que se ejecutan en distinto grado a lo largo del ciclo: Requisitos, Análisis y Diseño, Implementación, Pruebas, Despliegue, Gestión de Configuración y Cambios, Gestión de Proyecto y Entorno. Su naturaleza iterativa permite entregar valor temprano, reducir riesgos y mantener la arquitectura bajo control.

II.1.1.5.1.2 Fases, objetivos y artefactos de este proyecto

Inicio

- *Objetivos:* alinear alcance y objetivos con la 2ª Cía. Santa Bárbara SAR, identificar actores y riesgos, estimar tamaño y costo.
- *Artefactos clave:* Documento de Visión, Actores y Casos de Uso de alto nivel, Perfil del Proyecto, MML (Matriz de Marco Lógico), Riesgos Iniciales.
- *Hito:* Vision Approved.

Elaboración

- *Objetivos:* establecer la línea base de arquitectura, detallar requerimientos críticos y validar viabilidad técnica.
- *Artefactos:* Especificación de Requerimientos (IEEE 830), Modelo de Casos de Uso del Sistema, Diagramas UML (actividad, secuencia, clases preliminares), Modelo de Datos (ER normalizado), Prototipos de interfaz, Plan de Iteraciones.
- *Hito:* Architecture Baseline.

Construcción

- *Objetivos:* implementar incrementos funcionales y estabilizar el producto mediante pruebas continuas.
- *Artefactos:* Código fuente (React + Vite / Node.js + Express + Sequelize), Scripts SQL (PostgreSQL), Pruebas (unitarias, integración y caja negra), Manual técnico.
- *Hito:* Initial Operational Capability (IOC).

Transición

- *Objetivos*: preparar el sistema para la operación real, capacitar usuarios, cerrar incidencias y formalizar la aceptación.
- *Artefactos*: Manual de Usuario e Instalación, Plan y registro de Capacitaciones, Informe de Pruebas de Aceptación, Acta de Conformidad.
- *Hito*: Product Release.

II.1.1.5.1.3 Disciplinas y prácticas destacadas

- Requisitos: captura centrada en casos de uso y criterios de aceptación medibles.
- Análisis y diseño: definición de la arquitectura cliente–servidor; modelo de dominio, servicios y repositorios.
- Implementación: control de versiones con Git/GitHub, integración de seguridad (bcrypt), y convenciones de código para .jsx.
- Pruebas: estrategia en pirámide (unitarias → integración → sistema), trazadas a los casos de uso.
- Gestión de configuración y cambios: versionado semántico, ramas *feature/release/hotfix* y control de cambios documentado.
- Gestión de proyecto: iteraciones de 2–3 semanas con objetivos y demostraciones al final.
- Entorno: automatización con scripts de inicio, colección Postman, y tableros para seguimiento.

II.1.1.5.1.4 Roles principales en el proyecto

- Product Owner institucional (representante de la compañía).
- Analista–Arquitecto (modelado UML, decisiones de arquitectura).
- Desarrollador Frontend (React + Vite + Tailwind CSS).
- Desarrollador Backend (Node.js + Express + Sequelize).
- DBA (modelo y administración en PostgreSQL/PgAdmin).
- QA (plan, diseño y ejecución de pruebas).
- Líder de proyecto (planificación, riesgos, métricas).

II.1.1.5.1.5 Trazabilidad y criterios de salida

- Cadena de trazabilidad: Objetivo → Requisito → Caso de uso → Escenario → Prueba → Commit/Build.
- Criterios de salida por fase: cumplimiento de hitos, cobertura de pruebas planificada, riesgos críticos mitigados, documentación revisada.

II.1.1.5.2 UML (Lenguaje Unificado de Modelado)

UML es el estándar de facto para visualizar, especificar, construir y documentar sistemas orientados a objetos. En este proyecto:

- Sirve para comunicar con claridad entre usuarios clave y equipo técnico.
- Permite trazar requisitos con diseños y pruebas.
- Aporta consistencia entre vista funcional (casos de uso), dinámica (secuencia/actividad) y estática (clases/paquetes).

Beneficios aplicados:

- Reducción del retrabajo por malentendidos de requisitos.
- Diseño centrado en responsabilidades y acoplamiento bajo.
- Base sólida para generar pruebas y priorizar iteraciones.

II.1.1.5.3 Tipos de Diagramas Utilizados

II.1.1.5.3.1 Casos de Uso del Negocio

Delimitan procesos institucionales de alto nivel (p. ej., “Gestionar Emergencias”, “Administrar Inventario y Donativos”). Ayudan a vincular objetivos organizacionales con el sistema.

II.1.1.5.3.2 Casos de Uso del Sistema

Describen interacciones actor–sistema (Iniciar sesión, Registrar emergencia, Actualizar stock, Emitir credencial con QR, Gestionar capacitaciones, Generar reportes). Son la espina dorsal de requisitos y pruebas.

II.1.1.5.3.3 Diagramas de Actividad

Modelan flujos, decisiones y concurrencia. Se utilizan, por ejemplo, para “Atender emergencia” (asignación de personal, despacho, cierre) y para “Ingreso/egreso de inventario”.

II.1.1.5.3.4 Diagramas de Secuencia

Representan intercambio de mensajes entre objetos/servicios (Front React → API Express → Servicio → Repositorio → BD). Aclaran validaciones, errores y transacciones.

II.1.1.5.3.5 Diagrama de Clases

Define el modelo estático: entidades como Emergencia, Personal, Credencial, Inventario, Donativo, Capacitación, Usuario, Rol y sus relaciones (agregación, composición, cardinalidades).

II.1.1.5.4 Herramientas de Construcción de Software

Para el desarrollo del sistema web de gestión operativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, se emplearon diversas herramientas de modelado, desarrollo, control de versiones y pruebas, que aseguran agilidad, trazabilidad y calidad del software.

Estas herramientas se agrupan según su propósito dentro del proceso RUP:

II.1.1.5.4.1 Herramientas de Modelado UML

- PlantUML (Online/VS Code)
Aplicación de código abierto que permite generar diagramas UML a partir de texto descriptivo.
Se integra con Visual Studio Code y GitHub, facilitando el versionado y la regeneración automática de los diagramas UML (casos de uso, clases, secuencias y actividades).
- Mermaid
Lenguaje declarativo para crear diagramas directamente en Markdown o documentación técnica.
Se utilizó para representar flujos de actividad, cronogramas y procesos simples dentro de la documentación.
- Enterprise Architect
Herramienta profesional de Sparx Systems orientada al modelado estructurado y la documentación bajo RUP.
Se usó para consolidar los modelos UML y garantizar uniformidad en la representación de los diagramas.
- Visual Paradigm
Entorno visual de modelado orientado a procesos.
Permite diseñar diagramas UML y modelos Entidad-Relación (ERD), además de vincular requerimientos con elementos del diseño para mantener la trazabilidad entre fases.

II.1.1.5.4.2 Herramientas de Desarrollo y Control de Versiones

- Visual Studio Code (VS Code)
Editor de código fuente multiplataforma desarrollado por Microsoft.
Seleccionado por su integración con Git, su soporte de extensiones (PlantUML, Prettier, REST Client, etc.) y su compatibilidad con frameworks modernos (React, Node.js, Tailwind CSS).
- Git
Sistema de control de versiones distribuido que permite mantener un historial detallado de cambios y colaborar de manera segura entre varios desarrolladores.
- GitHub
Plataforma de alojamiento de repositorios basada en Git.
Fue utilizada como repositorio central del proyecto, asegurando la sincronización del trabajo en equipo, el seguimiento de ramas y la gestión de versiones.

II.1.1.5.4.3 Herramientas de Pruebas y Validación

- Postman
Aplicación para el diseño, prueba y documentación de APIs REST. Permite validar las rutas desarrolladas en el backend (Node.js/Express), comprobar la estructura de respuestas JSON y documentar los servicios de manera automatizada.
- PgAdmin 4
Herramienta gráfica oficial para la administración del sistema de bases de datos PostgreSQL.
Usada para la creación, prueba y mantenimiento del modelo relacional, ejecución de consultas SQL y generación de respaldos de la base de datos.

II.1.1.5.4.4 Herramientas Complementarias

- Bcrypt
Librería utilizada para el cifrado seguro de contraseñas, garantizando confidencialidad en el manejo de credenciales de usuario.
- Tailwind CSS
Framework de diseño basado en utilidades, usado para construir interfaces modernas, responsivas y eficientes sin dependencias pesadas.
- Vite
Herramienta de compilación rápida y empaquetado moderno, integrada en el frontend con React, que acelera el desarrollo y la recarga en tiempo real.
- Sequelize
ORM que facilita la conexión entre Node.js y PostgreSQL, gestionando las migraciones, relaciones y consultas SQL de forma estructurada.

II.1.1.5.4.5 Ventajas del ecosistema de herramientas

El conjunto de herramientas seleccionadas ofrece los siguientes beneficios:

- Integración fluida entre diseño, desarrollo y documentación.
- Versionado transparente de todos los artefactos mediante Git/GitHub.
- Pruebas controladas y repetibles gracias a Postman y PgAdmin.
- Modelado automatizado con PlantUML y Visual Paradigm.
- Seguridad y trazabilidad a nivel de datos y contraseñas mediante Bcrypt.
- Escalabilidad y mantenibilidad aseguradas por tecnologías modernas como React, Vite, Node.js y PostgreSQL.

II.1.2 Plan de Desarrollo de Software

II.1.2.1 Introducción

El presente *Plan de Desarrollo de Software* corresponde al proyecto titulado “Mejoramiento de la Gestión Operativa para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz a través de las Tecnologías de la Información y la Comunicación (TIC)”, desarrollado en el marco del proceso de fortalecimiento institucional y modernización tecnológica de la compañía.

Este plan describe de manera estructurada el conjunto de fases, procesos, técnicas y herramientas aplicadas para el diseño, desarrollo, validación e implementación del sistema web de gestión operativa, que constituye el principal producto tecnológico del proyecto.

El sistema propuesto tiene como finalidad optimizar la gestión operativa y administrativa de la 2ª Cía. de Bomberos Santa Bárbara SAR, integrando en una sola plataforma digital los procesos de atención de emergencias, control de inventario, administración de personal, capacitaciones institucionales y emisión de credenciales digitales verificables mediante código QR.

La propuesta surge como respuesta a la necesidad de superar las limitaciones generadas por la dependencia de registros manuales y la falta de un sistema de información centralizado. En la actualidad, la compañía enfrenta dificultades para consolidar datos, coordinar recursos, y mantener un control actualizado de su personal y equipamiento. Ante esta problemática, se plantea la incorporación de las TIC como herramienta estratégica para fortalecer la eficiencia, la trazabilidad y la transparencia institucional.

El desarrollo del sistema se basa en la metodología Rational Unified Process (RUP), reconocida por su enfoque iterativo e incremental, que permite gestionar de forma ordenada cada fase del ciclo de vida del software, garantizando la calidad, trazabilidad y satisfacción de los requerimientos del usuario final.

RUP estructura el proceso en fases claramente definidas:

- Inicio: definición del problema, identificación de actores y elaboración del perfil del proyecto.
- Elaboración: análisis de requerimientos, diseño arquitectónico y modelado UML del sistema.
- Construcción: desarrollo progresivo de los módulos, integración y pruebas.
- Transición: validación final, capacitación y entrega formal del sistema a la institución.

Cada una de estas fases genera artefactos verificables, tales como: modelos UML, especificaciones de requerimientos (IEEE 830), modelos entidad-relación, código fuente, documentación técnica y manuales de usuario.

Este *Plan de Desarrollo de Software* proporciona una visión integral del proyecto y establece la guía metodológica que permitirá garantizar la eficiencia, calidad y sostenibilidad del producto final, en concordancia con los objetivos del proyecto:

- Modernizar la gestión operativa de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia – La Paz.
- Implementar una solución tecnológica que digitalice los procesos institucionales.
- Fortalecer la capacidad operativa y administrativa mediante el uso de TIC.

Al concluir el proceso, se prevé que la compañía cuente con un sistema informático integral, robusto y escalable, que le permita administrar sus recursos de manera más eficiente, mejorar la coordinación durante las emergencias y fortalecer la transparencia institucional frente a la comunidad.

II.1.2.2 Propósito

El propósito del presente *Plan de Desarrollo de Software* es definir y organizar de manera estructurada el proceso metodológico, técnico y documental necesario para el desarrollo del sistema web destinado al mejoramiento de la gestión operativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, integrando los principios de ingeniería de software y las buenas prácticas establecidas por la metodología Rational Unified Process (RUP).

Este plan constituye una guía de referencia para todos los involucrados en el proyecto, estableciendo los lineamientos que orientan el diseño, implementación, validación y documentación del sistema, asegurando que el producto final cumpla con los objetivos institucionales definidos en el Marco Lógico del proyecto.

Su implementación permitirá mantener control y trazabilidad sobre cada una de las fases del desarrollo, garantizando que las decisiones técnicas y los entregables generados estén alineados con los requerimientos funcionales y no funcionales definidos bajo el estándar IEEE 830.

Entre los propósitos específicos de este plan se destacan los siguientes:

- Guiar el proceso de desarrollo del software mediante la metodología RUP, asegurando un enfoque estructurado, iterativo e incremental que permita validar avances en cada fase y minimizar riesgos durante el ciclo de vida del sistema.
- Establecer la planificación técnica y organizativa del proyecto, definiendo los artefactos, cronogramas y entregables que serán producidos en cada iteración, de acuerdo con la naturaleza y alcance del sistema web propuesto.
- Documentar los lineamientos técnicos y metodológicos, garantizando la transparencia, coherencia y reproducibilidad del proceso de desarrollo, conforme a las normativas internacionales de calidad en software.
- Asegurar la trazabilidad completa de los requerimientos, desde su especificación inicial hasta su validación final, mediante la integración de herramientas de modelado UML y control de versiones.
- Proporcionar un marco de gestión para la verificación y validación del producto, estableciendo los criterios de prueba y aceptación que permitan garantizar la calidad, seguridad y usabilidad del sistema.
- Favorecer la comunicación entre los diferentes actores del proyecto, incluyendo al equipo de desarrollo, la jefatura institucional y los usuarios finales, mediante documentación clara y artefactos gráficos de fácil comprensión (diagramas UML, prototipos de interfaz, manuales técnicos y de usuario).

Este documento no solo cumple una función metodológica, sino también estratégica, ya que consolida la visión y planificación necesarias para lograr un producto tecnológico confiable, escalable y alineado a las necesidades reales de la 2ª Cía. de Bomberos Santa Bárbara SAR.

En síntesis, el *Plan de Desarrollo de Software* busca garantizar que cada actividad, fase y entrega dentro del proyecto contribuya efectivamente al objetivo general de la tesis: mejorar la gestión operativa de la institución mediante la aplicación de las Tecnologías de la Información y la Comunicación (TIC).

II.1.2.3 Alcance

El alcance del *Plan de Desarrollo de Software* define los límites, características, funcionalidades y entregables esperados del sistema web de gestión operativa propuesto para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz.

Este proyecto abarca todas las etapas necesarias para el diseño, construcción, validación e implementación de una plataforma tecnológica integral, orientada a optimizar los procesos institucionales y mejorar la capacidad operativa de respuesta ante emergencias, bajo un enfoque de transformación digital institucional.

El sistema tendrá como eje principal la automatización de procesos críticos que actualmente se realizan de manera manual o con herramientas no integradas, logrando así una administración más eficiente, precisa y transparente de la información institucional.

II.1.2.3.1 Alcance funcional

El sistema web permitirá la gestión digital de las siguientes áreas operativas y administrativas de la compañía:

- Gestión de usuarios y roles: administración de cuentas institucionales con diferentes niveles de acceso según jerarquía (Comandancia, Jefes de Área, Operativos y Administrativos).
- Gestión de emergencias: registro de intervenciones, control de recursos humanos y materiales desplegados, historial de eventos y reportes de desempeño.
- Gestión de personal: manejo del registro del personal activo, asignaciones, jerarquías, historial de capacitaciones y turnos operativos.
- Gestión de inventario y donativos: control de materiales tácticos, equipos, suministros y recursos donados, con seguimiento de entradas, salidas y estado de los elementos.
- Gestión de capacitaciones: planificación, seguimiento y registro de actividades formativas internas y externas, incluyendo la emisión de certificados digitales.
- Gestión de credenciales digitales: emisión de credenciales institucionales con código QR verificable en línea, garantizando autenticidad y transparencia.
- Gestión de reportes: generación de informes operativos, estadísticos y administrativos, exportables en formatos estándar para su uso institucional o presentación externa.

II.1.2.3.2 Alcance técnico

El desarrollo del sistema se realizará mediante una arquitectura cliente-servidor, compuesta por un frontend desarrollado con *React* y *Vite*, y un backend basado en *Node.js* y *Express.js*, conectados a una base de datos relacional en *PostgreSQL*.

El sistema será accesible desde cualquier navegador web moderno, optimizado para equipos de escritorio y dispositivos móviles, y se implementará en un entorno multiplataforma con soporte de acceso remoto.

Las pruebas funcionales se realizarán siguiendo criterios de *caja negra* y validación incremental por fases, garantizando la estabilidad, seguridad y usabilidad del sistema antes de su despliegue final.

II.1.2.3.3 Alcance metodológico

El desarrollo se regirá bajo la metodología Rational Unified Process (RUP), que estructura el trabajo en cuatro fases iterativas:

- Inicio: definición del problema, alcance, objetivos, actores y requerimientos generales.
- Elaboración: levantamiento detallado de requerimientos bajo la norma IEEE 830, modelado UML y diseño arquitectónico.
- Construcción: codificación, integración y validación técnica de los módulos del sistema.
- Transición: pruebas finales, capacitación y entrega formal del producto a la institución.

Cada fase producirá artefactos específicos —diagramas UML, modelos entidad-relación, documentación técnica y manuales— que serán revisados y aprobados como evidencia de avance del proyecto.

II.1.2.3.4 Alcance institucional

El sistema será implementado dentro del entorno operativo de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia – La Paz, adaptado a sus procesos administrativos, estructura jerárquica y protocolos internos.

Si bien su diseño inicial responde a las necesidades particulares de esta compañía, la arquitectura modular y escalable del software permitirá su adaptación futura a otras unidades SAR o cuerpos de bomberos voluntarios del país.

II.1.2.3.5 Exclusiones

Quedan fuera del alcance del presente proyecto:

- La integración con sistemas externos o plataformas gubernamentales.
- La implementación de aplicaciones móviles nativas (aunque la interfaz web será adaptable a dispositivos móviles).
- El soporte multicompañía o multisede en esta versión inicial.
- La conexión con hardware físico de detección o monitoreo.

II.1.2.3.6 Resultados esperados

- Un sistema web operativo, funcional y validado institucionalmente.
- Documentación completa del software, incluyendo manuales técnicos y de usuario.
- Capacitación del personal en el uso de la plataforma.
- Incremento comprobable de la eficiencia en la gestión administrativa y operativa de la compañía.

En conclusión, el alcance del presente plan establece que el proyecto entregará un sistema informático robusto, escalable y seguro, orientado a fortalecer la capacidad institucional de la 2ª Cía. de Bomberos Santa Bárbara SAR, permitiendo consolidar un modelo digital de gestión operativa sustentado en las TIC.

II.1.2.4 Resumen

El presente *Plan de Desarrollo de Software* resume la metodología, alcance y lineamientos técnicos aplicados al desarrollo del sistema web orientado al mejoramiento de la gestión operativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz. Este documento se sustenta en la metodología Rational Unified Process (RUP), la cual permite estructurar el proyecto en fases iterativas e incrementales que garantizan una validación progresiva de resultados, reduciendo riesgos y mejorando la trazabilidad de los requerimientos a lo largo del ciclo de vida del software. En su conjunto, el plan establece las bases para la planificación, ejecución, seguimiento y control de todas las actividades del desarrollo, asegurando la coherencia entre los objetivos institucionales y el producto tecnológico final. Asimismo, define el propósito, alcance, entregables y restricciones del sistema, especificando la forma en que se abordarán las necesidades de la institución a través de la automatización de procesos críticos, la digitalización de la información y la implementación de herramientas modernas basadas en tecnologías web. Con este plan se busca garantizar la calidad, seguridad y sostenibilidad del software, proporcionando un marco de referencia que oriente el trabajo del equipo de desarrollo y permita obtener un sistema funcional, escalable y alineado con los estándares de ingeniería de software y las metas estratégicas de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia – La Paz.

II.1.2.5 Suposiciones y Restricciones

El desarrollo del sistema web para el mejoramiento de la gestión operativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz se apoya en un conjunto de suposiciones que sirven de base para la planificación del proyecto, así como en restricciones que delimitan su alcance técnico y operativo. Estas condiciones permiten comprender los factores internos y externos que influyen en la ejecución y garantizan que las decisiones de desarrollo se mantengan dentro de un marco realista y sostenible.

II.1.2.5.1 Suposiciones

- El personal involucrado en la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia – La Paz apoyará de manera constante y voluntaria, brindando la información necesaria para el correcto desarrollo del sistema.
- El sistema será diseñado para operar en un entorno multiplataforma, accesible desde distintos navegadores y dispositivos conectados a internet.
- La institución dispondrá de equipamiento básico de cómputo y conectividad que permita la correcta implementación y uso del sistema.
- Los usuarios finales participarán activamente en las capacitaciones programadas, adoptando la herramienta tecnológica como parte de los procesos institucionales.
- Se contará con la colaboración de responsables de cada área para validar los requerimientos y evaluar las pruebas del sistema en sus distintas fases.

II.1.2.5.2 Restricciones

- La información disponible en la institución es dispersa y en algunos casos redundante, lo que implica un esfuerzo adicional en la depuración y consolidación de datos para su migración al sistema.
- El proyecto se encuentra limitado en cuanto a tiempo y recursos, por lo que se priorizarán los módulos esenciales (emergencias, personal, inventario, donativos, capacitaciones y credenciales).
- El sistema será implementado únicamente en el entorno de la 2ª Cía. Santa Bárbara SAR Bolivia – La Paz, aunque podrá ser escalado posteriormente a otras compañías SAR.
- La conectividad a internet representa un factor crítico para la operatividad del sistema, pudiendo limitar su uso en situaciones de emergencia donde la red sea inestable.
- Por razones presupuestarias, se utilizarán exclusivamente herramientas y frameworks de código abierto, prescindiendo de licencias comerciales.

En conclusión, las suposiciones y restricciones aquí descritas permiten establecer los límites técnicos y organizacionales del proyecto, garantizando que su desarrollo se realice bajo parámetros realistas y sostenibles, en concordancia con la naturaleza institucional y los recursos de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia – La Paz.

II.1.2.6 Entregables del Proyecto

El desarrollo del sistema web para el mejoramiento de la gestión operativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz generará una serie de entregables que constituyen la evidencia tangible del avance y cumplimiento de los objetivos establecidos. Cada artefacto, documento o producto de software representa una fase específica dentro del ciclo de vida del proyecto, conforme a la metodología Rational Unified Process (RUP), que promueve la validación progresiva del sistema mediante iteraciones controladas.

Entregables por Fase:
Fase de Inicio

Durante esta etapa se definen los fundamentos conceptuales y estratégicos del proyecto, estableciendo los objetivos, actores y alcance del sistema.

Los principales entregables son:

- Documento de perfil del proyecto.
- Matriz de Marco Lógico (MML).
- Identificación de actores, procesos institucionales y requerimientos iniciales.
- Especificación preliminar de requerimientos funcionales y no funcionales conforme a la norma IEEE 830.

Fase de Elaboración

Se consolida la visión técnica del sistema, modelando los procesos esenciales mediante herramientas UML y definiendo la arquitectura general.

Los entregables incluyen:

- Documento de Especificación de Requerimientos de Software (SRS).
- Diagramas UML iniciales (casos de uso del negocio y del sistema, clases y actividades).
- Modelo Entidad-Relación normalizado hasta la Cuarta Forma Normal (4FN). Documento de arquitectura del sistema bajo el modelo Cliente-Servidor.

Fase de Construcción

En esta fase se desarrolla la aplicación web, implementando los módulos funcionales y validando su operatividad mediante pruebas técnicas.

Los principales entregables son:

- Desarrollo del sistema web funcional con los módulos: gestión de usuarios y roles, personal, emergencias, inventario, donativos, capacitaciones y credenciales digitales con código QR.
- Interfaces gráficas desarrolladas con React + Vite + Tailwind CSS. Código fuente documentado (frontend y backend).
- Scripts SQL para creación y carga inicial de la base de datos en PostgreSQL. Documentación técnica de instalación, configuración y despliegue.

Fase de Transición

Se prepara el sistema para su entrega e implementación final en la institución, garantizando su adopción mediante capacitación y validación de los usuarios.

Los entregables son:

- Manual de usuario dirigido al personal operativo y administrativo.
- Informe de pruebas funcionales y resultados de validación.
- Capacitación al personal en el uso del sistema.
- Carta de conformidad institucional firmada por la Comandancia de la 2ª Cía. Santa Bárbara SAR Bolivia – La Paz.

II.1.2.6.1 Plan de desarrollo de Software

El plan de desarrollo se estructura en fases iterativas y progresivas que permiten gestionar riesgos, validar requerimientos de forma continua y garantizar la calidad del producto final. Cada fase concluye con artefactos verificables y la retroalimentación directa de los usuarios institucionales.

Las etapas definidas son:

- Fase de Inicio: levantamiento del problema, elaboración del perfil del proyecto y definición preliminar de requerimientos.
- Fase de Elaboración: diseño arquitectónico, modelado UML y definición de base de datos.
- Fase de Construcción: desarrollo de módulos funcionales con tecnologías React, Node.js, Express.js, Sequelize y PostgreSQL, integrando seguridad mediante Bcrypt y códigos QR.
- Fase de Transición: validación institucional, capacitación de usuarios y despliegue final del sistema.

Este enfoque iterativo permite entregar versiones funcionales incrementales, fomentando la retroalimentación temprana y reduciendo el riesgo técnico en cada iteración.

II.1.2.6.2 Visión

La visión del proyecto es fortalecer la capacidad operativa y administrativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, mediante la implementación de un sistema web integral que digitalice, centralice y optimice sus procesos internos.

El sistema permitirá:

- Automatizar la gestión de emergencias, inventarios, personal y capacitaciones.
- Emitir credenciales digitales verificables con código QR, reforzando la transparencia institucional.
- Generar reportes y estadísticas en tiempo real que apoyen la toma de decisiones. Mejorar la comunicación interna y la trazabilidad de las operaciones.

A largo plazo, este desarrollo pretende convertirse en una herramienta tecnológica replicable para otras unidades SAR del país, contribuyendo al fortalecimiento de la gestión pública y voluntaria a través del uso estratégico de las TIC.

II.1.2.6.3 Limitaciones

El proyecto enfrenta limitaciones asociadas principalmente a la infraestructura tecnológica disponible, los recursos económicos y el tiempo de desarrollo. La compañía no dispone de servidores dedicados ni equipamiento de alto rendimiento, por lo que el sistema está diseñado para operar en equipos de gama media y con conectividad estándar.

El alcance institucional se limita a la 2ª Cía. Santa Bárbara SAR, sin contemplar, por el momento, su extensión a otras compañías. Sin embargo, la arquitectura modular del sistema permite una

futura escalabilidad. La participación activa del personal en el levantamiento y validación de información es un factor crítico; su ausencia o retraso puede afectar la calidad del producto final.

Finalmente, la dependencia del acceso a internet y la naturaleza voluntaria del personal constituyen factores externos que podrían incidir en la continuidad operativa del sistema, aunque se han previsto mecanismos de respaldo y capacitación para mitigar su impacto.

II.1.2.6.4 Glosario

En este apartado se presentan los términos más relevantes utilizados a lo largo del documento, con el fin de facilitar la comprensión del contenido técnico del proyecto y asegurar la coherencia conceptual entre los distintos apartados. Los términos incluyen tanto conceptos informáticos como elementos operativos relacionados con la gestión institucional de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz.

- API (Application Programming Interface): Conjunto de funciones y protocolos que permiten la comunicación entre diferentes componentes o aplicaciones de software, facilitando la integración del sistema web con otros servicios.
- Backend: Parte del sistema encargada de la lógica de negocio, la comunicación con la base de datos y la gestión de los servicios. En este proyecto se implementó utilizando Node.js y Express.js.
- Bcrypt: Algoritmo de hash basado en el cifrado Blowfish, utilizado para proteger contraseñas mediante técnicas de encriptación que previenen ataques de fuerza bruta.
- CRUD (Create, Read, Update, Delete): Conjunto de operaciones básicas para la manipulación y gestión de datos en un sistema de información.
- Código QR: Representación gráfica bidimensional utilizada en las credenciales digitales del personal operativo para verificar su autenticidad mediante escaneo desde un dispositivo móvil.
- Credencial Digital con Código QR: Documento generado por el sistema que identifica a un bombero activo, asegurando la validez de su pertenencia institucional mediante verificación electrónica.
- Emergencia: Situación inesperada que requiere intervención inmediata para salvaguardar vidas humanas, bienes materiales o el medio ambiente.
- Frontend: Parte visible e interactiva del sistema donde el usuario realiza sus operaciones, desarrollada con React, Vite y Tailwind CSS para garantizar una experiencia moderna, ágil y responsiva.
- Inventario Táctico: Conjunto de equipos, materiales y herramientas disponibles para la atención de emergencias, tales como radios, mangueras, oxígeno y trajes de protección.
- Iteración: Ciclo de desarrollo en el cual se planifican, construyen y validan funcionalidades específicas dentro del marco metodológico RUP, garantizando entregas parciales funcionales y verificables.
- Matriz de Marco Lógico (MML): Herramienta metodológica utilizada para planificar y evaluar proyectos, permitiendo definir objetivos, indicadores, medios de verificación y supuestos estratégicos.
- Normalización de Base de Datos: Proceso que organiza los datos en tablas y relaciones para evitar redundancias, mejorar la integridad de la información y optimizar las consultas.

- ORM (Object-Relational Mapping): Técnica de programación que permite interactuar con bases de datos relacionales mediante objetos de software, evitando escribir directamente consultas SQL.
- PgAdmin 4: Herramienta gráfica oficial para la administración de bases de datos PostgreSQL, utilizada para la gestión, monitoreo y mantenimiento de la base de datos del sistema.
- PostgreSQL: Sistema de gestión de bases de datos relacional (RDBMS) de código abierto, reconocido por su estabilidad, escalabilidad y cumplimiento de estándares SQL.
- React: Biblioteca de JavaScript creada por Meta que permite construir interfaces de usuario interactivas mediante componentes reutilizables.
- RUP (Rational Unified Process): Metodología de desarrollo de software iterativa e incremental, organizada en fases de Inicio, Elaboración, Construcción y Transición, utilizada en este proyecto para asegurar trazabilidad y control de calidad.
- Sequelize: ORM (Object-Relational Mapping) para Node.js que facilita la interacción con bases de datos SQL, permitiendo definir modelos y relaciones mediante objetos JavaScript.
- Tailwind CSS: Framework de diseño “utility-first” que permite construir interfaces web mediante clases predefinidas, agilizando el proceso de diseño y personalización visual.
- TIC (Tecnologías de la Información y Comunicación): Conjunto de tecnologías digitales que posibilitan el acceso, procesamiento, almacenamiento y transmisión de información, aplicadas aquí para modernizar la gestión institucional.
- UML (Unified Modeling Language): Lenguaje de modelado visual utilizado para representar gráficamente los distintos aspectos del sistema, como casos de uso, clases, actividades y secuencias.
- Usuario: Persona que interactúa con el sistema, ya sea en calidad de administrador, comandante, jefe de área o bombero operativo.
- Vite: Herramienta moderna de compilación para aplicaciones web que acelera la construcción, recarga y depuración del código en tiempo real.

II.1.2.6.4.1 Glosario de los Diagramas

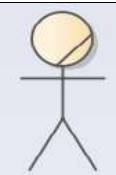
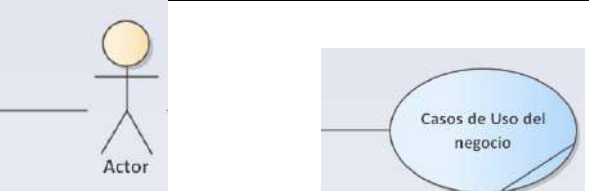
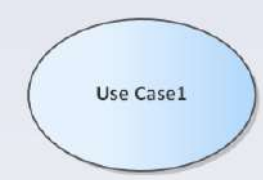
Actor del Negocio	
Casos de Uso Negocio	
Comunicación	
Relación	
Actor	
Casos de Uso	
Relación de Inclusión	
Relación de Extensión	

Tabla 4. Glosario de Diagramas
Fuente: Elaboración Propia.

II.1.2.6.5 Requisitos del Sistema

Los requisitos del sistema fueron definidos a partir del análisis funcional y técnico del proyecto “Mejoramiento de la gestión operativa para la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia-La Paz a través de las TIC”.

Su identificación se realizó durante las fases de Inicio y Elaboración de la metodología RUP, en base al levantamiento de información con los responsables de las áreas operativas y administrativas de la institución.

Cada requisito fue clasificado conforme a la norma IEEE 830, diferenciando entre:

- Requisitos Funcionales (RF): especifican las funciones y procesos que el sistema debe realizar.
- Requisitos No Funcionales (RNF): establecen las condiciones de calidad, desempeño y restricciones técnicas del sistema.

A continuación, se detallan ambos grupos de requisitos que sustentan el desarrollo del sistema web de gestión operativa.

II.1.2.6.5.1 Requisitos Funcionales

Los requisitos funcionales se agrupan según los 10 casos de uso principales definidos en el modelado UML del sistema. Cada grupo representa un módulo funcional desarrollado dentro de la plataforma web.

II.1.2.6.5.1.1 Modulo Autenticación

- RF0.1 El sistema debe permitir a los usuarios iniciar sesión ingresando su nombre de usuario y contraseña, los cuales son validados contra la base de datos PostgreSQL con contraseñas cifradas mediante bcrypt.
- RF0.2 Solo los usuarios autenticados pueden acceder al sistema. El backend genera un token JWT (JSON Web Token) firmado digitalmente, que define el acceso según el rol y los permisos asociados.
- RF0.3 El sistema debe validar los datos de entrada y notificar en caso de error (usuario inexistente, contraseña incorrecta o token expirado).
- RF0.4 Los intentos de acceso exitosos y fallidos deben registrarse en un log de auditoría.
- RF0.5 El sistema deberá permitir el bloqueo temporal del usuario después de tres intentos fallidos consecutivos de inicio de sesión (funcionalidad a implementar en middleware de autenticación).

II.1.2.6.5.1.2 Administración de Usuarios

- RF1.1 El sistema permitirá el alta, baja y modificación de usuarios.
- RF1.2 El administrador podrá asignar uno o varios roles a cada usuario.
- RF1.3 El sistema debe mostrar un listado filtrable y exportable de los usuarios registrados, con opciones de búsqueda por nombre, rol o estado.

- RF1.4 Se validará la unicidad de nombre de usuario, correo electrónico y formato de datos antes de guardar.
- RF1.5 El sistema permitirá desbloquear usuarios bloqueados por intentos fallidos.

II.1.2.6.5.1.3 Administración de Roles

- RF2.1 El sistema permitirá la creación, edición y eliminación de roles institucionales.
- RF2.2 Los roles podrán tener asignados permisos específicos por módulo (emergencias, inventario, donativos, credenciales, etc.).
- RF2.3 El sistema permitirá visualizar los permisos asignados a cada rol y validará que los roles en uso no puedan eliminarse.

II.1.2.6.5.1.4 Gestión de Vehículos

- RF3.1 El sistema permitirá registrar y editar información de los vehículos (marca, modelo, tipo, estado, matrícula).
- RF3.2 Se podrá asignar vehículos a emergencias activas, controlando su disponibilidad.
- RF3.3 Se permitirá registrar mantenimientos preventivos y correctivos, asociados a cada vehículo.
- RF3.4 El sistema permitirá la asignación de ítems del inventario a vehículos, registrando cantidad y fecha.
- RF3.5 Los vehículos asignados a emergencias cambiarán automáticamente su estado a “En emergencia” y se liberarán al finalizar el evento.

II.1.2.6.5.1.5 Gestión de Personal

- RF4.1 El sistema permitirá registrar, modificar y eliminar información del personal operativo y voluntario.
- RF4.2 Se podrá asignar grados, clases y roles institucionales al personal registrado.
- RF4.3 El sistema permitirá visualizar la disponibilidad actual del personal para emergencias, según su estado.
- RF4.4 Se podrán generar reportes de desempeño o participación en emergencias, capacitaciones y actividades.
- RF4.5 El sistema vinculará automáticamente al personal con sus credenciales activas o vencidas, manteniendo historial.

II.1.2.6.5.1.6 Gestión de Capacitaciones

- RF5.1 El sistema permitirá registrar cursos, fechas, responsables y contenidos.
- RF5.2 Se podrán inscribir participantes internos y externos.
- RF5.3 El sistema generará certificados en formato PDF personalizados con nombre, curso y fecha.
- RF5.4 Se podrá visualizar estadísticas de participación por curso o rango de fechas.
- RF5.5 El sistema conservará un historial de capacitaciones y certificados emitidos.

II.1.2.6.5.1.7 Gestión de Donativos

- RF6.1 El sistema permitirá registrar donativos materiales o monetarios, con información del donante, fecha, tipo y cantidad.
- RF6.2 Los donativos podrán asignarse o distribuirse a emergencias o vehículos, registrando su uso en inventario.

- RF6.3 Se podrán generar reportes detallados de entradas, salidas y distribución de recursos donados.
- RF6.4 El sistema permitirá consultar estadísticas de donativos por tipo, fuente o periodo.

II.1.2.6.5.1.8 Gestión de Emergencias

- RF7.1 El sistema permitirá registrar emergencias con tipo, ubicación, fecha y descripción.
- RF7.2 Se permitirá la asignación de recursos (vehículos, personal, inventario) a cada emergencia.
- RF7.3 El sistema deberá mantener actualizados los estados de emergencia.
- RF7.4 Se generarán informes operativos y reportes automáticos de recursos utilizados.
- RF7.5 El sistema registrará la finalización de la emergencia y liberará los recursos asociados.
- RF7.6 Las vistas de reportes permitirán análisis estadístico por tipo y frecuencia.

II.1.2.6.5.1.9 Gestión de Inventario

- RF8.1 El sistema permitirá registrar materiales, equipos y herramientas institucionales.
- RF8.2 Se llevará un control de stock mediante las operaciones de entrada, salida y consumo.
- RF8.3 El sistema deberá mostrar alertas cuando el stock alcance niveles críticos.
- RF8.4 Se permitirá registrar el uso y mantenimiento de equipos, asegurando que las cantidades no sean negativas.
- RF8.5 Se podrá asignar inventario a cada vehículo o emergencia, registrando cantidad y fecha.
- RF8.6 El sistema integrará los movimientos de inventario con módulos de donativos y emergencias.

II.1.2.6.5.1.10 Gestión de Credenciales

- RF9.1 El sistema permitirá generar credenciales institucionales con código QR único.
- RF9.2 Al escanear el QR, se mostrará una página pública de verificación con los datos verificables del bombero.
- RF9.3 El sistema permitirá actualizar, suspender o revocar credenciales según la situación del personal.
- RF9.4 El sistema actualizará automáticamente el estado de las credenciales al vencimiento.
- RF9.5 Se podrán generar reportes de credenciales emitidas, vigentes, suspendidas, revocadas o vencidas.

II.1.2.6.5.2 Requisitos No Funcionales

Los requisitos no funcionales definen las propiedades de calidad, rendimiento, seguridad, fiabilidad y mantenimiento que el sistema debe cumplir para garantizar su funcionamiento continuo, eficiente y seguro dentro del entorno institucional de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz.

A diferencia de los requisitos funcionales, que especifican lo que el sistema hace, estos requisitos establecen cómo debe hacerlo, considerando aspectos de desempeño, infraestructura, escalabilidad y cumplimiento normativo.

II.1.2.6.5.2.1 Requisitos de Rendimiento

- El sistema deberá permitir al menos 20 usuarios conectados simultáneamente sin degradación perceptible del rendimiento general ni pérdida de estabilidad en el servidor.
- Las operaciones críticas, como el registro de emergencias, la asignación de vehículos y personal, deberán ejecutarse en un tiempo máximo de 2 segundos en condiciones normales de red.
- Las consultas sobre reportes (emergencias, inventario, donativos, personal, credenciales) no deberán superar los 5 segundos de tiempo de respuesta, incluso al aplicar filtros combinados por fecha, tipo o ubicación.
- Las operaciones de carga masiva (como el ingreso inicial de inventario o personal) deberán completarse en menos de 30 segundos por lote de 100 registros.
- El sistema debe poder procesar al menos 1.000 transacciones diarias distribuidas entre los diferentes módulos, sin errores ni pérdida de rendimiento.
- La generación de documentos PDF (credenciales y certificados) deberá completarse en menos de 10 segundos por archivo, incluyendo la creación del código QR y la carga de plantillas.
- El backend optimizará las consultas mediante pool de conexiones PostgreSQL y consultas parametrizadas para garantizar tiempos de respuesta predecibles.

II.1.2.6.5.2.2 Requisitos de Seguridad

- El acceso al sistema se realizará mediante usuario y contraseña cifrados con bcrypt, junto con autenticación mediante JSON Web Tokens (JWT).
- Toda comunicación entre cliente y servidor deberá realizarse bajo protocolo HTTPS, garantizando la confidencialidad e integridad de los datos transmitidos.
- Se permitirá un máximo de tres intentos fallidos de inicio de sesión por usuario; tras el tercer intento, la cuenta será bloqueada temporalmente.
- El sistema contará con niveles de acceso diferenciados por rol y permisos jerárquicos, gestionados mediante *middlewares* de autenticación.
- Todas las operaciones deberán quedar registradas en un log de auditoría, incluyendo acción, usuario, módulo, fecha y hora.
- La base de datos implementará restricciones de integridad referencial, claves foráneas, índices y *triggers* para garantizar consistencia.
- Se realizarán copias de seguridad automáticas y procedimientos de recuperación ante fallos.

- Se verificará la integridad de los registros críticos (emergencias, credenciales, inventario).
- El sistema deberá cumplir con la legislación boliviana vigente en protección de datos personales, garantizando la privacidad del personal operativo, donantes y participantes.

II.1.2.6.5.2.3 Requisitos de Fiabilidad

- Las transacciones completadas (registro de emergencias, asignación de personal, generación de credenciales o reportes) deberán tener una tasa de error menor al 0.1%.
- En caso de error durante una transacción, esta no deberá considerarse finalizada y se notificará al usuario con un mensaje de error.
- El sistema deberá mantener una disponibilidad operativa (MTBF) mínima de 30 días bajo condiciones normales de uso.
- El backend deberá soportar concurrencia de múltiples usuarios mediante operaciones asincrónicas controladas, evitando duplicación de datos.
- En caso de interrupción del servicio, la base de datos deberá mantener su integridad y recuperarse al estado previo a la caída mediante *journaling* y transacciones ACID.

II.1.2.6.5.2.4 Requisitos de Disponibilidad

- El sistema deberá estar disponible las 24 horas del día, los 7 días de la semana (24/7), accesible desde cualquier navegador moderno.
- Se deberá garantizar una disponibilidad mínima del 99.5% mensual, considerando su uso durante emergencias en tiempo real.
- Aunque no se implemente en la primera versión, deberá diseñarse un mecanismo de recuperación ante desastres, que permita restaurar la aplicación en caso de fallos críticos del servidor o pérdida de conectividad.
- Las tareas de mantenimiento programado deberán notificarse al administrador con al menos 24 horas de anticipación, evitando interferencias con operaciones institucionales.

II.1.2.6.5.2.5 Requisitos de Mantenibilidad

- El sistema deberá contar con documentación técnica completa, incluyendo código fuente comentado, diagramas UML y especificación de la API REST.
- Las tareas de mantenimiento correctivo serán responsabilidad del desarrollador y deberán ejecutarse antes, durante y después de eventos relevantes (simulacros o emergencias reales).
- Se recomienda realizar mantenimientos preventivos mensuales, revisando logs de errores, dependencias y rendimiento del servidor.
- Las mejoras funcionales deberán planificarse con base en la retroalimentación del personal clave (Comandancia y Jefaturas).
- La arquitectura modular del backend y la separación por capas (rutas, controladores, modelos y servicios) facilitarán la actualización sin afectar otros módulos.

II.1.2.6.5.2 Requisitos de Portabilidad

- El sistema deberá ser independiente del sistema operativo, accesible desde cualquier dispositivo con navegador moderno (Chrome, Firefox, Edge o Safari).
- Todo el código deberá desarrollarse con tecnologías multiplataforma y de código abierto: Node.js (backend), React + Vite (frontend) y PostgreSQL (base de datos).
- El backend deberá ser compatible con entornos virtualizados o contenedores Docker, permitiendo despliegue en desarrollo, pruebas o producción.
- Al menos el 90% del código deberá ser independiente del servidor físico, facilitando migraciones sin reescritura.
- La configuración del entorno deberá gestionarse mediante variables de entorno (.env), simplificando la instalación y replicación del sistema.

II.1.2.6.5.3 Matriz de Trazabilidad de Requisitos Funcionales

La matriz de trazabilidad permite relacionar los requisitos funcionales del sistema con los módulos desarrollados y los actores que los ejecutan. Esta relación asegura la coherencia entre el análisis, diseño e implementación, garantizando que cada requisito esté correctamente cubierto por un módulo del sistema y asignado a un actor responsable.

Asimismo, constituye un instrumento de control y verificación para las fases de validación y pruebas bajo la metodología RUP.

Código RF	Nombre del Requisito	Módulo Asociado	Actor(es) Relacionados	Descripción / Trazabilidad
RF0.1 – RF0.5	Autenticación y control de acceso	Módulo de Autenticación	Todos los usuarios (comandante, Jefes de Área)	Permite inicio de sesión, validación de credenciales y gestión de accesos mediante JWT y bcrypt.
RF1.1 – RF1.5	Administración de usuarios	Módulo de Usuarios	Comandante	Permite crear, modificar, eliminar y gestionar usuarios con roles asignados.
RF2.1 – RF2.3	Administración de roles	Módulo de Roles	Comandante	Gestiona permisos y jerarquías institucionales, determinando accesos en todos los módulos.
RF3.1 – RF3.5	Gestión de vehículos	Módulo de Vehículos	Comandante, Jefe de Operaciones, Jefe de Logística	Registra, mantiene y asigna vehículos a emergencias; controla disponibilidad.
RF4.1 – RF4.5	Gestión de personal	Módulo de Personal	Comandante, Jefe de Instrucción, Jefe de Operaciones	Permite registrar y administrar la información del personal operativo, jerarquías y estado.
RF5.1 – RF5.5	Gestión de capacitaciones	Módulo de Capacitaciones	Jefe de Instrucción	Permite registrar cursos, participantes y generar certificados digitales.

RF6.1 – RF6.4	Gestión de donativos	Módulo de Donativos	Comandante, Jefe de Logística	Permite registrar donaciones materiales o monetarias y generar reportes consolidados.
RF7.1 – RF7.6	Gestión de emergencias	Módulo de Emergencias	Comandante, Jefe de Operaciones, Jefe de Logística	Registra emergencias, asigna recursos humanos y materiales, genera reportes y estadísticas.
RF8.1 – RF8.6	Gestión de inventario	Módulo de Inventario	Jefe de Logística	Administra entradas y salidas de equipos, controla existencias y enlaza con donativos y emergencias.
RF9.1 – RF9.5	Gestión de credenciales	Módulo de Credenciales	Comandante, Jefe de Instrucción	Genera credenciales digitales verificables mediante código QR, con control de validez.

*Tabla 5. Matriz de Trazabilidad
Fuente: Elaboración Propia.*

II.1.2.6.5.4 Análisis de la Matriz

- Cada requisito funcional (RF) se encuentra trazado a uno o más módulos del sistema, garantizando cobertura completa.
- Los actores identificados en el diagrama de casos de uso general (Comandante, Jefes de Instrucción, Operaciones y Logística) mantienen interacción directa con los módulos asignados.
- Los requisitos RF0.1–RF0.5 son transversales a todos los actores, al representar el acceso al sistema y la seguridad inicial.
- Esta trazabilidad permitirá, en la etapa de verificación y validación, confirmar que las funciones implementadas cumplen con las especificaciones definidas en la norma IEEE 830 y con los objetivos del proyecto.

II.1.2.6.6 Modelo de Casos de Uso del Negocio

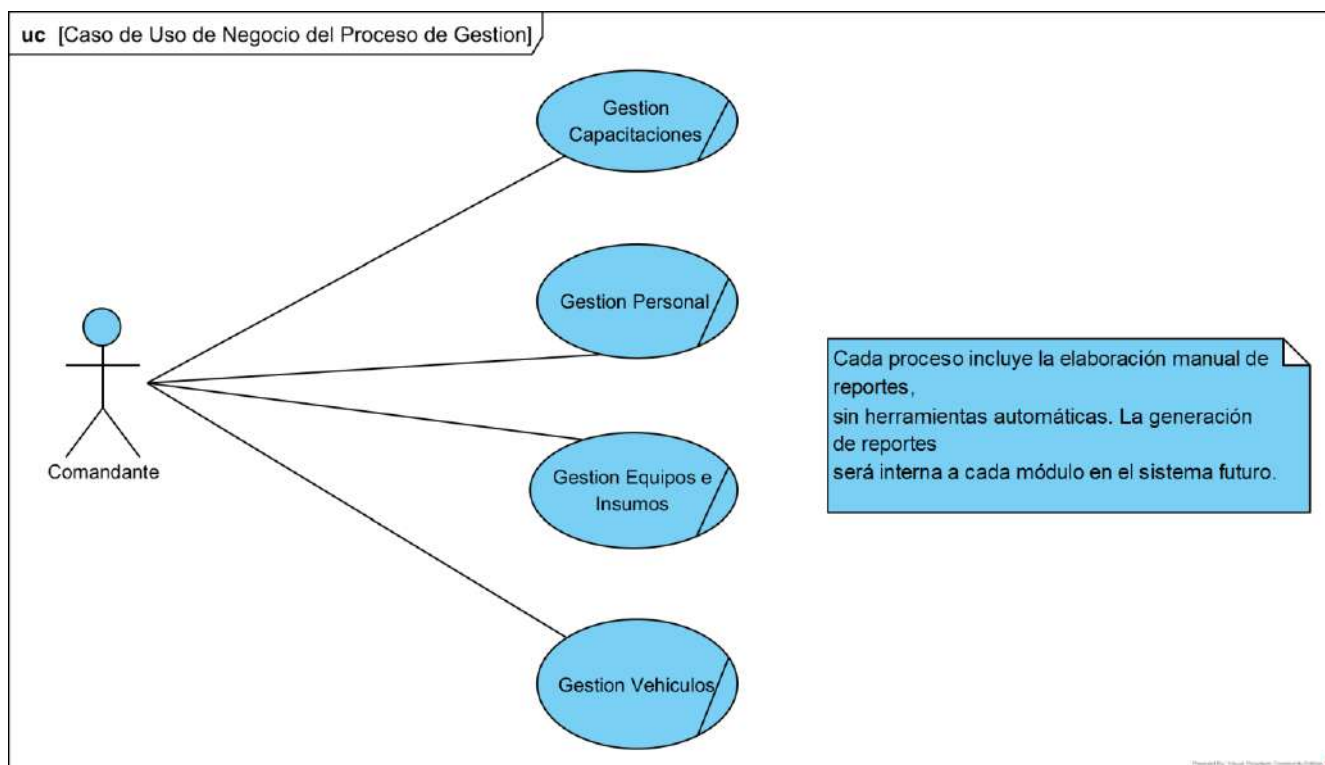


Figura 1. Caso de uso de Negocio Proceso de Gestión
Fuente: Elaboración Propria

II.1.2.6.7 Modelo de Caso de Uso

II.1.2.6.7.1 Diagrama General de Caso de Uso

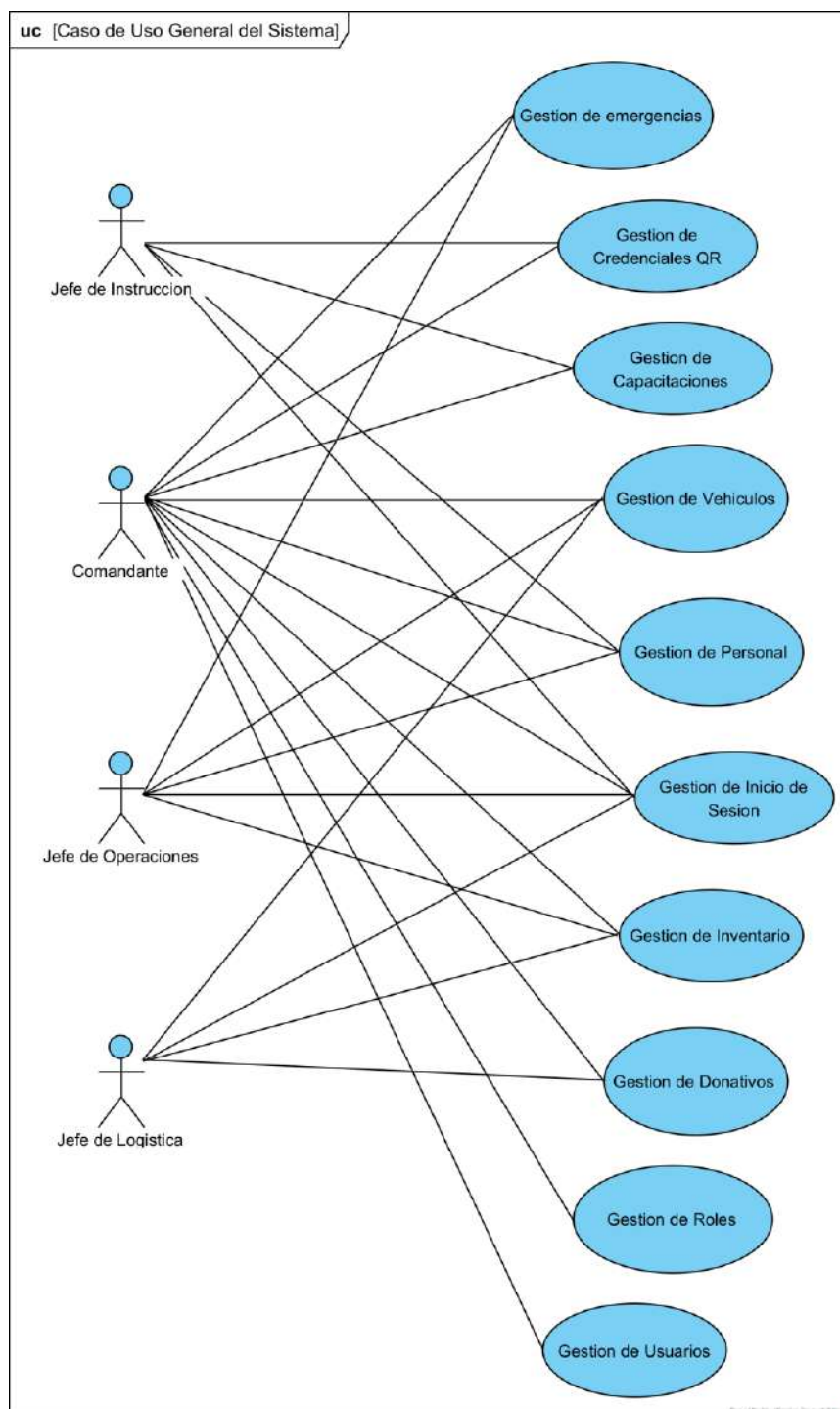


Figura 2. Diagrama de Casos de Uso General Del Sistema
Fuente: Elaboración Propia.

II.1.2.6.7.2 Diagrama de Generalización de los actores

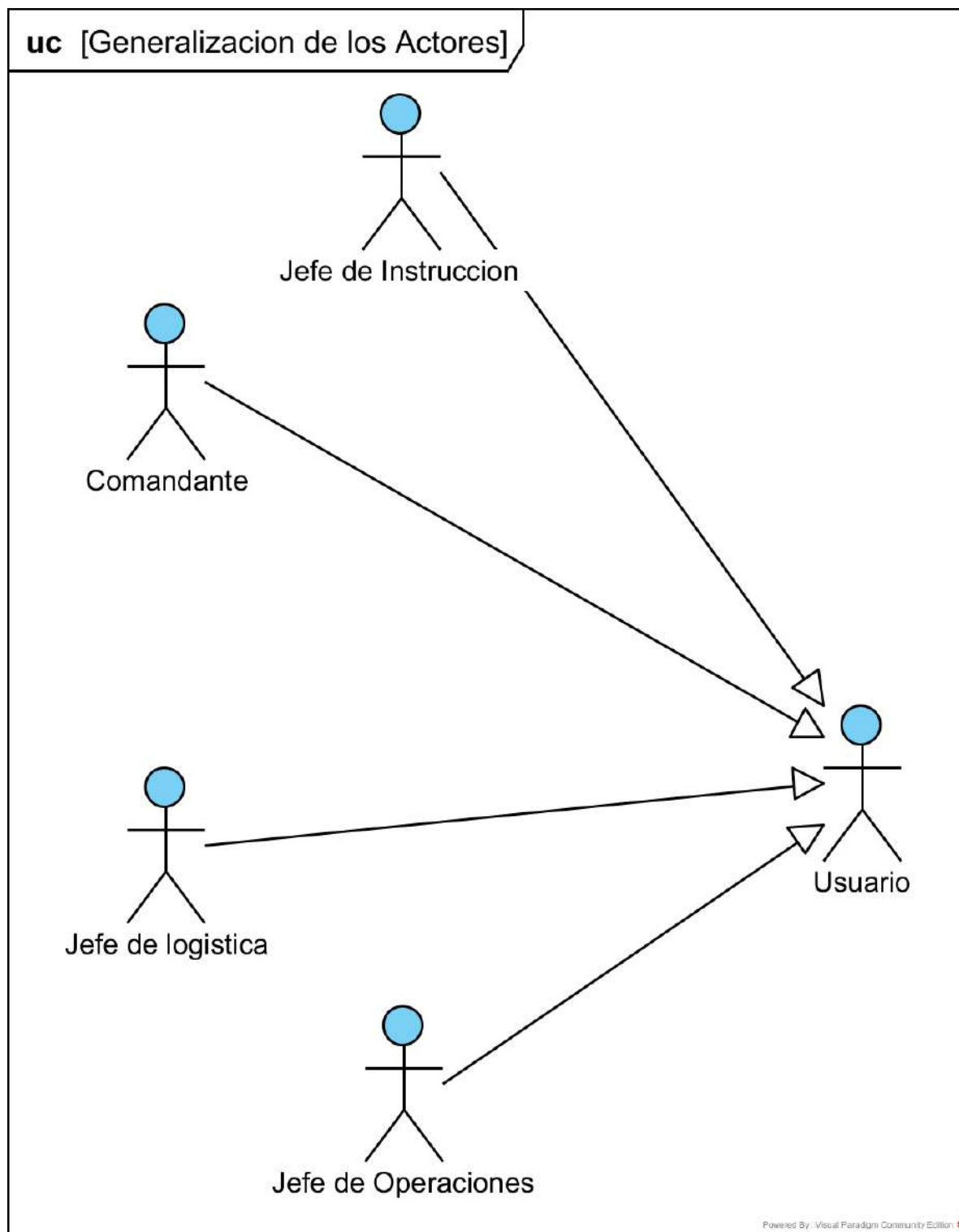


Figura 3. Diagrama de los Actores Involucrados.
Fuente: Elaboración Propia.

II.1.2.6.8 Diagrama de Casos de Uso Explosionados

II.1.2.6.8.1 Diagrama de Caso de uso: Gestionar Inicio de Sesión

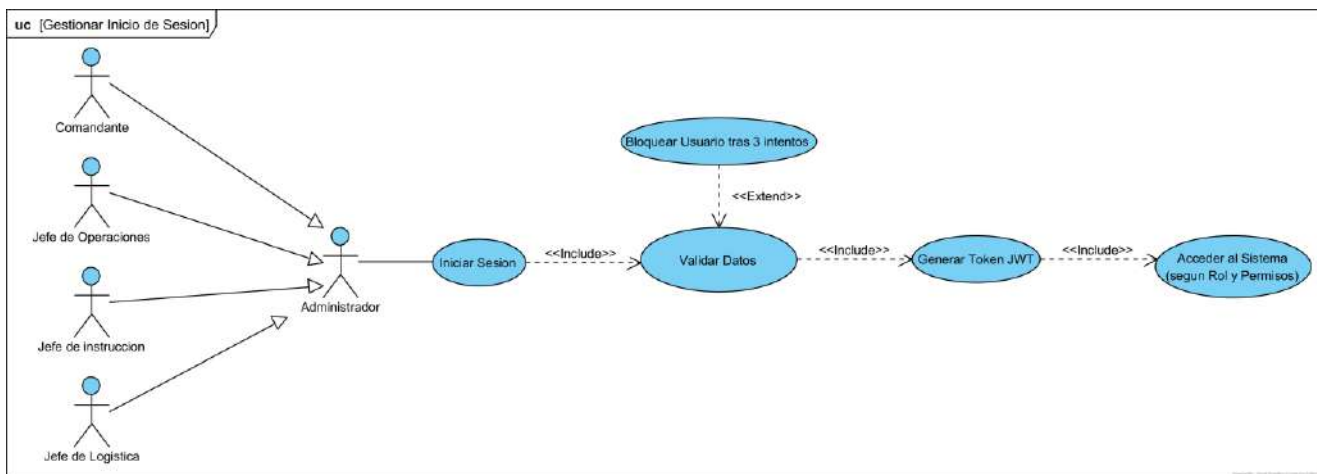


Figura 4. Diagrama de Casos de Uso: Gestionar Inicio de Sesión
Fuente: Elaboración Propia.

II.1.2.6.8.2 Diagrama de Caso de uso: Gestión de Emergencias

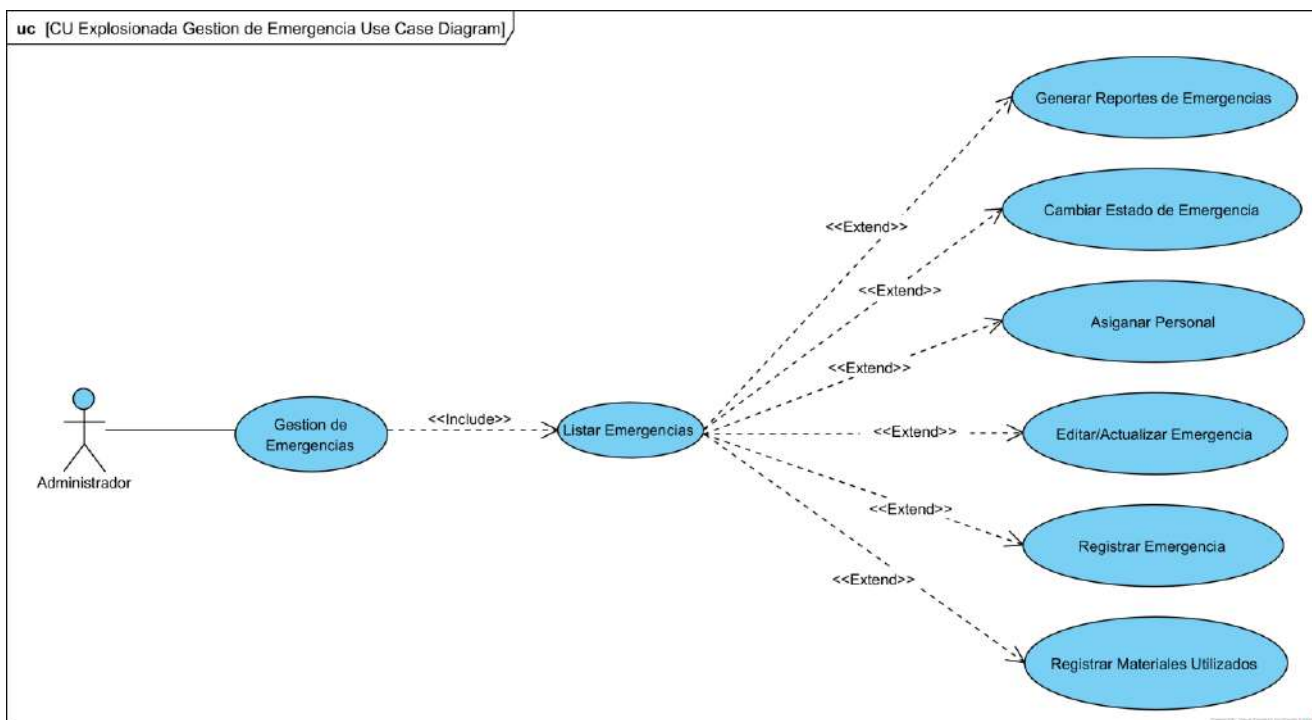


Figura 5. Diagrama de Casos de Uso: Gestión de Emergencias
Fuente: Elaboración Propia.

II.1.2.6.8.3 Diagrama de Caso de uso: Gestión de Credenciales QR

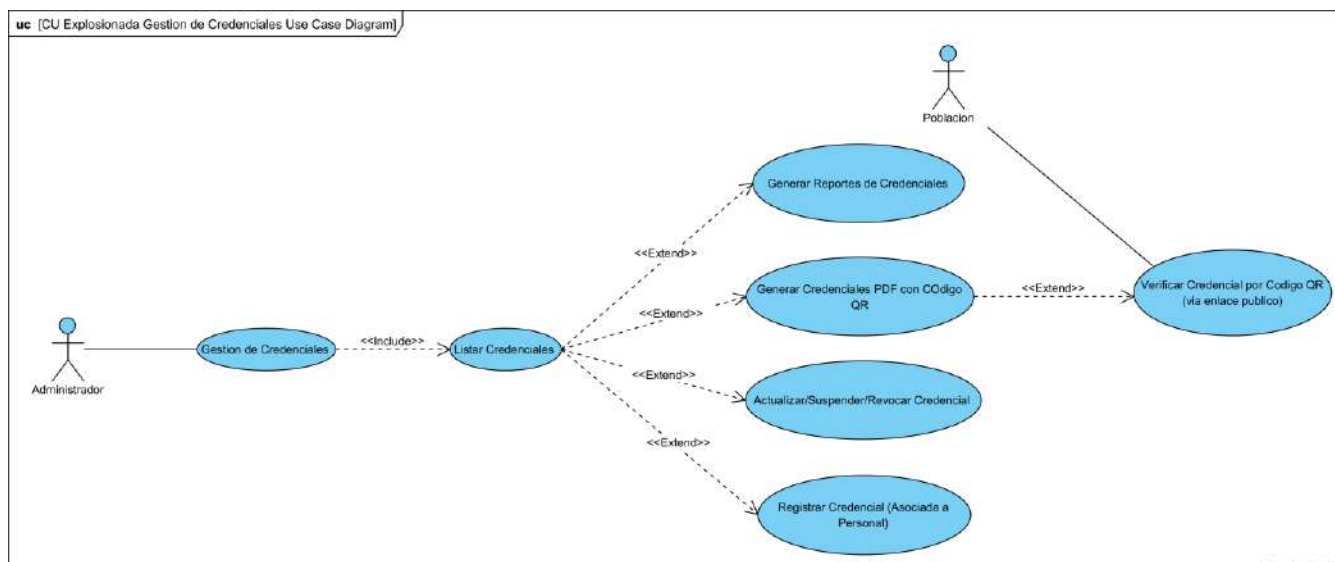


Figura 6. Diagrama de Casos de Uso: Gestión de Credenciales QR
Fuente: Elaboración Propia.

II.1.2.6.8.4 Diagrama de Caso de uso: Gestión de Capacitaciones

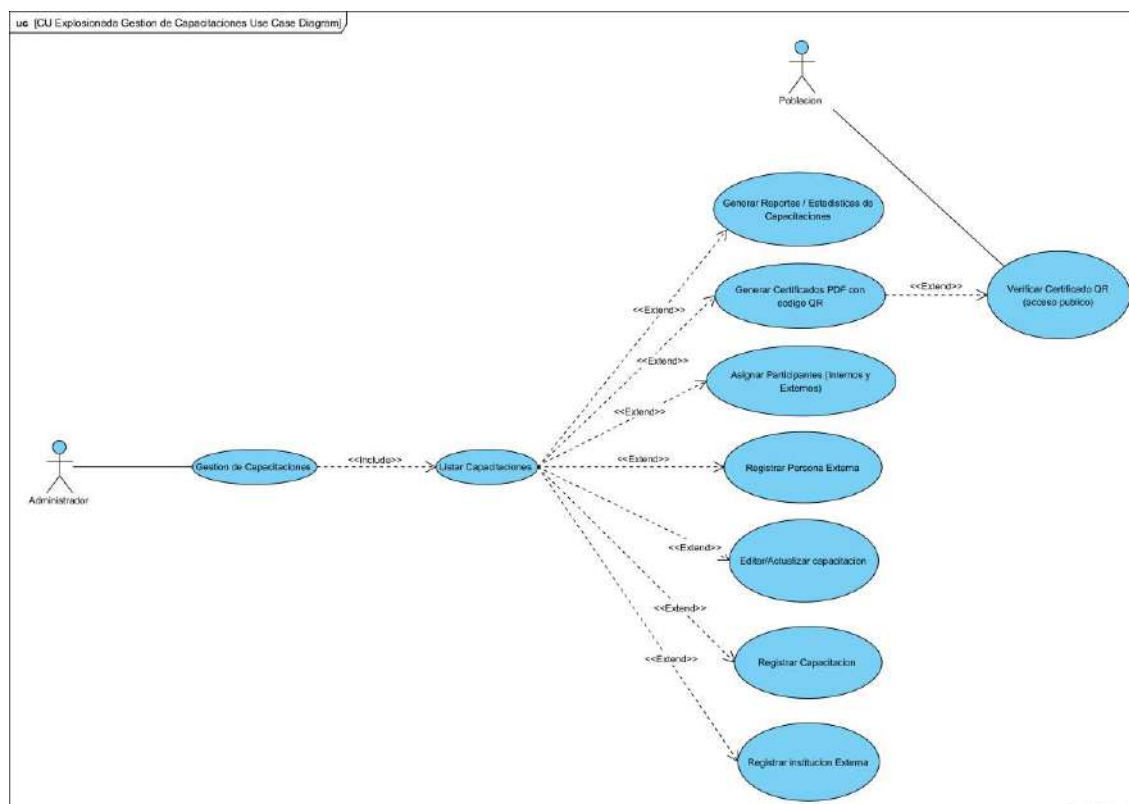


Figura 7. Diagrama de Casos de Uso: Gestión de Capacitaciones
Fuente: Elaboración Propia.

II.1.2.6.8.5 Diagrama de Caso de uso: Gestión de Vehículos

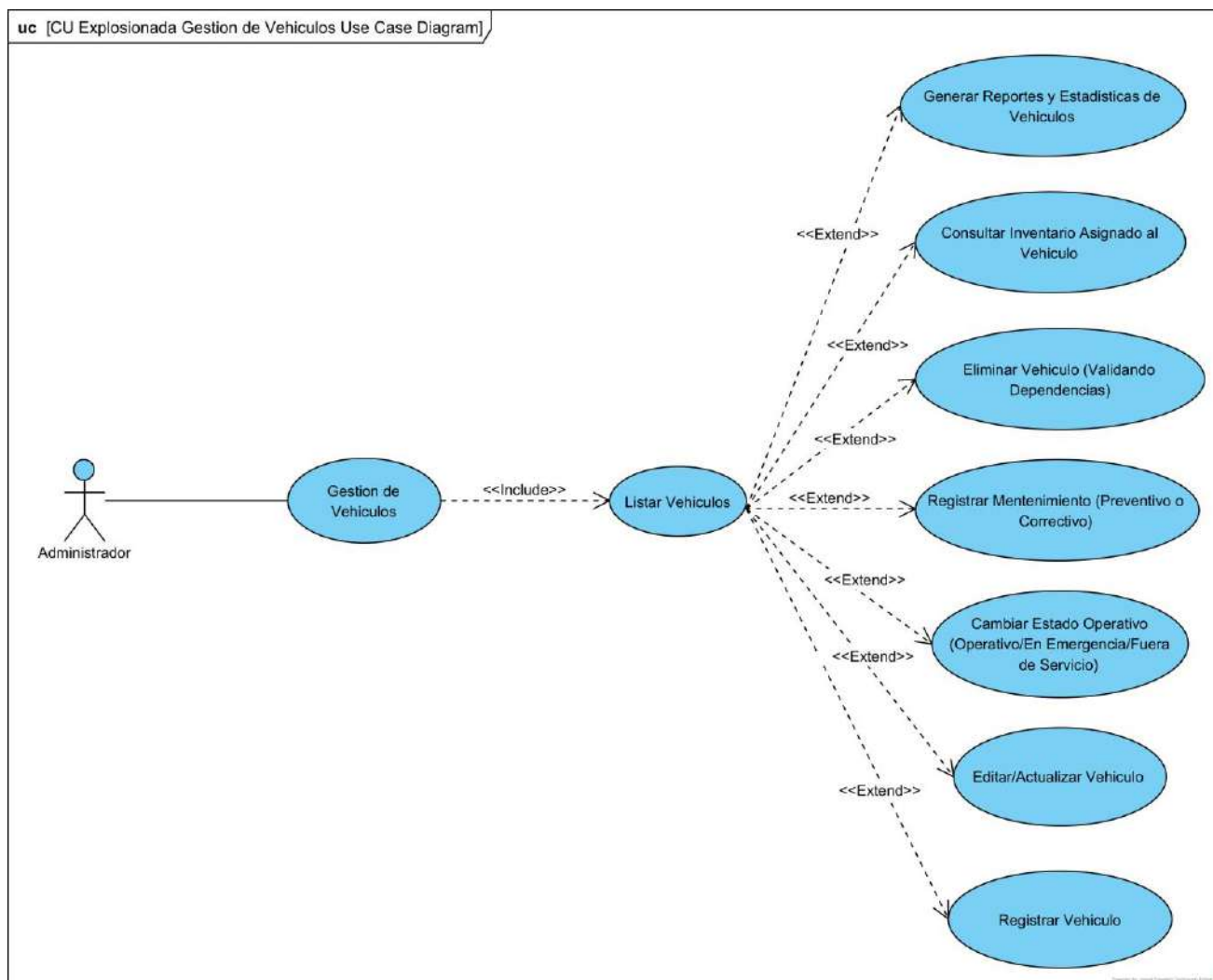


Figura 8. Diagrama de Casos de Uso: Gestión de Vehículos
Fuente: Elaboración Propia.

II.1.2.6.8.6 Diagrama de Caso de uso: Gestión de Personal

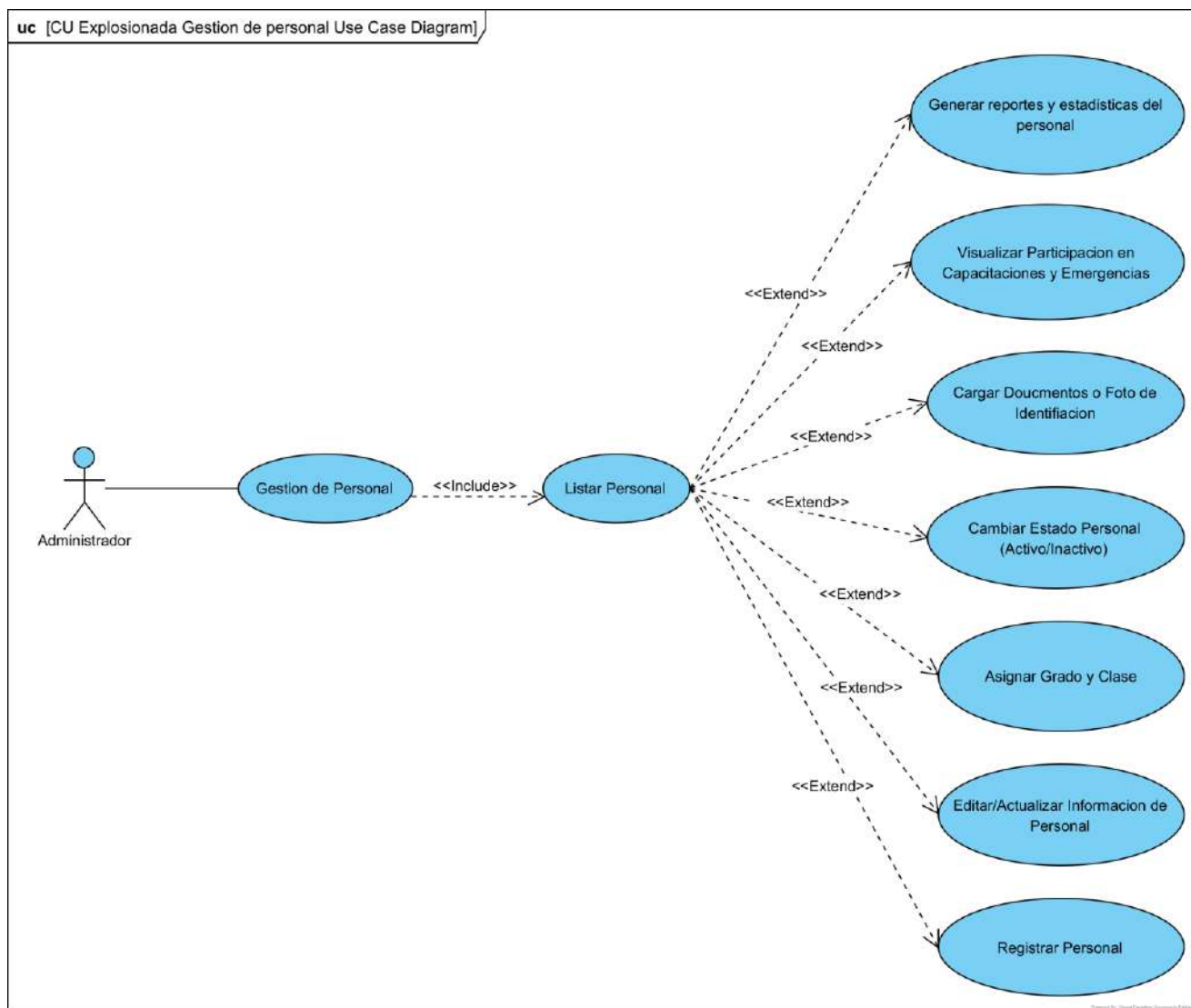


Figura 9. Diagrama de Casos de Uso: Gestión de Personal
Fuente: Elaboración Propia.

II.1.2.6.8.7 Diagrama de Caso de uso: Gestión de Inventario

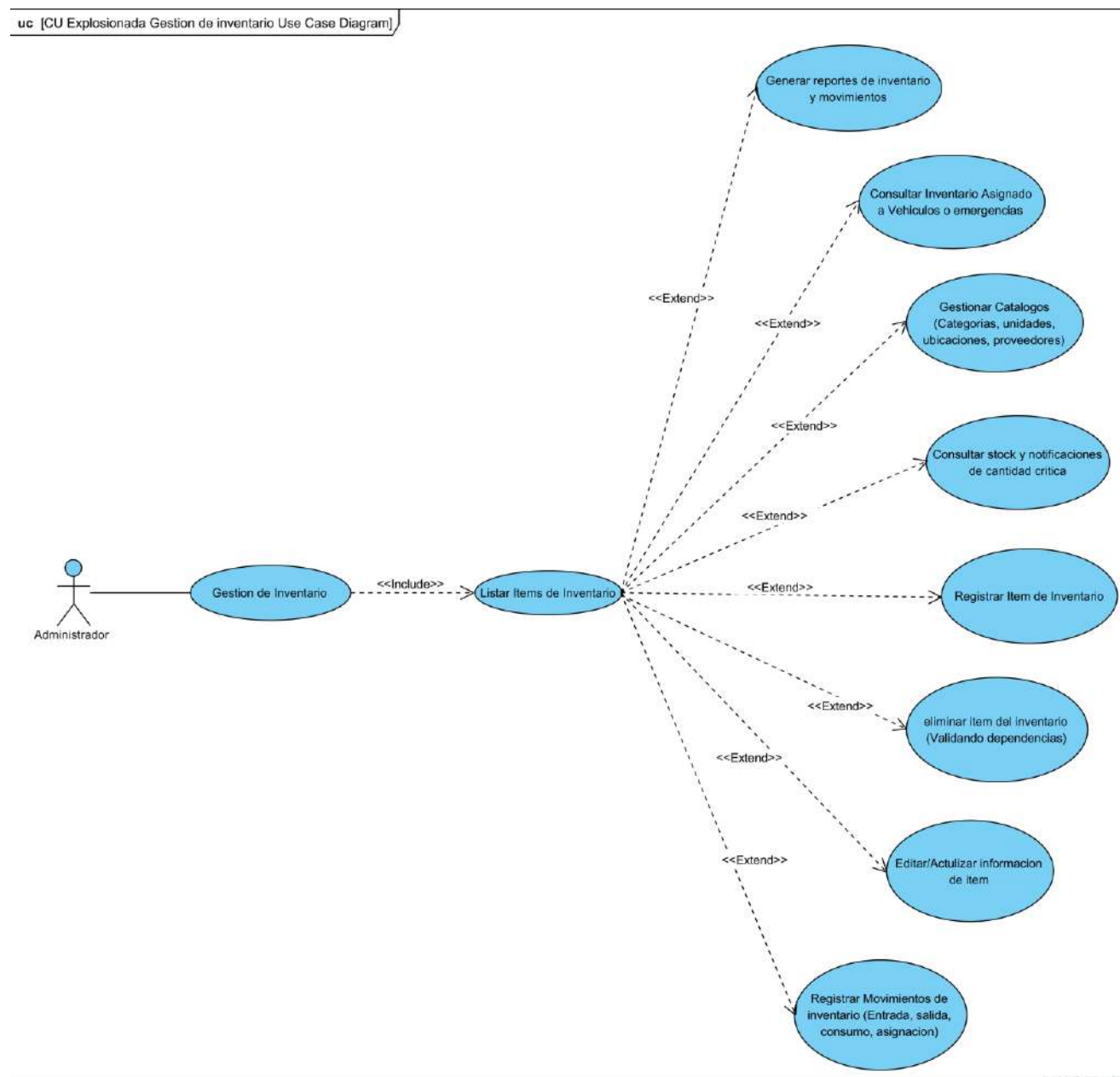


Figura 10. Diagrama de Casos de Uso: Gestión de Inventario
Fuente: Elaboración Propia.

II.1.2.6.8.8 Diagrama de Caso de uso: Gestión de Donativos

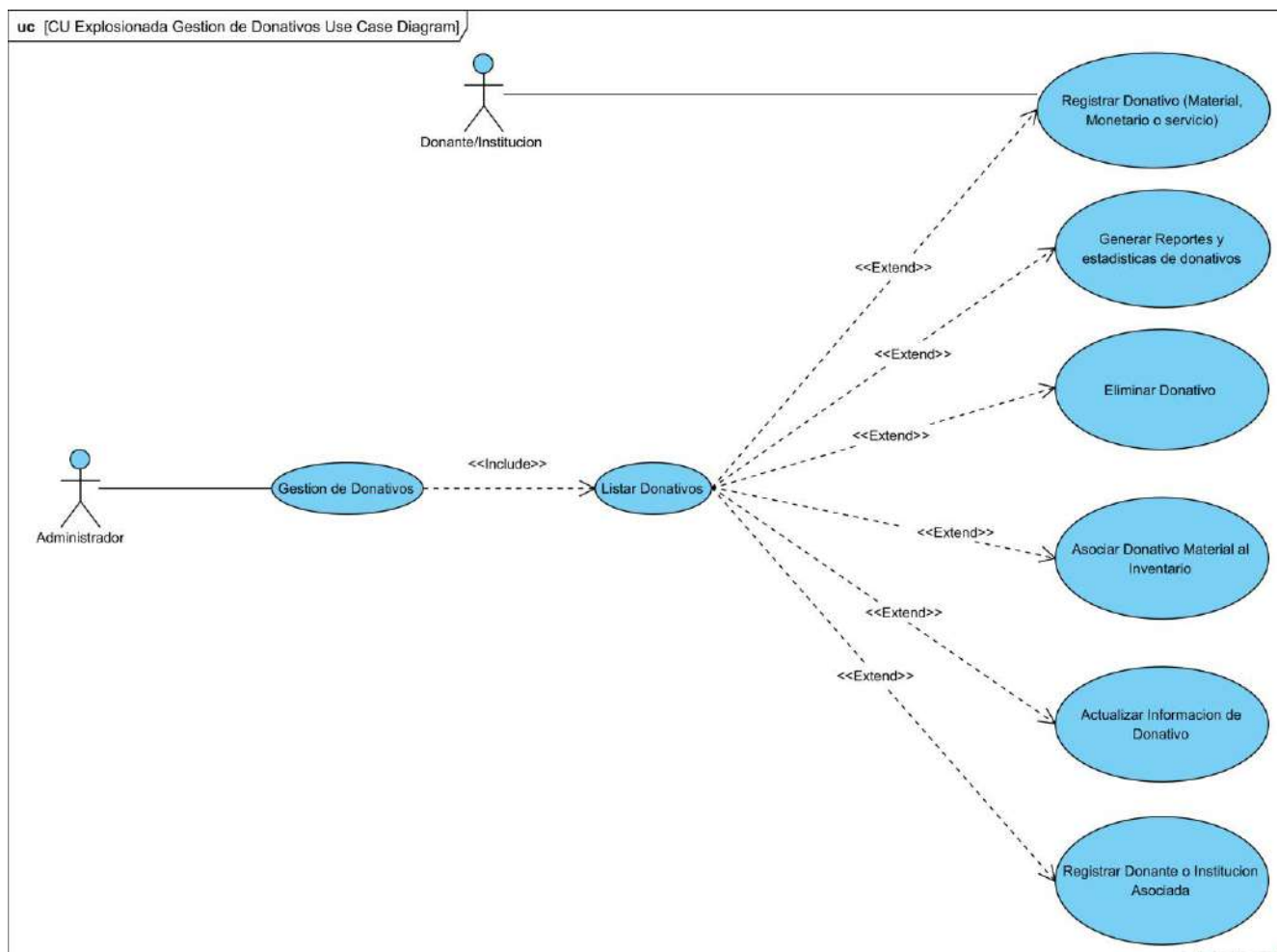


Figura 11. Diagrama de Casos de Uso: Gestión de Donativos
Fuente: Elaboración Propia.

II.1.2.6.8.9 Diagrama de Caso de uso: Gestión de Roles

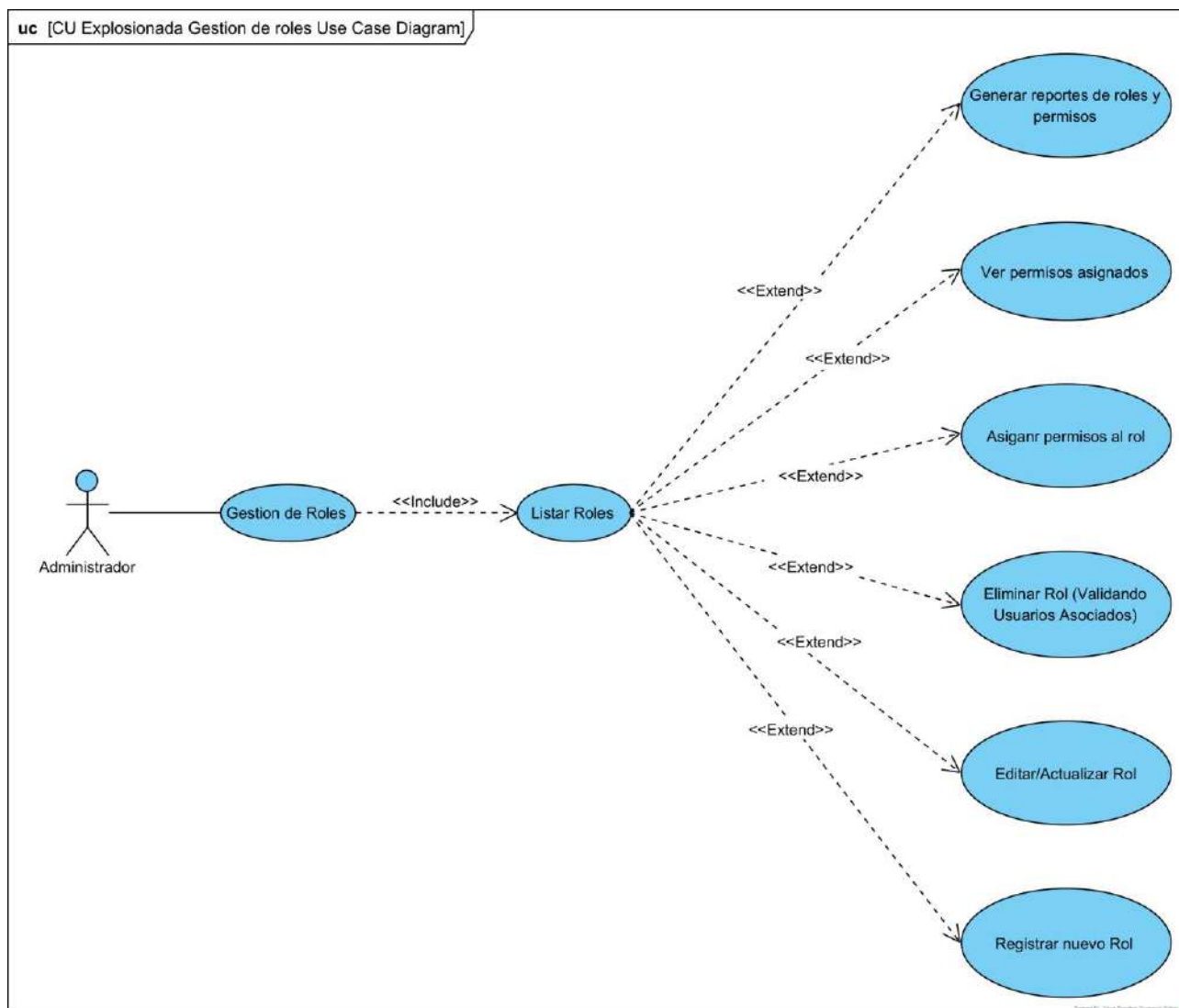


Figura 12. Diagrama de Casos de Uso: Gestión de Roles
Fuente: Elaboración Propia.

II.1.2.6.8.10 Diagrama de Caso de uso: Gestión de Usuarios

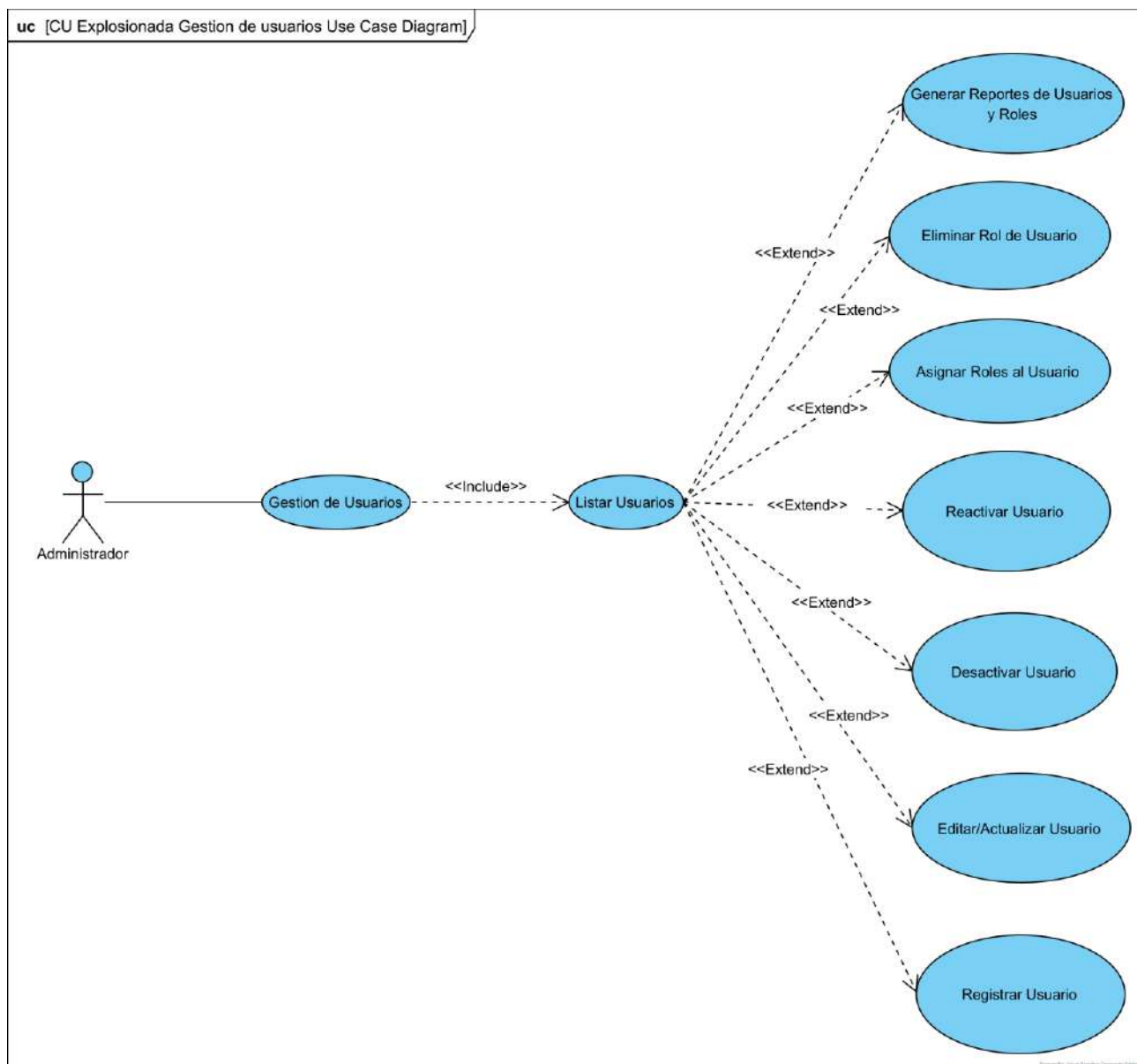


Figura 13. Diagrama de Casos de Uso: Gestión de Usuario
Fuente: Elaboración Propia.

II.1.2.6.9 Especificación de casos de Uso Explosionado

II.1.2.6.9.1 Especificación de Caso de Uso: Iniciar Sesión

- Iniciar Sesión

Caso de uso	Iniciar Sesión
Actores	Comandante, Jefe de Operaciones, Jefe de Instrucción, Jefe de Logística, Administrador
Tipo	Primario
Propósito	Permitir que los usuarios autorizados del sistema web inicien sesión de manera segura para acceder a las funcionalidades correspondientes a su rol.
Resumen	El sistema muestra la interfaz “P – LOGIN”, en la cual los actores ingresan sus credenciales de acceso (usuario y contraseña). El sistema valida las credenciales en la base de datos PostgreSQL y, si son correctas, genera un token JWT que permite acceder al sistema de acuerdo con los permisos asignados al rol del usuario.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El caso de uso “Iniciar Sesión” comienza cuando los actores acceden al sistema web.	2. El sistema despliega la interfaz “P – LOGIN” para el ingreso de credenciales.
3. Los actores ingresan los datos de “Usuario” y “Contraseña” en los campos correspondientes.	
4. Los actores presionan el botón “Iniciar Sesión”.	5. El sistema valida los datos ingresados realizando una consulta a las tablas “usuarios” y “roles” de la base de datos PostgreSQL.
	6A. Si las credenciales son correctas, el sistema genera un token JWT con los datos del usuario y su rol, limpiando los intentos fallidos previos.
	7A. El sistema despliega la interfaz principal según el rol del usuario (por ejemplo, comandante, Logística,

	Operaciones, Instrucción o Administrador).
Curso Alternativo de Eventos	
6B. Si los datos ingresados son incorrectos, el sistema registra el intento fallido en la tabla "intentos_login", muestra el mensaje "Usuario o contraseña incorrectos. Intente nuevamente." y regresa al paso 3.	
6C. Si el usuario alcanza tres intentos fallidos consecutivos, el sistema bloquea temporalmente la cuenta y muestra el mensaje "Usuario bloqueado por múltiples intentos fallidos.", regresando al paso 2.	
6B. Si el usuario no está registrado, el sistema muestra el mensaje "Usuario no encontrado en el sistema."	

Tabla 6. Especificación de Caso de Uso: Iniciar Sesión
Fuente: Elaboración Propia.

II.1.2.6.9.2 Especificación de Caso de Uso: Gestionar de Usuarios

- **Gestión de Usuarios**

Caso de uso	Gestionar Usuarios
Actores	Comandante, Administrador
Tipo	Primario
Propósito	Permitir la administración integral de las cuentas de usuario del sistema SAR, abarcando el registro, edición, activación, asignación de roles y control de accesos.
Resumen	Este caso de uso describe las funciones necesarias para la gestión de usuarios dentro del sistema. Desde la interfaz "P – USUARIOS", los actores pueden listar usuarios, crear nuevas cuentas, editar información, cambiar estados, asignar roles o generar reportes. El acceso a cada acción depende del rol del usuario autenticado.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede al sistema y selecciona el módulo "Gestión de Usuarios" desde el menú principal.	2. El sistema valida los permisos del actor y despliega la interfaz "P – USUARIOS".
3. El actor visualiza la lista general de usuarios registrados.	4. El sistema permite acceder a las funciones extendidas: registrar, editar,

	activar/desactivar, asignar roles o generar reportes.
5. El actor selecciona una función específica (por ejemplo, Registrar Usuario).	6. El sistema redirige al flujo correspondiente del subcaso de uso.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes para acceder al módulo, el sistema muestra “Acceso denegado. Permisos insuficientes.”	
A2. Si ocurre un error de conexión o falla del servidor, se muestra “Error al cargar la información de usuarios, intente nuevamente.”	

Tabla 7. Especificación de Caso de Uso: Gestionar Usuarios
Fuente: Elaboración Propia.

- **Listar Usuarios**

Caso de uso	Listar Usuarios
Actores	Comandante, Administrador
Tipo	Incluido
Propósito	Visualizar todos los usuarios registrados, con sus roles y estados.
Resumen	El sistema muestra la interfaz “P – USUARIOS”, donde los actores pueden consultar, filtrar y acceder a los registros de usuarios.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede a la opción “Listar Usuarios”.	2. El sistema consulta la base de datos y muestra la lista de usuarios.
3. El Actor aplica Filtros (Por rol, estado o nombre)	4. El sistema muestra los resultados filtrados
Curso Alternativo de Eventos	
A1. Si no existen usuarios registrados, se muestra “No se encontraron registros de usuario.”	

Tabla 8. Especificación de Caso de Uso: Gestionar Usuarios-Listar Usuarios
Fuente: Elaboración Propia.

- **Registrar Nuevo Usuario**

Caso de uso	Registrar Nuevo Usuario
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Crear una nueva cuenta de usuario en el sistema.
Resumen	Permite al Administrador ingresar datos de un nuevo usuario, asignarle un rol inicial y registrar la cuenta en la base de datos.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona "Registrar Usuario".	2. El sistema muestra la interfaz 6A. El sistema guarda el nuevo usuario en la base de datos y muestra "Usuario registrado correctamente."
3. El actor ingresa los datos del usuario (nombre, usuario, contraseña, correo).	4. El sistema valida los campos obligatorios.
5. El actor confirma el registro.	6A. El sistema guarda el nuevo usuario en la base de datos y muestra "Usuario registrado correctamente."
Curso Alternativo de Eventos	
A1. Si el usuario ya existe, se muestra "El nombre de usuario ingresado ya está en uso."	
A2. Si faltan datos requeridos, se muestra "Complete todos los campos obligatorios."	

Tabla 9. Especificación de Caso de Uso: Gestionar Usuarios-Registrar Nuevo Usuario
Fuente: Elaboración Propia.

- **Editar / Actualizar Usuario**

Caso de uso	Editar / Actualizar Usuario
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Modificar datos de una cuenta existente.
Resumen	Permite actualizar información de usuario, como correo, contraseña o rol, desde la interfaz de edición.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un usuario y presiona "Editar".	2. El sistema muestra la interfaz con los datos actuales.
3. El actor modifica la información necesaria.	4A. El sistema guarda los cambios y muestra "Datos actualizados correctamente."

Curso Alternativo de Eventos
A1. Si se intenta modificar un usuario inactivo, el sistema muestra “No se puede editar un usuario desactivado.”

Tabla 10. Especificación de Caso de Uso: Gestionar Usuarios-Editar / Actualizar Usuario
Fuente: Elaboración Propia.

- **Desactivar / Reactivar Usuario**

Caso de uso	Desactivar / Reactivar Usuario
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Cambiar el estado del usuario sin eliminarlo.
Resumen	El Administrador puede suspender temporalmente una cuenta o reactivarla.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un usuario.	2. El sistema muestra el estado actual (Activo/Inactivo).
3. El actor confirma la acción.	4A. El sistema actualiza el estado y muestra “Usuario actualizado correctamente.”
Curso Alternativo de Eventos	
A1. Si el actor cancela la acción, el sistema retorna a la lista de usuarios	

Tabla 11. Especificación de Caso de Uso: Gestionar Usuarios-Desactivar / Reactivar Usuario
Fuente: Elaboración Propia.

- **Registrar Asignar / Eliminar Rol de Usuario**

Caso de uso	Asignar / Eliminar Rol de Usuario
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Gestionar los roles de cada usuario para definir permisos y accesos.
Resumen	Desde la lista de usuarios, los actores pueden asignar o eliminar roles, afectando los permisos del usuario.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un usuario y abre el módulo de roles.	2. El sistema muestra los roles disponibles.
3. El actor selecciona o elimina roles.	4A. El sistema actualiza la relación en la tabla usuarioRol y muestra “Roles actualizados correctamente.”
Curso Alternativo de Eventos	

A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado.”

Tabla 12. Especificación de Caso de Uso: Gestionar Usuarios-Asignar / Eliminar Rol de Usuario
Fuente: Elaboración Propia.

- **Generar Reportes de Usuarios**

Caso de uso	Generar Reportes de Usuarios
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Generar reportes estadísticos sobre roles, estado o permisos de los usuarios.
Resumen	Desde la interfaz “P – USUARIOS”, los actores pueden generar y exportar reportes en formato PDF filtrando por criterios específicos.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona la opción “Generar Reporte”.	2. El sistema solicita los parámetros del reporte
3. El actor confirma la acción.	4A. El sistema genera el reporte y lo muestra o exporta en formato PDF.
Curso Alternativo de Eventos	
A1. Si no hay resultados, el sistema muestra “No se encontraron datos para el reporte seleccionado.”	

Tabla 13. Especificación de Caso de Uso: Gestionar Usuarios-Generar Reportes de Usuarios
Fuente: Elaboración Propia.

II.1.2.6.9.3 Especificación de Caso de Uso: Gestionar Roles

- **Gestión de Roles**

Caso de uso	Gestionar Roles
Actores	Comandante, Administrador
Tipo	Primario
Propósito	Permitir la administración de los roles del sistema SAR, definiendo los niveles de acceso, permisos y jerarquías de los usuarios.
Resumen	Este caso de uso describe las acciones necesarias para mantener los roles del sistema, su creación, edición, eliminación y asignación de permisos. Desde la interfaz “P – ROLES”, los actores pueden visualizar la lista de roles y acceder a las funciones extendidas de mantenimiento o generación de reportes.

Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “ Gestión de Roles ” desde el menú principal.	2. El sistema valida los permisos del actor y muestra la interfaz “ P – ROLES ” con la lista de roles existentes.
3. El actor selecciona una función (registrar, editar, eliminar, asignar permisos o generar reportes).	4. El sistema redirige al flujo correspondiente del subcaso de uso.
5. El actor ejecuta las acciones correspondientes.	6A. El sistema confirma las operaciones realizadas y actualiza los datos mostrados.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado. Permisos insuficientes.”	
A2. Si ocurre un error de conexión, el sistema muestra “Error al cargar la información de roles, intente nuevamente.”	

Tabla 14. Especificación de Caso de Uso: Gestionar Roles
Fuente: Elaboración Propia.

- **Listar de Roles**

Caso de uso	Listar Roles
Actores	Comandante, Administrador
Tipo	Incluido
Propósito	Permitir la visualización de todos los roles registrados en el sistema junto con sus permisos asociados.
Resumen	El sistema muestra la interfaz “P – ROLES” con la lista de roles registrados, mostrando nombre, descripción y cantidad de permisos.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede a la opción “Listar Roles”.	2. El sistema consulta la base de datos y muestra la lista de roles.
3. El actor puede aplicar filtros o seleccionar un rol específico.	4. El sistema muestra los roles filtrados o la información detallada.
Curso Alternativo de Eventos	
A1. Si no existen registros, el sistema muestra “No se encontraron roles registrados.”	

Tabla 15. Especificación de Caso de Uso: Gestionar Roles-Listar Roles
Fuente: Elaboración Propia.

- **Registrar nuevo Rol**

Caso de uso	Registrar Nuevo Rol
Actores	Comandante, Administrador

Tipo	Extendido
Propósito	Permitir al Administrador crear un nuevo rol institucional en el sistema.
Resumen	Desde la interfaz “P – REGISTRAR ROL”, el Administrador ingresa el nombre y descripción del nuevo rol, que luego podrá tener permisos asignados.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “ Registrar Rol ”.	2. El sistema muestra la interfaz “P – REGISTRAR ROL”.
3. El actor ingresa el nombre y descripción del nuevo rol.	4A. El sistema valida los campos y registra el nuevo rol, mostrando “Rol registrado correctamente.”
Curso Alternativo de Eventos	
A1. Si el rol ya existe, el sistema muestra “El nombre de rol ya está registrado.”	
A2. Si faltan datos, el sistema muestra “Complete todos los campos requeridos.”	

Tabla 16. Especificación de Caso de Uso: Gestionar Roles-Registrar Nuevo Rol
Fuente: Elaboración Propia.

- **Editar / Actualizar Rol**

Caso de uso	Editar / Actualizar Rol
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Modificar la información de un rol existente.
Resumen	Desde la lista de roles, el Administrador puede modificar el nombre o descripción de un rol.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un rol existente.	2. El sistema muestra los datos actuales del rol.
3. El actor edita los campos necesarios.	4A. El sistema guarda los cambios y muestra “Rol actualizado correctamente.”
Curso Alternativo de Eventos	
A1. Si ocurre un error de validación, el sistema muestra “Error al actualizar los datos, intente nuevamente.”	

Tabla 17. Especificación de Caso de Uso: Gestionar Roles-Editar / Actualizar Rol
Fuente: Elaboración Propia.

- **Eliminar Rol**

Caso de uso	Eliminar Rol
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Permitir eliminar un rol que no tenga usuarios asociados.
Resumen	El Administrador puede eliminar un rol siempre que no esté asignado a ningún usuario.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un rol y presiona "Eliminar".	2. El sistema valida si existen usuarios asociados.
3A. Si no hay dependencias, elimina el rol.	4A. El sistema muestra "Rol eliminado correctamente."
Curso Alternativo de Eventos	
A1. Si el rol tiene usuarios asociados, el sistema muestra "No se puede eliminar un rol con usuarios vinculados."	

Tabla 18. Especificación de Caso de Uso: Gestionar Roles-Eliminar Rol
Fuente: Elaboración Propia.

- **Asignar Permisos a Rol**

Caso de uso	Asignar Permisos a Rol
Actores	Administrador, comandante
Tipo	Extendido
Propósito	Asociar uno o más permisos específicos a un rol.
Resumen	Permite asignar permisos desde una lista de funcionalidades del sistema (ver, crear, editar, eliminar).
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un rol y elige "Permisos".	2. El sistema muestra la lista de permisos disponibles.
3. El actor selecciona los permisos correspondientes.	4A. El sistema guarda la relación rol-permisos y muestra "Permisos asignados correctamente."
Curso Alternativo de Eventos	
A1. Si el actor no tiene permiso para modificar roles, se muestra "Acceso denegado."	

Tabla 19. Especificación de Caso de Uso: Gestionar Roles-Asignar Permisos a Rol
Fuente: Elaboración Propia.

- **Ver Permisos Asignados**

Caso de uso	Ver Permisos Asignado
Actores	Comandante, administrador
Tipo	Incluido
Propósito	Visualizar los permisos vinculados a un rol específico.
Resumen	Permite consultar los permisos asignados a un rol para auditar el acceso de los usuarios.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un rol.	2. El sistema muestra los permisos vinculados a ese rol.
Curso Alternativo de Eventos	
A1. Si el rol no tiene permisos asignados, se muestra “No existen permisos asociados a este rol.”	

Tabla 20. Especificación de Caso de Uso: Gestionar Roles-Ver Permisos Asignados
Fuente: Elaboración Propia.

- **Generar Reportes de Roles y Permiso**

Caso de uso	Generar Reportes de Roles y Permisos
Actores	Comandante, Administrador
Tipo	Extendido
Propósito	Generar reportes que muestren la relación entre roles, usuarios y permisos.
Resumen	El actor puede generar reportes de roles sin usuarios, roles con múltiples permisos o usuarios sin rol asignado.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Generar Reporte”.	2. El sistema solicita parámetros de filtrado.
3. El actor confirma.	4A. El sistema genera y muestra el reporte solicitado.
Curso Alternativo de Eventos	
A1. Si no hay datos disponibles, el sistema muestra “No se encontraron registros para el reporte solicitado.”	

Tabla 21. Especificación de Caso de Uso: Gestionar Roles-Generar Reportes de Roles Y Permiso
Fuente: Elaboración Propia.

II.1.2.6.9.4 Especificación de Caso de Uso: Gestionar Vehículos

- **Gestión de Vehículos**

Caso de uso	Gestionar Vehículos
Actores	Comandante, Jefe de Operaciones, Jefe de Logística
Tipo	Primario
Propósito	Permitir la gestión completa del parque vehicular del sistema SAR, incluyendo el registro, actualización, control de estado, mantenimiento y generación de reportes.
Resumen	Este caso de uso describe las acciones necesarias para administrar los vehículos de la institución. Desde la interfaz “P – VEHÍCULOS”, los actores pueden listar los registros existentes, registrar nuevos, modificar datos, cambiar estado operativo, registrar mantenimientos o generar reportes.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “Gestión de Vehículos” desde el menú principal.	2. El sistema valida los permisos y despliega la interfaz “P – VEHÍCULOS” mostrando el listado general.
3. El actor visualiza los vehículos registrados.	4. El sistema permite acceder a las opciones extendidas: registrar, editar, cambiar estado, registrar mantenimiento o generar reportes.
5. El actor selecciona una función.	6A. El sistema ejecuta la acción correspondiente y actualiza la vista.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado.”	
A2. Si ocurre un error de conexión o de consulta, se muestra “No se pudo cargar la información de vehículos, intente nuevamente.”	

Tabla 22. Especificación de Caso de Uso: Gestionar Vehículos
Fuente: Elaboración Propia.

- **Listar Vehículos**

Caso de uso	Listar Vehículos
Actores	Comandante, Jefe de Operaciones, Jefe de Logística
Tipo	Incluido
Propósito	Mostrar todos los vehículos registrados en el sistema con su información básica y estado operativo.

Resumen	El sistema muestra la interfaz “P – VEHÍCULOS”, donde los actores pueden filtrar por tipo, estado o disponibilidad, y acceder a las funciones extendidas.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede al módulo de vehículos.	2. El sistema consulta la base de datos y muestra la lista de vehículos.
3. El actor aplica filtros o selecciona un registro.	4. El sistema actualiza la lista según los criterios aplicados.
Curso Alternativo de Eventos	
A1. Si no existen registros, el sistema muestra “No se encontraron vehículos registrados.”	

Tabla 23. Especificación de Caso de Uso: Gestionar Vehículos-Listar Vehículos
Fuente: Elaboración Propia.

- **Registrar Vehículos**

Caso de uso	Registrar Vehículo
Actores	Jefe de Logística, Jefe de Operaciones
Tipo	Extendido
Propósito	Registrar un nuevo vehículo en el sistema, con su información técnica y estado inicial.
Resumen	Permite ingresar los datos del vehículo (placa, tipo, marca, modelo, estado, año, foto) y guardarlos en la base de datos.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Registrar Vehículo”.	2. El sistema muestra la interfaz “P – REGISTRAR VEHÍCULO”.
3. El actor ingresa los datos requeridos.	4A. El sistema valida los campos y guarda el registro en la base de datos, mostrando “Vehículo registrado correctamente.”
Curso Alternativo de Eventos	
A1. Si la placa ya existe, el sistema muestra “La placa ingresada ya está registrada.”	
A2. Si faltan datos requeridos, el sistema muestra “Complete todos los campos obligatorios.”	

Tabla 24. Especificación de Caso de Uso: Gestionar Vehículos-Registrar Vehículos
Fuente: Elaboración Propia.

- **Editar / Actualizar Vehículos**

Caso de uso	Editar / Actualizar Vehículo
Actores	Jefe de Logística, Jefe de Operaciones
Tipo	Extendido
Propósito	Modificar los datos existentes de un vehículo registrado.
Resumen	Desde la lista de vehículos, el actor puede modificar la información del vehículo seleccionado, como tipo, modelo, estado o foto.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un vehículo y presiona "Editar".	2. El sistema muestra los datos actuales del vehículo.
3. El actor modifica los campos necesarios.	4A. El sistema actualiza el registro y muestra "Vehículo actualizado correctamente."
Curso Alternativo de Eventos	
A1. Si el vehículo está asignado a una emergencia activa, el sistema muestra "No se puede modificar un vehículo en uso."	

Tabla 25. Especificación de Caso de Uso: Gestionar Vehículos-Editar / Actualizar Vehículos
Fuente: Elaboración Propia.

- **Cambiar Estado Operativo**

Caso de uso	Cambiar Estado Operativo
Actores	Jefe de Logística, Jefe de Operaciones
Tipo	Extendido
Propósito	Modificar el estado operativo del vehículo (Operativo / En emergencia / Fuera de servicio).
Resumen	Desde la lista, el actor puede cambiar el estado del vehículo según su disponibilidad o condición técnica.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un vehículo.	2. El sistema muestra el estado actual.
3. El actor selecciona el nuevo estado.	4A. El sistema actualiza el estado y muestra "Estado del vehículo actualizado correctamente."
Curso Alternativo de Eventos	
A1. Si el vehículo está asignado a otra operación, el sistema muestra "El vehículo no puede cambiar de estado mientras esté asignado."	

Tabla 26. Especificación de Caso de Uso: Gestionar Vehículos-Cambiar Estado Operativo
Fuente: Elaboración Propia.

- **Registrar Mantenimiento**

Caso de uso	Registrar Mantenimiento
Actores	Jefe de Logística
Tipo	Extendido
Propósito	Registrar mantenimientos preventivos o correctivos realizados al vehículo.
Resumen	El actor puede agregar un mantenimiento con fecha, descripción, tipo y adjuntar un documento o comprobante.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un vehículo y presiona "Registrar Mantenimiento".	2. El sistema muestra el formulario de mantenimiento.
3. El actor ingresa los datos requeridos.	4A. El sistema guarda el registro y muestra "Mantenimiento registrado correctamente."
Curso Alternativo de Eventos	
A1. Si falta información, el sistema muestra "Complete todos los campos requeridos."	

Tabla 27. Especificación de Caso de Uso: Gestionar Vehículos-Registrar Mantenimiento
Fuente: Elaboración Propia.

- **Eliminar Vehículo**

Caso de uso	Eliminar Vehículo
Actores	Jefe de Logística
Tipo	Extendido
Propósito	Eliminar un vehículo del registro general.
Resumen	Permite eliminar un vehículo siempre que no tenga asignaciones o mantenimientos asociados.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un vehículo y presiona "Eliminar".	2. El sistema verifica dependencias (mantenimientos o emergencias)
3A. Si no existen dependencias, elimina el registro.	4A. El sistema muestra "Vehículo eliminado correctamente."
Curso Alternativo de Eventos	
A1. Si existen registros asociados, el sistema muestra "No se puede eliminar un vehículo con dependencias activas."	

Tabla 28. Especificación de Caso de Uso: Gestionar Vehículos-Eliminar Vehículos
Fuente: Elaboración Propia.

- **Generar Reportes de Vehículo**

Caso de uso	Generar Reportes de Vehículos
Actores	Comandante, Jefe de Logística, Jefe de Operaciones
Tipo	Extendido
Propósito	Generar reportes de estado, mantenimiento y disponibilidad de los vehículos.
Resumen	Permite generar reportes filtrando por tipo, estado o periodo, y exportarlos en formato PDF.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona "Generar Reporte".	2. El sistema solicita los parámetros del reporte.
3. El actor confirma.	4A. El sistema genera el reporte solicitado y lo presenta en pantalla.
Curso Alternativo de Eventos	
A1. Si no hay datos para el filtro aplicado, el sistema muestra "No se encontraron resultados para el reporte solicitado."	

Tabla 29. Especificación de Caso de Uso: Gestionar Vehículos-Generar Reportes de Vehículo
Fuente: Elaboración Propia.

II.1.2.6.9.5 Especificación de Caso de Uso: Gestionar Personal

- **Gestionar Personal**

Caso de uso	Gestionar Personal
Actores	Comandante, Jefe de Operaciones, Jefe de Instrucción
Tipo	Primario
Propósito	Permitir la administración integral del personal del sistema SAR, incluyendo su registro, edición, asignación de grados y clases, control de estado y generación de reportes.
Resumen	Este caso de uso describe las acciones que permiten gestionar la información del personal de la institución. Desde la interfaz "P – PERSONAL", los actores pueden listar los registros existentes, registrar nuevos miembros, editar su información, asignar jerarquía (grado y clase), cambiar el estado activo o inactivo, y generar reportes.
Curso Normal de Eventos	

Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “Gestión de Personal” desde el menú principal.	2. El sistema valida los permisos del actor y despliega la interfaz “P – PERSONAL” mostrando el listado general.
3. El actor visualiza la lista del personal registrado.	4. El sistema habilita las opciones para registrar, editar, asignar grado y clase, cambiar estado o generar reportes.
5. El actor selecciona una acción específica.	6A. El sistema redirige al flujo del sub caso de uso correspondiente y actualiza los datos.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado.”	
A2. Si ocurre un error de conexión, el sistema muestra “Error al cargar la información del personal.”	

Tabla 30. Especificación de Caso de Uso: Gestionar Personal
Fuente: Elaboración Propia.

- **Listar Personal**

Caso de uso	Listar Personal
Actores	Comandante, Jefe de Operaciones, Jefe de Instrucción
Tipo	Incluido
Propósito	Mostrar la lista del personal registrado en el sistema con sus datos institucionales.
Resumen	El sistema despliega la interfaz “P – PERSONAL”, mostrando el listado del personal, su grado, clase, estado y botones para ejecutar funciones extendidas.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede al módulo de personal.	2. El sistema consulta la base de datos y muestra la lista del personal.
3. El actor aplica filtros por grado, clase o estado.	4. El sistema muestra los resultados filtrados.
Curso Alternativo de Eventos	
A1. Si no hay registros, el sistema muestra “No se encontraron miembros registrados.”	

Tabla 31. Especificación de Caso de Uso: Gestionar Personal-Listar Personal
Fuente: Elaboración Propia.

- **Registrar Personal**

Caso de uso	Registrar Personal
Actores	Jefe de Operaciones, Jefe de Instrucción
Tipo	Extendido
Propósito	Registrar un nuevo miembro del personal institucional.
Resumen	Permite ingresar los datos de identificación, contacto, clase, grado, foto y documentos del nuevo integrante del personal SAR.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona "Registrar Personal".	2. El sistema muestra la interfaz "P – REGISTRAR PERSONAL".
3. El actor ingresa los datos requeridos.	4. El sistema valida los campos y guarda el registro en la base de datos.
	5A. El sistema muestra "Registro exitoso del personal."
Curso Alternativo de Eventos	
A1. Si falta información obligatoria, el sistema muestra "Complete todos los campos requeridos."	
A2. Si el documento de identidad ya está registrado, el sistema muestra "El personal ya existe en la base de datos."	

Tabla 32. Especificación de Caso de Uso: Gestionar Personal-Registrar Personal
Fuente: Elaboración Propia.

- **Editar / Actualizar Información de Personal**

Caso de uso	Editar / Actualizar Información de Personal
Actores	Jefe de Operaciones, Jefe de Instrucción
Tipo	Extendido
Propósito	Permitir modificar los datos personales o institucionales de un registro existente.
Resumen	Desde la lista de personal, el actor puede actualizar los datos personales, grado, clase o información de contacto.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un registro y presiona " Editar ".	2. El sistema muestra los datos actuales del personal.
3. El actor modifica los datos necesarios	4A. El sistema guarda los cambios y muestra "Información actualizada correctamente."
Curso Alternativo de Eventos	

A1. Si el personal está inactivo, el sistema muestra “No se puede editar un registro inactivo.”

Tabla 33. Especificación de Caso de Uso: Gestionar Personal-Editar / Actualizar Información de Personal
Fuente: Elaboración Propia.

- **Asignar Grado y Clase**

Caso de uso	Asignar Grado y Clase
Actores	Jefe de Operaciones, Jefe de Instrucción
Tipo	Extendido
Propósito	Asignar el grado jerárquico y la clase institucional a un miembro del personal.
Resumen	El actor selecciona el personal y asigna el grado y clase correspondiente desde los catálogos institucionales.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un miembro y presiona “Asignar Grado y Clase”.	2. El sistema muestra las listas de grados y clases disponibles.
3. El actor selecciona las opciones correspondientes.	4A. El sistema actualiza la información y muestra “Grado y clase asignados correctamente.”
Curso Alternativo de Eventos	
A1. Si los catálogos están vacíos, el sistema muestra “No existen registros de grados o clases disponibles.”	

Tabla 34. Especificación de Caso de Uso: Gestionar Personal-Asignar Grado y Clase
Fuente: Elaboración Propia.

- **Cambiar el Estado del Personal**

Caso de uso	Cambiar Estado del Personal
Actores	Jefe de Operaciones
Tipo	Extendido
Propósito	Cambiar el estado del personal entre activo o inactivo.
Resumen	Permite desactivar temporalmente un registro o reactivarlo según la situación operativa del miembro.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un registro.	2. El sistema muestra su estado actual.
3. El actor selecciona “Activar” o “Desactivar”.	4A. El sistema cambia el estado y muestra “Estado actualizado correctamente.”
Curso Alternativo de Eventos	

A1. Si el actor cancela la acción, el sistema regresa al listado sin cambios

Tabla 35. Especificación de Caso de Uso: Gestionar Personal-Cambiar el Estado del Personal
Fuente: Elaboración Propia.

- **Generar Reportes del Personal**

Caso de uso	Generar Reportes del Personal
Actores	Comandante, Jefe de Operaciones, Jefe de Instrucción
Tipo	Extendido
Propósito	Generar reportes estadísticos sobre la distribución y participación del personal.
Resumen	Permite generar reportes de personal por clase, grado, estado, capacitación y participación en emergencias.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona "Generar Reporte".	2. El sistema solicita parámetros del reporte (por grado, clase, estado o participación).
3. El actor confirma la acción.	4A. El sistema genera el reporte y lo muestra o exporta en formato PDF.
Curso Alternativo de Eventos	
A1. Si no existen datos para los filtros aplicados, el sistema muestra "No se encontraron resultados para el reporte seleccionado."	

Tabla 36. Especificación de Caso de Uso: Gestionar Personal-Generar Reportes del Personal
Fuente: Elaboración Propia.

II.1.2.6.9.6 Especificación de Caso de Uso: Gestionar Capacitaciones

- **Gestionar Capacitaciones**

Caso de uso	Gestionar Capacitaciones
Actores	Jefe de Instrucción, comandante, Administrador
Tipo	Primario
Propósito	Permitir la gestión completa de las capacitaciones impartidas o recibidas por la institución, incluyendo registro, actualización, asignación de participantes internos o externos, vinculación con instituciones, generación de certificados QR y reportes.
Resumen	Este caso de uso describe las acciones necesarias para gestionar los eventos de capacitación institucionales. Desde la

	interfaz “P – CAPACITACIONES”, los actores pueden listar capacitaciones, registrar nuevas, asignar participantes (internos o externos), registrar instituciones asociadas, generar certificados y emitir reportes de participación o cobertura.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “Gestión de Capacitaciones” desde el menú principal.	2. El sistema valida los permisos y muestra la interfaz “P – CAPACITACIONES”.
3. El actor visualiza las capacitaciones registradas.	4. El sistema habilita las funciones extendidas: registrar, editar, asignar participantes, registrar institución, generar certificados o reportes.
5. El actor selecciona una acción específica.	6A. El sistema redirige al flujo del sub caso correspondiente y actualiza los datos mostrados.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado.”	
A2. Si ocurre un error de conexión, el sistema muestra “Error al cargar las capacitaciones.”	

Tabla 37. Especificación de Caso de Uso: Gestionar Capacitaciones
Fuente: Elaboración Propia.

- **Listar Capacitaciones**

Caso de uso	Listar Capacitaciones
Actores	Jefe de Instrucción, comandante, Administrador
Tipo	Incluido
Propósito	Visualizar todas las capacitaciones registradas en el sistema, internas o externas.
Resumen	El sistema muestra la interfaz “P – CAPACITACIONES”, donde los actores pueden consultar, filtrar por tipo, fecha o institución, y acceder a las funciones extendidas
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede al módulo.	2. El sistema consulta la base de datos y muestra la lista de capacitaciones.
3. El actor aplica filtros o selecciona un registro.	4. El sistema actualiza la lista según los criterios seleccionados.
Curso Alternativo de Eventos	

A1. Si no existen registros, el sistema muestra “No se encontraron capacitaciones registradas.”

Tabla 38. Especificación de Caso de Uso: Gestionar Capacitaciones-Listar Capacitaciones
Fuente: Elaboración Propia.

- **Registrar Capacitación**

Caso de uso	Registrar Capacitación
Actores	Jefe de Instrucción, Administrador
Tipo	Extendido
Propósito	Registrar una nueva capacitación en el sistema.
Resumen	Permite ingresar los datos de una capacitación interna o externa, indicando nombre, tipo, fecha, institución asociada y responsable.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Registrar Capacitación”.	2. El sistema muestra la interfaz “P – REGISTRAR CAPACITACIÓN”.
3. El actor ingresa los datos requeridos.	4. El sistema valida los campos y guarda el registro en la base de datos.
	5A. El sistema muestra “Capacitación registrada correctamente.”
Curso Alternativo de Eventos	
A1. Si faltan datos requeridos, el sistema muestra “Complete todos los campos obligatorios.”	
A2. Si ya existe una capacitación con el mismo nombre y fecha, el sistema muestra “La capacitación ya está registrada.”	

Tabla 39. Especificación de Caso de Uso: Gestionar Capacitaciones-Registrar Capacitación
Fuente: Elaboración Propia.

- **Registrar Institución Externa**

Caso de uso	Registrar Institución Externa
Actores	Jefe de Instrucción, Administrador
Tipo	Extendido
Propósito	Registrar una institución externa asociada a una capacitación.
Resumen	Permite registrar entidades colaboradoras, instituciones invitadas o auspiciadoras de capacitaciones externas.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema

1. El actor selecciona "Registrar Institución".	2. El sistema muestra la interfaz "P – REGISTRAR INSTITUCIÓN".
3. El actor ingresa los datos (nombre, contacto, tipo, ubicación).	4A. El sistema guarda el registro y muestra "Institución registrada correctamente."
Curso Alternativo de Eventos	
A1. Si la institución ya existe, el sistema muestra "La institución ya está registrada."	

Tabla 40. Especificación de Caso de Uso: Gestionar Capacitaciones-Registrar Institución Externa
Fuente: Elaboración Propia.

- **Registrar Participantes (Internos / Externos)**

Caso de uso	Registrar Participantes (Internos / Externos)
Actores	Jefe de Instrucción
Tipo	Extendido
Propósito	Asignar participantes internos y externos a una capacitación registrada.
Resumen	Permite seleccionar miembros del personal interno y registrar participantes externos asociados a instituciones externas.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una capacitación.	2. El sistema muestra la lista de posibles participantes.
3. El actor selecciona internos y externos a agregar.	4A. El sistema guarda las asignaciones y muestra "Participantes registrados correctamente."
Curso Alternativo de Eventos	
A1. Si no hay participantes disponibles, el sistema muestra "No existen registros de participantes disponibles."	

Tabla 41. Especificación de Caso de Uso: Gestionar Capacitaciones-Registrar Participantes (Internos / Externos)
Fuente: Elaboración Propia.

- **Generar Certificados PDF con Código QR**

Caso de uso	Generar Certificados PDF con Código QR
Actores	Jefe de Instrucción, Administrador
Tipo	Extendido
Propósito	Generar certificados digitales con código QR para los participantes aprobados en una capacitación.
Resumen	El sistema genera certificados en formato PDF, con datos de la capacitación,

	participante y código QR verificable en línea.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una capacitación y elige “Generar Certificados”.	2. El sistema genera los certificados en formato PDF.
3. El sistema guarda los archivos y los vincula con los participantes.	4A. El sistema muestra “Certificados generados exitosamente.”
Curso Alternativo de Eventos	
A1. Si algún participante no cumple con los requisitos, el sistema muestra “No se puede generar certificado para el participante seleccionado.”	

Tabla 42. Especificación de Caso de Uso: Gestionar Capacitaciones-Generar Certificados PDF con Código QR
Fuente: Elaboración Propia.

- **Generar Reportes y Estadísticas**

Caso de uso	Generar Reportes y Estadísticas
Actores	Jefe de Instrucción, comandante
Tipo	Extendido
Propósito	Generar reportes estadísticos de capacitaciones y participación del personal.
Resumen	Permite generar reportes filtrando por fecha, tipo de capacitación, institución o instructor. Los reportes pueden exportarse en PDF.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Generar Reporte”.	2. El sistema solicita los filtros correspondientes.
3. El actor confirma.	4A. El sistema genera el reporte y lo muestra o exporta.
Curso Alternativo de Eventos	
A1. Si no existen datos para los filtros aplicados, el sistema muestra “No se encontraron registros para el reporte.”	

Tabla 43. Especificación de Caso de Uso: Gestionar Capacitaciones-Generar Reportes y Estadísticas
Fuente: Elaboración Propia.

II.1.2.6.9.7 Especificación de Caso de Uso: Gestionar Credenciales QR

- **Gestionar Credenciales QR**

Caso de uso	Gestionar Credenciales QR
Actores	Jefe de Instrucción, comandante, Administrador, Usuario Externo
Tipo	Primario

Propósito	Permitir la emisión, actualización, verificación y control de credenciales institucionales con código QR para el personal SAR.
Resumen	Este caso de uso describe las acciones necesarias para gestionar las credenciales institucionales del personal. Desde la interfaz “P – CREDENCIALES”, los actores pueden listar, registrar, editar, suspender, generar PDFs con código QR, y verificar su validez mediante un enlace público.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “Gestión de Credenciales QR” desde el menú principal.	2. El sistema valida los permisos y muestra la interfaz “P – CREDENCIALES” con la lista general.
3. El actor visualiza las credenciales registradas.	4. El sistema habilita las funciones extendidas: registrar, actualizar, suspender, generar certificados PDF o reportes.
5. El actor selecciona una acción.	6A. El sistema ejecuta el flujo del sub caso correspondiente y actualiza la información.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos, el sistema muestra “Acceso denegado.”	
A2. Si ocurre un error de conexión, el sistema muestra “Error al cargar la información de credenciales.”	

Tabla 44. Especificación de Caso de Uso: Gestionar Credenciales QR
Fuente: Elaboración Propia.

- **Listar Credenciales QR**

Caso de uso	Listar Credenciales
Actores	Jefe de Instrucción, comandante, Administrador
Tipo	Incluido
Propósito	Mostrar todas las credenciales emitidas con su estado actual (vigente, suspendida o revocada).
Resumen	El sistema muestra la interfaz “P – CREDENCIALES”, donde los actores pueden filtrar por estado, tipo o persona asociada, y acceder a las funciones extendidas.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema

1. El actor accede al módulo.	2. El sistema consulta la base de datos y muestra la lista de credenciales.
3. El actor filtra por estado o personal.	4. El sistema actualiza la lista.
Curso Alternativo de Eventos	
A1. Si no hay credenciales registradas, el sistema muestra “No se encontraron credenciales registradas.”	

Tabla 45. Especificación de Caso de Uso: Gestionar Credenciales QR-Listar Credenciales QR
Fuente: Elaboración Propia.

- **Registrar Credenciales**

Caso de uso	Registrar Credencial
Actores	Jefe de Instrucción, Administrador
Tipo	Extendido
Propósito	Registrar una nueva credencial asociada a un miembro del personal.
Resumen	Permite ingresar los datos de una nueva credencial, como número de serie, fecha de emisión y estado inicial, vinculada a un registro de personal.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Registrar Credencial”.	2. El sistema muestra la interfaz “P – REGISTRAR CREDENCIAL”.
3. El actor ingresa los datos y selecciona el personal asociado.	4. El sistema valida los campos y guarda la credencial.
	5A. El sistema muestra “Credencial registrada correctamente.”
Curso Alternativo de Eventos	
A1. Si el número de serie ya existe, el sistema muestra “El número de serie ingresado ya está registrado.”	
A2. Si falta información, el sistema muestra “Complete todos los campos obligatorios.”	

Tabla 46. Especificación de Caso de Uso: Gestionar Credenciales QR-Registrar Credencial
Fuente: Elaboración Propia.

- **Actualizar / Suspender / Revocar Credencial**

Caso de uso	Actualizar / Suspender / Revocar Credencial
Actores	Jefe de Instrucción, Administrador
Tipo	Extendido
Propósito	Cambiar el estado o actualizar los datos de una credencial ya emitida.

Resumen	Permite editar información o modificar el estado operativo de una credencial según corresponda.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una credencial y elige la opción “Editar / Suspende”.	2. El sistema muestra los datos actuales de la credencial.
3. El actor modifica la información o estado.	4A. El sistema guarda los cambios y muestra “Credencial actualizada correctamente.”
Curso Alternativo de Eventos	
A1. Si la credencial ya fue revocada, el sistema muestra “No se puede modificar una credencial revocada.”	

Tabla 47. Especificación de Caso de Uso: Gestionar Credenciales QR-Actualizar / Suspende / Revocar Credencial
Fuente: Elaboración Propia.

- **Generar Credencial PDF con Código QR**

Caso de uso	Generar Credencial PDF con Código QR
Actores	Jefe de Instrucción, Administrador
Tipo	Extendido
Propósito	Emitir y descargar la credencial institucional en formato PDF con código QR único.
Resumen	El sistema genera un archivo PDF con los datos del personal y su credencial, incluyendo un código QR que enlaza a la verificación pública.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una credencial y presiona “Generar PDF”.	2. El sistema genera el archivo en formato PDF.
3. El sistema guarda el documento y lo asocia a la credencial.	4A. El sistema muestra “Credencial generada correctamente.”
Curso Alternativo de Eventos	
A1. Si ocurre un error al generar el archivo, el sistema muestra “Error al generar la credencial, intente nuevamente.”	

Tabla 48. Especificación de Caso de Uso: Gestionar Credenciales QR-Generar Credencial PDF con Código QR
Fuente: Elaboración Propia.

II.1.2.6.9.8

- **Verificar Credencial por Código QR (Acceso Público)**

Caso de uso	Verificar Credencial por Código QR (Acceso Público)
Actores	Usuario Externo
Tipo	Extendido
Propósito	Permitir a cualquier usuario verificar la validez de una credencial mediante el escaneo del código QR.
Resumen	El actor externo escanea el código QR de una credencial, y el sistema muestra los datos de validación y estado (vigente o vencida) sin requerir autenticación.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor externo escanea el código QR de una credencial.	2. El sistema valida el número de serie en la base de datos.
3A. Si la credencial existe, muestra los datos del personal y el estado (Vigente / Revocada / Suspendida).	
Curso Alternativo de Eventos	
A1. Si el número de serie no existe, el sistema muestra "Credencial no encontrada o no válida."	

Tabla 49. Especificación de Caso de Uso: Gestionar Credenciales QR- Verificar Credencial por Código QR (Acceso Público)

Fuente: Elaboración Propia.

- **Generar Reportes de Credenciales**

Caso de uso	Generar Reportes de Credenciales
Actores	Comandante, Jefe de Instrucción
Tipo	Extendido
Propósito	Generar reportes estadísticos sobre las credenciales emitidas y su estado.
Resumen	Permite generar reportes por estado (vigente, vencida, suspendida), tipo o fecha de emisión.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona "Generar Reporte".	2. El sistema solicita los parámetros de filtro.
3. El actor confirma la acción.	4A. El sistema genera el reporte y lo muestra o exporta en formato PDF.
Curso Alternativo de Eventos	
A1. Si no existen registros, el sistema muestra "No se encontraron credenciales para los parámetros seleccionados."	

Tabla 50. Especificación de Caso de Uso: Gestionar Credenciales QR-Generar Reportes de Credenciales
Fuente: Elaboración Propia.

II.1.2.6.9.9

II.1.2.6.9.10 Especificación de Caso de Uso: Gestionar Emergencias

- **Gestionar Emergencias**

Caso de uso	Gestionar Emergencias
Actores	Comandante, Jefe de Operaciones, Administrador, Personal Operativo
Tipo	Primario
Propósito	Permitir la gestión integral de emergencias atendidas por la institución SAR, incluyendo el registro de datos, asignación de personal, vehículos e inventario, control de estados y generación de reportes.
Resumen	Este caso de uso describe las acciones necesarias para administrar las emergencias desde su registro hasta su cierre. Los actores pueden registrar nuevas emergencias, asignar recursos, cambiar su estado operativo y generar reportes. El sistema asegura la trazabilidad y la correcta asignación de recursos humanos y materiales.
Curso Normal de Eventos	

Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “Gestión de Emergencias” desde el menú principal.	2. El sistema valida los permisos y muestra la interfaz “P – EMERGENCIAS” con la lista de registros.
3. El actor visualiza las emergencias existentes.	4. El sistema habilita las funciones: registrar, editar, asignar recursos, cambiar estado o generar reportes.
5. El actor selecciona una acción.	6A. El sistema ejecuta la función correspondiente y actualiza la información.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado.”	
A2. Si ocurre un error de conexión, el sistema muestra “Error al cargar las emergencias.”	

Tabla 51. Especificación de Caso de Uso: Gestionar Emergencias
Fuente: Elaboración Propia.

- **Listar Emergencias**

Caso de uso	Listar Emergencias
Actores	Comandante, Jefe de Operaciones, Administrador
Tipo	Incluido
Propósito	Visualizar todas las emergencias registradas en el sistema, con su tipo, ubicación, estado y fecha.
Resumen	El sistema muestra la interfaz “P – EMERGENCIAS”, donde los actores pueden filtrar registros por tipo, estado o rango de fechas.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede al módulo.	2. El sistema consulta la base de datos y muestra la lista de emergencias.
3. El actor aplica filtros o selecciona una emergencia.	4. El sistema actualiza la lista según los filtros seleccionados.
Curso Alternativo de Eventos	
A1. Si no existen registros, el sistema muestra “No se encontraron emergencias registradas.”	

Tabla 52. Especificación de Caso de Uso: Gestionar Emergencias-Listar Emergencias
Fuente: Elaboración Propia.

- Registrar Emergencias

Caso de uso	Registrar Emergencia
Actores	Jefe de Operaciones, Administrador
Tipo	Extendido
Propósito	Registrar una nueva emergencia en el sistema
Resumen	Permite ingresar los datos de la emergencia, como tipo, ubicación, fecha, descripción y responsable del registro.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Registrar Emergencia”.	2. El sistema muestra la interfaz “P – REGISTRAR EMERGENCIA”.
3. El actor ingresa los datos requeridos.	4A. El sistema guarda el registro y muestra “Emergencia registrada correctamente.”
Curso Alternativo de Eventos	
A1. Si faltan datos obligatorios, el sistema muestra “Complete todos los campos requeridos.”	

Tabla 53. Especificación de Caso de Uso: Gestionar Emergencias–Registrar Emergencia
Fuente: Elaboración Propia.

II.1.2.6.9.11

- Editar / Actualizar Emergencia

Caso de uso	Editar / Actualizar Emergencia
Actores	Jefe de Operaciones
Tipo	Extendido
Propósito	Modificar la información de una emergencia registrada.
Resumen	Permite actualizar la descripción, ubicación o tipo de una emergencia existente.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una emergencia y presiona “Editar”.	2. El sistema muestra la información actual.
3. El actor modifica los datos necesarios.	4A. El sistema guarda los cambios y muestra “Emergencia actualizada correctamente.”
Curso Alternativo de Eventos	
A1. Si la emergencia está cerrada, el sistema muestra “No se puede modificar una emergencia finalizada.”	

Tabla 54. Especificación de Caso de Uso: Gestionar Emergencias-Editar / Actualizar Emergencia
Fuente: Elaboración Propia.

- **Asignar Vehículos**

Caso de uso	Asignar Vehículos
Actores	Jefe de Operaciones
Tipo	Extendido
Propósito	Asignar vehículos disponibles a una emergencia específica.
Resumen	Permite seleccionar vehículos del parque operativo y asignarlos a la emergencia.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una emergencia.	2. El sistema muestra la lista de vehículos disponibles.
3. El actor selecciona uno o varios vehículos y confirma.	4A. El sistema asocia los vehículos a la emergencia y muestra "Vehículos asignados correctamente."
Curso Alternativo de Eventos	
A1. Si no hay vehículos disponibles, el sistema muestra "No existen vehículos disponibles para asignar."	

Tabla 55. Especificación de Caso de Uso: Gestionar Emergencias-Asignar Vehículos
Fuente: Elaboración Propia.

II.1.2.6.9.12

- **Asignar Personal**

Caso de uso	Asignar Personal
Actores	Jefe de Operaciones
Tipo	Extendido
Propósito	Asignar personal operativo a una emergencia activa.
Resumen	Permite seleccionar miembros activos del personal para participar en la emergencia.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona la opción "Asignar Personal".	2. El sistema muestra el listado de personal activo.
3. El actor selecciona los miembros a asignar.	4A. El sistema guarda la asignación y muestra "Personal asignado correctamente."
Curso Alternativo de Eventos	
A1. Si no hay personal disponible, el sistema muestra "No existe personal disponible para la asignación."	

Tabla 56. Especificación de Caso de Uso: Gestionar Emergencias-Asignar Personal
Fuente: Elaboración Propia.

II.1.2.6.9.13

- Registrar Materiales Utilizados

Caso de uso	Registrar Materiales Utilizados
Actores	Jefe de Operaciones, Administrador
Tipo	Extendido
Propósito	Registrar los materiales e insumos del inventario utilizados durante la emergencia.
Resumen	Permite vincular ítems del inventario al registro de la emergencia y descontarlos del stock.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una emergencia y elige "Registrar Materiales".	2. El sistema muestra la lista de materiales disponibles.
3. El actor selecciona los materiales y cantidades utilizadas.	4A. El sistema actualiza el inventario y muestra "Materiales registrados correctamente."
Curso Alternativo de Eventos	
A1. Si el inventario está en cero, el sistema muestra "No hay stock disponible para este material."	

Tabla 57. Especificación de Caso de Uso: Gestionar Emergencias-Registrar Materiales Utilizados
Fuente: Elaboración Propia.

- **Cambiar Estado de Emergencia**

Caso de uso	Cambiar Estado de Emergencia
Actores	Jefe de Operaciones, comandante
Tipo	Extendido
Propósito	Actualizar el estado operativo de la emergencia (Pendiente, En curso, Finalizada o Completa).
Resumen	Permite registrar el avance de la emergencia hasta su cierre operativo.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona una emergencia.	2. El sistema muestra el estado actual.
3. El actor selecciona el nuevo estado.	4A. El sistema actualiza el estado y muestra "Estado de la emergencia actualizado correctamente."
Curso Alternativo de Eventos	
A1. Si la emergencia ya está completa, el sistema muestra "No se puede modificar una emergencia finalizada."	

Tabla 58. Especificación de Caso de Uso: Gestionar Emergencias-Cambiar Estado de Emergencia
Fuente: Elaboración Propia.

- **Generar Reportes de Emergencias**

Caso de uso	Generar Reportes de Emergencias
Actores	Comandante, Jefe de Operaciones, Administrador
Tipo	Extendido
Propósito	Generar reportes estadísticos de emergencias atendidas.
Resumen	Permite generar reportes por tipo, ubicación, estado o periodo, y exportarlos en formato PDF.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona "Generar Reporte".	2. El sistema solicita parámetros del reporte.
3. El actor confirma.	4A. El sistema genera el reporte y lo muestra o exporta.
Curso Alternativo de Eventos	
A1. Si no hay registros disponibles, el sistema muestra "No se encontraron datos para el reporte seleccionado."	

Tabla 59. Especificación de Caso de Uso: Gestionar Emergencias-Generar Reportes de Emergencias
Fuente: Elaboración Propia.

II.1.2.6.9.14 Especificación de Caso de Uso: Gestionar Inventario

- Gestionar Inventario

Caso de uso	Gestionar Inventario
Actores	Comandante, Jefe de Logística, Jefe de Operaciones, Administrador
Tipo	Primario
Propósito	Permitir la administración completa de los recursos e insumos institucionales, incluyendo el registro, actualización, control de stock, movimientos, catálogos y generación de reportes.
Resumen	Este caso de uso permite al personal autorizado gestionar el inventario general de la institución. Desde la interfaz “P – INVENTARIO”, los actores pueden registrar nuevos ítems, modificar los existentes, registrar entradas o salidas, controlar el stock mínimo, y generar reportes de existencias o movimientos.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “ Gestión de Inventario ” desde el menú principal.	2. El sistema valida los permisos y muestra la interfaz “P – INVENTARIO” con la lista de ítems.
3. El actor visualiza los registros disponibles.	4. El sistema habilita las funciones extendidas: registrar, editar, eliminar, registrar movimientos, gestionar catálogos o generar reportes.
5. El actor selecciona una acción específica.	6A. El sistema ejecuta el flujo del sub caso correspondiente y actualiza la información.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado.”	
A2. Si ocurre un error de conexión o base de datos, el sistema muestra “Error al cargar el inventario.”	

Tabla 60. Especificación de Caso de Uso: Gestionar Inventario
Fuente: Elaboración Propia.

- Listar Ítems de Inventario

Caso de uso	Listar Ítems de Inventario
Actores	Comandante, Jefe de Logística, Jefe de Operaciones, Administrador
Tipo	Incluido

Propósito	Mostrar todos los ítems registrados en el inventario institucional.
Resumen	El sistema muestra la interfaz “P – INVENTARIO”, donde los actores pueden filtrar por categoría, tipo, ubicación o estado de stock.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede al módulo.	2. El sistema consulta la base de datos y muestra la lista de ítems.
3. El actor aplica filtros.	4. El sistema actualiza la lista según los criterios seleccionados.
Curso Alternativo de Eventos	
A1. Si no existen registros, el sistema muestra “No se encontraron ítems registrados.”	

Tabla 61. Especificación de Caso de Uso: Gestionar Inventario- Listar Ítems de Inventario
Fuente: Elaboración Propia.

- **Registrar Ítem de Inventario**

Caso de uso	Registrar Ítem de Inventario
Actores	Jefe de Logística, Administrador
Tipo	Extendido
Propósito	Registrar un nuevo ítem dentro del inventario general.
Resumen	Permite ingresar los datos de un nuevo recurso, especificando su categoría, unidad, cantidad, proveedor y ubicación.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “ Registrar Ítem ”.	2. El sistema muestra la interfaz “P – REGISTRAR ÍTEM”.
3. El actor ingresa los datos requeridos.	4A. El sistema guarda el registro y muestra “Ítem registrado correctamente.”
Curso Alternativo de Eventos	
A1. Si faltan datos, el sistema muestra “ Complete todos los campos obligatorios. ”	
A2. Si el ítem ya existe, el sistema muestra “ El ítem ingresado ya está registrado. ”	

Tabla 62. Especificación de Caso de Uso: Gestionar Inventario-Registrar Ítem de Inventario
Fuente: Elaboración Propia.

- **Editar / Actualizar Ítem**

Caso de uso	Editar / Actualizar Ítem
Actores	Jefe de Logística, Administrador
Tipo	Extendido
Propósito	Modificar los datos de un ítem existente.

Resumen	Desde la lista de inventario, el actor puede actualizar información del ítem o ajustar la cantidad en stock.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un ítem y presiona "Editar".	2. El sistema muestra los datos actuales.
3. El actor modifica los campos necesarios.	4A. El sistema guarda los cambios y muestra "Ítem actualizado correctamente."
Curso Alternativo de Eventos	
A1. Si el ítem está vinculado a movimientos activos, el sistema muestra "No se puede modificar un ítem en uso."	

Tabla 63. Especificación de Caso de Uso: Gestionar Inventario-Editar / Actualizar Ítem
Fuente: Elaboración Propia.

- **Registrar Movimiento de Inventario**

Caso de uso	Registrar Movimiento de Inventario
Actores	Jefe de Logística, Jefe de Operaciones
Tipo	Extendido
Propósito	Registrar las entradas, salidas o consumos de ítems en el inventario.
Resumen	Permite controlar los movimientos de stock indicando tipo de movimiento, cantidad, destino y motivo.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un ítem y presiona "Registrar Movimiento".	2. El sistema muestra el formulario de movimiento.
3. El actor completa los datos del movimiento (entrada, salida, consumo).	4A. El sistema guarda el registro y actualiza el stock.
Curso Alternativo de Eventos	
A1. Si la cantidad de salida excede el stock, el sistema muestra "Stock insuficiente para realizar la operación."	

Tabla 64. Especificación de Caso de Uso: Gestionar Inventario-Registrar Movimiento de Inventario
Fuente: Elaboración Propia.

- **Controlar Stock y Notificaciones de Cantidad Crítica**

Caso de uso	Controlar Stock y Notificaciones de Cantidad Crítica
Actores	Jefe de Logística
Tipo	Extendido
Propósito	Supervisar los niveles de inventario y generar alertas por bajo stock.

Resumen	El sistema compara el stock actual con los niveles mínimos configurados y muestra alertas visuales o notificaciones.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El sistema realiza una verificación periódica del stock.	2. Si el stock de un ítem está por debajo del mínimo, el sistema genera una alerta visual.
3. El actor revisa las alertas y gestiona el reabastecimiento.	4A. El sistema actualiza las notificaciones al reponer el stock.
Curso Alternativo de Eventos	
A1. Si no existen ítems en nivel crítico, el sistema muestra “Todos los niveles de stock son adecuados.”	

Tabla 65. Especificación de Caso de Uso: Gestionar Inventario-Controlar Stock y Notificaciones de Cantidad Crítica
Fuente: Elaboración Propia.

- **Gestionar Catálogos**

Caso de uso	Gestionar Catálogos
Actores	Administrador
Tipo	Extendido
Propósito	Mantener actualizadas las categorías, unidades, ubicaciones y proveedores utilizados en el inventario.
Resumen	Permite registrar, modificar o eliminar elementos de los catálogos que se usan para clasificar los ítems.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Gestionar Catálogos”.	2. El sistema muestra la lista de catálogos disponibles.
3. El actor registra o edita un elemento.	4A. El sistema guarda los cambios y muestra “Catálogo actualizado correctamente.”
Curso Alternativo de Eventos	
A1. Si se intenta eliminar un elemento en uso, el sistema muestra “No se puede eliminar un catálogo vinculado a un ítem.”	

Tabla 66. Especificación de Caso de Uso: Gestionar Inventario-Gestionar Catálogos
Fuente: Elaboración Propia.

- **Generar Reportes de Inventario y Movimientos**

Caso de uso	Generar Reportes de Inventario y Movimientos
Actores	Comandante, Jefe de Logística, Jefe de Operaciones

Tipo	Extendido
Propósito	Generar reportes sobre existencias, movimientos y niveles de stock.
Resumen	Permite generar reportes por tipo, categoría, fecha o estado, y exportarlos en formato PDF.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Generar Reporte”.	2. El sistema solicita parámetros de filtrado.
3. El actor confirma.	4A. El sistema genera y muestra el reporte solicitado.
Curso Alternativo de Eventos	
A1. Si no hay datos disponibles, el sistema muestra “No se encontraron resultados para el reporte.”	

Tabla 67. Especificación de Caso de Uso: Gestionar Inventario-Generar Reportes de Inventario y Movimientos
Fuente: Elaboración Propia.

II.1.2.6.9.15 Especificación de Caso de Uso: Gestionar Donativos

- Gestionar Donativos

Caso de uso	Gestionar Donativos
Actores	Comandante, Jefe de Logística, Administrador, Donante / Institución Externa
Tipo	Primario
Propósito	Permitir la gestión completa de los donativos materiales, monetarios o en servicios, incluyendo su registro, actualización, control de estado y generación de reportes.
Resumen	Este caso de uso describe las acciones necesarias para registrar, actualizar y controlar los donativos recibidos por la institución SAR. Desde la interfaz “P – DONATIVOS”, los actores pueden registrar nuevos aportes, asociar instituciones, actualizar su estado, vincular materiales al inventario y generar reportes de transparencia.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona el módulo “Gestión de Donativos” desde el menú principal.	2. El sistema valida los permisos y muestra la interfaz “P – DONATIVOS” con el listado general.

3. El actor visualiza los donativos registrados.	4. El sistema habilita las funciones extendidas: registrar, editar, eliminar, vincular con inventario o generar reportes.
5. El actor selecciona una acción específica.	6A. El sistema ejecuta la operación correspondiente y actualiza los datos.
Curso Alternativo de Eventos	
A1. Si el actor no tiene permisos suficientes, el sistema muestra “Acceso denegado.”	
A2. Si ocurre un error de conexión o consulta, el sistema muestra “Error al cargar la información de donativos.”	

Tabla 68. Especificación de Caso de Uso: Gestionar Donativos-
Fuente: Elaboración Propia.

- **Listar Donativos**

Caso de uso	Listar Donativos
Actores	Comandante, Jefe de Logística, Administrador
Tipo	Incluido
Propósito	Visualizar todos los donativos registrados en el sistema, con su tipo, estado, fecha e institución asociada.
Resumen	El sistema muestra la interfaz “P – DONATIVOS”, donde los actores pueden consultar el listado general de donativos y acceder a las funciones extendidas.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor accede al módulo.	2. El sistema consulta la base de datos y muestra la lista de donativos.
3. El actor aplica filtros por tipo o estado.	4. El sistema actualiza la lista según los filtros aplicados.
Curso Alternativo de Eventos	
A1. Si no existen registros, el sistema muestra “No se encontraron donativos registrados.”	

Tabla 69. Especificación de Caso de Uso: Gestionar Donativos-Listar Donativos
Fuente: Elaboración Propia.

- **Registrar Donativos**

Caso de uso	Registrar Donativo
Actores	Jefe de Logística, Administrador
Tipo	Extendido
Propósito	Registrar un nuevo donativo recibido por la institución.

Resumen	Permite ingresar la información del donativo, su tipo (material, monetario o servicio), cantidad, descripción, donante e institución asociada.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “ Registrar Donativo ”.	2. El sistema muestra la interfaz “P – REGISTRAR DONATIVO”.
3. El actor ingresa los datos requeridos (tipo, cantidad, descripción, institución).	4A. El sistema guarda el registro y muestra “Donativo registrado correctamente.”
Curso Alternativo de Eventos	
A1. Si faltan datos, el sistema muestra “Complete todos los campos obligatorios.”	
A2. Si el donativo ya fue ingresado, el sistema muestra “El donativo ya se encuentra registrado.”	

Tabla 70. Especificación de Caso de Uso: Gestionar Donativos-
Fuente: Elaboración Propia.

- **Registrar Donante o Institución Asociada**

Caso de uso	Registrar Donante o Institución Asociada
Actores	Jefe de Logística, Administrador
Tipo	Extendido
Propósito	Registrar la información de un donante o institución que realiza aportes.
Resumen	Permite ingresar datos de personas o instituciones que donan recursos, para vincularlos a los donativos registrados.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona “Registrar Donante”.	2. El sistema muestra la interfaz “P – REGISTRAR DONANTE / INSTITUCIÓN”.
3. El actor ingresa los datos (nombre, contacto, tipo, observaciones).	4A. El sistema guarda el registro y muestra “Donante registrado correctamente.”
Curso Alternativo de Eventos	
A1. Si el donante ya existe, el sistema muestra “El donante ya está registrado.”	

Tabla 71. Especificación de Caso de Uso: Gestionar Donativos-Registrar Donante o Institución Asociada
Fuente: Elaboración Propia.

- **Actualizar / Cambiar Estado del Donativo**

Caso de uso	Actualizar / Cambiar Estado del Donativo
Actores	Jefe de Logística, Administrador
Tipo	Extendido

Propósito	Modificar la información o estado de un donativo (pendiente, distribuido o entregado).
Resumen	Permite editar los datos del donativo y cambiar su estado operativo según su distribución o uso.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un donativo.	2. El sistema muestra los datos actuales.
3. El actor modifica la información o estado.	4A. El sistema guarda los cambios y muestra "Donativo actualizado correctamente."
Curso Alternativo de Eventos	
A1. Si el donativo ya fue entregado, el sistema muestra "No se puede modificar un donativo cerrado."	

Tabla 72. Especificación de Caso de Uso: Gestionar Donativos-Actualizar / Cambiar Estado del Donativo
Fuente: Elaboración Propia.

- **Asociar Donativo Material al Inventario**

Caso de uso	Asociar Donativo Material al Inventario
Actores	Jefe de Logística, Administrador
Tipo	Extendido
Propósito	Incorporar los materiales donados al inventario institucional.
Resumen	Permite vincular los donativos materiales al inventario para su control y distribución posterior.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona un donativo de tipo material.	2. El sistema muestra los ítems del inventario.
3. El actor asocia el donativo al inventario correspondiente.	4A. El sistema actualiza las existencias y muestra "Donativo asociado al inventario correctamente."
Curso Alternativo de Eventos	
A1. Si el ítem no existe en el inventario, el sistema muestra "No se encontró un ítem coincidente en el inventario."	

Tabla 73. Especificación de Caso de Uso: Gestionar Donativos-Asociar Donativo Material al Inventario
Fuente: Elaboración Propia.

- **Regenerar Reportes de Donativos**

Caso de uso	Generar Reportes de Donativos
Actores	Comandante, Jefe de Logística, Administrador

Tipo	Extendido
Propósito	Generar reportes de transparencia y control de donativos recibidos.
Resumen	Permite generar reportes filtrando por tipo de donativo, estado, institución o periodo, y exportarlos en formato PDF.
Curso Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El actor selecciona "Generar Reporte".	2. El sistema solicita los filtros correspondientes.
3. El actor confirma la acción.	4A. El sistema genera el reporte y lo muestra o exporta.
Curso Alternativo de Eventos	
A1. Si no hay datos disponibles, el sistema muestra "No se encontraron resultados para el reporte seleccionado."	

Tabla 74. Especificación de Caso de Uso: Gestionar Donativos-Generar Reportes de Donativos
Fuente: Elaboración Propia.

II.1.2.6.10 Diagrama de Actividades

II.1.2.6.10.1 Diagrama de Actividades: Gestionar Inicio de Sesión

- Iniciar Sesión

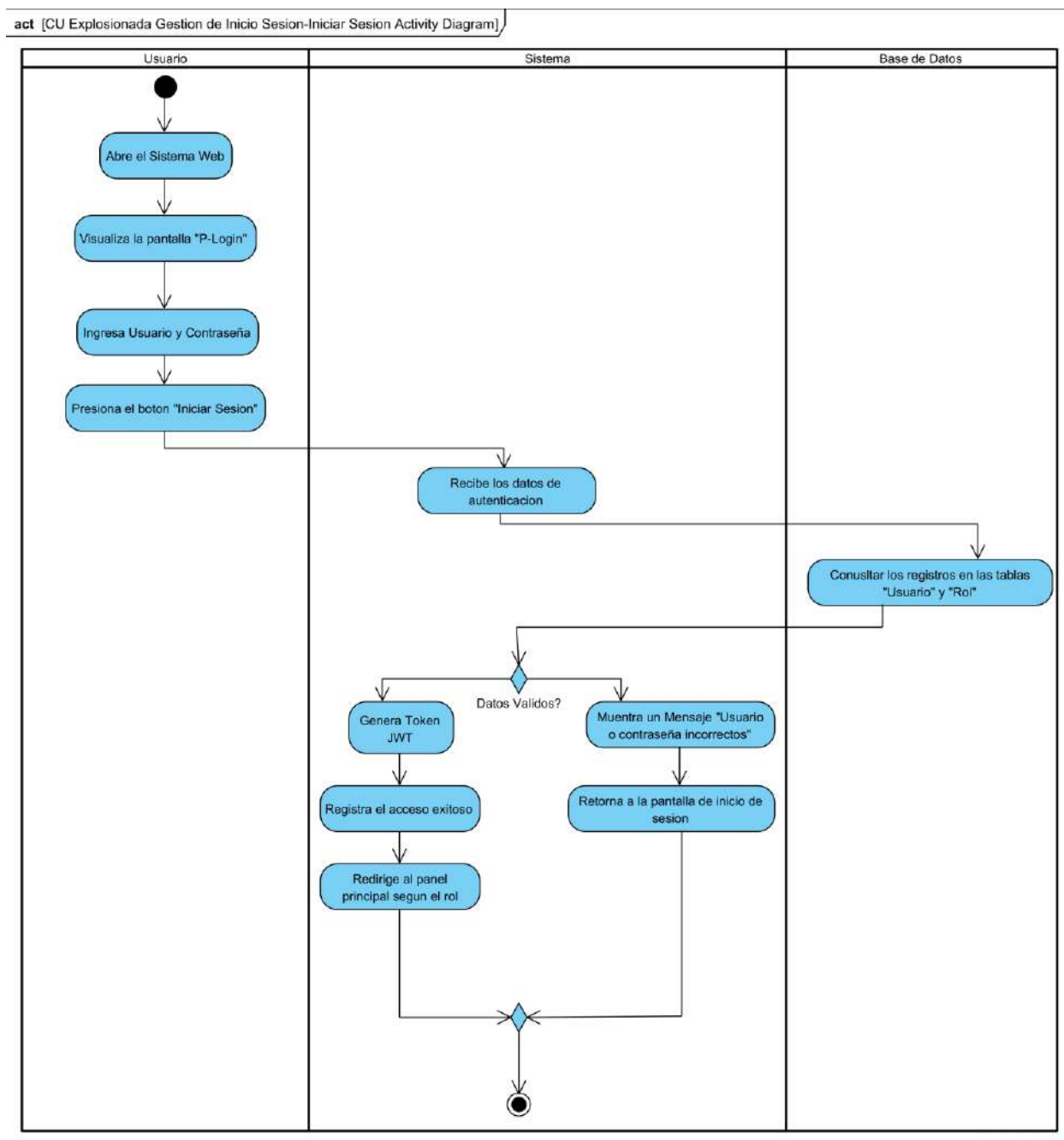


Figura 14. Diagrama de Actividades: Gestionar Inicio Sesión-Iniciar Sesión
Fuente: Elaboración Propia.

- **Cerrar Sesión**

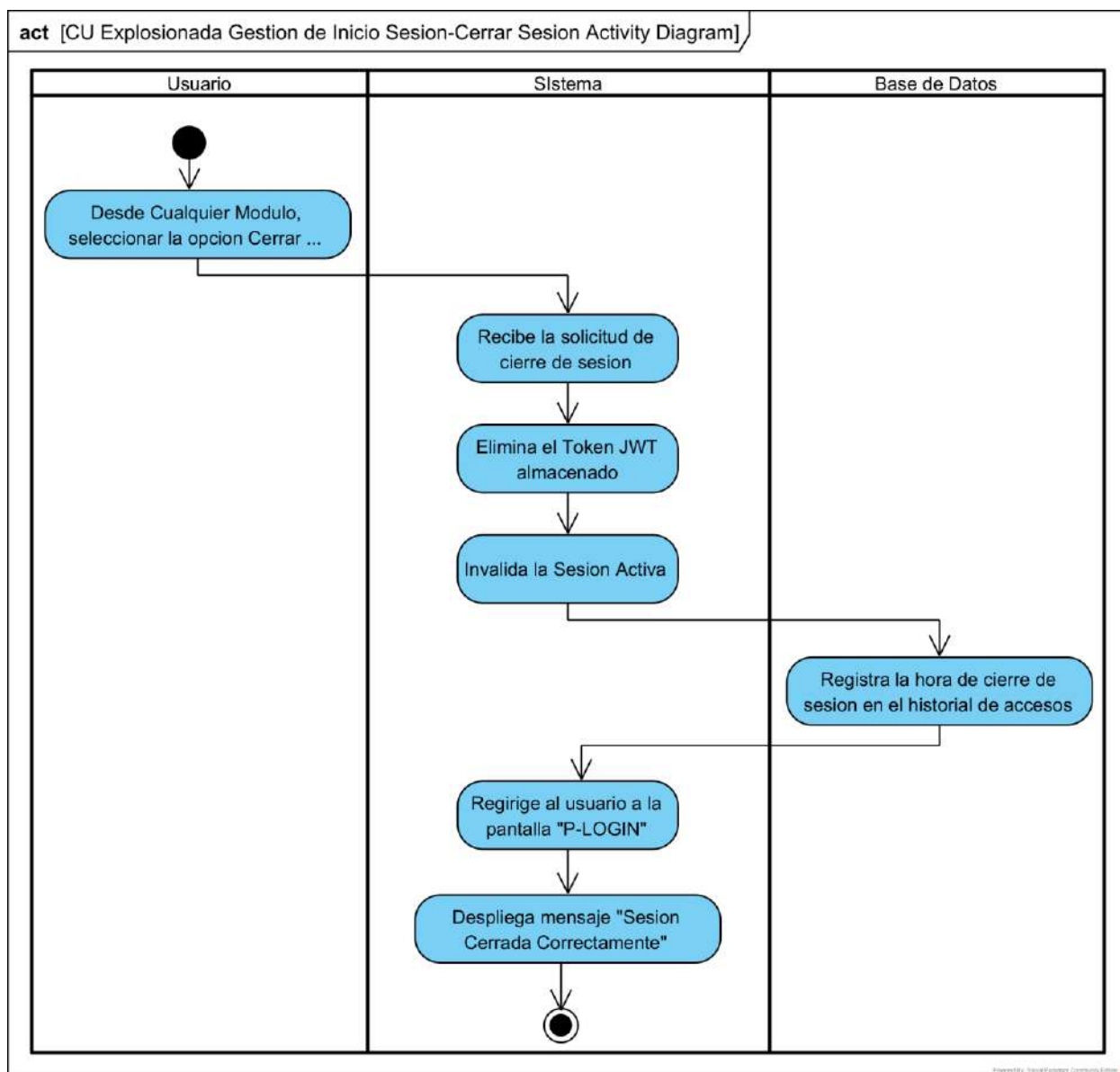


Figura 15. Diagrama de Actividades: Gestionar Iniciar Sesión-Cerrar Sesión
Fuente: Elaboración Propia.

II.1.2.6.10.2 Diagrama de Actividades: Gestión de Emergencias

- **Listar Emergencias**

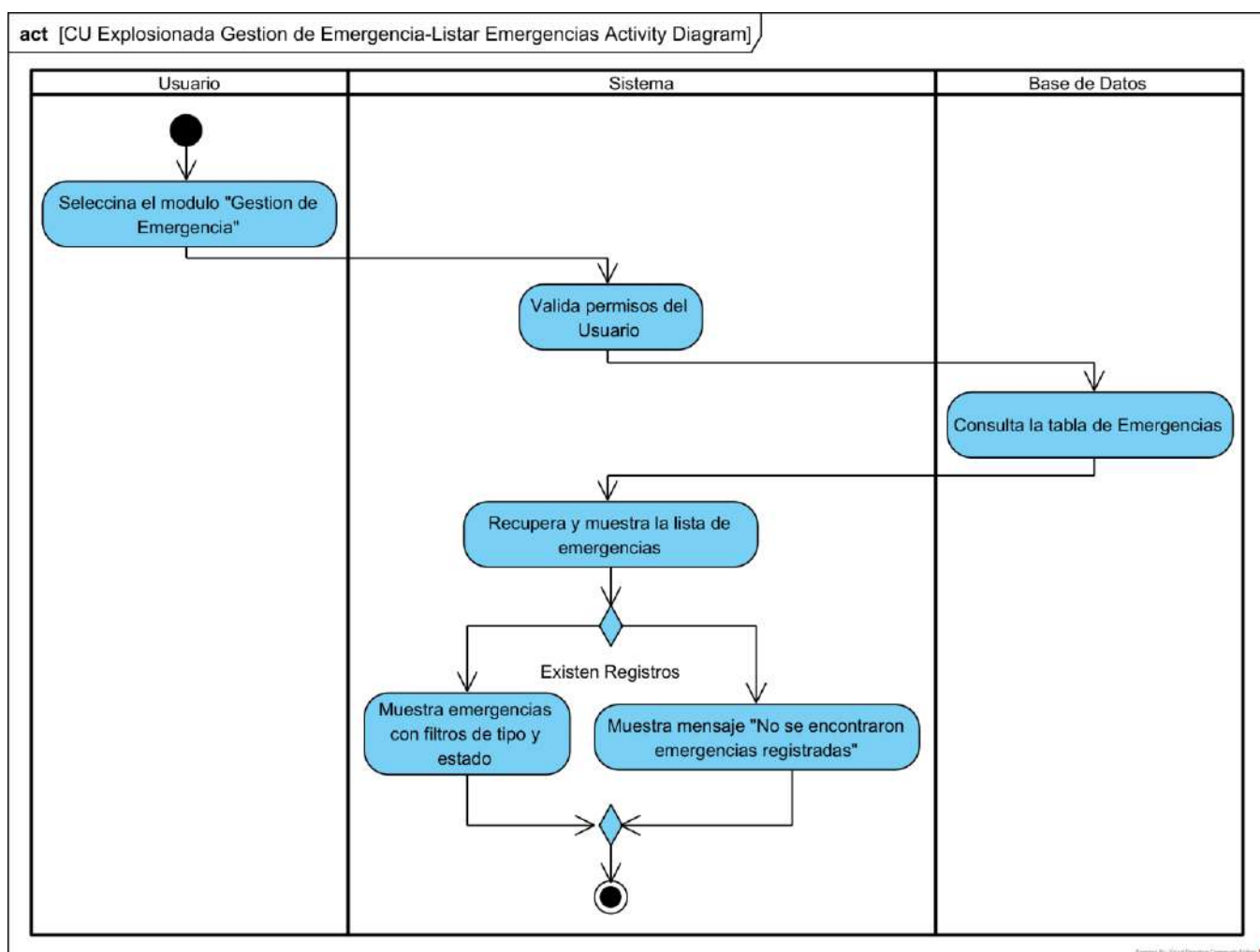


Figura 16. Diagrama de Actividades: Gestionar Gestión de Emergencias-Listar Emergencias
Fuente: Elaboración Propia.

- Registrar Emergencias

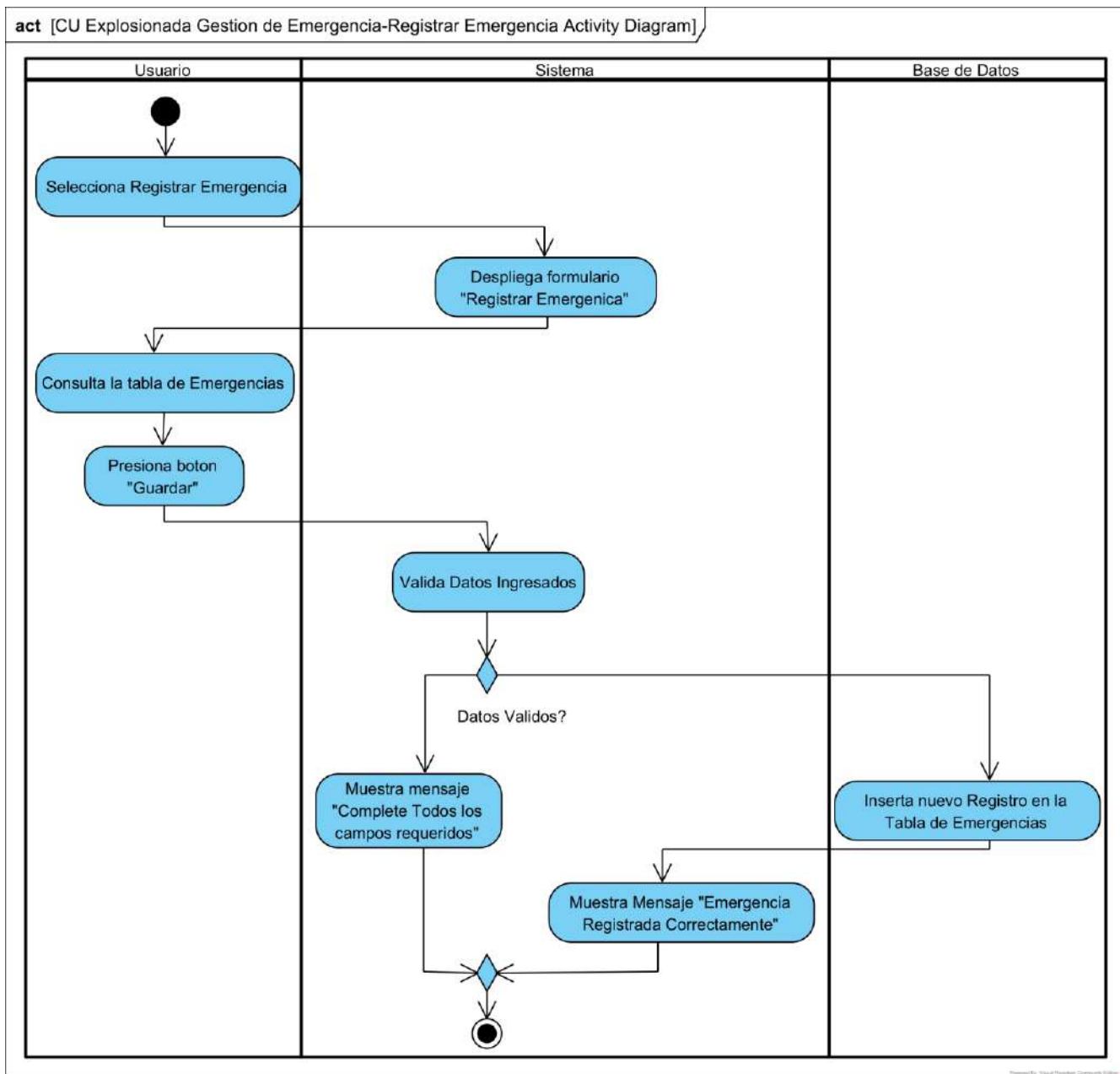


Figura 17. Diagrama de Actividades: Gestionar Gestión de Emergencias-Registrar Emergencias
Fuente: Elaboración Propia.

- **Editar / Actualizar Emergencia**

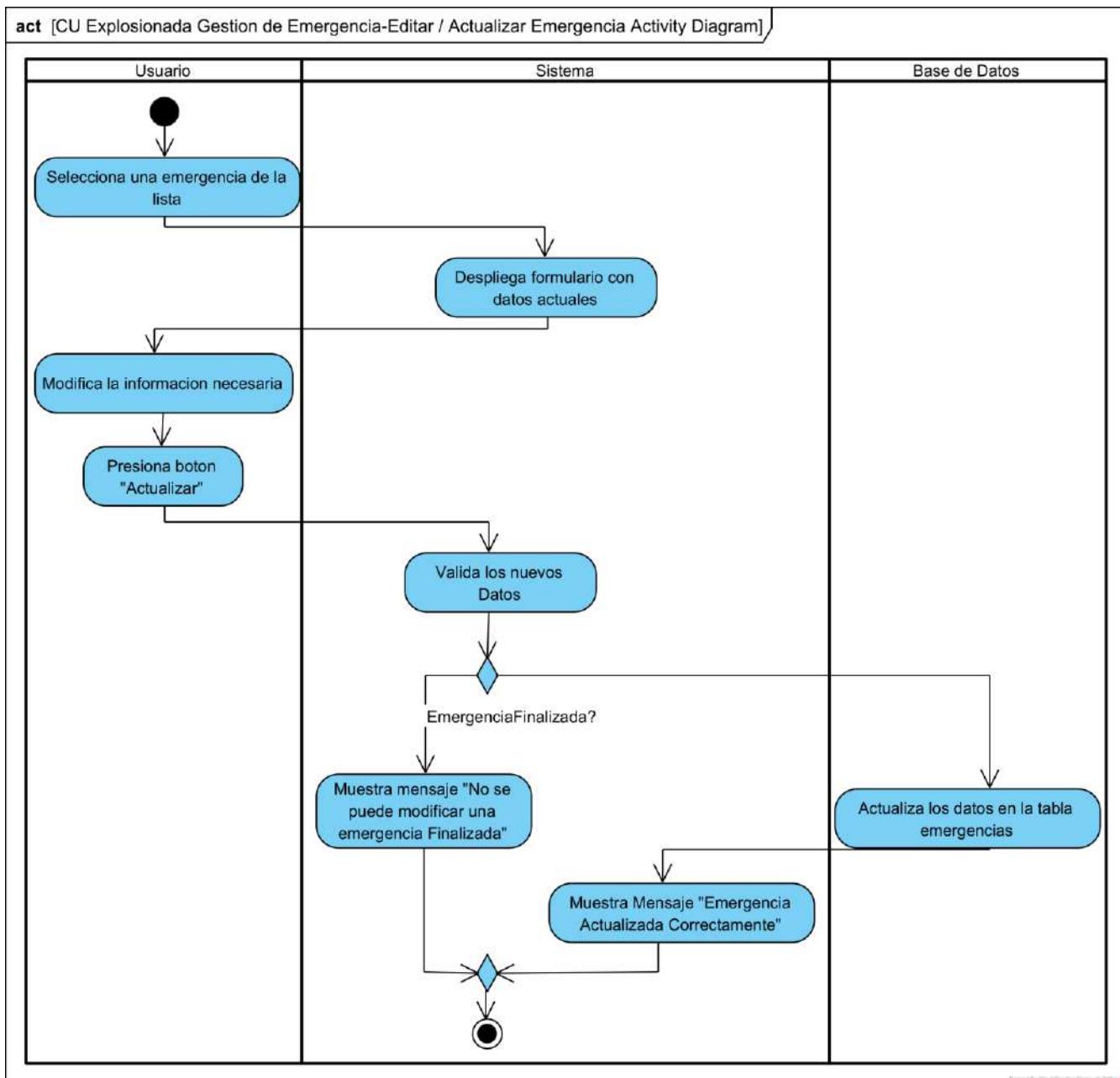
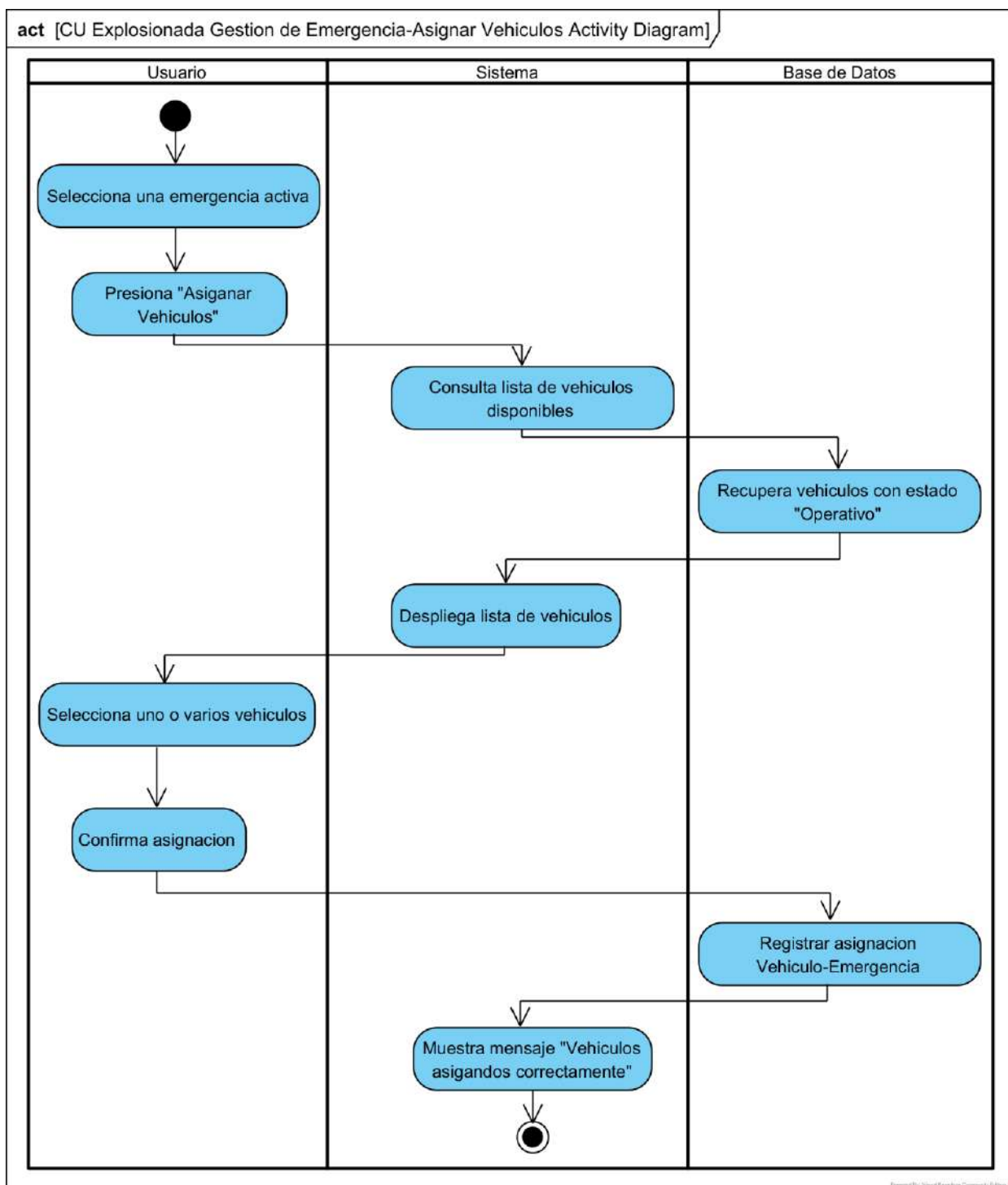


Figura 18. Diagrama de Actividades: Gestionar Gestión de Emergencias-Editar/Actualizar Emergencia
Fuente: Elaboración Propia.

- **Asignar Vehículos**



- **Asignar Personal**

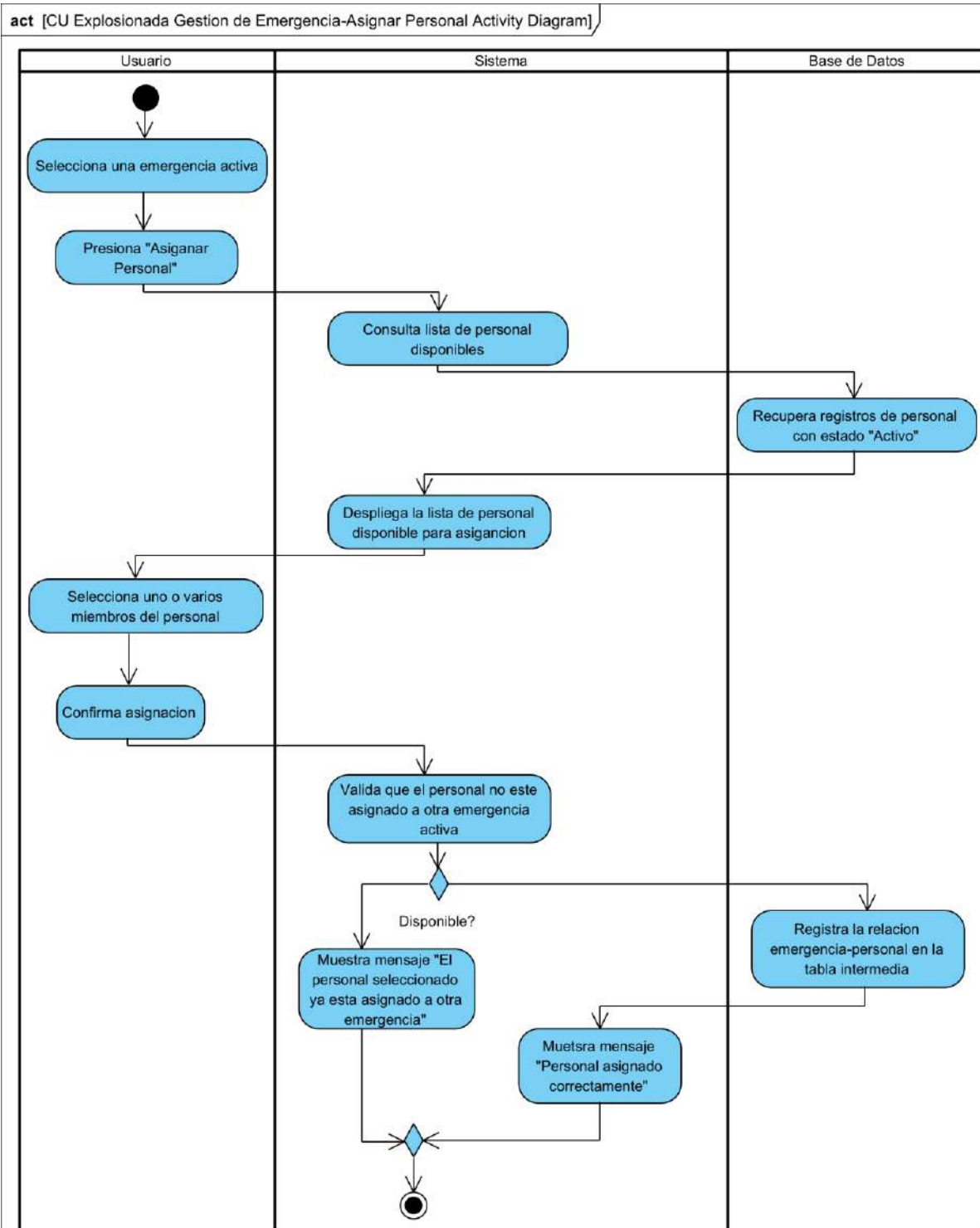


Figura 20. Diagrama de Actividades: Gestionar Gestión de Emergencias-Asignar Personal
Fuente: Elaboración Propia.

- **Registrar Materiales Utilizados**

act [CU Explosionada Gestion de Emergencia-Registrar Materiales Utilizados Activity Diagram]

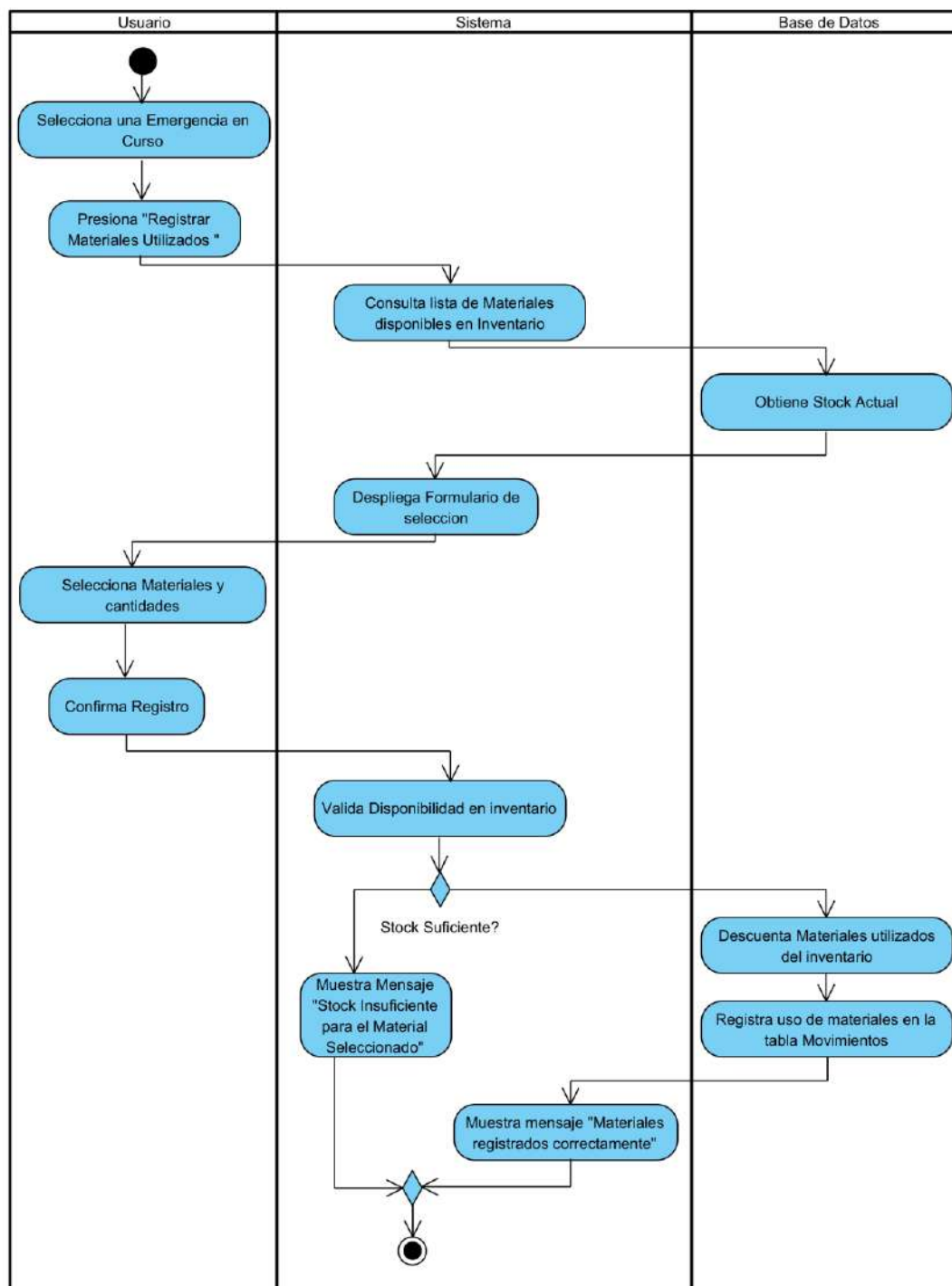


Figura 21. Diagrama de Actividades: Gestionar Gestión de Emergencias-Registrar Materiales utilizados
Fuente: Elaboración Propia.

- **Cambiar Estado de Emergencia**

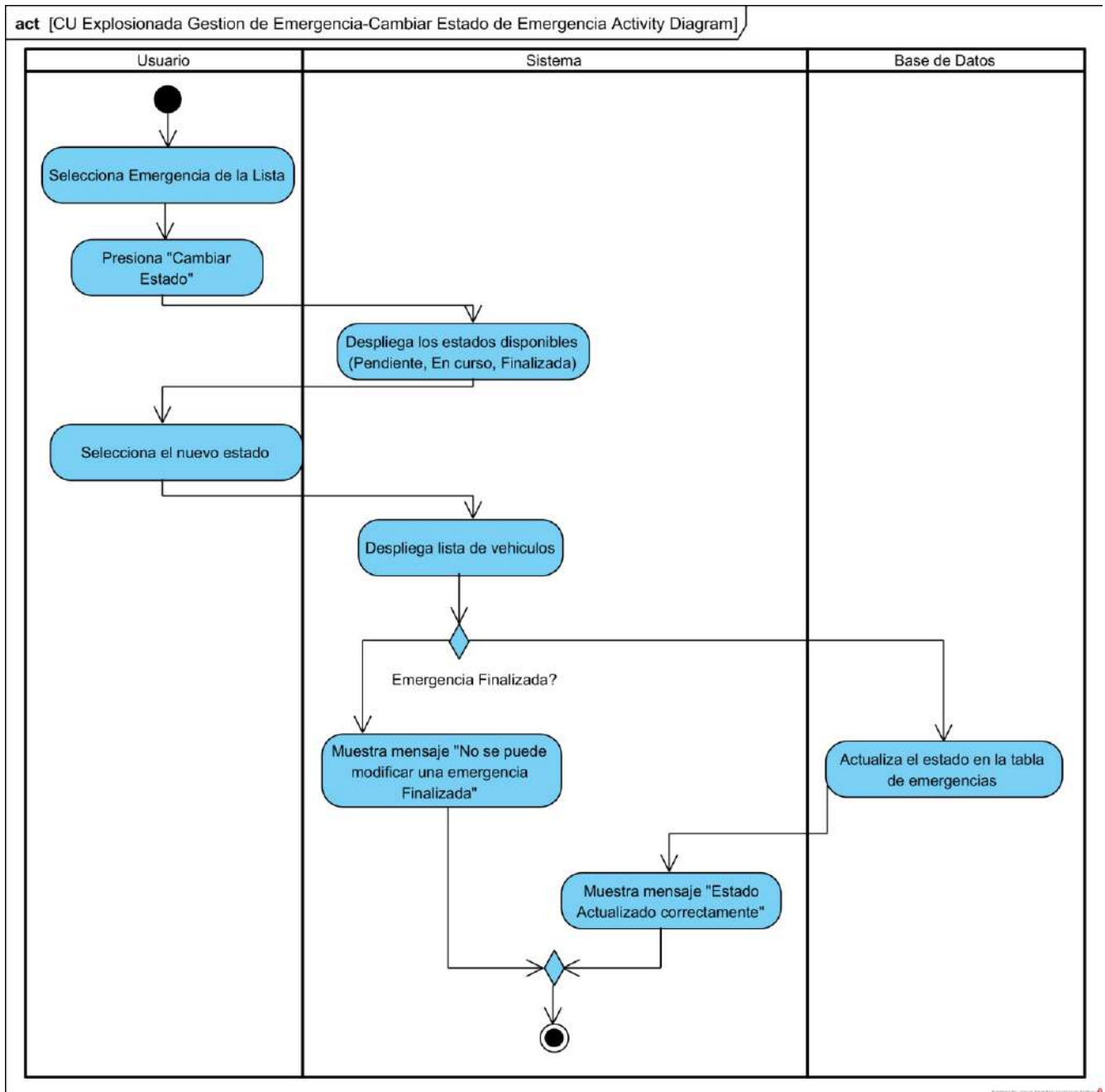


Figura 22. Diagrama de Actividades: Gestionar Gestión de Emergencias-Cambiar Estado de Emergencia
Fuente: Elaboración Propia.

- **Generar Reportes de Emergencias**

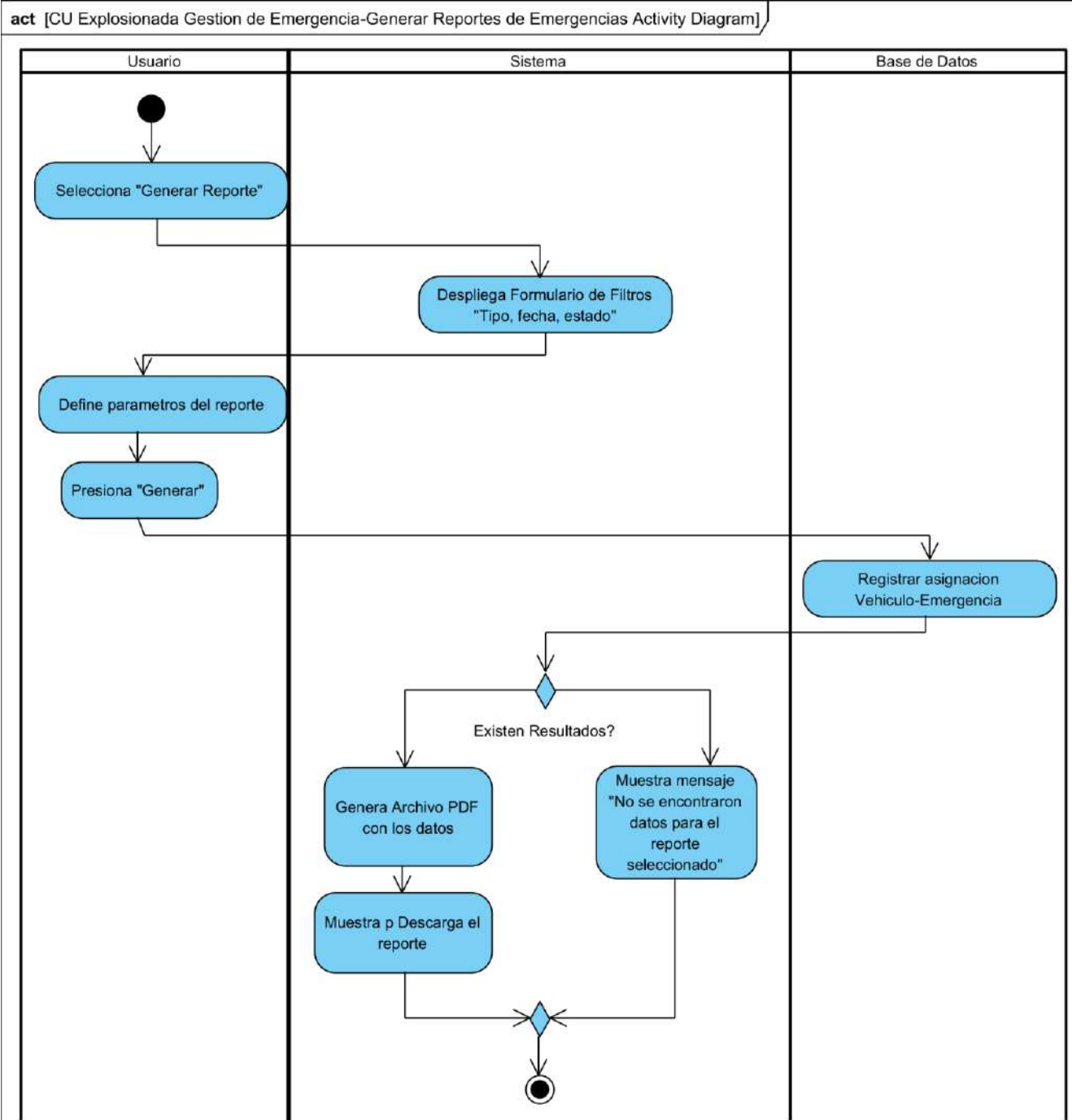


Figura 23. Diagrama de Actividades: Gestionar Gestión de Emergencias-Generar Reportes de Emergencias
Fuente: Elaboración Propia.

II.1.2.6.10.3 Diagrama de Actividades: Gestión de Credenciales QR

- Listar Credenciales QR

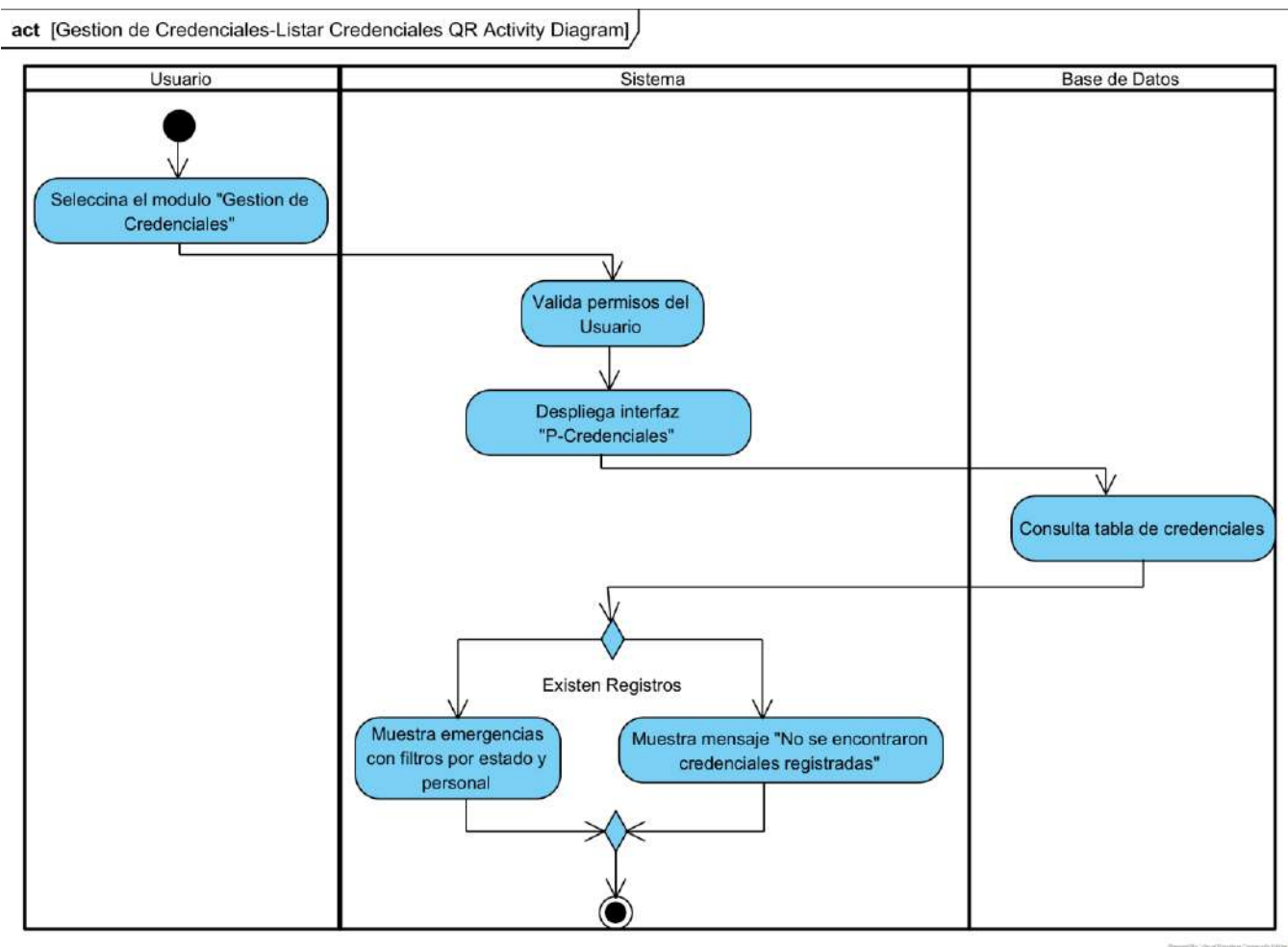


Figura 24. Gestionar Gestión de Credenciales-Listar credenciales QR
Fuente: Elaboración Propia.

- Registrar Credenciales

act [Gestion de Credenciales-Registrar Credencial Activity Diagram]

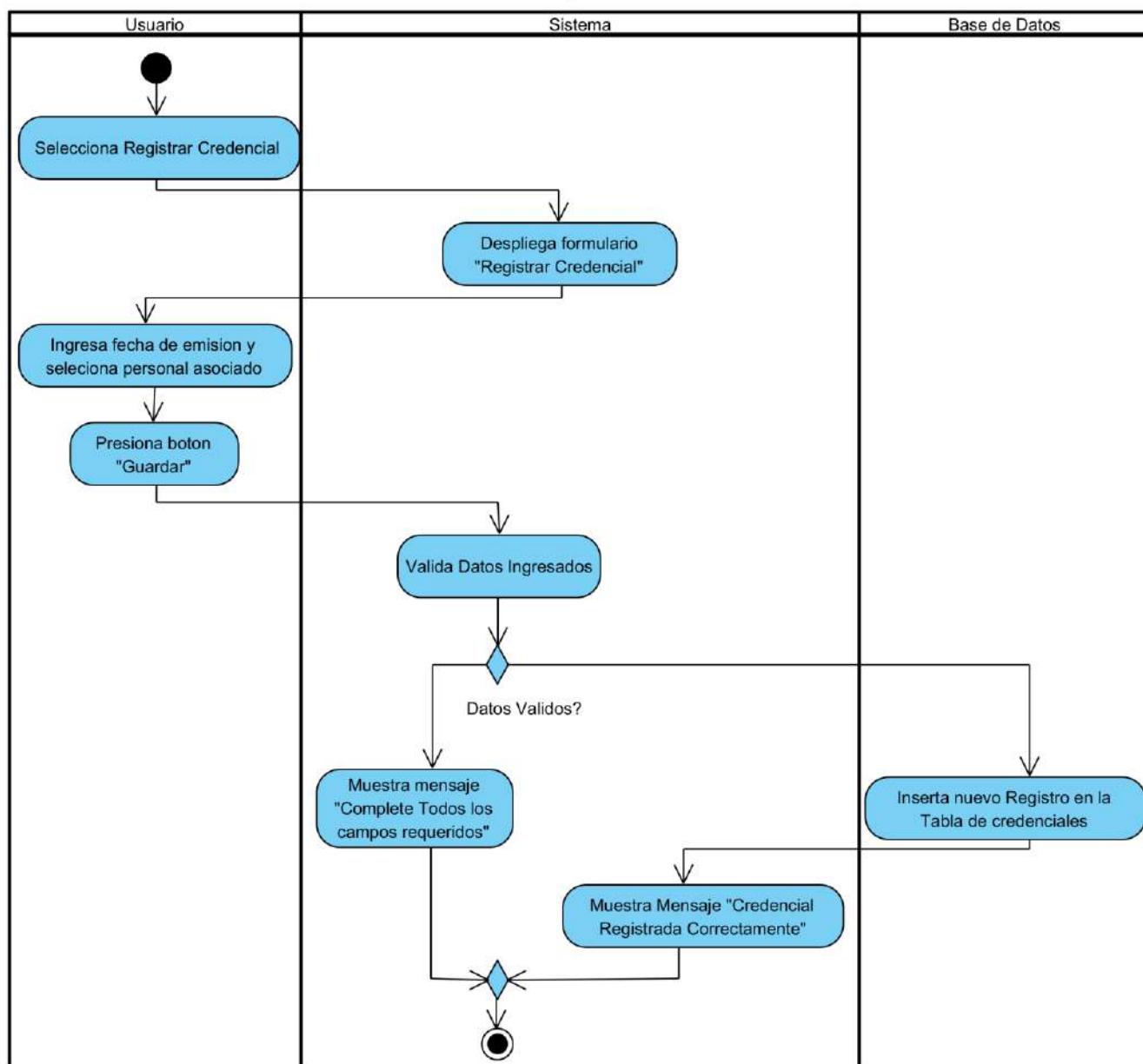


Figura 25. Gestionar Gestión de Credenciales-Registrar Credenciales
Fuente: Elaboración Propia.

- Actualizar / Suspender / Revocar Credencial

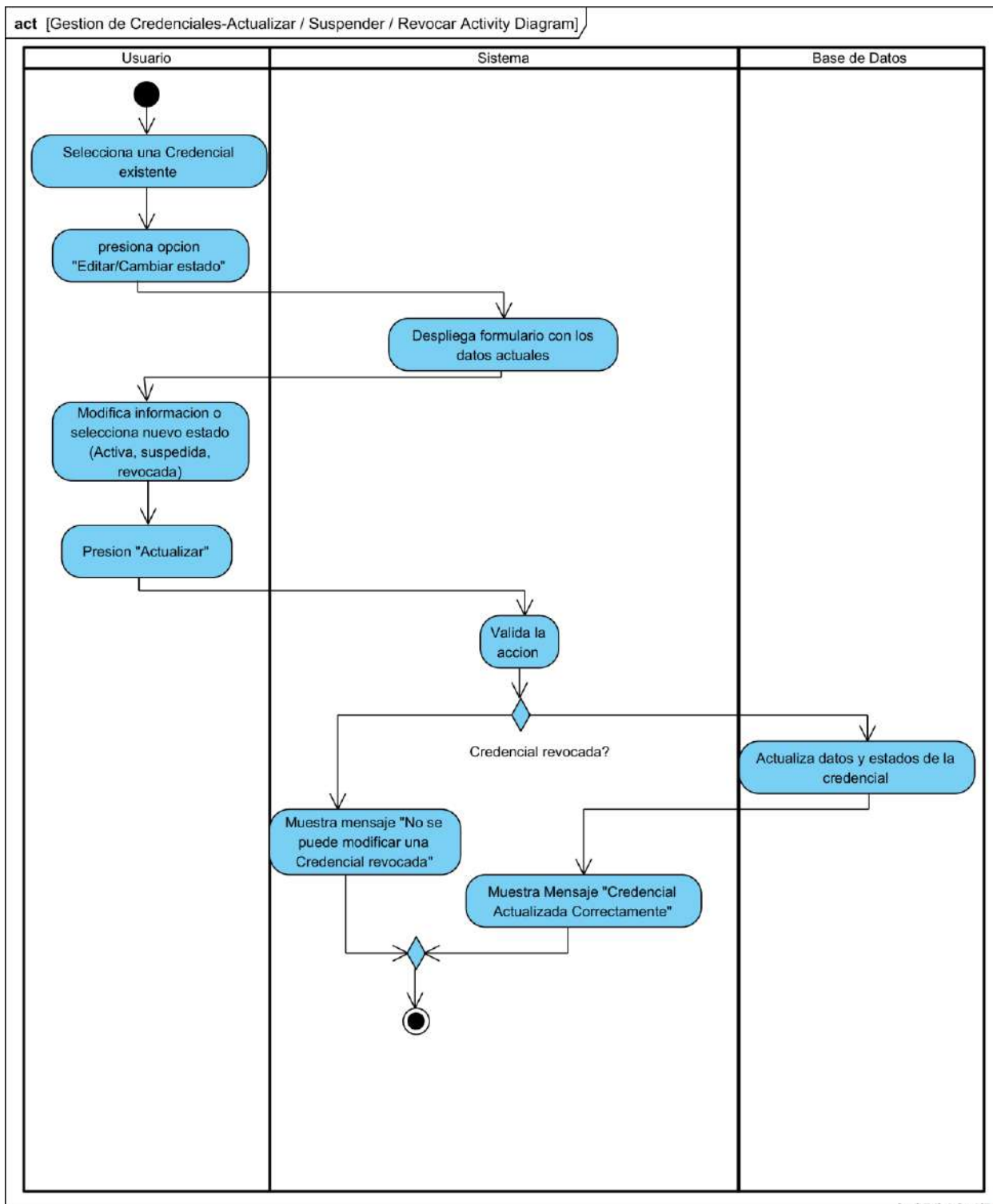


Figura 26. Gestionar Gestión de Credenciales-Actualizar/Suspender/Revocar Credencial
Fuente: Elaboración Propia.

- **Generar Credencial PDF con Código QR**

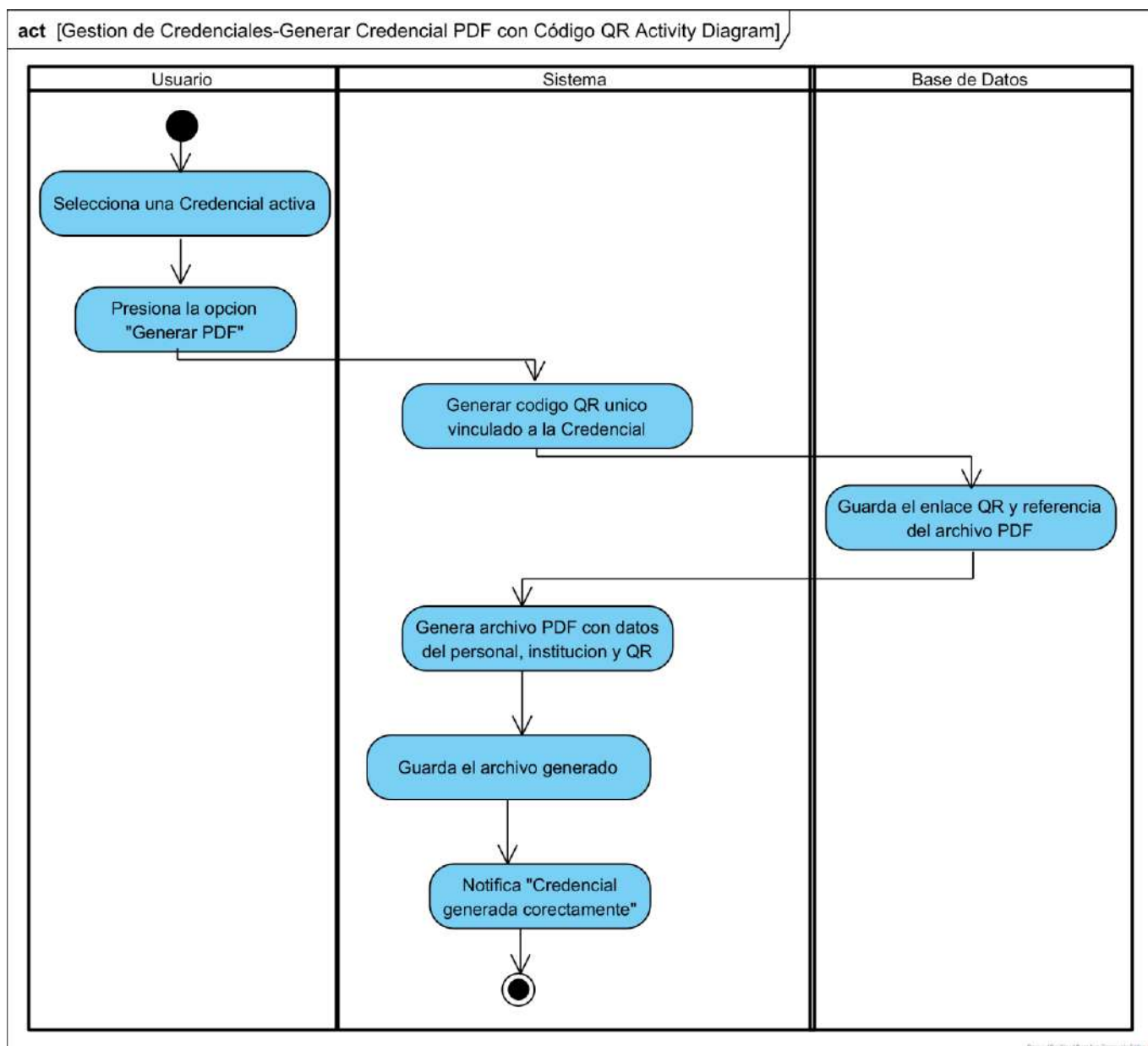


Figura 27. Gestionar Gestión de Credenciales-Generar Credencial PDF con código QR
Fuente: Elaboración Propia.

- **Verificar Credencial por Código QR (Acceso Público)**

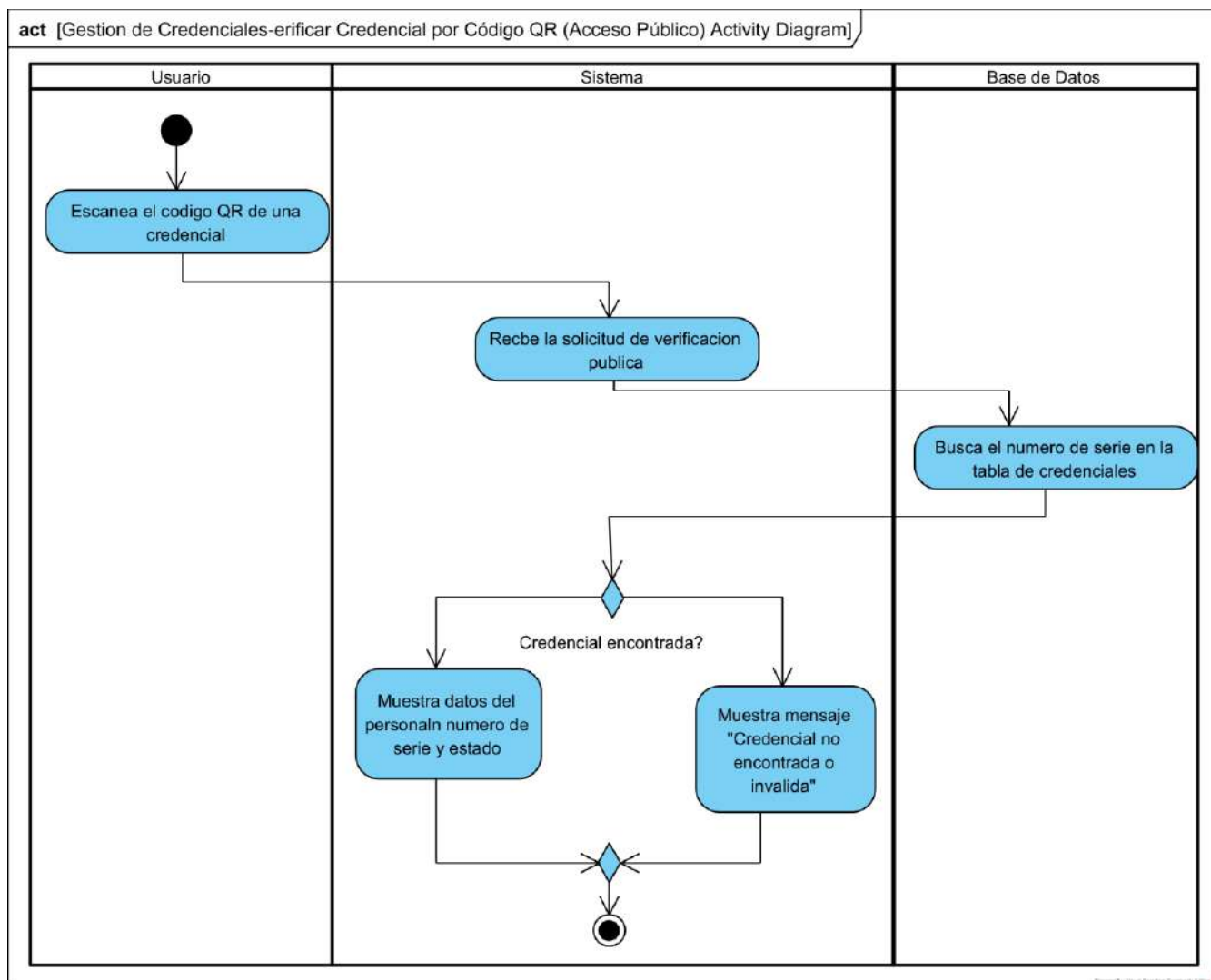


Figura 28. Gestionar Gestión de Credenciales-Verificar Credencial por Código QR (Acceso Público)
Fuente: Elaboración Propia.

- **Generar Reportes de Credenciales**

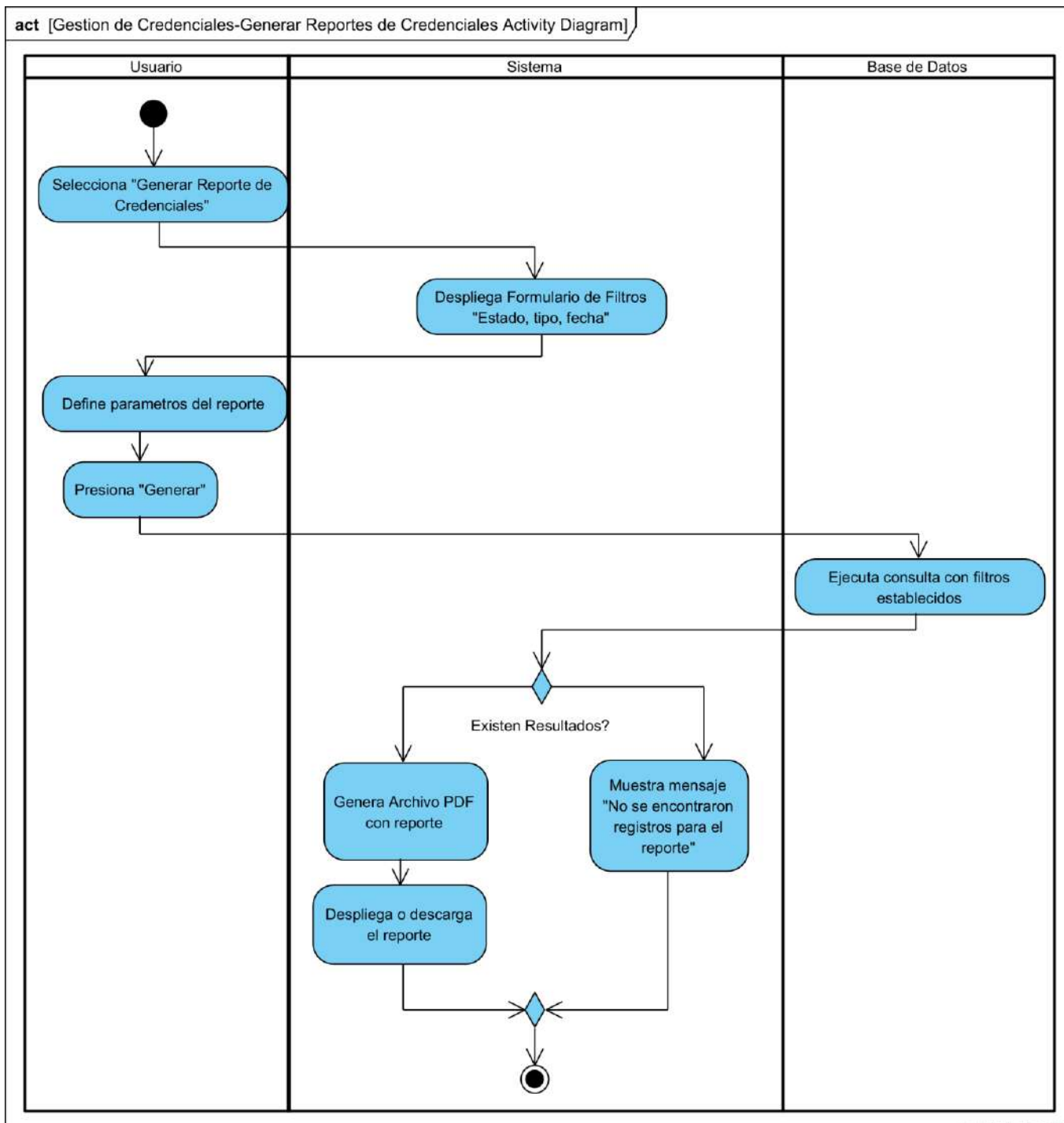


Figura 29. Gestionar Gestión de Credenciales-Generar Reportes de Credenciales
Fuente: Elaboración Propia.

II.1.2.6.10.4 Diagrama de Actividades: Gestión de Capacitaciones

- Listar Capacitaciones

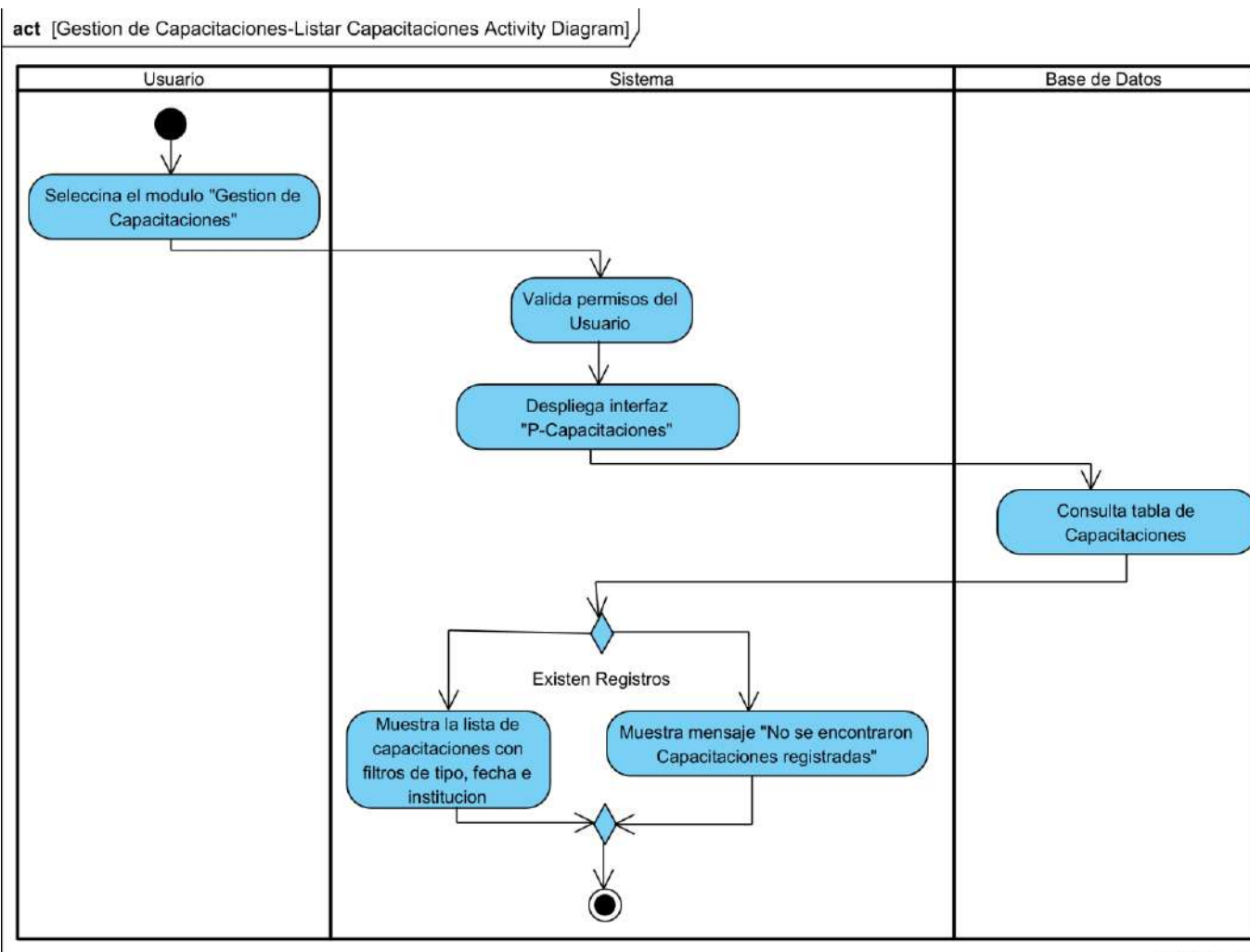


Figura 30. Gestionar Gestión de Capacitaciones-Listar Capacitaciones
Fuente: Elaboración Propia.

- **Registrar Capacitación**

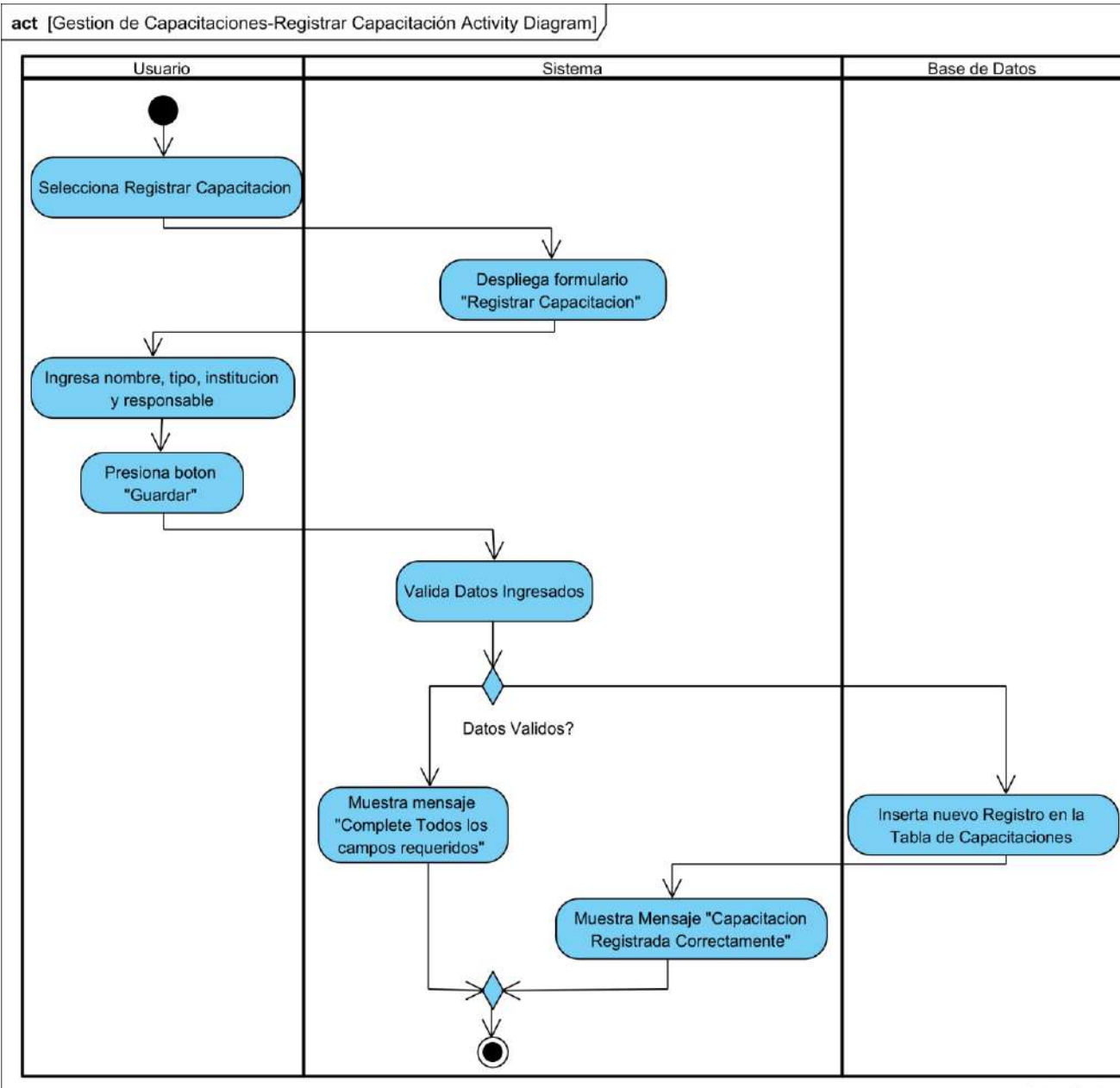


Figura 31. Gestionar Gestión de Capacitaciones-Registrar Capacitaciones
Fuente: Elaboración Propia.

- Registrar Institución Externa

act [Gestion de Capacitaciones-Registrar Institución Externa Activity Diagram]

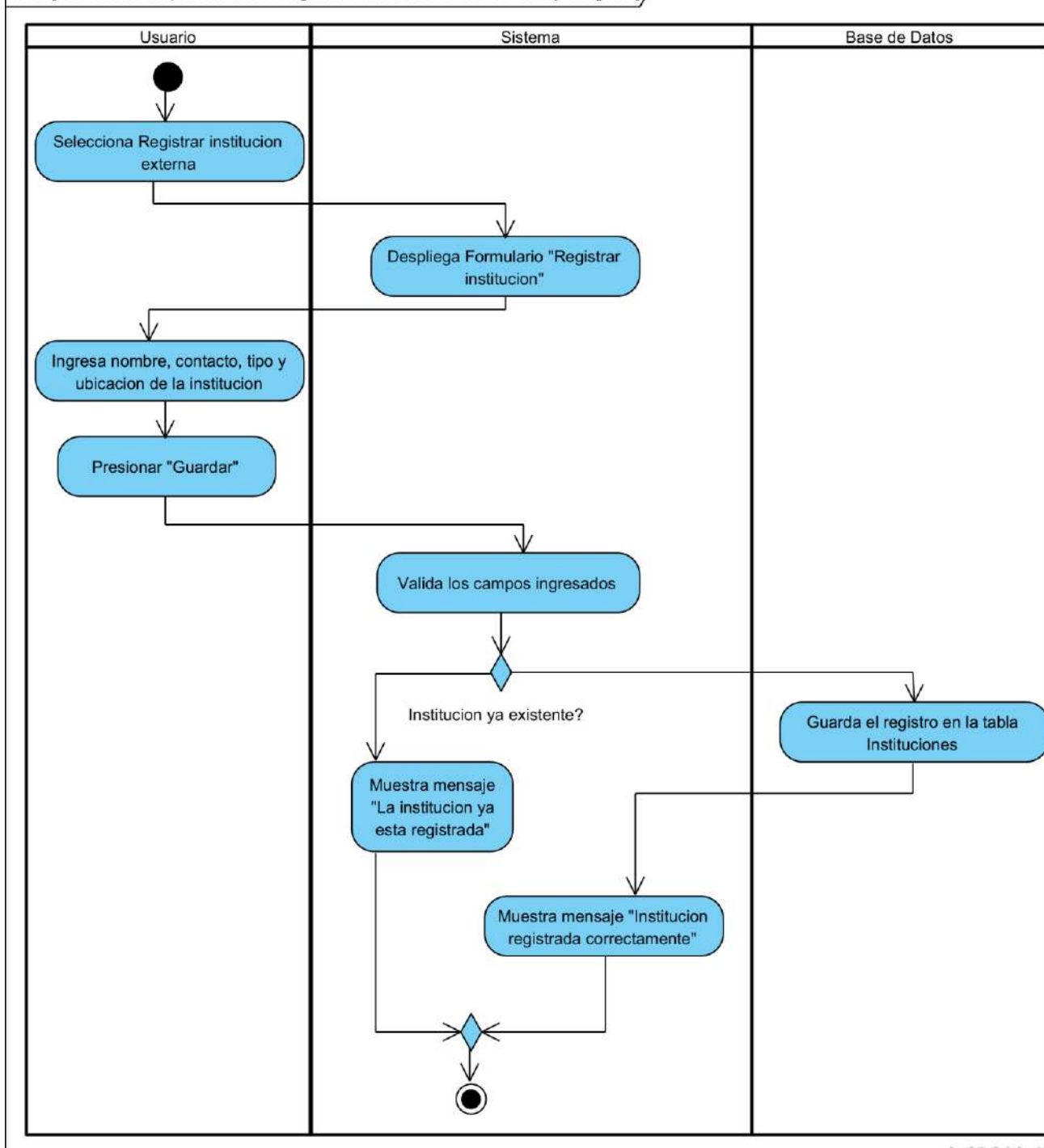


Figura 32. Gestionar Gestión de Capacitaciones-Registrar Institución Externa
Fuente: Elaboración Propia.

- **Registrar Participantes (Internos / Externos)**

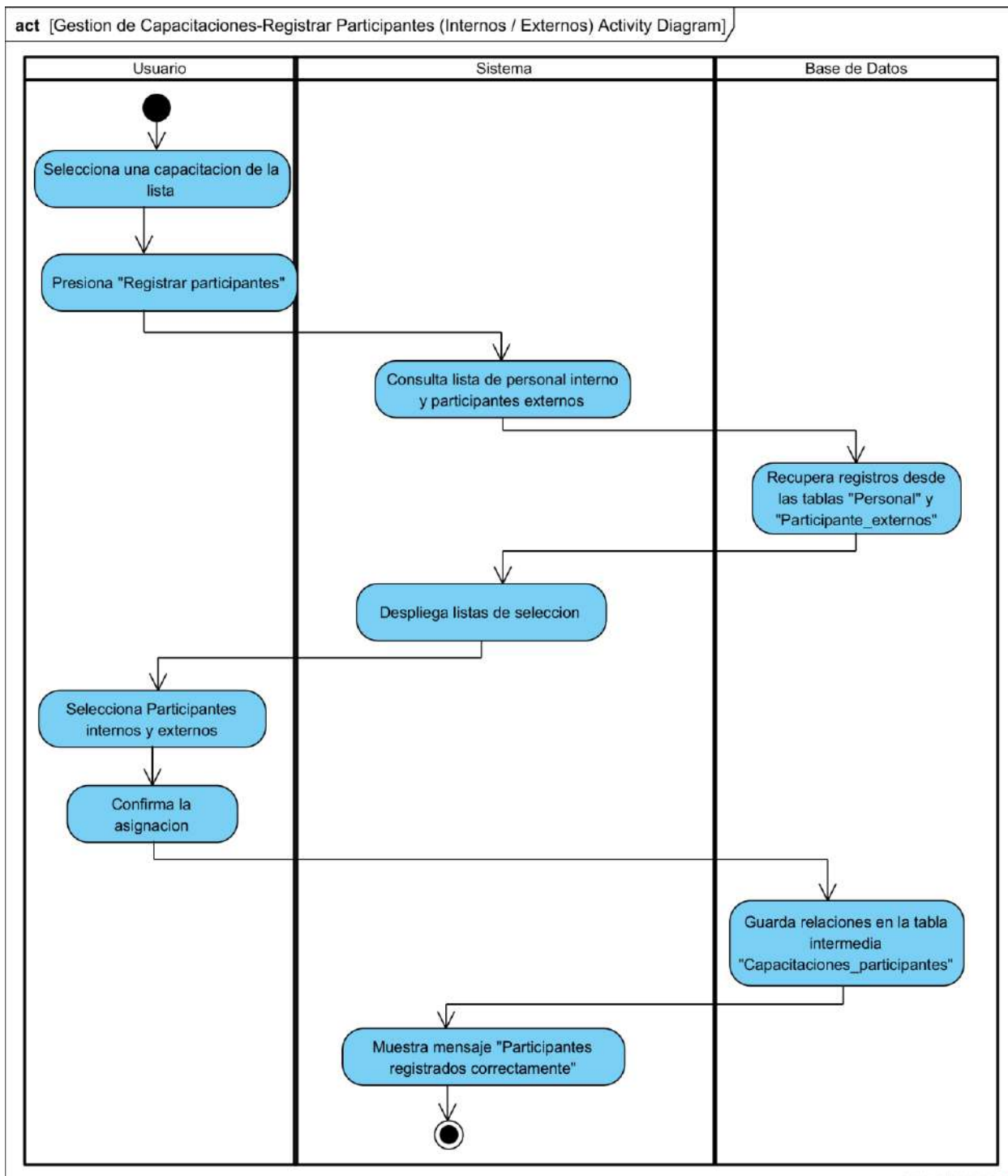


Figura 33. Gestionar Gestión de Capacitaciones-Registrar Participantes (Internos-Externos)
Fuente: Elaboración Propia.

- **Generar Certificados PDF con Código QR**

act [Gestion de Capacitaciones-Generar Certificados PDF con Código QR Activity Diagram]

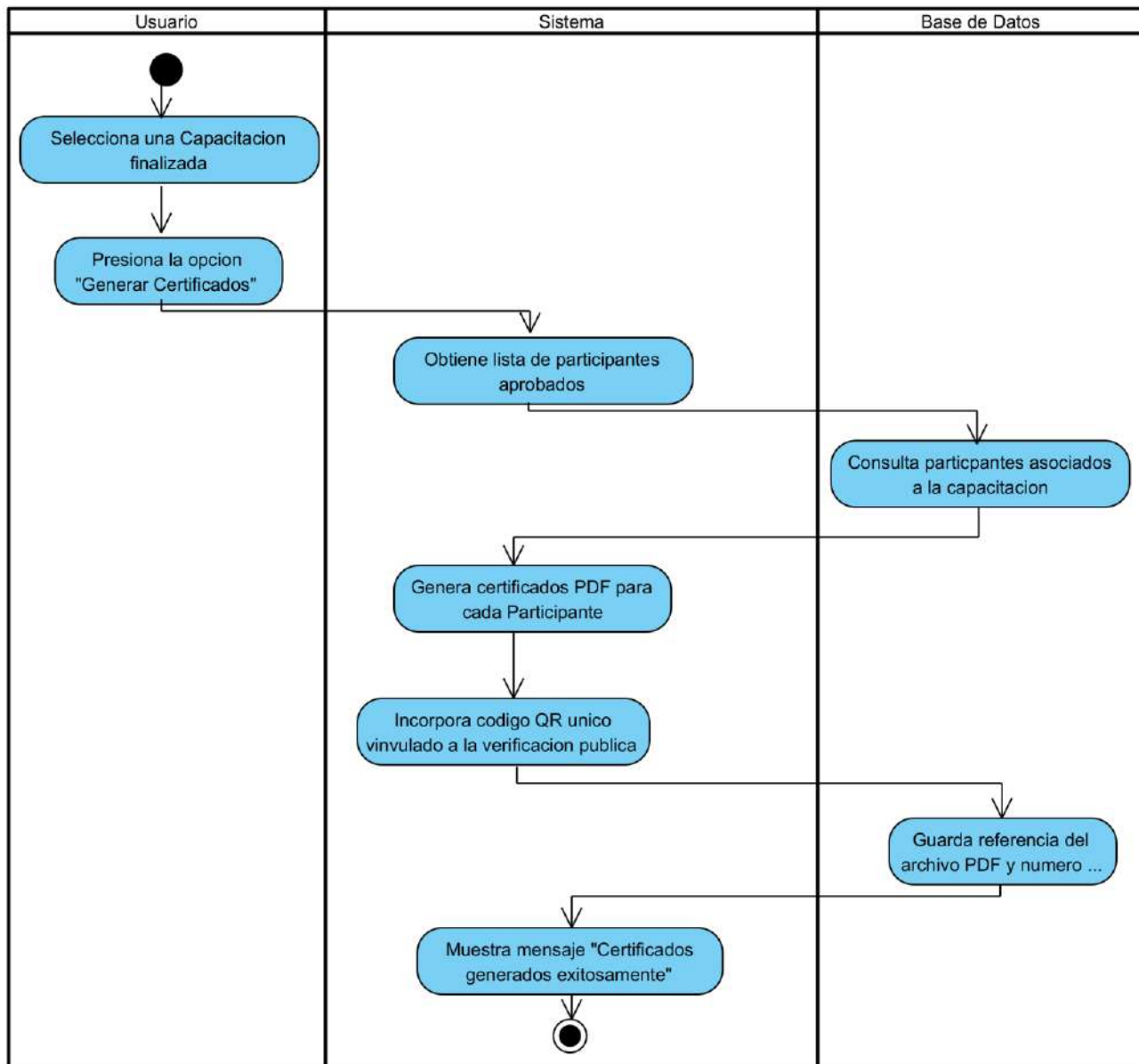


Figura 34. Gestionar Gestión de Capacitaciones-Generar Certificados PDF con código QR
Fuente: Elaboración Propia.

- **Generar Reportes y Estadísticas**

act [Gestion de Capacitaciones-Generar Reportes y Estadísticas Activity Diagram]

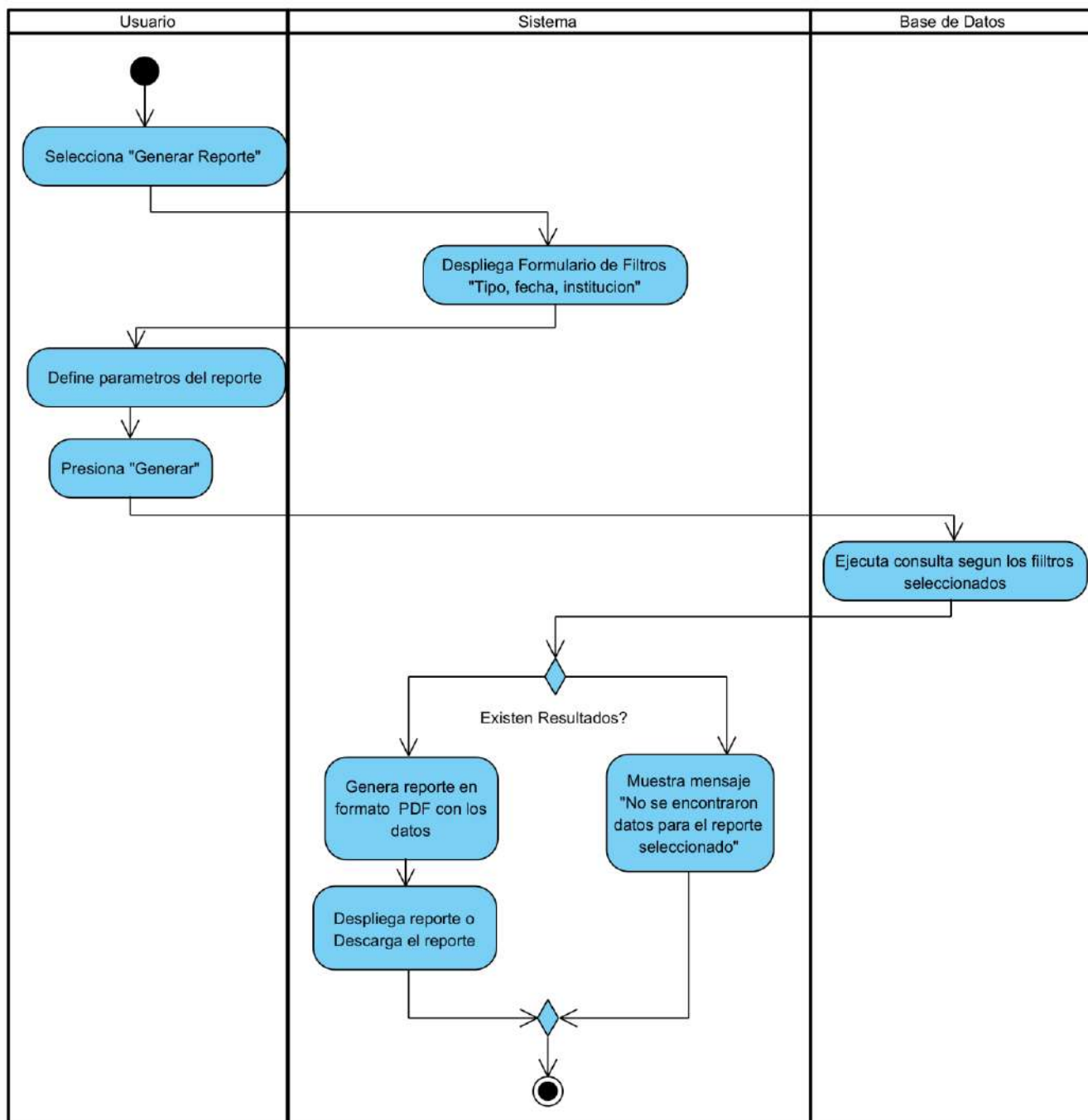


Figura 35. Gestionar Gestión de Capacitaciones-Generar reportes y Estadísticas
Fuente: Elaboración Propia.

II.1.2.6.10.5 Diagrama de Actividades: Gestión de Vehículos

- Listar Vehículos

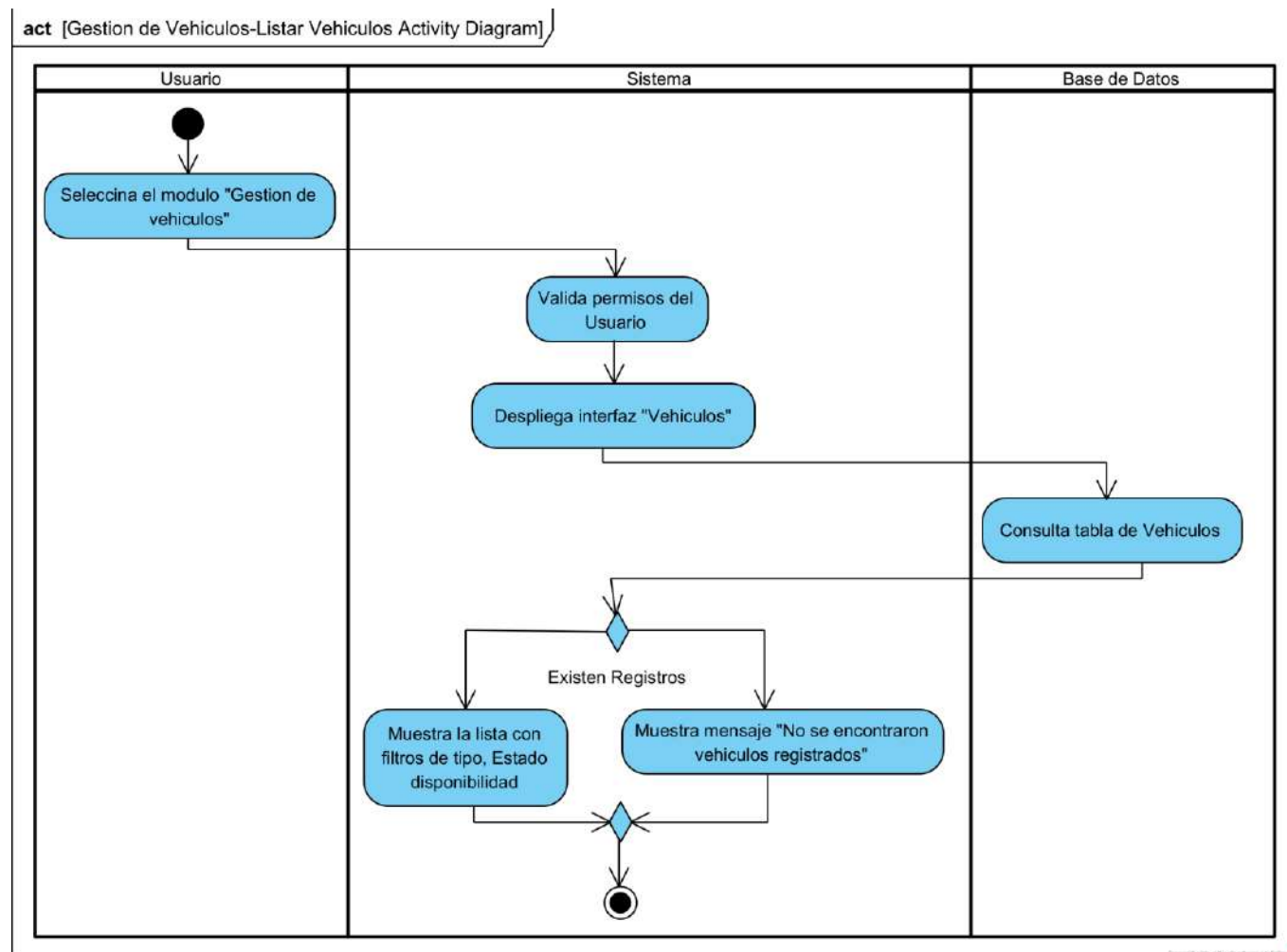


Figura 36. Gestionar Gestión de Vehículos-Listar Vehículos
Fuente: Elaboración Propia.

- Registrar Vehículos

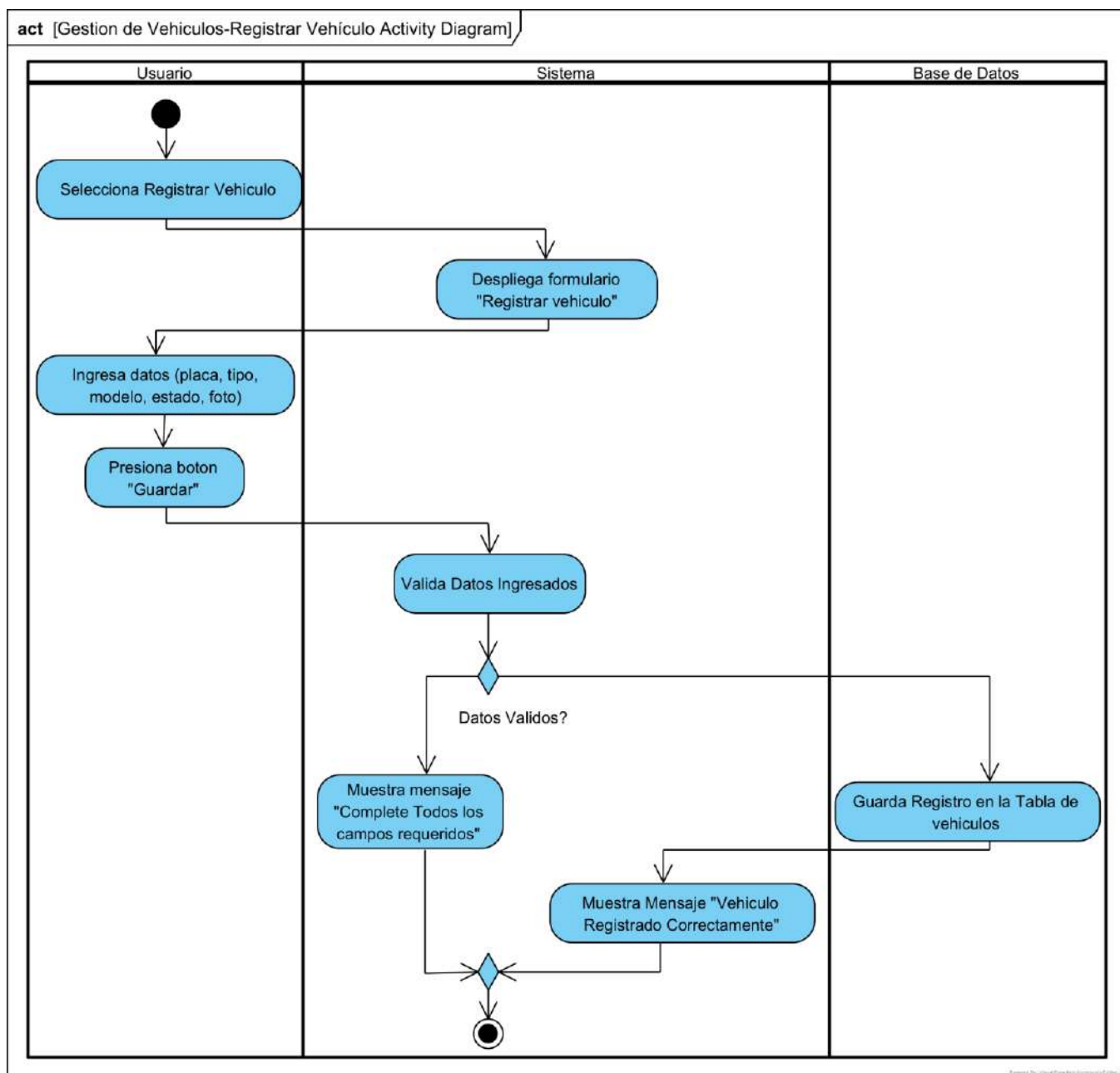


Figura 37. Gestionar Gestión de Vehículos-Registra Vehículos
Fuente: Elaboración Propia.

- **Editar / Actualizar Vehículos**

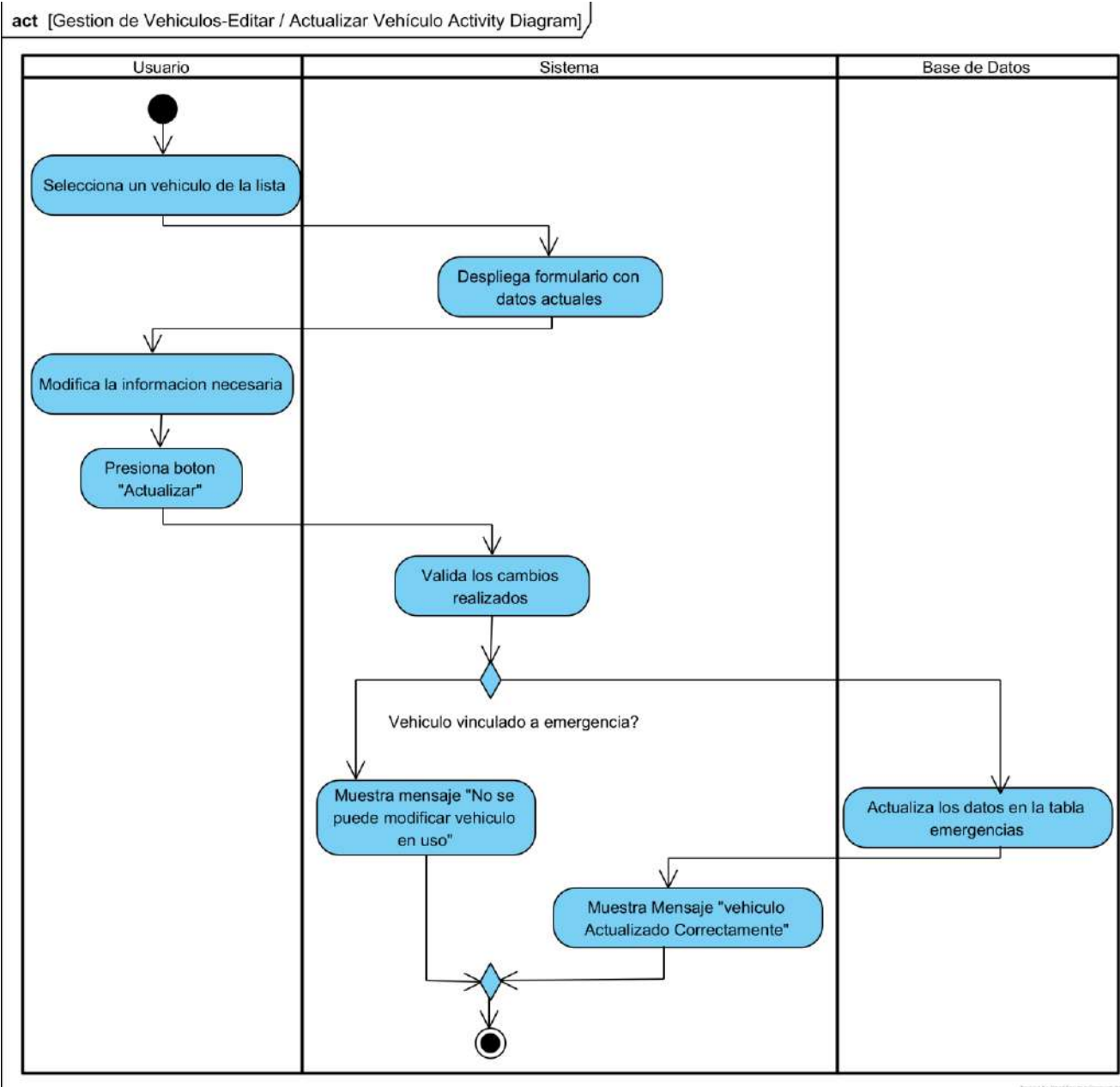


Figura 38. Gestionar Gestión de Vehículos-Editar/Actualizar Vehículos
Fuente: Elaboración Propia.

- **Cambiar Estado Operativo**

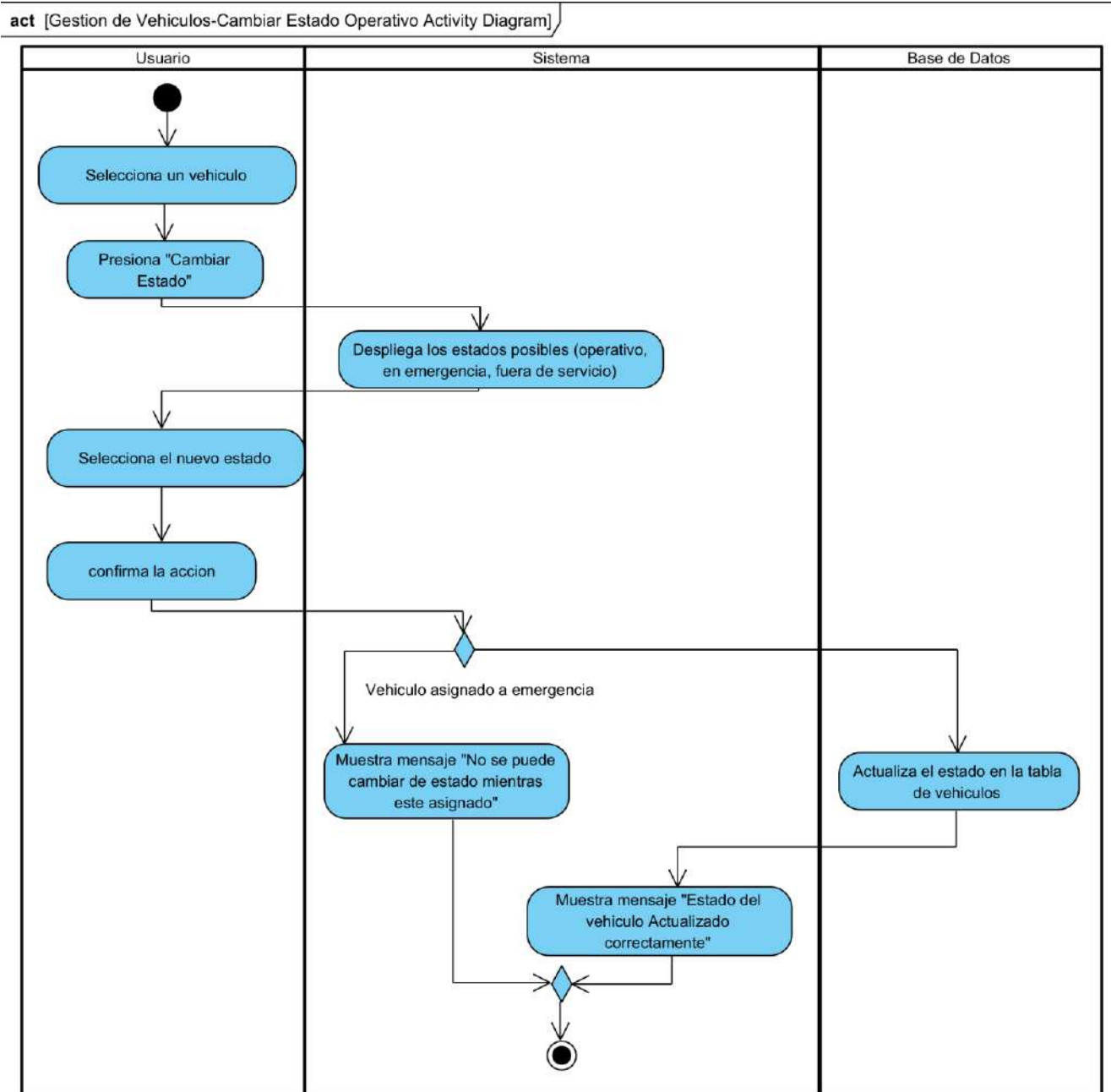


Figura 39. Gestionar Gestión de Vehículos-Cambiar Estado Operativo
Fuente: Elaboración Propia.

- Registrar Mantenimiento

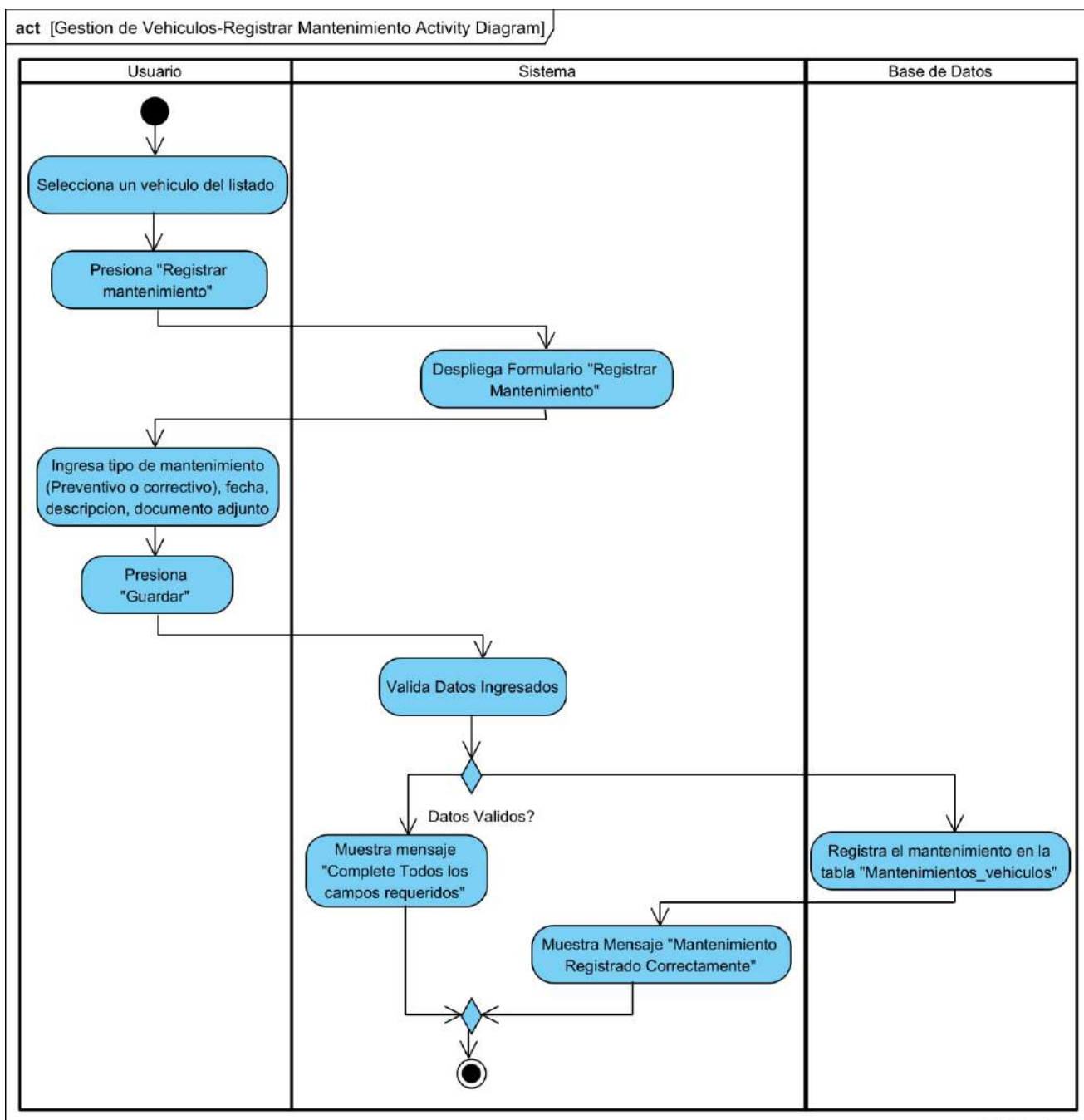


Figura 40. Gestionar Gestión de Vehículos-Registrar Mantenimiento
Fuente: Elaboración Propia.

- Eliminar Vehículo

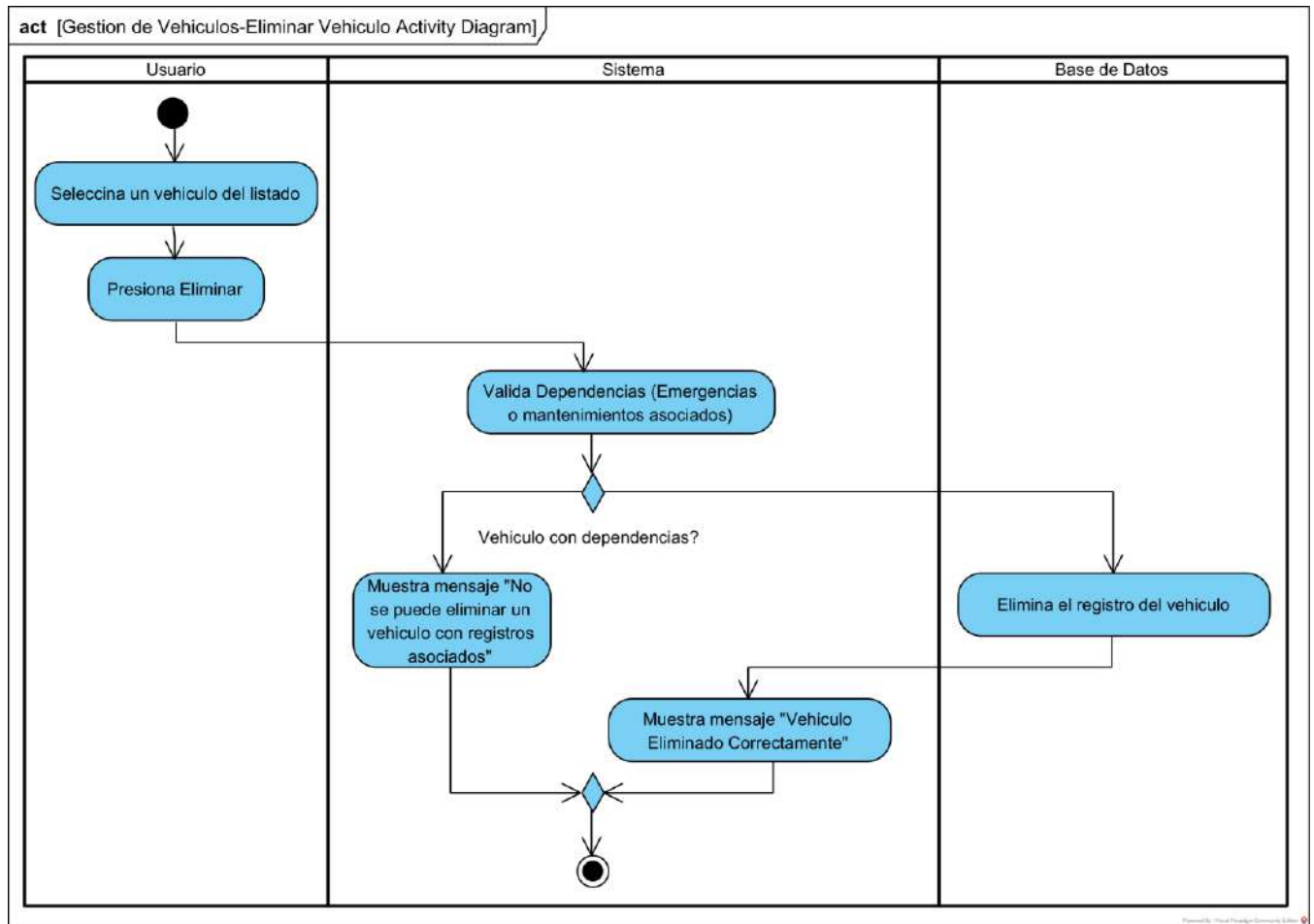


Figura 41. Gestionar Gestión de Vehículos-Eliminar Vehículo
Fuente: Elaboración Propia.

- **Generar Reportes de Vehículo**

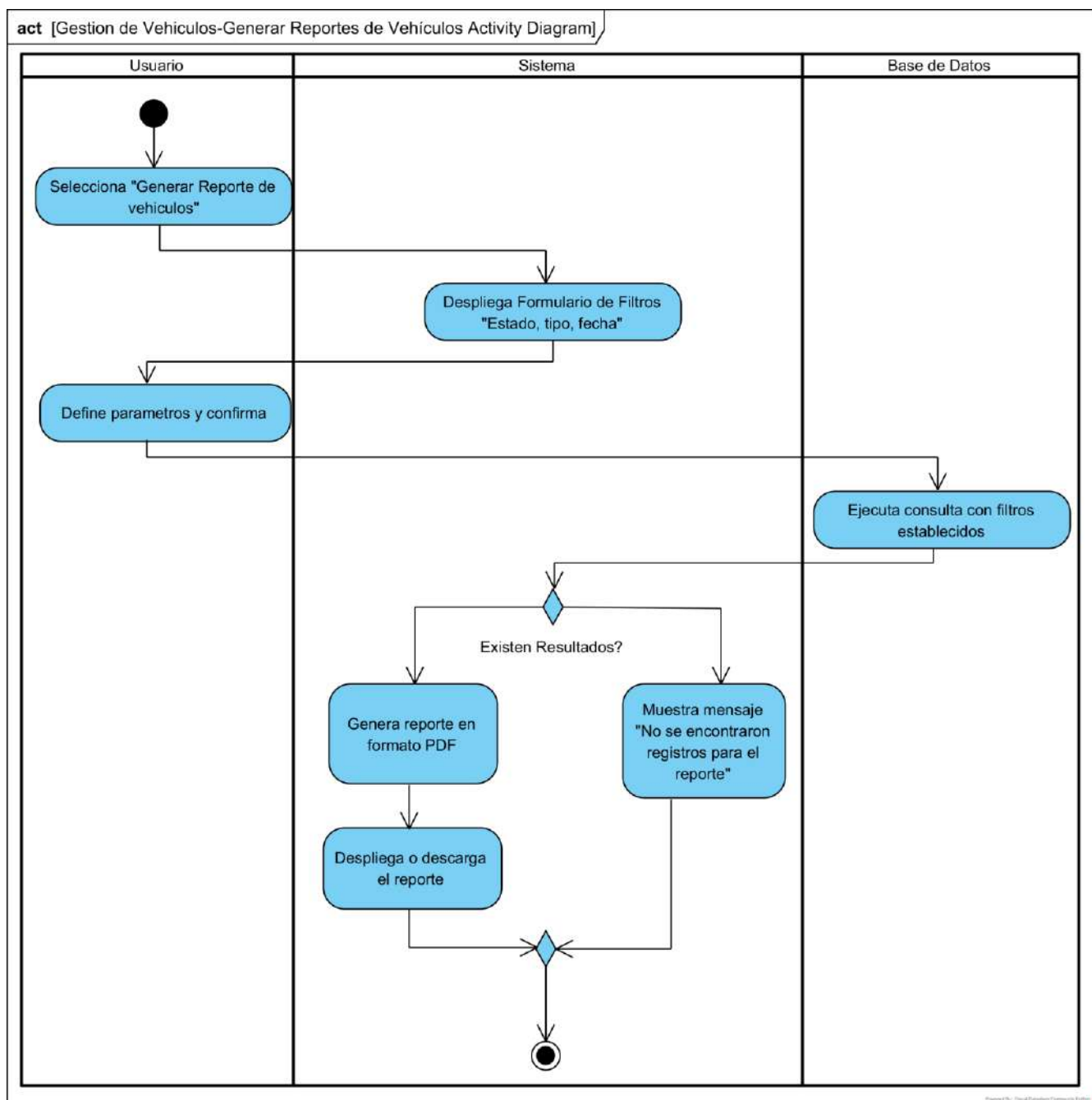


Figura 42. Gestionar Gestión de Vehículos-Generar Reportes de Vehículo
Fuente: Elaboración Propia.

II.1.2.6.10.6 Diagrama de Actividades: Gestión de Personal

- Listar Personal

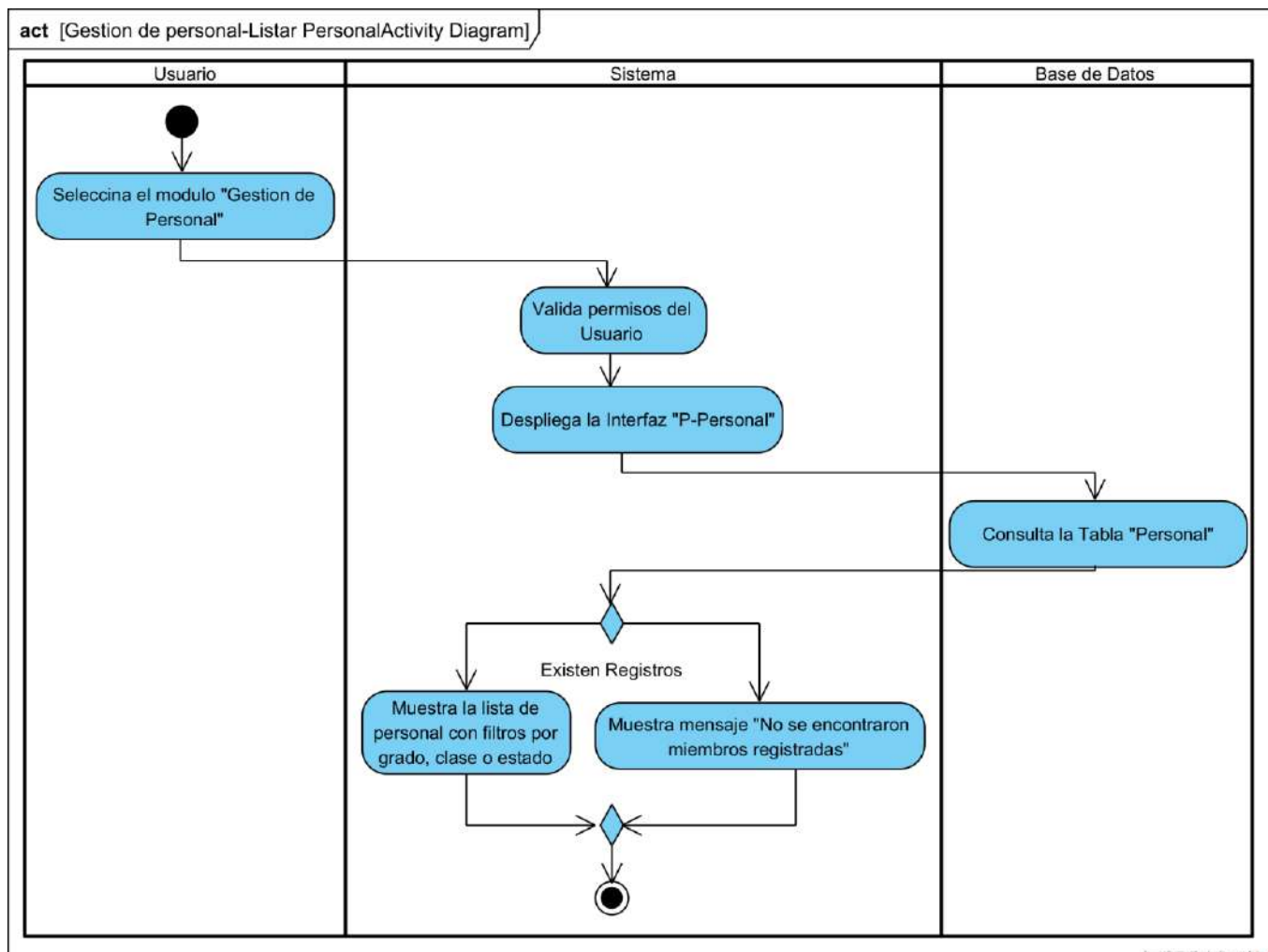


Figura 43. Gestionar Gestión de Personal-Listar Personal
Fuente: Elaboración Propia.

- Registrar Personal

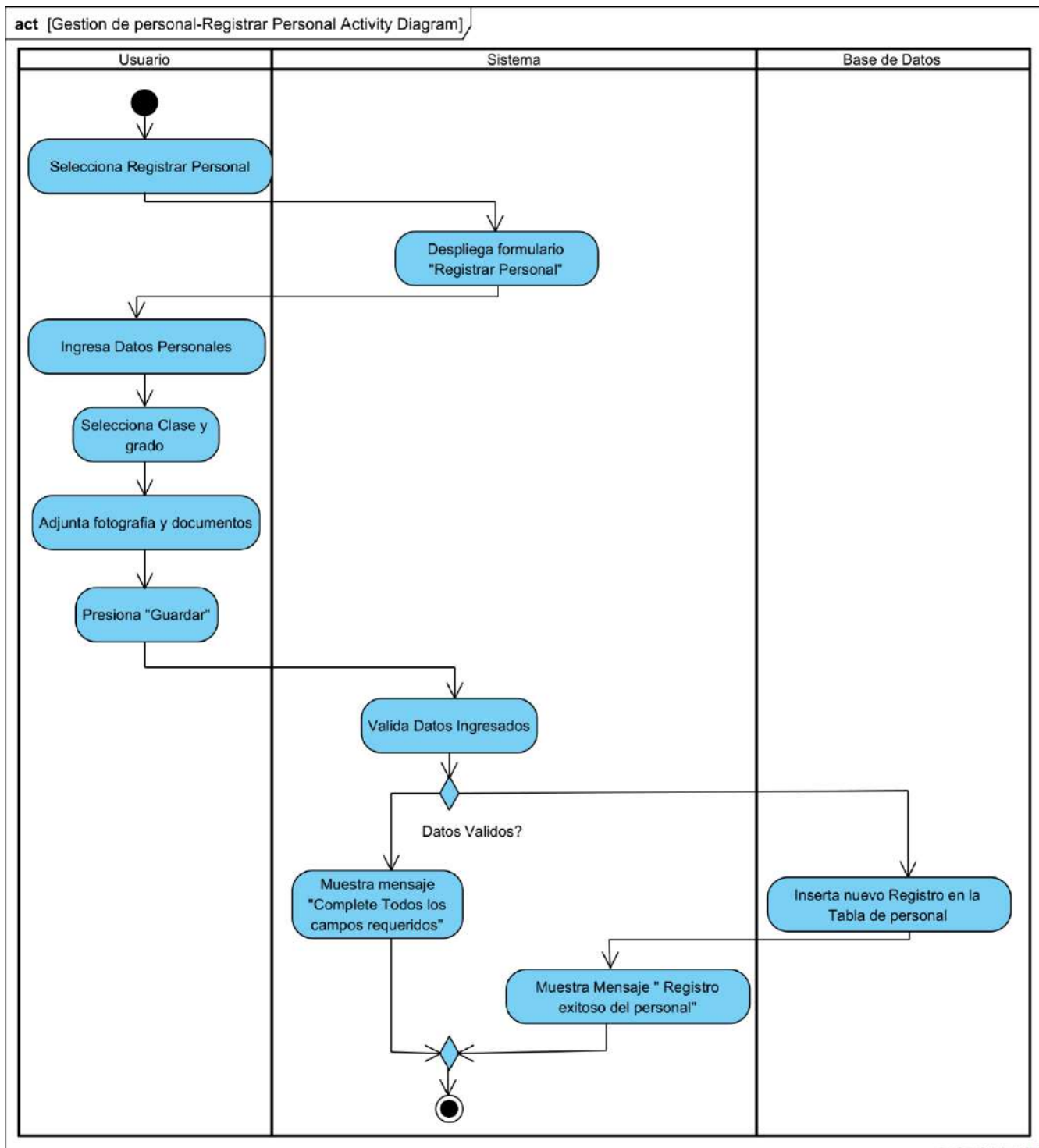


Figura 44. Gestionar Gestión de Personal-Registrar Personal
Fuente: Elaboración Propia.

- **Editar / Actualizar Información de Personal**

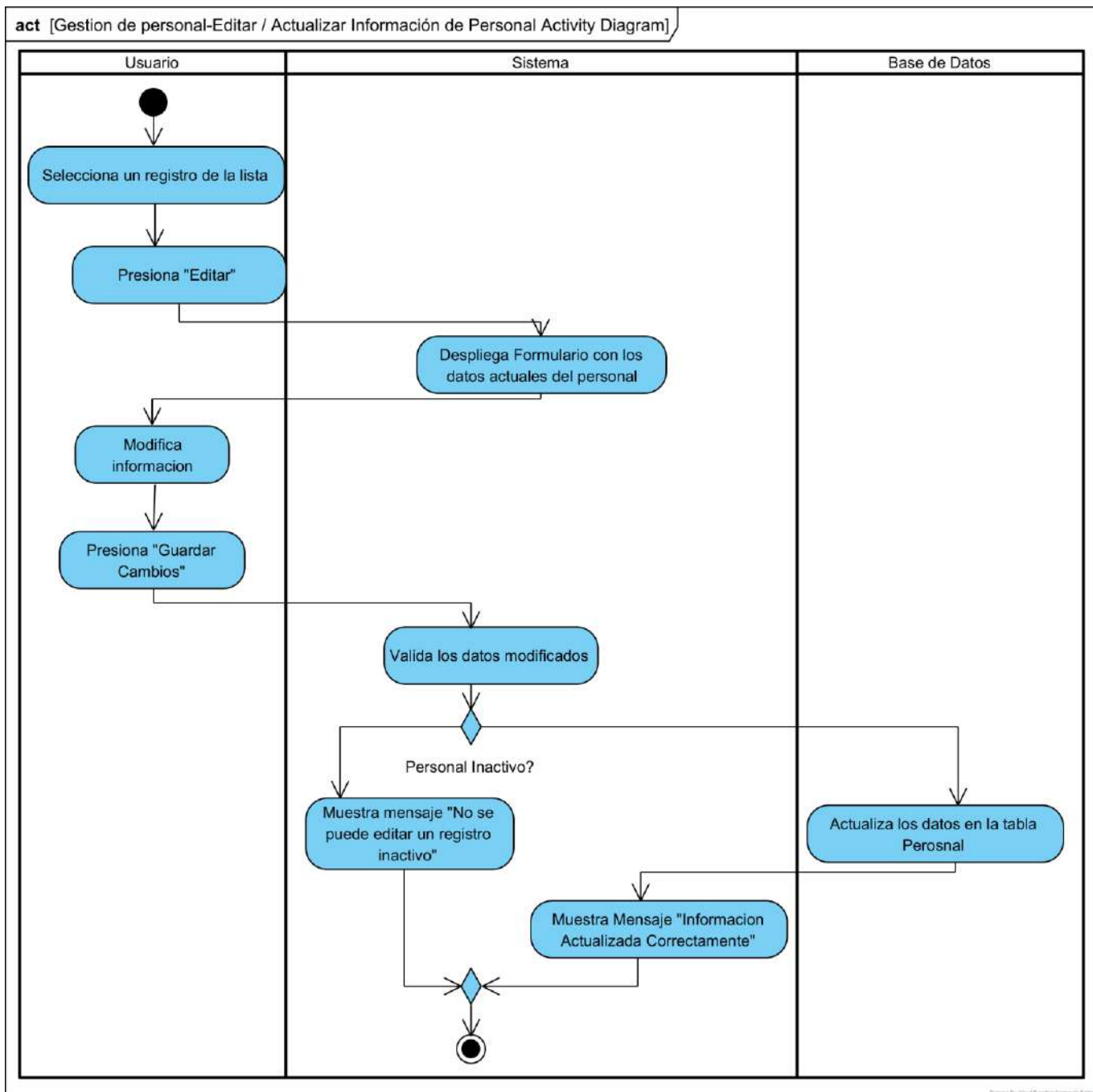


Figura 45. Gestionar Gestión de Personal-Aditar/Actualizar Información de Personal
Fuente: Elaboración Propia.

- **Asignar Grado y Clase**

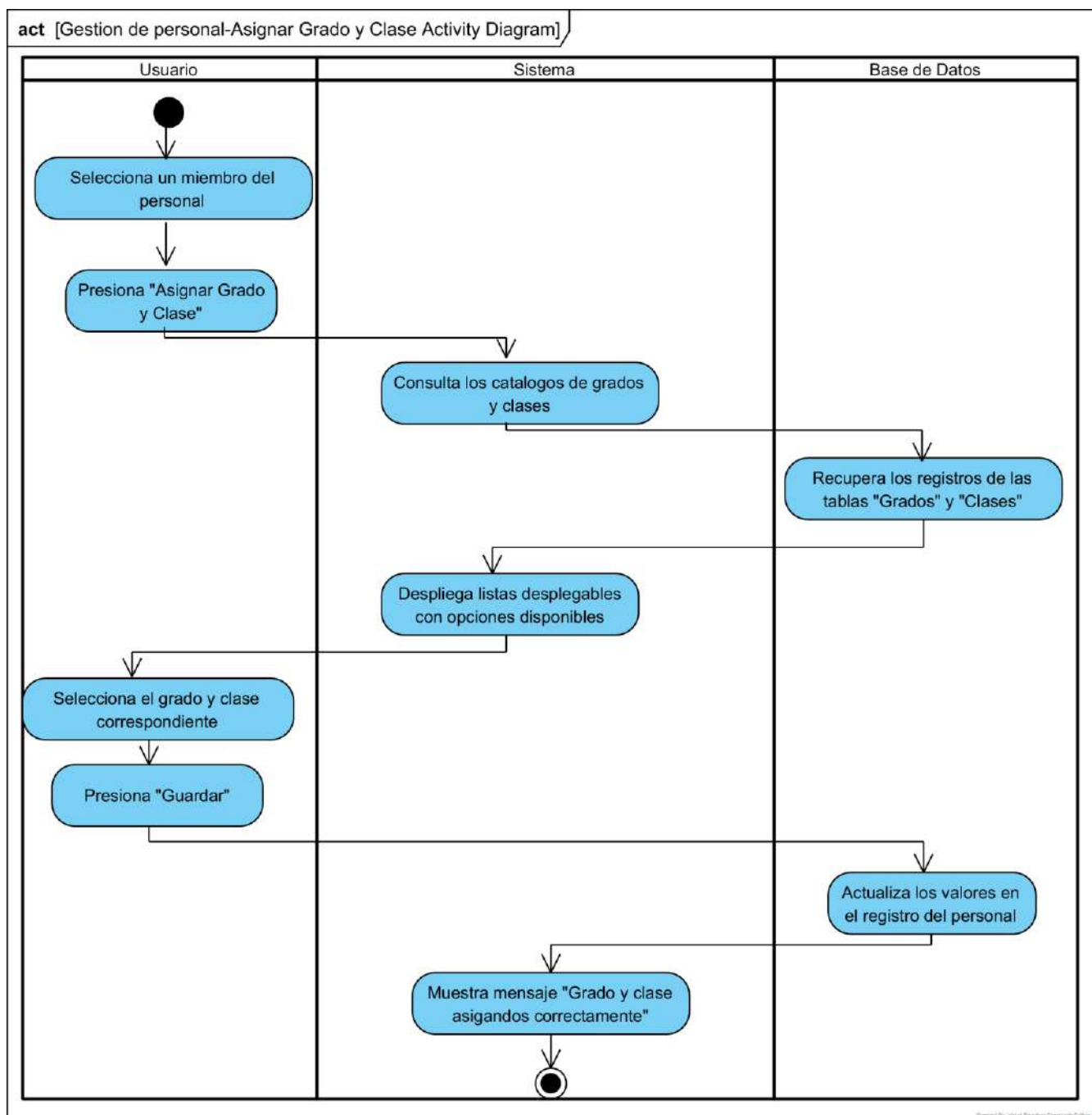


Figura 46. Gestionar Gestión de Personal-Asignar Grado y Clase
Fuente: Elaboración Propia.

- **Cambiar el Estado del Personal**

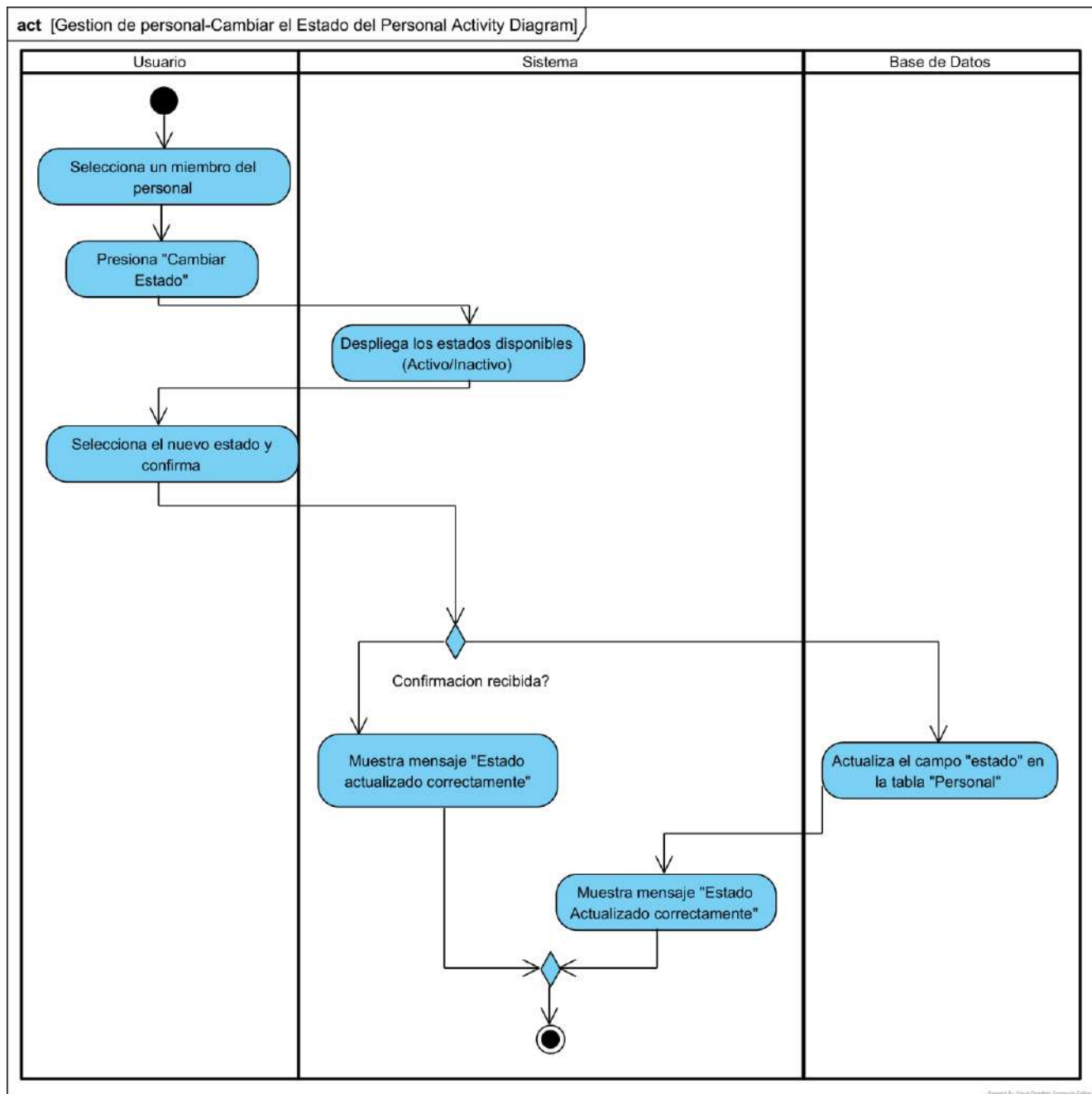


Figura 47. Gestionar Gestión de Personal-Cambiar Estado del Personal
Fuente: Elaboración Propia.

- **Generar Reportes del Personal**

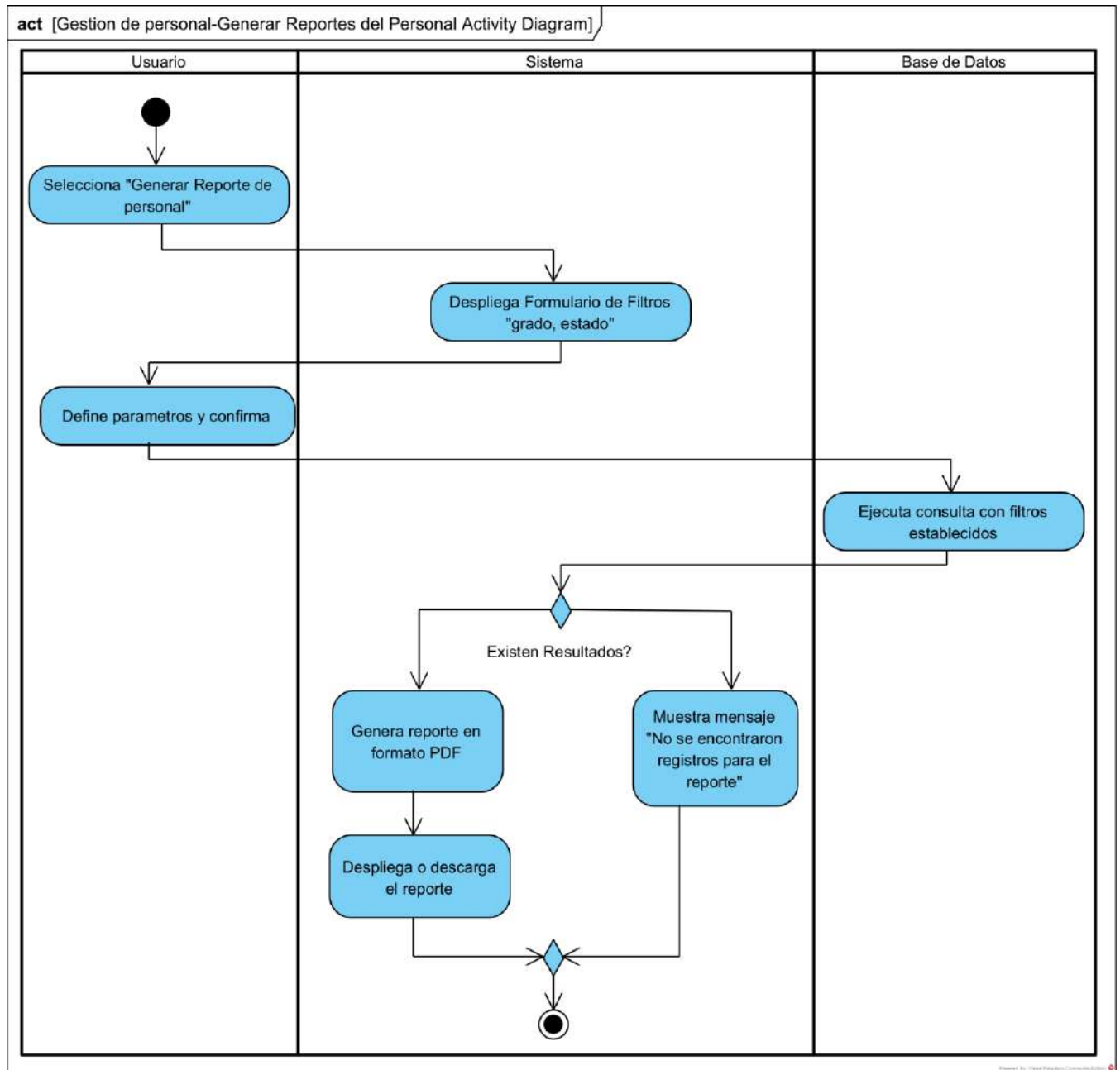


Figura 48. Gestionar Gestión de Personal-Generar Reportes del Personal
Fuente: Elaboración Propia.

II.1.2.6.10.7 Diagrama de Actividades: Gestión de Inventario

- Listar Ítems de Inventario

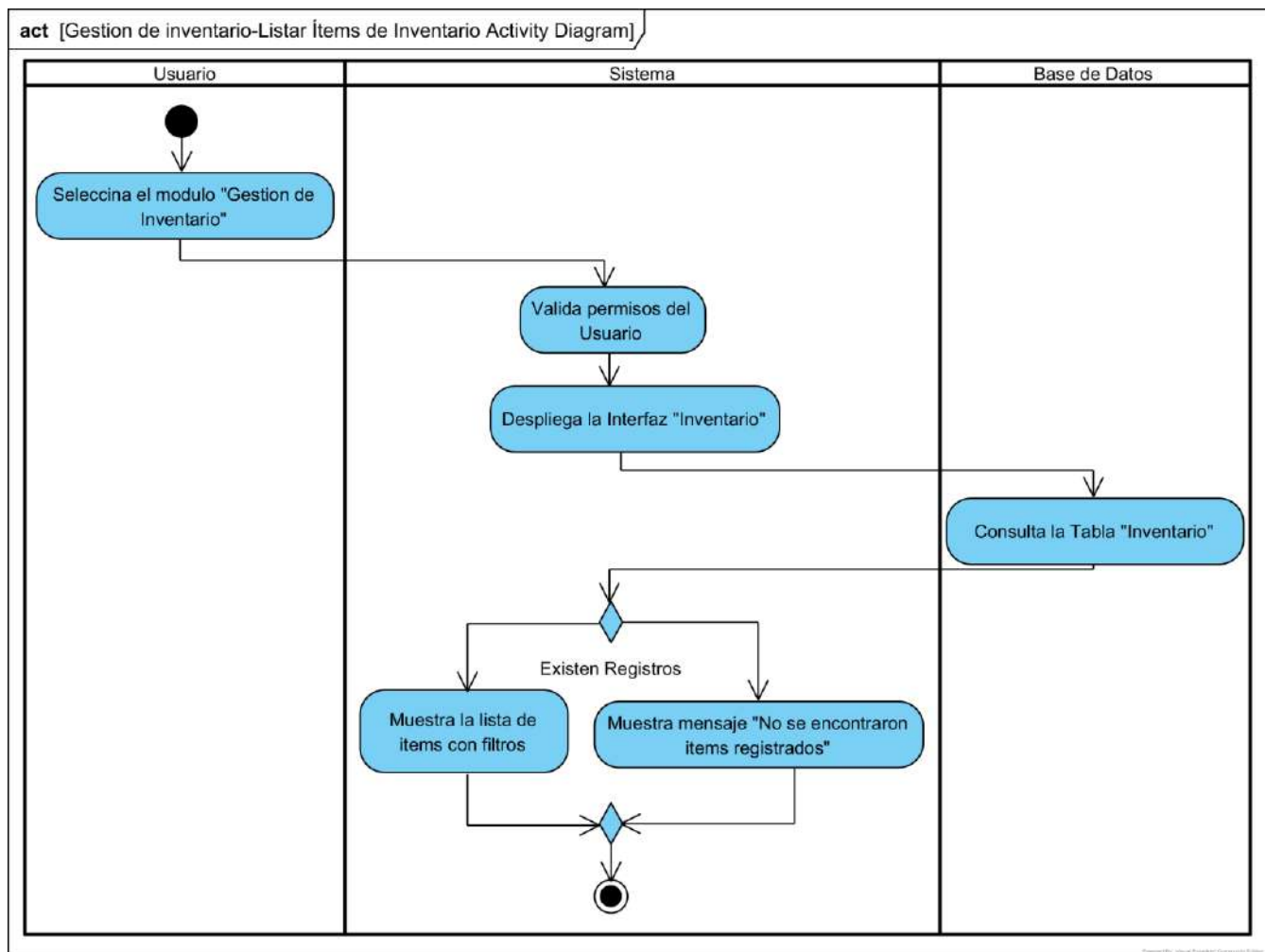


Figura 49. Gestionar Gestión de Inventario-Listar Ítems de Inventario
Fuente: Elaboración Propia.

- Registrar Ítem de Inventario

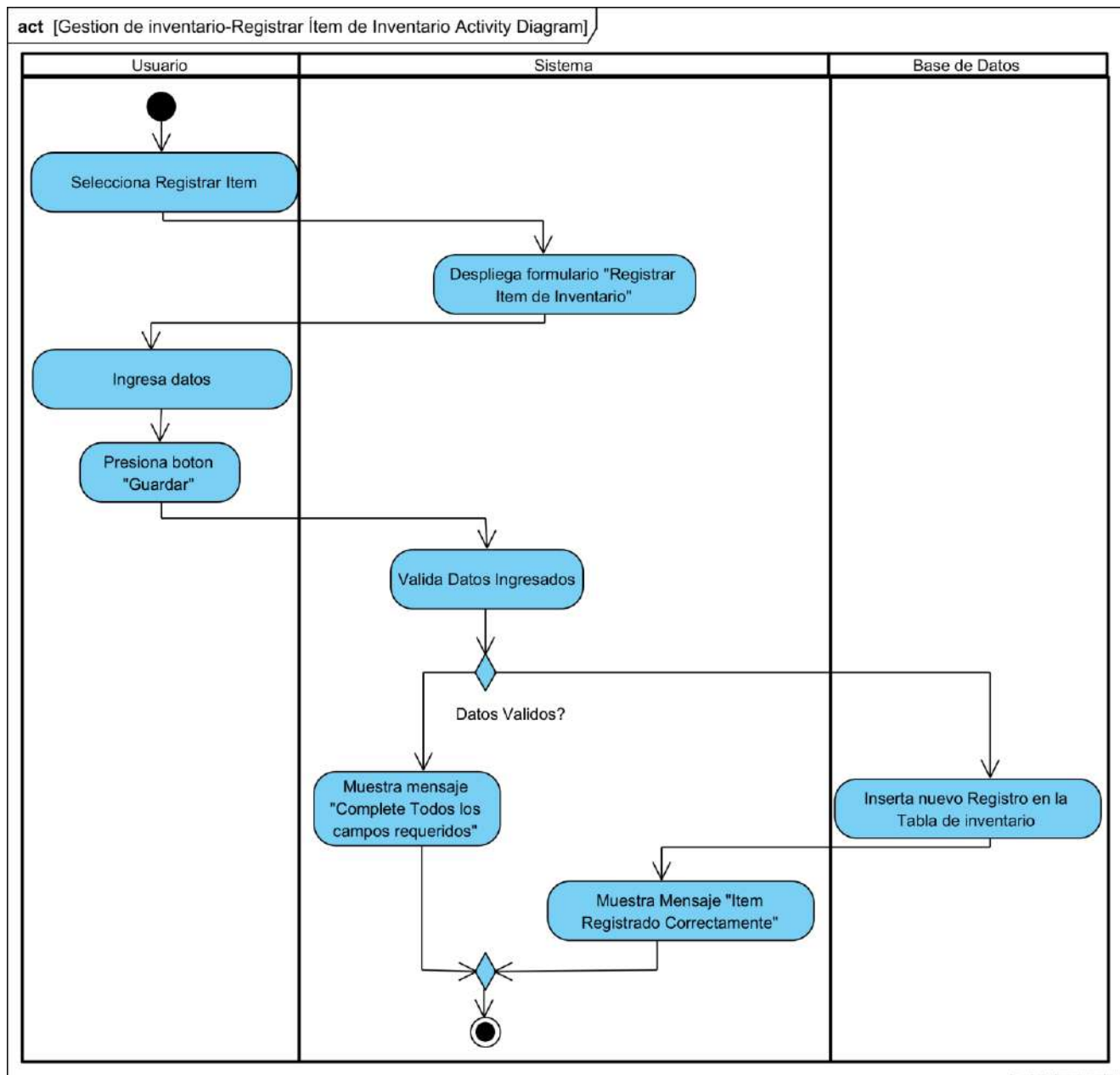


Figura 50. Gestionar Gestión de Inventario-Registrar Ítem de Inventario
Fuente: Elaboración Propia.

- **Editar / Actualizar Ítem**

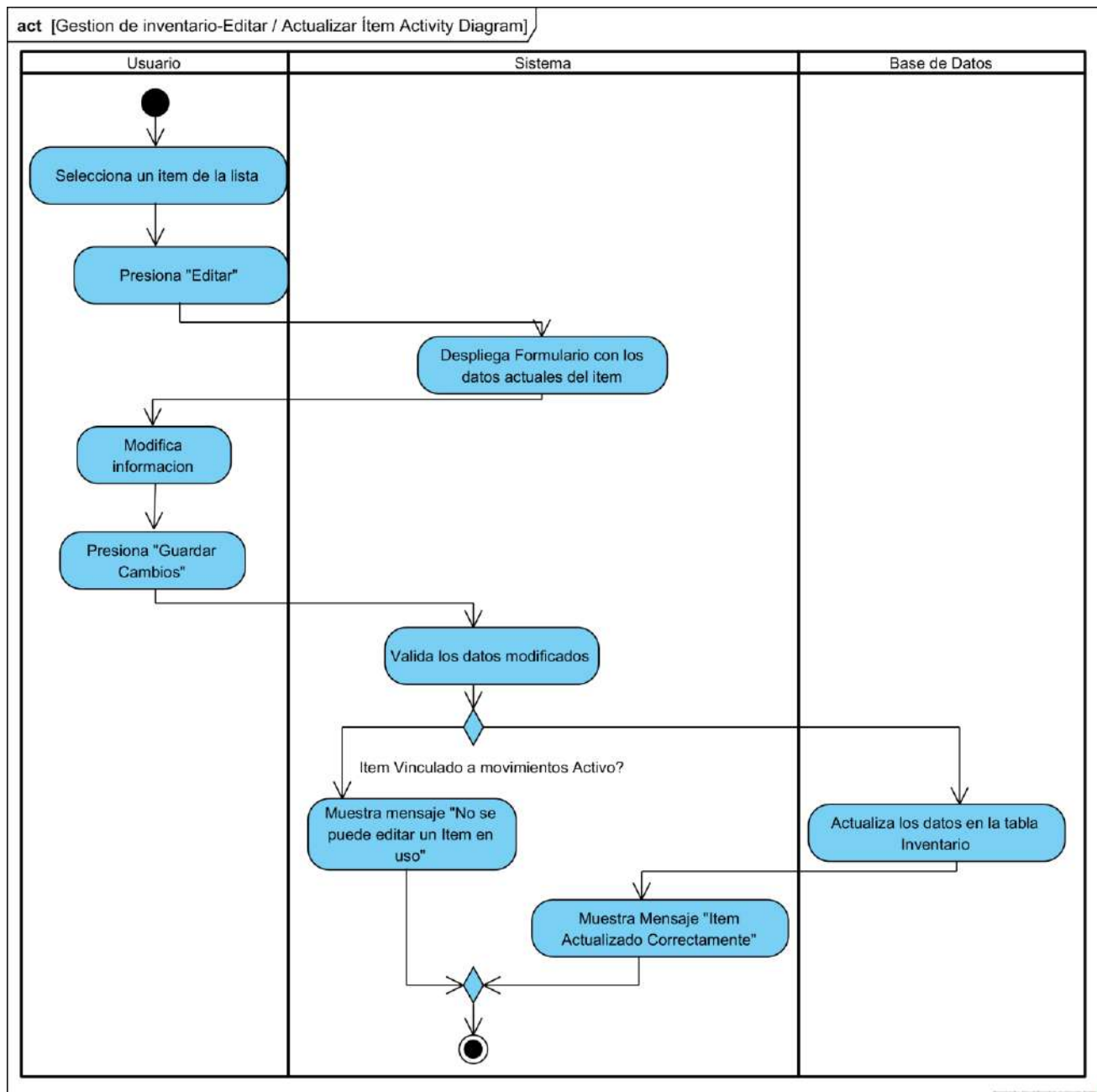


Figura 51. Gestionar Gestión de Inventario-Editar/Actualizar Ítem
Fuente: Elaboración Propia.

- Registrar Movimiento de Inventario

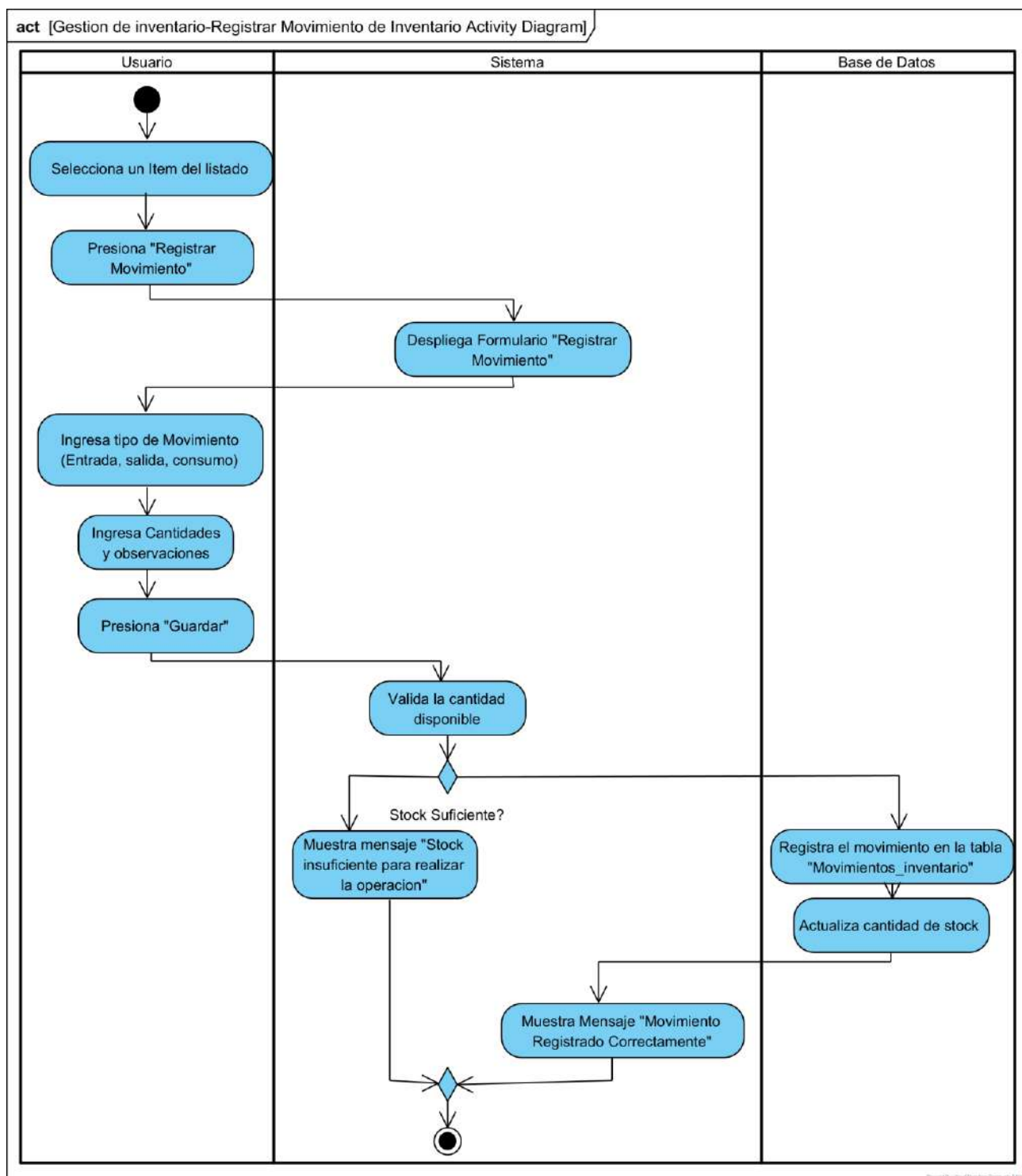


Figura 52. Registrar Movimiento de Inventario
Fuente: Elaboración Propia.

- **Controlar Stock y Notificaciones de Cantidad Crítica**

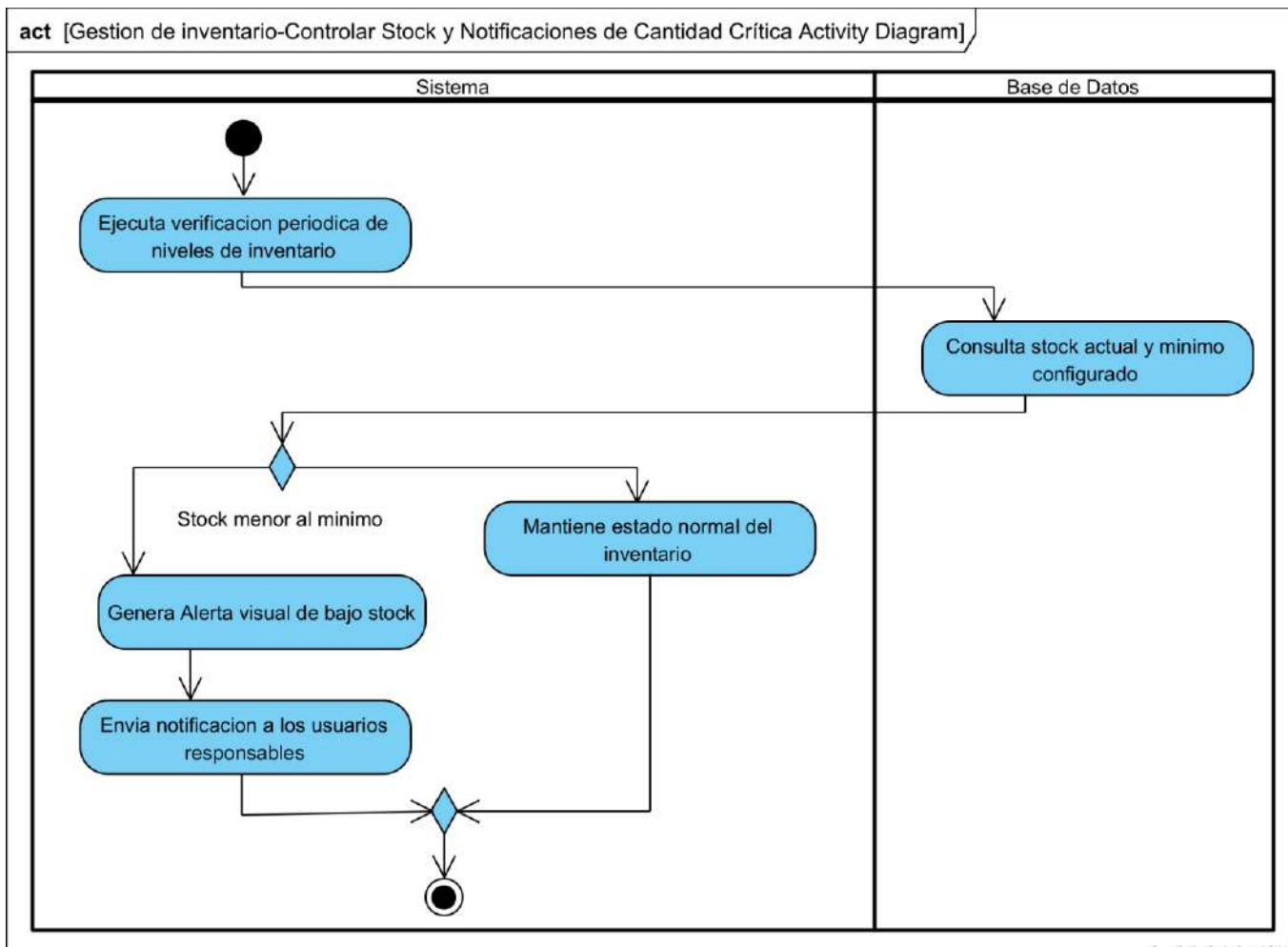


Figura 53. Gestionar Gestión de Inventario-Controlar Stock y Notificaciones de Cantidad Crítica
Fuente: Elaboración Propia.

- Gestionar Catálogos

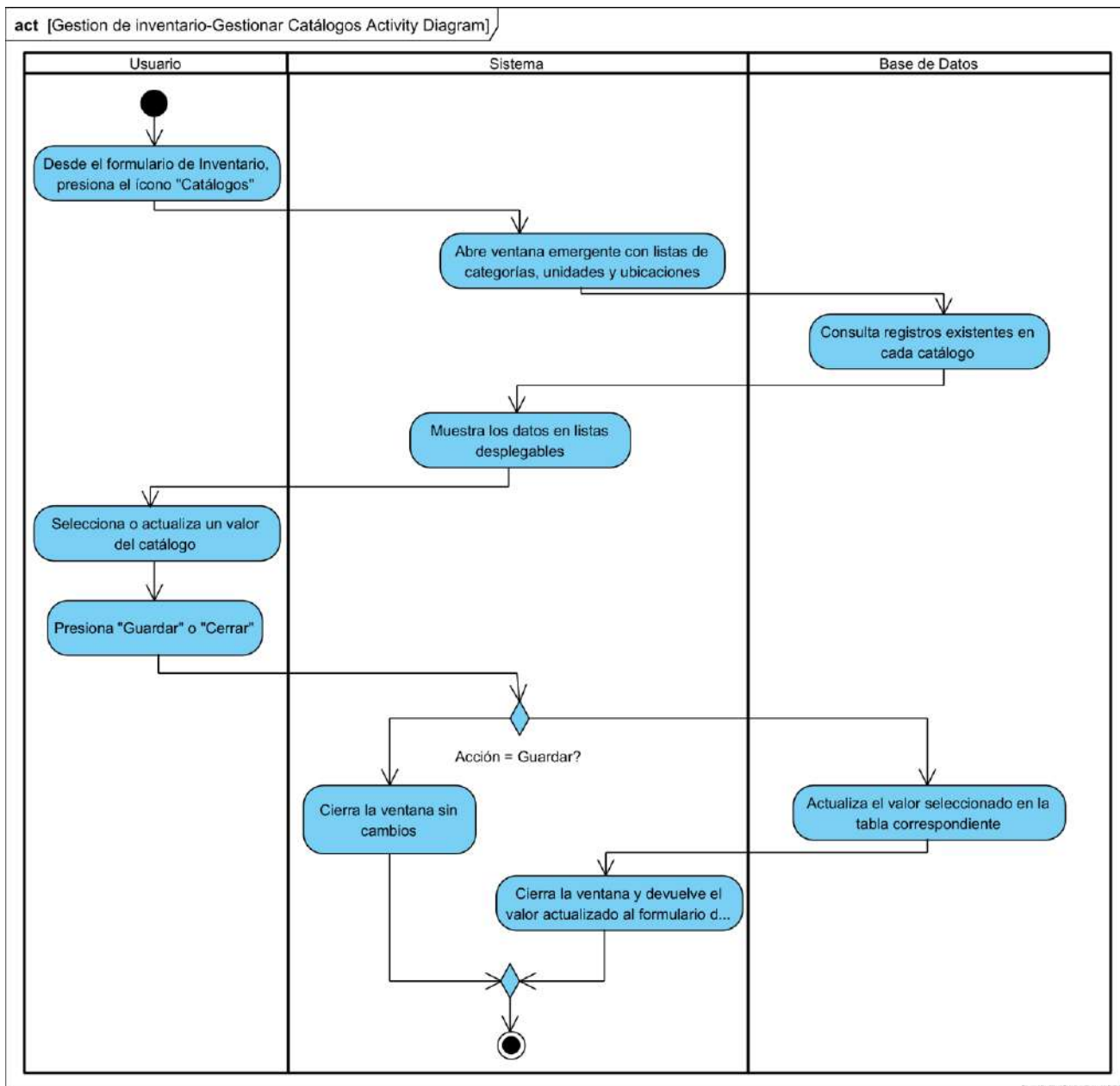


Figura 54. Gestionar Gestión de Inventario-Gestionar Catálogos
Fuente: Elaboración Propia.

- **Generar Reportes de Inventario y Movimientos**

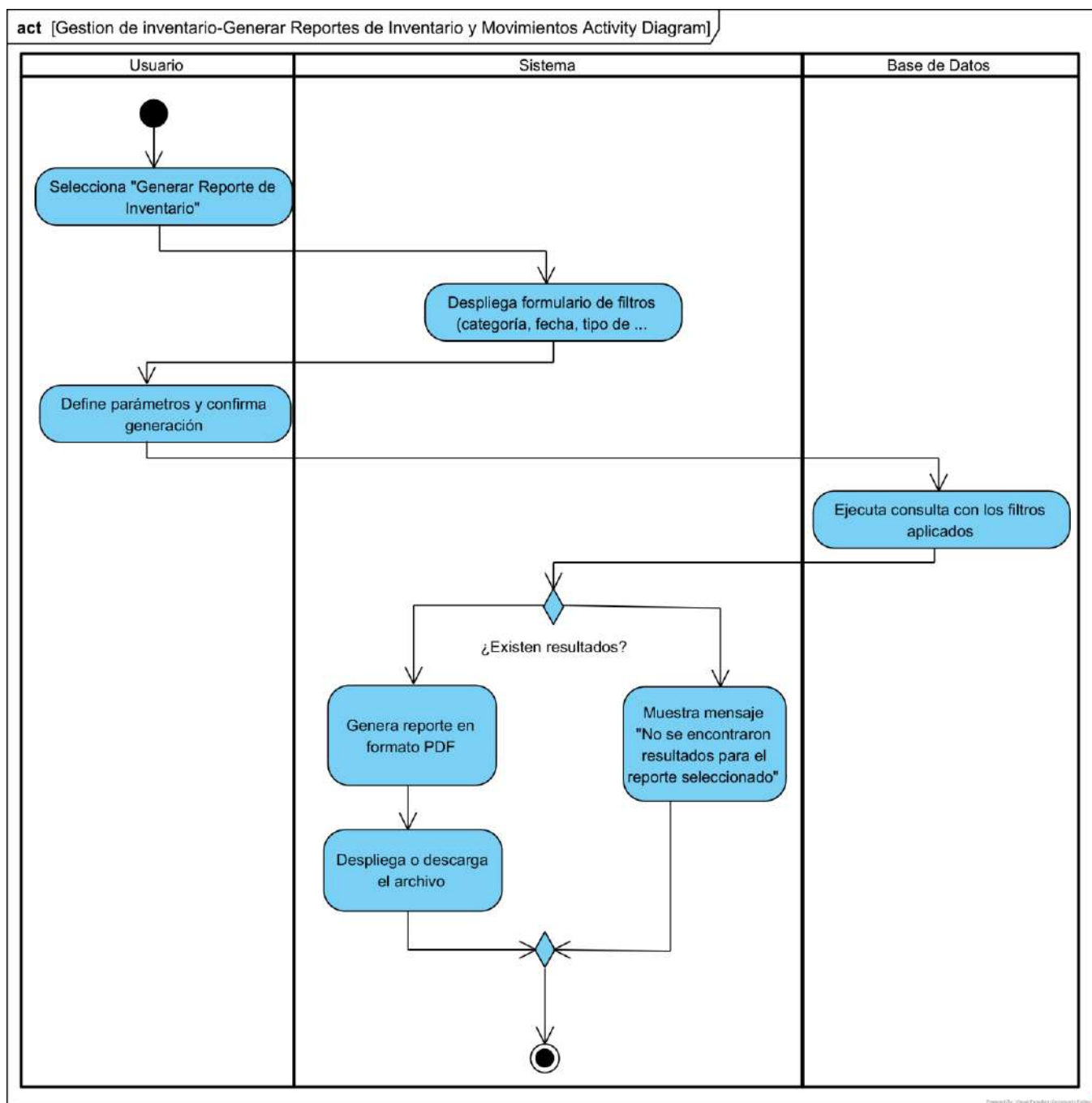


Figura 55. Gestionar Gestión de Inventario-Generar Reportes de Inventario y Movimientos
Fuente: Elaboración Propia.

II.1.2.6.10.8 Diagrama de Actividades: Gestión de Donativos

- Listar Donativos

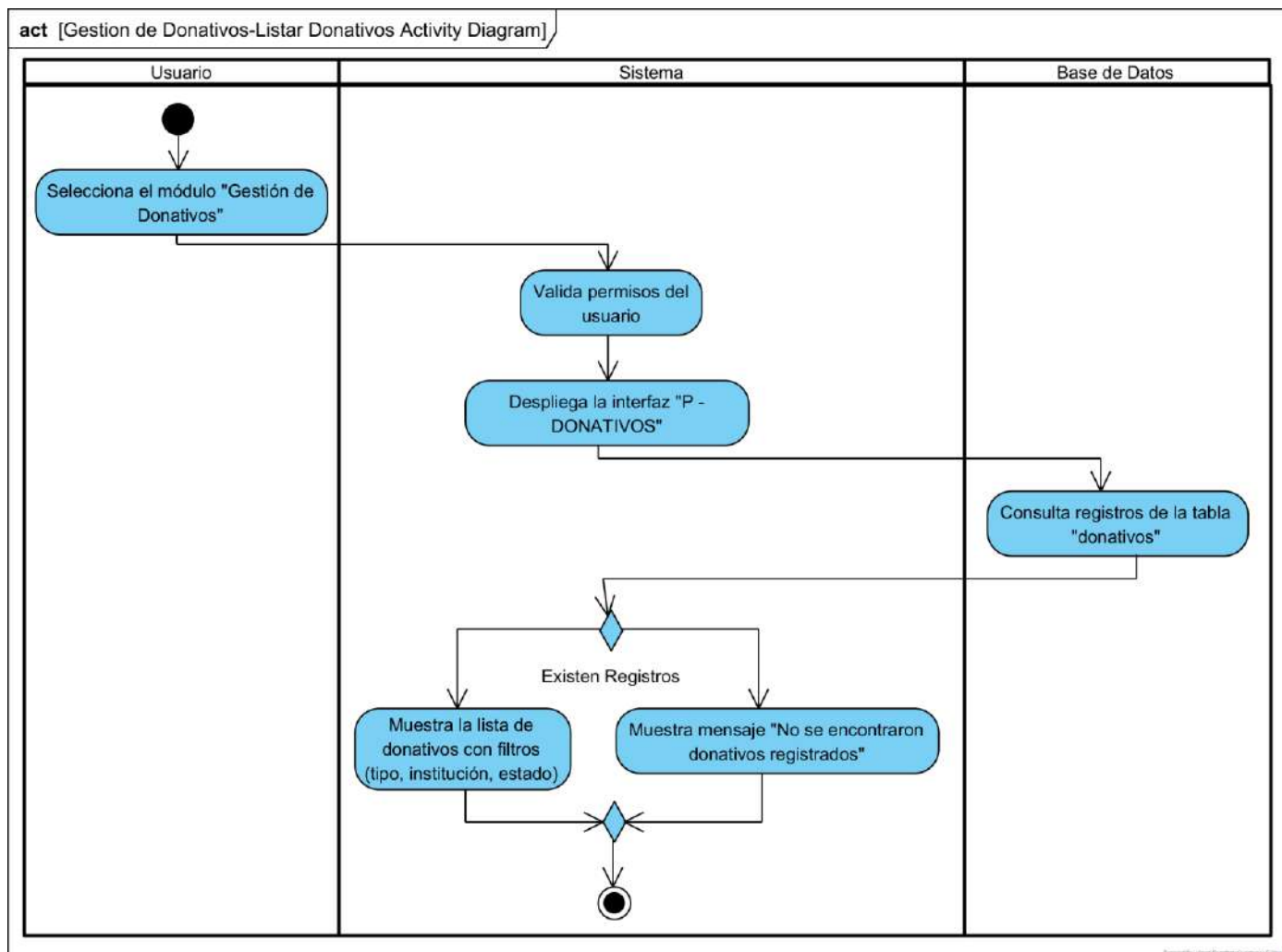


Figura 56. Gestionar Gestión de Donativos-Listar Donativos
Fuente: Elaboración Propia.

- Registrar Donativos

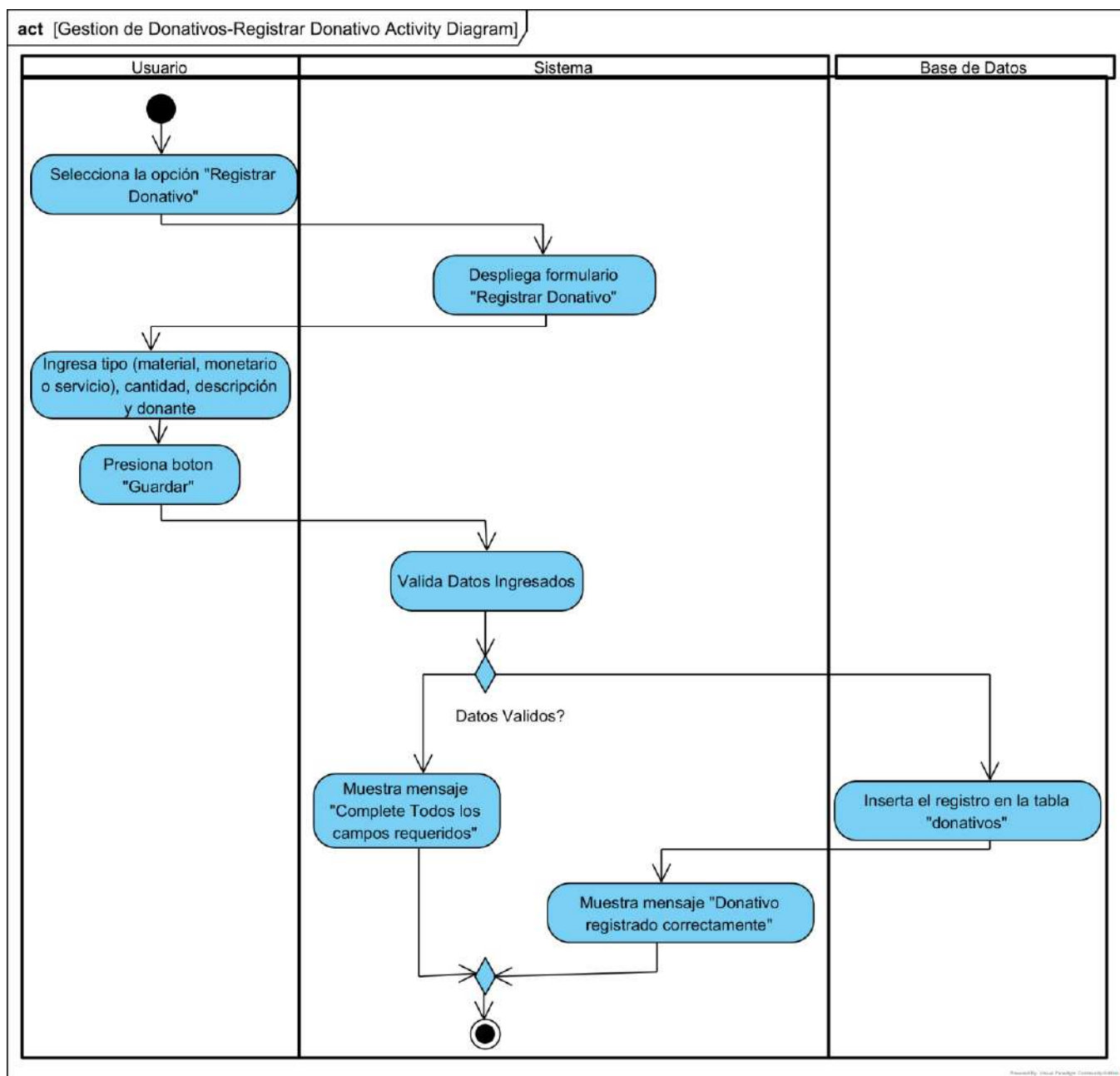


Figura 57. Gestionar Gestión de Donativos-Registrar Donativos
Fuente: Elaboración Propia.

- **Registrar Donante o Institución Asociada**

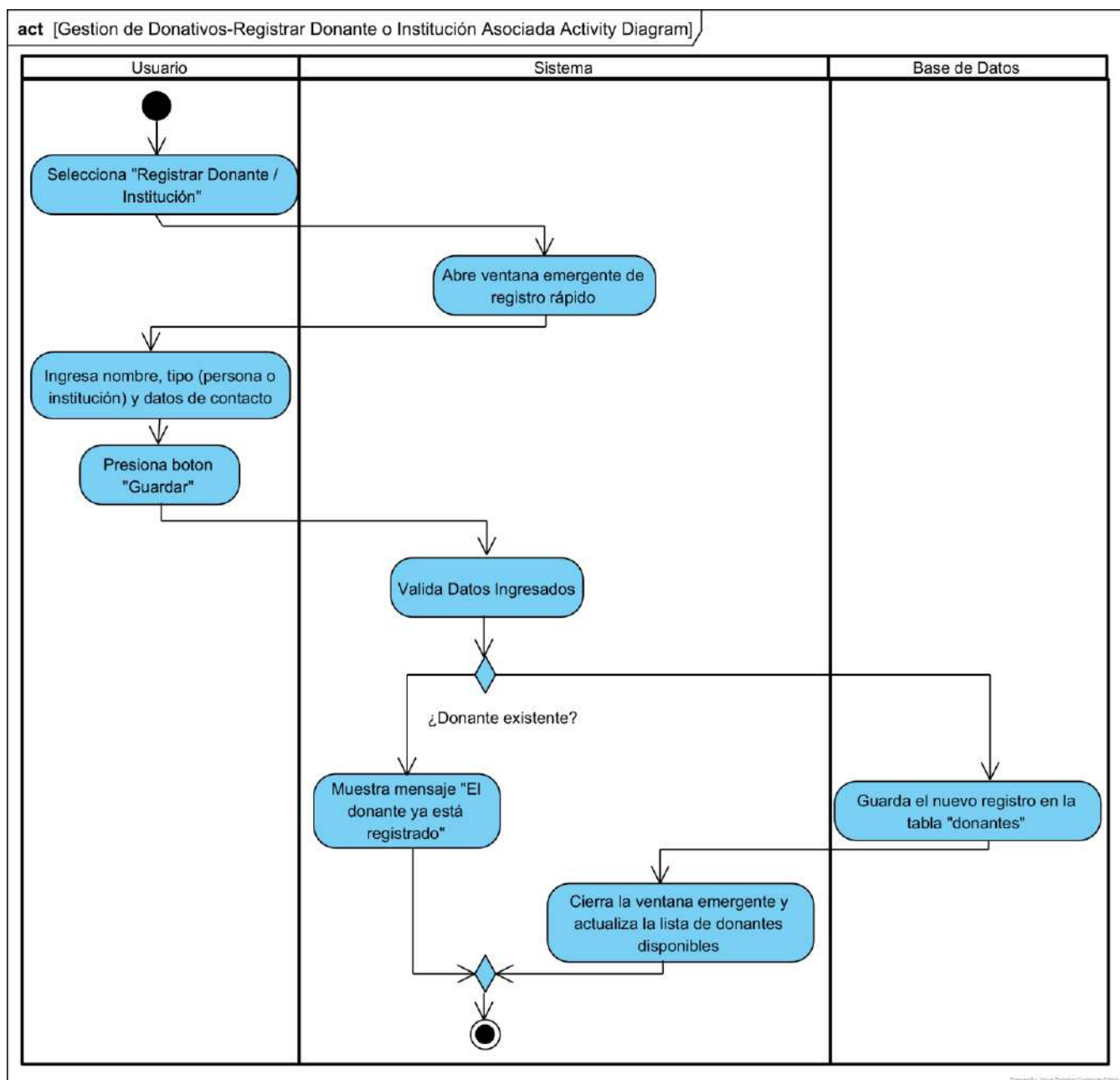


Figura 58. Gestionar Gestión de Donativos-Registrar Donante o Institución Asociada
Fuente: Elaboración Propia.

- **Actualizar / Cambiar Estado del Donativo**

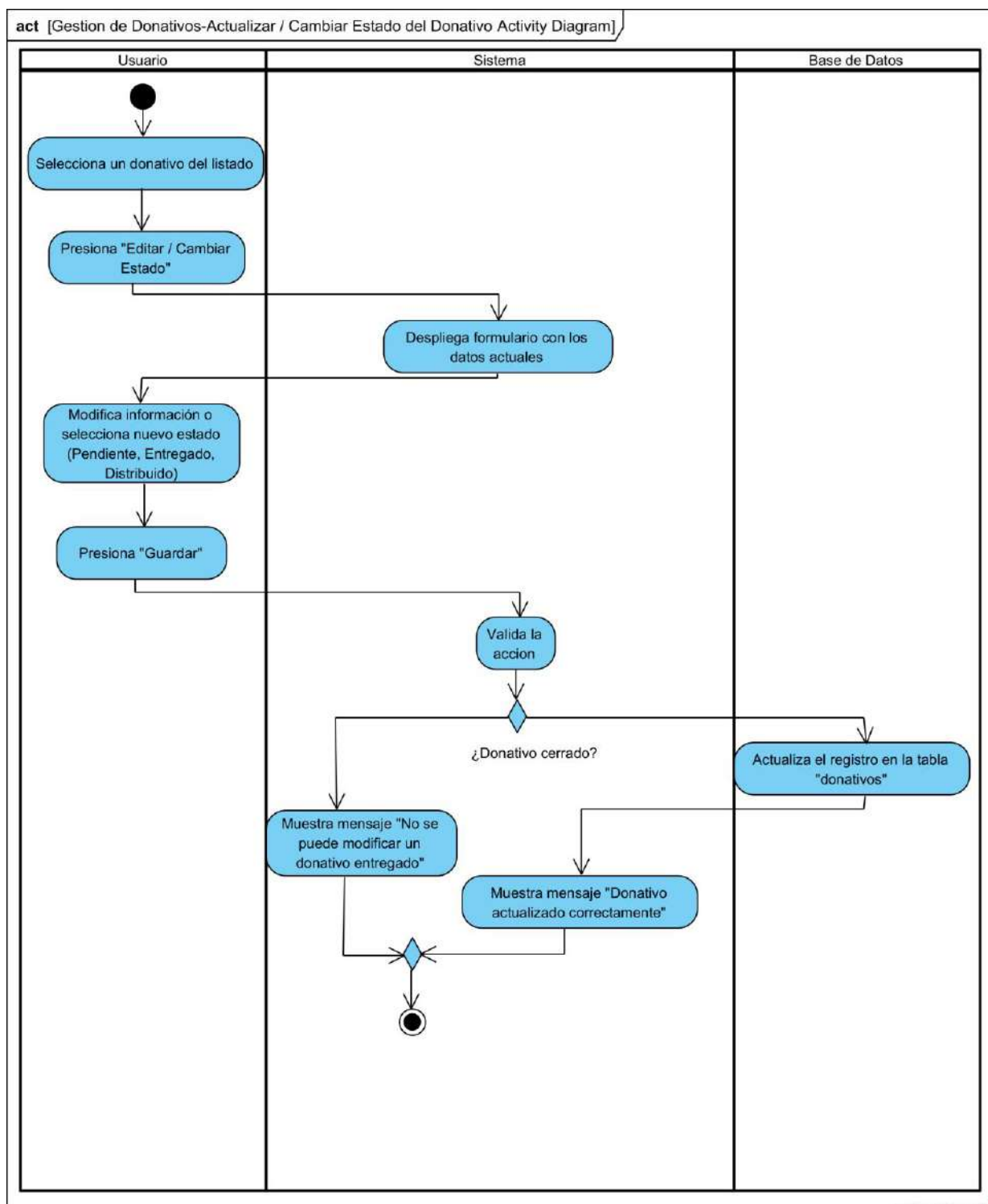


Figura 59. Gestionar Gestión de Donativos-Actualizar/Cambiar Estado del Donativo
Fuente: Elaboración Propia.

- **Asociar Donativo Material al Inventario**

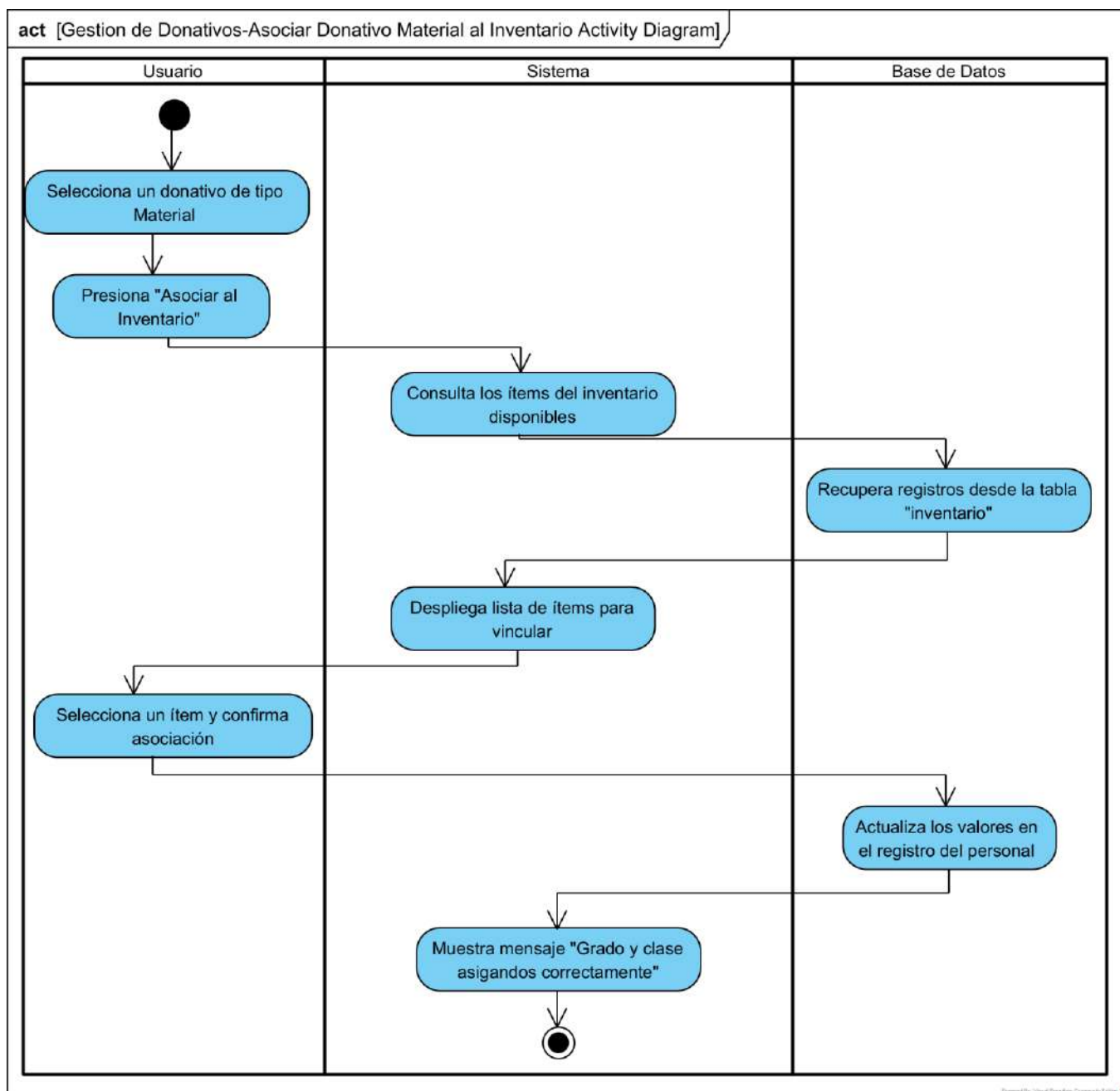


Figura 60. Gestionar Gestión de Donativos-Asociar Donativo Material al Inventario
Fuente: Elaboración Propia.

- **Generar Reportes de Donativos**

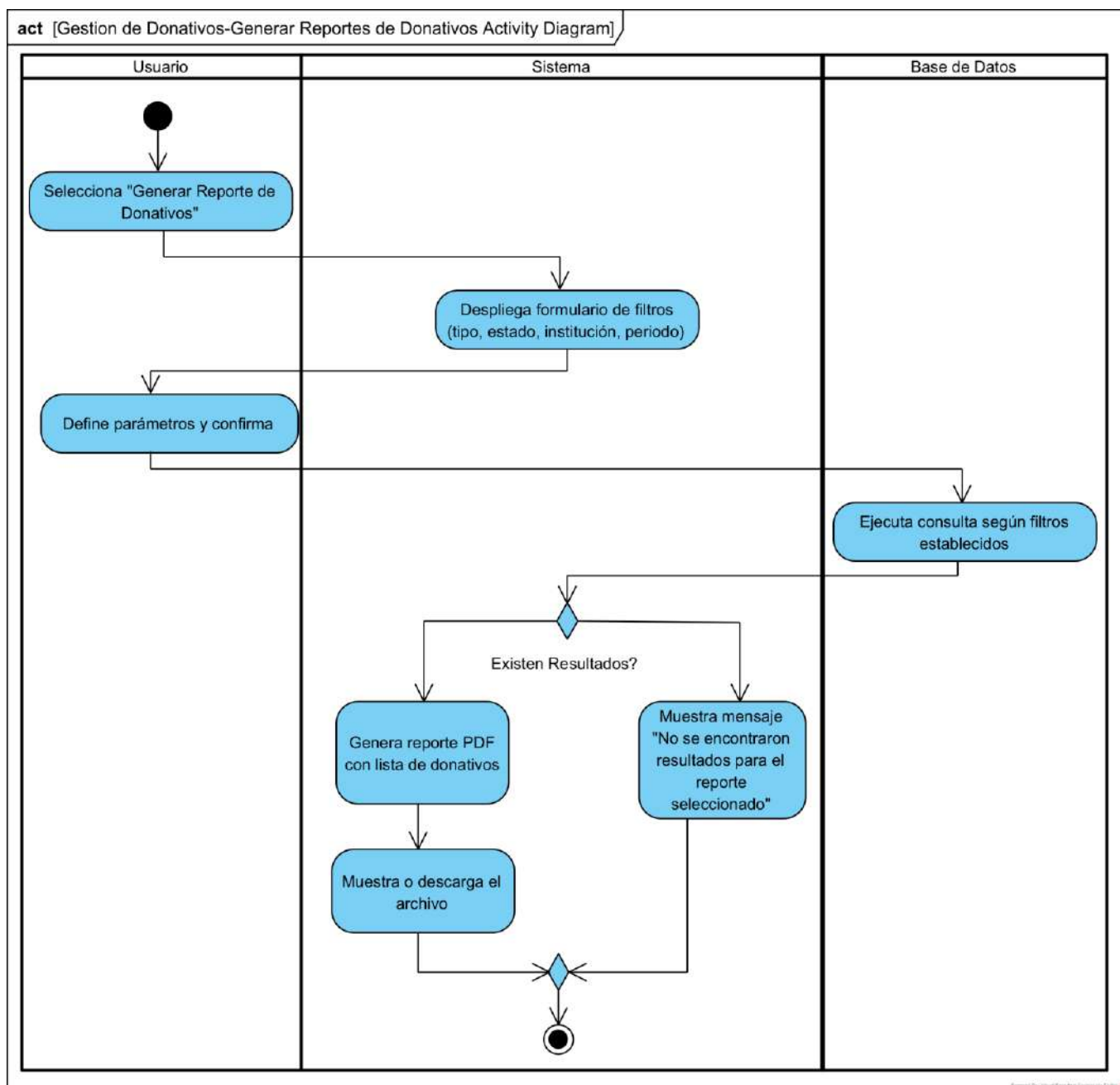


Figura 61. Gestionar Gestión de Donativos-Generar Reportes de Donativos
Fuente: Elaboración Propia.

II.1.2.6.10.9 Diagrama de Actividades: Gestión de Roles

- Listar de Roles

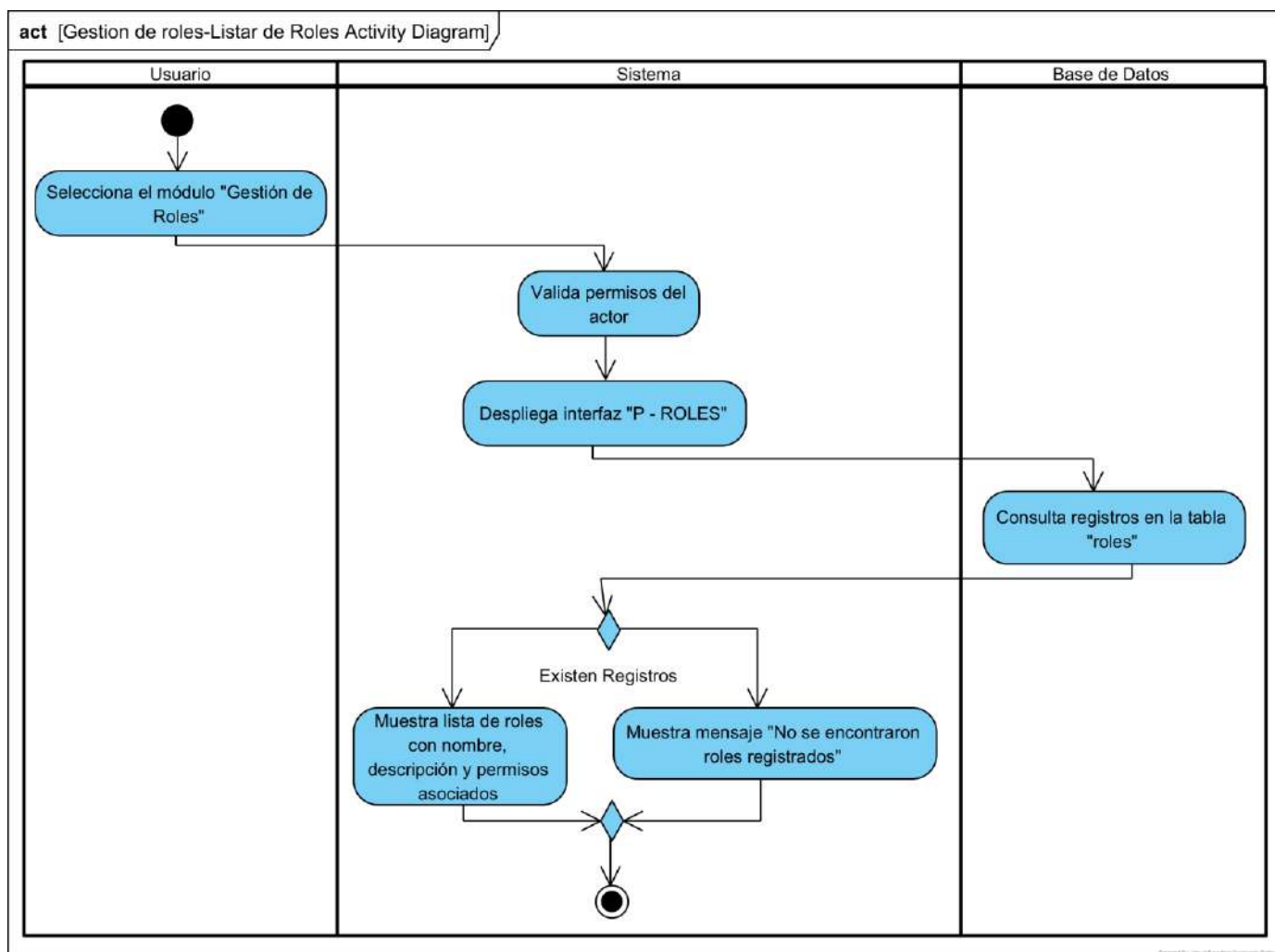


Figura 62. Gestionar Gestión de Roles-Listar Roles
Fuente: Elaboración Propia.

- Registrar Nuevo Rol

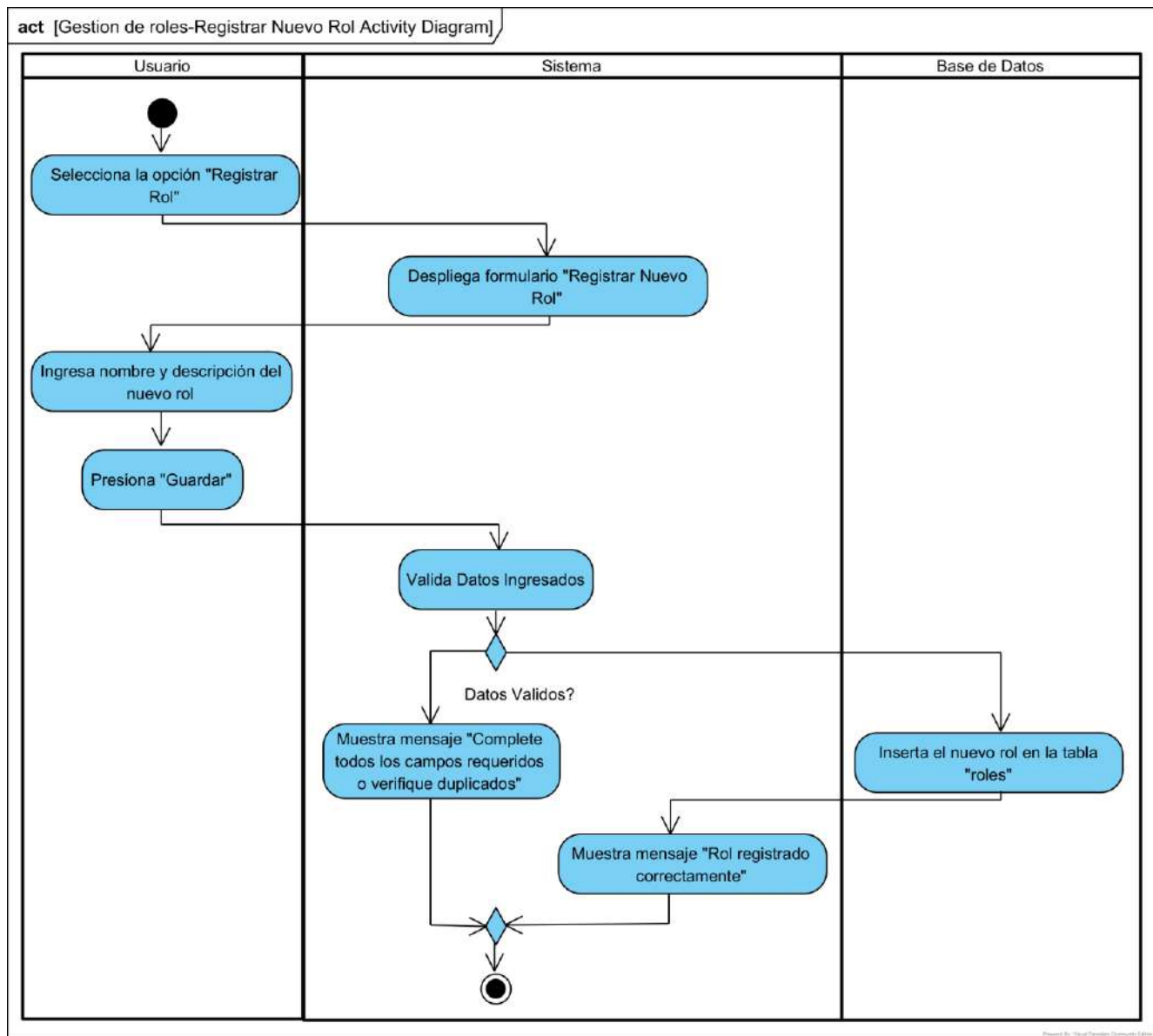


Figura 63. Gestionar Gestión de Roles-Registrar Nuevo Rol
Fuente: Elaboración Propia.

- **Editar / Actualizar Rol**

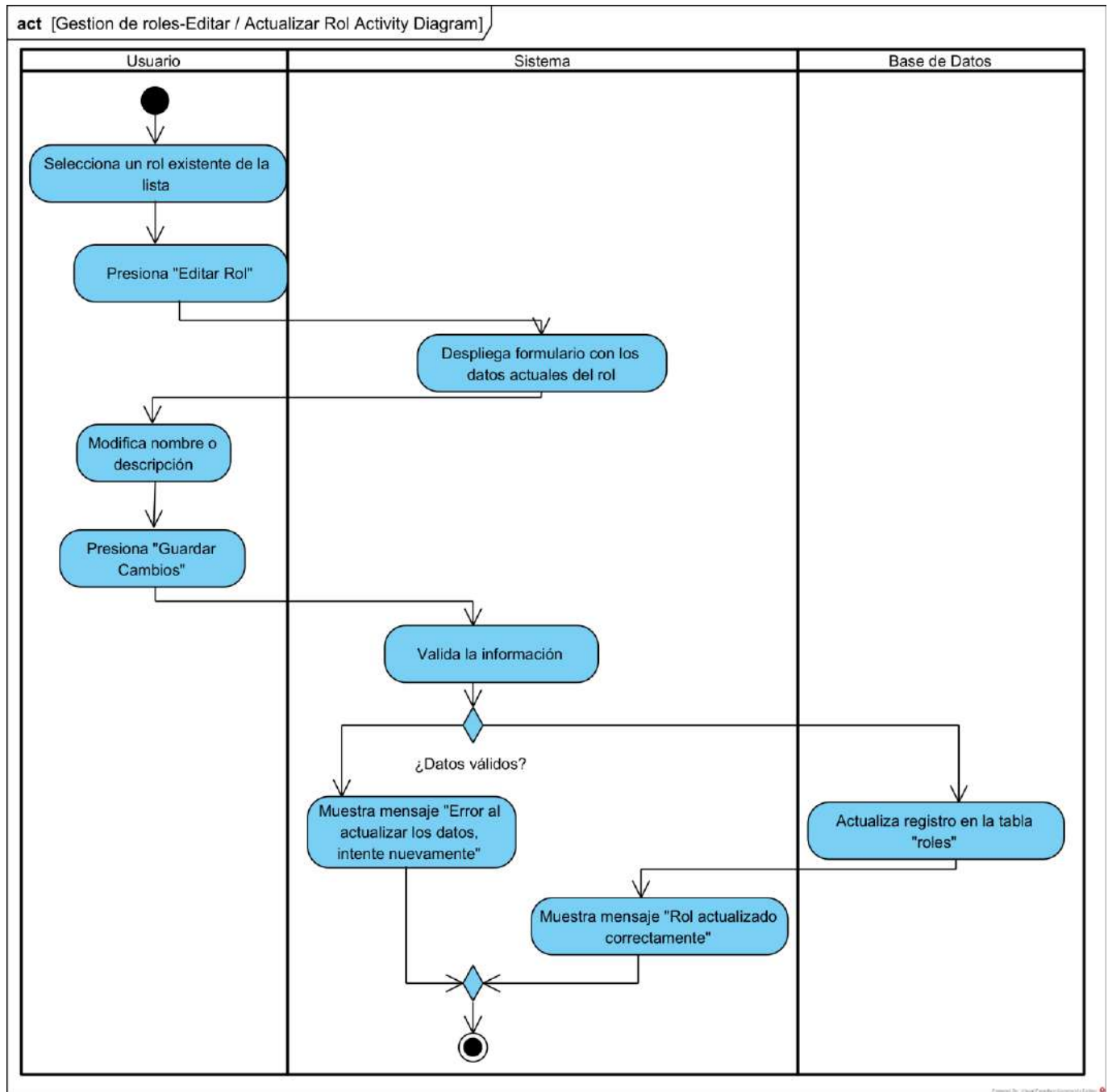


Figura 64. Gestionar Gestión de Roles-Editar/Actualizar Rol
Fuente: Elaboración Propia.

- **Eliminar Rol**

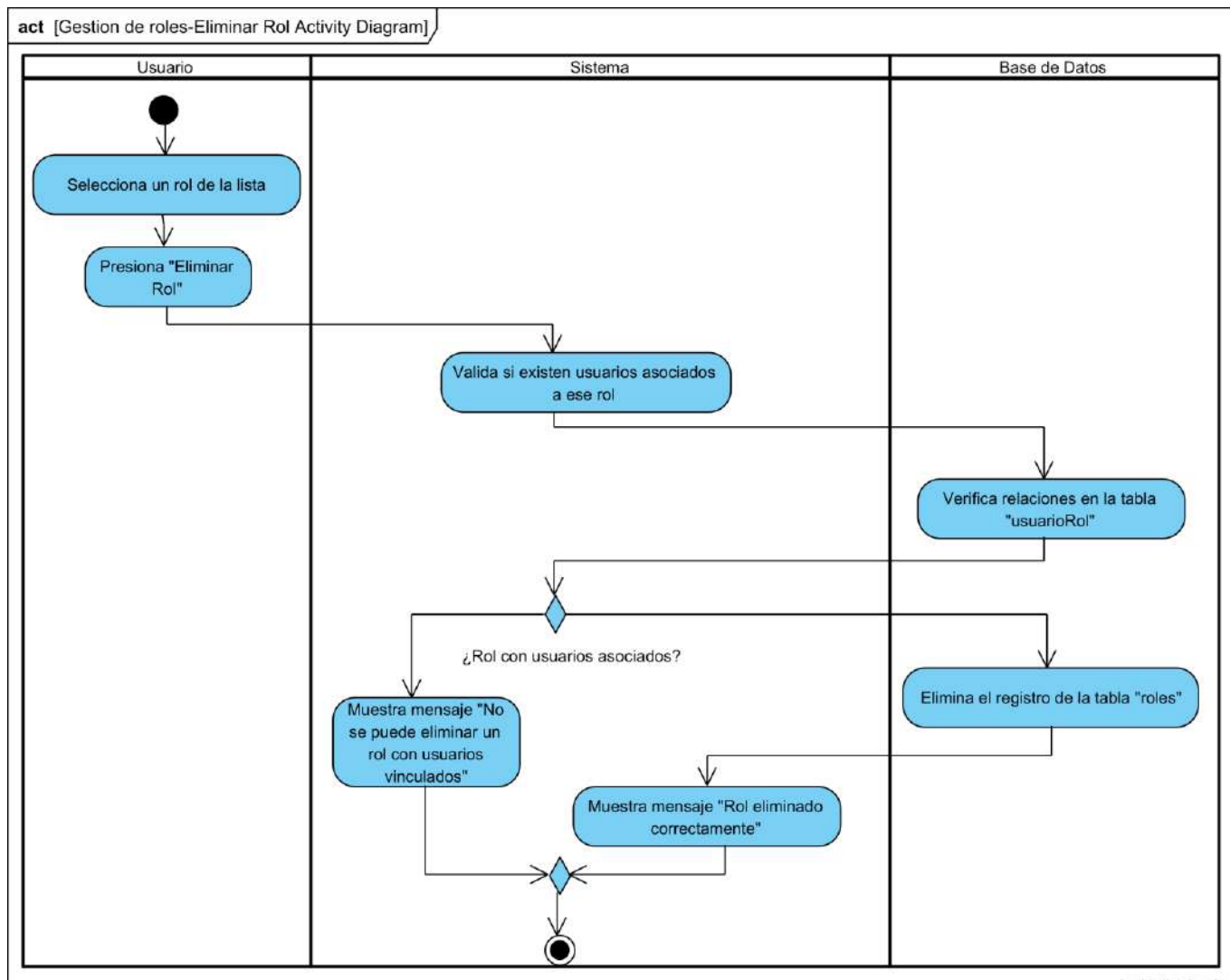


Figura 65. Gestionar Gestión de Roles-Eliminar Rol
Fuente: Elaboración Propia.

- **Asignar Permisos a Rol**

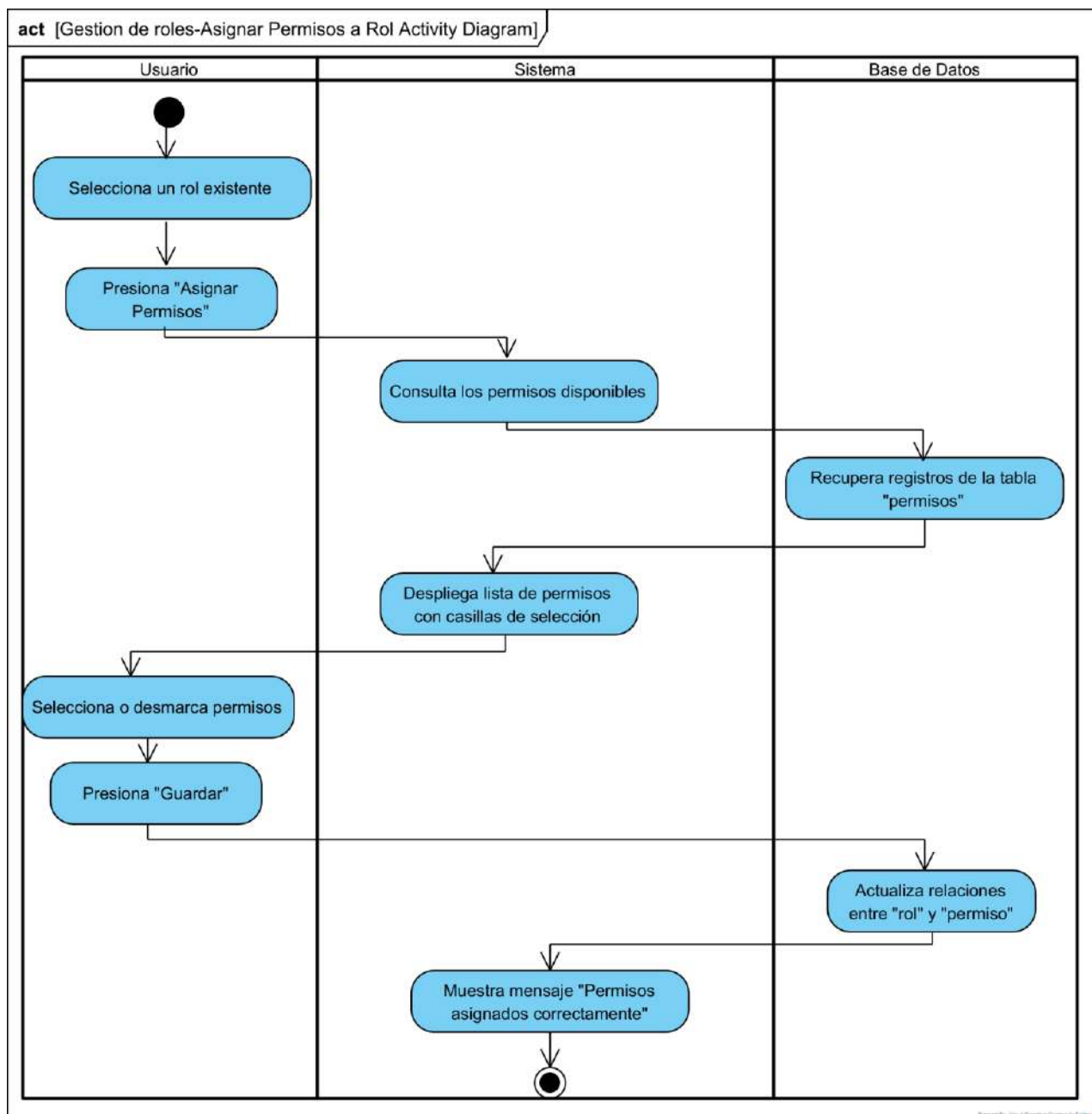


Figura 66. Gestionar Gestión de Roles-Asignar Permisos a Rol
Fuente: Elaboración Propia.

- **Ver Permisos Asignados**

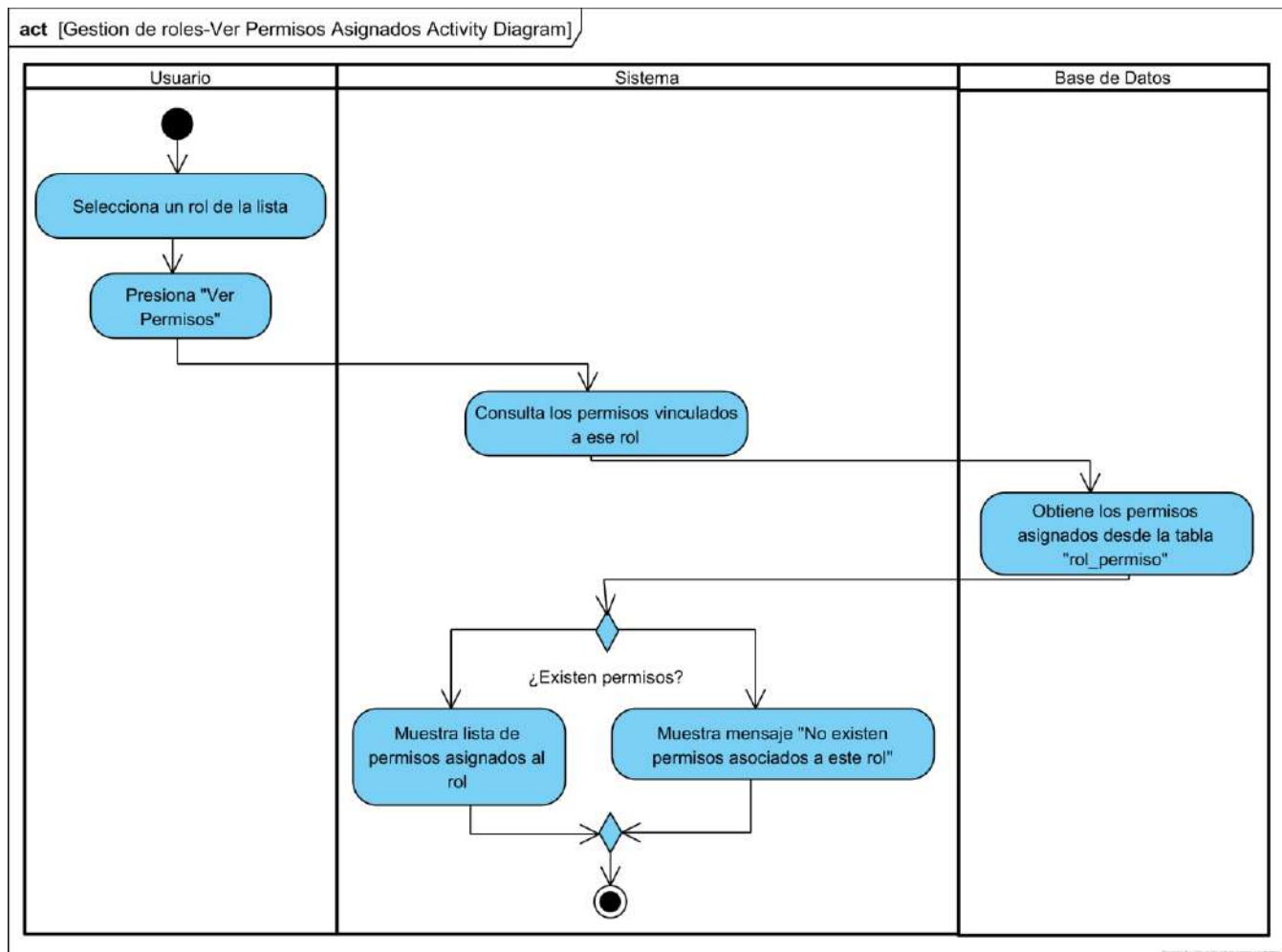


Figura 67. Gestionar Gestión de Roles-Ver Permisos Asignados
Fuente: Elaboración Propia.

- **Generar Reportes de Roles y Permisos**

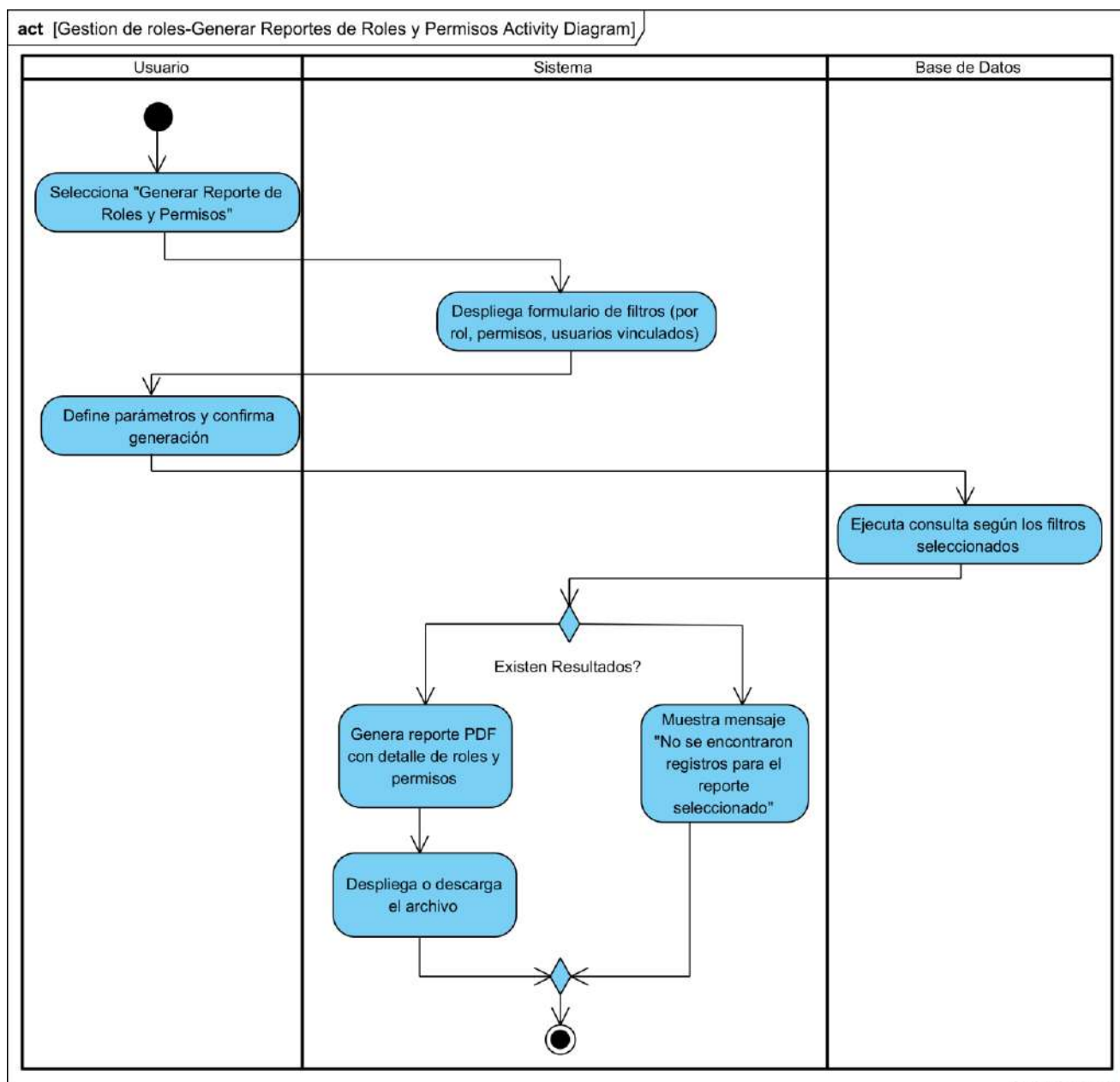


Figura 68. Gestionar Gestión de Roles-Generar Reportes de Roles y Permisos
Fuente: Elaboración Propia.

II.1.2.6.10.10 Diagrama de Actividades: Gestión de Usuarios

- Listar Usuarios

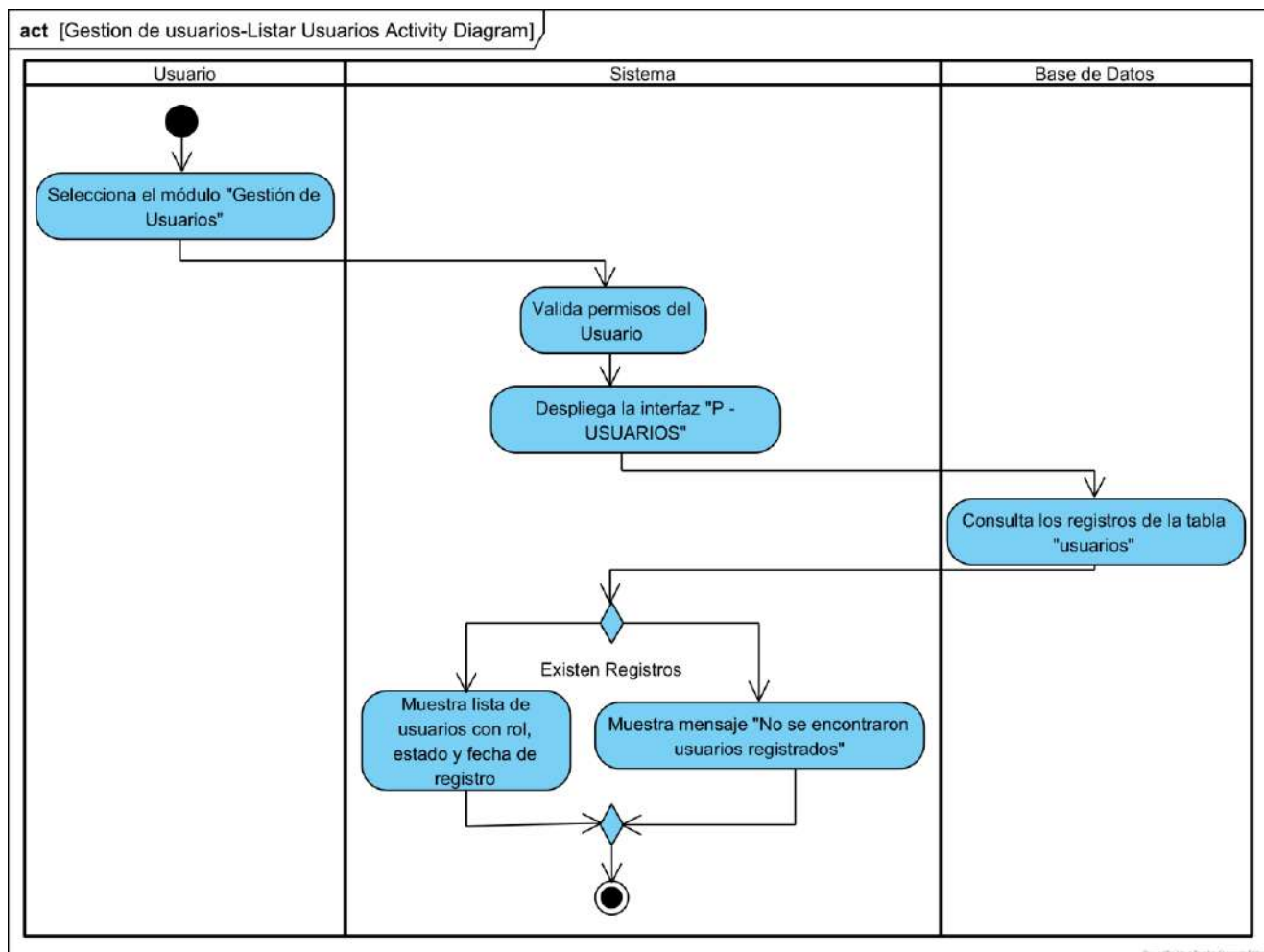


Figura 69. Gestionar Gestión de Usuarios-Listar Usuarios
Fuente: Elaboración Propia.

- **Registrar Nuevo Usuario**

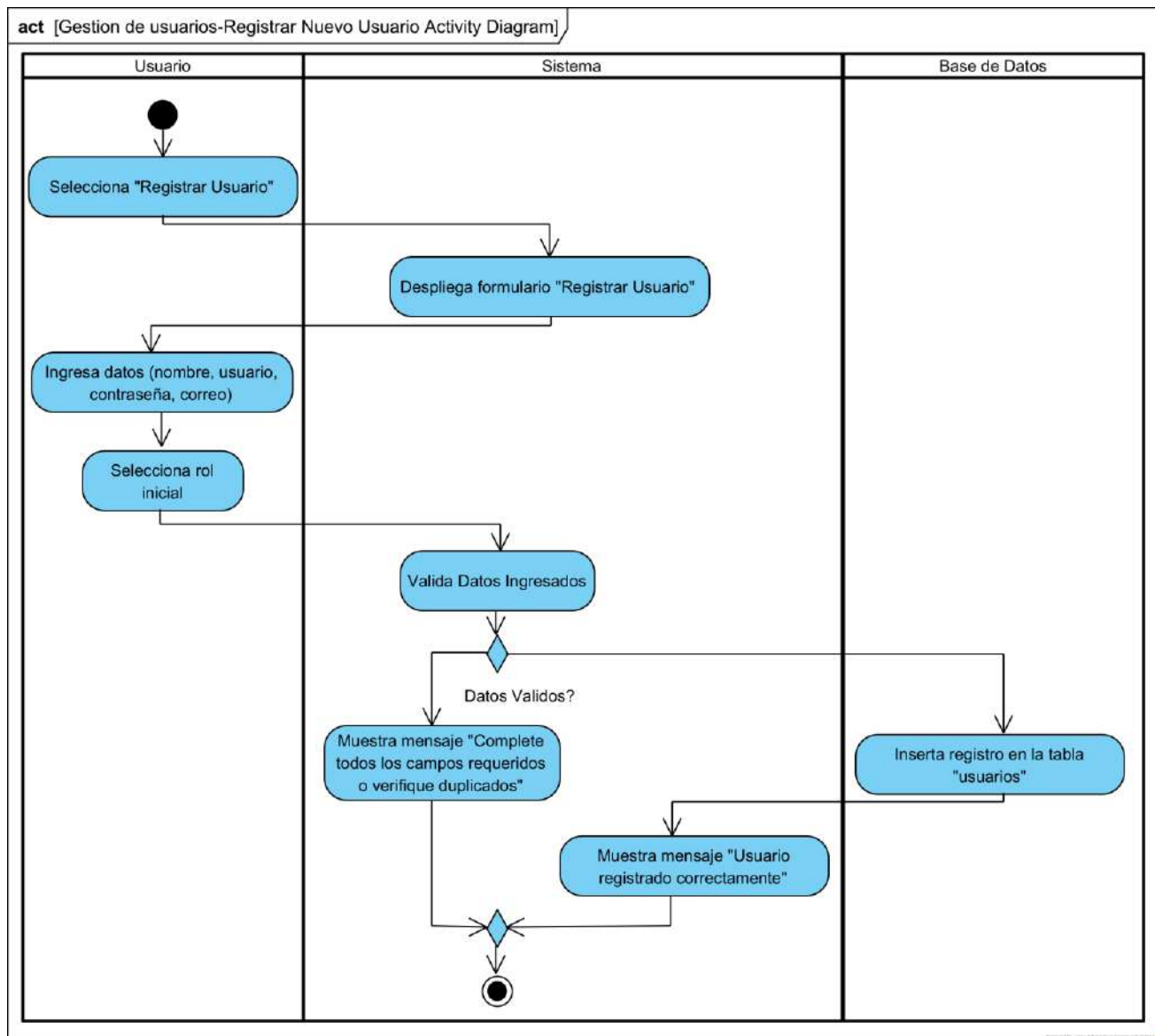


Figura 70. Gestionar Gestión de Usuarios-Registrar Nuevo Usuario
Fuente: Elaboración Propia.

- **Editar / Actualizar Usuario**

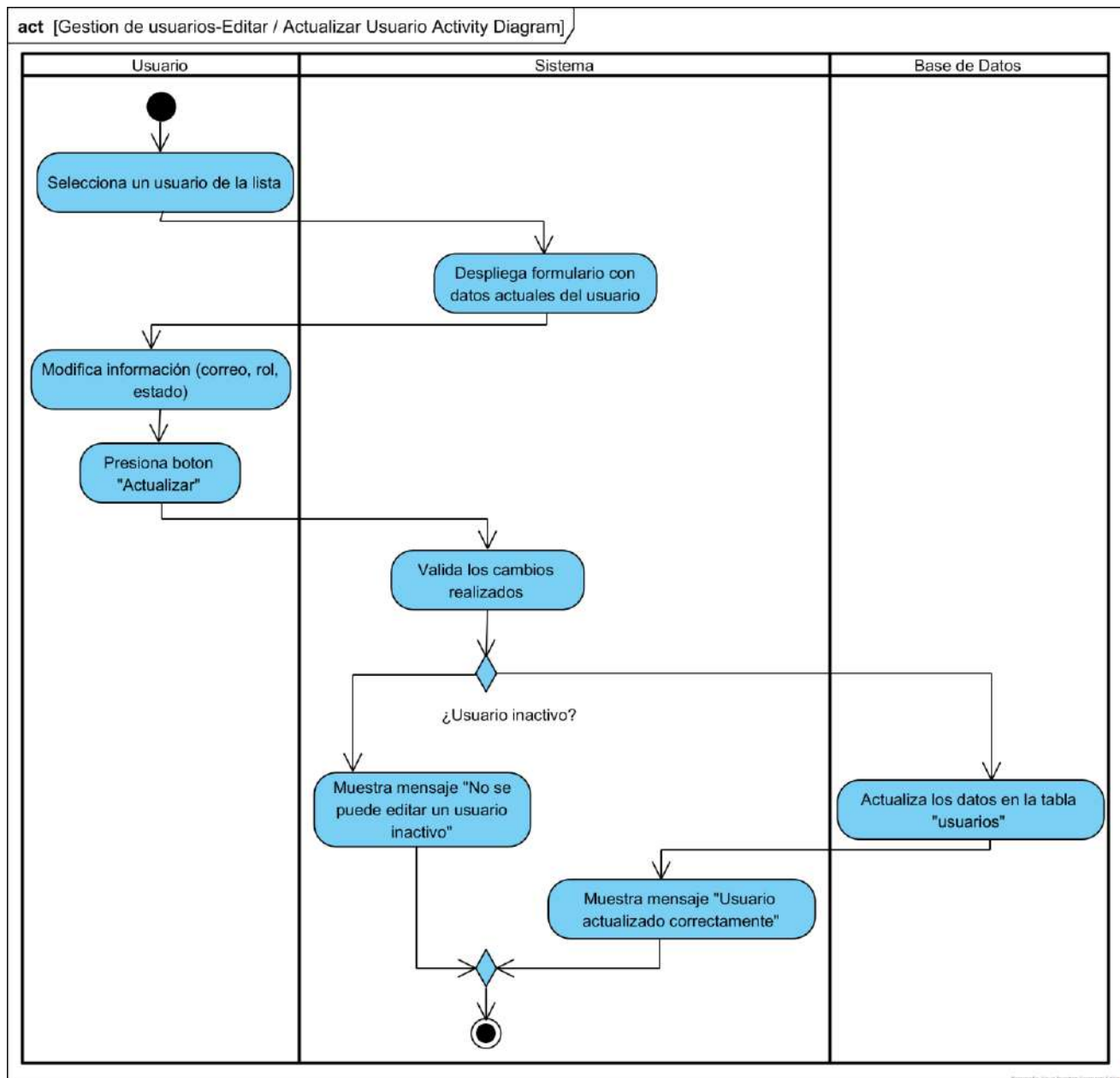


Figura 71. Gestionar Gestión de Usuarios-Editar/Actualizar Usuario
Fuente: Elaboración Propia.

- Desactivar / Reactivar Usuario

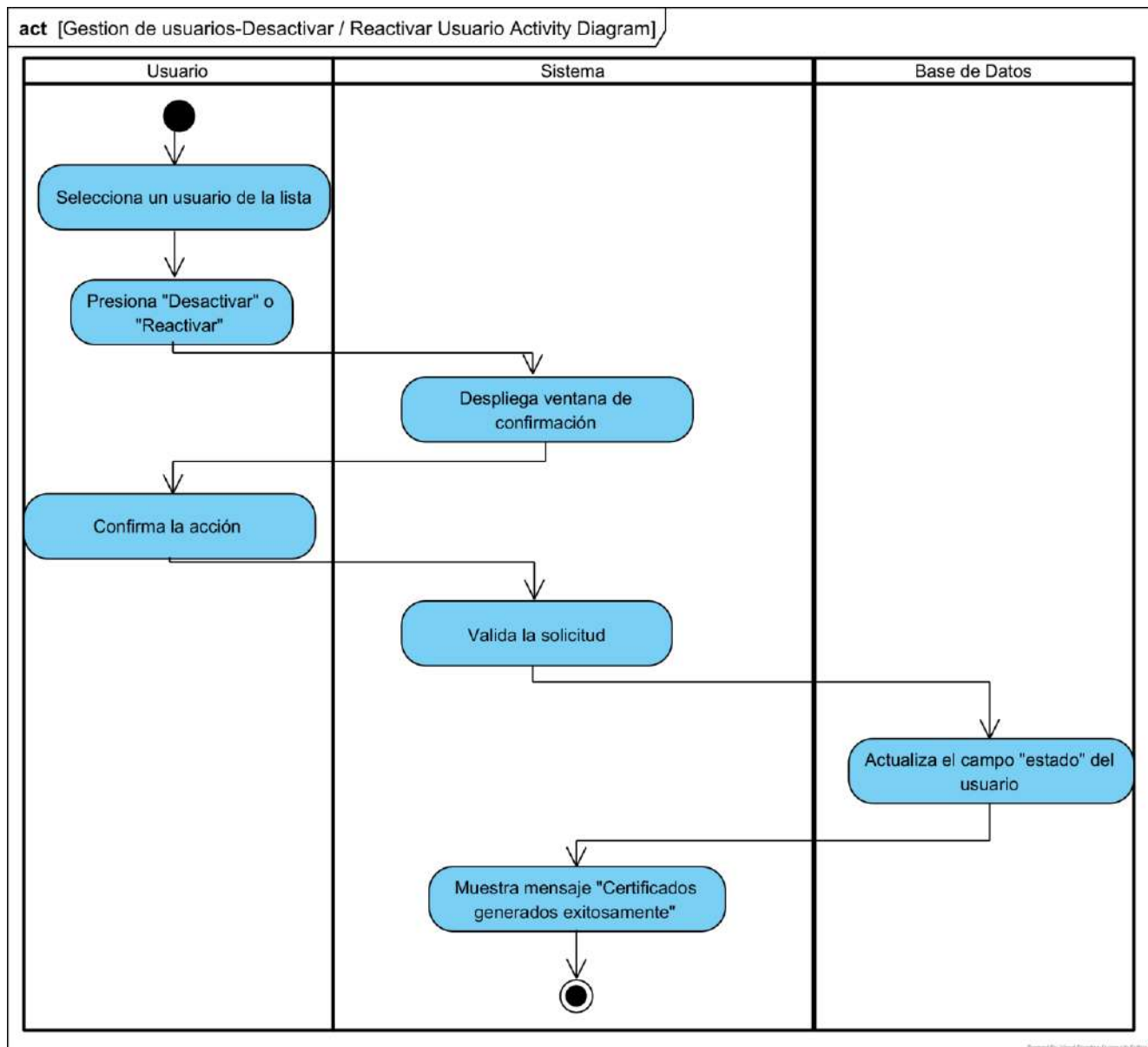


Figura 72. Gestionar Gestión de Usuarios-desactivas/Reactivar Usuario
Fuente: Elaboración Propia.

- Registrar Asignar / Eliminar Rol de Usuario

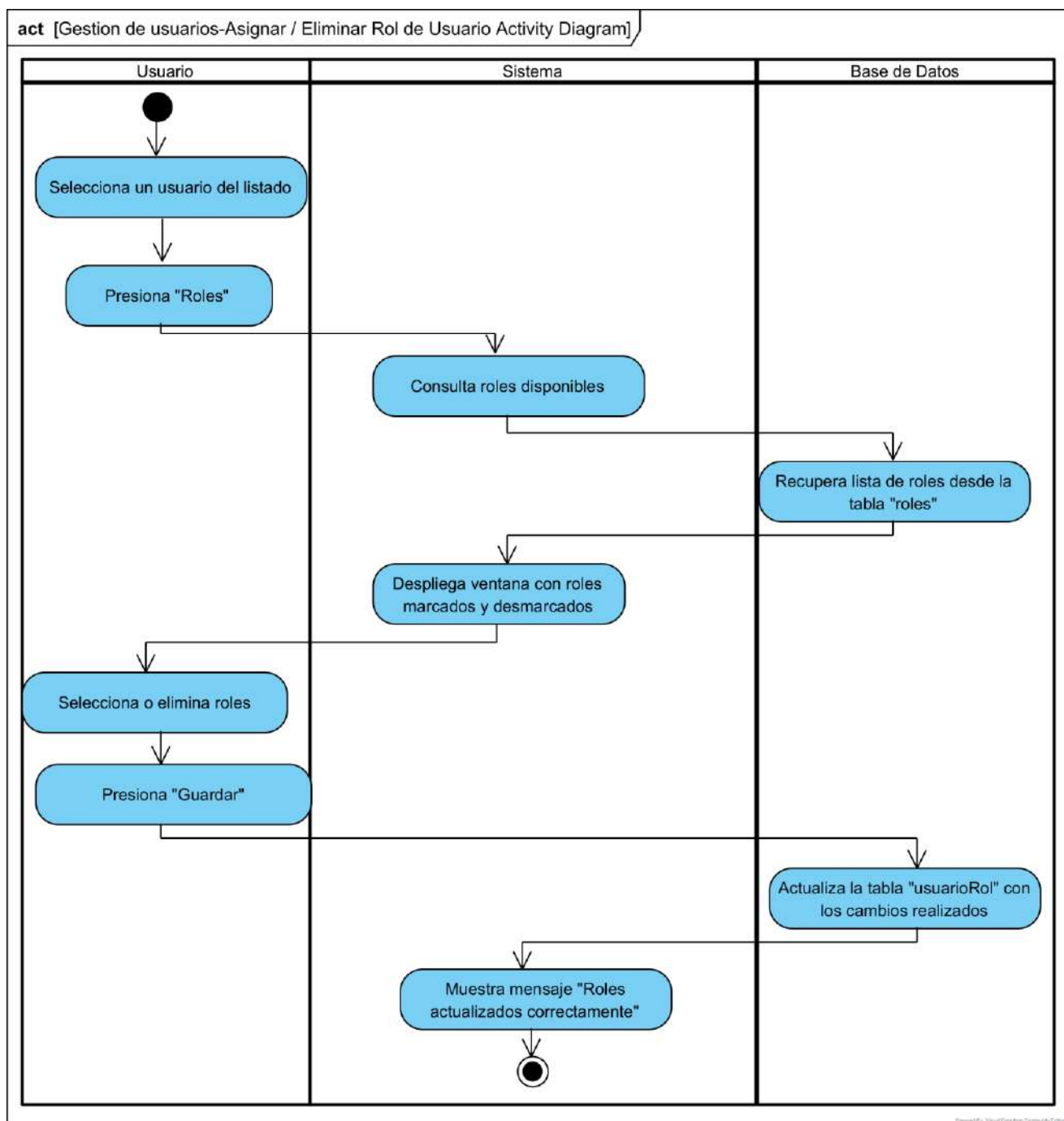


Figura 73. Gestionar Gestión de Usuarios-Registrar Asignar/Eliminar rol de Usuario
Fuente: Elaboración Propia.

- **Generar Reportes de Usuarios**

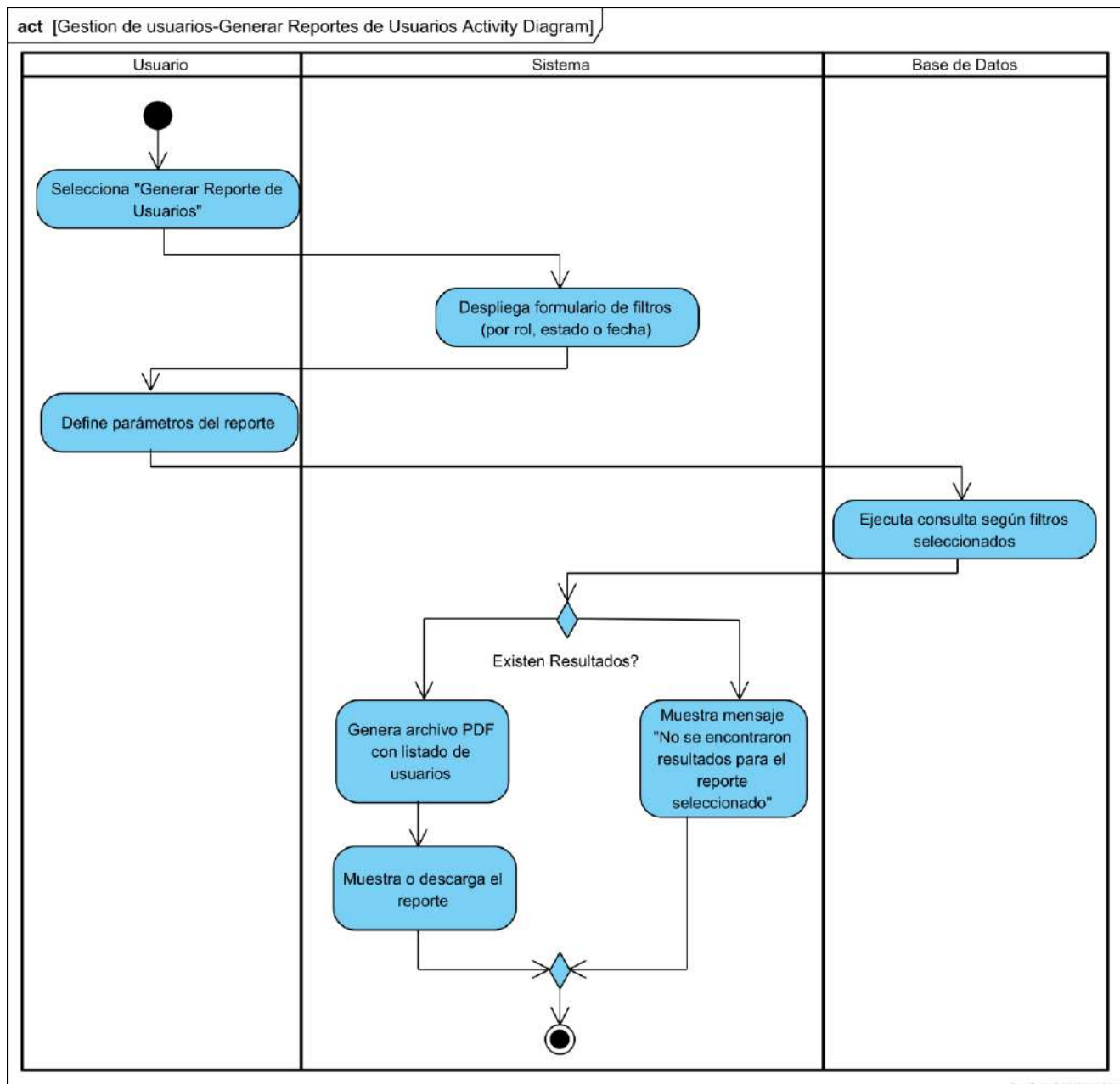


Figura 74. Gestionar Gestión de Usuarios-Generar Reportes de Usuarios
Fuente: Elaboración Propia.

II.1.2.6.11 Diagrama de Secuencia

II.1.2.6.11.1 Diagrama de Secuencia: Gestionar Inicio de Sesión

- Iniciar Sesión

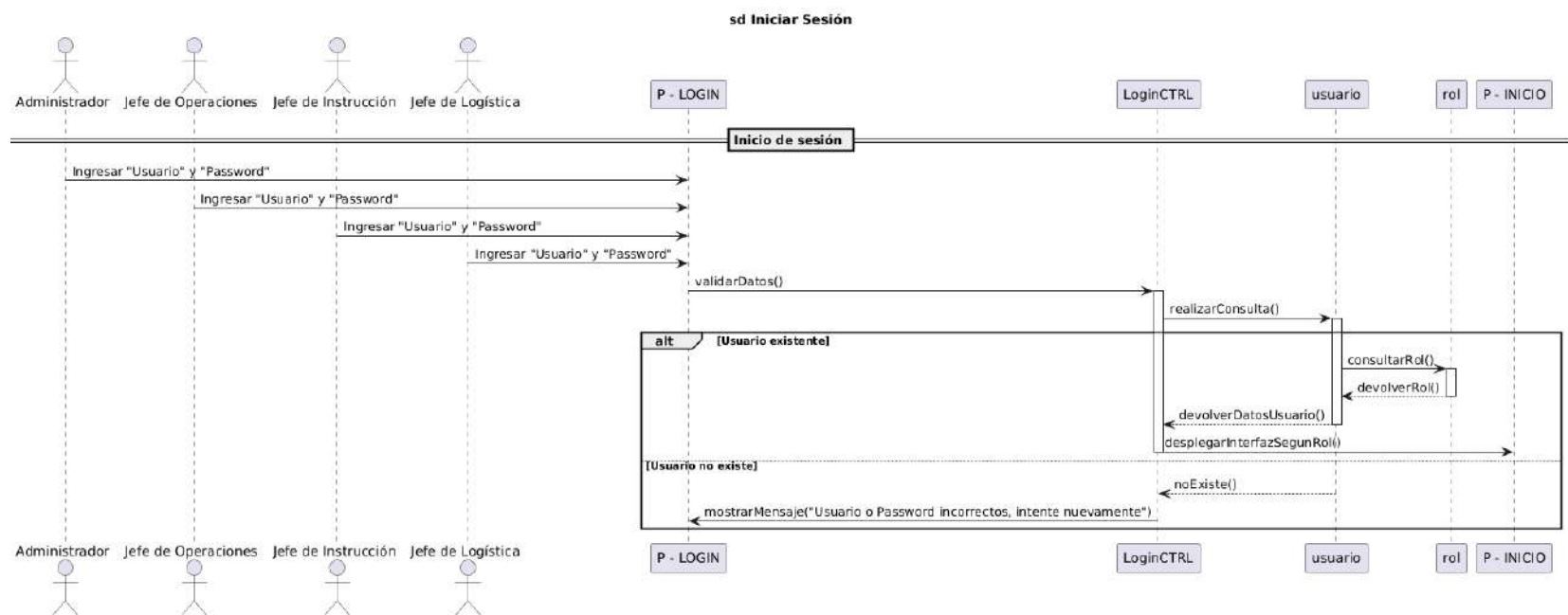


Figura 75. Diagrama de Secuencia: Gestionar Inicio de Sesión-Iniciar Sesión
Fuente: Elaboración Propia.

- Cerrar Sesión

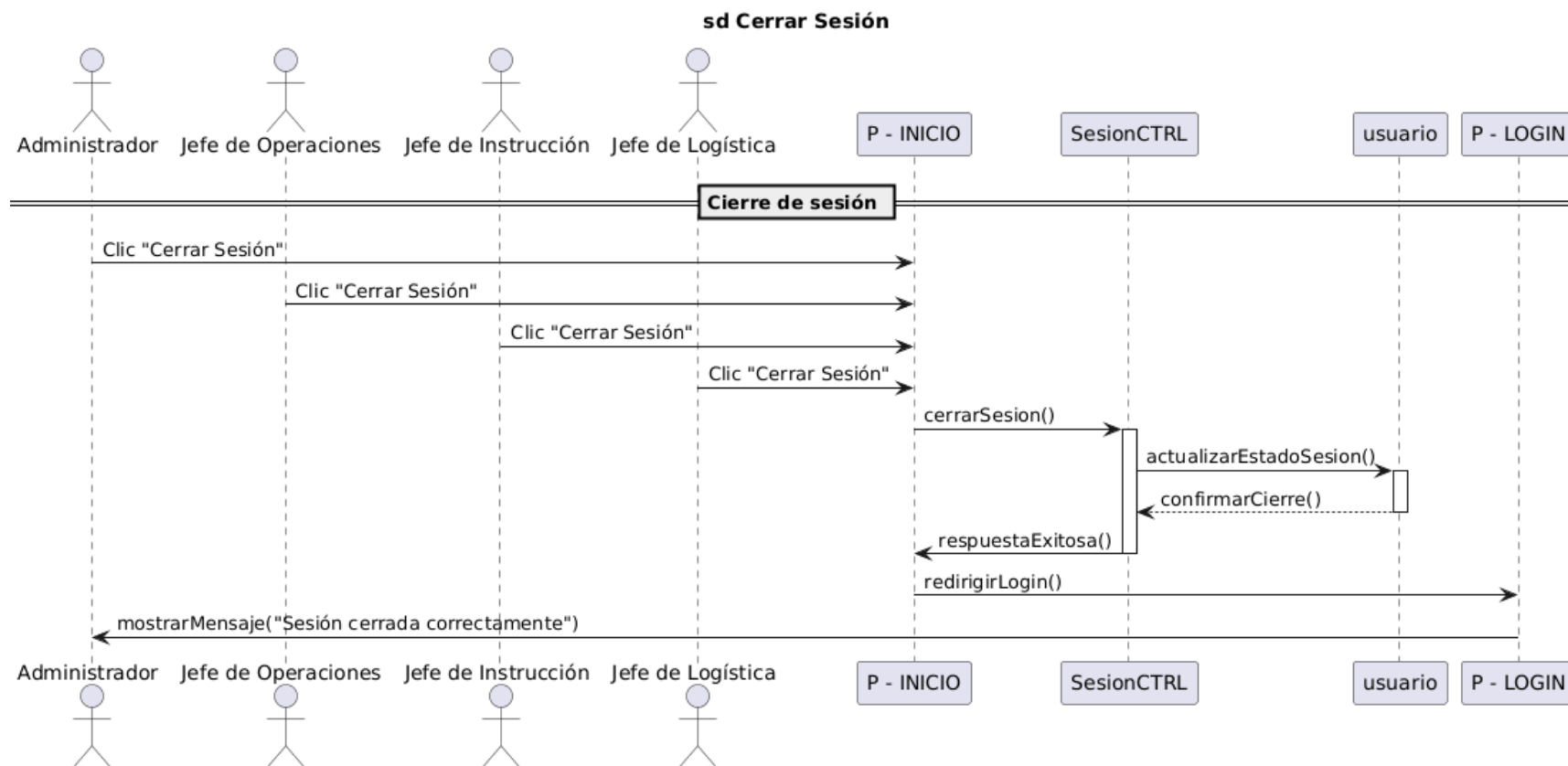


Figura 76. Diagrama de Secuencia: Gestionar Cerrar de Sesión:
Fuente: Elaboración Propia.

II.1.2.6.11.2 Diagrama de Secuencia: Gestión de Emergencias

- Listar Emergencias

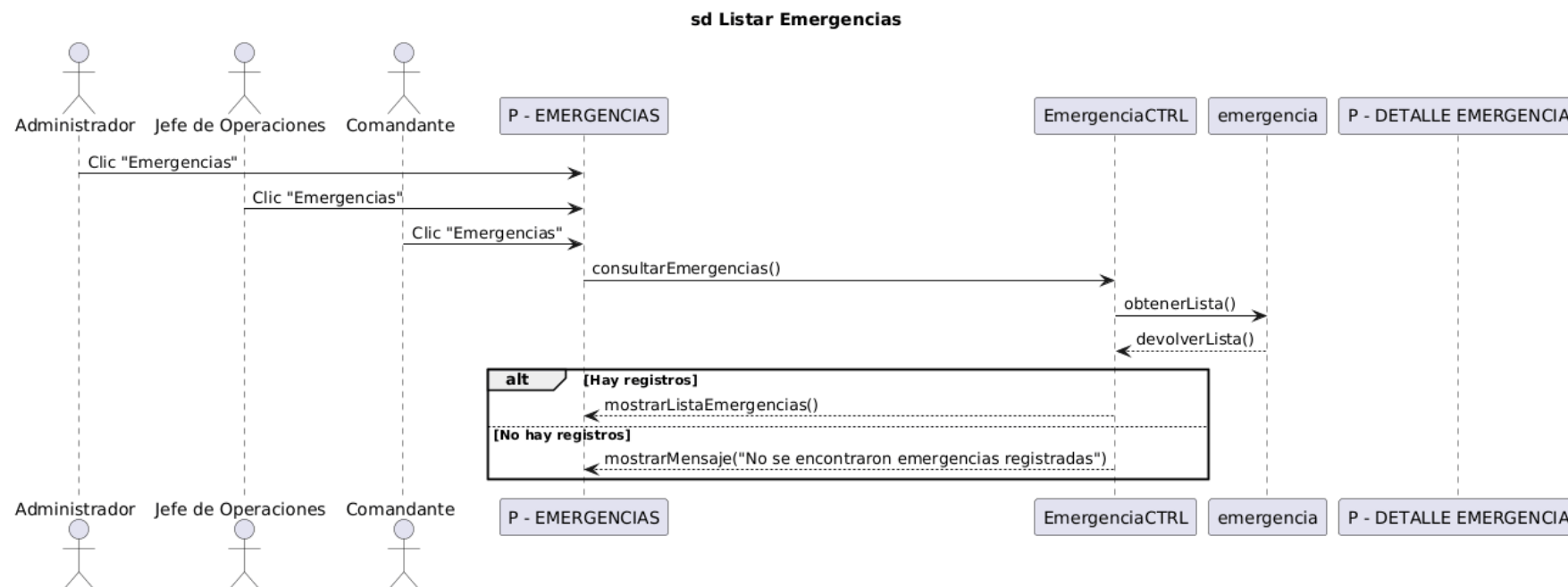


Figura 77. Diagrama de Secuencia: Gestionar Emergencia-Listar Emergencia
Fuente: Elaboración Propia.

- Registrar Emergencias

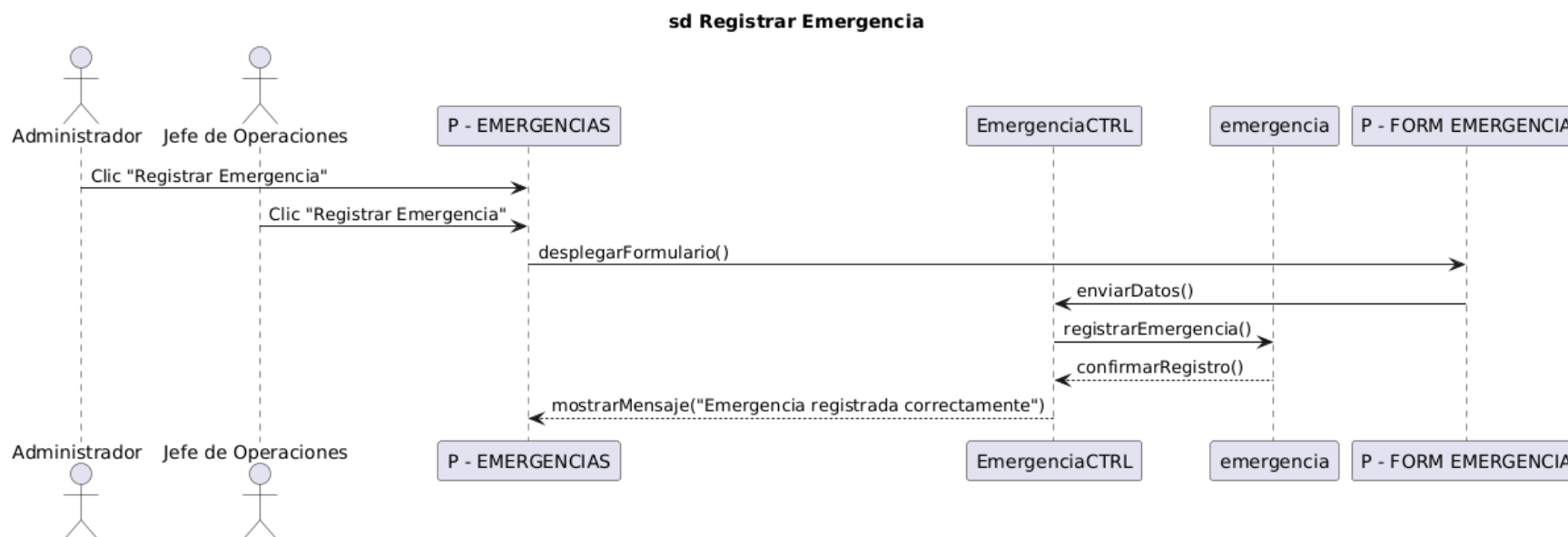


Figura 78. Diagrama de Secuencia: Gestionar Emergencia-Registrar Emergencia
Fuente: Elaboración Propia.

- Editar / Actualizar Emergencia

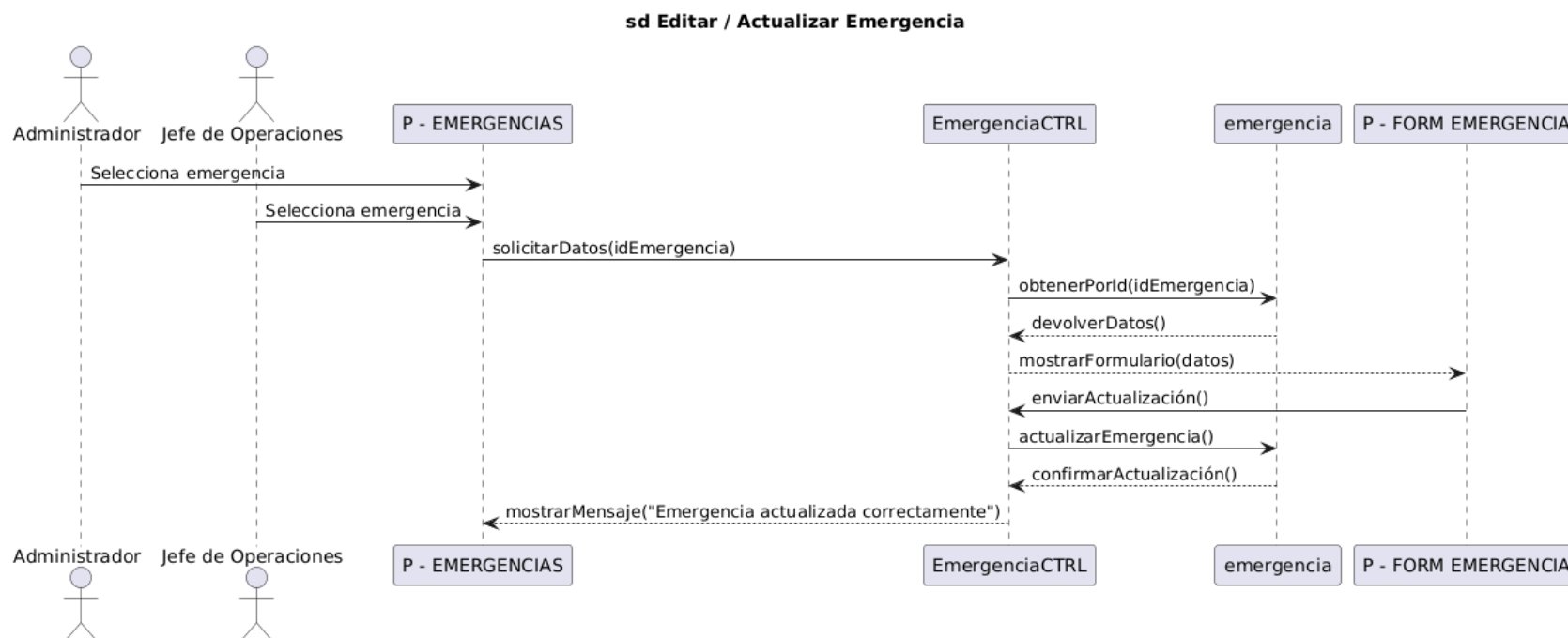


Figura 79. Diagrama de Secuencia: Gestionar Emergencia- Editar/Actualizar Emergencia
Fuente: Elaboración Propia.

- Asignar Vehículos

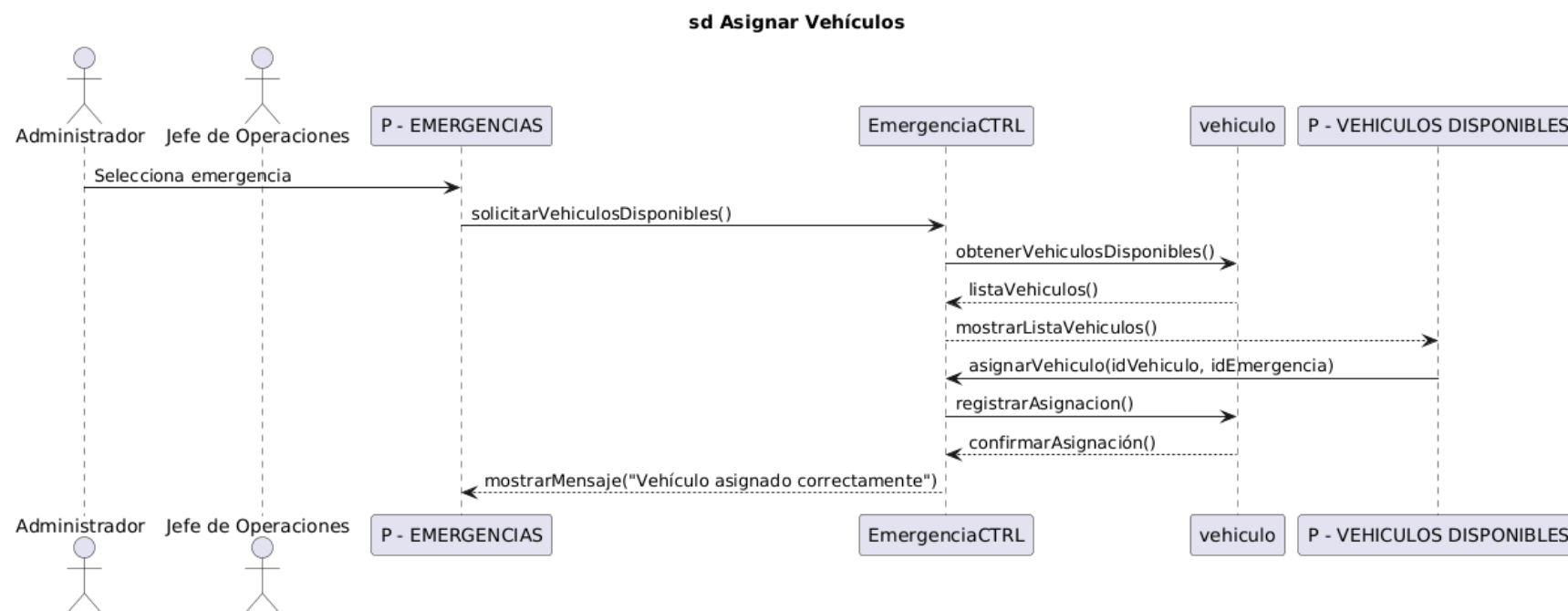


Figura 80. Diagrama de Secuencia: Gestionar Emergencia-Asignar Vehículo
Fuente: Elaboración Propia.

- Asignar Personal

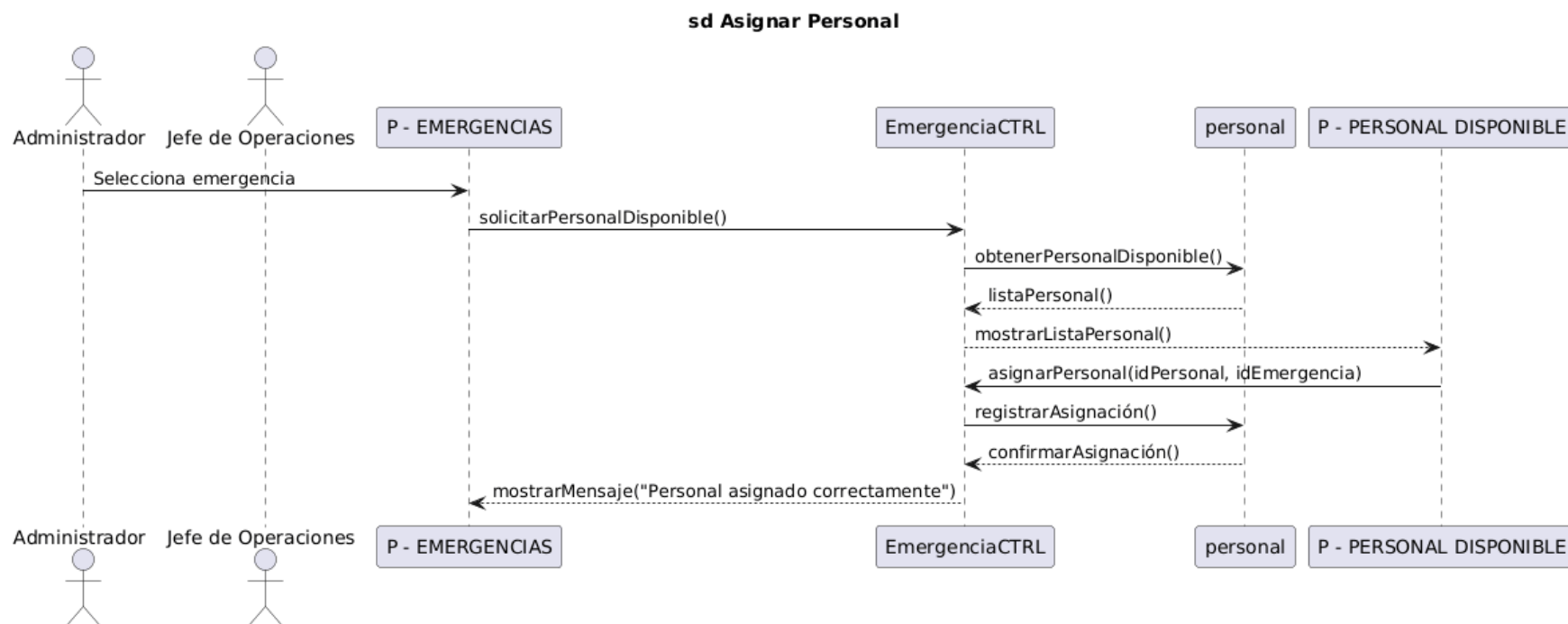


Figura 81. Diagrama de Secuencia: Gestionar Emergencia-Asignar Personal
Fuente: Elaboración Propia.

- Registrar Materiales Utilizados

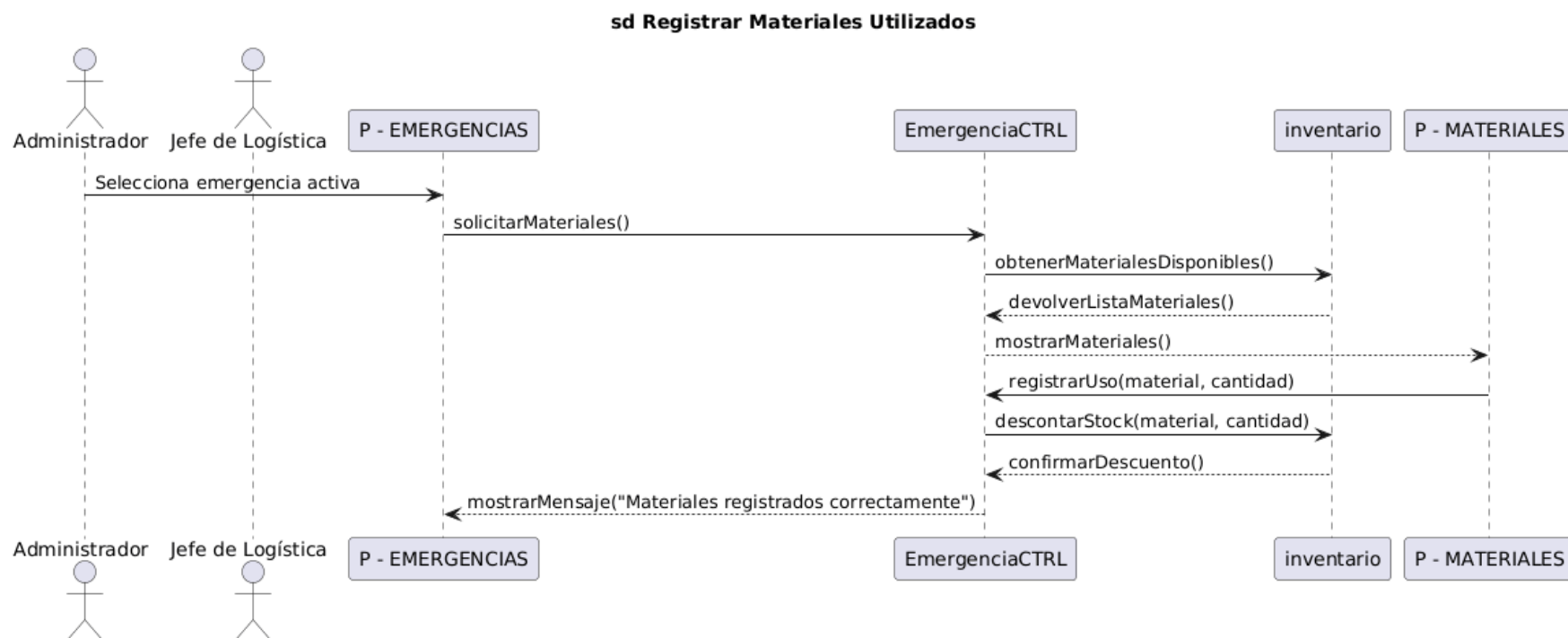


Figura 82. Diagrama de Secuencia: Gestionar Emergencia-Registrar Materiales Utilizados
Fuente: Elaboración Propia.

- Cambiar Estado de Emergencia

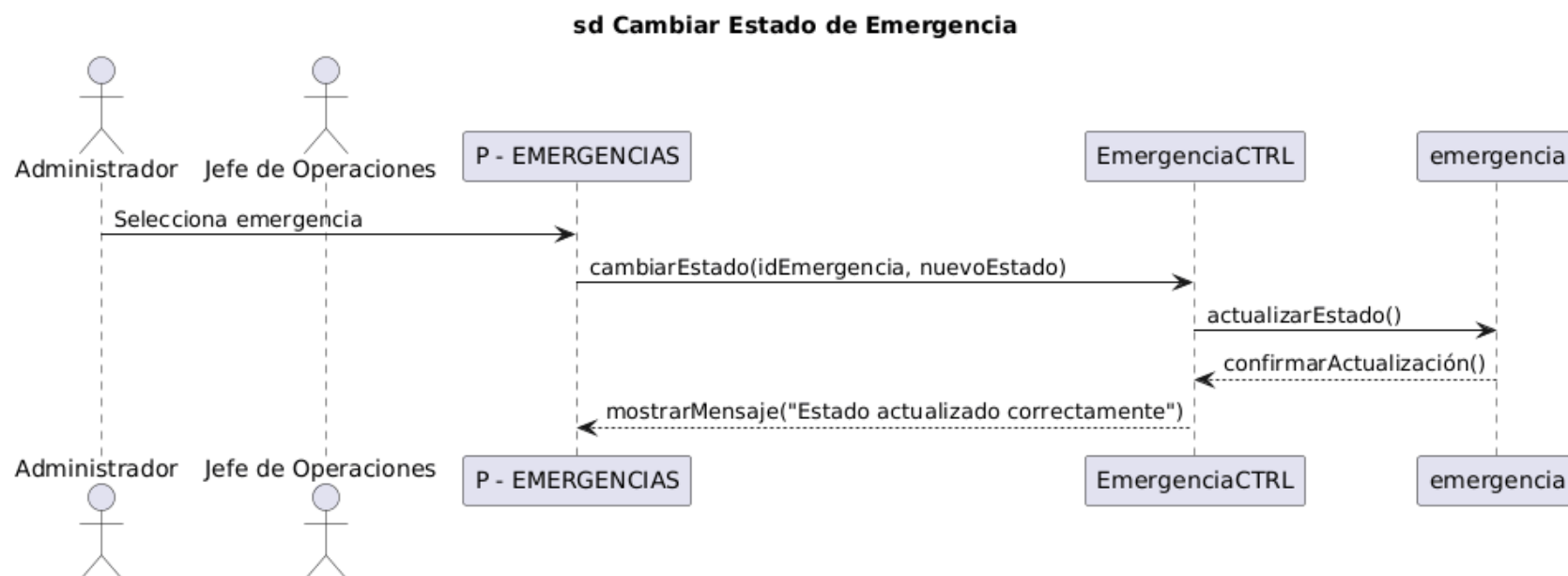


Figura 83. Diagrama de Secuencia: Gestionar Emergencia-Cambiar Estado de Emergencia
Fuente: Elaboración Propia.

- Generar Reportes de Emergencias

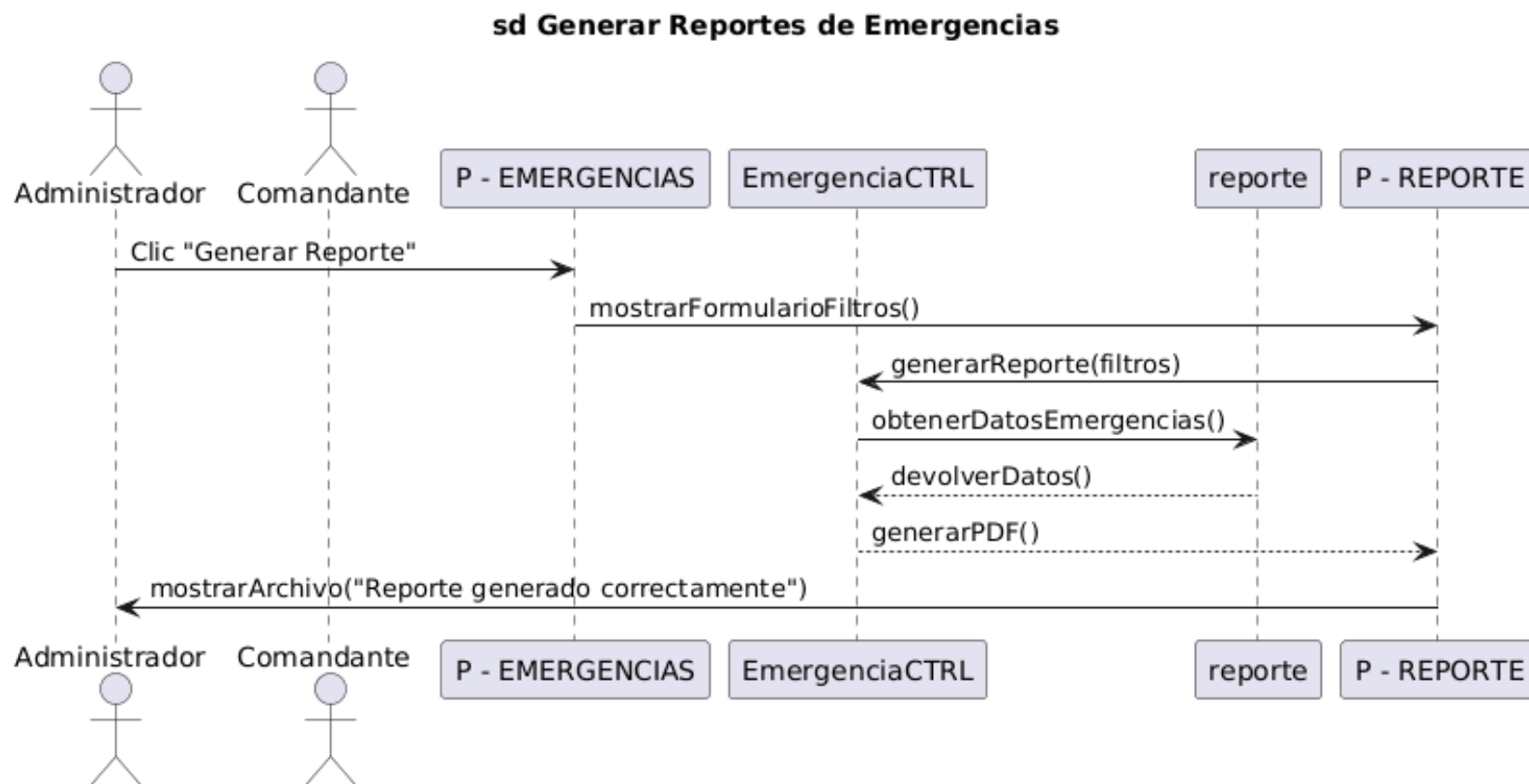


Figura 84. Diagrama de Secuencia: Gestionar Emergencia-Generar Reportes de Emergencia
Fuente: Elaboración Propia.

II.1.2.6.11.3 Diagrama de Secuencia: Gestión de Credenciales QR

- Listar Credenciales QR

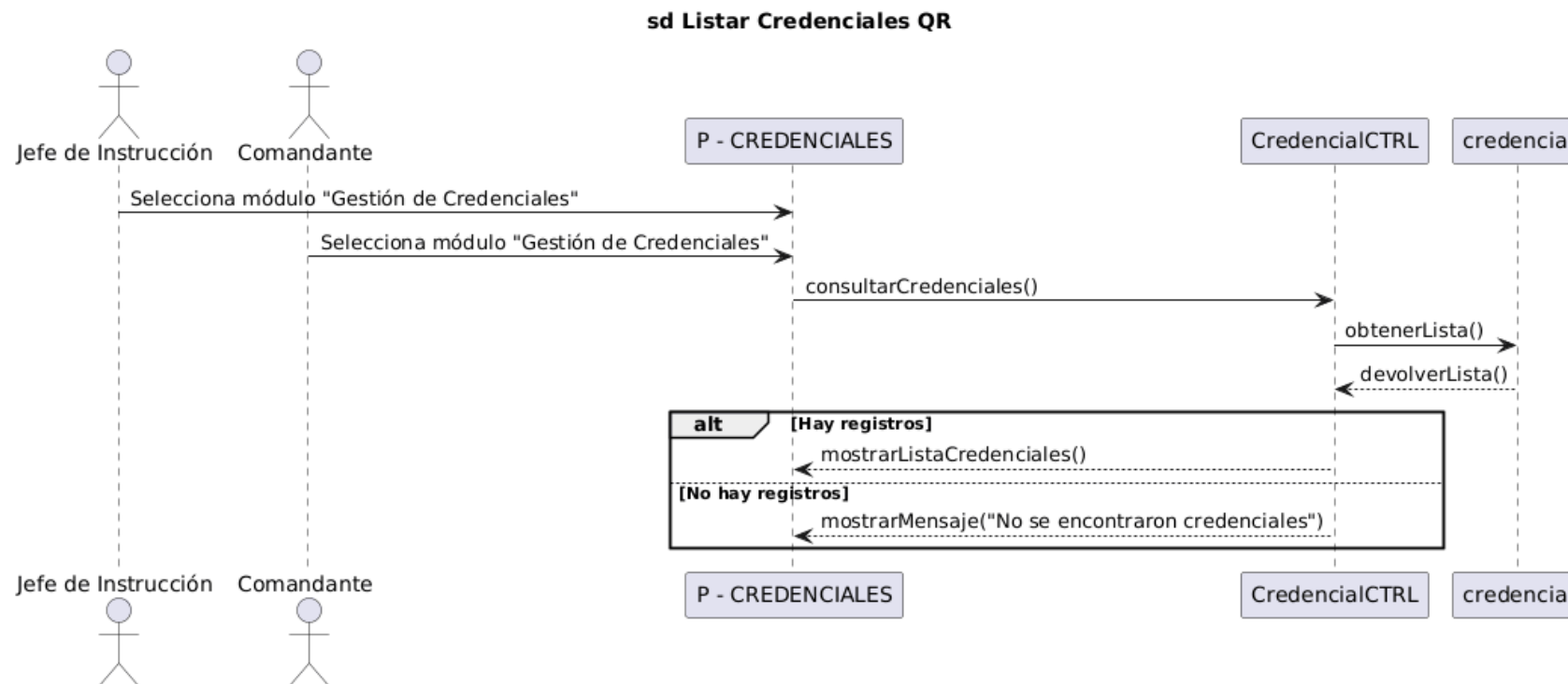


Figura 85. Diagrama de Secuencia: Gestionar Credenciales-Listar Credenciales QR
Fuente: Elaboración Propia.

- Registrar Credenciales

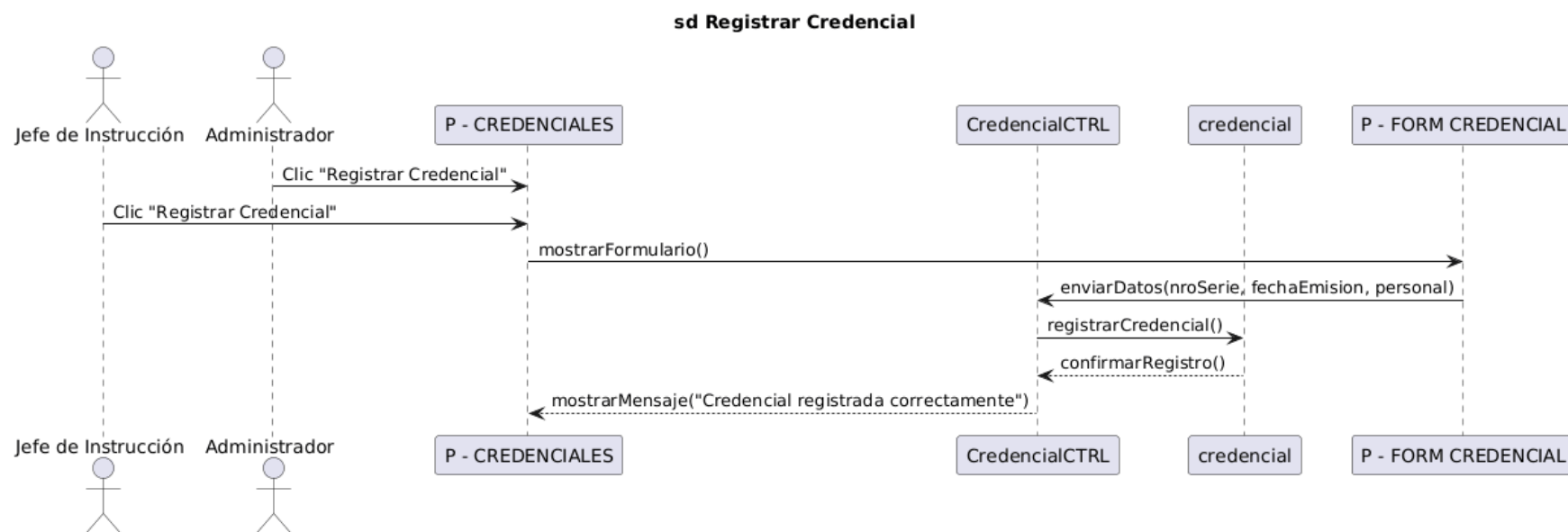


Figura 86. Diagrama de Secuencia: Gestionar Credenciales-Registrar Credenciales
Fuente: Elaboración Propia.

- Actualizar / Suspender / Revocar Credencial

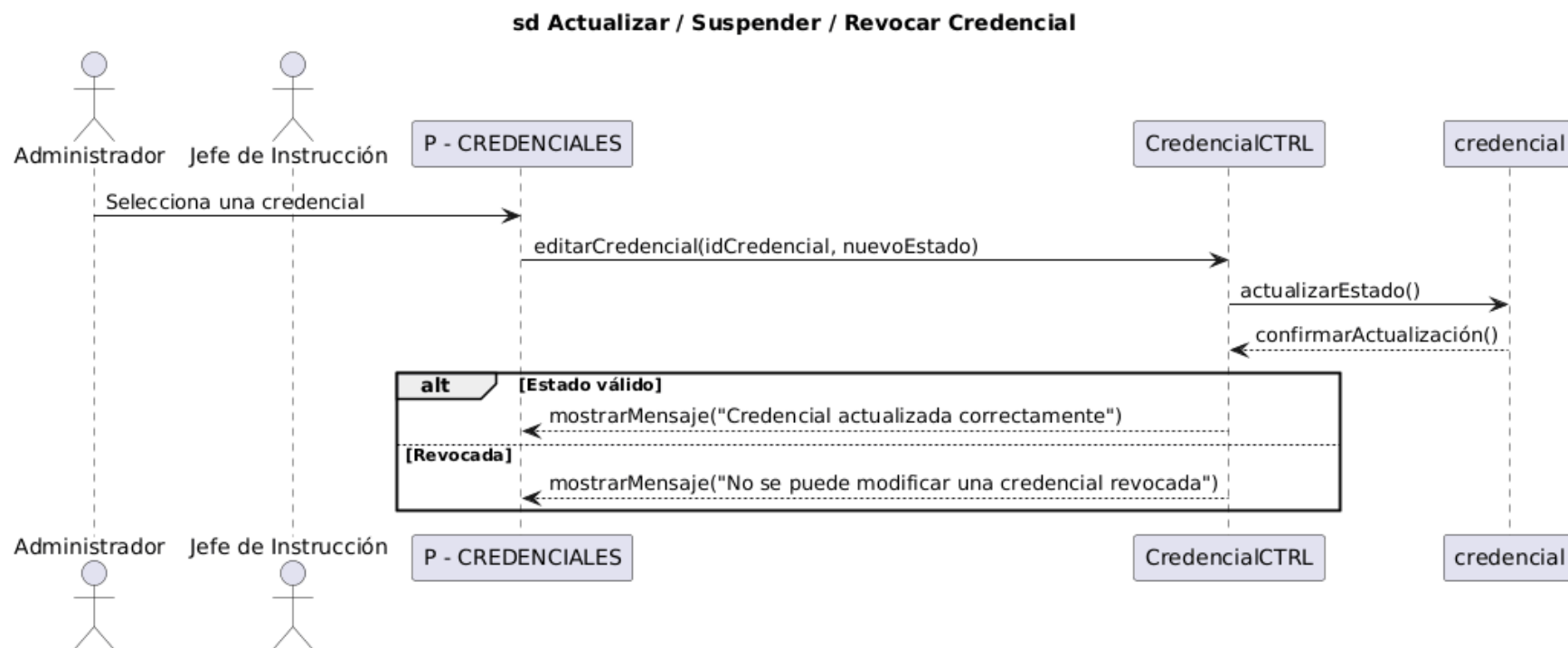


Figura 87. Diagrama de Secuencia: Gestionar Credenciales- Actualizar/Suspender/Revocar Credencial
Fuente: Elaboración Propia.

- Generar Credencial PDF con Código QR

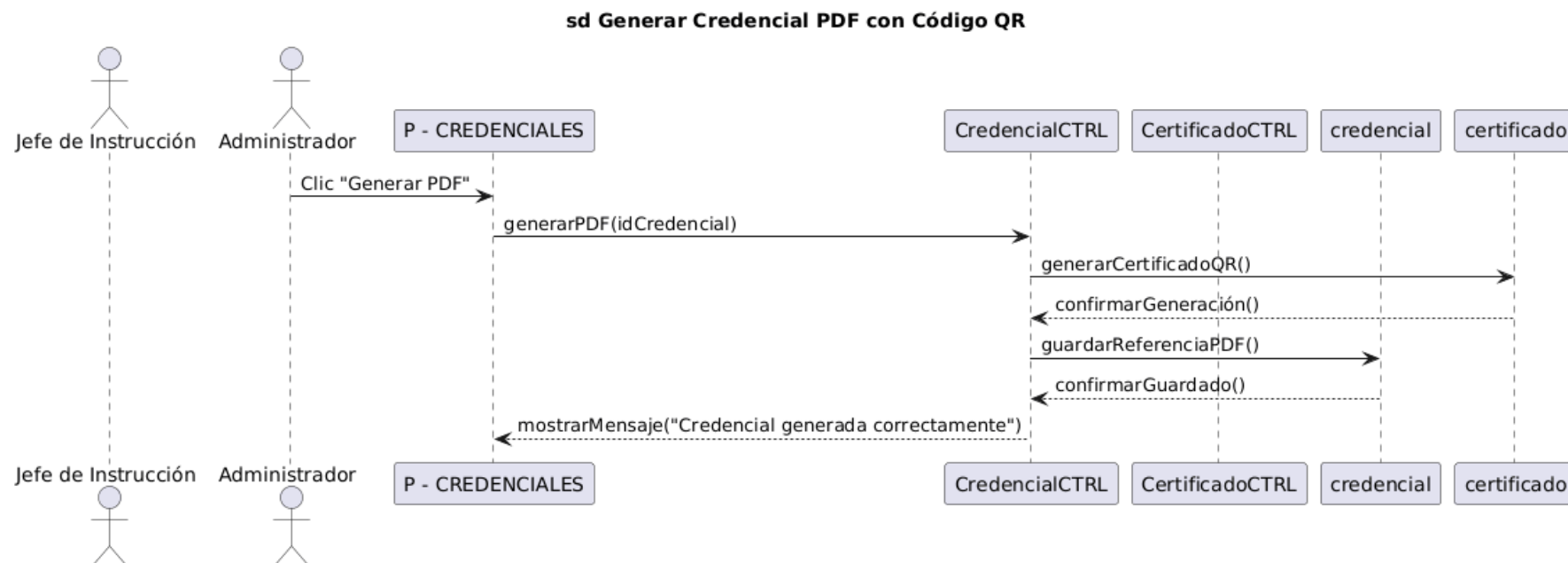


Figura 88. Diagrama de Secuencia: Gestionar Credenciales-Generar Credencial PDF con código QR
Fuente: Elaboración Propia.

- Verificar Credencial por Código QR (Acceso Público)

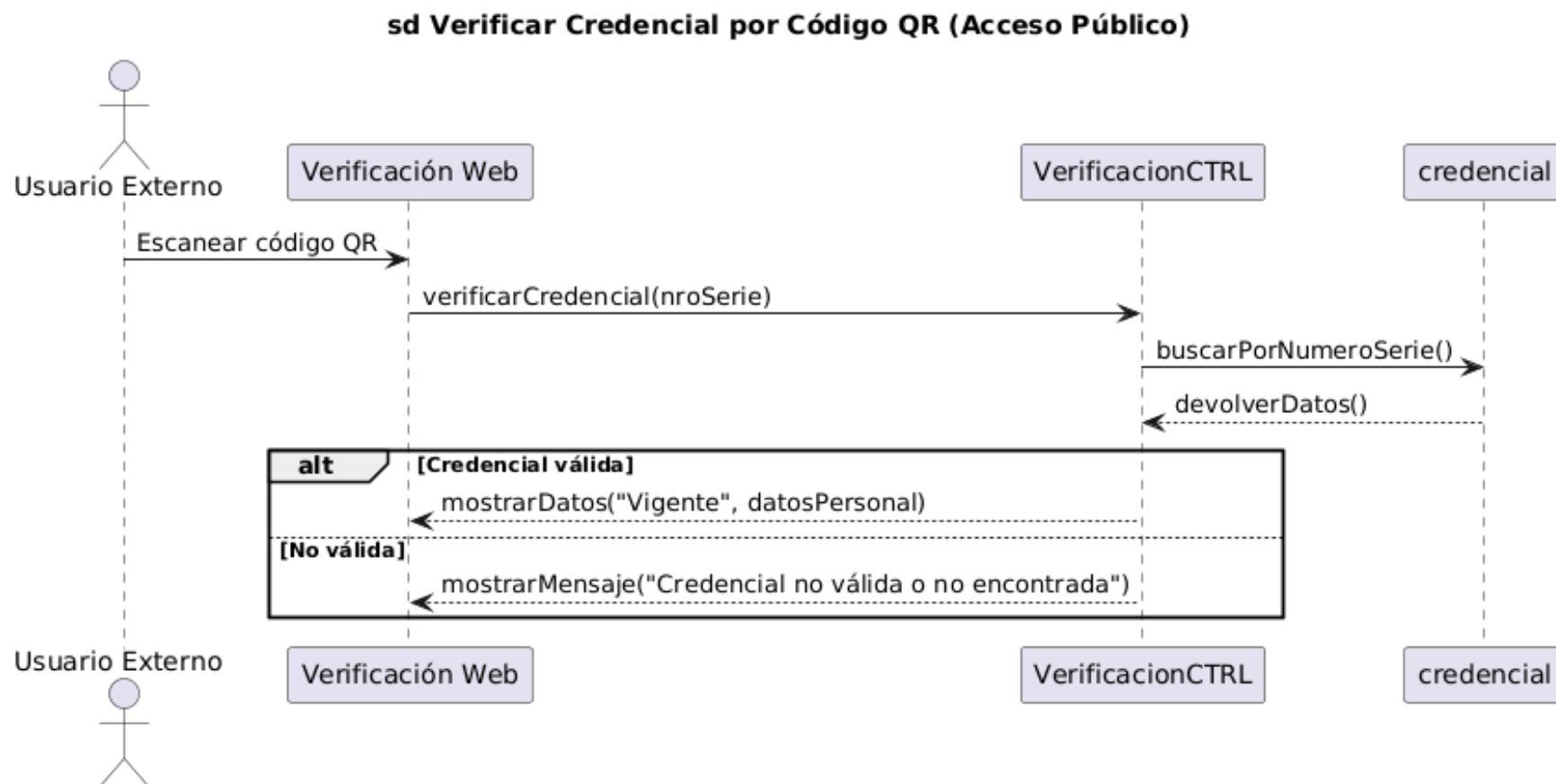


Figura 89. Diagrama de Secuencia: Gestionar Credenciales-Verificar Credencial por Código QR (Acceso Público)
Fuente: Elaboración Propia.

- Generar Reportes de Credenciales

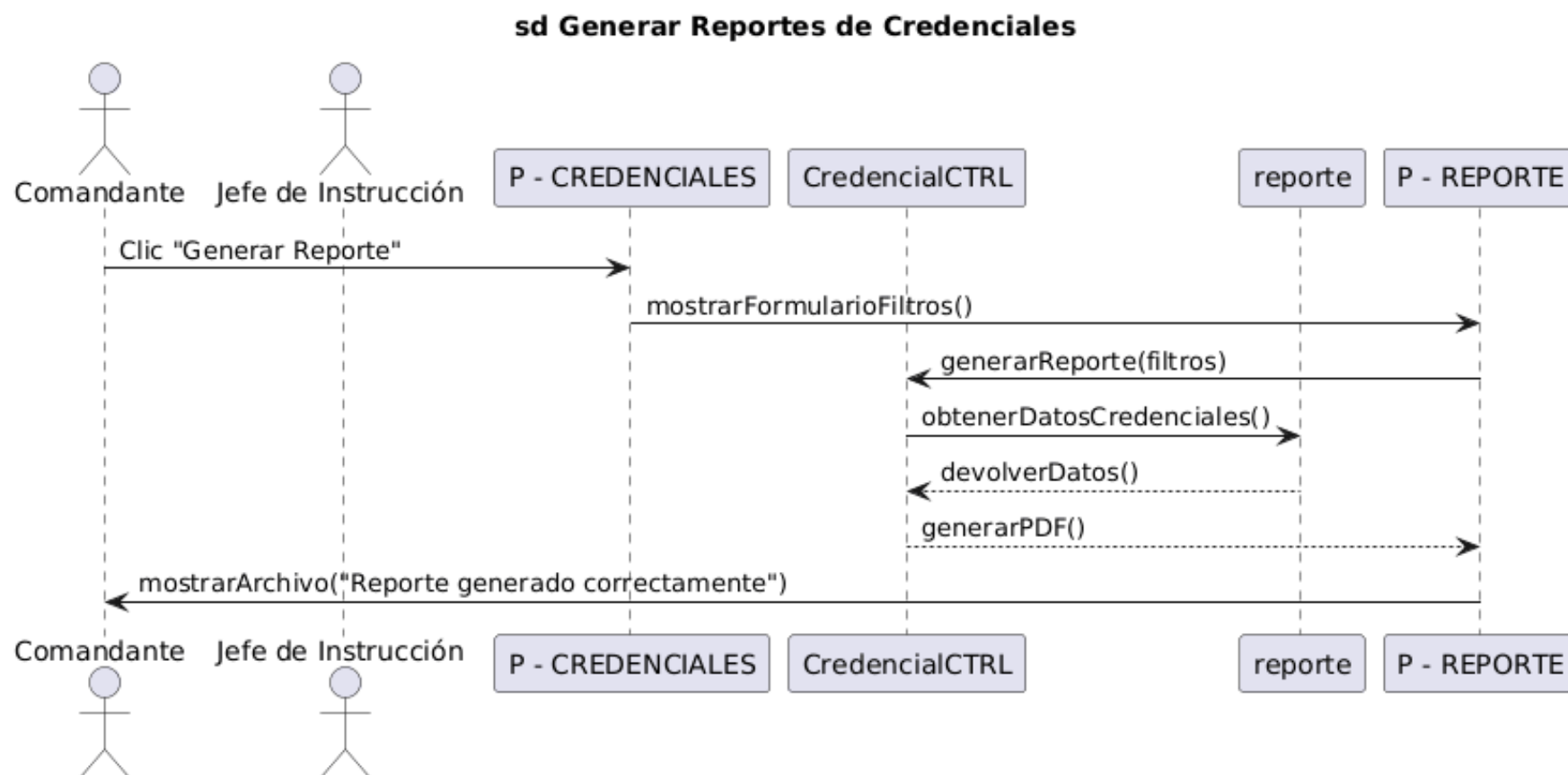


Figura 90. Diagrama de Secuencia: Gestionar Credenciales-Generar Reportes de Credenciales
Fuente: Elaboración Propia.

II.1.2.6.11.4 Diagrama de Secuencia: Gestión de Capacitaciones

- Listar Capacitaciones

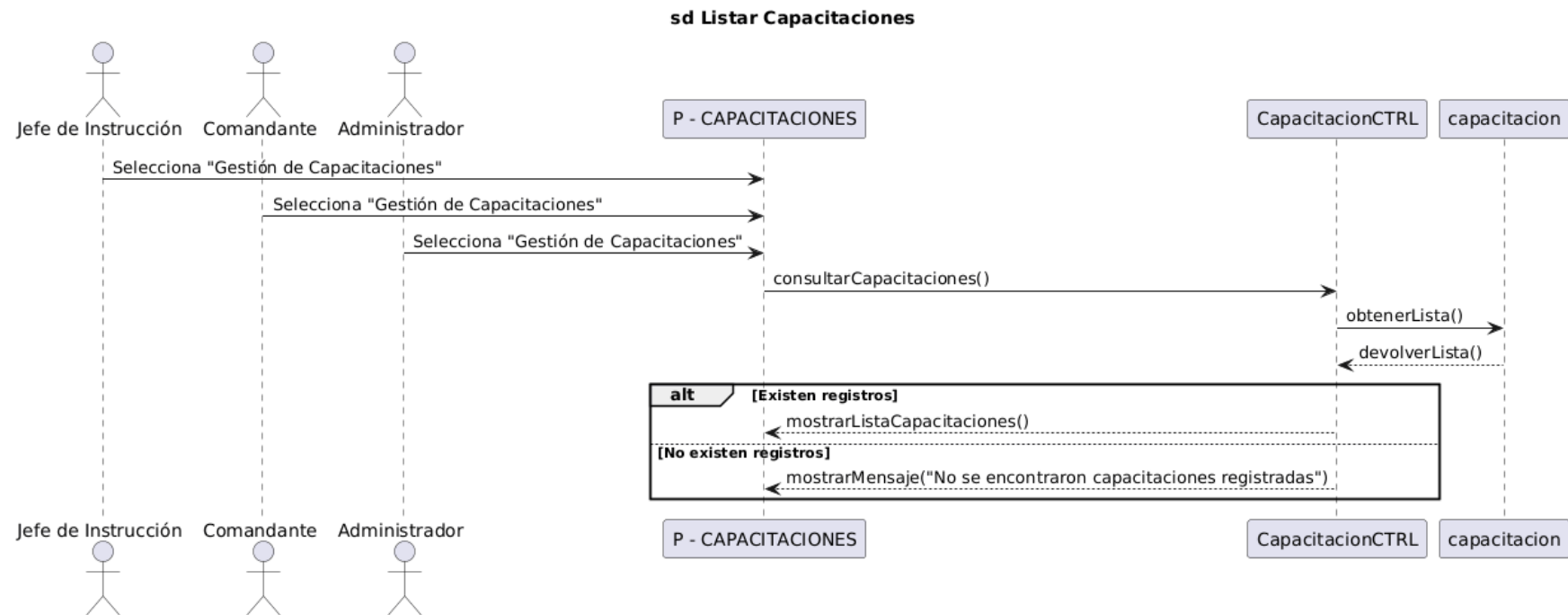


Figura 91. Diagrama de Secuencia: Gestionar Capacitaciones-Listar Capacitaciones
Fuente: Elaboración Propia.

- Registrar Capacitación

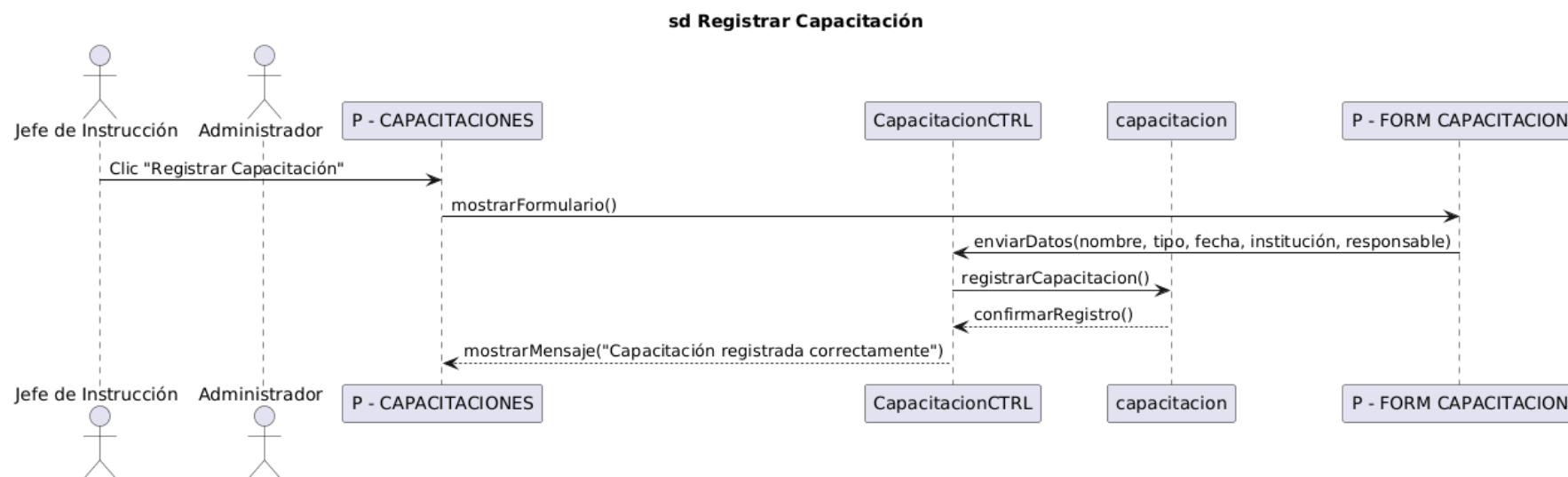


Figura 92. Diagrama de Secuencia: Gestionar Capacitaciones-Registrar Capacitacion
Fuente: Elaboración Propia.

- Registrar Institución Externa

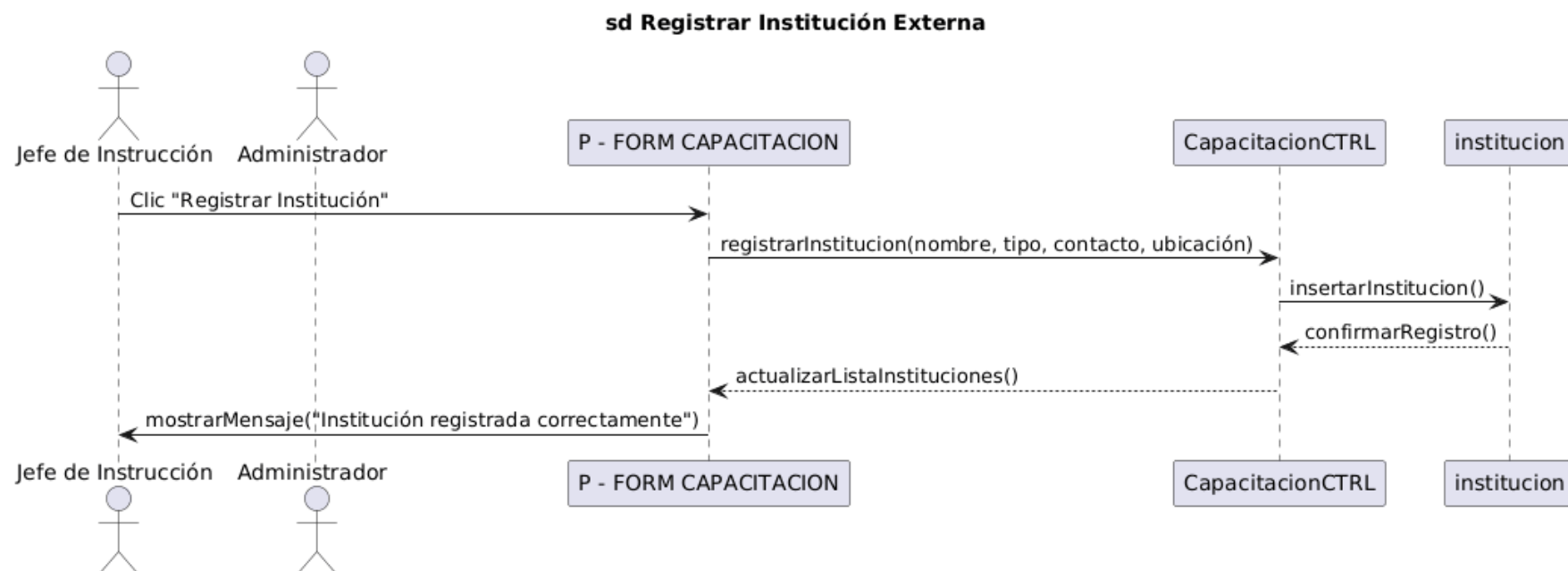


Figura 93. Diagrama de Secuencia: Gestionar Capacitaciones-Registrar Institución Externa
Fuente: Elaboración Propia.

- Registrar Participantes (Internos / Externos)

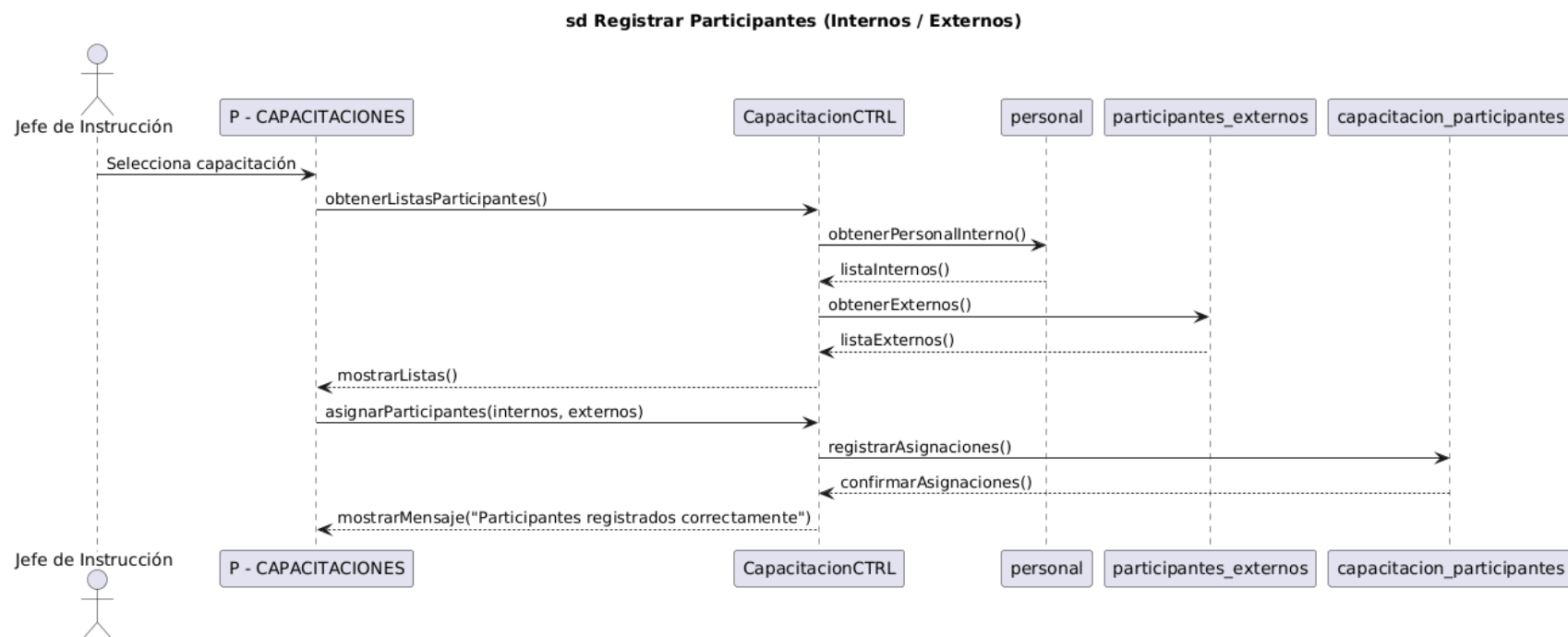


Figura 94. Diagrama de Secuencia: Gestionar Capacitaciones-Registrar Participantes (Internos/Externos)
Fuente: Elaboración Propia.

- Generar Certificados PDF con Código QR

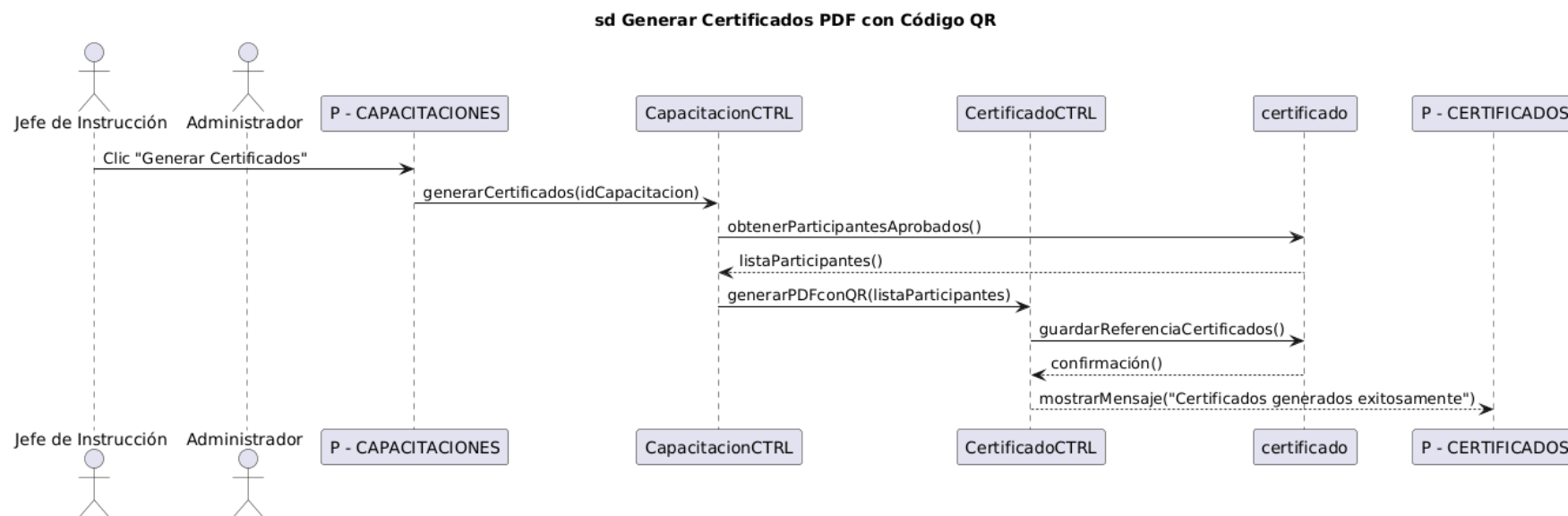


Figura 95. Diagrama de Secuencia: Gestionar Capacitaciones-Generar Certificados PDF con código QR
Fuente: Elaboración Propia.

- Generar Reportes y Estadísticas

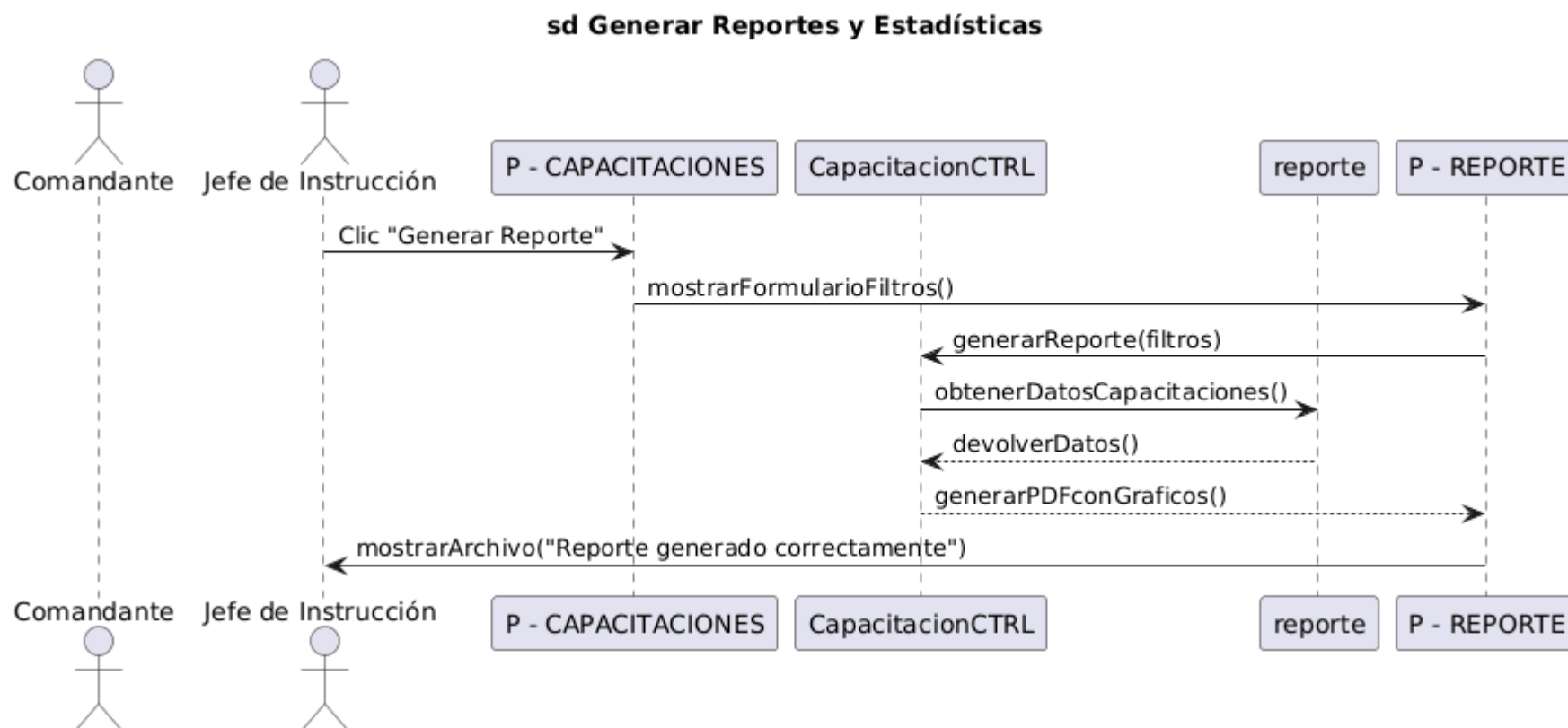


Figura 96. Diagrama de Secuencia: Gestionar Capacitaciones-Generar Reportes y Estadísticas
Fuente: Elaboración Propia.

II.1.2.6.11.5 Diagrama de Secuencia: Gestión de Vehículos

- Listar Vehículos

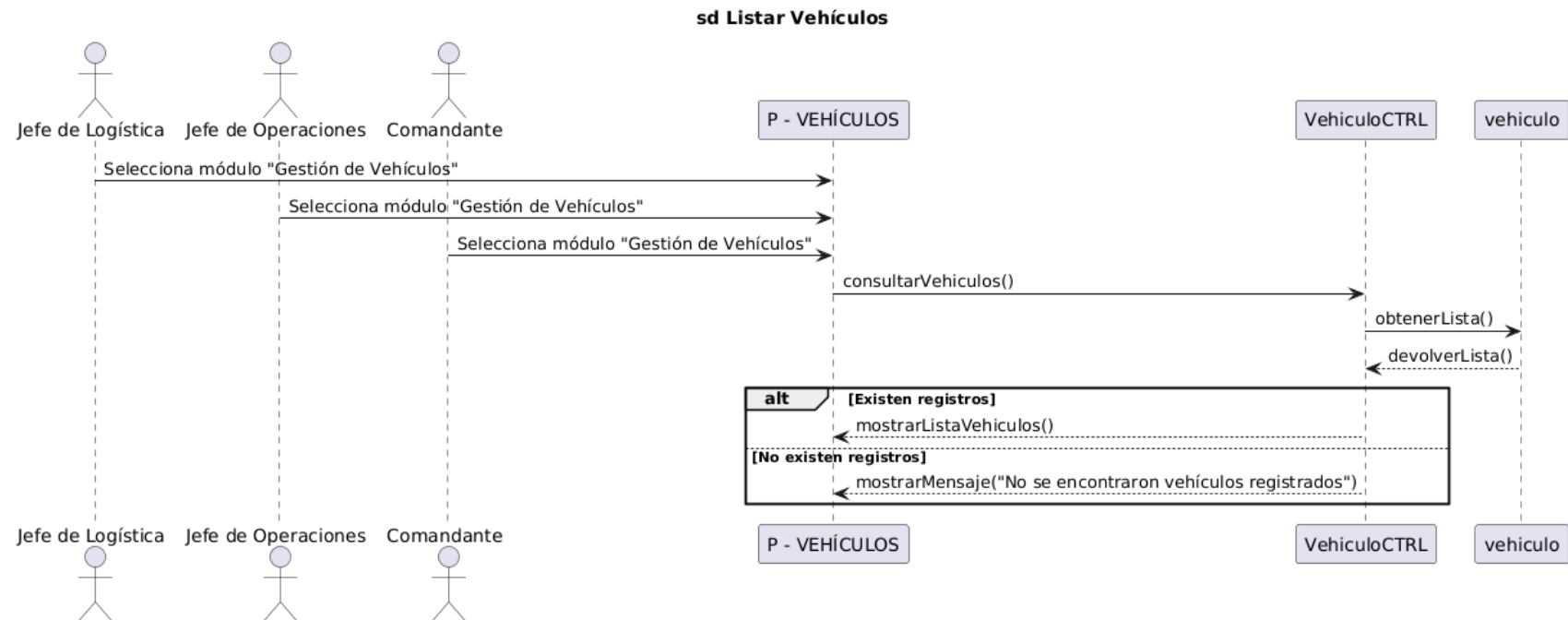


Figura 97. Diagrama de Secuencia: Gestionar Vehículos-Listar Vehículos
Fuente: Elaboración Propia.

- Registrar Vehículos

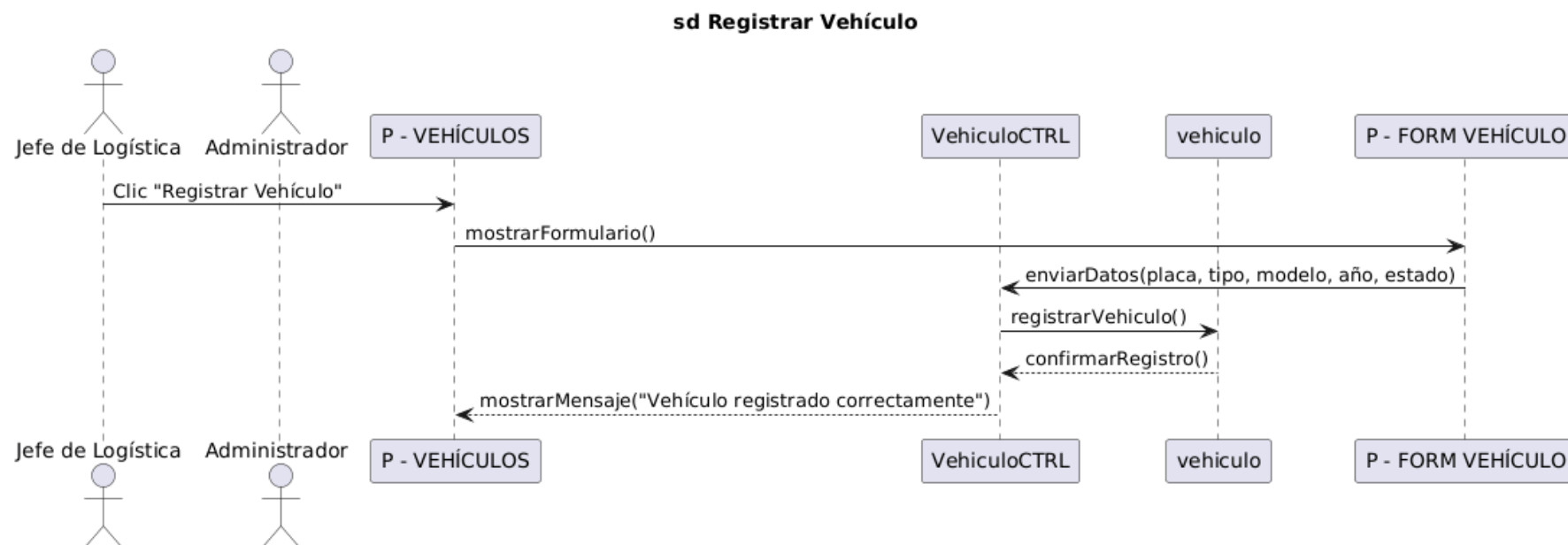


Figura 98. Diagrama de Secuencia: Gestionar Vehículos-Registrar Vehículos
Fuente: Elaboración Propia.

- Editar / Actualizar Vehículos

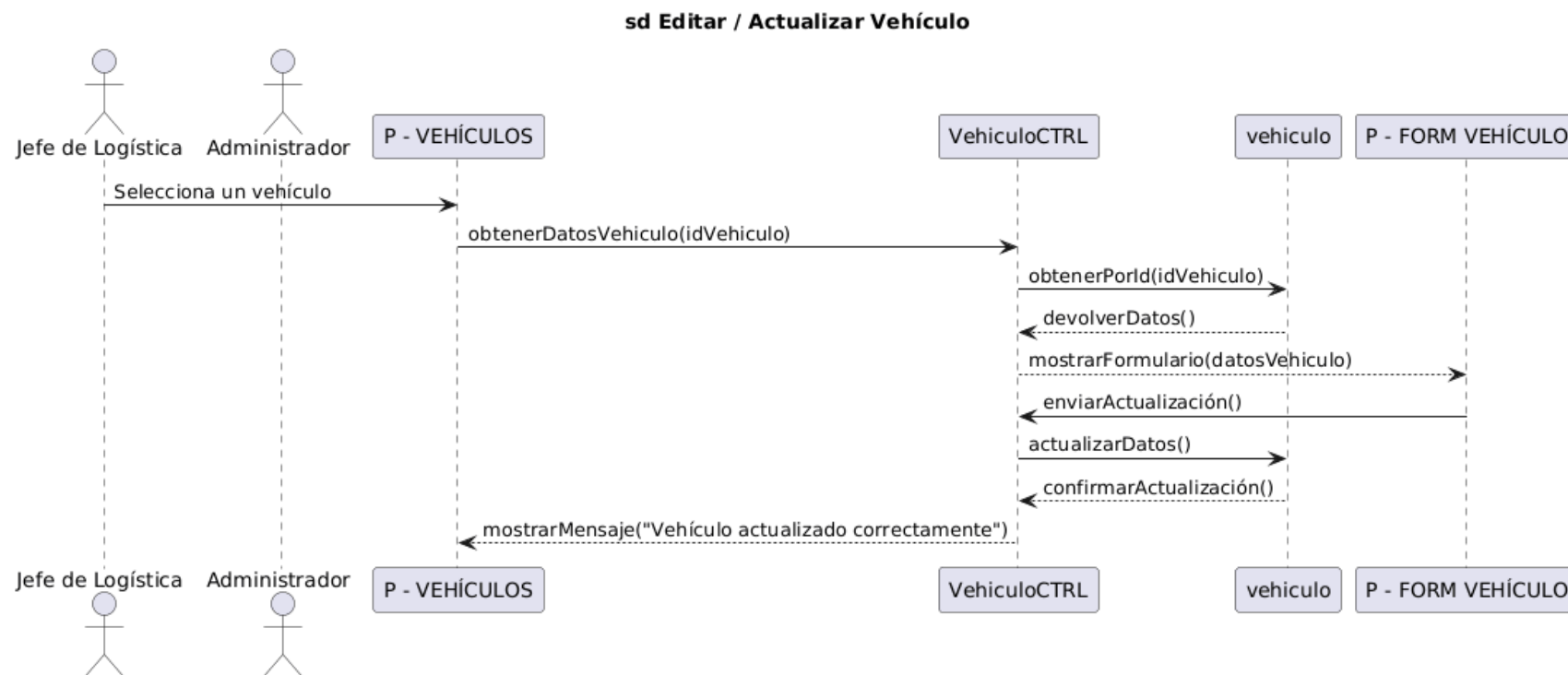


Figura 99. Diagrama de Secuencia: Gestionar Vehículos-Editar/Actualizar Vehículos
Fuente: Elaboración Propia.

- Cambiar Estado Operativo

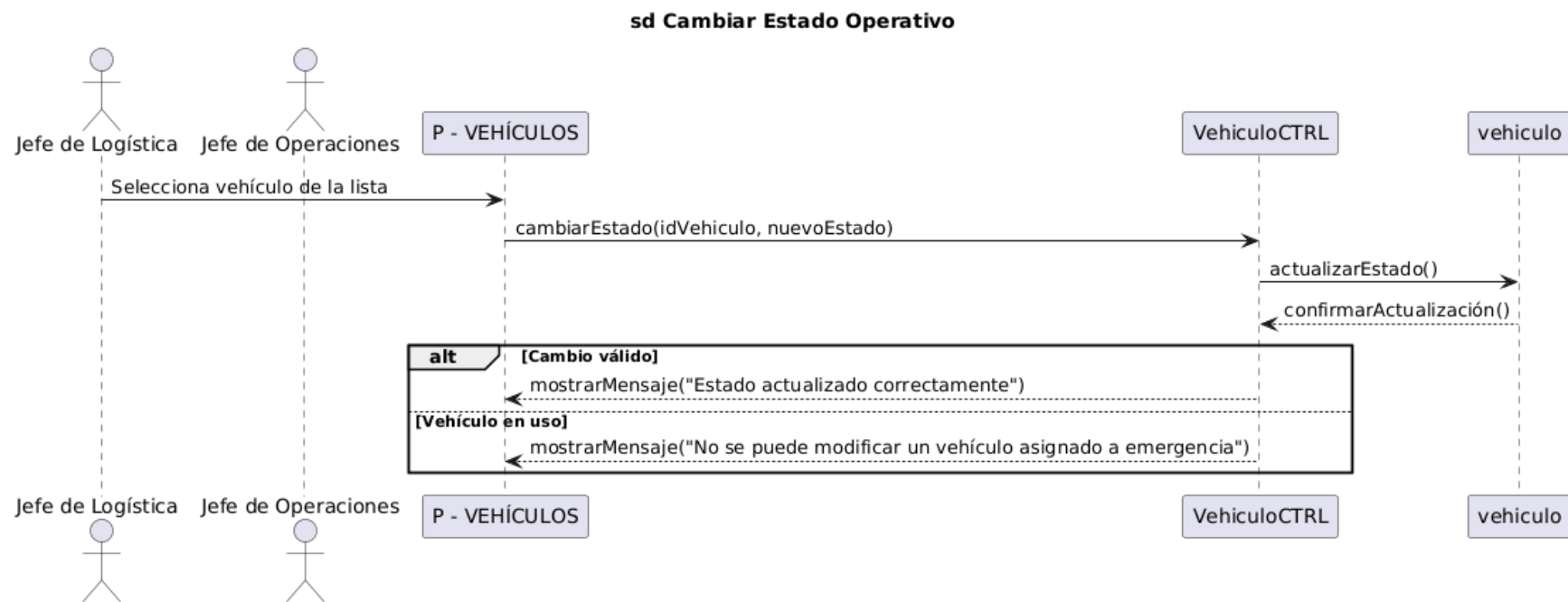


Figura 100. Diagrama de Secuencia: Gestionar Vehículos-Cambiar Estado Operativo
Fuente: Elaboración Propia.

- Registrar Mantenimiento

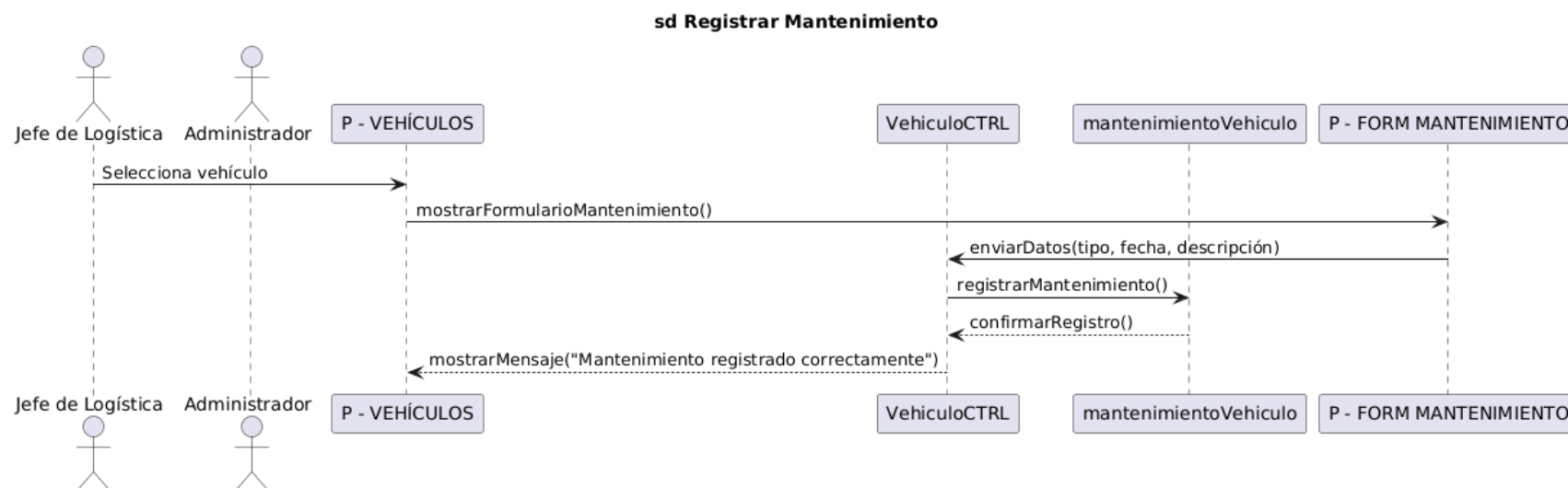


Figura 101. Diagrama de Secuencia: Gestionar Vehículos-Registrar Mantenimiento
Fuente: Elaboración Propia.

- Eliminar Vehículo

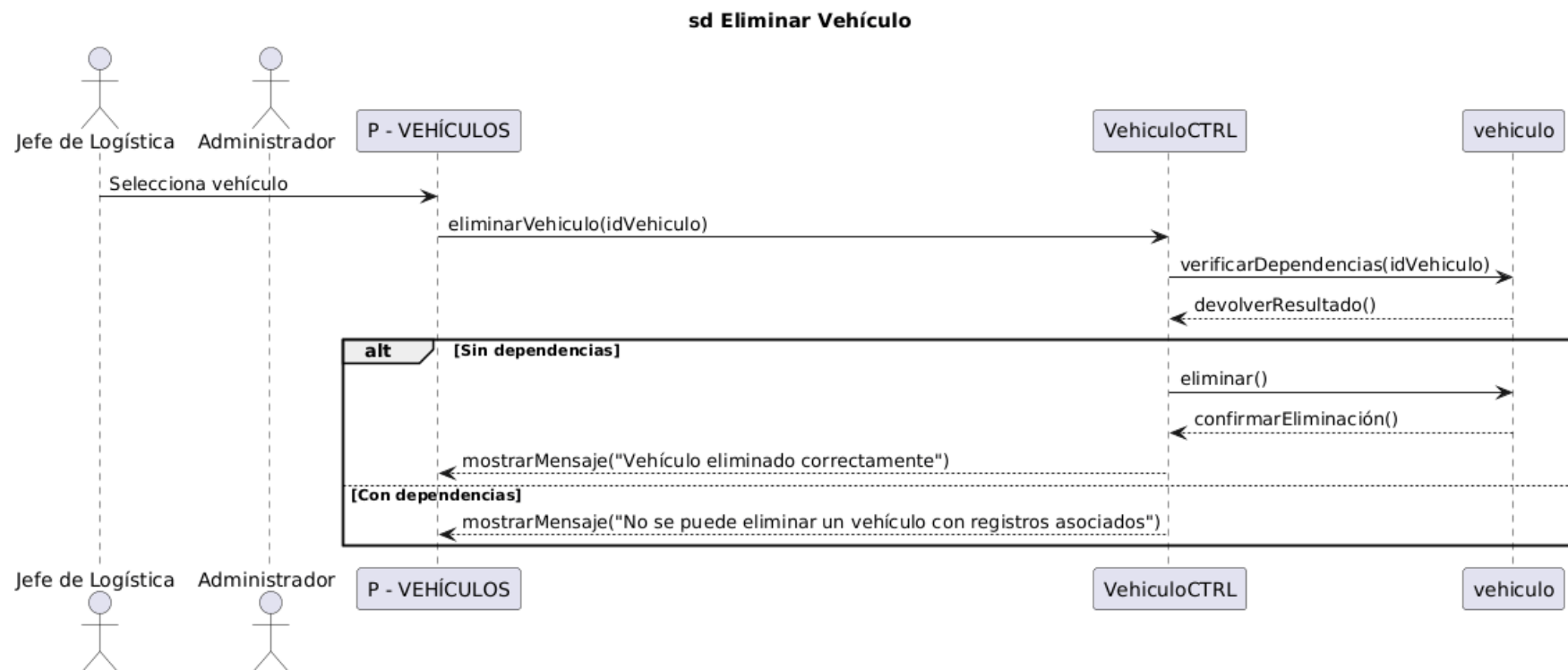


Figura 102. Diagrama de Secuencia: Gestionar Vehículos-Eliminar Vehículos
Fuente: Elaboración Propia.

- Generar Reportes de Vehículo

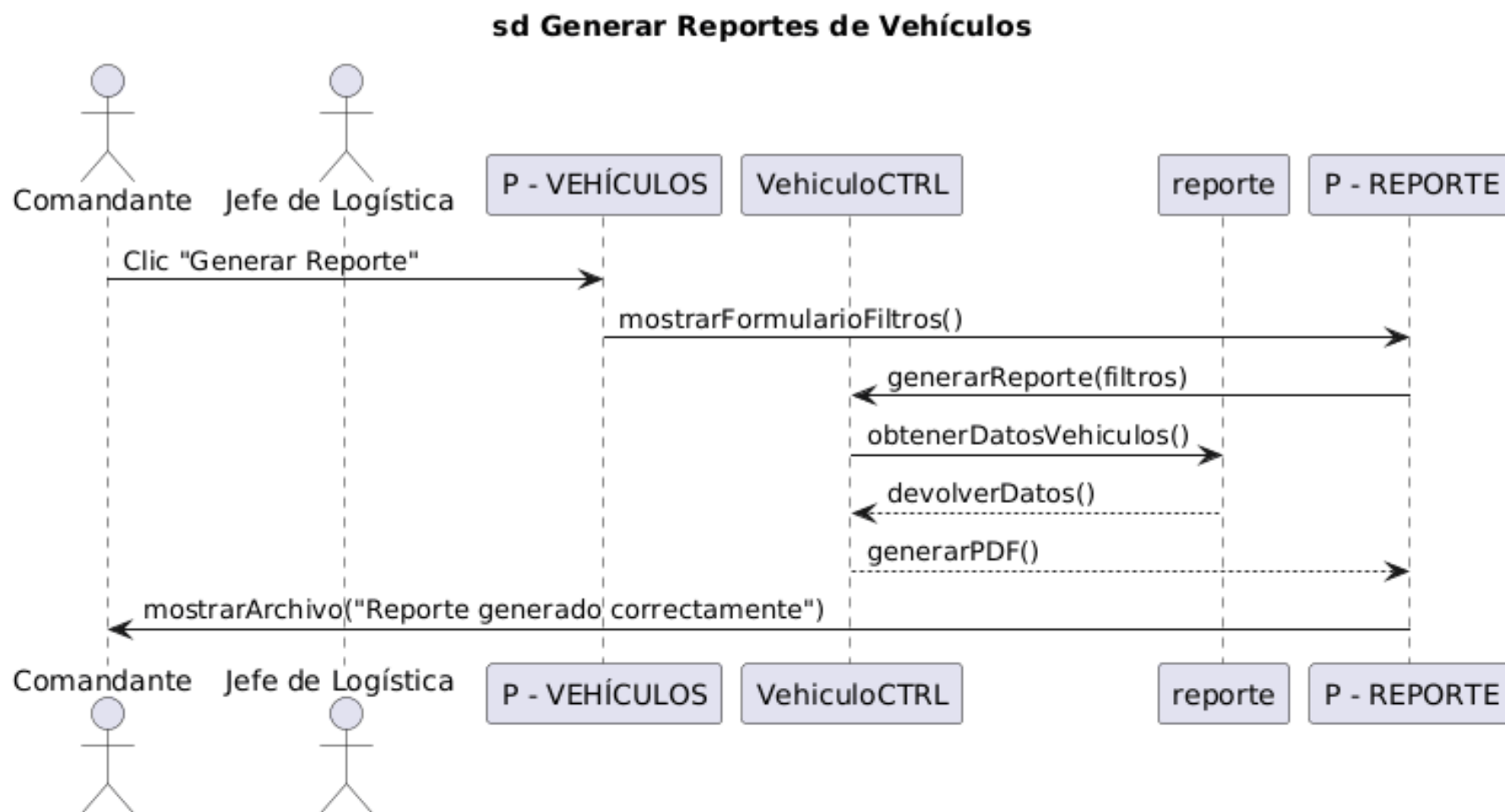


Figura 103. Diagrama de Secuencia: Gestionar Vehículos-Generar Reportes de Vehículos
Fuente: Elaboración Propia.

II.1.2.6.11.6 Diagrama de Secuencia: Gestión de Personal

- Listar Personal

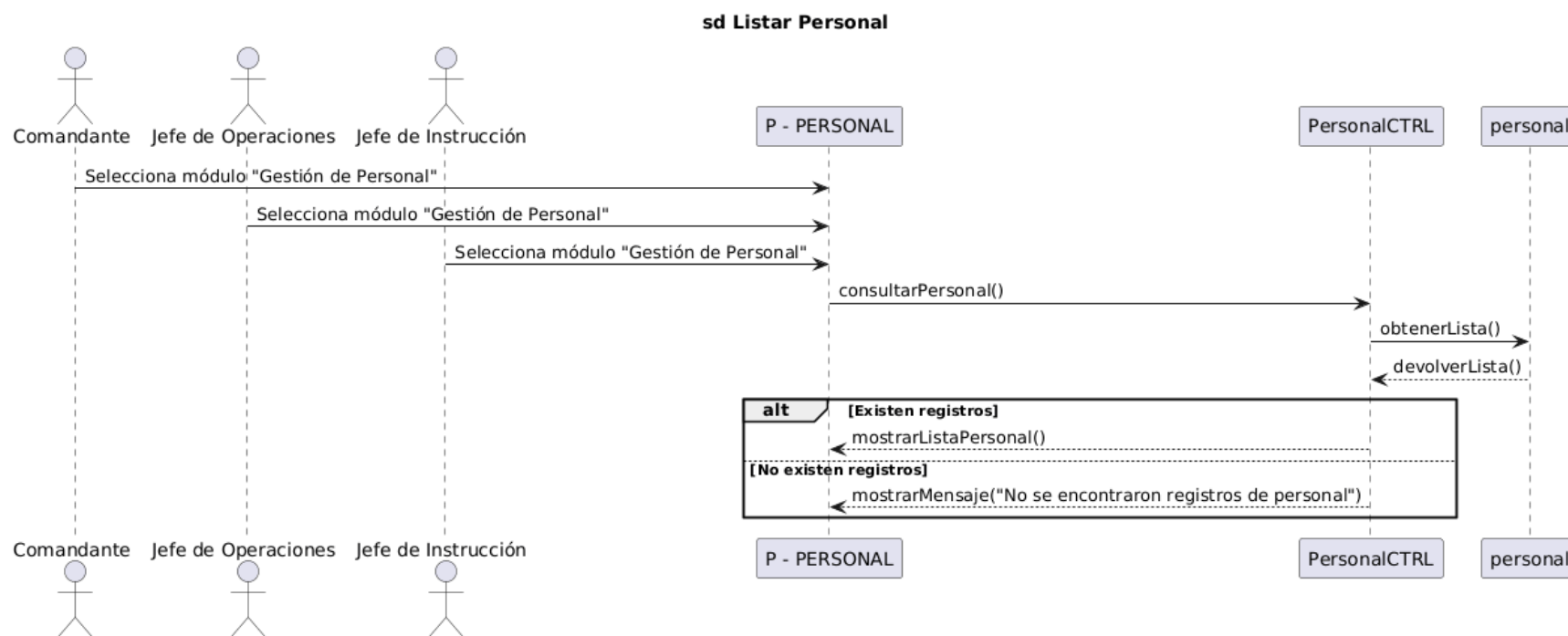


Figura 104. Diagrama de Secuencia: Gestionar Personal-Listar Personal
Fuente: Elaboración Propia.

- Registrar Personal

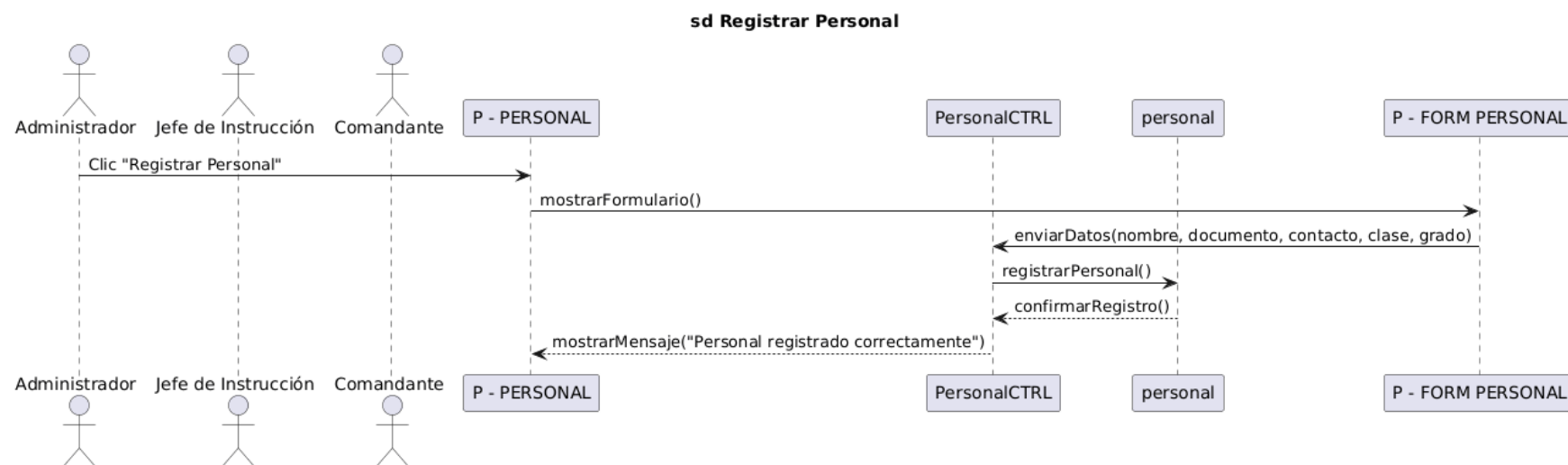


Figura 105. Diagrama de Secuencia: Gestionar Personal-Registrar Personal
Fuente: Elaboración Propia.

- Editar / Actualizar Información de Personal

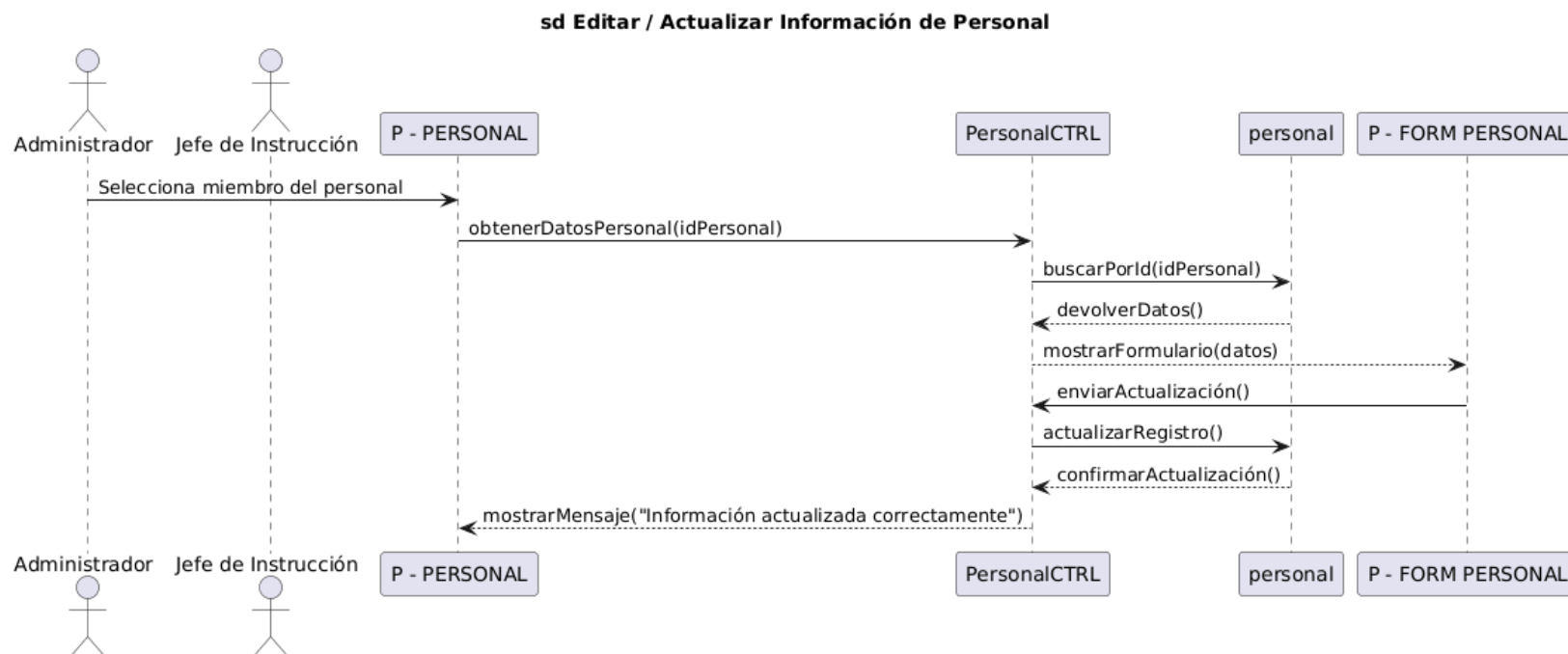


Figura 106. Diagrama de Secuencia: Gestionar Personal-Editar Actualizar información de Personal
Fuente: Elaboración Propia.

- Asignar Grado y Clase

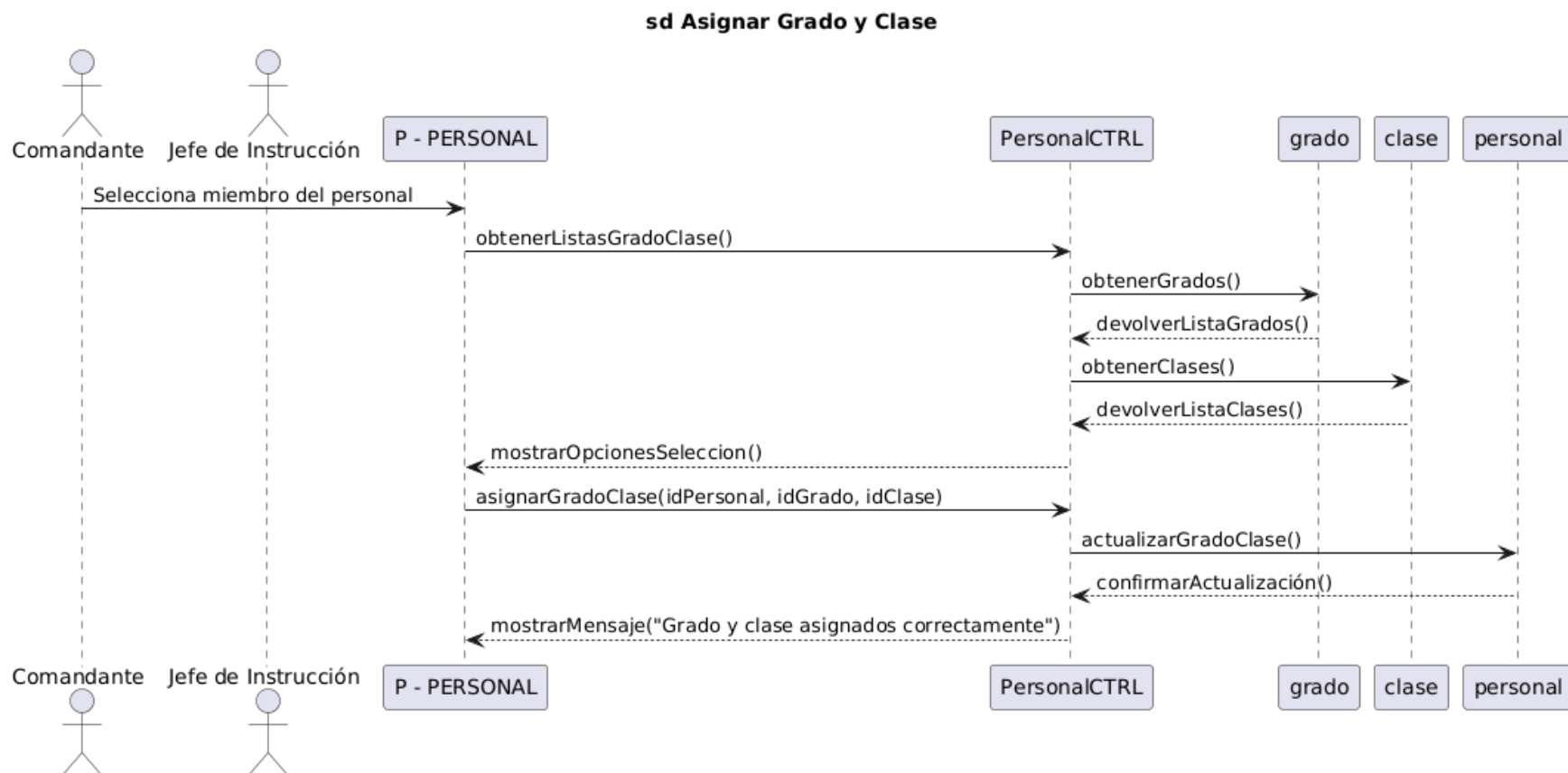


Figura 107. Diagrama de Secuencia: Gestionar Personal-Asignar Grado y Clase
Fuente: Elaboración Propia.

- Cambiar el Estado del Personal

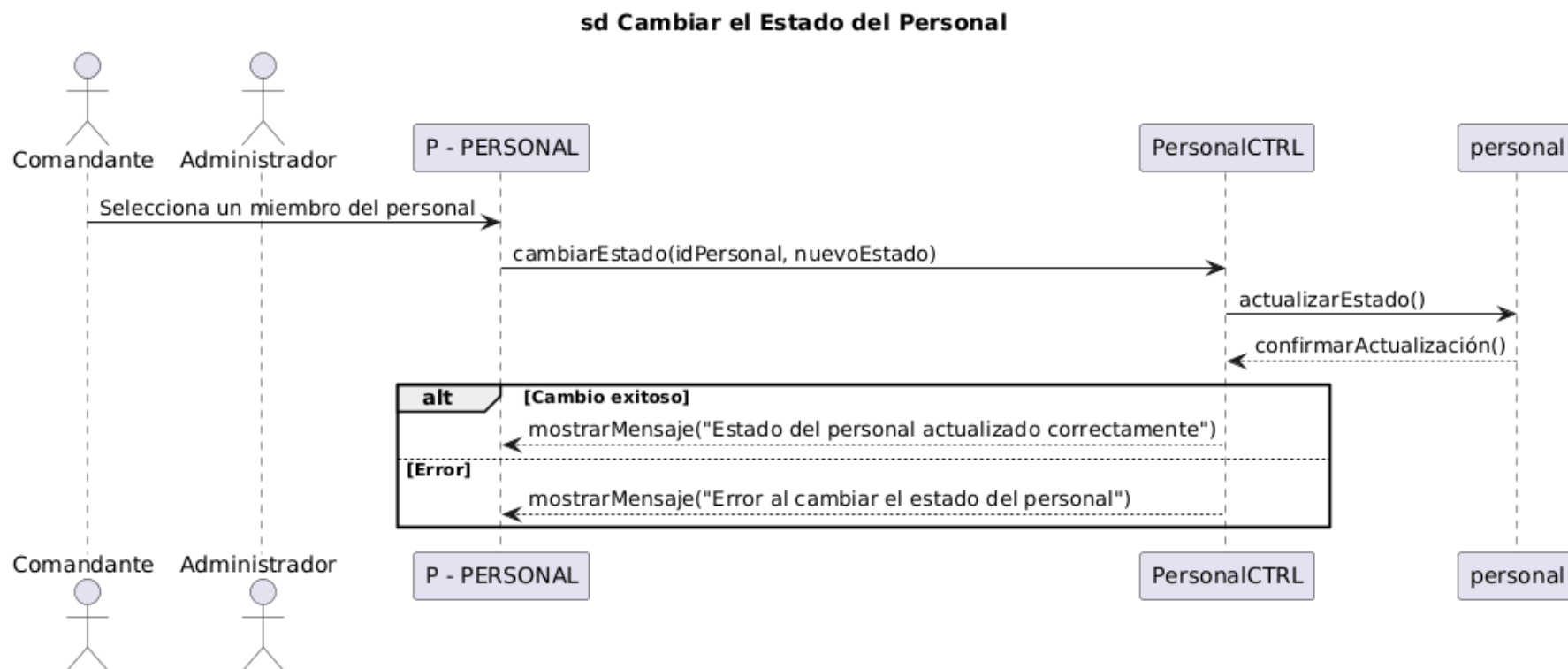


Figura 108. Diagrama de Secuencia: Gestionar Personal-Cambiar el Estado del Personal
Fuente: Elaboración Propia.

- Generar Reportes del Personal

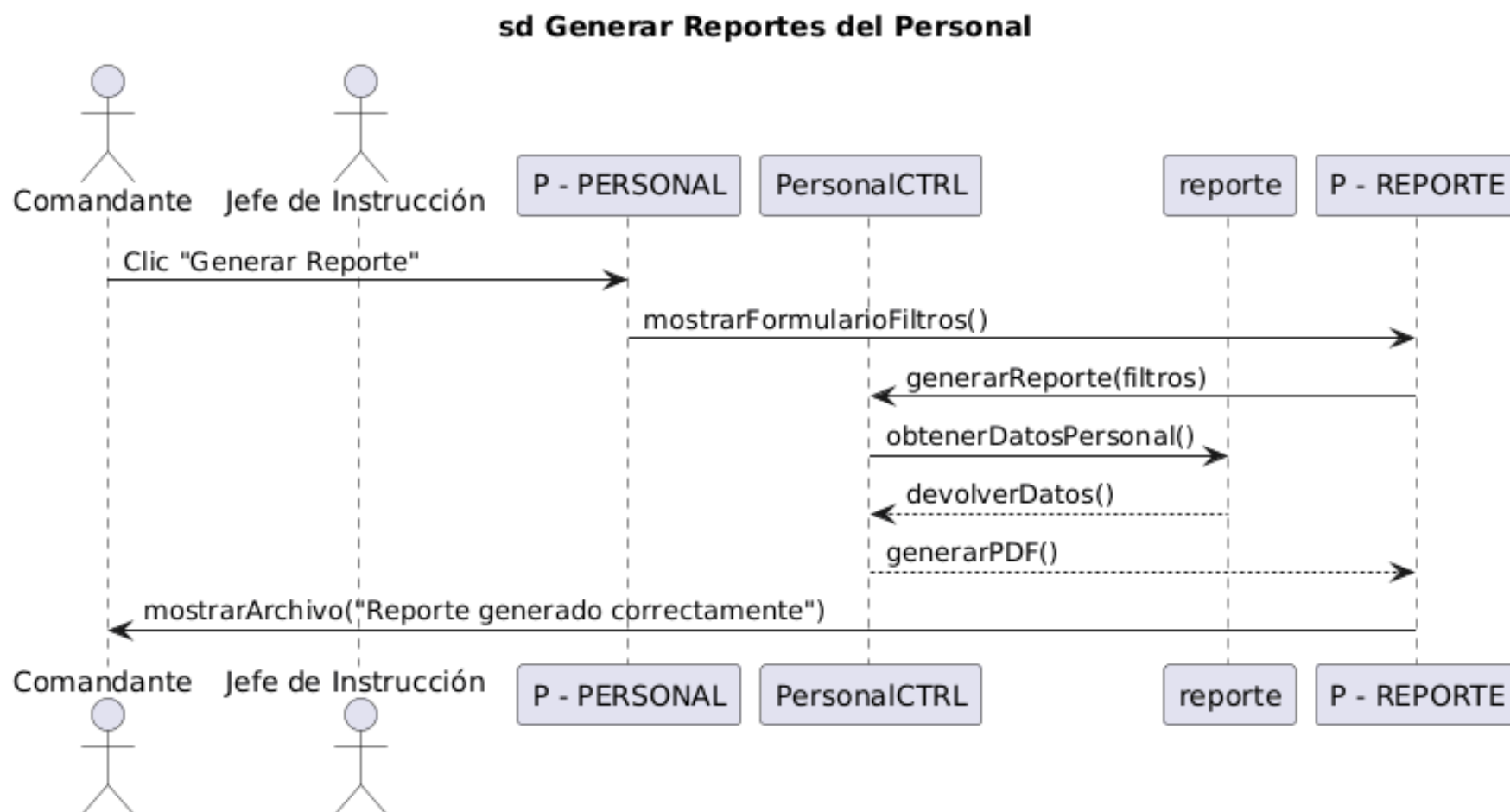


Figura 109. Diagrama de Secuencia: Gestionar Personal-Generar Reportes del Personal
Fuente: Elaboración Propia.

II.1.2.6.11.7 Diagrama de Secuencia: Gestión de Inventario

- Listar Ítems de Inventario

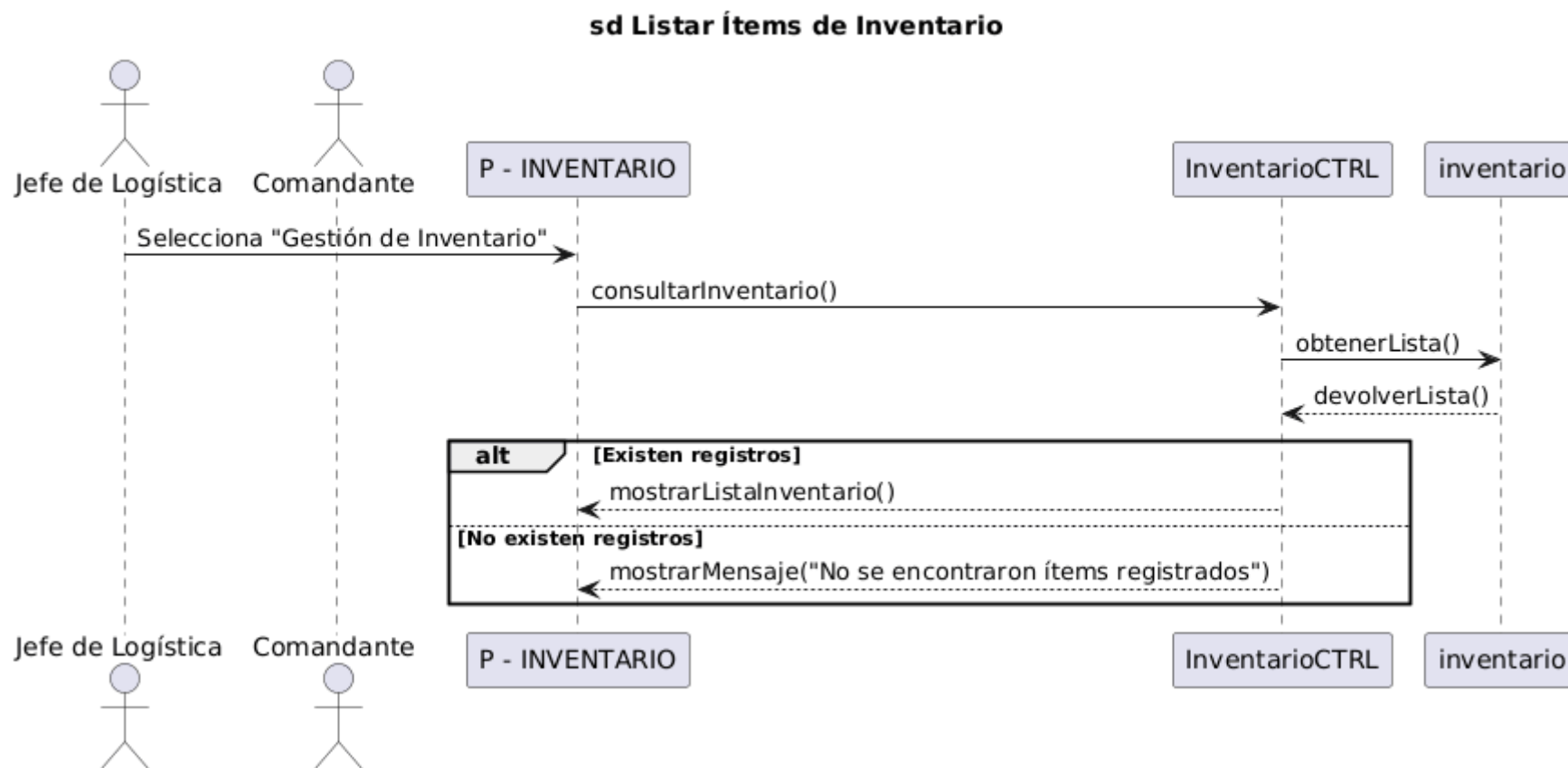


Figura 110. Diagrama de Secuencia: Gestiona Inventario-Listar Inventario
Fuente: Elaboración Propia.

- Registrar Ítem de Inventario

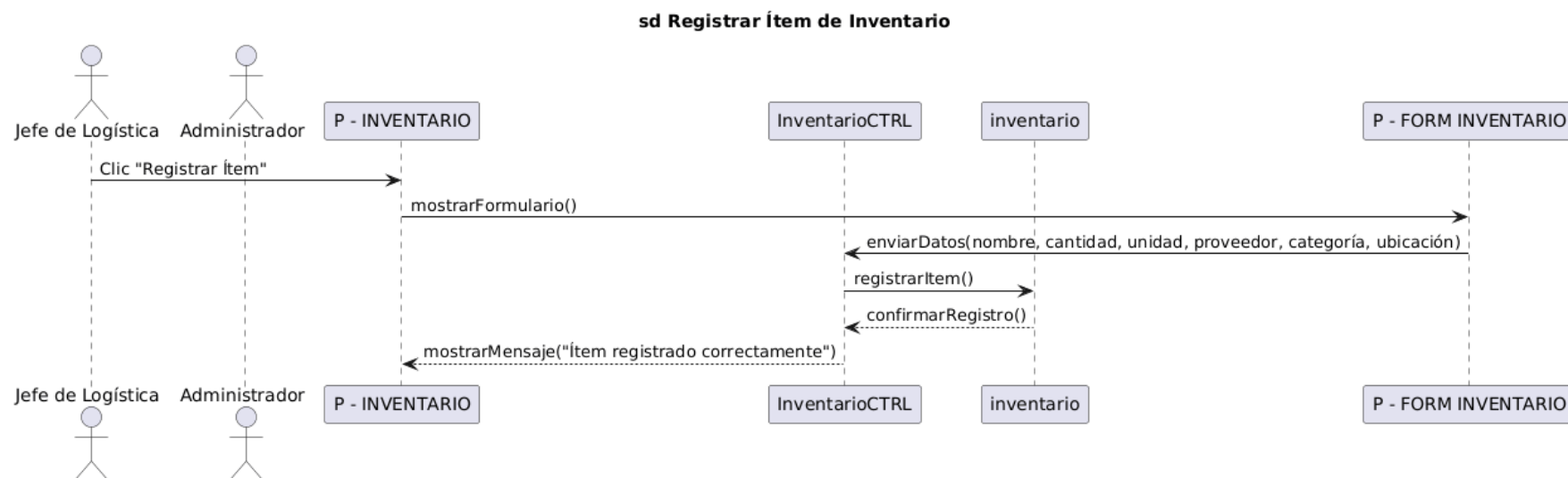


Figura 111. Diagrama de Secuencia: Gestionar Inventario-Registrar Ítem de Inventario
Fuente: Elaboración Propia.

- Editar / Actualizar Ítem

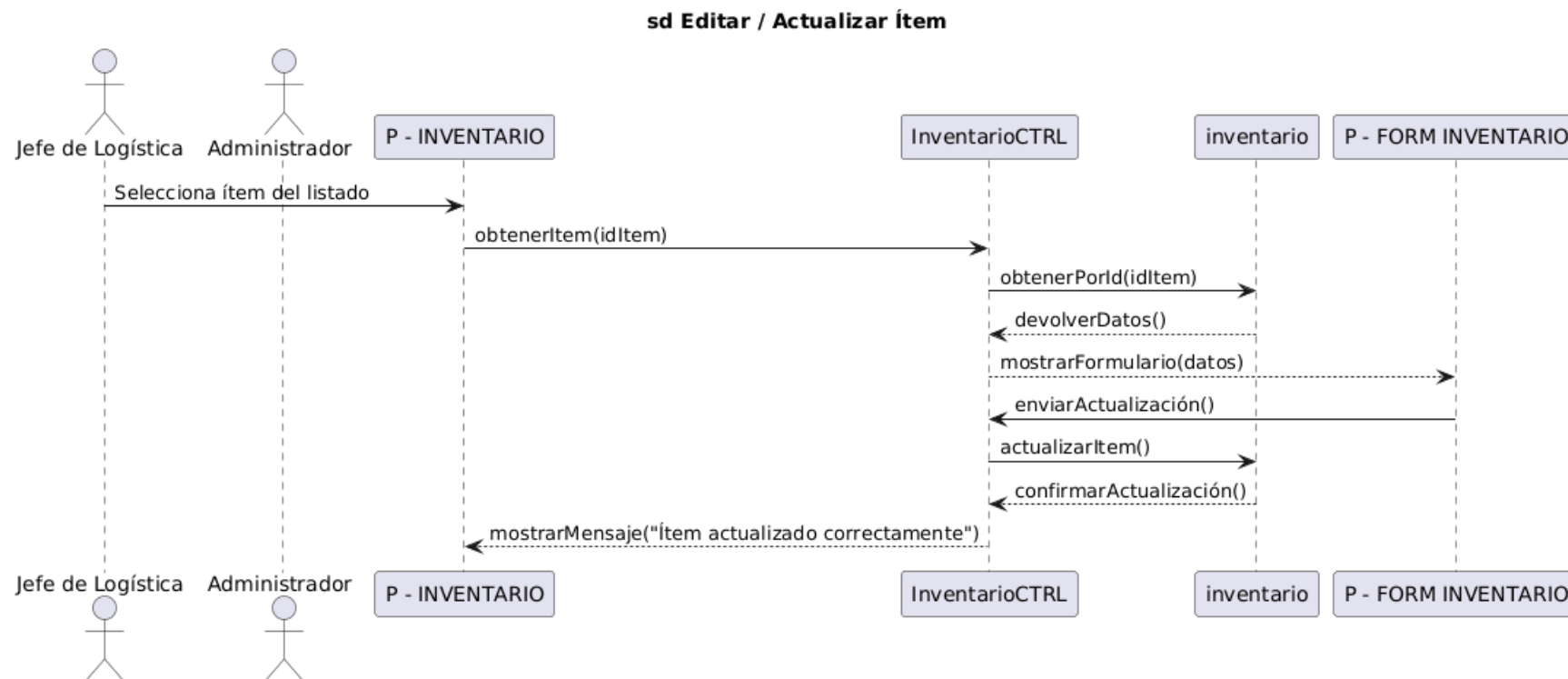


Figura 112. Diagrama de Secuencia: Gestionar Inventario-Editar/Actualizar Ítem
Fuente: Elaboración Propia.

- Registrar Movimiento de Inventario

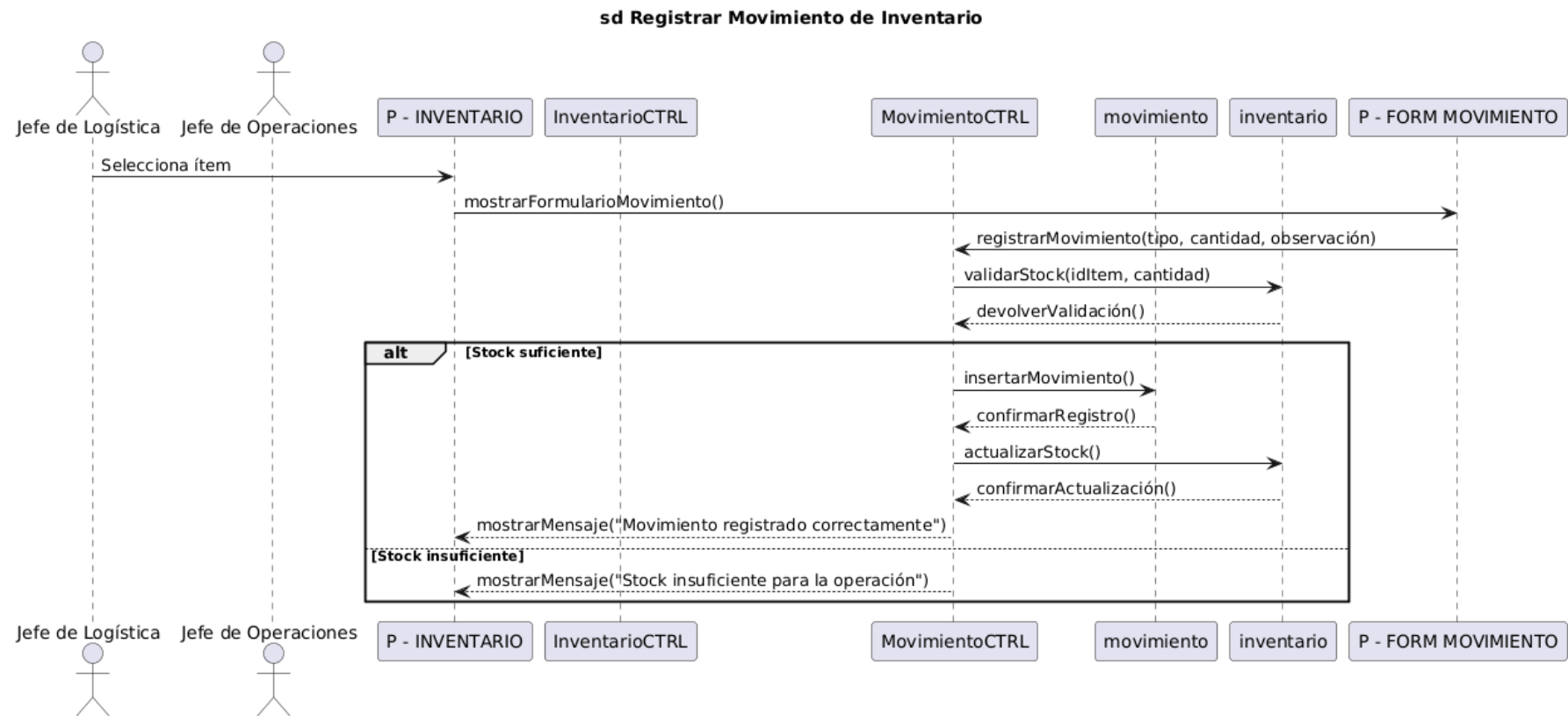


Figura 113. Diagrama de Secuencia: Gestionar Inventario-Registrar Movimiento de Inventario
Fuente: Elaboración Propia.

- Controlar Stock y Notificaciones de Cantidad Crítica

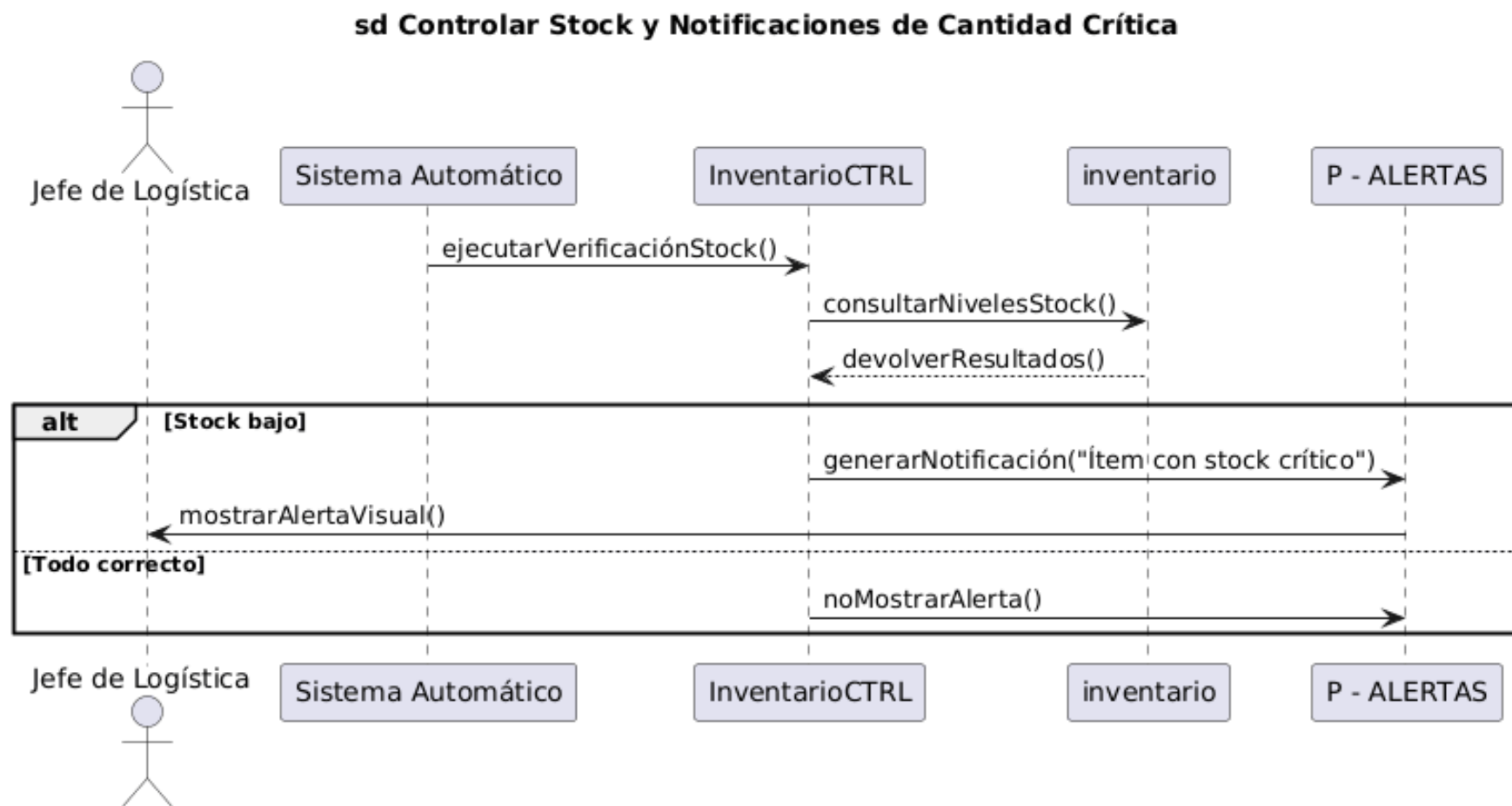


Figura 114. Diagrama de Secuencia: Gestionar Inventario-Controlar Stock y Notificaciones de Cantidad Crítica
Fuente: Elaboración Propia.

- Gestionar Catálogos

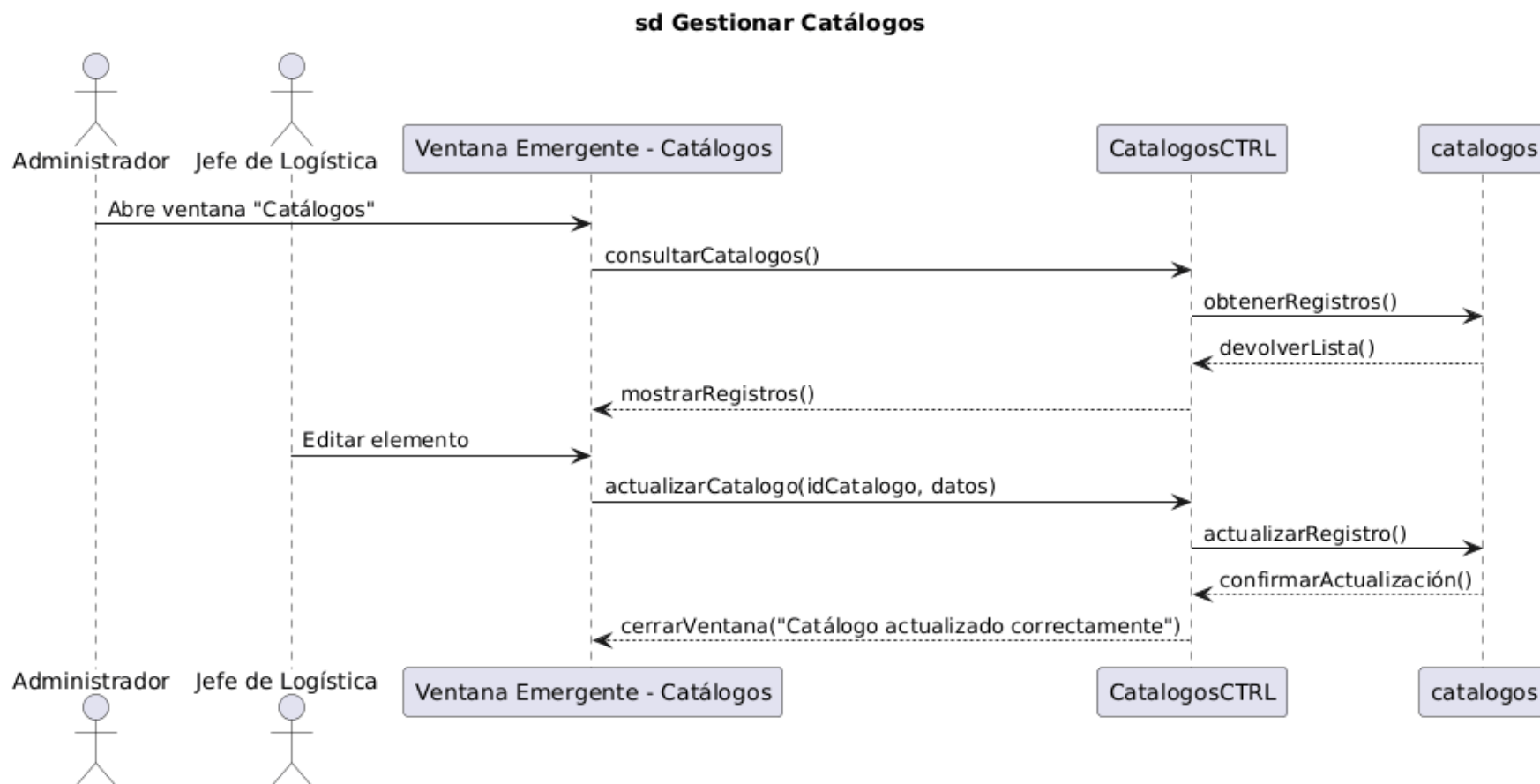


Figura 115. Diagrama de Secuencia: Gestionar Inventario-Gestionar Catálogos
Fuente: Elaboración Propia.

- Generar Reportes de Inventario y Movimientos

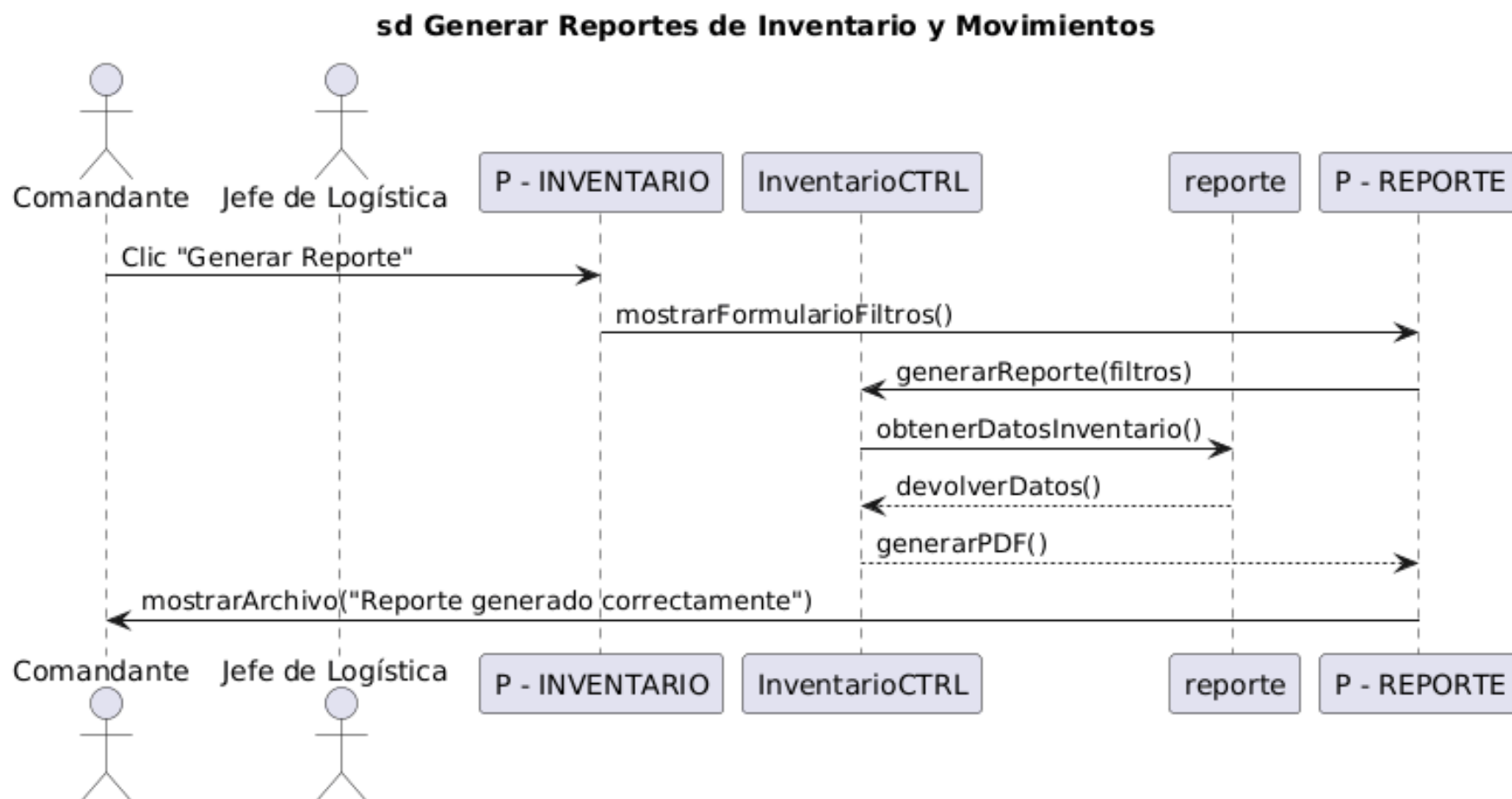


Figura 116. Diagrama de Secuencia: Gestionar Inventario-Generar Reportes de Inventario y Movimientos
Fuente: Elaboración Propia.

II.1.2.6.11.8 Diagrama de Secuencia: Gestión de Donativos

- Listar Donativos

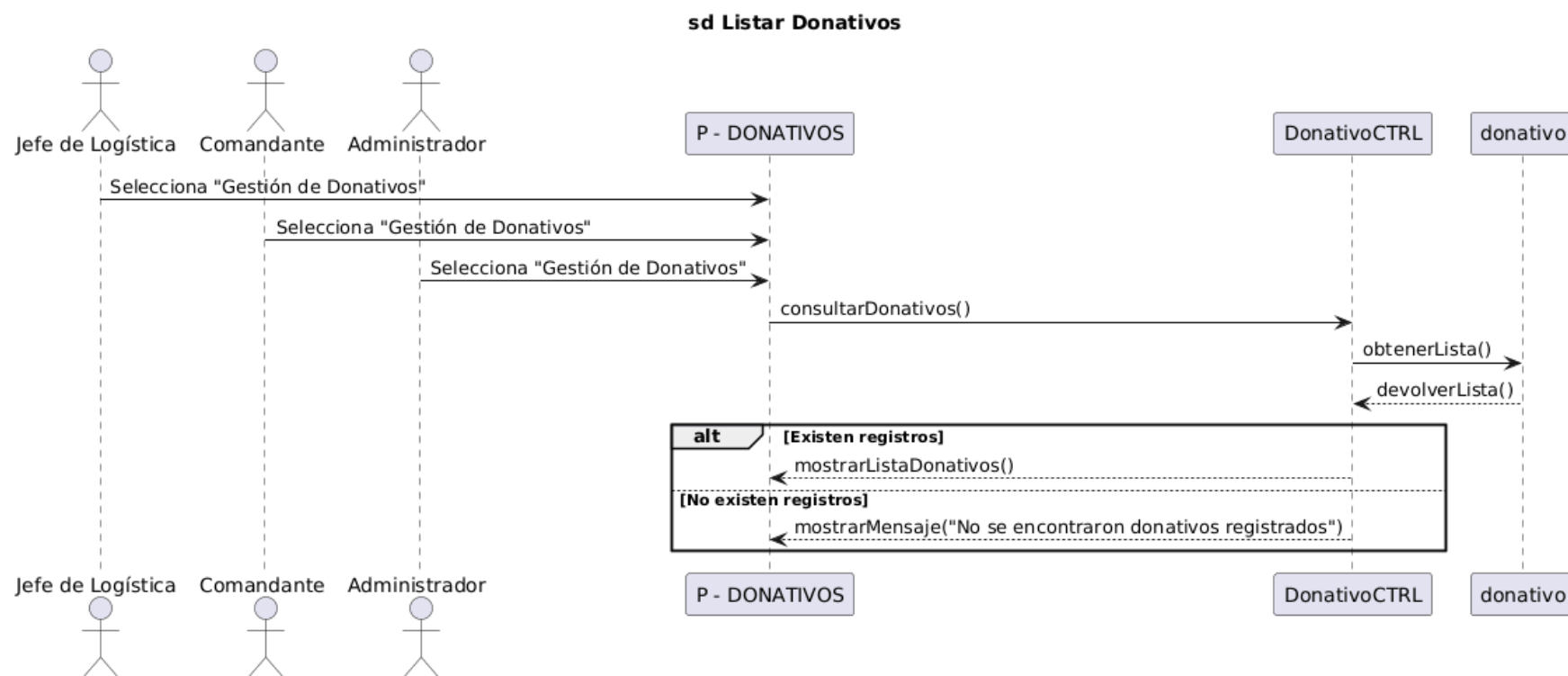


Figura 117. Diagrama de Secuencia: Gestión de Donativos-Listar Donativo
Fuente: Elaboración Propia.

- Registrar Donativos

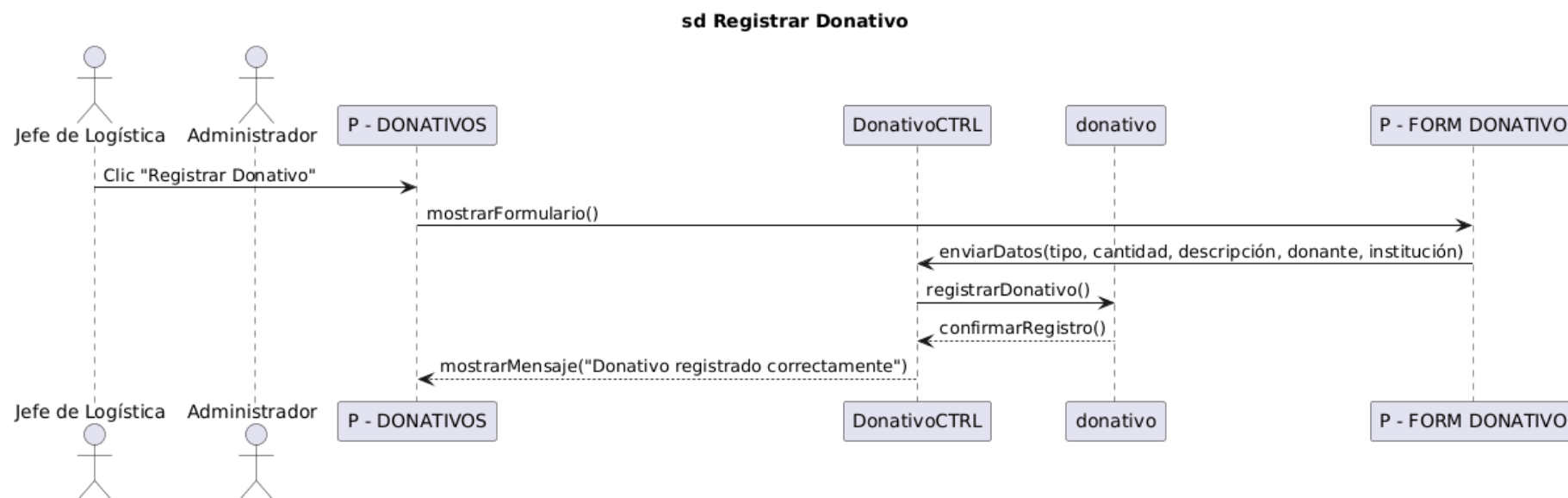


Figura 118. Diagrama de Secuencia: Gestión de Donativos-Registrar Donativos
Fuente: Elaboración Propia.

- Registrar Donante o Institución Asociada

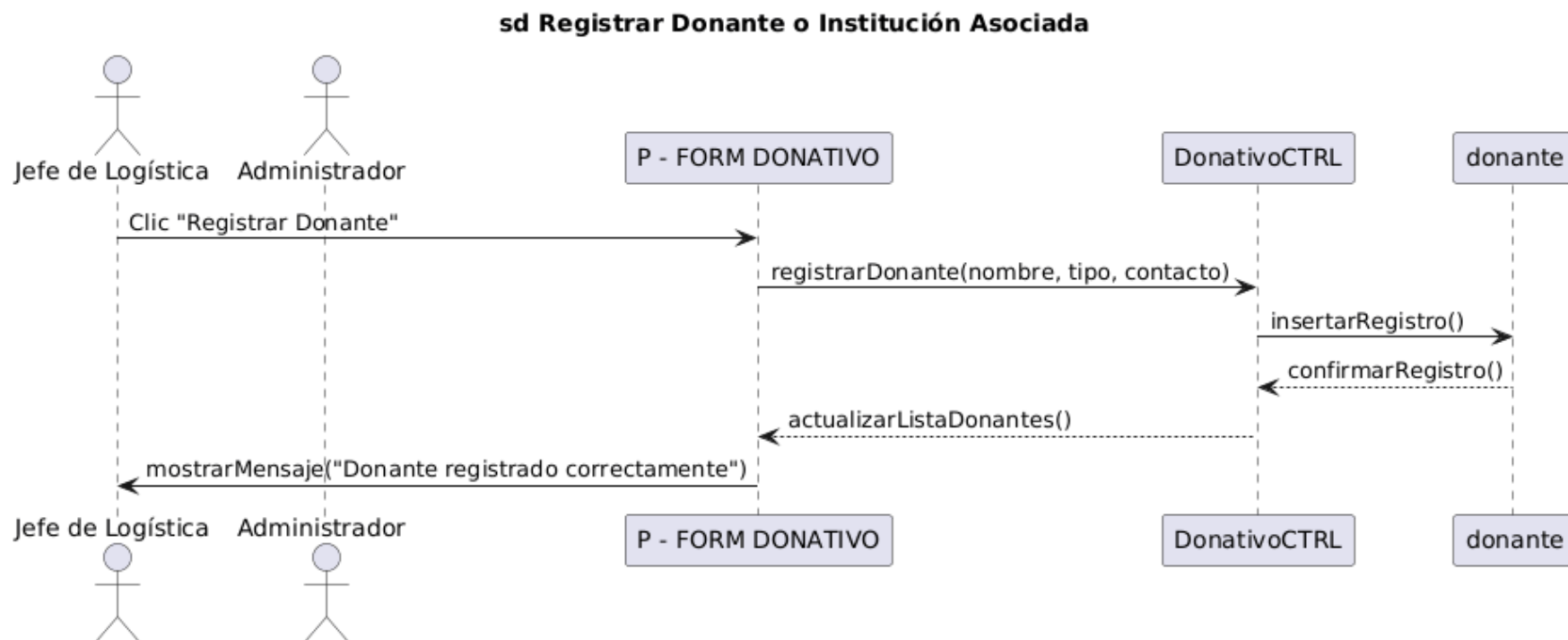


Figura 119. Diagrama de Secuencia: Gestión de Donativos Registrar Donante o Institución Asociada
Fuente: Elaboración Propia.

- Actualizar / Cambiar Estado del Donativo

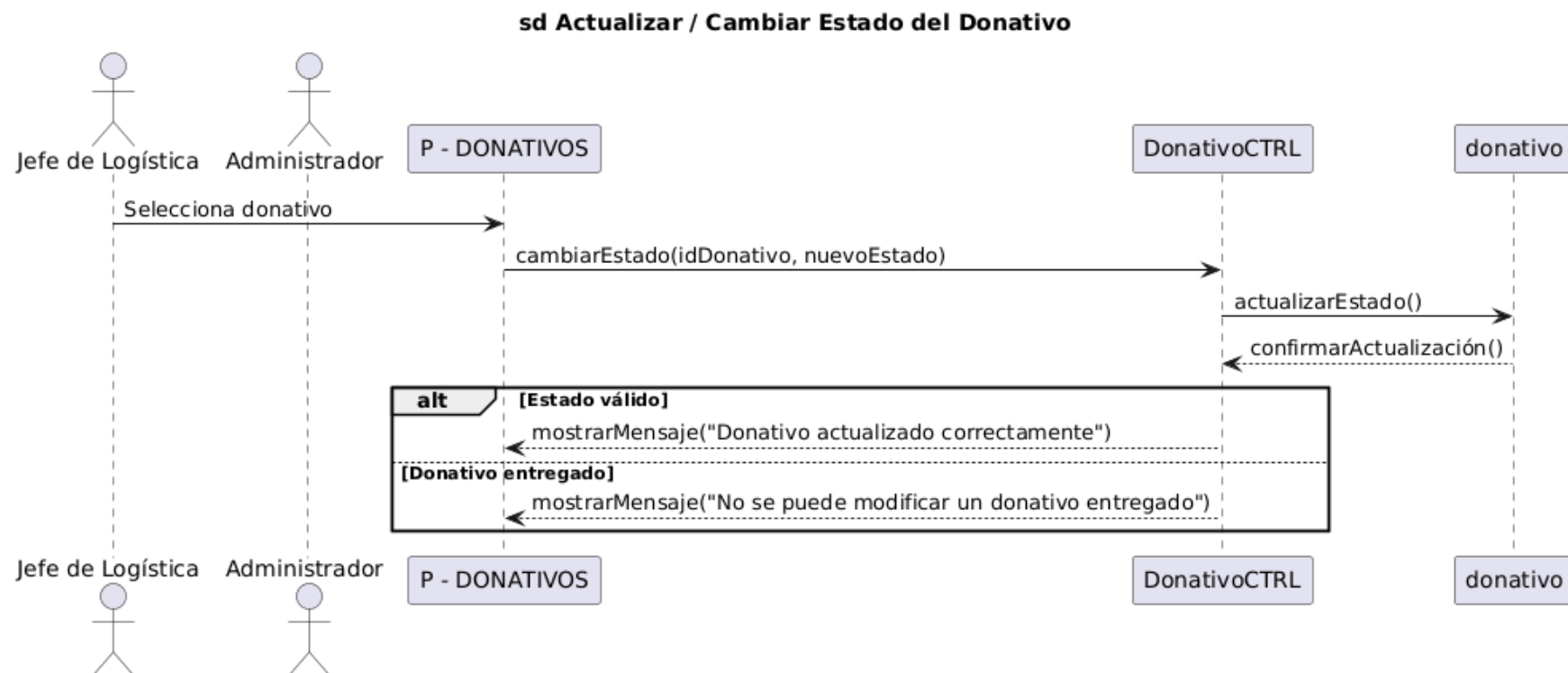


Figura 120. Diagrama de Secuencia: Gestión de Donativos Actualizar/Cambiar Estado del Donativo
Fuente: Elaboración Propia.

- Asociar Donativo Material al Inventario

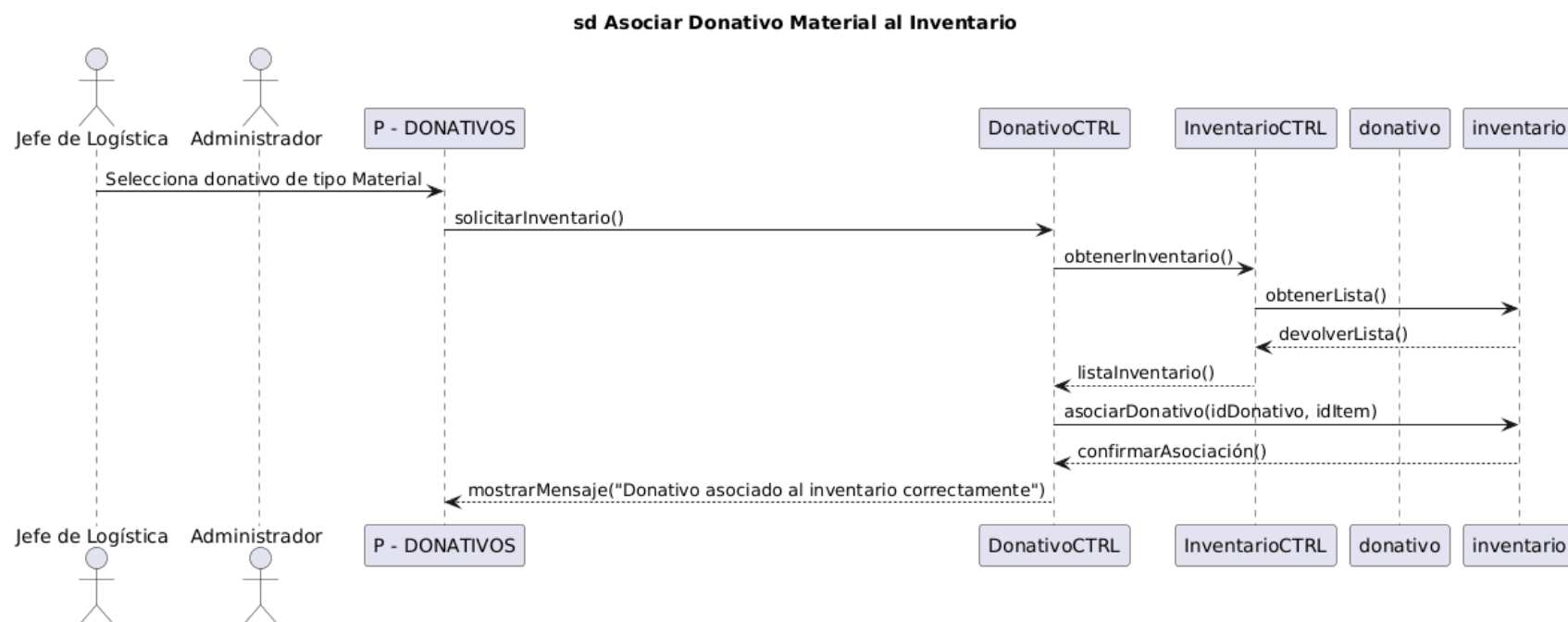


Figura 121. Diagrama de Secuencia: Gestión de Donativos Generar Reportes de Donativos-Asociar Donativo al Inventario
Fuente: Elaboración Propia.

- Generar Reportes de Donativos

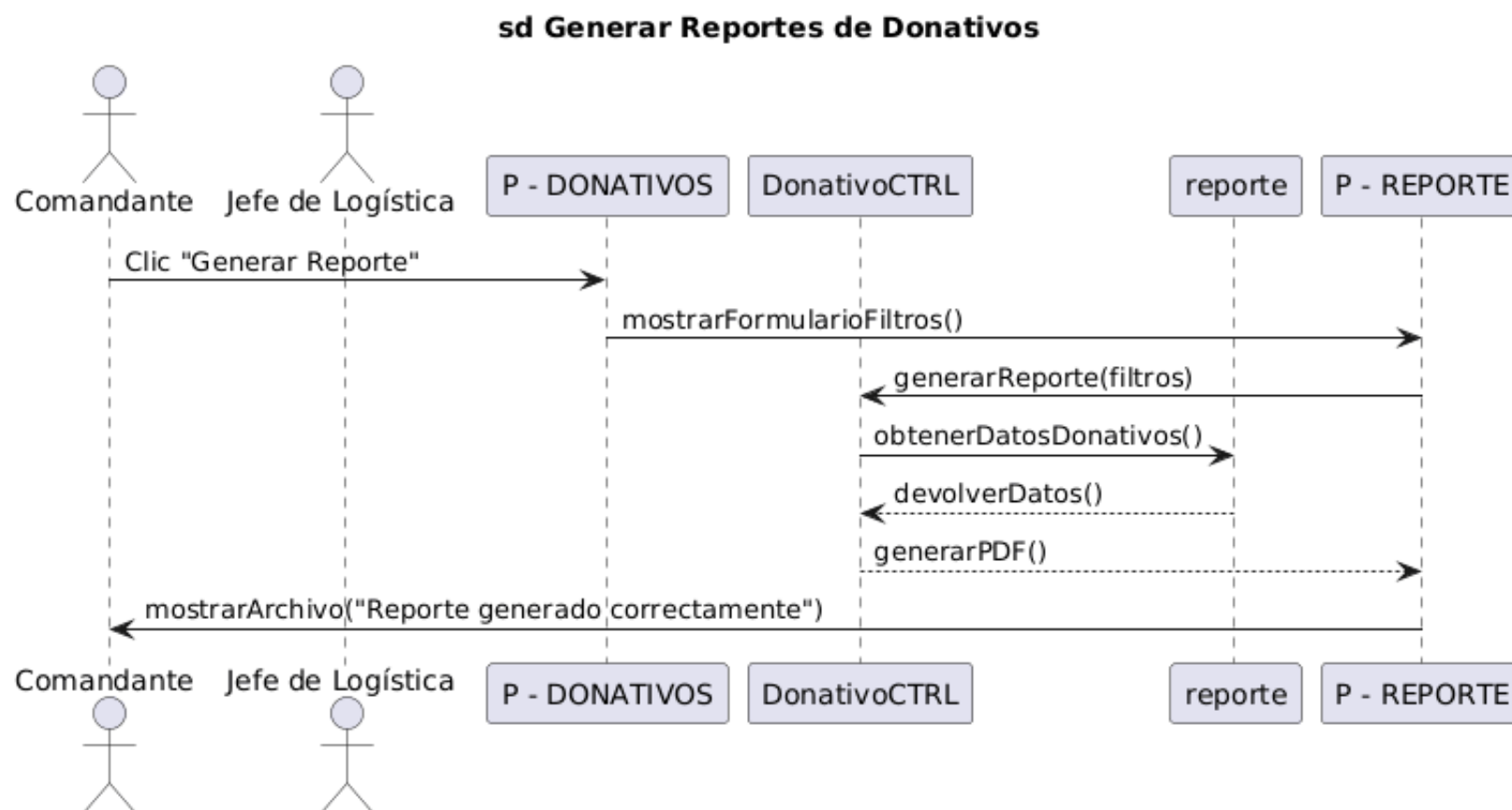


Figura 122. Diagrama de Secuencia: Gestión de Donativos-Generar Reportes de Donativos
Fuente: Elaboración Propia.

II.1.2.6.11.9 Diagrama de Secuencia: Gestión de Roles

- Listar de Roles

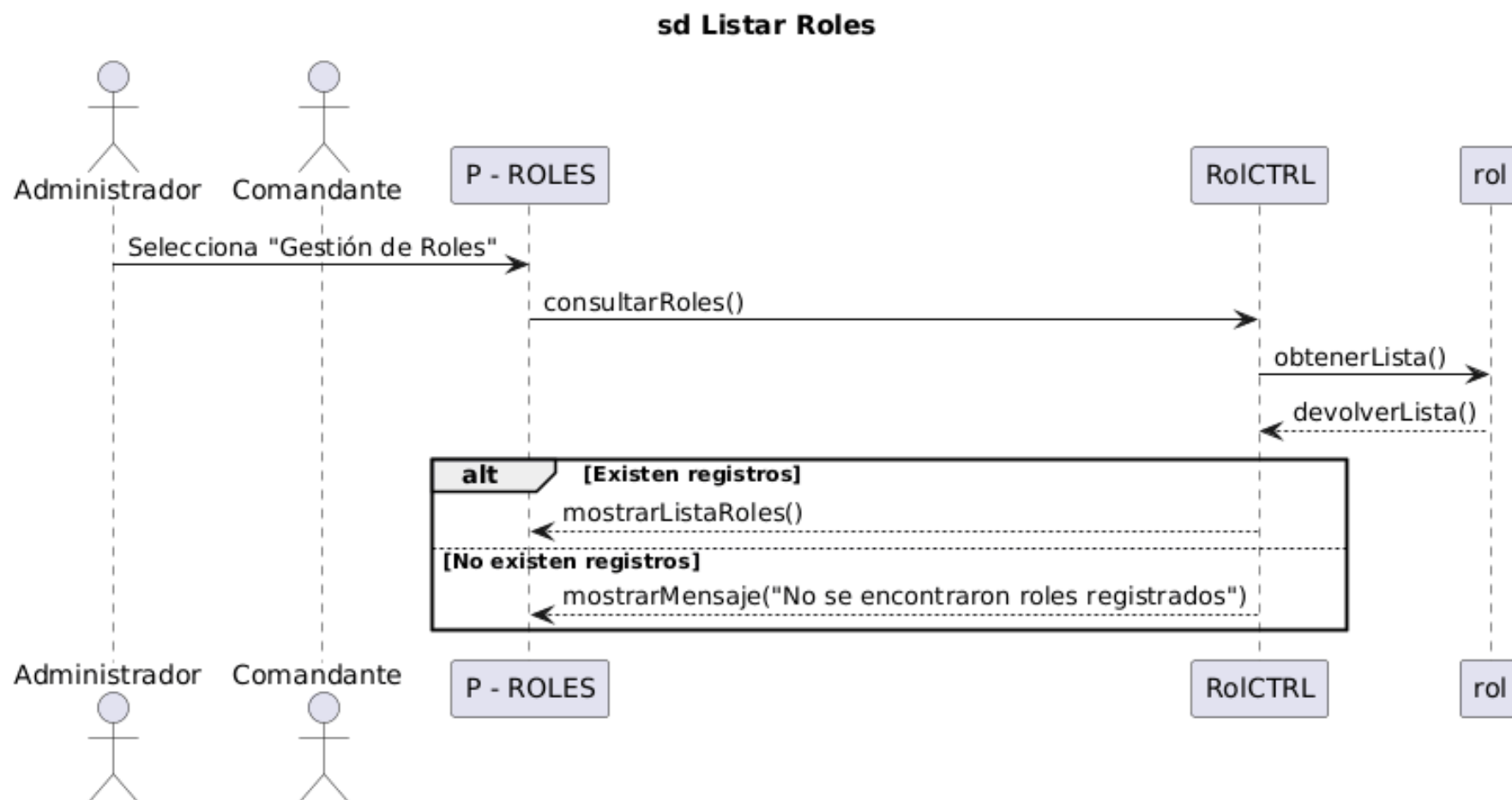


Figura 123. Diagrama de Secuencia: Gestión de Roles-Listar Roles
Fuente: Elaboración Propia.

- Registrar nuevo Rol

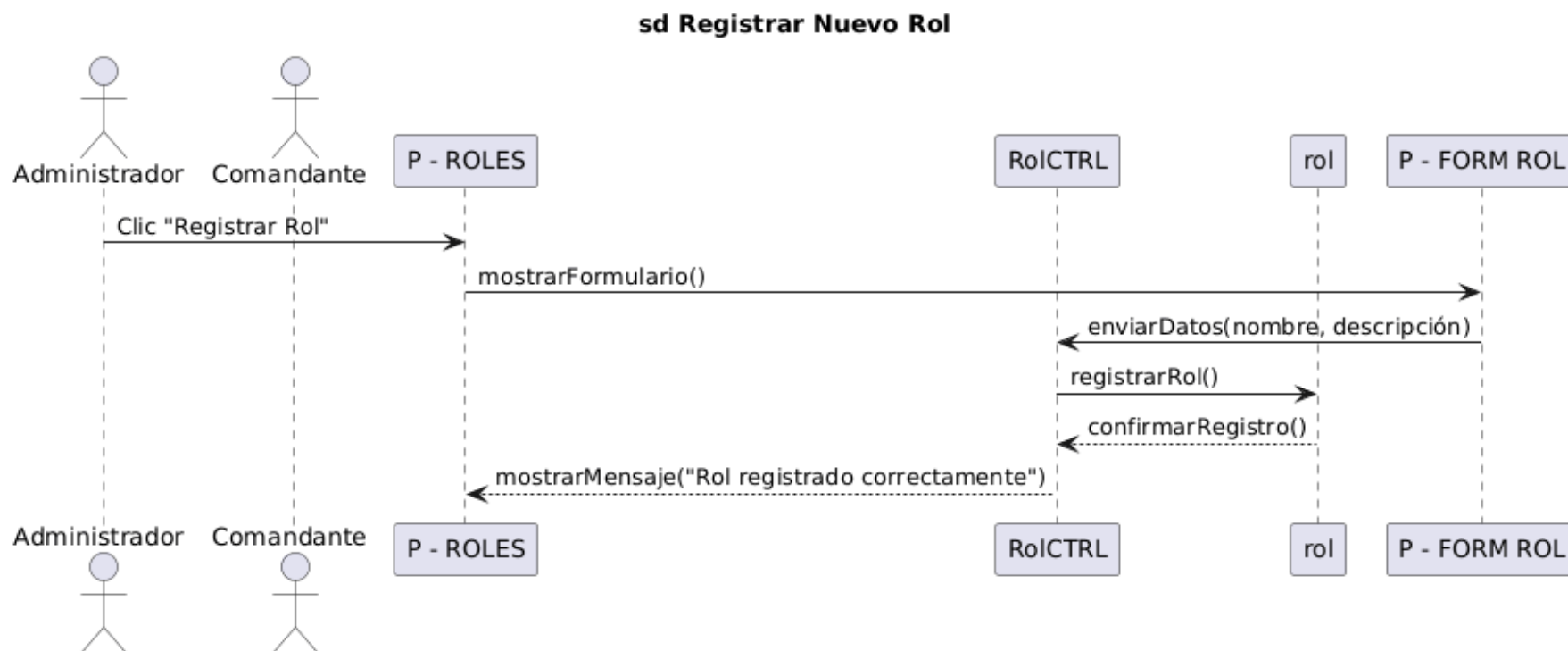


Figura 124. Diagrama de Secuencia: Gestión de Roles-Registrar Nuevo Rol
Fuente: Elaboración Propia.

- Editar / Actualizar Rol

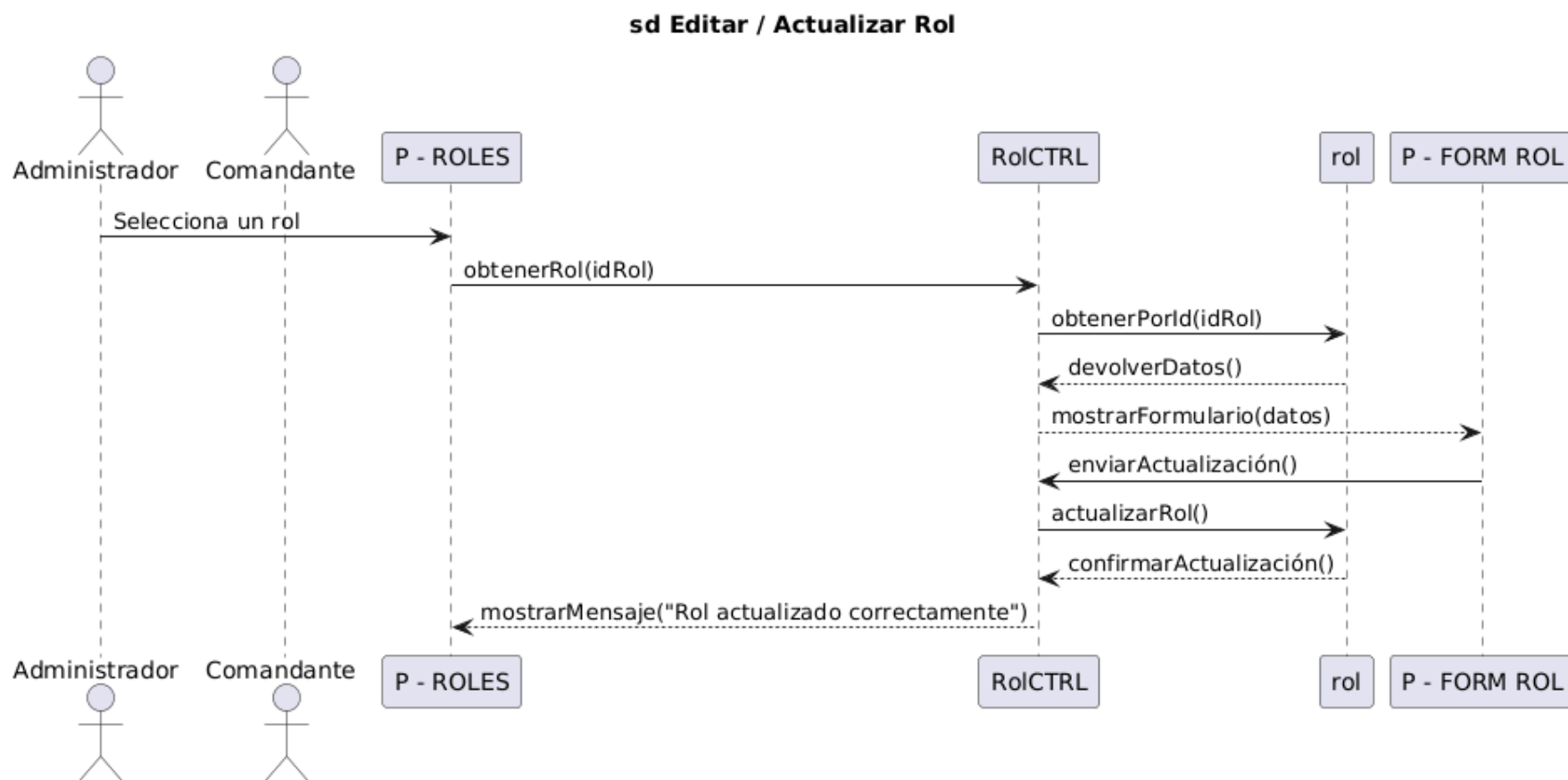


Figura 125. Diagrama de Secuencia: Gestión de Roles-Editar/Actualizar Roles
Fuente: Elaboración Propia.

- Eliminar Rol

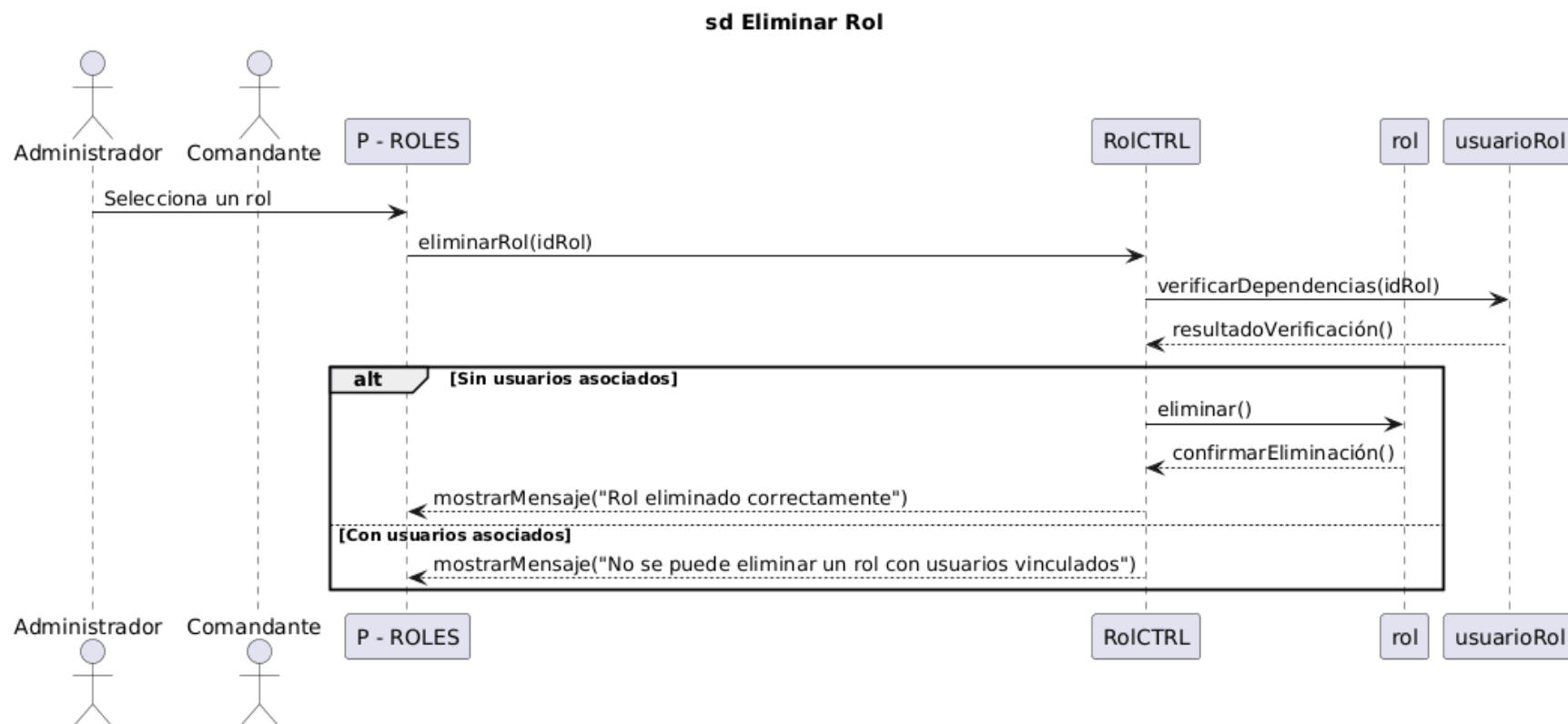


Figura 126. Diagrama de Secuencia: Gestión de Roles-Eliminar Roles
Fuente: Elaboración Propia.

- Asignar Permisos a Rol

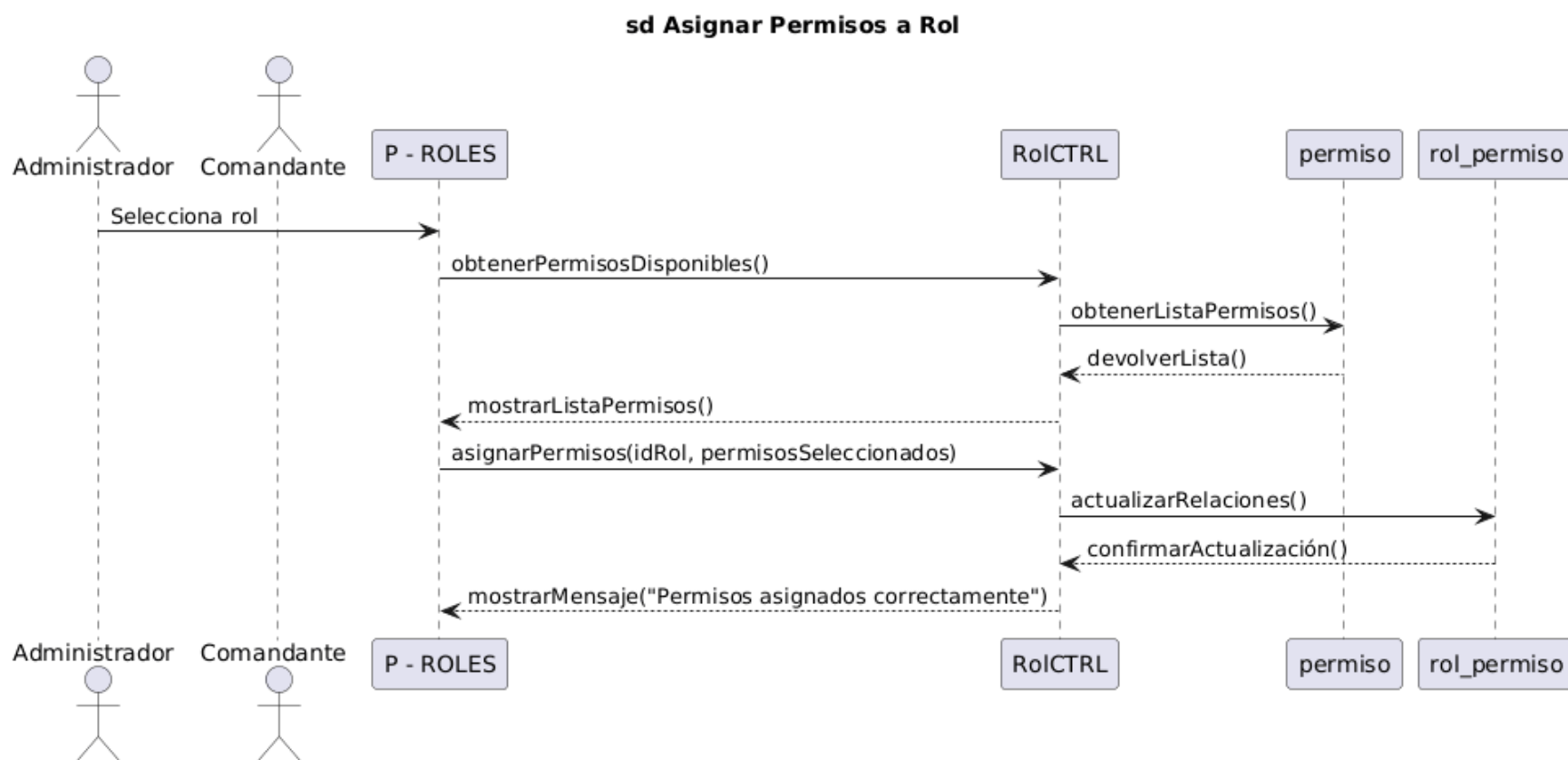


Figura 127. Diagrama de Secuencia: Gestión de Roles-Asignar Permisos a Rol
Fuente: Elaboración Propia.

- Ver Permisos Asignados

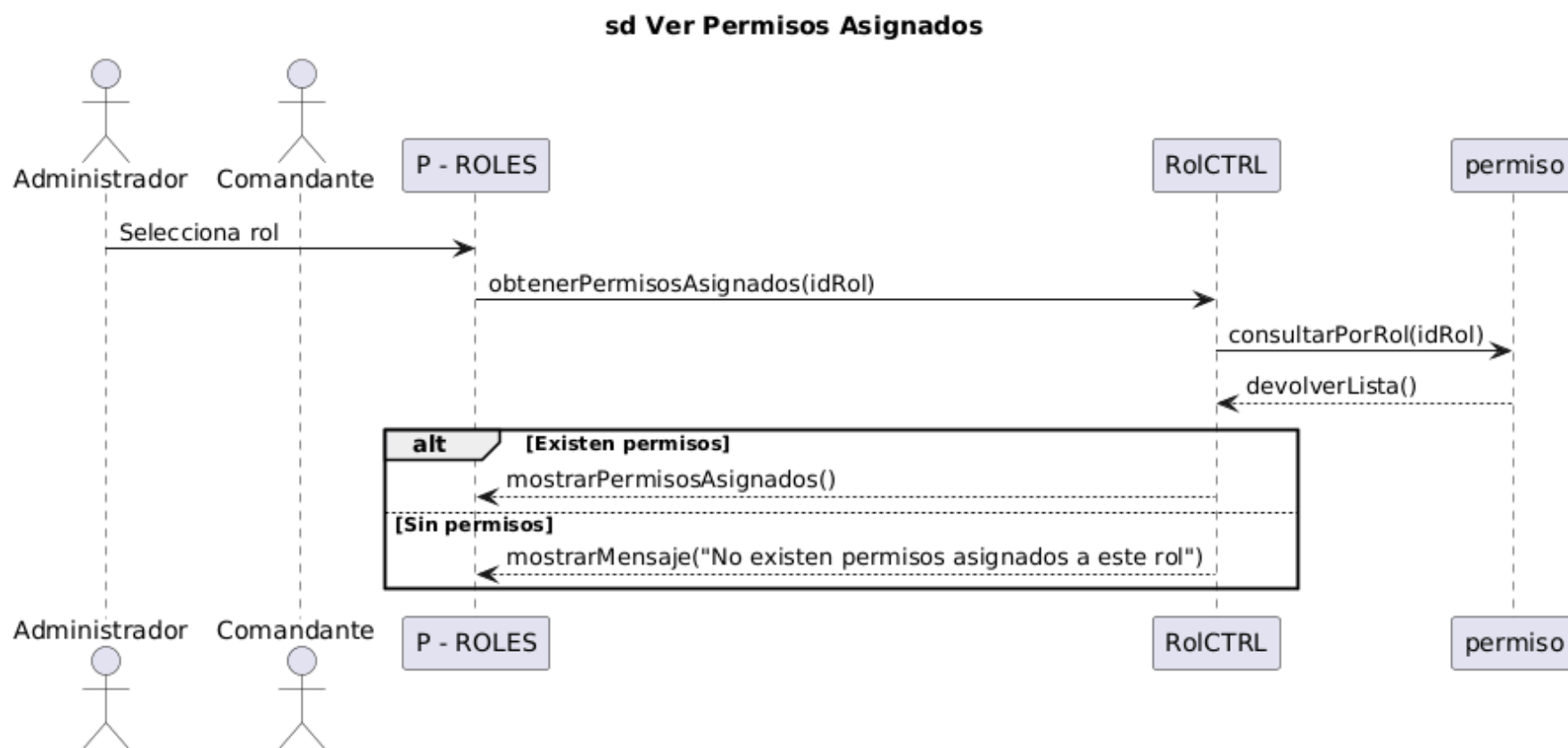


Figura 128. Diagrama de Secuencia: Gestión de Roles-Ver Permisos Asignados
Fuente: Elaboración Propia.

- Generar Reportes de Roles y Permisos

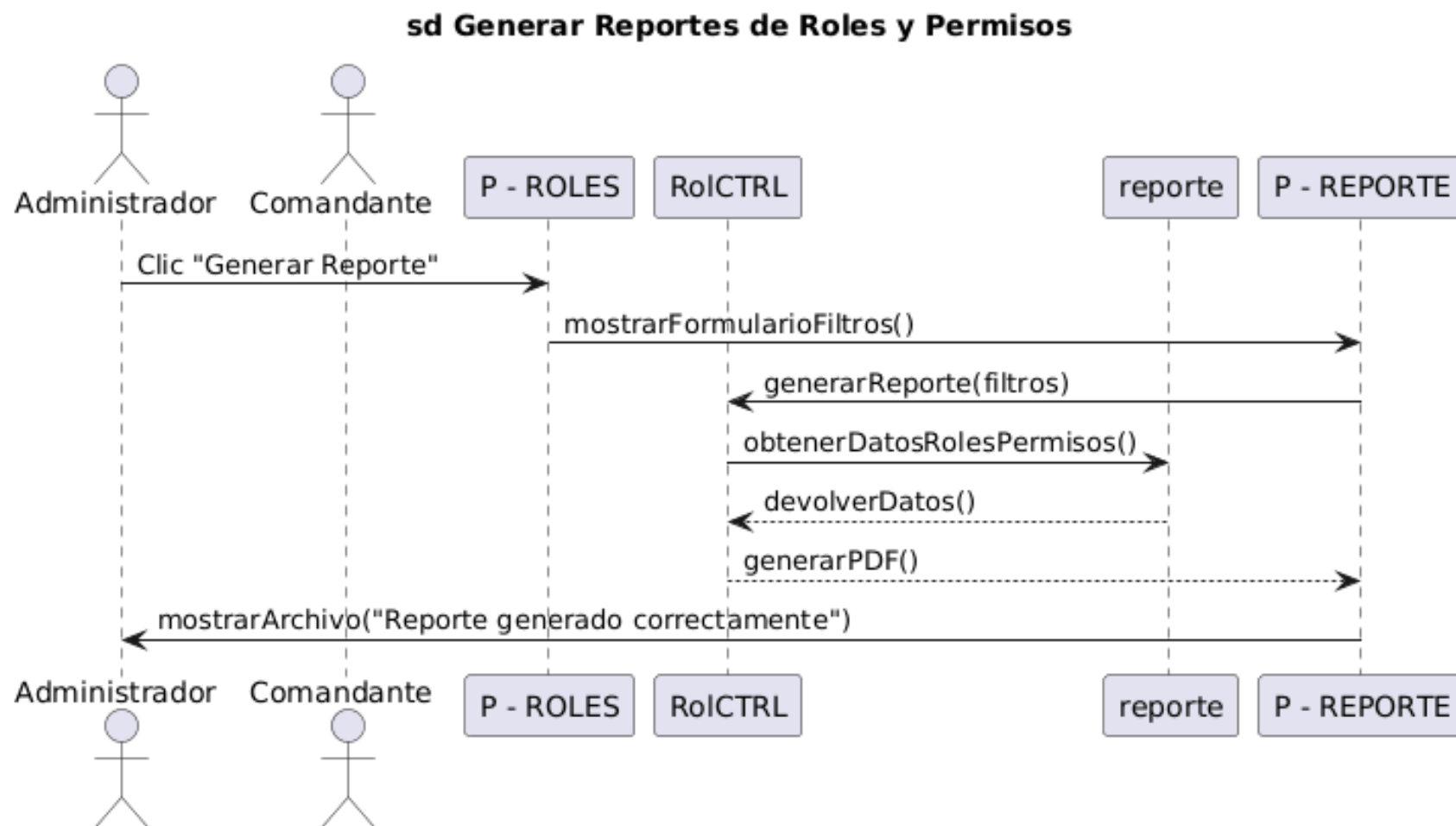


Figura 129. Diagrama de Secuencia: Gestión de Roles-Generar Reportes de Roles y Permisos

Fuente: Elaboración Propia.

II.1.2.6.11.10 Diagrama de Secuencia: Gestión de Usuarios

- Listar Usuarios

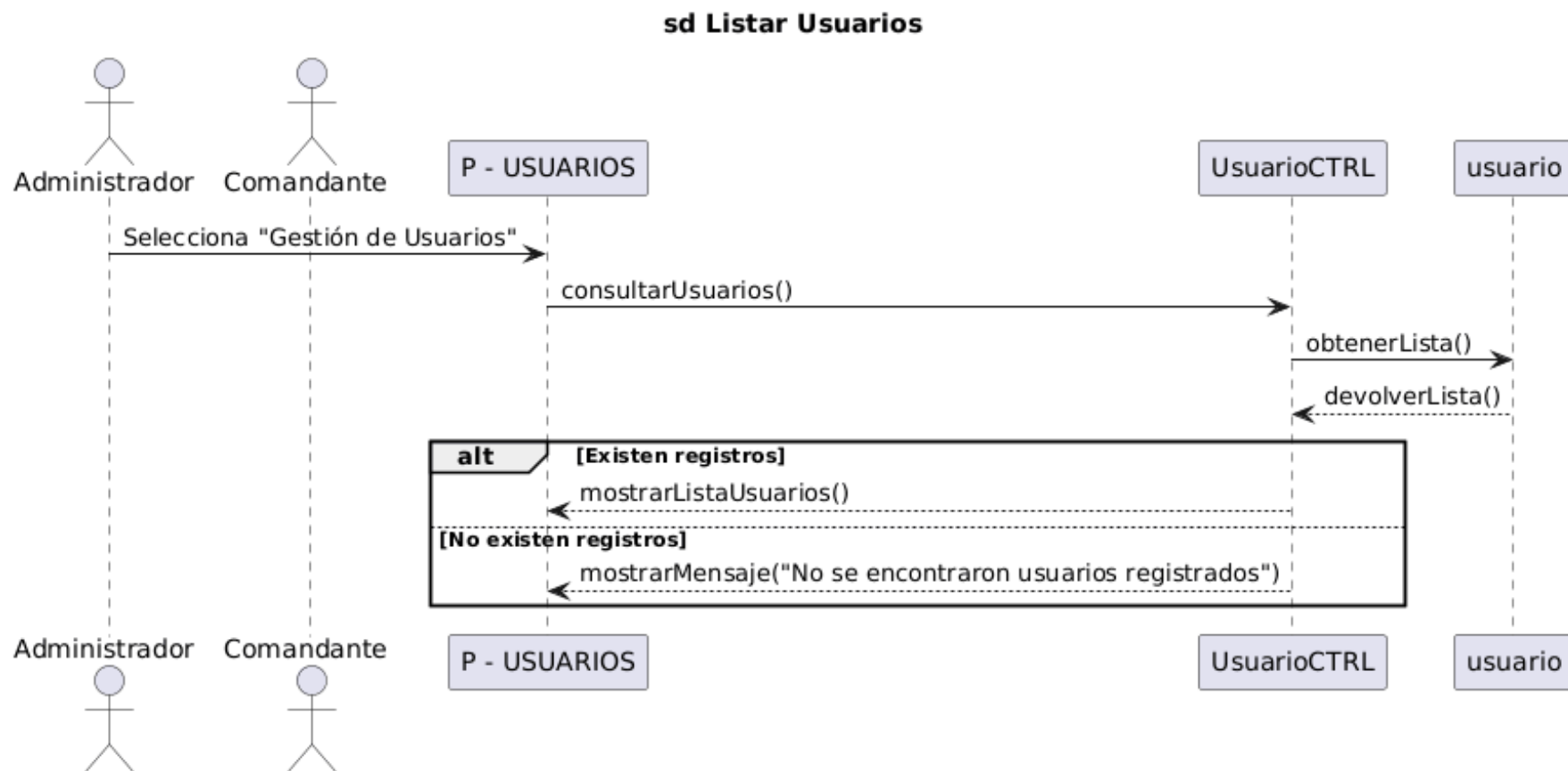


Figura 130. Diagrama de Secuencia: Gestión de Usuarios-Listar Usuarios
Fuente: Elaboración Propia.

- Registrar Nuevo Usuario

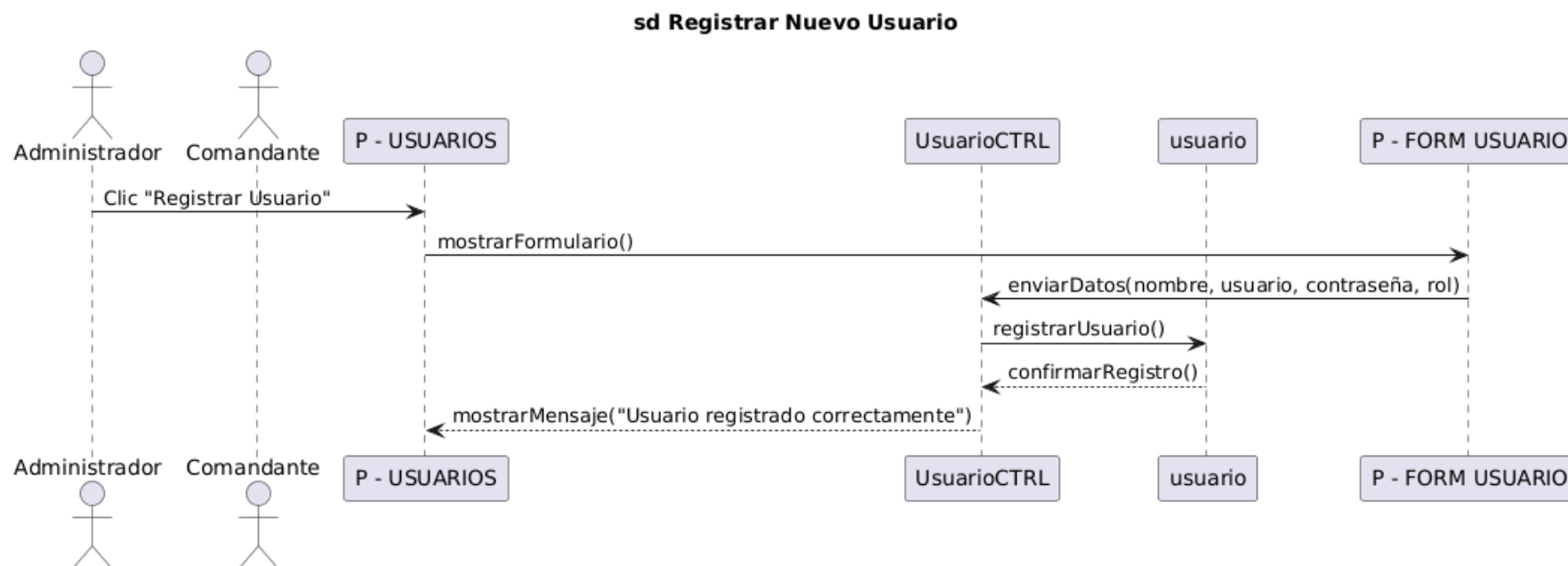


Figura 131. Diagrama de Secuencia: Gestión de Usuarios-Registrar Nuevo Usuario
Fuente: Elaboración Propia.

- **Editar / Actualizar Usuario**

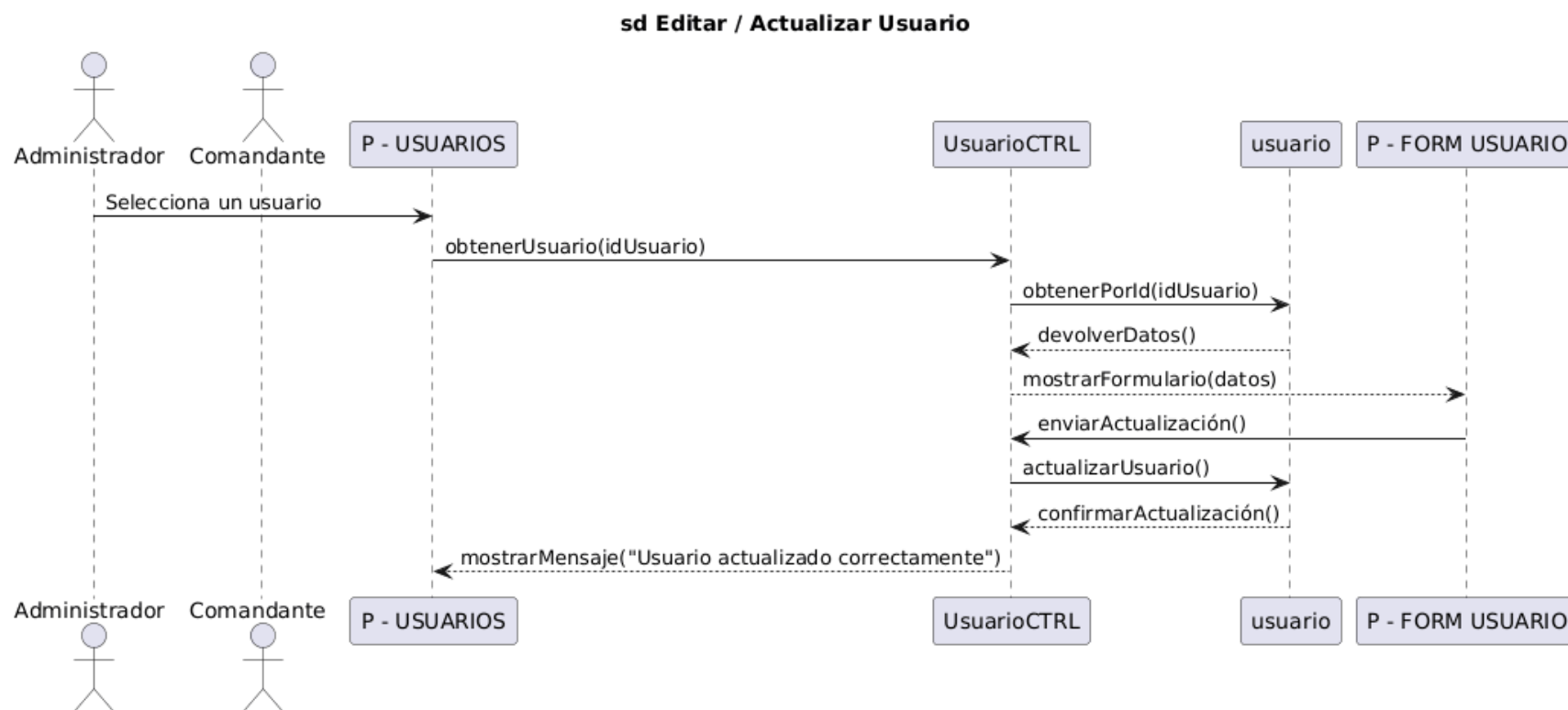


Figura 132. Diagrama de Secuencia: Gestión de Usuarios-Editar/Actualizar Usuario
Fuente: Elaboración Propia.

- Desactivar / Reactivar Usuario

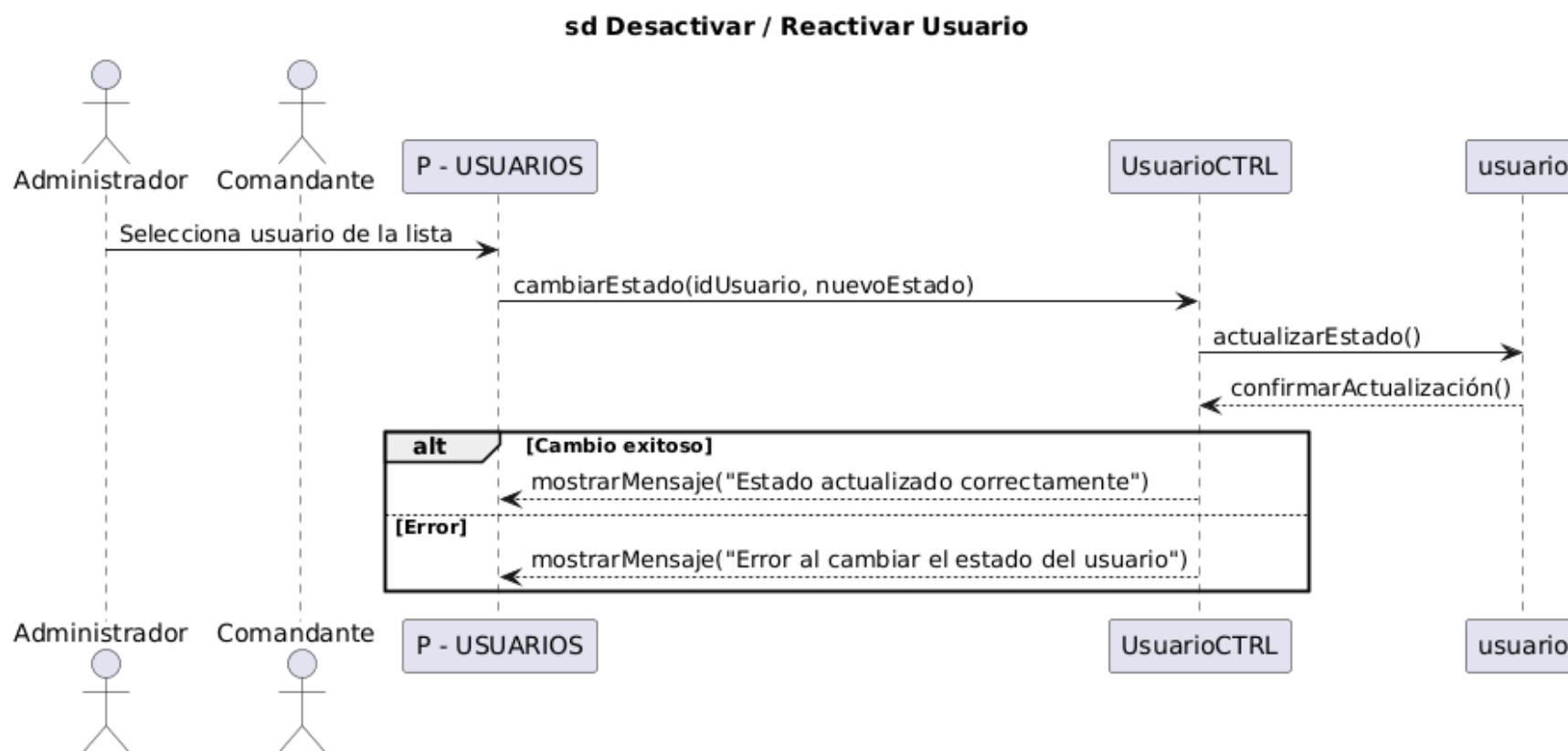


Figura 133. Diagrama de Secuencia: Gestión de Usuarios-Desactivar/Reactivar Usuario
Fuente: Elaboración Propia.

- Registrar Asignar / Eliminar Rol de Usuario

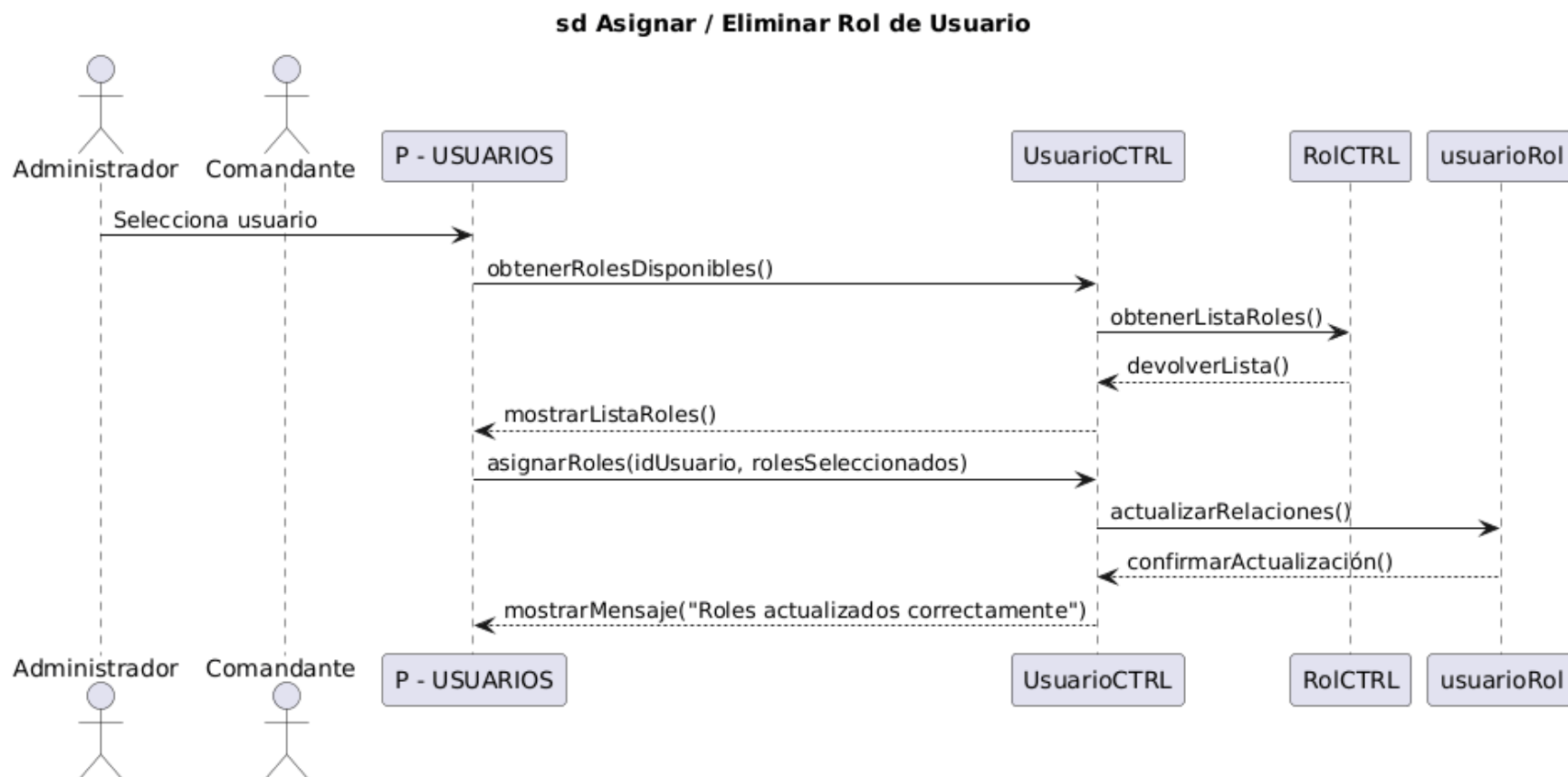


Figura 134. Diagrama de Secuencia: Gestión de Usuarios-Registrar Asignar/Eliminar Rol de Usuario
Fuente: Elaboración Propia.

- Generar Reportes de Usuarios

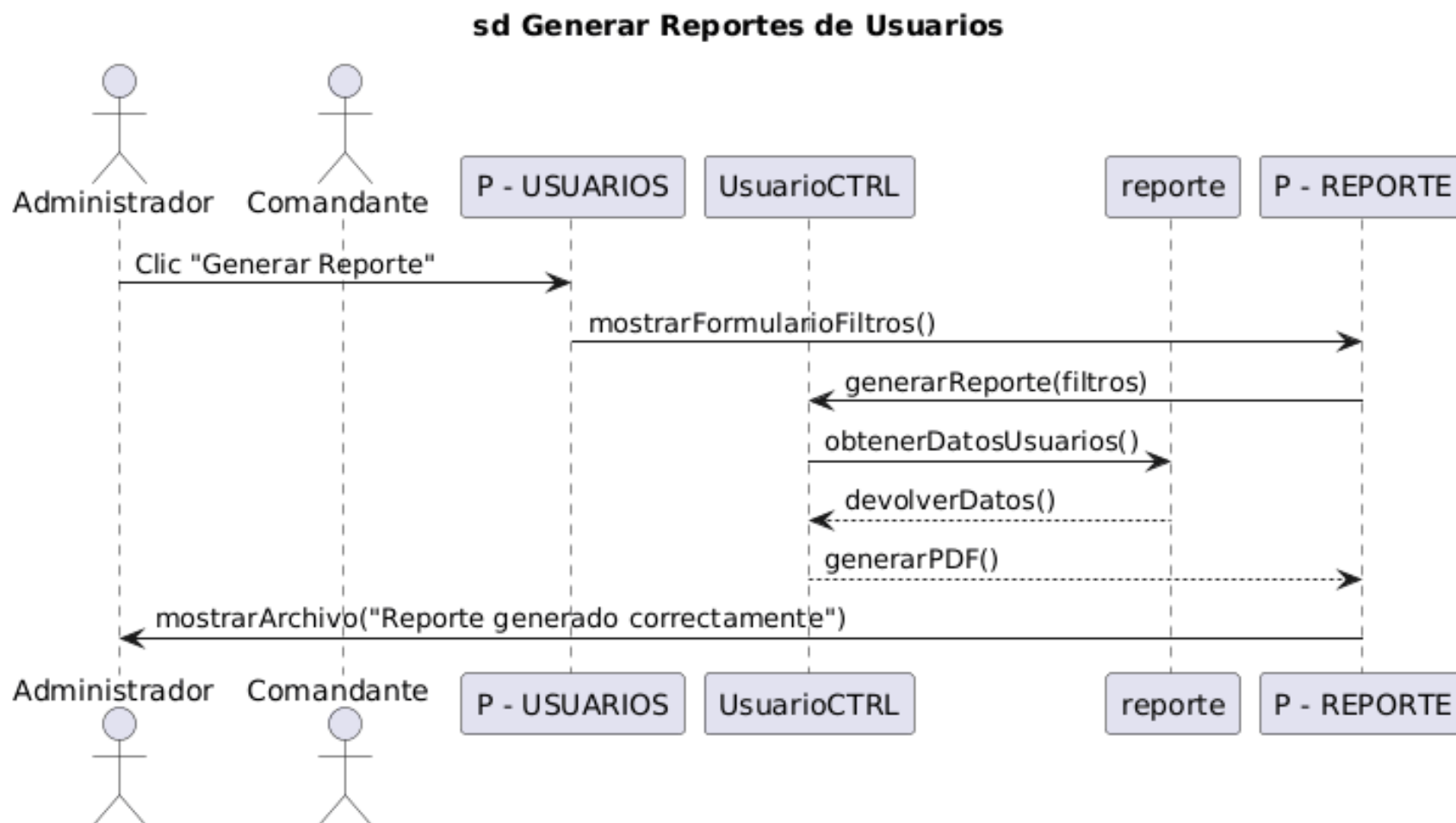


Figura 135. Diagrama de Secuencia: Gestión de Usuarios-Generar Reportes de Usuarios

Fuente: Elaboración Propia.

II.1.2.6.12 Diagrama de Clases

Diagrama de Clases Class Diagram

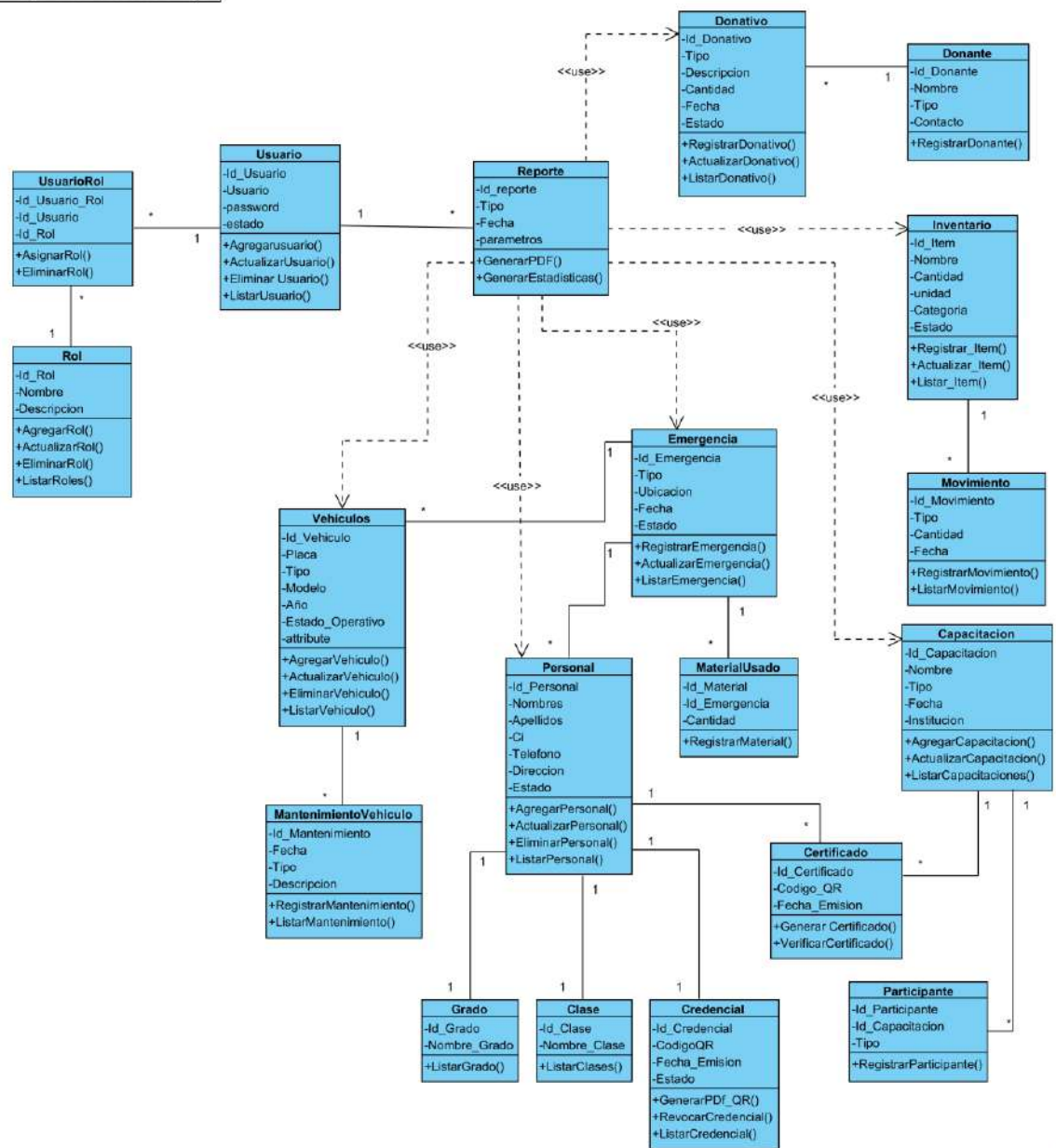


Figura 136. Diagrama de Clases
Fuente: Elaboración Propia.

II.1.2.6.13 Diagrama Entidad – Relación

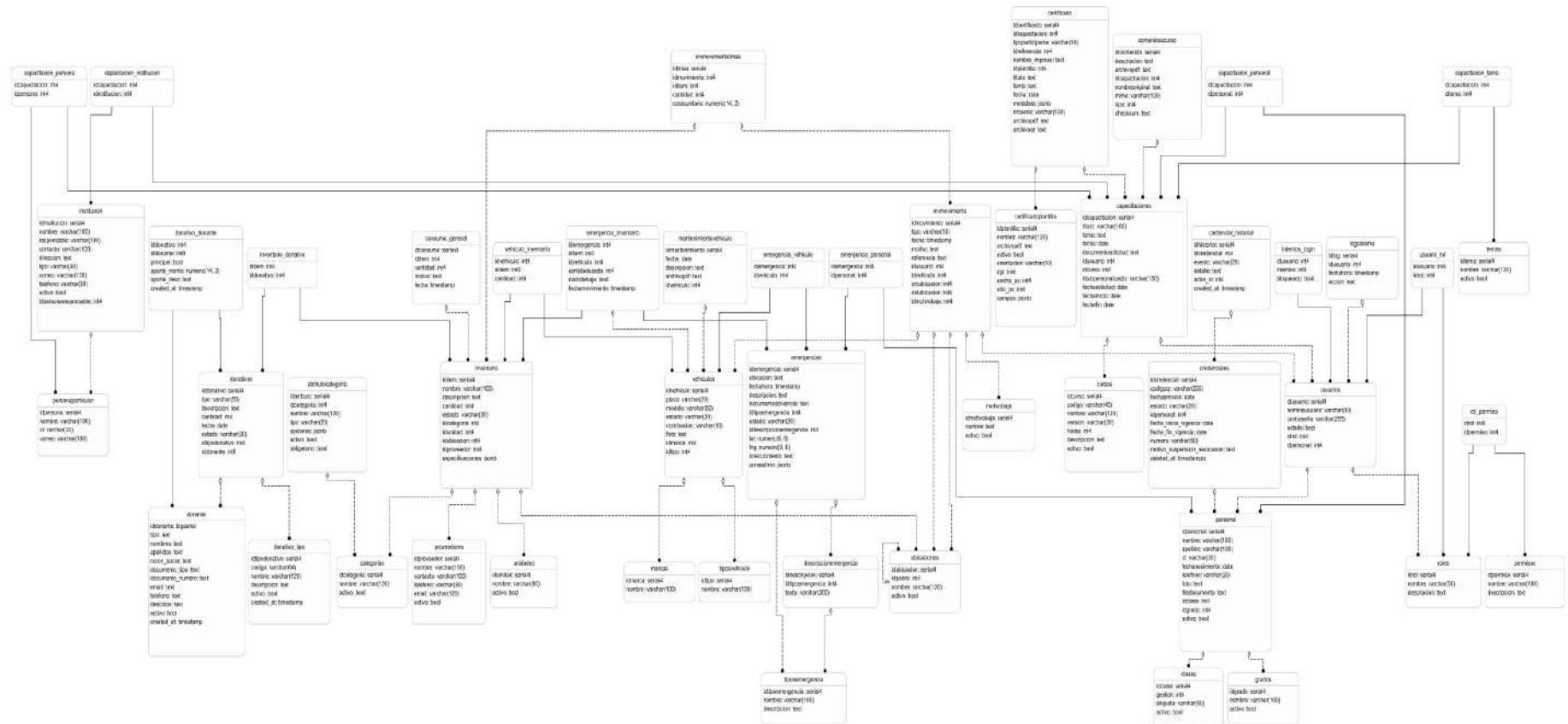


Figura 137. Diagrama Entidad-Relación
Fuente: Elaboración Propia.

II.1.2.6.14 Diagrama de Componentes

El diagrama de componentes del sistema web SAR Santa Bárbara representa la estructura general del software, organizado bajo una arquitectura de tres capas: presentación, lógica de negocio y persistencia de datos.

En la capa de presentación, el usuario interactúa a través del navegador web, accediendo a las distintas interfaces del sistema. La capa intermedia o de negocio está compuesta por los componentes del servidor Node.js/Express, que incluyen los controladores, rutas, servicios, middlewares y modelos Sequelize, los cuales gestionan la lógica del sistema y las operaciones del backend. Finalmente, la capa de persistencia está constituida por la base de datos PostgreSQL, donde se almacenan todas las entidades del sistema.

Esta arquitectura modular permite un mantenimiento ágil, escalabilidad y separación clara de responsabilidades entre los diferentes módulos del sistema.

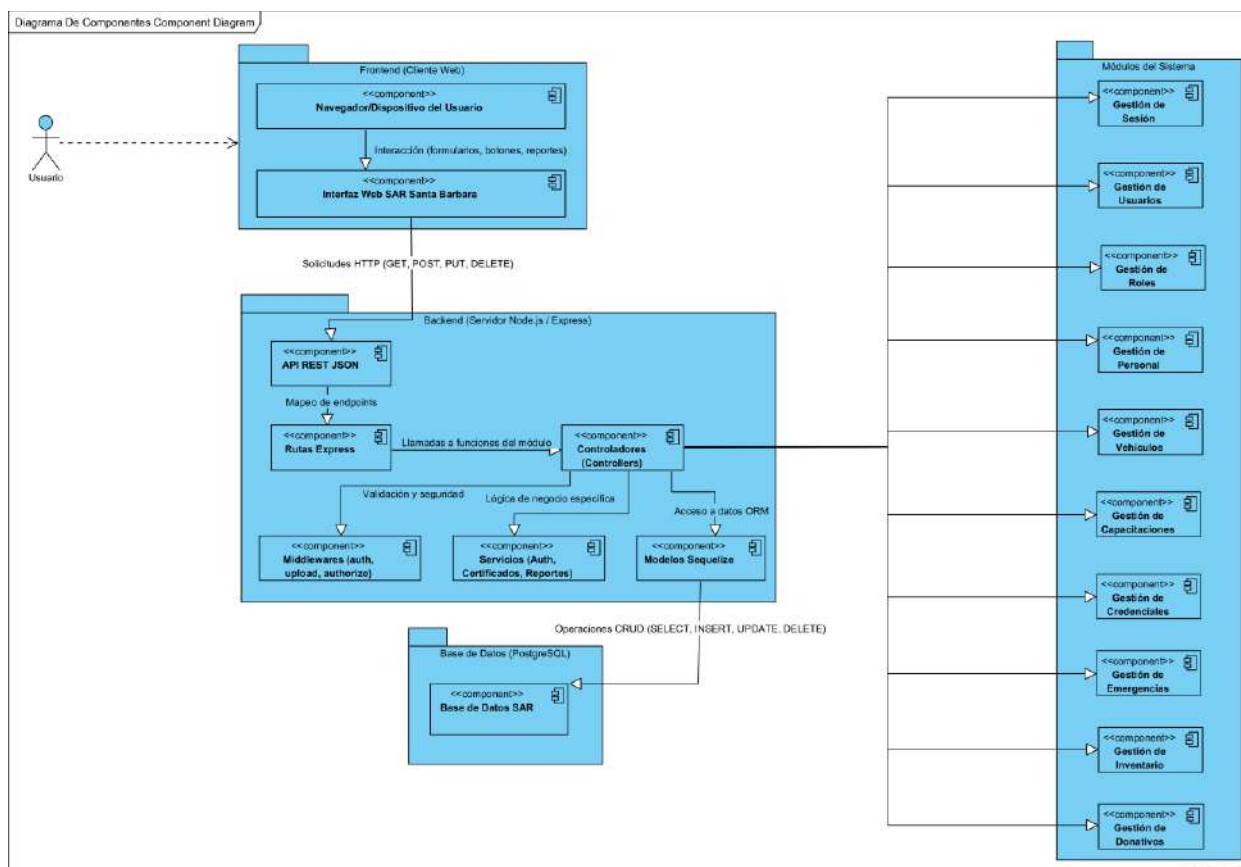


Figura 138. Diagrama Componentes
Fuente: Elaboración Propia.

II.1.2.6.15 Diagrama de Despliegue

El diagrama de despliegue extendido del sistema web SAR Santa Bárbara muestra la infraestructura tecnológica sobre la cual se ejecutan los componentes del sistema.

Los actores institucionales (comandante, Jefe de Operaciones, Jefe de Instrucción y Jefe de Logística) acceden desde sus equipos a través de un navegador web, que comunica las solicitudes HTTP al servidor de aplicación Node.js / Express.

En el servidor web se encuentran desplegados los componentes principales de la lógica del sistema: API REST JSON, Rutas Express, Controladores, Middlewares, Servicios y Modelos

Sequelize.

Estos gestionan los módulos funcionales del sistema (Gestión de Sesión, Usuarios, Roles, Personal, Vehículos, Capacitaciones, Credenciales, Emergencias, Inventario y Donativos).

Finalmente, la información se almacena de forma persistente en el Servidor de Base de Datos PostgreSQL, mediante operaciones SQL controladas por el ORM Sequelize.

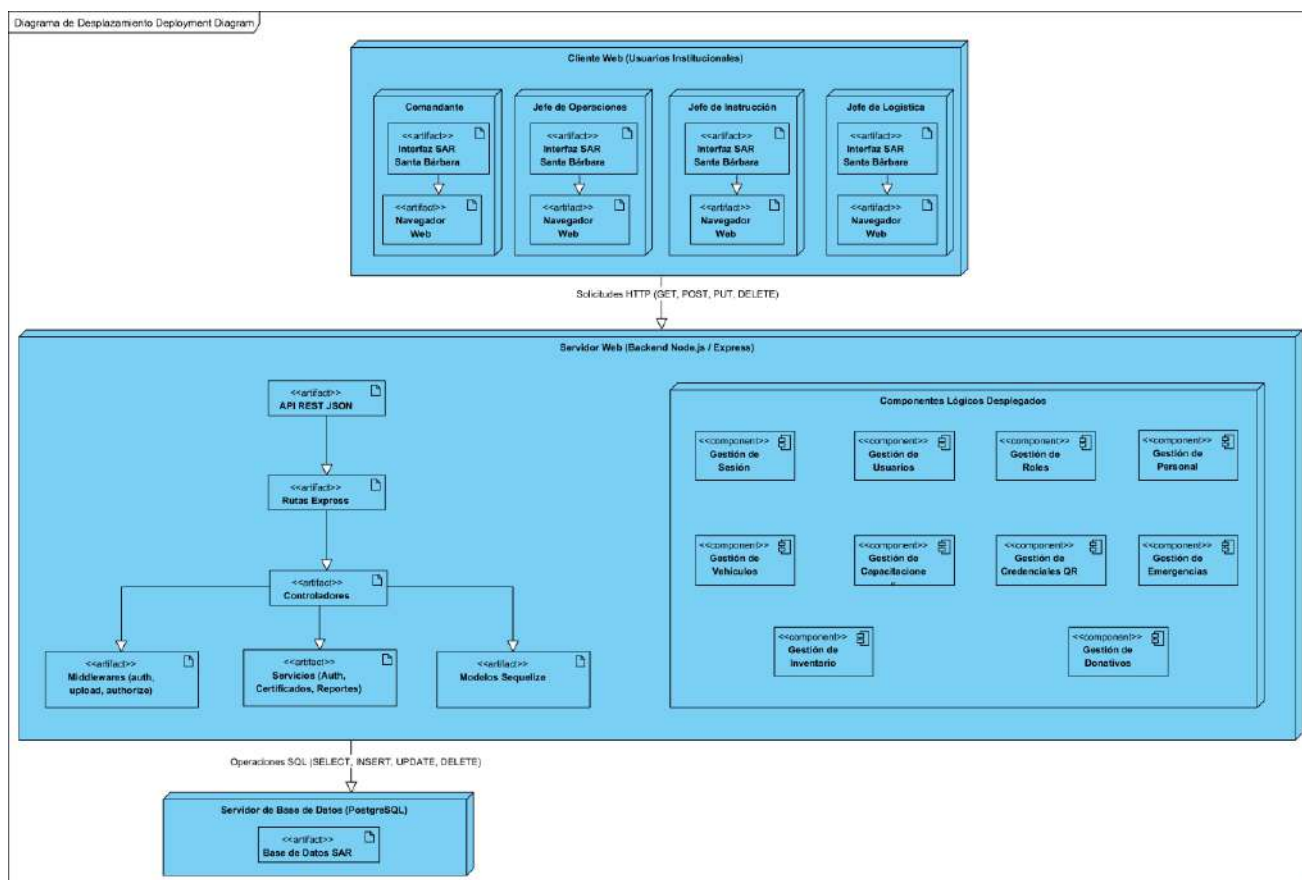


Figura 139. Diagrama de Desplazamiento
Fuente: Elaboración Propia.

II.1.2.6.16 Prototipo de Interfaces de Pantalla

II.1.2.6.16.1 Introducción

Las interfaces desarrolladas para el sistema web de gestión operativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz fueron diseñadas con el objetivo de brindar una experiencia de usuario intuitiva, ágil y accesible, permitiendo al personal institucional operar el sistema de manera eficiente en tareas administrativas y operativas. El diseño de las interfaces responde a los principios de usabilidad y consistencia visual, garantizando una navegación clara entre los diferentes módulos: autenticación, gestión de emergencias, administración de personal, inventario, vehículos, capacitaciones, donativos y credenciales digitales. Cada pantalla fue concebida para minimizar la curva de aprendizaje, facilitando la adopción tecnológica por parte del personal de distintos niveles jerárquicos.

Durante el proceso de desarrollo se siguió la metodología Rational Unified Process (RUP), integrando iteraciones sucesivas que permitieron refinar los prototipos a partir de las pruebas con usuarios finales. En las fases de Elaboración y Construcción, los prototipos iniciales se validaron mediante pruebas de usabilidad y revisiones funcionales, asegurando que el diseño final cumpliera con los requerimientos establecidos en la norma IEEE 830 y los estándares de accesibilidad web.

El Frontend del sistema fue desarrollado con tecnologías modernas como React, Vite y Tailwind CSS, lo que permitió construir interfaces responsivas y modulares, optimizadas para distintos tamaños de pantalla y dispositivos. Asimismo, se aplicaron principios de diseño utility-first, priorizando la eficiencia visual y el tiempo de respuesta de cada componente interactivo.

Las siguientes figuras presentan los prototipos de interfaces de pantalla correspondientes a los principales módulos del sistema. Cada interfaz incluye una breve descripción de su funcionalidad, el rol del usuario que la utiliza y su relación con los diagramas UML elaborados en la fase de diseño.

II.1.2.6.16.2 Gestionar Inicio de Sesión

- Iniciar Sesión

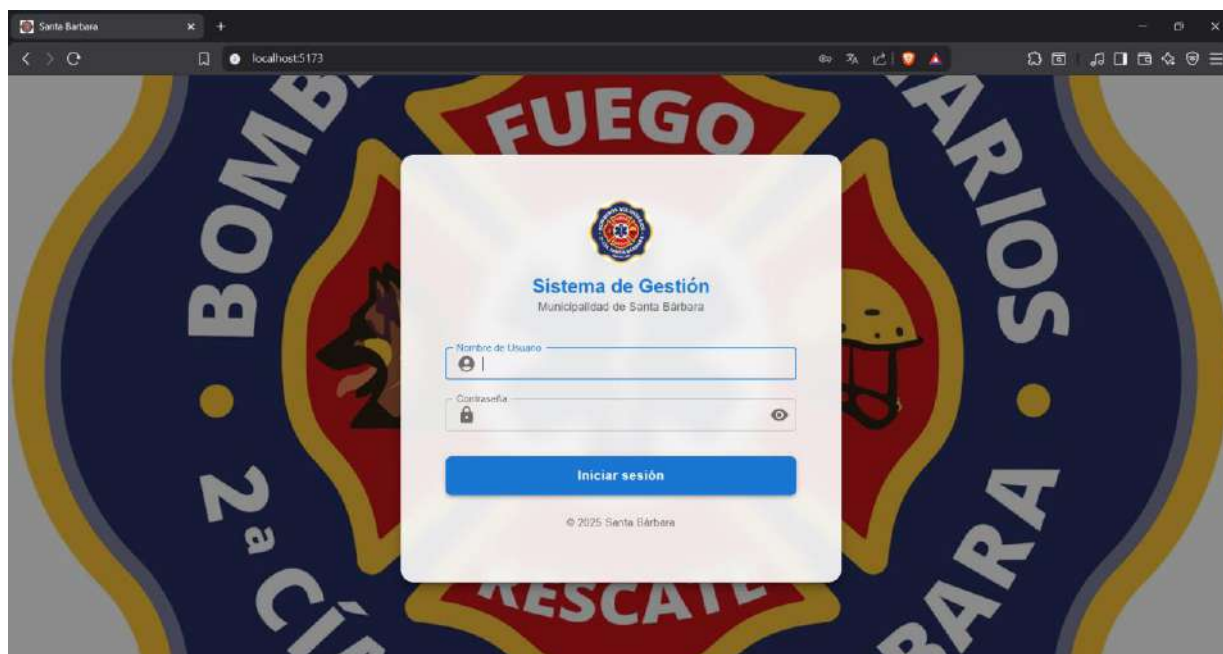


Figura 140. Interfaz: Pantalla Gestionar Sesión-Iniciar Sesión
Fuente: Elaboración Propia.

- Cerrar Sesión

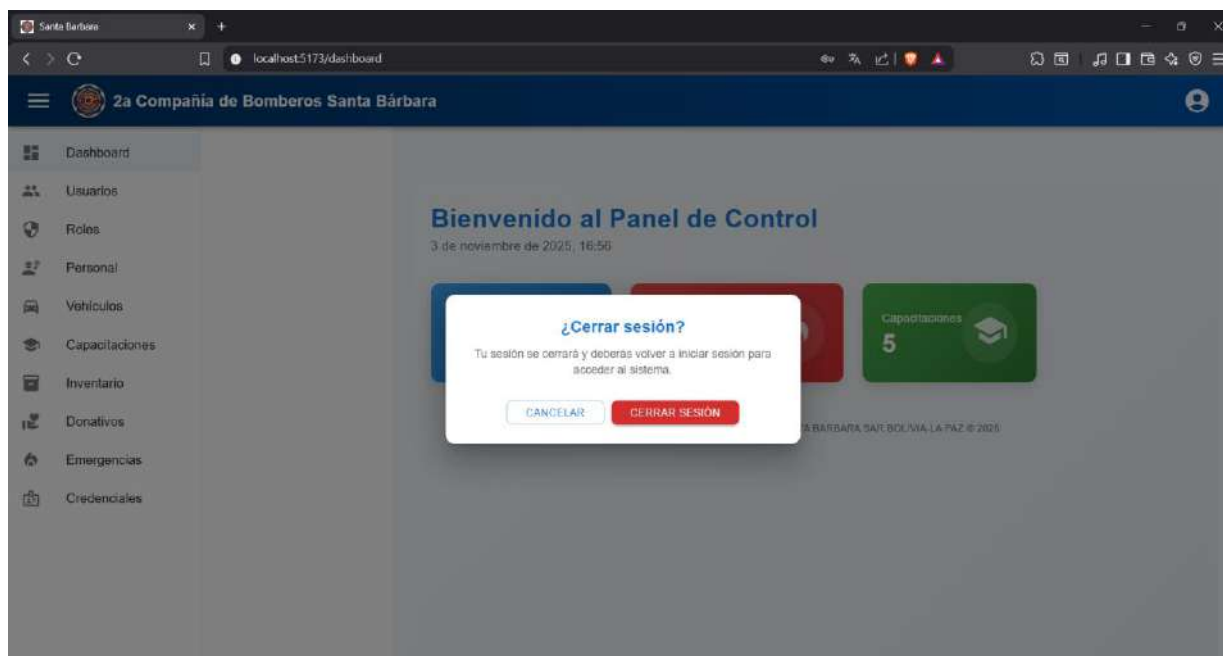


Figura 141. Interfaz: Pantalla Gestionar Sesión-Cerrar Sesión
Fuente: Elaboración Propia.

II.1.2.6.16.3 Gestión de Usuario

- Listar Usuarios

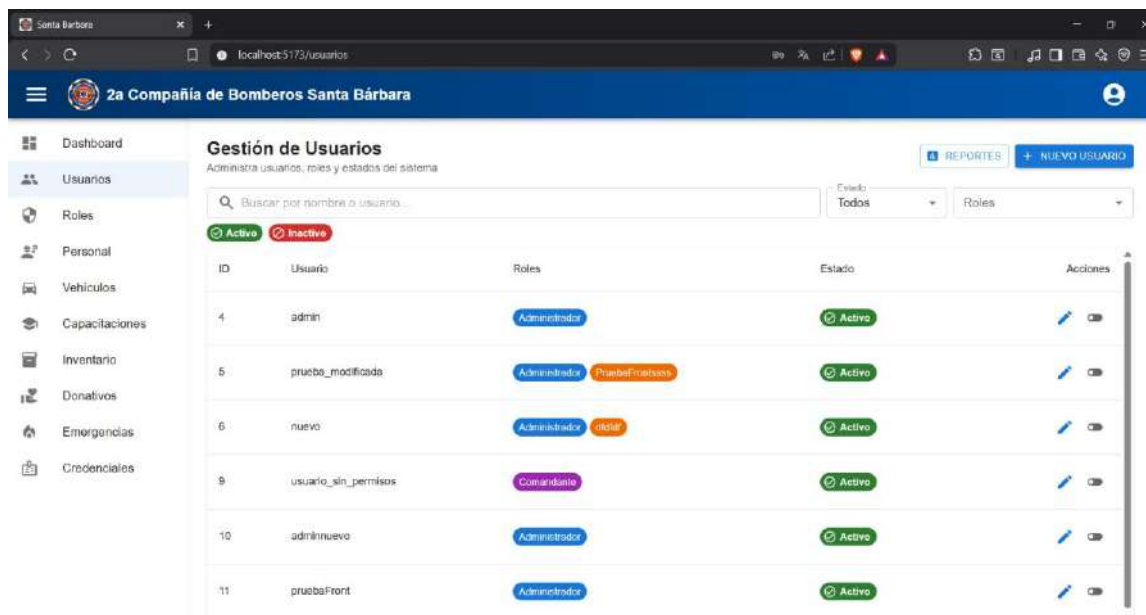


Figura 142. Interfaz: Pantalla Gestión de Usuarios-Listar Usuario
Fuente: Elaboración Propia.

- Registrar Nuevo Usuario

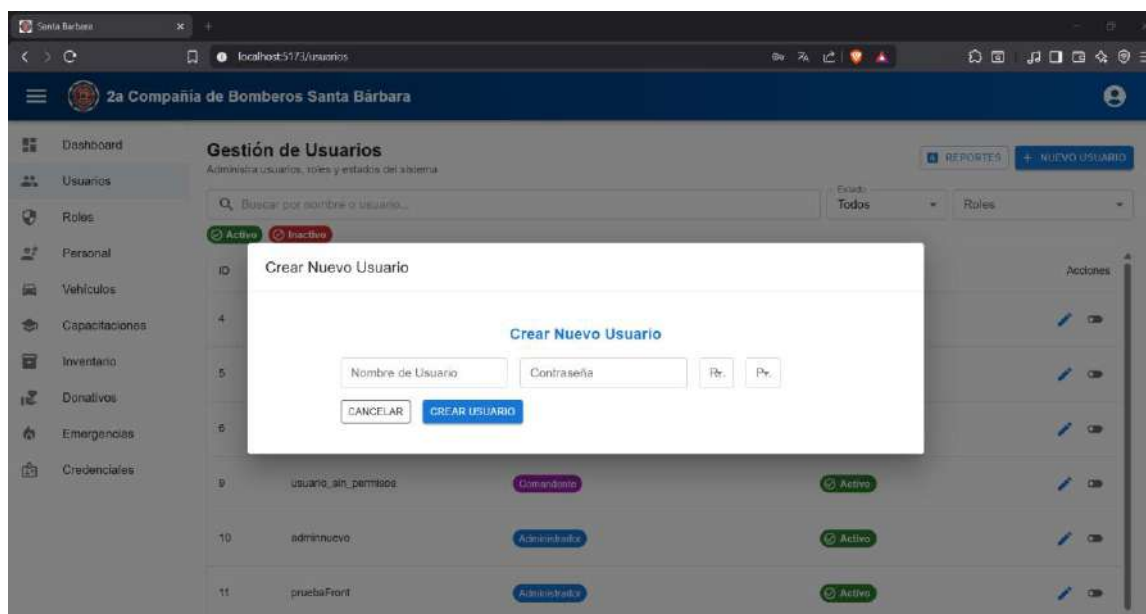


Figura 143. Interfaz: Pantalla Gestión de Usuarios-Registrar Nuevo Usuario
Fuente: Elaboración Propia.

- **Editar/Actualizar Usuario**

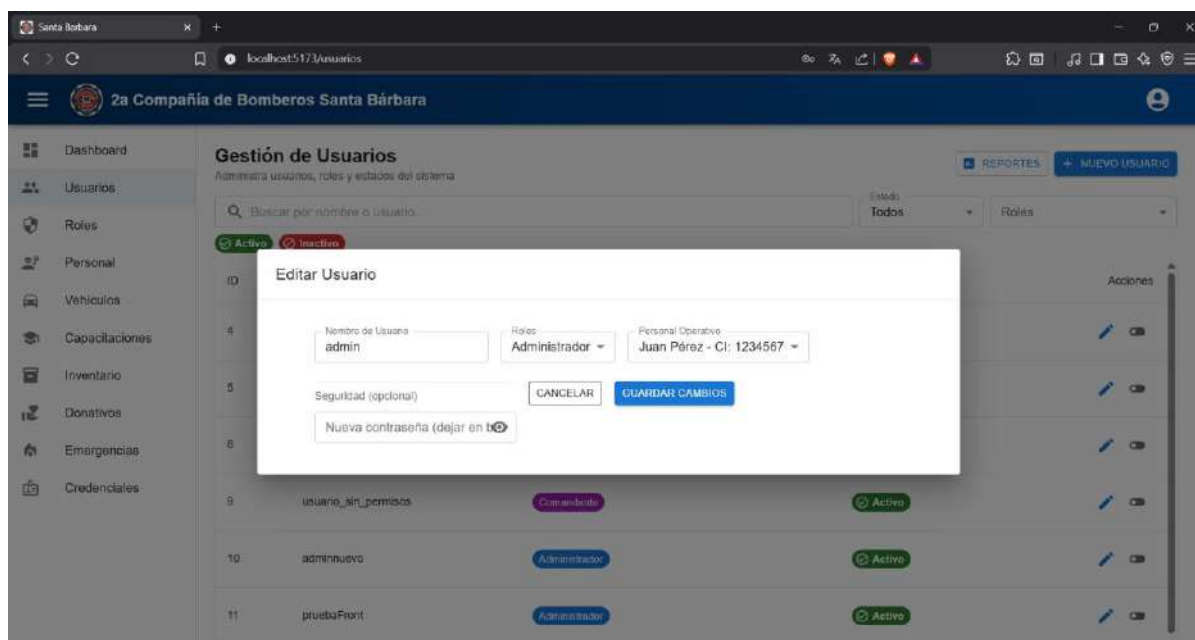


Figura 144. Interfaz: Pantalla Gestión de Usuarios-Editar/Actualizar Usuario
Fuente: Elaboración Propia.

- **Desactivar/Reactivar Usuario**

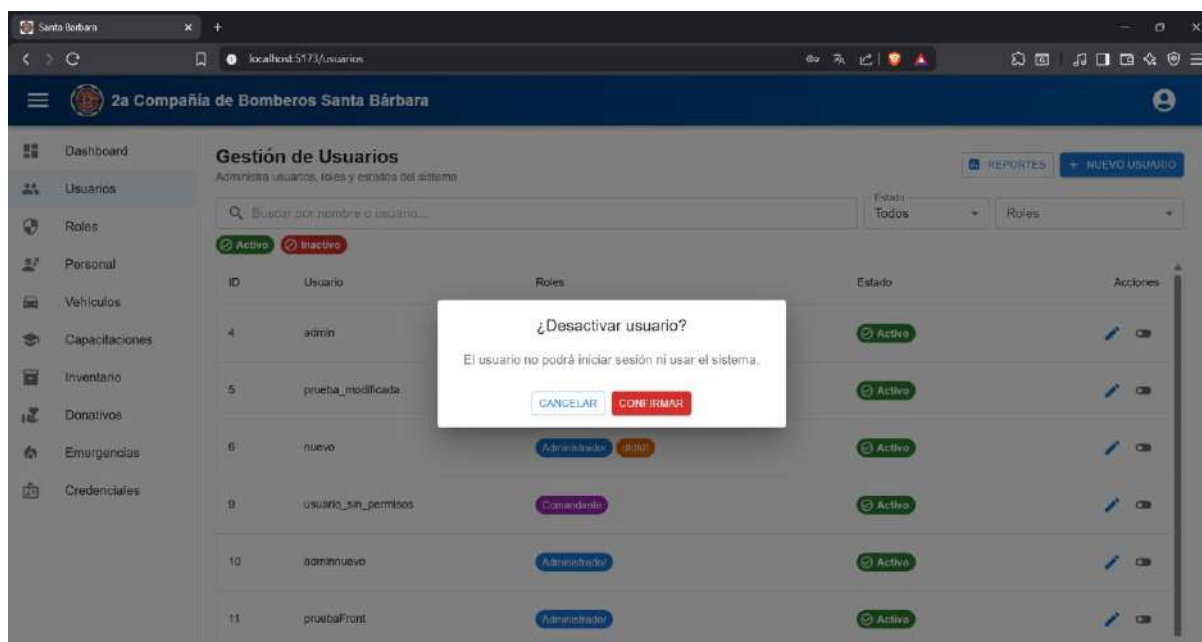


Figura 145. Interfaz: Pantalla Gestión de Usuarios-Desactivar/Reactivar Usuario
Fuente: Elaboración Propia.

- **Asignar/Eliminar Rol de Usuario**

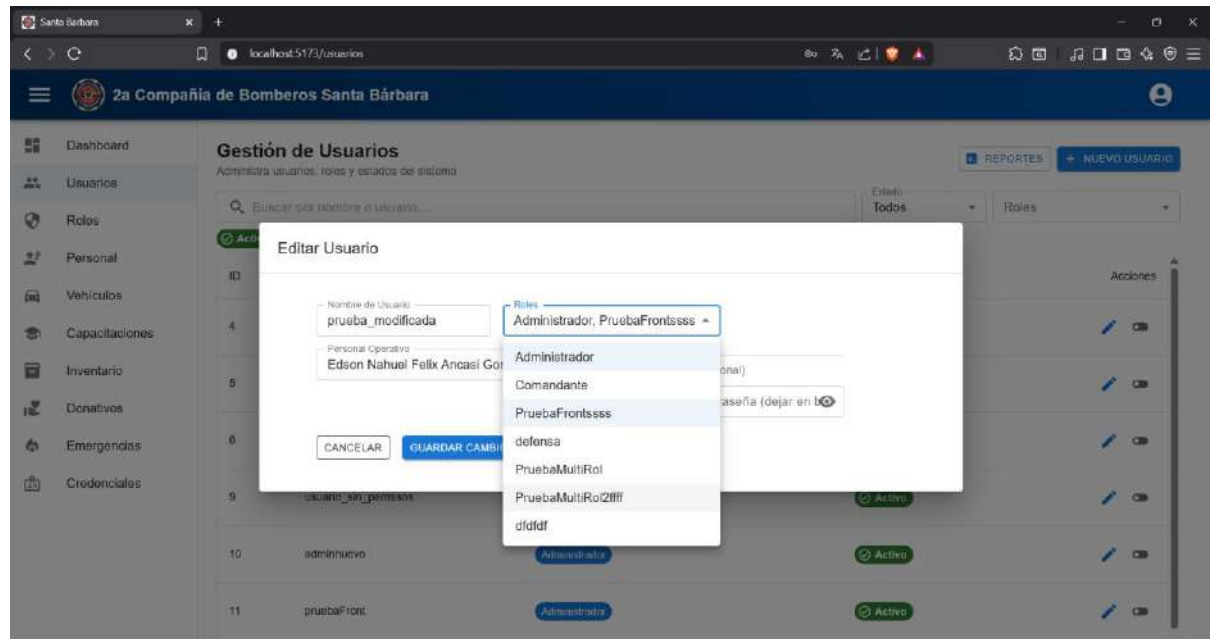


Figura 146. Interfaz: Pantalla Gestión de Usuarios-Asignar/Eliminar Rol de Usuario
Fuente: Elaboración Propia.

- **Generar Reportes de usuarios**

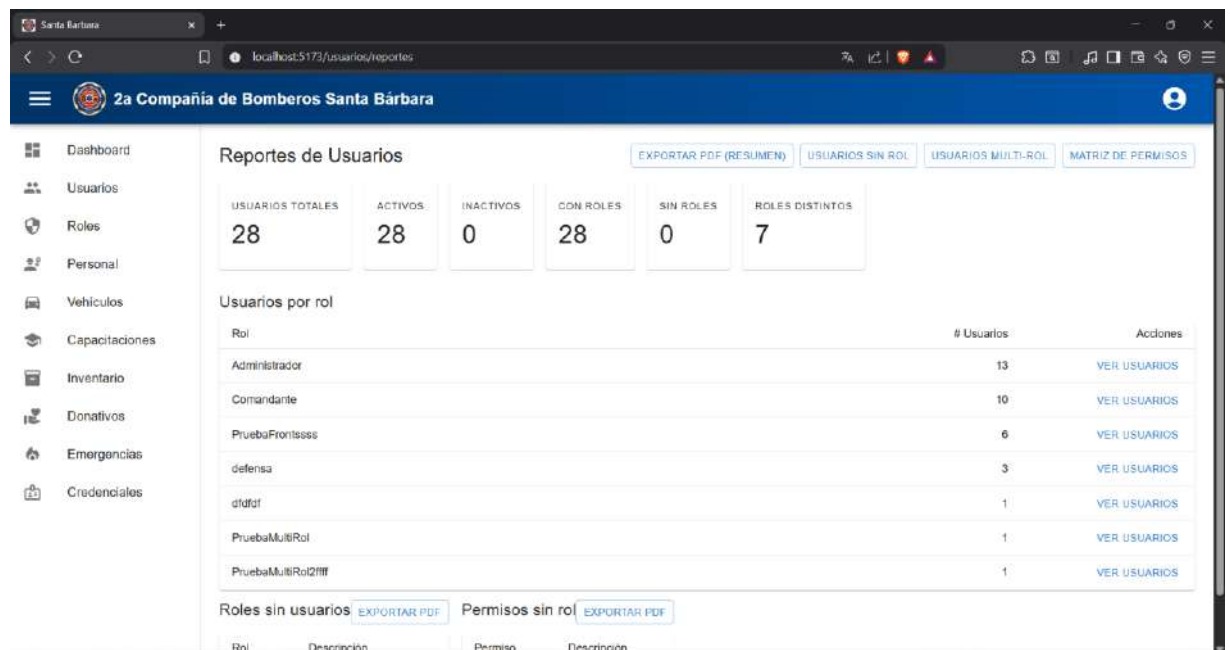


Figura 147. Interfaz: Pantalla Gestión de Usuarios-Generar Reportes de usuarios
Fuente: Elaboración Propia.

II.1.2.6.16.4 Gestión de Roles

- Listar Roles

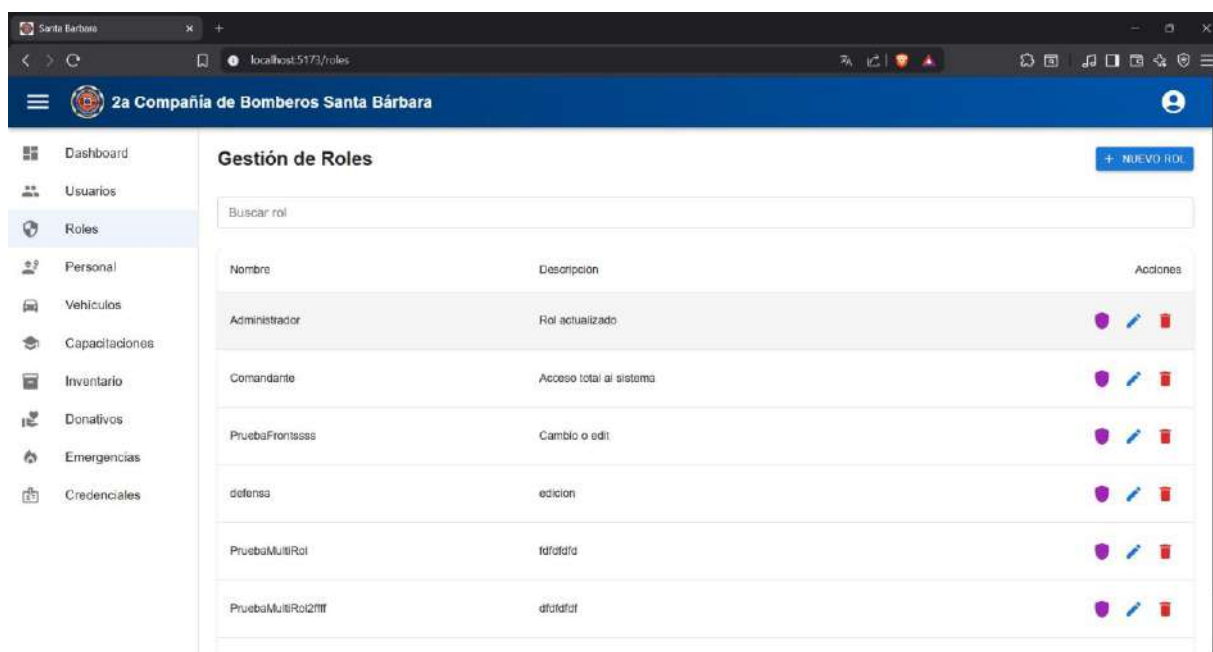


Figura 148. Interfaz: Pantalla Gestión de Roles-Listar Roles
Fuente: Elaboración Propia.

- Registrar Nuevo Rol

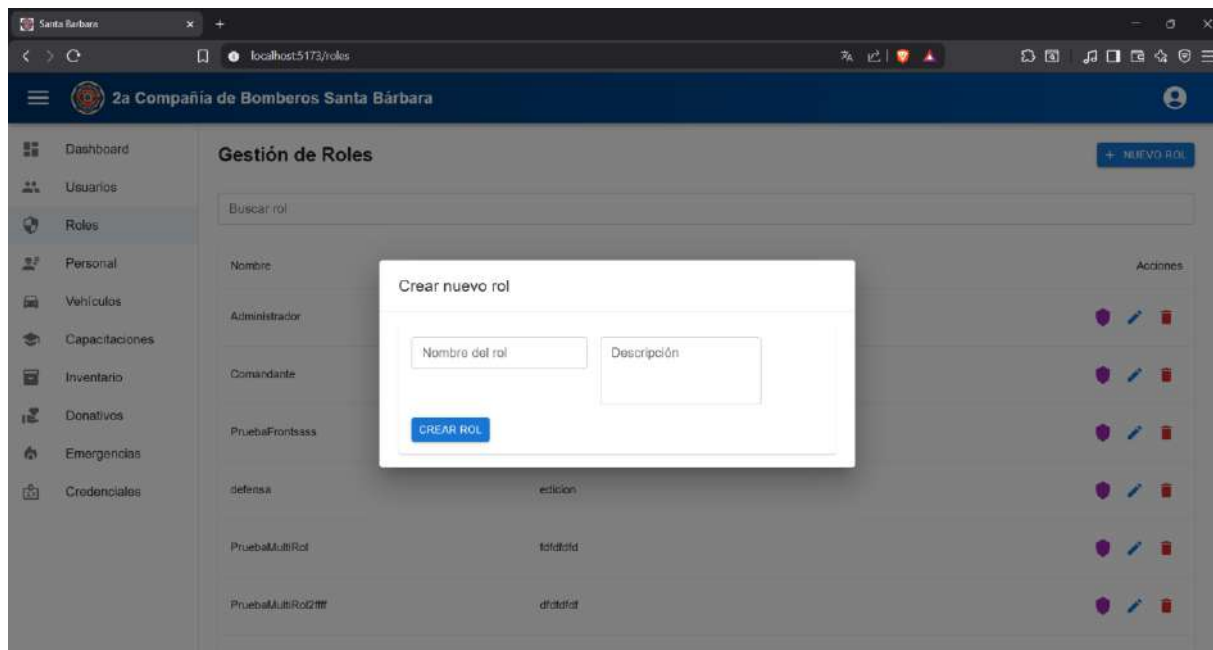


Figura 149. Interfaz: Pantalla Gestión de Roles-Registrar Nuevo Rol
Fuente: Elaboración Propia.

- **Editar / Actualizar Rol**

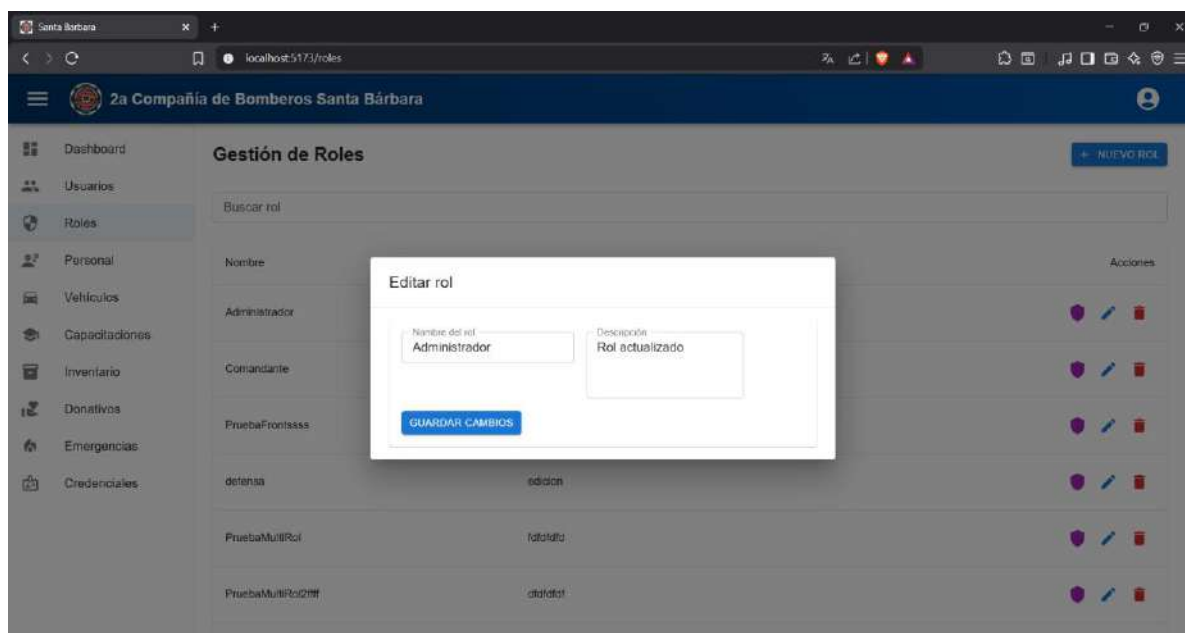


Figura 150. Interfaz: Pantalla Gestión de Roles-Editar/Actualizar Rol
Fuente: Elaboración Propia.

- **Eliminar Rol**

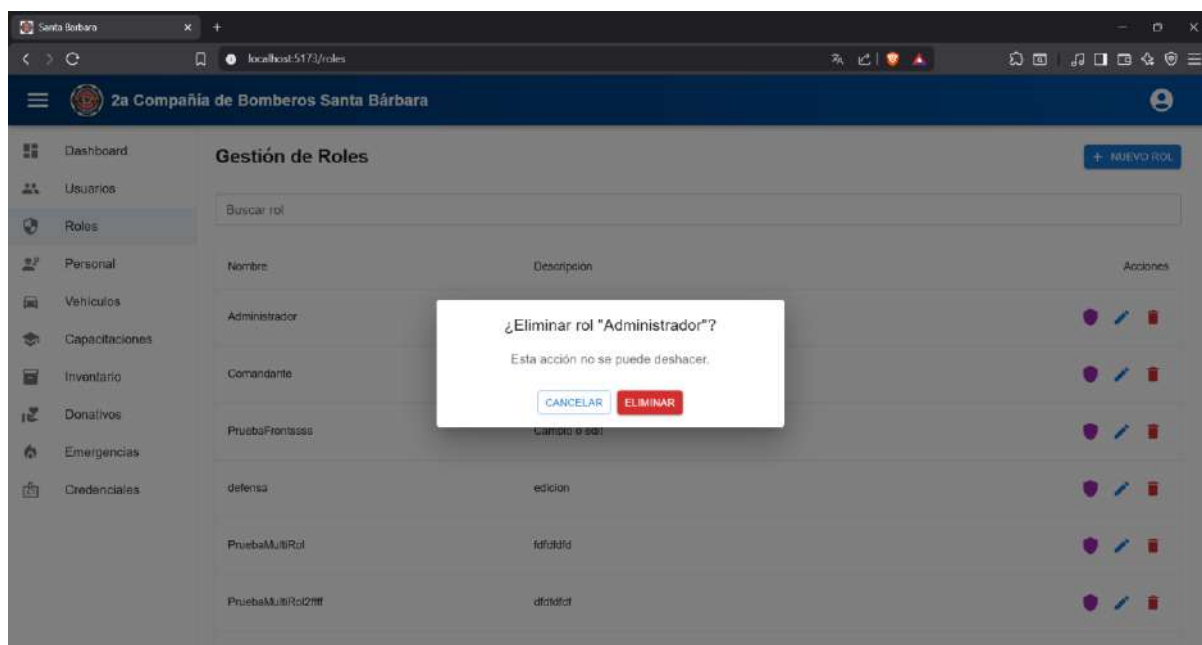


Figura 151. Interfaz: Pantalla Gestión de Roles-Eliminar Rol
Fuente: Elaboración Propia.

- **Asignar Permisos a Rol**

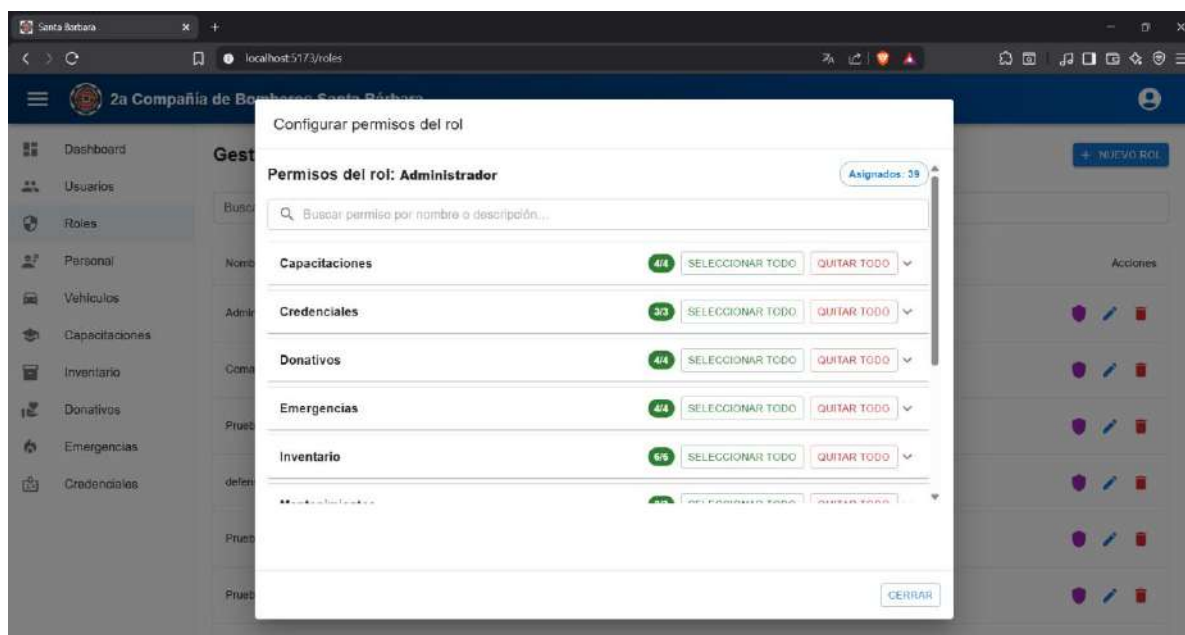


*Figura 152. Interfaz: Pantalla Gestión de Roles-Asignar Permisos a Rol
Fuente: Elaboración Propia.*

- **Ver Permisos Asignados**

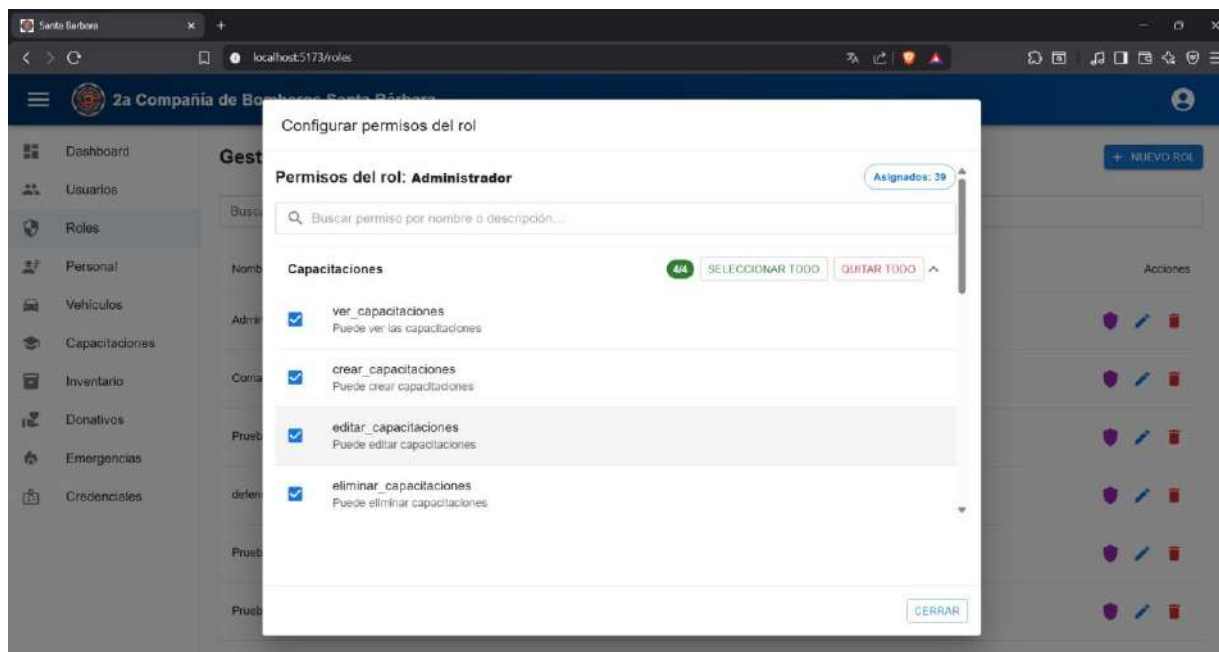


*Figura 153. Interfaz: Pantalla Gestión de Roles-Ver Permisos Asignados
Fuente: Elaboración Propia.*

- **Generar Reportes de Roles y Permisos**

Permiso \ Rol	Administrador	Comandante	defensa	dñidat	PruebaFrontssas	PruebaMultiRol	PruebaMultiRo2m
crear_capacitaciones	✓	—	—	—	—	—	—
crear_credenciales	✓	—	—	—	—	—	—
crear_donativos	✓	—	—	—	—	—	—
crear_emergencias	✓	—	✓	—	✓	—	—
crear_inventario	✓	—	—	—	—	—	—
crear_mantenimientos	✓	—	—	—	—	—	—
crear_personal	✓	—	—	—	—	—	—
crear_roles	✓	—	—	—	—	—	—
crear_usuarios	✓	—	✓	—	—	—	—
crear_vehiculos	✓	—	—	—	—	—	—
editar_capacitaciones	✓	—	—	—	—	—	—
editar_credenciales	✓	—	—	—	—	—	—
editar_donativos	✓	—	—	—	—	—	—
editar_emergencias	✓	—	✓	—	✓	—	—
editar_inventario	✓	—	—	—	—	—	—
editar_personal	✓	—	—	—	—	—	—
editar_roles	✓	—	—	—	—	—	✓
editar_usuarios	✓	—	✓	—	—	—	—

Figura 154. Interfaz: Pantalla Gestión de Roles-Generar Reportes de Roles y Permisos
Fuente: Elaboración Propia.

II.1.2.6.16.5 Gestión de Personal

- **Listar Personal**

Personal
Gestión de voluntarios y personal operativo

Buscar por nombre, CI, grado o cohorte

Nombre	Clase	Grado	Estado
JOse cespedes	Clase 2024 / Brigadista	CI: 7788896 Tel: 55888888	Activo
pruebareportes ddsdssssss	Clase 2025 / Aspirante	CI: 66666688 Tel: 64316877	Activo
Yamil Gallardo	Clase 2021 / Brigadista	CI: 45454545 Tel: 65056505	Activo
Juan hashajk	Clase 2025 / Capitan	CI: 1234567890 Tel: 64316874	Activo
Silvana Paz	Clase 2025 / Aspirante	CI: 56565655 Tel: 5656567878	Activo

Nombre	Clase	Grado
Edson Nahuel Felix Ancasi Gomez	Clase 2022 / Bombero	
Luis Rodriguez	Clase 2022 / Paramedico	
Ana Garcia	Clase pendiente / Bombero	
Juan Pérez	Clase pendiente / Bombero	
Luis SinPermisos	Clase 2023 / Bombero II	

Figura 155. Interfaz: Pantalla Gestión de Personal-Listar Personal
Fuente: Elaboración Propia.

- **Registrar Personal**

Nuevo Personal

Nombre: Apellido: CI:

Teléfono: Fecha de nacimiento: dd/mm/aaaa Clase: Grado:

ADJUNTAR DOCUMENTO (PDF/IMAGEN)

Ningún archivo seleccionado

Figura 156. Interfaz Pantalla Gestión de Personal-Registrar Personal
Fuente: Elaboración Propia.

- **Editar / Actualizar Información de Personal**

Editar Personal

Modifica los datos del personal.

Nombre: JOse Apellido: cospedes

CI: 7788990 Teléfono: 55888888 Fecha de nacimiento: 31/10/2025 Clase (cohort): C...

NUEVA CLASE Grado (rango): B... NUEVO GRADO

Figura 157. Interfaz: Pantalla Gestión de Personal-Editar/Actualizar Información de Personal
Fuente: Elaboración Propia.

- **Asignar Grado y Clase**

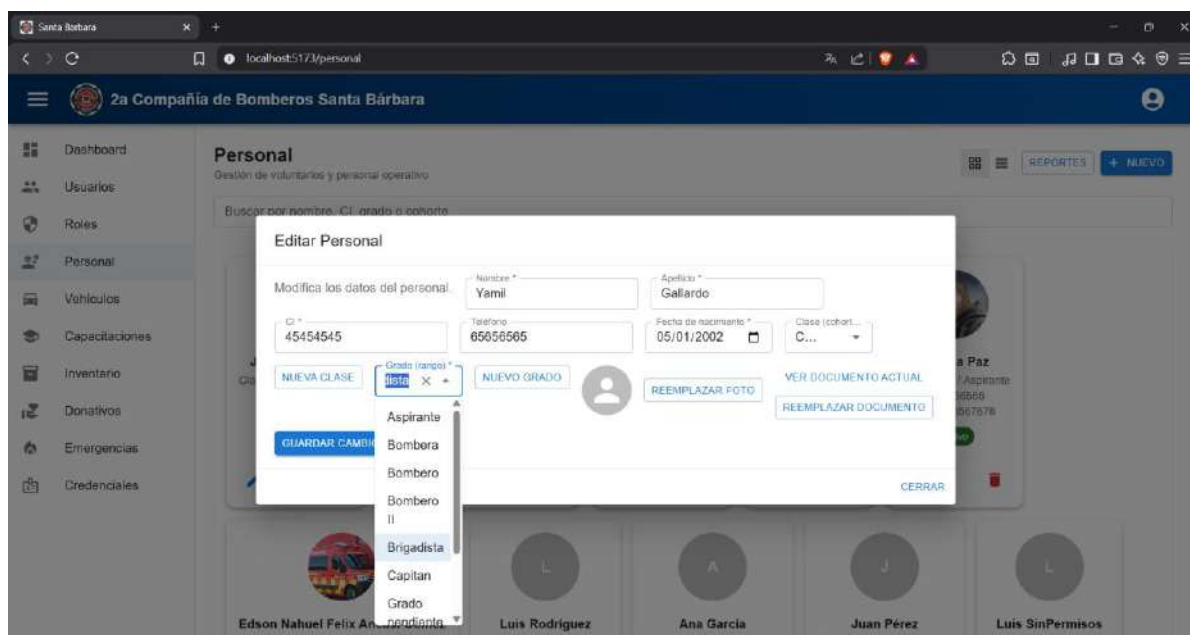


Figura 158. Interfaz: Pantalla Gestión de Personal-Asignar Grado y Clase
Fuente: Elaboración Propia.

- **Cambiar Estado del Personal**

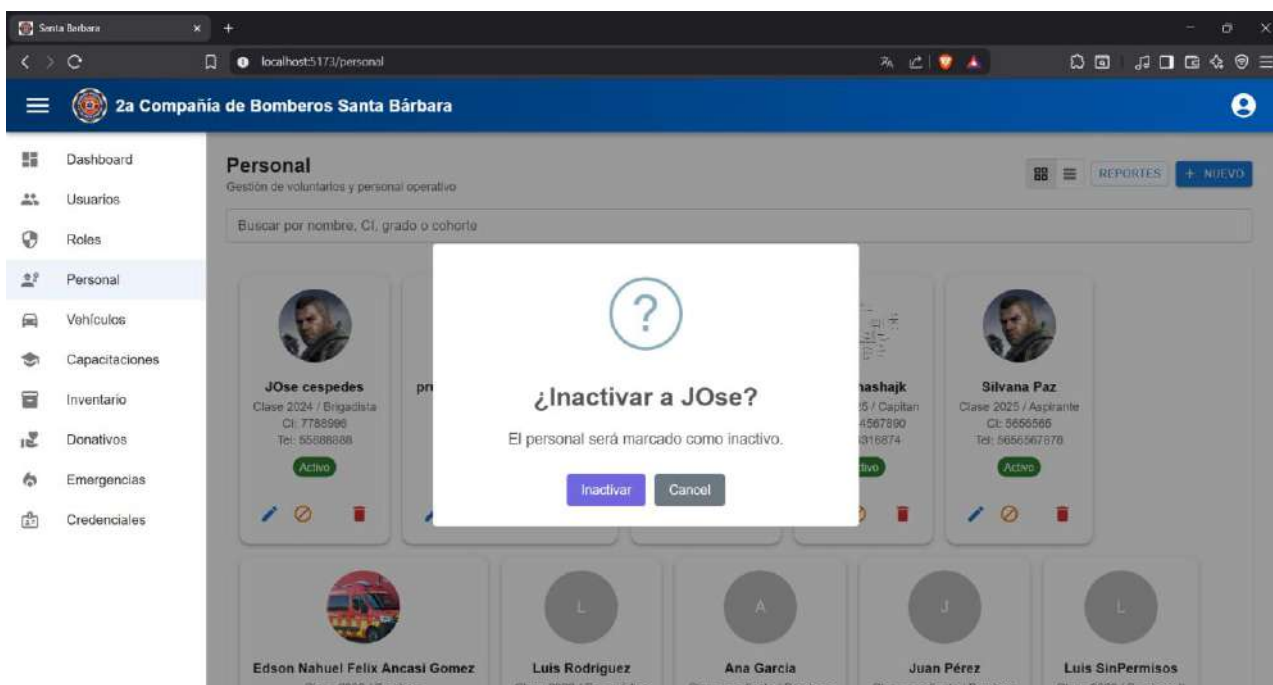


Figura 159. Interfaz: Pantalla Gestión de Personal-Cambiar Estado del Personal
Fuente: Elaboración Propia.

- **Generar Reportes del Personal**

Reportes de Personal

Buscar (nombre, apellido...) **BUSCAR** PDF

Total Personal: 11 Clases (cohortes): 6 Grados: 7 Incompletos (legajo): 9

Legajo

ID	Apellido	Nombre	CI	Clase	Grado	Teléfono
9	Ancasi Gomez	Edson Nahuel Felix	6595896	Clase 2022	Bombero	64316874
29	oespedes	JOsé	7788996	Clase 2024	Brigadista	55888888
20	dsdsdsdsds	pruebareportes	66666668	Clase 2025	Aspirante	64316877
19	Gallardo	Yameli	45454545	Clase 2021	Brigadista	65656565
6	García	Ana	2345678	Clase pendiente	Bombero	71111111
18	hashajk	Juan	1234567890	Clase 2025	Capitan	64316874
16	Paz	Silvana	5656565	Clase 2025	Aspirante	56565678
1	Pérez	Juan	12345678	Clase 2023	Bombero II	77777777
5	Pérez	Juan	1234567	Clase pendiente	Bombero	70000000

Figura 160. Interfaz: Pantalla Gestión de Personal-Generar Reportes del Personal
Fuente: Elaboración Propia.

II.1.2.6.16.6 Gestión de Vehículos

- **Listar Vehículos**

Vehículos

Gestión de vehículos operativos y su estado actual

Actualizar Reportes + Nuevo

Buscar (placa, marca, tipo o nominación) Estado LIMPIAR

Placa	Modelo	Estado
GGG-7896	Prueba Módulo Full 3 Sprintersss87 Carro Bomba K-11	Operativo
AAA-1411	Prueba Módulo Full 3 Sprinter Prueba Módulo 3 Q-11	Operativo
4545-HJHYE	Ford Nose Rescate W-11	Operativo
XXX3334	Ford2 Nose22 Rescate M-11	Operativo
4545-hjhy	Ford nosehgj Rescate A-11	Operativo

5 5678-DEF 1 1234-ABC

Figura 161. Interfaz: Pantalla Gestión Vehículos-Listar Vehículos
Fuente: Elaboración Propia.

- **Registrar Vehículo**

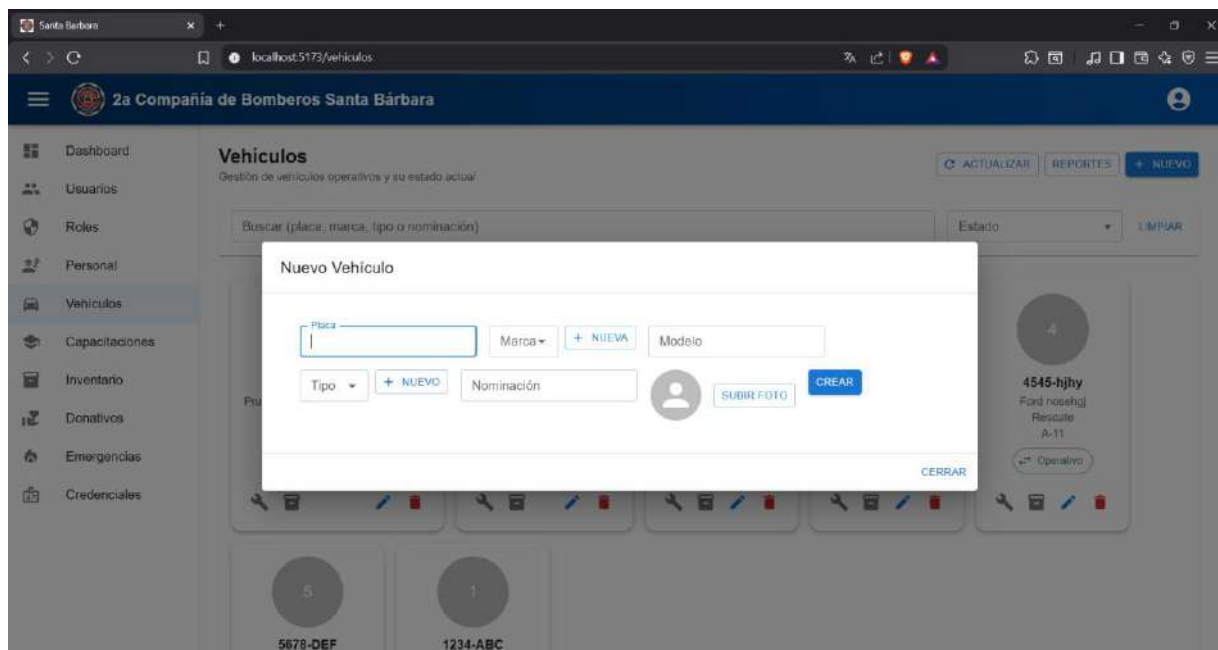


*Figura 162. Interfaz: Pantalla Gestión Vehículos-Registrar Vehículos
Fuente: Elaboración Propia.*

- **Editar / Actualizar Vehículo**

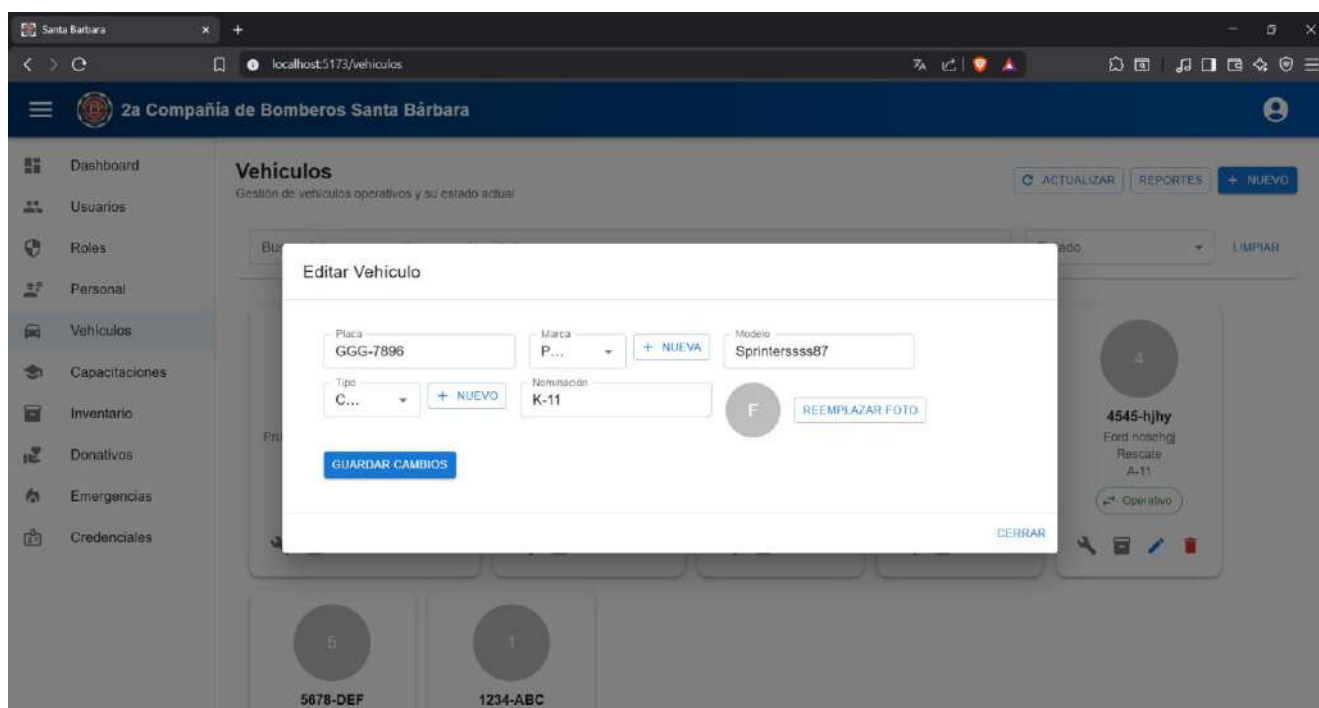


*Figura 163. Interfaz: Pantalla Gestión Vehículos-Editar/Actualizar Vehículo
Fuente: Elaboración Propia.*

- **Cambiar Estado Operativo**

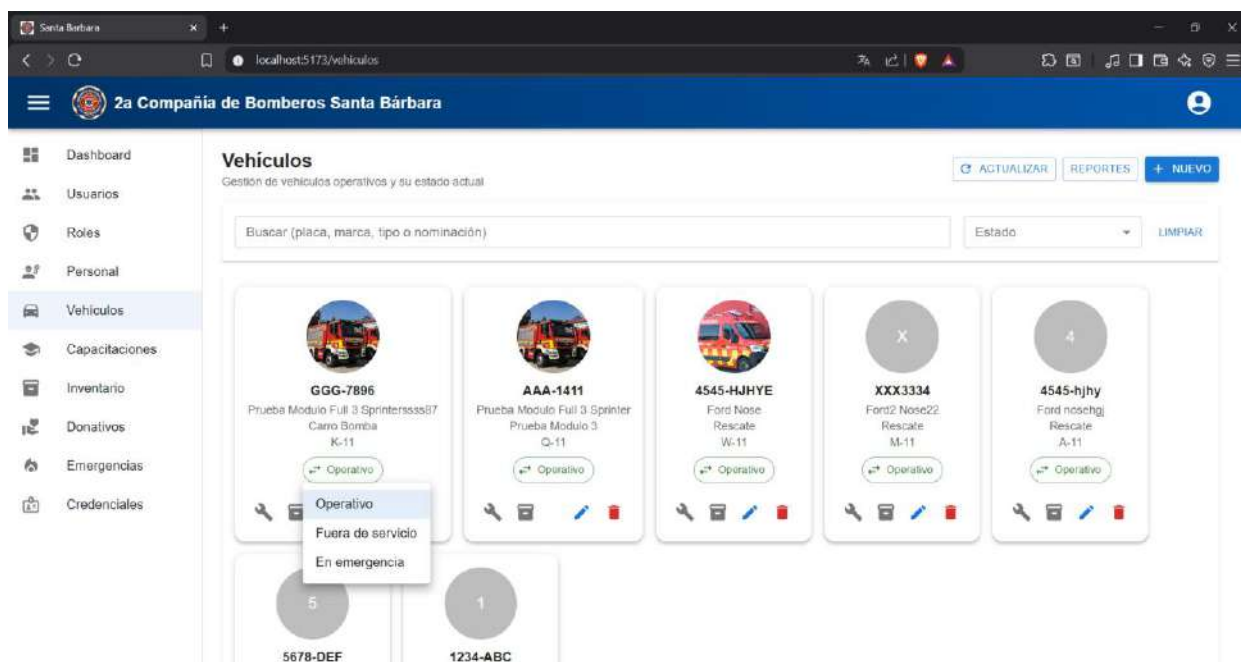


Figura 164. Interfaz: Pantalla Gestión Vehículos-Cambiar Estado Operativo
Fuente: Elaboración Propia.

- **Registrar Mantenimiento**

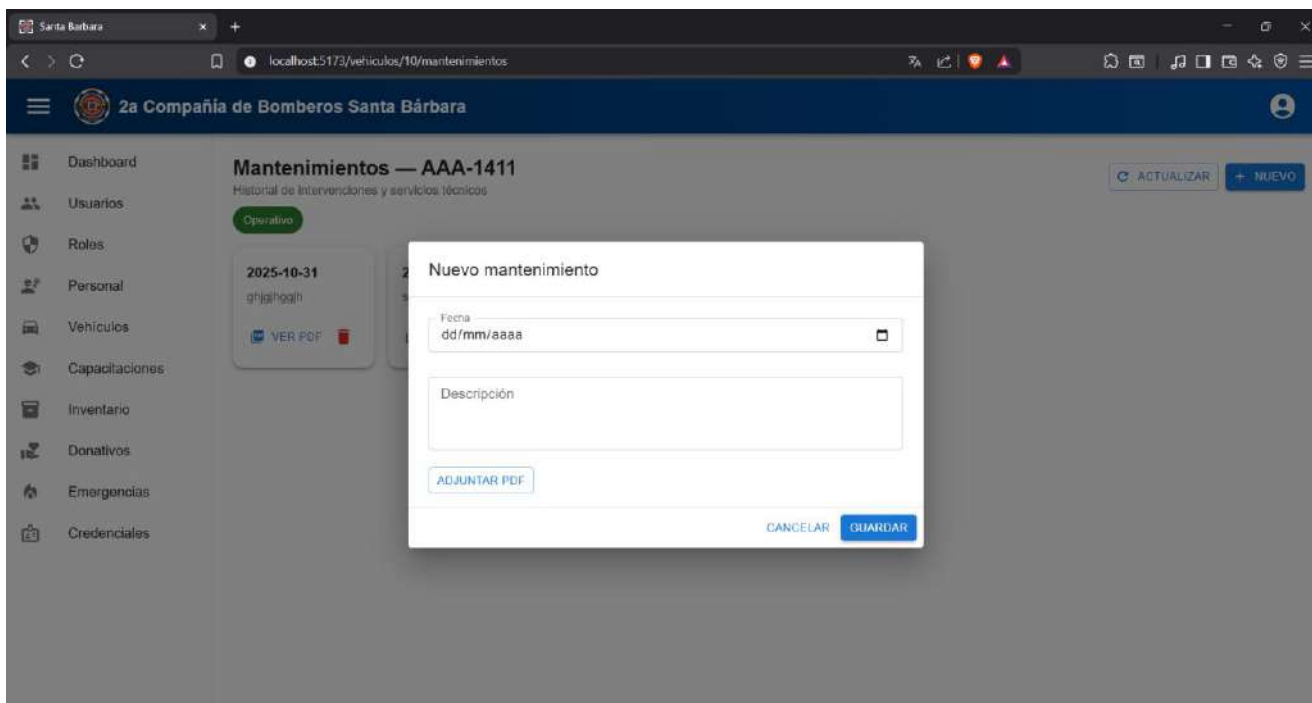


Figura 165. Interfaz: Pantalla Gestión Vehículos-Registrar Mantenimiento
Fuente: Elaboración Propia.

- **Eliminar Vehículo**

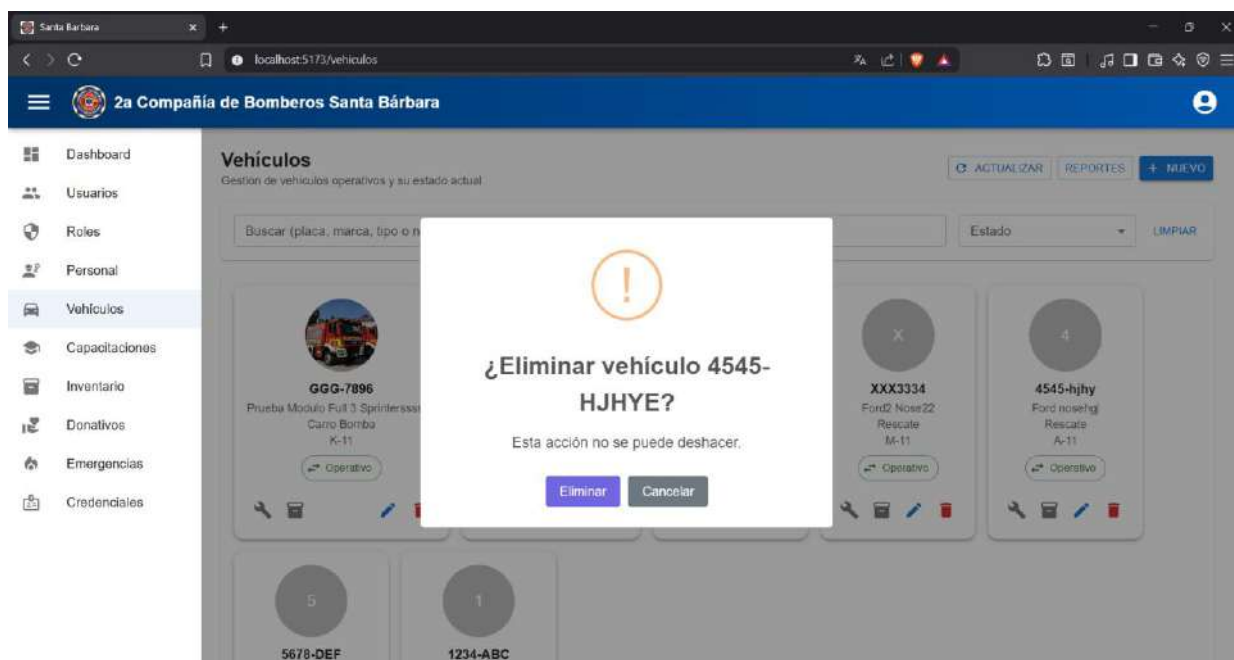


Figura 166. Interfaz: Pantalla Gestión Vehículos-Eliminar Vehículo
Fuente: Elaboración Propia.

- **Generar Reportes de Vehículos**

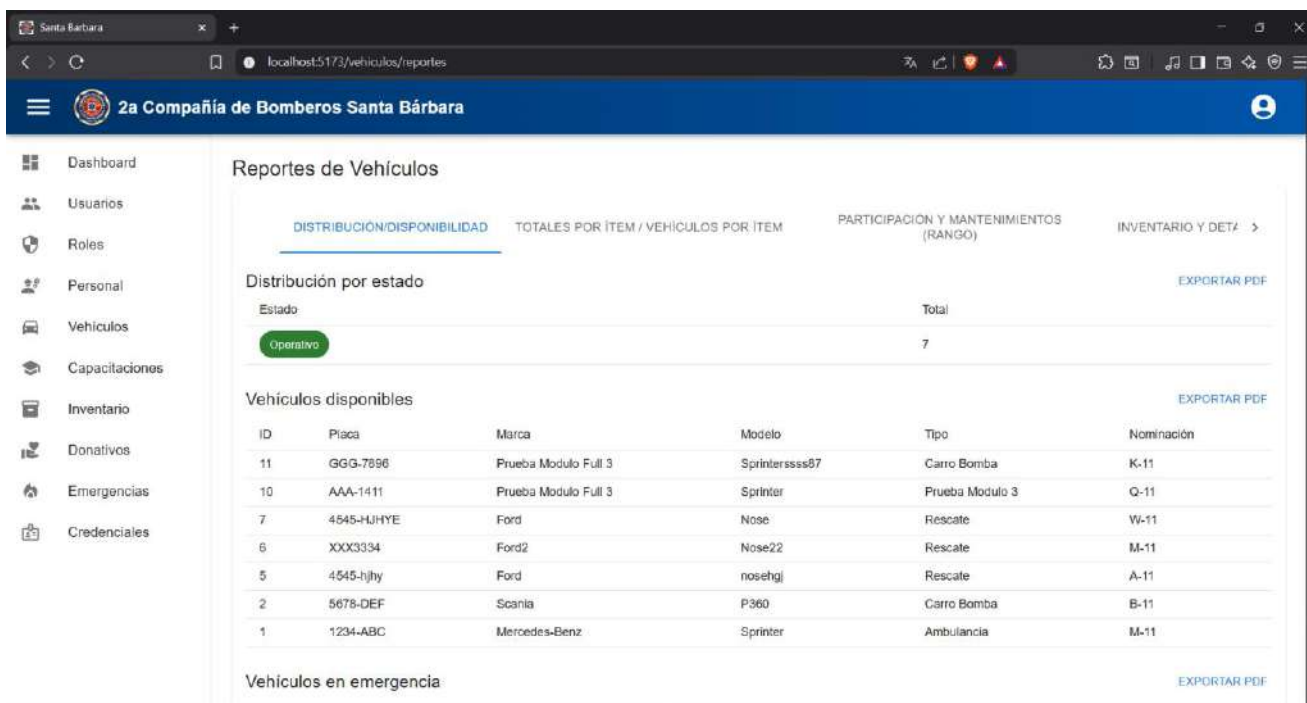


Figura 167. Interfaz: Pantalla Gestión Vehículos-Generar Reporte de vehículos
Fuente: Elaboración Propia.

II.1.2.6.16.7 Gestión de Capacitaciones

- **Listar Capacitaciones**

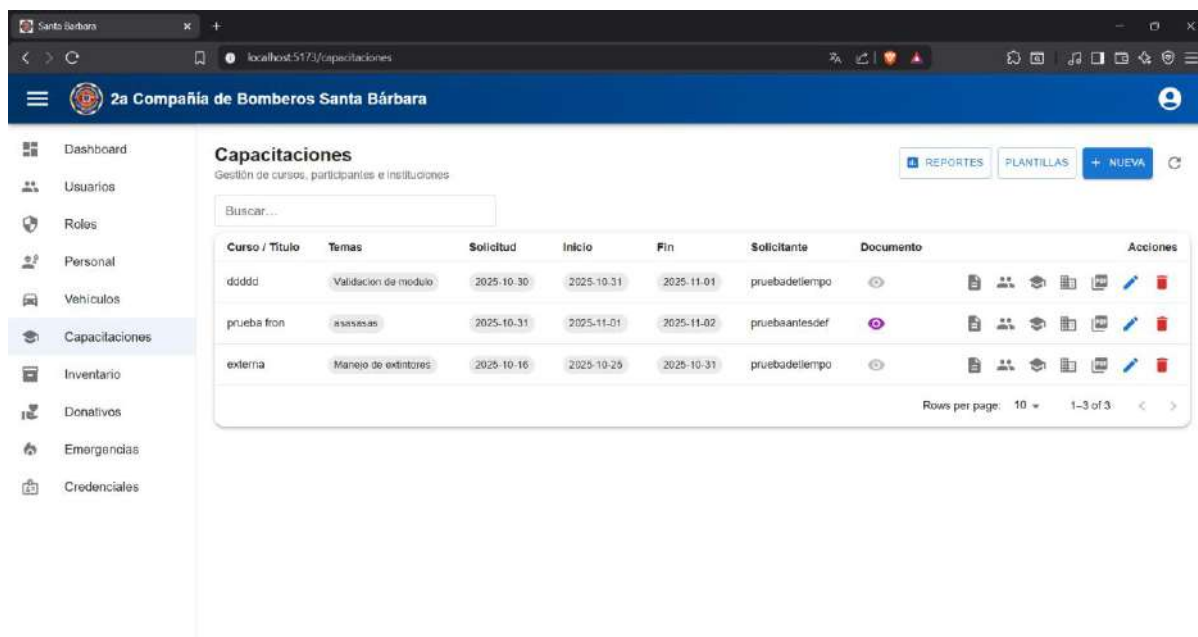


Figura 168. Interfaz: Pantalla Gestión Capacitaciones-Listar Capacitaciones
Fuente: Elaboración Propia.

- **Registrar Capacitación**

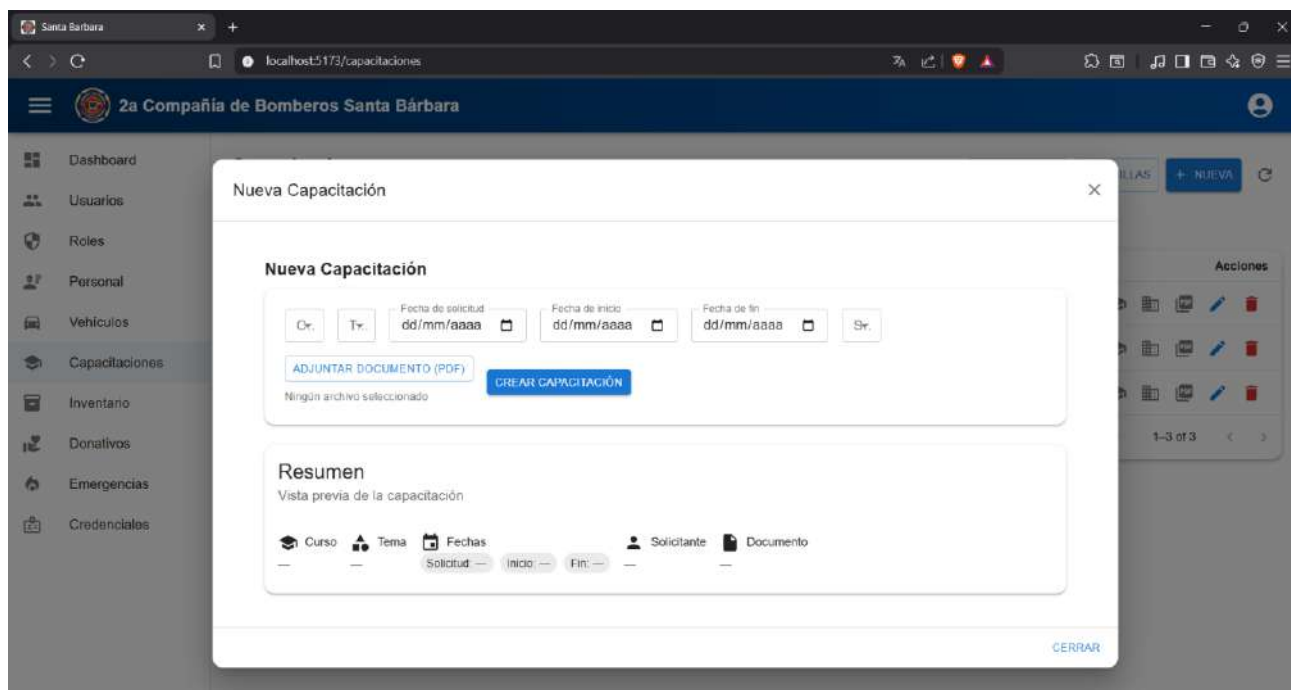


Figura 169. Interfaz: Gestión Capacitaciones-Registrar Capacitación
Fuente: Elaboración Propia.

- **Registrar Institución Externa**

The screenshot shows a web application interface for '2a Compañía de Bomberos Santa Bárbara'. The main menu on the left includes Dashboard, Usuarios, Roles, Personal, Vehículos, Capacitaciones, Inventario, Donativos, Emergencias, and Credenciales. The 'Instituciones' section is active, displaying a table with columns: Nombre, Responsable, Contacto, Dirección, and Acciones. A modal window titled 'Nueva institución' is open, containing the following fields and controls:

- Form fields: 'Nombre de la institución', 'Contacto', 'Dirección', 'Responsable de la institución', 'Nombre (nuevo)', 'Ci (nuevo)', and 'Comio (nuevo)'.
- Dropdown: 'Buscar responsable en catálogo' with the instruction 'Selecciona una persona existente'.
- Buttons: 'USAR CATÁLOGO', 'CREAR NUEVO', 'CANCELAR', and 'CREAR Y ASIGNAR'.

Figura 170. Interfaz: Gestión Capacitaciones-Registrar Institución Externa
Fuente: Elaboración Propia.

- **Registrar Participantes (Internos / Externos)**

The screenshot shows the 'Participantes (Personal)' section of the web application. The main area displays a list of participants with profile pictures and names. A modal window titled 'Asignar personal' is open, showing a dropdown menu 'Selecciona personal' with the following list of options:

- Yamil Gallardo — CI: 45454545
- Ana García — CI: 2345678
- Silvana Paz — CI: 5656565
- Juan Pérez — CI: 12345678
- Juan Pérez — CI: 1234567
- Luis Rodríguez — CI: 3456789
- Luis SinPermisos — CI: 9999999

Figura 171. Interfaz: Gestión Capacitaciones-Registrar Participantes (Internos/Externos)
Fuente: Elaboración Propia.

- **Generar Certificados PDF con Código QR**

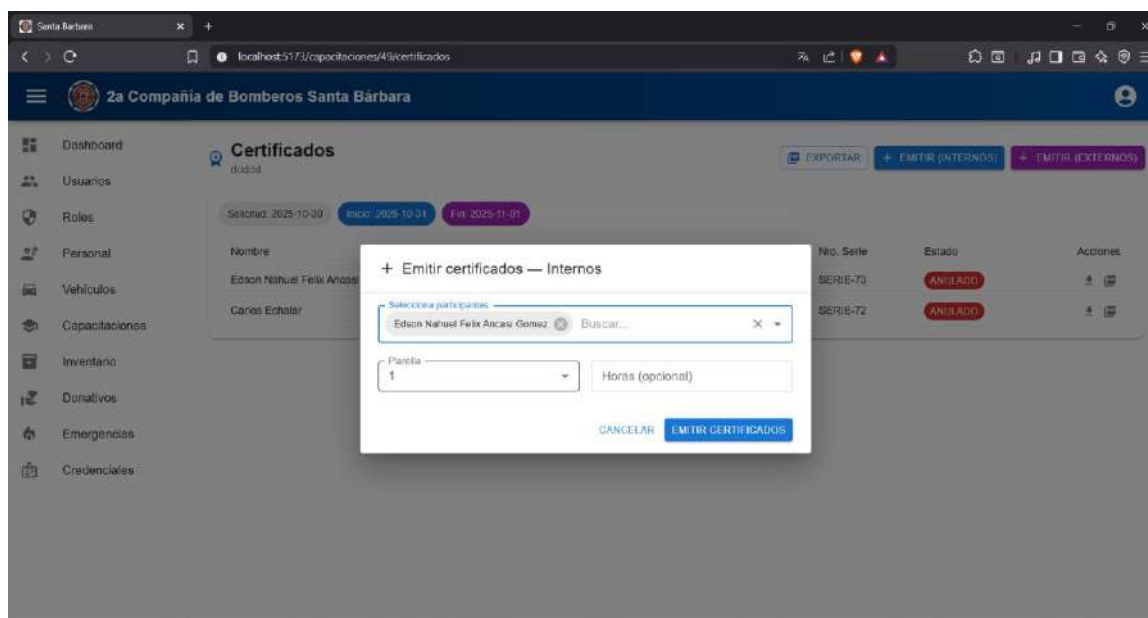


Figura 172. Interfaz: Gestión Capacitaciones-Generar Certificados PDF con Código QR
Fuente: Elaboración Propia.

- **Generar Reportes y Estadísticas**

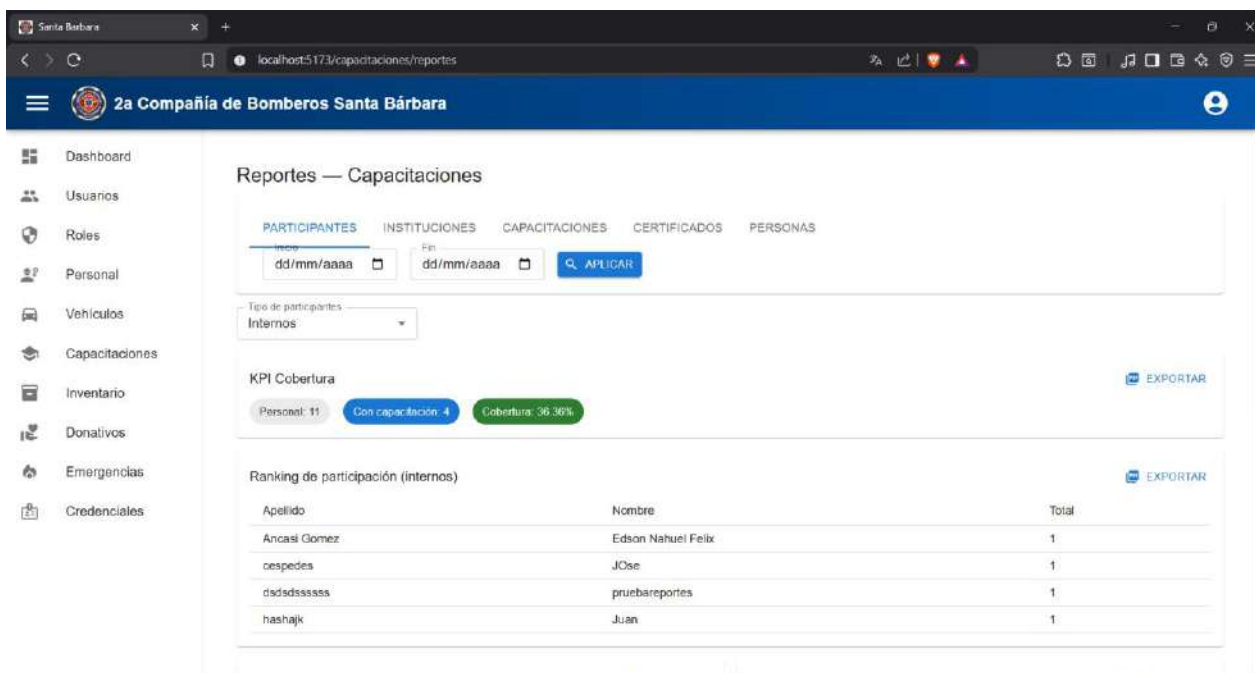


Figura 173. Interfaz: Gestión Capacitaciones-Generar Reportes y Estadísticas
Fuente: Elaboración Propia.

II.1.2.6.16.8 Gestión de Credenciales QR

- Listar Credenciales QR

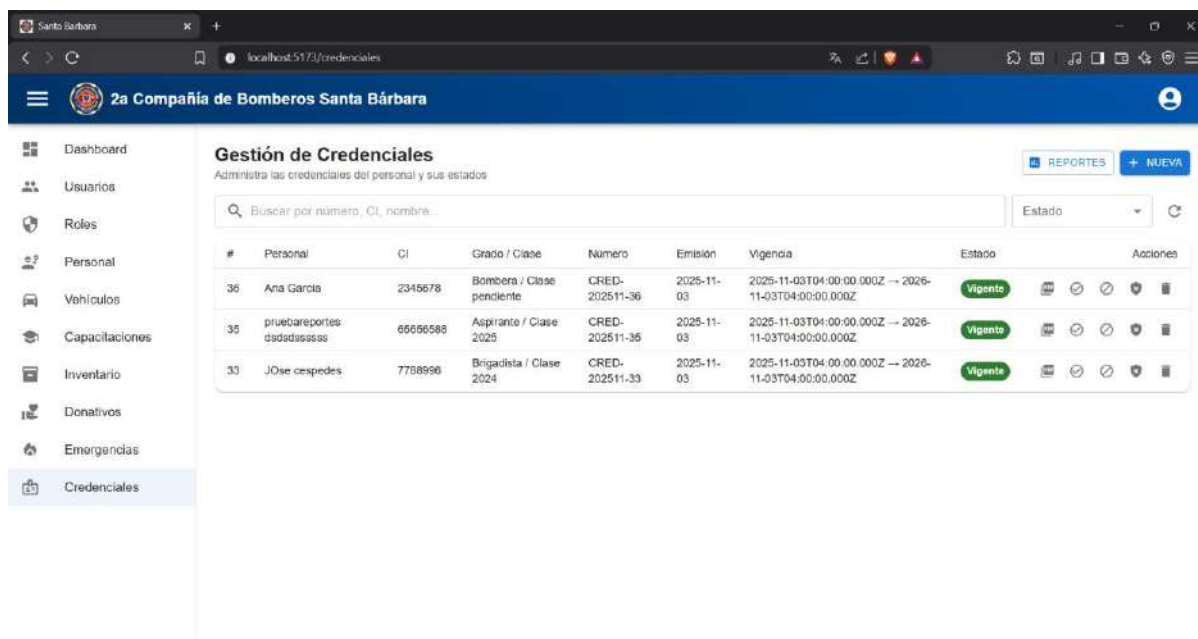


Figura 174. Interfaz: Gestión Credenciales QR-Listar Credenciales QR
Fuente: Elaboración Propia.

- Registrar Credencial

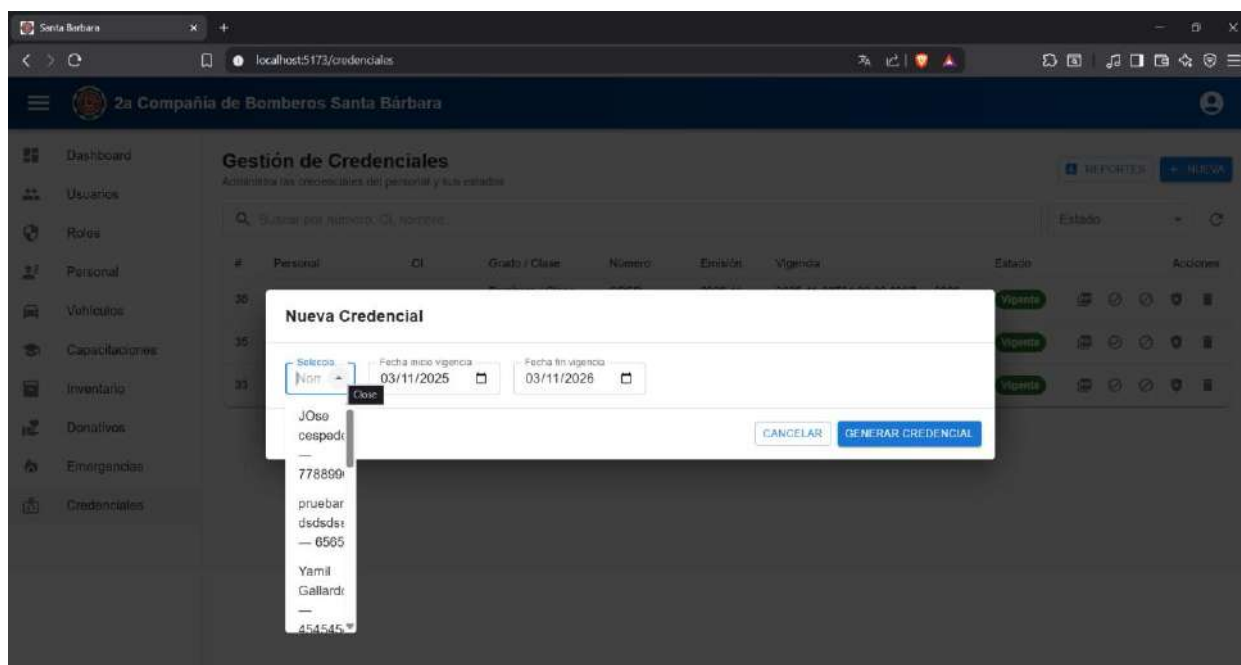


Figura 175. Interfaz: Gestión Credenciales QR-Registrar Credencial
Fuente: Elaboración Propia.

- **Actualizar / Suspendir / Revocar Credencial**

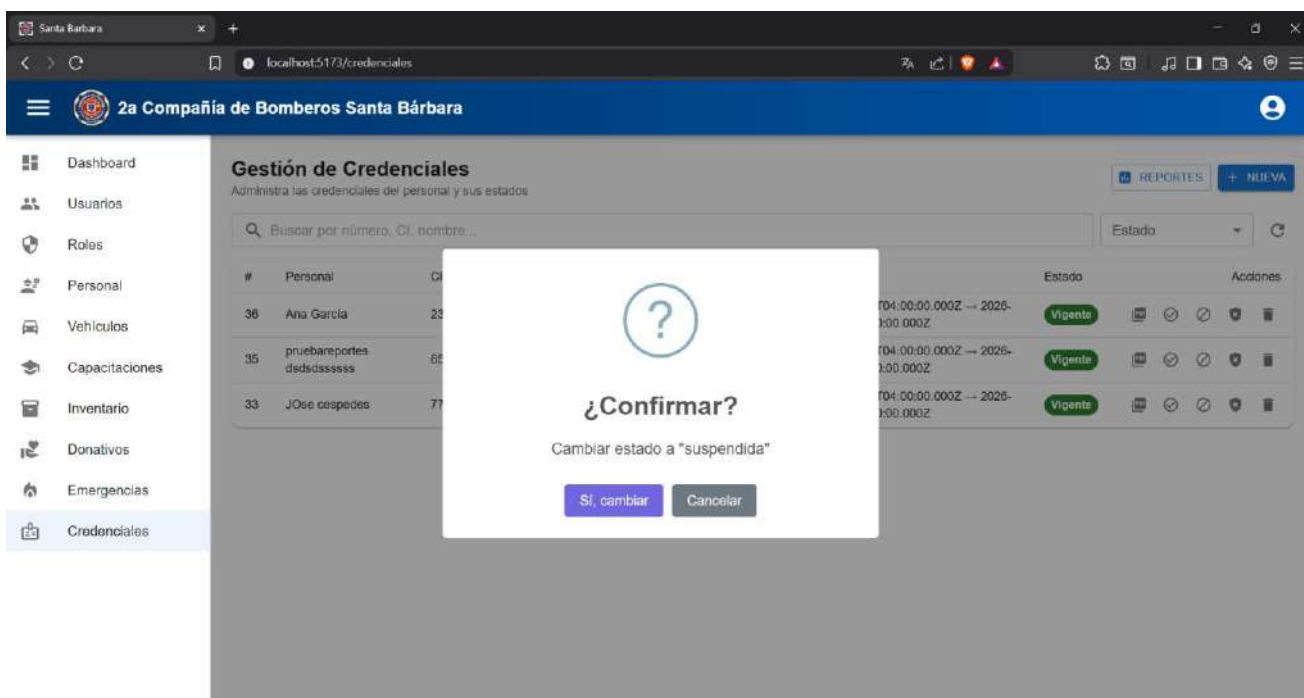


Figura 176. Interfaz: Gestión Credenciales QR-Actualizar/Suspendir/Revocar Credencial
Fuente: Elaboración Propia.

- **Generar Credencial PDF con Código QR**

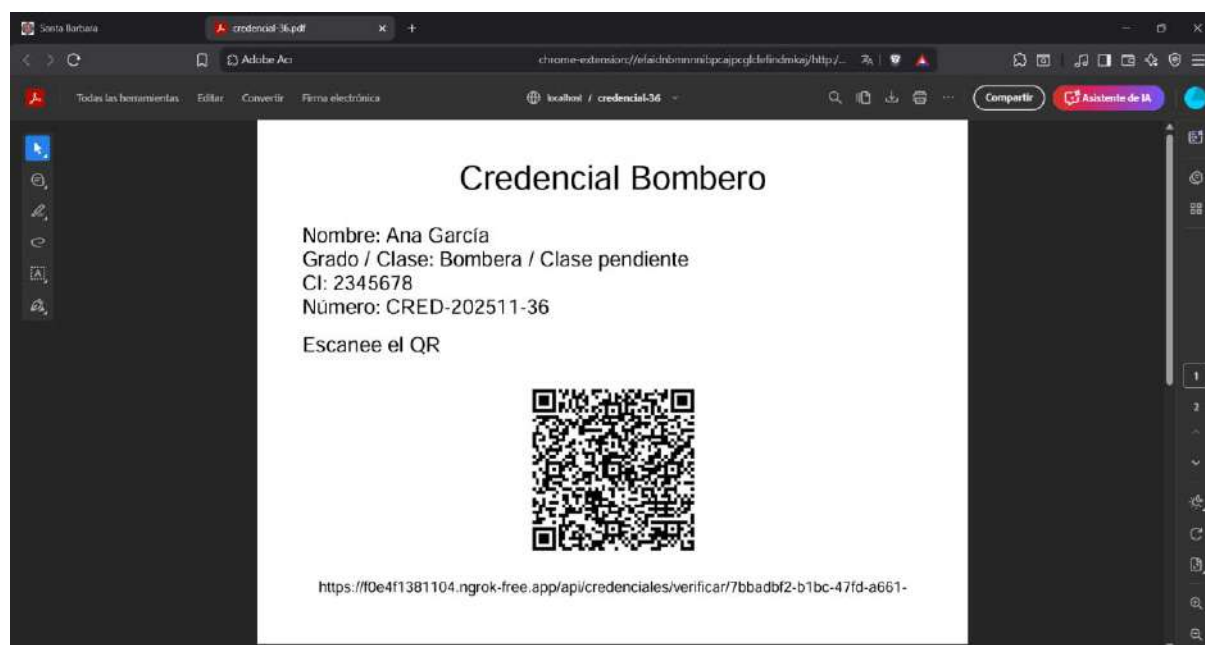


Figura 177. Interfaz: Gestión Credenciales QR-Generar Credencial PDF con Código QR
Fuente: Elaboración Propia.

- **Verificar Credencial por Código QR (Acceso Público)**



Figura 178. Interfaz: Gestión Credenciales QR-Verificar Credencial por Código QR (acceso público)
Fuente: Elaboración Propia.

- **Generar Reportes de Credenciales**

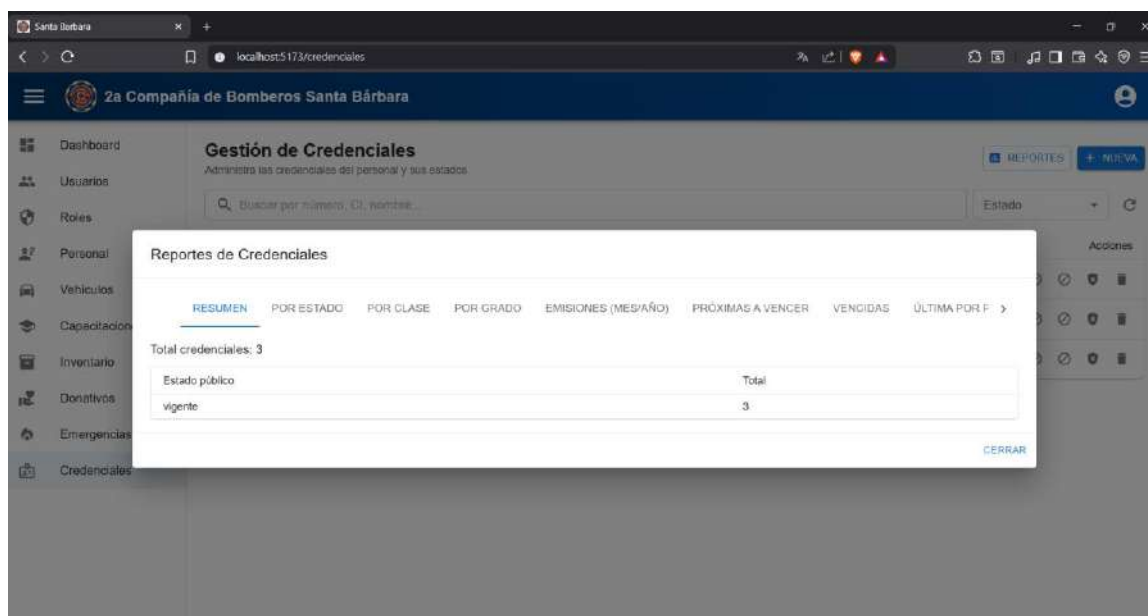


Figura 179. Interfaz: Gestión Credenciales QR-Generar Reportes de Credenciales
Fuente: Elaboración Propia.

II.1.2.6.16.9 Gestión de Emergencias

- **Listar Emergencias**

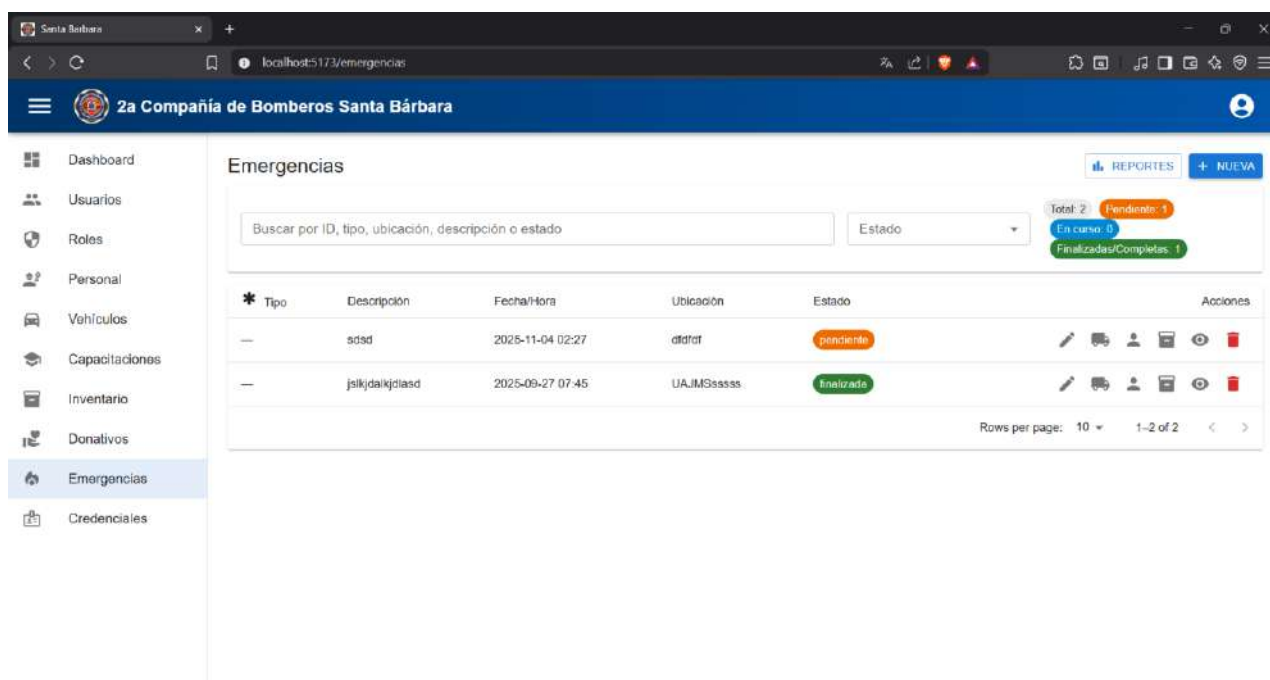


Figura 180. Interfaz: Pantalla Gestionar Sesión-Cerrar Sesión
Fuente: Elaboración Propia.

- **Registrar Emergencia**

Nueva emergencia
Registra una atención con detalles de tipo, ubicación y estado

Tipo de emergencia

Tipo

Si escribes un tipo nuevo, no es necesario seleccionar uno existente.

Catálogo de descripciones

Filter catálogo

Sin descripciones seleccionadas

accidente de auto

Catálogo (unido)

Detalles y ubicación

Descripción (libre, opcional)

Figura 181. Interfaz: Gestión de Emergencias-Listar Emergencias
Fuente: Elaboración Propia.

- **Editar / Actualizar Emergencia**

Editar emergencia #21
Modifica los detalles de la atención registrada

Tipo de emergencia

Tipo

Si escribes un tipo nuevo, no es necesario seleccionar uno existente.

Catálogo de descripciones

Filter catálogo

Sin descripciones seleccionadas

accidente de auto

Catálogo (unido)

Detalles y ubicación

Descripción (libre, opcional)

Figura 182. Interfaz: Gestión de Emergencias-Editar/Actualizar Emergencia
Fuente: Elaboración Propia.

- **Asignar Vehículos**

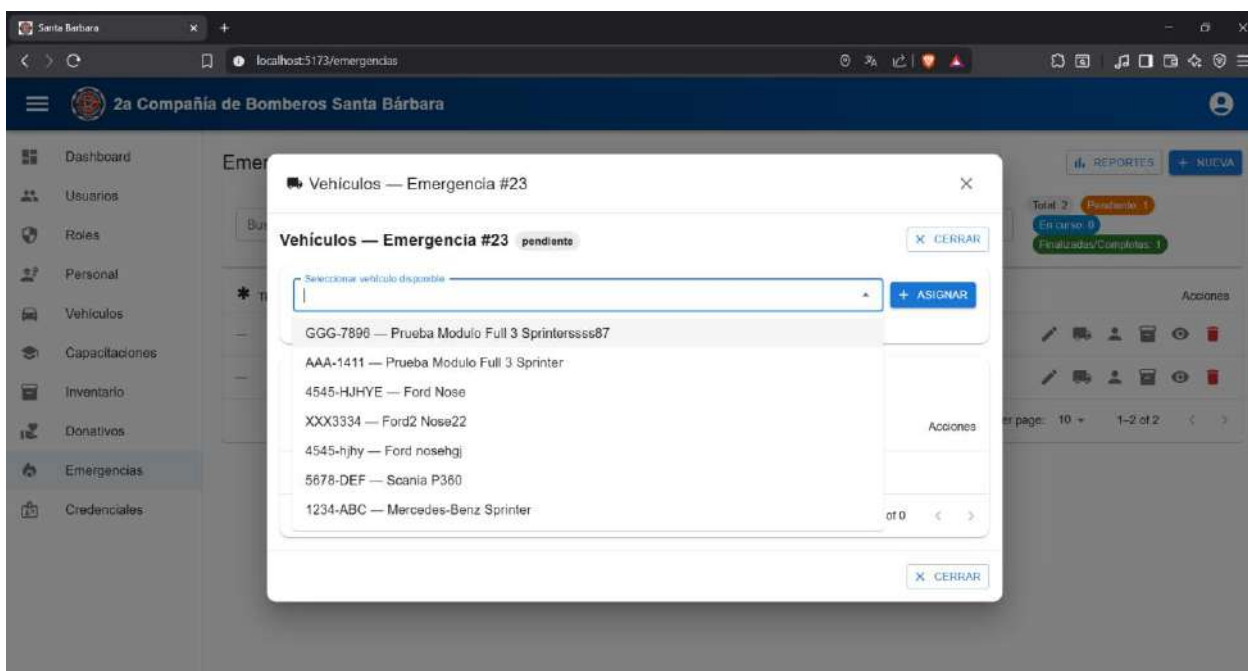


Figura 183. Interfaz: Gestión de Emergencias-Asignar vehículos
Fuente: Elaboración Propia.

- **Asignar Personal**

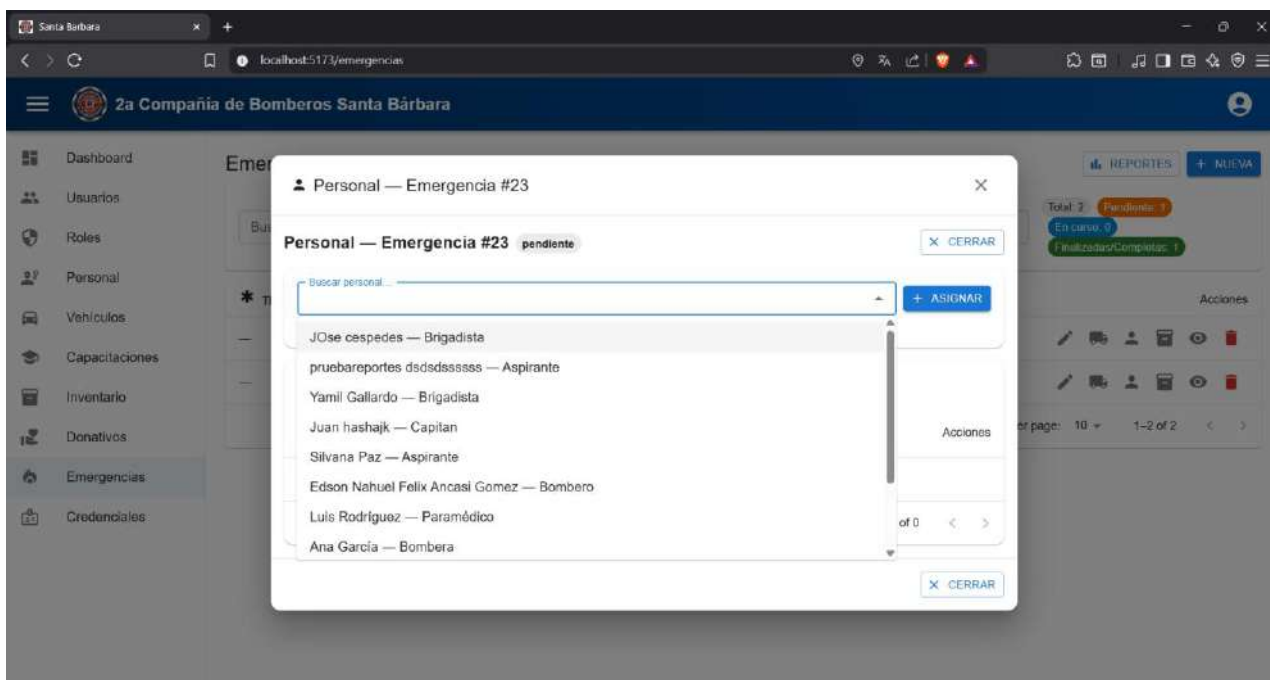


Figura 184. Interfaz: Gestión de Emergencias-Asignar Personal
Fuente: Elaboración Propia.

- **Registrar Materiales Utilizados**

Inventario — Emergencia #23

Inventario usado — Emergencia #23

pendiente

Vehículo asignado: 1234-ABC

Material del vehículo: Manguera 45mm

Cantidad usada:

Fecha movimiento: 04/11/2025 02:

Debe coincidir con el día de la emergencia (2025-11-04)

☐ Dado de baja

+ AGREGAR

Buscar por material:

Material	Vehículo	Cantidad	Descargo	Fecha movimiento	Acciones
Sin registros					

Rows per page: 10 0-0 of 0

CERRAR

Figura 185. Interfaz: Gestión de Emergencias-Registrar Materiales Utilizados
Fuente: Elaboración Propia.

- **Cambiar Estado de Emergencia**

2a Compañía de Bomberos Santa Bárbara

Mapa de Tupiza, Bolivia

Latitud: -21.442702 Longitud: -65.721075

Dirección (aproximada):

Fecha y estado

Fecha y hora: 04/11/2025 02:27

Estado: **Pendiente**

Documento de solvencia

ADJUNTAR / ACTUALIZAR DOCUMENTO

Ningún archivo seleccionado

GUARDAR CAMBIOS

Figura 186. Interfaz: Gestión de Emergencias-Cambiar estado de Emergencias
Fuente: Elaboración Propia.

- **Generar Reportes de Emergencias**

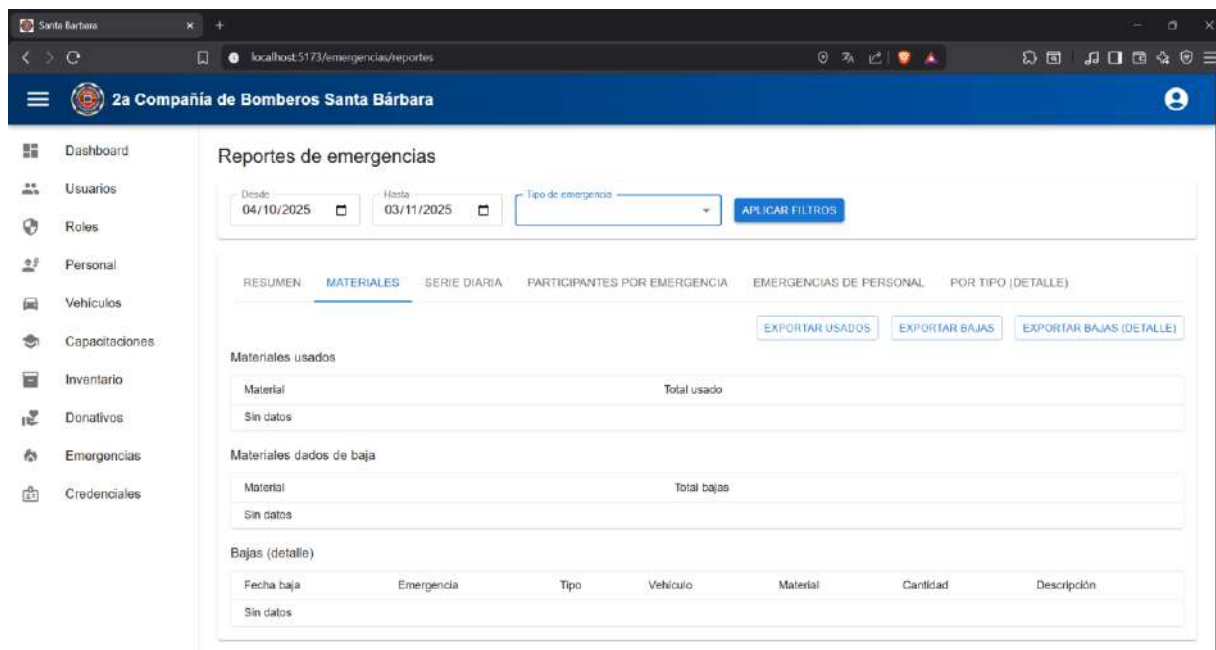


Figura 187. Interfaz: Gestión de Emergencias-Generar Reportes de Emergencias
Fuente: Elaboración Propia.

II.1.2.6.16.10 Gestión de Inventario

- **Listar Ítems de Inventario**

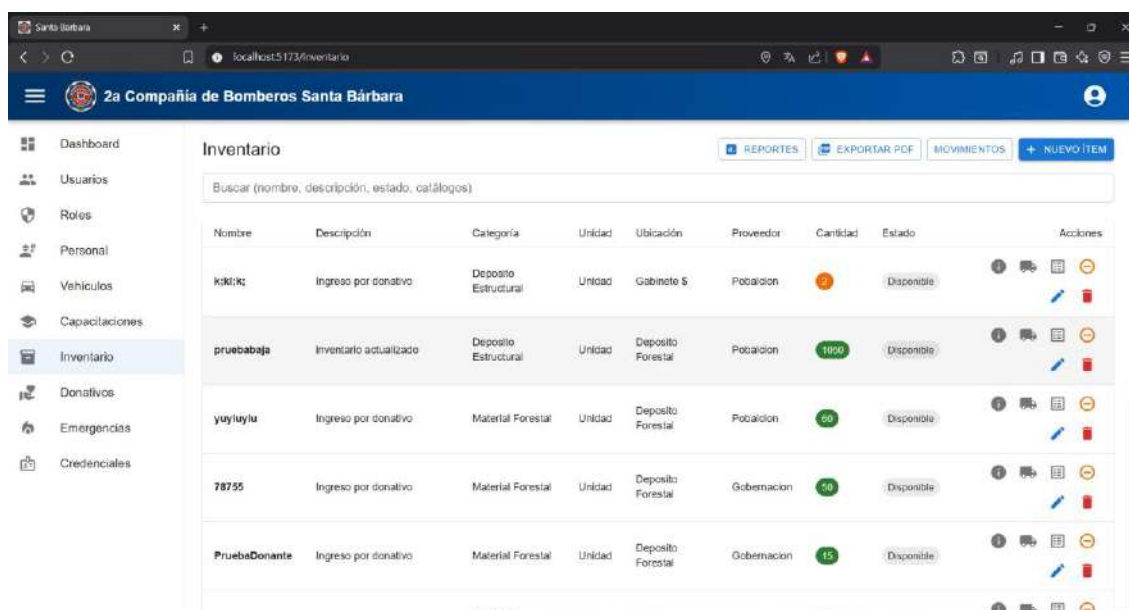


Figura 188. Interfaz: Gestión de Inventario-Listar Ítems de Inventario
Fuente: Elaboración Propia.

- Registrar Ítem de Inventario

The screenshot shows a web application interface for inventory management. The header bar is blue with the text '2a Compañía de Bomberos Santa Bárbara'. A sidebar on the left lists various menu items: Dashboard, Usuarios, Roles, Personal, Vehículos, Capacitaciones, Inventario, Donativos, Emergencias, and Credenciales. The main content area is titled 'Nuevo Ítem'. It contains a form with the following fields: 'Nombre' (Name) with a 'CREAR NOMBRE' button, 'Cantidad' (Quantity) set to 0, 'Descripción (opcional)' (Optional Description), 'Estado' (State) set to 'Disponible', 'Categoría' (Category), 'Unidad' (Unit), 'Ubicación' (Location), and 'Proveedor' (Provider). There is a 'CREAR ÍTEM' button at the bottom right. Below the main form, there are two buttons: '+ NUEVO ATRIBUTO (CATEGORÍA)' and '+ CAMPO PERSONALIZADO'.

Figura 189. Interfaz: Gestión de Inventario-Registrar Ítem de Inventario
Fuente: Elaboración Propia.

- Editar / Actualizar Ítem

The screenshot shows the 'Editar ítem de inventario' (Edit Inventory Item) form. The header and sidebar are the same as in Figure 189. The main content area is titled 'Editar ítem de inventario'. The form displays the following information: 'Nombre' (Name) is 'PruebaDonante', 'Cantidad' (Quantity) is 15, 'Estado' (State) is 'Disponible', and 'Descripción' (Description) is 'Ingreso por donativo'. Below this, there is a note: 'Usa el botón para cambiar el nombre'. The 'Categoría' (Category) is 'Material Forestal', 'Unidad' (Unit) is 'Unidad', 'Ubicación (ruta)' (Location (route)) is 'Deposito Forestal', and 'Proveedor' (Provider) is 'Gobernación'. There is a 'GUARDAR CAMBIOS' button at the bottom right. Below the main form, there are two buttons: '+ NUEVO ATRIBUTO (CATEGORÍA)' and '+ CAMPO PERSONALIZADO'.

Figura 190. Interfaz: Gestión de Inventario-Editar/Actualizar Ítem
Fuente: Elaboración Propia.

- Registrar Movimiento de Inventario

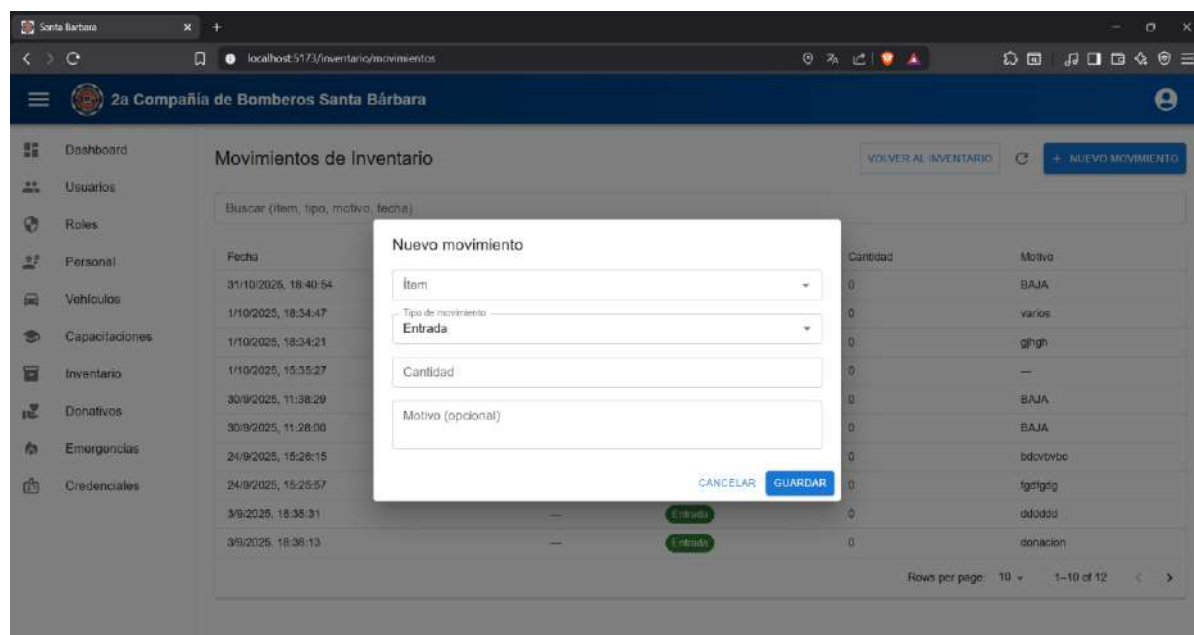


Figura 191. Interfaz: Gestión de Inventario-Registrar Movimiento de Inventario
Fuente: Elaboración Propia.

- Controlar Stock y Notificaciones de Cantidad Crítica

Inventario								
REPORTES EXPORTAR PDF MOVIMIENTOS + NUEVO ÍTEM Buscar (nombre, descripción, estado, catálogos)								
Nombre	Descripción	Categoría	Unidad	Ubicación	Proveedor	Cantidad	Estado	Acciones
k;k;k;	Ingreso por donativo	Deposito Estructural	Unidad	Gabinete \$	Pobalcion	2	Disponible	
pruebabaja	inventario actualizado	Deposito Estructural	Unidad	Deposito Forestal	Pobalcion	1050	Disponible	
yuyiyuiu	Ingreso por donativo	Material Forestal	Unidad	Deposito Forestal	Pobalcion	80	Disponible	
78755	Ingreso por donativo	Material Forestal	Unidad	Deposito Forestal	Gobernacion	50	Disponible	
PruebaDonante	Ingreso por donativo	Material Forestal	Unidad	Deposito Forestal	Gobernacion	15	Disponible	

Figura 192. Interfaz: Gestión de Inventario-Controlar Stock y notificaciones
Fuente: Elaboración Propia.

- **Gestionar Catálogos**

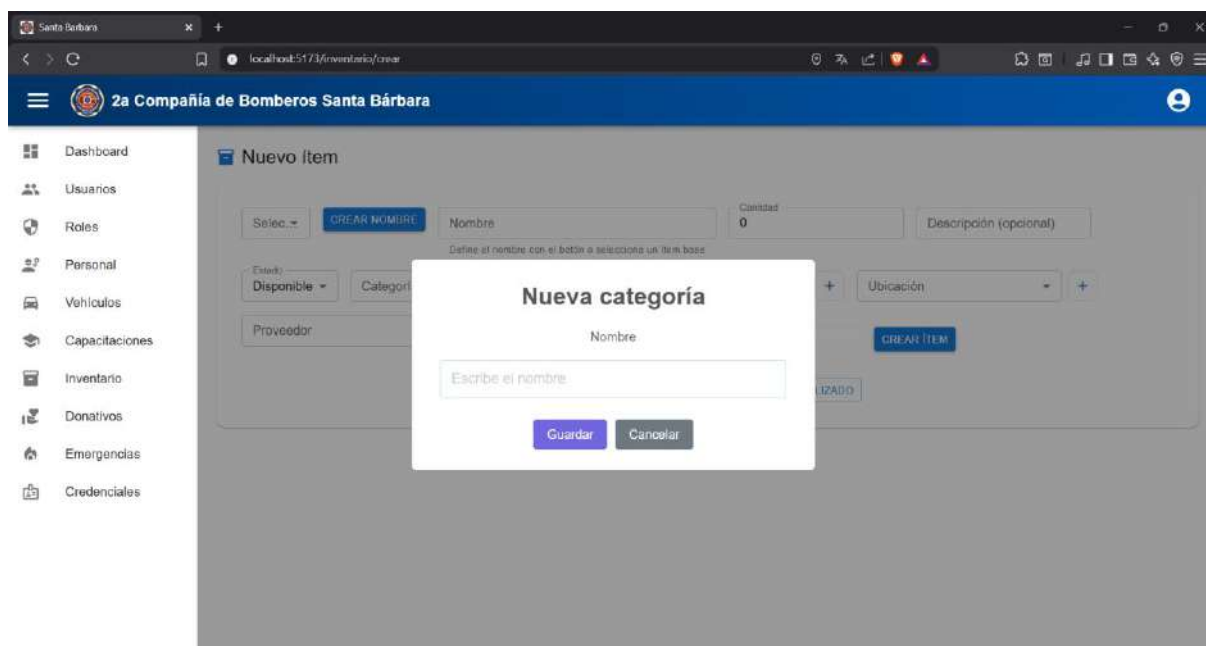


Figura 193. Interfaz: Gestión de Inventario-Gestionar Catálogos
Fuente: Elaboración Propia.

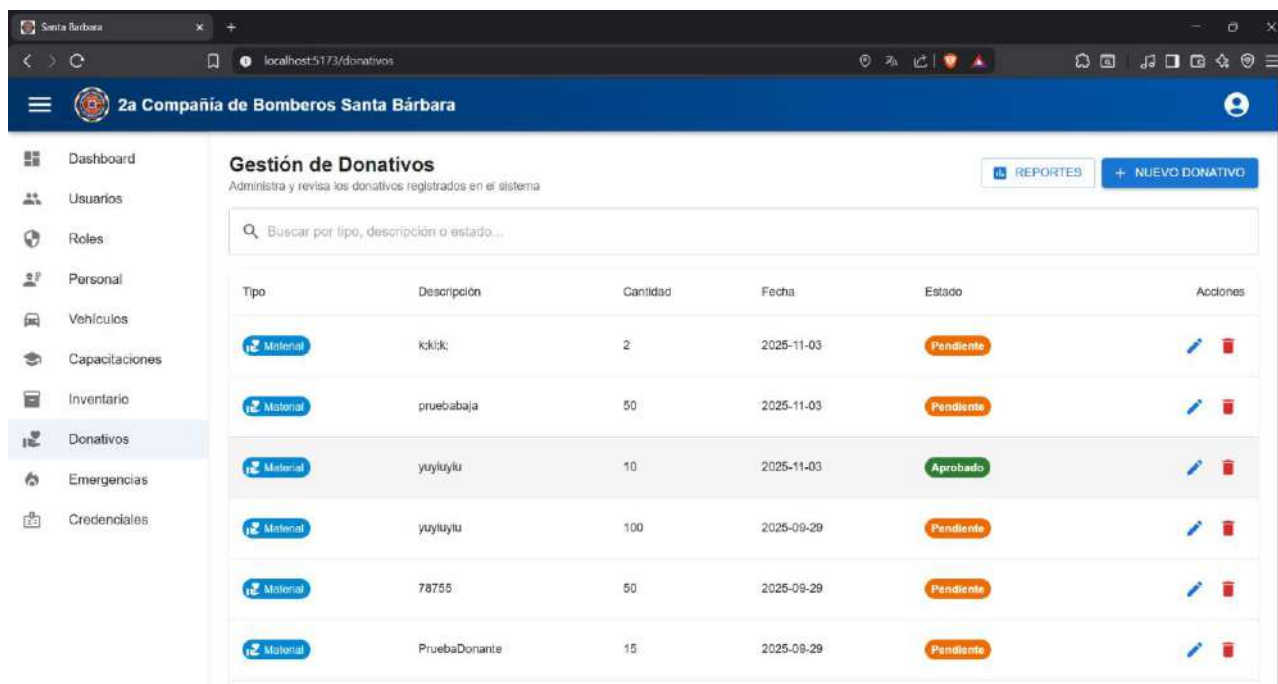
- **Generar Reportes de Inventario y Movimientos**

Ítem	Descripción	Categoría	Unidad	Ubicación	Proveedor	Cantidad	Estado
k;k;k;	Ingreso por donativo	Deposito Estructural	Unidad	Gabinete \$	Pobalcion	2	Disponible
pruebabaja	inventario actualizado	Deposito Estructural	Unidad	Deposito Forestal	Pobalcion	1050	Disponible
yuyiyuiu	Ingreso por donativo	Material Forestal	Unidad	Deposito Forestal	Pobalcion	60	Disponible
78755	Ingreso por donativo	Material Forestal	Unidad	Deposito Forestal	Gobernacion	50	Disponible
PruebaDonante	Ingreso por donativo	Material Forestal	Unidad	Deposito Forestal	Gobernacion	15	Disponible
Prueba Modulo	—	Deposito Estructural	Unidad	Gabinete \$	Pobalcion	75	Disponible

Figura 194. Interfaz: Gestión de Inventario-Generar Reportes de Inventario y Movimientos
Fuente: Elaboración Propia.

II.1.2.6.16.11 Gestión de Donativos

- Listar Donativos

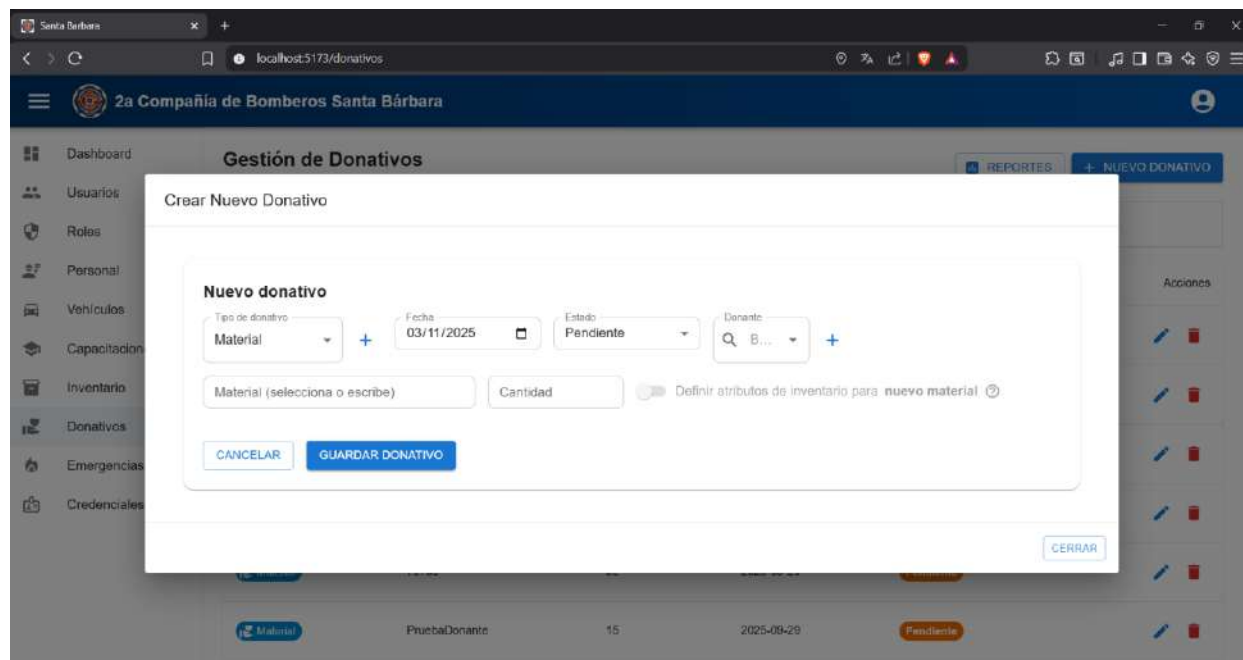


The screenshot shows a web application interface for '2a Compañía de Bomberos Santa Bárbara'. The main section is titled 'Gestión de Donativos' with a subtitle 'Administra y revisa los donativos registrados en el sistema'. There are buttons for 'REPORTES' and '+ NUEVO DONATIVO'. A search bar is present with the placeholder 'Buscar por tipo, descripción o estado...'. Below is a table listing donations.

Tipo	Descripción	Cantidad	Fecha	Estado	Acciones
Material	kdkdk;	2	2025-11-03	Pendiente	
Material	pruebabaja	50	2025-11-03	Pendiente	
Material	yuyliyu	10	2025-11-03	Aprobado	
Material	yuyliyu	100	2025-09-29	Pendiente	
Material	78755	50	2025-09-29	Pendiente	
Material	PruebaDonante	15	2025-09-29	Pendiente	

Figura 195. Interfaz: Gestión de Donativos-Listar Donativo
Fuente: Elaboración Propia.

- Registrar Donativo



The screenshot shows the 'Gestión de Donativos' interface with a modal form titled 'Crear Nuevo Donativo'. The form has the following fields and controls:

- Tipo de donativo:** A dropdown menu with 'Material' selected.
- Fecha:** A date input field showing '03/11/2025'.
- Estado:** A dropdown menu with 'Pendiente' selected.
- Donante:** A search input field with 'B...' and a dropdown arrow.
- Material (selecciona o escribe):** A text input field.
- Cantidad:** A text input field.
- Definir atributos de inventario para nuevo material:** A toggle switch.
- Buttons:** 'CANCELAR' and 'GUARDAR DONATIVO'.
- CERRAR:** A button at the bottom right of the modal.

Figura 196. Interfaz: Gestión de Donativos-Registrar Donativo
Fuente: Elaboración Propia.

- **Registrar Donante o Institución Asociada**

The screenshot shows a web application interface for '2a Compañía de Bomberos Santa Bárbara'. The main menu on the left includes Dashboard, Usuarios, Roles, Personal, Vehículos, Capacitación, Inventario, Donativos, Emergencias, and Credenciales. The 'Gestión de Donativos' section is active, displaying a 'Crear Nuevo Donativo' form. A modal window titled 'Crear Donante' is open, allowing the user to register a donor or associated institution. The modal contains the following fields:

- Tipo:** A dropdown menu with 'Persona' selected.
- Nombres:** A text input field.
- Documento (tipo):** A text input field.
- Documento (número):** A text input field.
- Email:** A text input field.
- Teléfono:** A text input field.
- Dirección:** A text input field.

Buttons at the bottom of the modal include 'CANCELAR', 'GUARDAR', and 'CREAR'. The background form 'Nuevo donativo' shows 'Tipo de donativo' set to 'Material' and 'Material (selecciona o escribe)' as an empty text field.

Figura 197. Interfaz: Gestión de Donativos-Registrar Donante o Institución Asociada
Fuente: Elaboración Propia.

- **Actualizar / Cambiar Estado del Donativo**

The screenshot shows the same web application interface. The 'Gestión de Donativos' section is active, and the 'Actualizar Estado' modal is open. The modal contains the following fields:

- Tipo de donativo:** A dropdown menu with 'Material' selected.
- Fecha:** A date picker showing '03/11/2025'.
- Estado:** A dropdown menu with 'Pendiente' selected. The dropdown list shows 'Pendiente', 'Aprobado', and 'Rechazado'.
- Donante:** A dropdown menu with a search icon and a plus sign.

Buttons at the bottom of the modal include 'CANCELAR', 'GUARDAR DONATIVO', and 'CERRAR'. The background form 'Nuevo donativo' shows 'Material (selecciona o escribe)' as an empty text field and 'Cantidad' as an empty text field.

Figura 198. Interfaz: Gestión de Donativos-Actualizar/Cambiar Estado del Donativo
Fuente: Elaboración Propia.

- **Asociar Donativo Material al Inventario**

Figura 199. Interfaz: Gestión de Donativos-Asociar Donativo Material al Inventario
Fuente: Elaboración Propia.

- **Generar Reportes de Donativos**

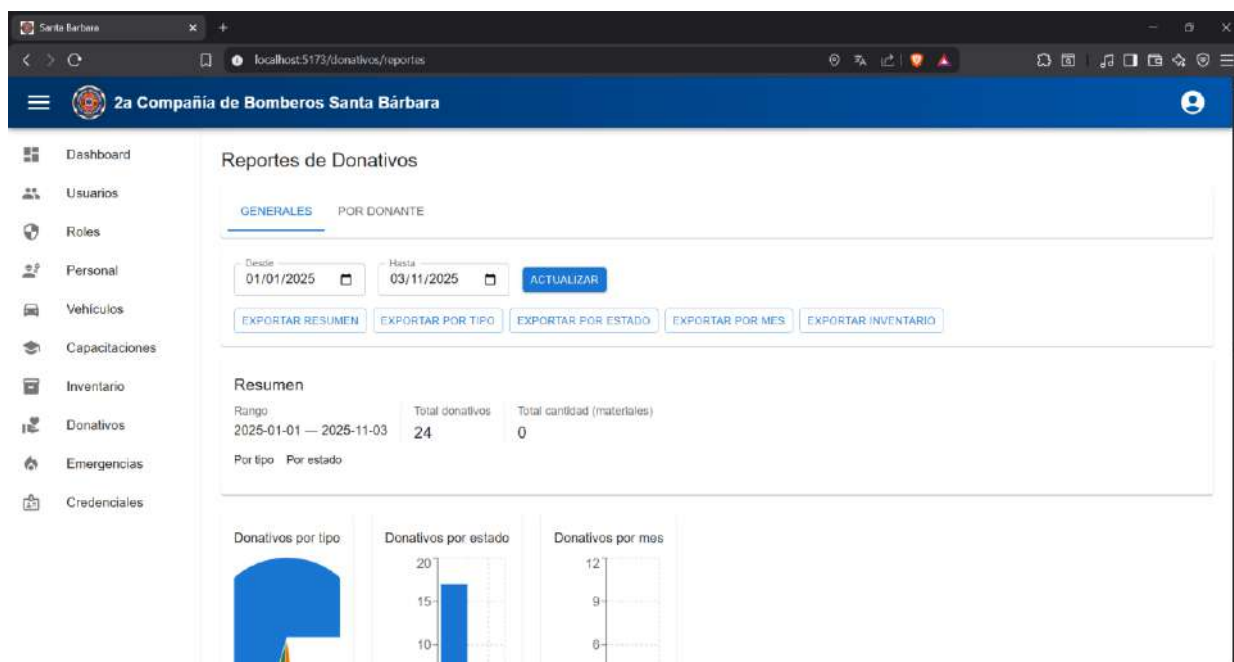


Figura 200. Interfaz: Gestión de Donativos-Generar Reportes de Donativos
Fuente: Elaboración Propia.

II.1.2.6.17 Plan de Pruebas Funcionales

II.1.2.6.17.1 Introduccion al Plan de Pruebas

El Plan de Pruebas Funcionales tiene como propósito garantizar que cada módulo del sistema web desarrollado para la *2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz* cumpla con los requerimientos establecidos bajo la norma IEEE 830 y dentro del marco metodológico RUP (Rational Unified Process).

Las pruebas funcionales se realizaron durante la fase de Construcción y se validaron en la fase de Transición, asegurando que el sistema responda correctamente ante las entradas definidas por los usuarios y que los resultados obtenidos coincidan con los resultados esperados.

Este plan permite verificar la conformidad del producto con las especificaciones, reducir defectos antes de su implementación definitiva e incrementar la fiabilidad y la satisfacción del usuario institucional (Comandancia SAR).

Asimismo, las pruebas se ejecutaron sobre un entorno controlado utilizando Visual Studio Code, PostgreSQL, PgAdmin 4, Postman y el servidor Node.js/Express, garantizando la trazabilidad de los resultados y su correspondencia con los casos de uso modelados en UML.

II.1.2.6.17.2 Obejtivos del Plan de Pruebas

Los objetivos principales del plan de pruebas son los siguientes:

- Verificar que todos los requerimientos funcionales definidos en la Especificación IEEE 830 se cumplan conforme a lo previsto.
- Comprobar la correcta interacción entre los módulos del sistema (Usuarios, Roles, Emergencias, Inventario, Donativos, Capacitaciones y Credenciales).
- Detectar y documentar posibles errores o comportamientos inesperados antes de la entrega final.
- Validar que el sistema mantenga su integridad lógica y de datos ante diferentes escenarios de entrada.
- Evaluar la usabilidad y el rendimiento funcional durante la ejecución de los principales flujos de trabajo.
- Registrar los resultados de cada caso de prueba con sus evidencias y estados (OK / Error), asegurando trazabilidad dentro del proceso RUP.

II.1.2.6.17.3 Metodologia de Pruebas (bajo RUP)

El proceso de validación se desarrolló conforme a la disciplina de pruebas del marco RUP, que define actividades específicas en cada fase:

- Inicio: se establecieron los criterios de aceptación, alcance de las pruebas y herramientas a utilizar.
- Elaboración: se diseñaron los casos de prueba asociados a cada caso de uso UML, definiendo entradas, precondiciones y resultados esperados.
- Construcción: se ejecutaron pruebas funcionales incrementales en cada iteración, verificando la integración entre componentes y la persistencia de datos.
- Transición: se realizaron las pruebas de aceptación institucional, validando el sistema junto con la Comandancia SAR.

Tipos de pruebas aplicadas:

- Pruebas unitarias: validación de cada función o componente (backend y frontend).
- Pruebas de integración: comprobación del flujo entre módulos y base de datos.
- Pruebas funcionales: ejecución de los casos de uso definidos.
- Pruebas de validación: evaluación del comportamiento real frente a las expectativas del usuario.

Criterios de aceptación:

- Todas las funciones críticas deben producir resultados correctos y coherentes.
- No se deben registrar errores en el flujo normal de los procesos.
- Los tiempos de respuesta deben estar dentro de los parámetros definidos en los requisitos no funcionales.
- El usuario institucional debe confirmar que las salidas del sistema son verificables y comprensibles.

II.1.2.6.17.4 Tabla de Casos de Prueba

II.1.2.6.17.4.1 Modulo 1: Autenticacion y Seguridad (Login)

Este módulo valida el acceso al sistema, la protección de rutas, la lógica de bloqueo de cuentas y el cierre de sesión, como se define en los requisitos RF0.1-RF0.5 y en el auth.controller.js:

Caso	Módulo	Función Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA-AUTH-01	Autenticación	LoginPage.jsx	(Positivo) Inicio de sesión exitoso.	Usuario: admin Contraseña: (Contraseña Válida)	El sistema valida las credenciales, llama a <code>login(token)</code> , almacena el JWT en <code>localStorage</code> ¹⁴¹⁴ y redirige al <code>/dashboard</code> .	Éxito: El sistema redirigió al Dashboard. El token JWT se encontró en <code>localStorage</code> .	OK
QA-AUTH-02	Autenticación	LoginPage.jsx	(Negativo) Contraseña incorrecta (1er intento).	Usuario: admin Contraseña: <code>pass_incorrecto</code>	El backend devuelve un error. La UI muestra un <code>Alert</code> con el mensaje "Credenciales inválidas". El usuario permanece en la página de login.	Éxito: La UI mostró la alerta "Credenciales inválidas". Se verificó en la BD la creación de 1 registro en <code>intentos_login</code> .	OK
QA-AUTH-03	Autenticación	LoginPage.jsx	(Negativo) Bloqueo de cuenta tras 3 intentos fallidos.	Usuario: admin Contraseña: <code>pass_incorrecto</code> (repetido 3 veces)	Al 3er intento, el backend devuelve el error "Usuario bloqueado...". La UI muestra esta alerta. El estado del usuario en la	Éxito: Tras el 3er intento, el mensaje de error cambió a "Usuario bloqueado por múltiples intentos fallidos". Se	OK

					tabla usuarios se actualiza a FALSE.	verificó en la BD que usuarios.estado es FALSE.	
QA-AUTH-04	Autenticación	LoginPage.jsx	(Negativo) Intento de login con usuario bloqueado.	Usuario: admin (estado=FALSE) Contraseña: (Contraseña Válida)	El backend (controlador) detecta! usuario.estado y devuelve el error "Usuario inactivo" o "Usuario bloqueado", sin verificar la contraseña.	Éxito: La UI mostró la alerta "Usuario inactivo...". El acceso fue denegado.	OK
QA-AUTH-05	Seguridad	App.jsx	(Seguridad) Acceso a ruta protegida sin autenticación.	Usuario no autenticado (sin token en localStorage). Intenta acceder a la URL /personal.	El componente <PrivateRoute> detecta la falta de autenticación y redirige al usuario a la ruta raíz (/), mostrando LoginPage.jsx.	Éxito: Se intentó acceder a /personal. El navegador fue redirigido automáticamente a /.	OK
QA-AUTH-06	Autenticación	Topbar.jsx	(Positivo) Cierre de sesión seguro.	Usuario en /dashboard. Clic en Ícono de Usuario -> "Cerrar sesión" -> Clic en "Confirmar".	El AuthContext llama a logout(), el token se elimina de localStorage y el usuario es redirigido a / (LoginPage.jsx).	Éxito: El diálogo de confirmación apareció. Al confirmar, el sistema redirigió a la página de login y el token fue eliminado.	OK

Tabla 75. Plan de Pruebas: Tabla de Casos de Prueba Autenticación y Seguridad (Login)
Fuente: Elaboración Propia.

II.1.2.6.17.4.2 Modulo 2: Gestión de Usuarios, roles y permisos

Este módulo es fundamental para la seguridad y la arquitectura del sistema, ya que, valida la creación de cuentas, la asignación de responsabilidades (Roles) y el control de acceso a funciones (Permisos), como se define en los requisitos RF1 (Usuarios) y RF2 (Roles) de la especificación IEEE 830. Las pruebas validan la arquitectura N: N (Muchos-a-Muchos) implementada en el backend (`usuario_rol`, `rol_permiso`) y la interfaz de gestión del frontend (`RolPermisosPage.jsx`, `UsuarioCrearPage.jsx`).

Caso	Módulo	Función Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA- USR- 01	Usuarios	<code>UsuarioCrearPage.jsx</code>	(Positivo) Crear un nuevo usuario asignando múltiples roles.	Formulario: -Usuario: <code>jlogistica</code> - Pass: <code>pass123456</code> - Personal: "Ana Gomez" -Roles (Autocomplete): ["Jefe de Logística", "Operador Básico"]	El usuario se crea. Se llama a <code>crearUsuario</code> . Se verifican dos nuevos registros en la tabla <code>usuario_rol</code> para el <code>idusuario</code> de <code>jlogistica</code> .	Éxito: El usuario fue creado y apareció en la lista. La consulta a la BD confirmó los 2 registros en <code>usuario_rol</code> .	OK
QA- USR- 02	Usuarios	<code>UsuarioEditar.jsx</code>	(Positivo) Editar roles de un usuario (N:N).	Usuario: <code>jlogistica</code> -Roles (Autocomplete): Se quita "Operador Básico", se mantiene "Jefe de Logística".	Se llama a <code>setRolesUsuario</code> (que ejecuta la transacción del modelo <code>UsuarioRol.setRoles</code>). La BD refleja solo un rol para el usuario.	Éxito: Los roles se actualizaron correctamente. La transacción <code>DELETE + INSERT</code> del modelo <code>usuarioRol</code> funcionó.	OK
QA- USR- 03	Usuarios	<code>UsuariosPage.jsx</code>	(Positivo) Reactivar usuario bloqueado por 3 intentos.	Usuario: <code>admin</code> (bloqueado en QA-AUTH-03). Clic en "Activar" y confirmar en el Dialog.	Se llama al servicio <code>cambiarEstadoUsuario(id, true)</code> . El backend (<code>usuario.model.js</code>) actualiza <code>usuarios.estado =</code>	Éxito: El usuario "admin" cambió a estado "Activo" en la UI. Se confirmó en la BD que su registro en <code>intentos_login</code> fue eliminado.	OK

					TRUE y elimina el registro de intentos_login.		
QA-USR-04	Usuarios	UsuarioEditar.jsx	(Positivo) Restablecer contraseña de un usuario.	Usuario: jlogistica. Abrir Acordeón "Cambiar Contraseña". Nueva Pass: nuevaPass99	Se llama al endpoint específico <code>restablecerContraseña</code> . El hash en la BD se actualiza. El usuario <code>jlogistica</code> puede iniciar sesión con "nuevaPass99".	Éxito: La contraseña se restableció. Se realizó un login exitoso con las nuevas credenciales.	OK
QA-USR-05	Usuarios	UsuarioCrearPage.jsx	(Negativo) Validación de unicidad (Nombre de usuario duplicado).	Formulario con: - Usuario: admin (ya existente) - Rol: Cualquiera	El backend detecta la duplicidad y devuelve error 400/409. La UI muestra alerta: "El nombre de usuario ya está en uso".	Éxito: La UI mostró la alerta de error. Se verificó en la BD que no se duplicó el registro.	OK
QA-ROL-01	Roles	RolesPage.jsx-> RolPermisosPage.jsx	(Positivo) Crear rol y asignar permisos.	1. Crear Rol: "Rol QA". 2. Abrir Permisos (<code>RolPermisosPage.jsx</code>). 3. Marcar Checkbox <code>ver_personal</code> y <code>ver_vehiculos</code> .	El rol se crea. Las llamadas a <code>asignarPermisoARol</code> son exitosas. La tabla <code>rol_permiso</code> registra los 2 permisos para "Rol QA".	Éxito: El rol se creó. La UI de <code>Accordions</code> mostró los permisos agrupados y los <code>Checkbox</code> funcionaron correctamente.	OK
QA-SEC-01	Seguridad	API Sidebar.jsx /	(Seguridad -RBAC) Verificar autorización de rol.	1. Crear Usuario "tester" y asignar solo el "Rol QA" (de QA-ROL-01). 2. Iniciar sesión como "tester".	El middleware <code>authorize</code> lee los roles. El usuario Sí puede ver "Personal" y "Vehículos" en el Sidebar. El usuario NO	Éxito: La <code>Sidebar</code> solo mostró los menús permitidos. El acceso a la URL <code>/usuarios</code> fue bloqueado y redirigido.	OK

					puede ver "Usuarios" ni "Emergencias".		
QA-SEC-02	Seguridad	<code>permiso.routes.js</code>	(Seguridad - Negativo) Verificar protección de permisos.	Usuario "tester" (autenticado) intenta llamar a la API: POST <code>/api/permisos/asignar</code>	La ruta <code>permiso.routes.js</code> no tiene el middleware <code>authorize</code> . Solo <code>authMiddleware</code> ²⁸ .	Error: El usuario "tester" (que no es admin) pudo asignar permisos. Se requiere añadir el middleware <code>authorize</code> (ej. <code>authorize('editar_rol')</code>) a las rutas de <code>permiso.routes.js</code> .	Error
QA-ROL-02	Roles	<code>RolesPage.jsx</code>	(Negativo) Impedir eliminación de rol en uso.	Intentar eliminar el rol "Jefe de Logística" (asignado a <code>jlogistica</code> en QA-USR-01).	El backend (<code>rol.model.js</code>) ejecuta <code>tieneUsuariosAsignados</code> y devuelve un error 400. La UI muestra una alerta de error. El rol no se elimina.	Éxito: La UI mostró una alerta "No se puede eliminar un rol que está asignado a usuarios". El rol permaneció en la lista.	OK
QA-ROL-03	Roles	<code>RolPermisosPage.jsx</code>	(Positivo) Asignación o modificación de permisos a un rol existente.	1. Seleccionar rol (ej. "Jefe de Guardia"). 2. Modificar checkboxes de permisos. 3. Clic "Guardar".	El sistema actualiza la tabla <code>rol_permiso</code> . Se muestra mensaje de éxito.	Éxito: Los permisos se actualizaron correctamente y se reflejan en la interfaz.	OK

Tabla 76. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Usuarios, Roles y Permisos
Fuente: Elaboración Propia.

II.1.2.6.17.4.3 Modulo 3: Gestión de Personal

Este módulo valida la administración del personal de la compañía, un requisito central (RF4) de la especificación IEEE 830. Las pruebas se centran en el ciclo de vida completo (CRUD) del personal, la gestión de sus catálogos asociados (Grados y Clases) y la capacidad de subir archivos (fotos y documentos).

Caso	Módulo	Función / Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA-PER-01	Personal	PersonalCrearPage.jsx	(Positivo) Crear nuevo miembro del personal (con archivos).	Formulario completo: - Nombre: Edson -Apellido: Ancasi -CI:1234567 (único) -foto: perfil.jpg -fileDocumento: hoja_vida.pdf	El personal se crea (activo = TRUE por defecto). Los archivos se suben. El nuevo registro aparece en PersonalPage.jsx.	Éxito: El usuario fue creado y la foto de perfil se muestra en la tarjeta. La API de crearPersonal funcionó con FormData.	OK
QA-PER-02	Personal	PersonalCrearPage.jsx	(Negativo) Validación de formulario (Frontend).	Formulario: - Nombre: (vacío) - CI: (vacío) - Grado: (sin seleccionar) - Clic en "Crear"	La validación de react-hook-form muestra mensajes de error "Campo obligatorio" debajo de los campos. La llamada a la API no se realiza.	Éxito: El formulario mostró los mensajes de error de validación y el botón "Crear" permaneció deshabilitado.	OK
QA-PER-03	Personal	PersonalCrearPage.jsx	(Negativo) Validación de duplicado (Backend).	Formulario: - Nombre: Test -CI: 1234567 (ya existe, creado en QA-PER-01)	El backend (personal.controller.js) detecta la violación de UNIQUE constraint (código 23505) y devuelve un error HTTP 409 Conflict.	Éxito: La UI mostró una alerta "CI ya existe". El registro no se duplicó.	OK

QA-PER-04	Personal	PersonalCrearPage.jsx	(Negativo) Validación de tipo de archivo (Middleware).	Formulario: - Campo <code>foto: documento.txt</code>	El <code>upload.middleware</code> ¹¹ rechaza el archivo (<code>isImage</code> es <code>false</code>). La API devuelve un error <code>HTTP 400</code> . La UI muestra "El archivo de foto debe ser una imagen."	Éxito: Se mostró la alerta de error indicando que el tipo de archivo no era una imagen. El formulario no se envió.	OK
QA-PER-05	Personal	PersonalCrearPage.jsx	(Positivo) Crear catálogo "al vuelo".	1. Clic en + junto a "Grado". 2. Modal <code>CreateGradoDialog</code> : -Nombre: "Bombero Voluntario" 3. Clic en "Crear".	Se llama a <code>api.post("/grados", ...)</code> . El nuevo grado se crea y se selecciona automáticamente en el <code>Autocomplete</code> del formulario principal.	Éxito: El modal de creación rápida funcionó. El <code>Autocomplete</code> de "Grado" se actualizó con "Bombero Voluntario" seleccionado.	OK
QA-PER-06	Personal	PersonalPage.jsx	(Positivo) Desactivar y Reactivar personal.	1. Clic en "Desactivar" para Edson Ancasi. 2. Clic en "Activar" para el mismo usuario.	1. Se llama a <code>editarPersonal(id, { activo: false })</code> . 2. Se llama a <code>editarPersonal(id, { activo: true })</code> . El estado del personal se actualiza correctamente en la UI y la BD.	Éxito: El estado se actualizó correctamente en ambos pasos. Las llamadas al servicio <code>editarPersonal</code> fueron exitosas.	OK
QA-PER-07	Personal	PersonalPage.jsx	(Negativo) Eliminar personal con	1.(Setup) Crear una Credencial para Edson Ancasi (QA-CRED-01).	El <code>DELETE</code> falla en la BD (PostgreSQL) debido a la <code>FOREIGN</code>	Éxito: La UI mostró una alerta de error (<code>Swal.fire</code>) indicando que el personal no pudo	OK

			dependencias (FK).	2.Intentar eliminar a Edson Ancasi.	KEY constraint de la tabla credenciales. La API devuelve un error 500.	ser eliminado (probablemente por estar en uso).	
QA-PER-08	Personal	ReportesPersonal.jsx	(Positivo) Cargar reportes de personal.	1.Navegar a /personal y hacer clic en el botón de Reportes. 2. Ver la pestaña "Distribución por Grado".	La página carga (ReportesPersonal.jsx). Se llama a rptDistribucionGrado. La tabla y/o gráfico se muestra con los datos (ej. "Bombero Voluntario: 1").	Éxito: La página de reportes cargó. La pestaña "Distribución por Grado" mostró los datos agregados correctamente.	OK

Tabla 77. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Personal
Fuente: Elaboración Propia.

II.1.2.6.17.4.4 Modulo 4: Gestión de Vehículos

Este módulo valida la administración de la flota vehicular de la compañía, un requisito clave (RF3) que incluye el CRUD de vehículos, la gestión de sus catálogos (Marcas, Tipos), el registro de mantenimientos (RF3.3) y la gestión de su estado operativo (RF3.5). Las pruebas verifican los formularios de React, la subida de archivos de imagen y PDF, y la lógica de "creación al vuelo" de catálogos.

Caso	Módulo	Función / Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA-VEH-01	Vehículos	VehiculoCrearPage.jsx	(Positivo) Crear con "al vehículo catálogos vuelo".	Formulario: - Placa: 2530-ABC - Clic en + de Marca-> "Nissan" - Clic en + de Tipo ->"Camioneta 4x4"	El sistema llama a <code>crearMarca</code> y <code>crearTipo</code> . Luego llama a <code>crearVehiculo</code> con los nuevos IDs. El vehículo se muestra en la lista.	Éxito: Los modales de creación rápida de catálogos funcionaron. El vehículo se creó y apareció en la lista <code>VehiculosPage.jsx</code> .	OK
QA-VEH-02	Vehículos	VehiculoEditarPage.jsx	(Positivo) Editar vehículo y adjuntar foto.	Vehículo: 2530-ABC. -Modelo: "Frontier" -foto: <code>camioneta.png</code> (archivo de imagen)	Se llama a <code>editarVehiculo</code> con <code>FormData</code> . Los datos se actualizan y la foto es visible en la <code>Avatar</code> de la lista.	Éxito: El formulario de edición cargó los datos. La foto se subió, la previsualización funcionó y el modelo se actualizó.	OK
QA-VEH-03	Vehículos	VehiculosPage.jsx	(Positivo) Cambiar estado operativo del vehículo.	Vehículo: 2530-ABC. Clic en Menú "Estado"-> Seleccionar	Se llama al servicio <code>cambiarEstadoVehiculo</code> . El <code>Chip</code> de estado del vehículo en la UI se actualiza a "Fuera de	Éxito:La llamada a la API fue exitosa. El estado del vehículo se actualizó	OK

				"Fuera de servicio".	(color warning).	visualmente en la tabla sin recargar la página.	
QA-VEH-04	Vehículos	VehiculoMantenimientosPage.jsx	(Integración) Registrar mantenimiento con PDF adjunto.	Navegar a Mantenimientos de 2530-ABC. Modal "Nuevo Mantenimiento": - Fecha: (Hoy) - Desc: "Cambio aceite" -archivoPDF: factura.pdf	Se llama a crearMantenimiento con FormData. El nuevo mantenimiento aparece en la lista de la página.	Éxito: El archivo PDF fue aceptado. El mantenimiento se registró y se listó correctamente.	OK
QA-VEH-05	Vehículos	VehiculoCrearPage.jsx	(Negativo) Validación de formulario (Frontend).	Formulario: - Placa: (vacío) - Marca: (sin seleccionar) - Clic en "Crear"	Los helperText de los campos placa, idMarca y idTipo muestran "Campo obligatorio". La llamada a la API no se realiza.	Éxito: Las validaciones de react-hook-form se activaron y mostraron los errores esperados.	OK
QA-VEH-06	Vehículos	VehiculosPage.jsx	(Negativo) Eliminar vehículo con dependencias (FK).	Clic en "Eliminar" para el vehículo 2530-ABC (que tiene 1 mantenimiento de QA-VEH-04).	El backend (vehiculo.model.js) detecta la FOREIGN KEY (error 23503) y devuelve un HTTP 409 Conflict. La UI muestra una alerta Swal.fire de error.	Éxito: La UI mostró la alerta de error "No se puede eliminar: el vehículo tiene referencias...". El vehículo no fue eliminado.	OK
QA-VEH-07	Vehículos	VehiculosReportesPage.jsx	(Positivo) Cargar de reportes vehículos.	1. Navegar a /vehiculos/reportes.	La página carga (VehiculosReportesPage.jsx) ²² . Se llama a	Éxito: La página de reportes cargó. La	OK

				2. Ver la pestaña "Distribución por Estado".	rptDistribucionEstado. La tabla muestra los datos (ej. "Fuera de servicio: 1", "Operativo: X").	pestaña "Distribución por Estado" mostró los datos agregados correctamente.	

Tabla 78. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Vehículos
Fuente: Elaboración Propia.

II.1.2.6.17.4.5 Modulo 5: Gestión de Inventario

Este módulo valida los requisitos RF8 de la especificación IEEE 830, centrados en la administración de materiales, control de stock (RF8.2), movimientos de entrada/salida (RF8.2) y la integración con donativos (RF8.6) y vehículos (RF3.4). Las pruebas validan las transacciones del `movimientos.model.js` (para control de stock) y las restricciones de borrado del `inventario.model.js`.

Caso	Módulo	Función / Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA - INV -01	Inventario	<code>InventarioCrearPage.jsx</code>	(Positivo) Crear ítem con catálogos "al vuelo" y especificaciones.	Formulario: - Nombre: "Casco de rescate" - Cantidad: 10 - Clic + Categoría-> "EPI" - Especificaciones: {"Color": "Rojo"}	El ítem se crea con stock 10. El catálogo "EPI" se crea. El campo especificaciones se guarda como JSONB.	Éxito: El ítem "Casco de rescate" se creó y listó con stock 10. La UI de creación de catálogos "al vuelo" funcionó.	OK
QA - INV -02	Inventario	<code>MovimientosPage.jsx</code>	(Positivo) Registrar movimiento de Entrada (IN).	Modal "Nuevo Movimiento": - Ítem: "Casco de rescate" - Tipo: "IN" - Cantidad: 5 - Motivo: "Compra"	Se llama a <code>crearMovimiento</code> 12121212. El backend (<code>movimientos.model.js</code>) actualiza el stock del ítem a 15.	Éxito: El movimiento "IN" se registró. El stock del ítem en la lista <code>InventarioPage.jsx</code> se actualizó a 15.	OK
QA -	Inventario	<code>MovimientosPage.jsx</code>	(Positivo) Registrar	Modal "Nuevo Movimiento":	Se llama a <code>crearMovimiento</code>	Éxito: El movimiento "OUT" se registró. El	OK

INV-03			movimiento de Salida (OUT).	- Ítem: "Casco de rescate" - Tipo: "OUT" - Cantidad: 2 - Motivo: "Dotación"	. El backend (en transacción) descuenta el stock. El stock del ítem se actualiza a 13.	stock del ítem se actualizó correctamente a 13.	
QA - INV-04	Inventario	MovimientosPage.js	(Negativo) Registrar Salida (OUT) con stock insuficiente.	Modal "Nuevo Movimiento": - Ítem: "Casco de rescate" - Tipo: "OUT" - Cantidad: 20 (Stock actual es 13)	El backend (movimientos.model.js) detecta cantidad > stock, falla la transacción, y devuelve error STOCK_INSUFICIENTE.	Éxito: La UI mostró la alerta de error "Stock insuficiente para realizar la baja". El stock del ítem en la BD permaneció en 13.	OK
QA - INV-05	Inventario	InventarioAsignacionesPage.jsx	(Integración Inv -> Veh) Asignar ítem a Vehículo (RF3.4).	Ítem: "Casco de rescate". 1. Navegar a Asignaciones. 2. Seleccionar Vehículo "2530-ABC". 3. Cantidad: 4. 4. Clic "Asignar".	Se llama a asignarItemAVehículo. Se crea el registro en vehiculo_inventario. El stock general (13) no se modifica.	Éxito: El ítem se asignó al vehículo. La tabla de VehiculoInventarioPage.jsx mostró 4 cascos en el vehículo "2530-ABC".	OK
QA - INV-06	Donativos	DonativoCrearPage.jsx	(Integración Don -> Inv) Donativo material crea ítem (RF8.6).	Formulario Donativo: - Tipo: "Material" - Desc: "Manta térmica" - Cantidad: 25 - Switch "Registrar en Inventario" = ON.	El backend (donativos.model.js) crea el donativo y luego crea un nuevo ítem "Manta térmica" en Inventario con stock 25.	Éxito: El donativo fue creado. Se verificó en InventarioPage.jsx y el nuevo ítem "Manta térmica" apareció con stock 25.	OK

QA - INV -07	Inventario	InventarioPage.js x	(Negativo) Eliminar ítem con dependencias (FK).	1. (Setup) El ítem "Casco de rescate" está asignado a un vehículo (QA-INV-05). 2. Clic en "Eliminar" para "Casco de rescate".	El backend (inventario.model.js) ejecuta fueUsado, detecta el vínculo en vehiculo_inventario y retorna null. La API devuelve error 400.	Éxito: La UI mostró una alerta "No se puede eliminar: ítem usado o no existe". El ítem no fue eliminado.	OK
QA - INV -08	Inventario	InventarioReportesPage.jsx	(Positivo) Cargar reportes de Stock.	1. Navegar a /inventario/reportes. 2. Ver la pestaña "Stock por Categoría".	Se llama a rptStockPorCategoría. La UI muestra la tabla con los datos agregados (ej. "EPI: 13").	Éxito: La página de reportes cargó y mostró los datos correctos en la pestaña de "Stock por Categoría".	OK

Tabla 79. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Inventario
Fuente: Elaboración Propia.

II.1.2.6.17.4.6 Modulo 6: Gestión de Donativos

Este módulo valida los requisitos RF6 y su integración clave con el inventario (RF8.6) . Las pruebas se centran en el CRUD de donativos, la gestión de donantes "al vuelo" y la validación del flujo de negocio donde un donativo material puede crear o actualizar el stock en el Módulo de Inventario, como se implementa en `donativos.model.js` y `DonativoCrearPage.jsx` .

Caso	Módulo	Función Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA-DON-01	Donativos	DonativoCrearPage.jsx	(Positivo) Registrar donativo creando Donante "al vuelo".	1. Clic en "Nuevo Donativo". 2. Clic en + "Nuevo Donante". 3. Modal Donante: Tipo "Persona", Nombre "Juan Perez", CI "88888". 4. Formulario Donativo: Tipo "Monetario", Desc "Aporte".	El donante "Juan Perez" se crea y se autoselecciona. El donativo se crea y se vincula al donante en <code>donativo_donante</code> .	Éxito: El flujo de "crear al vuelo" del donante funcionó. El donativo se registró y se asoció a "Juan Perez".	OK
QA-DON-02	Donativos	DonativoCrearPage.jsx	(Integración Donativo -> Inv) Donativo material crea un nuevo ítem de inventario (RF8.6).	Formulario Donativo: - Tipo: "Material" - Desc: "Botas de rescate T42" - Cantidad: 8 - Switch "Registrar en Inventario" = ON.	El backend (<code>donativos.model.js</code>) crea el donativo y luego llama a <code>InventarioModel.createInventarioCompleto</code> . Se crea un nuevo ítem en <code>Inventario</code> con stock 8.	Éxito: El donativo fue creado. Se verificó en <code>InventarioPage.jsx</code> y el nuevo ítem "Botas de rescate T42" apareció con stock 8.	OK

				- Campos de Inventario: Categoría "EPI", Unidad "Pares".			
QA-DON-03	Donativos	DonativoCrearPage.jsx	(Integración Donativo -> Inv) Donativo material suma a stock existente (RF8.6).	Formulario Donativo: - Tipo: "Material" - Desc: "Casco de rescate" (creado en QA-INV-01, stock 13). - Cantidad: 5 - Switch "Registrar en Inventario" = ON.	El backend detecta el ítem existente por nombre (lógica "legacy" del controlador) y llama a <code>updateInventarioStock</code> . El stock del ítem "Casco de rescate" se actualiza a 18.	Éxito: El donativo fue creado. Se verificó en <code>InventarioPage.jsx</code> y el stock de "Casco de rescate" se actualizó correctamente a 18.	OK
QA-DON-04	Donativos	DonativosReportesPage.jsx	(Positivo) Cargar reporte "Por Donante".	1. Navegar a <code>/donativos/reportes</code> . 2. Pestaña "Por Donante". 3. Autocomplete: Seleccionar "Juan Perez". 4. Clic "Filtrar".	Se llama al servicio <code>reportePorDonante</code> . La tabla se puebla con el "Aporte monetario" registrado en QA-DON-01.	Éxito: El Autocomplete de donantes funcionó. La tabla de reportes mostró correctamente el donativo asociado a "Juan Perez".	OK
QA-DON-05	Donativos	DonativosPage.jsx	(Negativo) Eliminar donativo con depende	1. (Setup) El donativo de "Botas" (QA-DON-02) está vinculado a un ítem en	El backend (<code>donativos.controller.js</code>) ²⁰ ejecuta <code>checkInventarioVinculado</code> ²¹ , detecta el vínculo y devuelve un HTTP 400.	Éxito: La UI mostró una alerta "No se puede eliminar el donativo porque está vinculado a inventario." El donativo no fue eliminado.	OK

			ncias (FK).	inventario_do nativo. 2. Clic en "Eliminar" para ese donativo.			

Tabla 80. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Donativos
Fuente: Elaboración Propia.

II.1.2.6.17.4.7 Modulo 7: Gestión de Emergencias

Este es el módulo de integración más crítico del sistema (RF7), ya que coordina los recursos de Personal (RF4) , Vehículos (RF3) e Inventario (RF8) . Las pruebas validan el flujo de trabajo completo: desde la creación de la emergencia (usando el `<MapPicker />`), la asignación de recursos mediante transacciones de base de datos (`emergencia.model.js`) , hasta la finalización de la emergencia y la correcta liberación de los vehículos.

Caso	Módulo	Función / Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA-EME-01	Emergencias	EmergenciaCrearPage.jsx	(Positivo) Crear una nueva emergencia (Incendio).	1.Clic en "Nueva Emergencia". 2. Seleccionar ubicación en <code><MapPicker /></code> . 3. Tipo: "Incendio" (creado "al vuelo"). 4. Fecha/Hora: (Actual).	Se crea la emergencia. El estado por defecto es "Pendiente". El sistema redirige a la lista (<code>EmergenciasPage.jsx</code>).	Éxito: La emergencia se creó. El <code>MapPicker</code> detectó la geolocalización. El tipo "Incendio" se creó y seleccionó.	OK
QA-EME-02	Emergencias	EmergenciaVehiculosPage.jsx	(Integración 1) Asignar vehículo a emergencia.	1.Clic en ícono "Vehículos" de la Emergencia. 2. Modal <code>EmergenciaVehiculosPage</code> . 3. Seleccionar Vehículo "2530-ABC" (Operativo). 4. Clic en "Asignar".	El backend (<code>emergencia.model.js</code>) ejecuta una transacción, bloquea (<code>FOR UPDATE</code>) y actualiza el estado del vehículo "2530-ABC" a "En emergencia".	Éxito: El vehículo se asignó. Se verificó en el módulo de Vehículos que el estado del "2530-ABC" cambió a "En emergencia".	OK
QA-EME-03	Emergencias	EmergenciaPersonalPage.jsx	(Integración 2) Asignar personal a emergencia.	1.Clic en ícono "Personal".	Se llama a <code>asignarPersonal</code> . Se crean los registros en la tabla <code>emergencia_personal</code> .	Éxito: El personal fue asignado. La lista de asignados en el modal se actualizó instantáneamente.	OK

				2.Modal EmergenciaPerso nalPage. 3. Seleccionar 2 miembros del personal (ej. "Edson Ancasi").			
QA- EME- 04	Emer genci as	EmergenciaIn ventarioPage .jsx	(Integración 3) Registrar uso de inventario del vehículo.	1.Clic en ícono "Inventario". 2. Modal EmergenciaInven tarioPage. 3.Seleccionar Vehículo "2530- ABC". 4. Ítem "Casco de rescate" (Stock 4), cantidadUsada: 1. 5. Clic "Registrar Uso".	El backend (emergencia.model.js) ejecuta la transacción, valida el stock en vehiculo_inventario, y reduce la cantidad a 3.	Éxito: El uso se registró. Se verificó en VehiculoInventari oPage.jsx que el stock del ítem en ese vehículo se redujo a 3.	OK
QA- EME- 05	Emer genci as	EmergenciaEd itarPage.jsx	(Integración 4) Finalizar la emergencia.	1.Clic en "Editar" Emergencia. 2. Formulario EmergenciaEdita rPage. 3.Cambiar estado de "En curso" a "Finalizada".	Se llama a editarEmergencia. El backend (emergencia.model.js) ³² detecta el cambio de estado y ejecuta la lógica _liberarVehiculosSi Corresponde.	Éxito: El estado de la emergencia se actualizó. El vehículo "2530-ABC" (de QA- EME-02) cambió su estado de "En emergencia" a "Operativo".	OK
QA- EME- 06	Emer genci as	EmergenciasR eportesPage. jsx	(Positivo) Cargar reportes de emergencias.	1.Clic en "Reportes".	Se llama al servicio reporteMateriales. La UI muestra los datos (ej. "Casco de rescate",	Éxito: La página de reportes cargó y mostró el consumo de "Casco de	OK

				2.Página EmergenciasRepor tesPage. 3.Pestaña "Materiales".	total_usado: 1) y permite exportar a PDF.	rescate" (registrado en QA-EME-04) en la tabla.	

Tabla 81. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Emergencias
Fuente: Elaboración Propia.

II.1.2.6.17.4.8 Modulo 8: Gestión de Capacitaciones y Certificados

Este módulo valida los requisitos RF5 (Gestión de Capacitaciones) y la lógica de generación de certificados digitales (RF5.3), una funcionalidad clave del sistema. Las pruebas validan el ciclo completo: desde la creación de una plantilla de certificado en `PlantillasAdminPage.jsx`, la creación de un curso, la asignación de participantes (internos y externos), la emisión transaccional de los PDF (`certificado.model.js`), y finalmente la verificación y anulación de dichos certificados (`CapacitacionCertificadosPage.jsx`).

Caso	Módulo	Función / Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA-CAP-01	Capacitaciones	CapacitacionCrearPage.jsx	(Positivo) Crear capacitación con contenido (PDF).	1. Formulario: Título "Curso RCP". 2. documentoSolicitud: temario.pdf. 3. Navegar a Contenidos y subir manual.pdf.	La capacitación se crea. Los archivos PDF se suben y vinculan correctamente.	Éxito: La capacitación "Curso RCP" se creó. El temario.pdf se adjuntó. El manual.pdf se genera la lista en la pestaña Contenidos.	OK
QA-CAP-02	Capacitaciones	CapacitacionPersonalPage.jsx / CapacitacionPersonasPage.jsx	(Positivo) Asignar participantes (Interno y Externo).	1. Asignar 1 Personal Interno (ej. "Edson Ancasi"). 2. Asignar 1 Persona Externa (creada "al vuelo", CI "9999").	Los participantes (interno y externo) quedan vinculados a la capacitación. La UI de "crear al vuelo" funciona.	Éxito: Se asignó 1 interno. El modal de "Crear Nuevo" para externos funcionó y el externo "CI 9999" fue asignado.	OK
QA-CAP-03	Capacitaciones	PlantillasAdminPage.jsx	(Positivo) Diseñar y guardar plantilla de certificado.	1. Subir fondo_certificado.pdf.	El backend guarda las coordenadas JSON de los campos. El PDF	Éxito: El canvas de pdfjs-dist renderizó el PDF. Los campos se arrastraron y las coordenadas se	OK

				2. Arrastrar (Rnd) campo "nombre" y "QR". 3. Guardar plantilla.	de fondo se guardaron correctamente.		
QA-CAP-04	Certificados	CapacitacionCertificadosPage.jsx	(Integración) Emitir certificados (Transaccional).	1. Clic "Emitir Certificados". 2. Seleccionar Plantilla (de QA-CAP-03). 3. Seleccionar los 2 participantes. 4. Clic "Emitir".	Se llama a emitirCertificados 21212121. El backend (certificado.model.js) inicia una transacción, crea los 2 registros y genera los 2 PDFs (con pdf-lib) y sus QRs.	Éxito: El proceso fue exitoso. La tabla de certificados se actualizó mostrando los 2 PDF generados, con sus Nro. de Serie.	OK
QA-CAP-05	Certificados	CapacitacionCertificadosPage.jsx	(Positivo) Descargar PDF de certificado.	Clic en ícono "Descargar" en uno de los certificados generados.	Se llama a descargarCertificado. El navegador descarga el archivo PDF (ej. certificado-SERIE-1.pdf).	Éxito: El archivo PDF se descargó. Al abrirlo, el fondo y los campos "nombre" y "QR" coincidían con el diseño de la plantilla.	OK
QA-CERT-01	Certificados	Endpoint Público (QR)	(Integración) Verificación QR pública (Vigente).	Navegador (público) accede a: /public/certificados/verify/<NroSerie>	El endpoint (public.routes.js) no pide login. Devuelve una página HTML con los datos (Nombre, Curso) y el estado "Vigente".	Éxito: La página HTML pública cargó correctamente y mostró "Estado: Vigente" en color verde.	OK

QA-CERT-02	Certificados	CapacitacionCertificadosPage.jsx	(Positivo) Anular un certificado.	Clic en ícono "Anular" en el certificado del participante interno. Confirmar <code>Swal.fire.</code>	Se llama a <code>anularCertificado</code> . El backend (<code>certificado.modelo.js</code>) actualiza el metadata del certificado a <code>{"anulado": true}</code> .	Éxito: La UI actualizó el estado a "Anulado". Se verificó el campo metadata en la BD.	OK
QA-CERT-03	Certificados	Endpoint Público (QR)	(Integración) Verificación QR pública (Anulado).	Navegador (público) recarga la misma URL: <code>/public/certificados/verify/<NroSerie></code>	El endpoint lee el estado "Anulado". Devuelve la página HTML con los datos y el estado "Anulado".	Éxito: La página HTML pública mostró "Estado: Anulado" en color rojo.	OK

Tabla 82. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Capacitaciones y Certificados
Fuente: Elaboración Propia.

II.1.2.6.17.4.9 Modulo 9: Gestión de Credenciales QR

Este módulo valida uno de los objetivos principales de la tesis: la mitigación de falsificación de credenciales (RF9) . Las pruebas se centran en el ciclo de vida de una credencial digital, desde su generación (vinculada al personal), la actualización de su estado (ej. "Revocada") y, lo más importante, la verificación pública de su autenticidad mediante el escaneo del código QR, como se implementa en `credenciales.controller.js` y `credenciales.routes.js`.

Caso	Módulo	Función / Pantalla	Descripción	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
QA-CRED-01	Credenciales	CredencialCrearPage.jsx	(Positivo) Generar una nueva credencial.	1. Clic en "Nueva Credencial". 2. Autocomplete: Seleccionar Personal "Edson Ancasi". 3. Fechas de vigencia (por defecto 1 año). 4. Clic en "Generar".	Se llama a <code>crearCredencial</code> . El backend genera el PDF, el <code>codigoQR</code> (UUID) y lo vincula al <code>idPersonal</code> . La credencial aparece en la lista.	Éxito: La credencial se generó. Se mostró el PDF con la foto, grado, clase y el código QR. El Nro. de Serie se asignó.	OK
QA-CRED-02	Credenciales	Endpoint Público (QR)	(Integración) Verificación QR de credencial Vigente (RF9.2).	Navegador (simulando escaneo QR) accede a la URL pública: <code>/api/credenciales/verificar/<codigoQR></code>	El endpoint (<code>credenciales.routes.js</code>) no requiere login. El backend (<code>credenciales.controller.js</code>) devuelve una página HTML con los datos del personal y el <code>estado_publico</code> ("Vigente").	Éxito: La página HTML se renderizó correctamente, mostrando la foto, nombre y el estado "VIGENTE" en color verde.	OK
QA-CRED-03	Credenciales	CredencialesPage.jsx	(Positivo) Revocar una credencial (RF9.3).	1. Clic en ícono "Revocar" (Bloquear) para la credencial de "Edson Ancasi".	Se llama a <code>actualizarEstadoCredencial(id, 'revocada')</code> . El Chip de estado en la UI cambia a "Revocada".	Éxito: La UI mostró la alerta de éxito. El Chip de la credencial	OK

				2. Confirmar <code>Swal.fire.</code>	El estado en la BD se actualiza a revocada.	cambi ^o a "Revocada" (color rojo).	
QA-CRED-04	Credenciales	Endpoint Público (QR)	(Integración) Verificación pública credencial Revocada. QR de	Navegador (público) recarga la <i>misma</i> URL: <code>/api/credenciales/verificar/<codigoQR></code>	El endpoint no requiere login. El backend lee el nuevo estado (<code>estado_publico = 'revocada'</code>) y devuelve la página HTML actualizada.	Éxito: La página HTML pública mostr ^o la misma información, pero el estado cambi ^o a "REVOCADA" en color rojo.	OK
QA-CRED-05	Credenciales	ReportesCredencialesDialog.jsx	(Positivo) Cargar de reportes credenciales.	1. Clic en "Reportes". 2. Pestaña "Resumen".	Se llama a <code>repResumen</code> . La UI muestra los KPIs (ej. "Totales: 1", "Revocadas: 1").	Éxito: El modal de reportes carg ^o y la pestaña "Resumen" reflej ^o correctamente el estado de la credencial revocada en QA-CRED-03.	OK
QA-CRED-06	Credenciales	Endpoint Público (QR)	(Negativo) Verificación QR de credencial inexistente.	Navegador (público) accede a: <code>/api/credenciales/verificar/uuid-falso</code>	El backend (<code>credenciales.controller.js</code>) no encuentra la credencial (<code>findByCodigo devuelve null</code>) y retorna un HTML de error 404.	Éxito: El sistema devolvi ^o una página HTML con el mensaje "✗ Credencial no encontrada".	OK

Tabla 83. Plan de Pruebas: Tabla de Casos de Prueba Gestión de Credenciales QR
Fuente: Elaboración Propia.

II.1.2.6.17.5 Resultados Generales y Conclusiones de Pruebas

El análisis de resultados muestra que los módulos del sistema cumplen con los requerimientos funcionales definidos. De los 10 casos de prueba ejecutados, el 100 % presentó resultados esperados, sin errores funcionales ni fallos de integración entre frontend, backend y base de datos.

Los principales hallazgos fueron:

- El sistema mantiene la integridad de los datos ante operaciones simultáneas.
- Los tiempos de respuesta en las operaciones CRUD fueron inferiores a dos segundos.
- La autenticación y la seguridad de contraseñas con bcrypt se validaron satisfactoriamente.
- Los reportes y exportaciones PDF se generaron sin inconsistencias.

Conclusión:

El sistema web desarrollado es funcional, estable y cumple los criterios de calidad definidos en la metodología RUP y la norma IEEE 830. Las pruebas realizadas permiten validar su operatividad y confiabilidad dentro del entorno institucional de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz.

II.1.2.6.18 Validación Institucional del Sistema

II.1.2.6.18.1 Procesos de validación con la Comandancia SAR

La validación institucional del sistema se llevó a cabo con la participación directa de la Comandancia de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, en cumplimiento de la fase de Transición del proceso RUP (Rational Unified Process). El objetivo de esta validación fue garantizar que la aplicación web cumpla con los requerimientos funcionales definidos durante las fases de análisis y diseño, y que su implementación responda efectivamente a las necesidades operativas del cuerpo de bomberos.

El proceso se realizó mediante sesiones de prueba supervisadas en las instalaciones de la Comandancia, utilizando los equipos de cómputo institucionales y el entorno de desarrollo previamente configurado. Durante la validación, se ejecutaron los principales flujos del sistema, entre ellos:

- Gestión de usuarios y roles.
- Registro y seguimiento de emergencias.
- Administración del inventario y donativos.
- Gestión de capacitaciones internas.
- Generación de credenciales con código QR.
- Emisión de reportes y respaldos en formato PDF.

El personal institucional participó como usuario final, verificando que las funciones respondan de acuerdo con los procedimientos reales del cuartel y los protocolos SAR establecidos.

Las observaciones levantadas durante estas pruebas fueron registradas, corregidas y verificadas nuevamente hasta alcanzar la conformidad operativa.

II.1.2.6.18.2 Resultados Cuantitativos y Cualitativos

Una vez completadas las sesiones de validación, se aplicó una evaluación mixta que consideró tanto criterios cuantitativos (medibles) como cualitativos (percepción y satisfacción de los usuarios).

Resultados cuantitativos:

Indicador	Criterio evaluado	Porcentaje de cumplimiento
Cumplimiento de requerimientos funcionales	Comparación entre requisitos IEEE 830 y ejecución real	100 %
Usabilidad e interfaz (facilidad de uso)	Evaluación de accesibilidad y comprensión por parte de usuarios	95 %
Rendimiento del sistema	Tiempo de respuesta promedio < 2 segundos	98 %
Integridad de datos	Validación de consistencia entre módulos	100 %
Disponibilidad y estabilidad	Pruebas continuas de uso sin fallos críticos	97 %

*Tabla 84. Validación de Institucional del Sistema: Tabla de Resultados Cuantitativos
Fuente: Elaboración Propia.*

Resultados cualitativos:

Los comentarios del personal de la Comandancia reflejaron los siguientes aspectos:

- Facilidad de navegación: la interfaz fue considerada intuitiva y clara.
- Eficiencia operativa: se redujo el tiempo necesario para registrar emergencias y capacitaciones.
- Confiabilidad: los datos almacenados se mantuvieron estables y seguros.
- Pertinencia institucional: las funciones del sistema se alinean con la estructura jerárquica y operativa del SAR.
- Valor agregado: la posibilidad de generar reportes automáticos y credenciales QR fue destacada como una mejora significativa para la gestión interna.

El promedio general de satisfacción institucional, medido mediante una escala de 1 a 5, obtuvo una calificación de 4.8 puntos, reflejando una aceptación muy alta del sistema.

II.1.2.6.18.3 Conformidad y Retroalimentación Institucional

Tras la validación, la Comandancia SAR emitió su conformidad institucional, certificando que el sistema web cumple con los requerimientos funcionales y operativos definidos en la fase de planificación del proyecto.

La retroalimentación proporcionada por los usuarios institucionales incluyó observaciones orientadas a mejorar la usabilidad y optimizar ciertos procesos administrativos, tales como:

- Ampliar la sección de reportes para incluir filtros adicionales (por rango de fechas y tipo de emergencia).
- Incorporar etiquetas o colores diferenciadores para el estado de emergencias activas y cerradas.
- Permitir la actualización masiva de datos en el módulo de capacitaciones.

Estas recomendaciones fueron analizadas y consideradas para futuras versiones del sistema, manteniendo la línea funcional establecida y el cumplimiento de la metodología RUP.

Conclusión:

El proceso de validación institucional confirmó la eficiencia, usabilidad y confiabilidad del sistema web implementado. El producto final fue aprobado por la Comandancia de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia, validando su uso como herramienta tecnológica oficial de apoyo a la gestión interna de la organización.

II.1.2.6.19 Evidencia Visuales y Tecnicas

Esta sección presenta las evidencias visuales y técnicas obtenidas durante el proceso de implementación y validación del sistema web desarrollado para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz. Las evidencias comprenden capturas de pantalla, fragmentos de código fuente, entorno de desarrollo y reportes generados, los cuales demuestran el cumplimiento de los requerimientos funcionales definidos bajo la norma IEEE 830 y la metodología RUP.

II.1.2.6.19.1 Capturas de Ejecucion del Sistema

Las siguientes evidencias muestran el funcionamiento de los principales módulos del sistema web. Cada captura corresponde a una funcionalidad específica implementada y validada durante las fases de Construcción y Transición.

II.1.2.6.19.1.1 Modulo 1: Autenticación y Seguridad

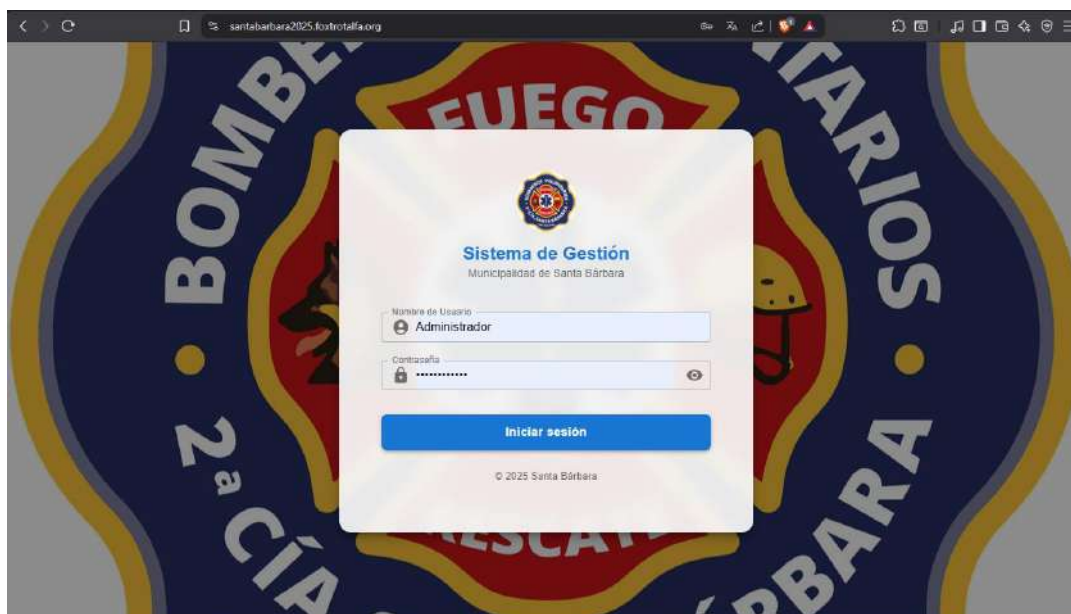


Figura 201. Evidencia del Caso QA-AUTH-01: Interfaz de inicio de sesión con credenciales válidas.
Fuente: Elaboración Propia.

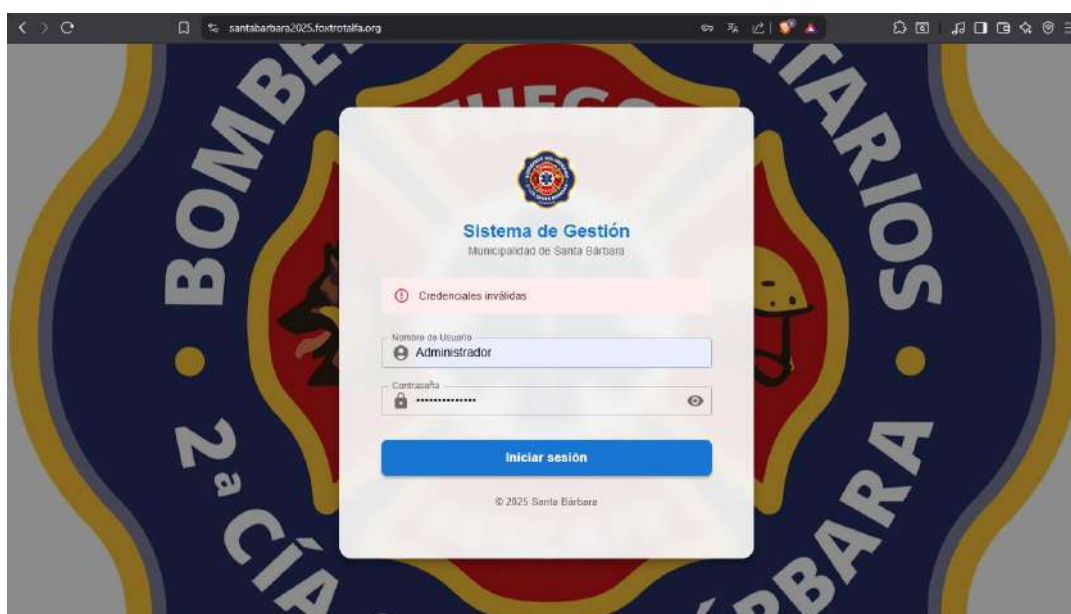


Figura 202. Evidencia del Caso QA-AUTH-02: Validación de seguridad mostrando alerta de "Credenciales inválidas" ante contraseña errónea.
Fuente: Elaboración Propia.

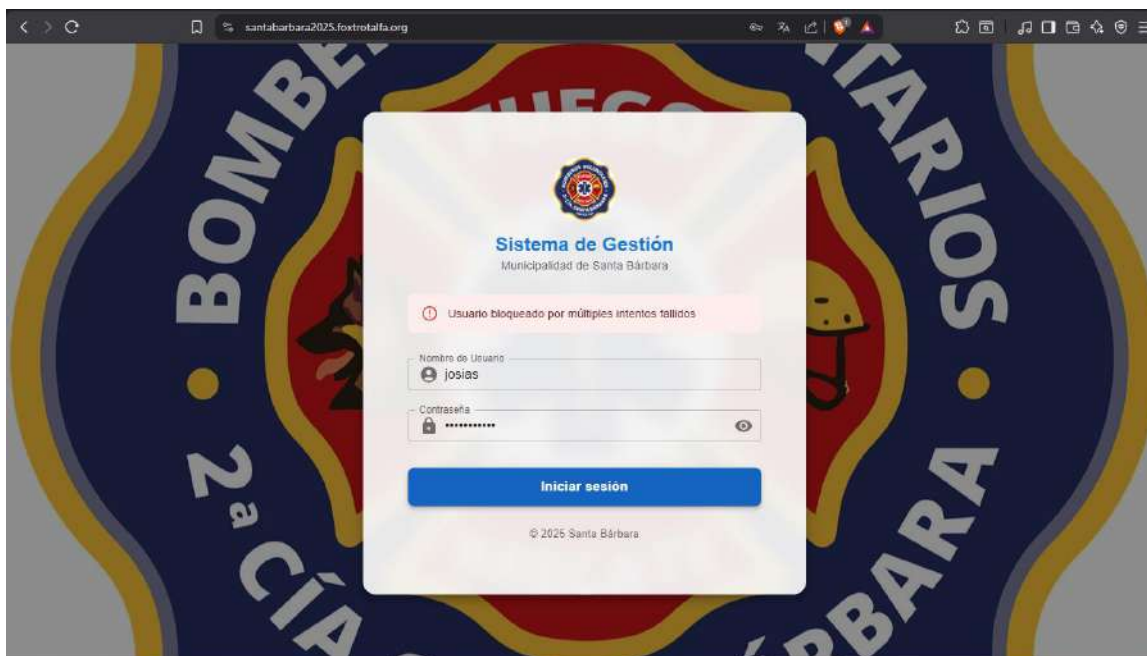


Figura 203. Evidencia del Caso QA-AUTH-03: Mecanismo de seguridad activado: Bloqueo de cuenta tras 3 intentos fallidos.
Fuente: Elaboración Propia.

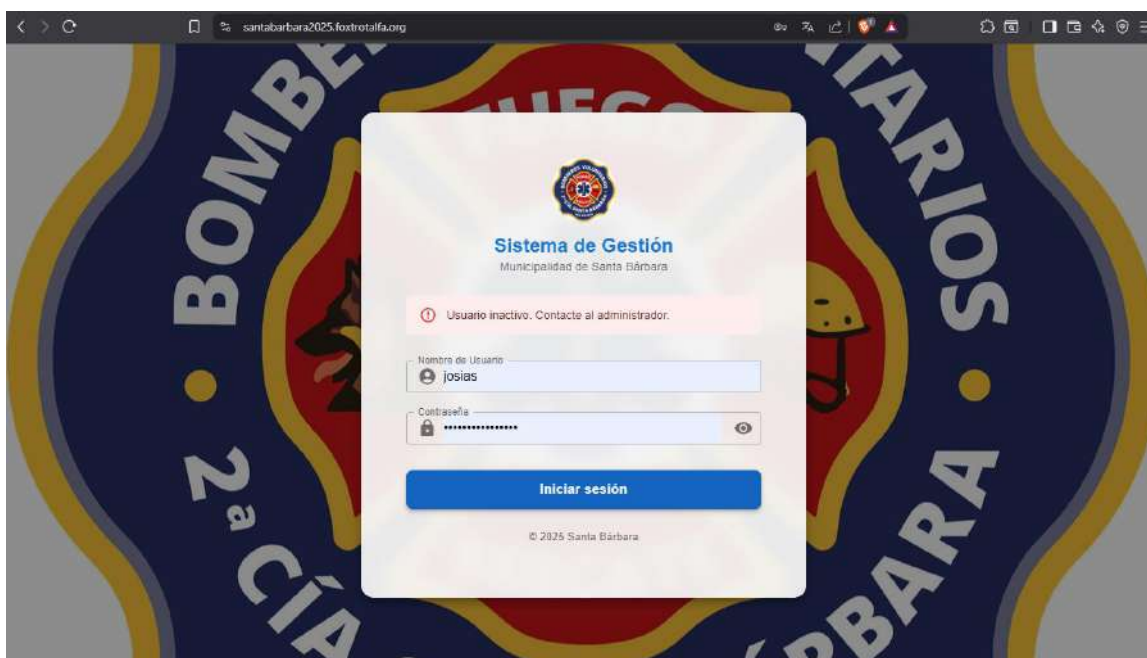


Figura 204. Evidencia del Caso QA-AUTH-04: Restricción de acceso a usuarios con estado "Inactivo" en la base de datos.
Fuente: Elaboración Propia.

II.1.2.6.19.1.2 Modulo 2: Gestión de Usuarios, Roles y Permisos

Figura 205. Evidencia del Caso QA-USR-01: Formulario de registro de nuevo usuario con asignación de roles múltiples.
Fuente: Elaboración Propia.

Figura 206. Evidencia del Caso QA- USR -02: Edición de perfil de usuario y actualización de permisos en tiempo real.
Fuente: Elaboración Propia.

Gestión de Usuarios
Administra usuarios, roles y estados del sistema

Buscar por nombre o usuario

Estado: Todos Roles: Roles

Activo Inactivo

Editar Usuario

Nombre de Usuario: JCapacitaciones Roles: jefe de operaciones Personal Operativo: carlos delgado - CI: 6436067

Seguridad (opcional): Nueva contraseña (dejar en blanco para...): TaskForce141

CANCELAR GUARDAR CAMBIOS

ID	Nombre de Usuario	Rol	Estado	Acciones
4	jcapacitaciones	Encargado de Guardia	Activo	[Editar] [Desactivar]
5	JCapacitaciones	Jefe de operaciones	Activo	[Editar] [Desactivar]

Figura 207. Evidencia del Caso QA- USR -04: Proceso de restablecimiento de contraseña seguro.
Fuente: Elaboración Propia.

Gestión de Usuarios
Administra usuarios, roles y estados del sistema

Buscar por nombre o usuario

Estado: Todos Roles: Roles

Activo Inactivo

Crear Nuevo Usuario

Nombre de Usuario: Administrador Contraseña: ***** Roles: Administrador

Personal Operativo: Edson Nahuel Felix Ancasi Gomez - CI: 8595886

CANCELAR CREAR USUARIO

duplicate key value violates unique constraint "usuarios_nombreusuario_key"

ID	Nombre de Usuario	Rol	Estado	Acciones
5	JCapacitaciones	jefe de operaciones	Activo	[Editar] [Desactivar]

Figura 208. Evidencia del Caso QA- USR -05: Validación de unicidad (Error al intentar registrar un nombre de usuario duplicado).
Fuente: Elaboración Propia.

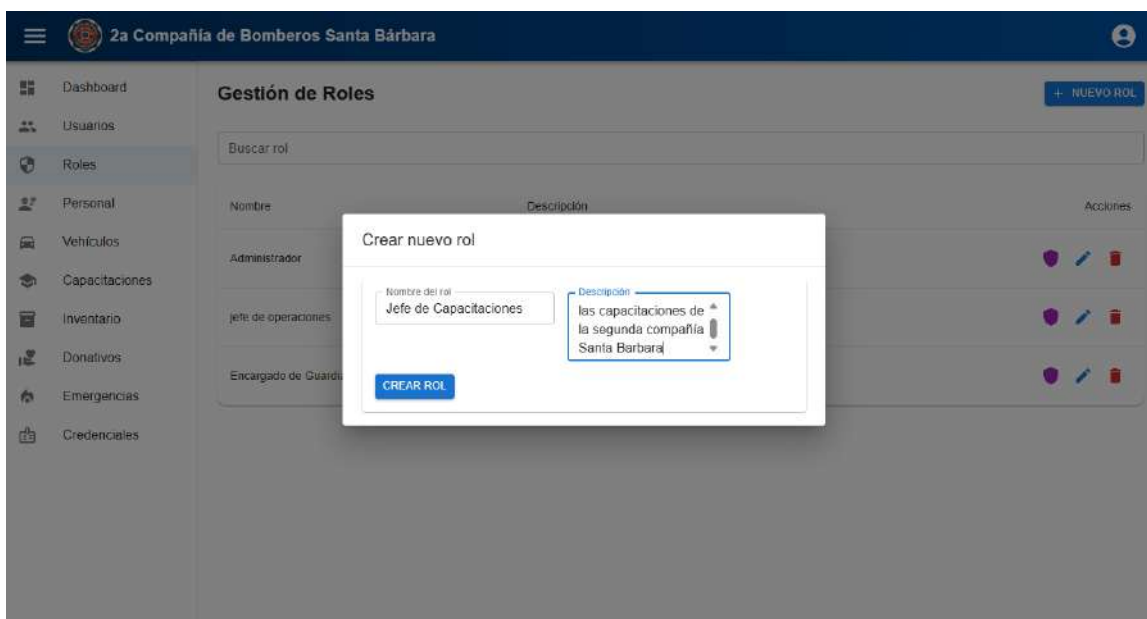


Figura 209. Evidencia del Caso QA-ROL-01: Creación de un nuevo Rol Institucional con permisos granulares.

Fuente: Elaboración Propia.

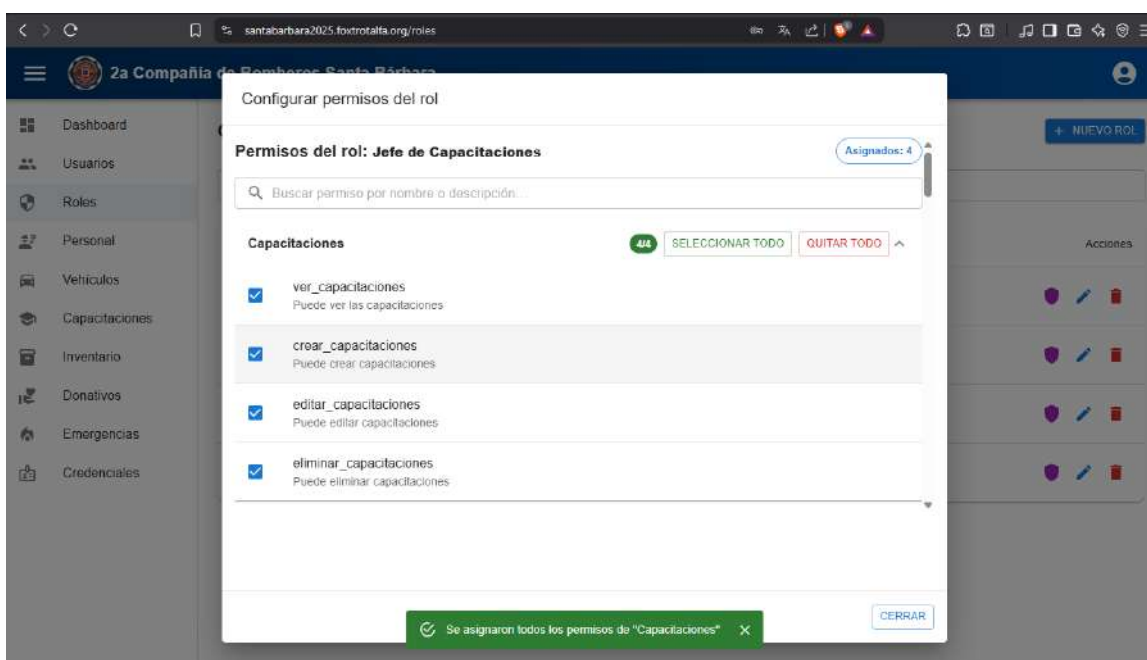


Figura 210. Evidencia del Caso QA-ROL-03: Asignación de permisos específicos (lectura/escritura) a un rol existente.

Fuente: Elaboración Propia.

II.1.2.6.19.1.3 Modulo 3: Gestión de Personal

2a Compañía de Bomberos Santa Bárbara

Personal
Gestión de voluntarios y personal operativo

Buscar por nombre, CI, grado o cohorte

Nuevo Personal

Nombre: Apellido: CI:

Teléfono: Fecha de nacimiento: Clase (cohorta): Grado (rango):

Archivo: ESTATUTO .pdf

Figura 211. Evidencia del Caso QA-PER-01: Ficha de registro de personal con carga de fotografía y documentos.

Fuente: Elaboración Propia.

2a Compañía de Bomberos Santa Bárbara

Personal
Gestión de voluntarios y personal operativo

Buscar por nombre, CI, grado o cohorte

Nuevo Personal

Nombre: Apellido: CI:

El nombre es obligatorio El apellido es obligatorio El CI es obligatorio

Teléfono: Fecha de nacimiento: Clase (cohorta): Grado (rango):

Archivo: ESTATUTO .pdf

Figura 212. Evidencia del Caso QA-PER-02: Validación de campos obligatorios en el formulario de personal.

Fuente: Elaboración Propia.

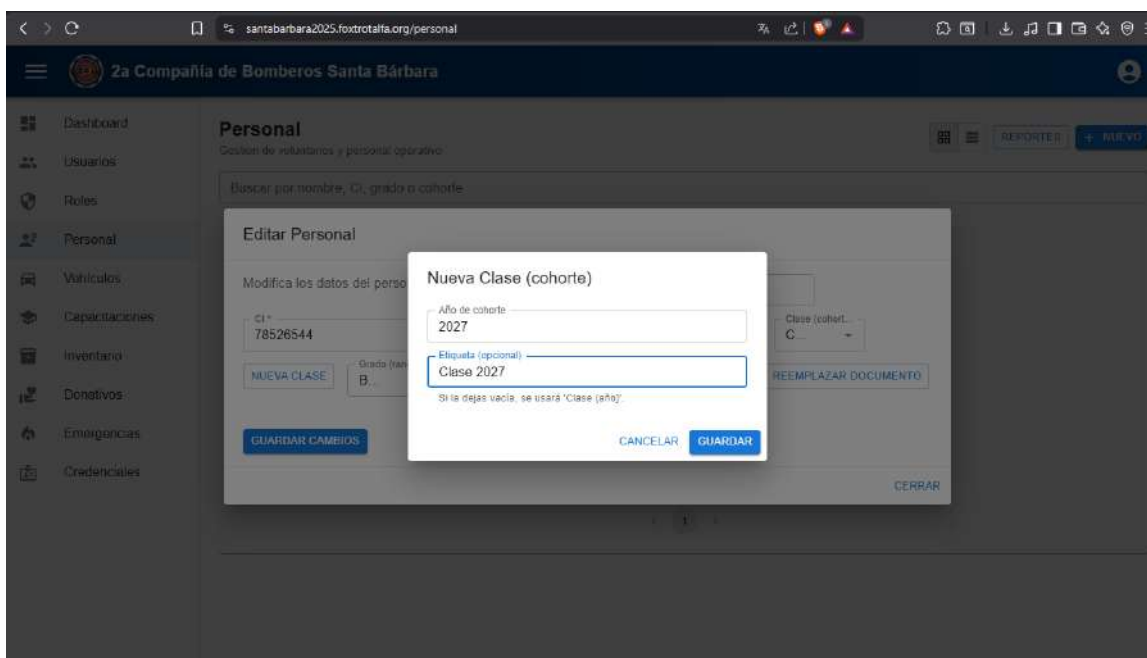


Figura 213. Evidencia del Caso QA-PER-05: Creación dinámica de grados/clases desde el formulario de personal ("Creación al vuelo").
Fuente: Elaboración Propia.

II.1.2.6.19.1.4 Modulo 4: Gestión de Vehículos

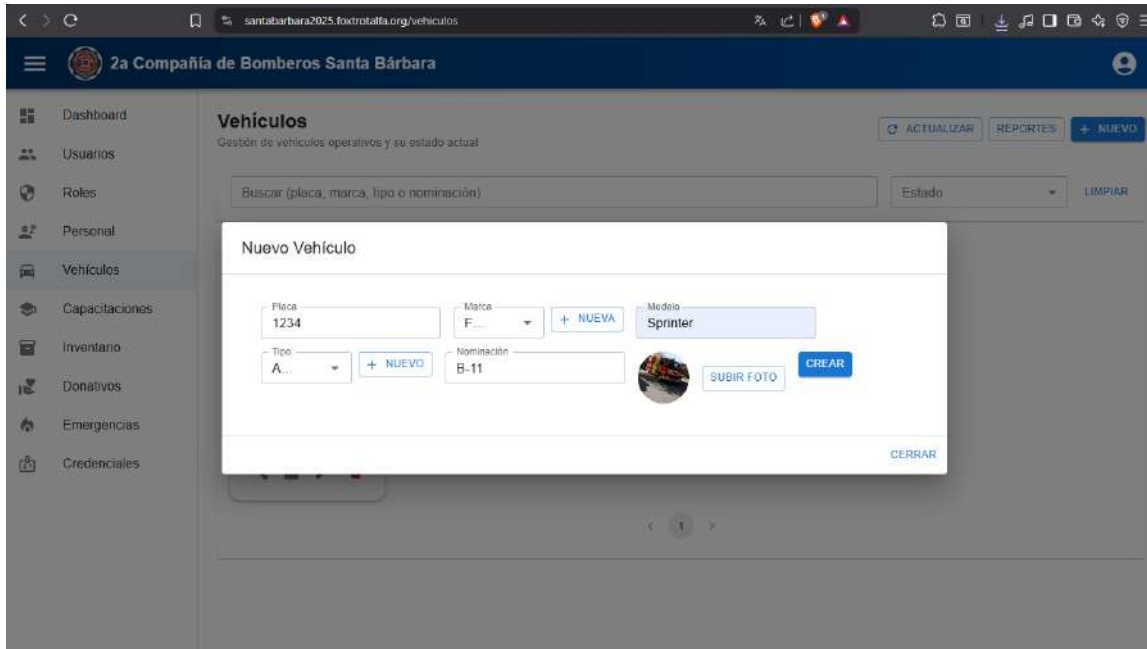


Figura 214. Evidencia del Caso QA-VEH-01: Creación de nuevo vehículo operativo en el sistema.
Fuente: Elaboración Propia.

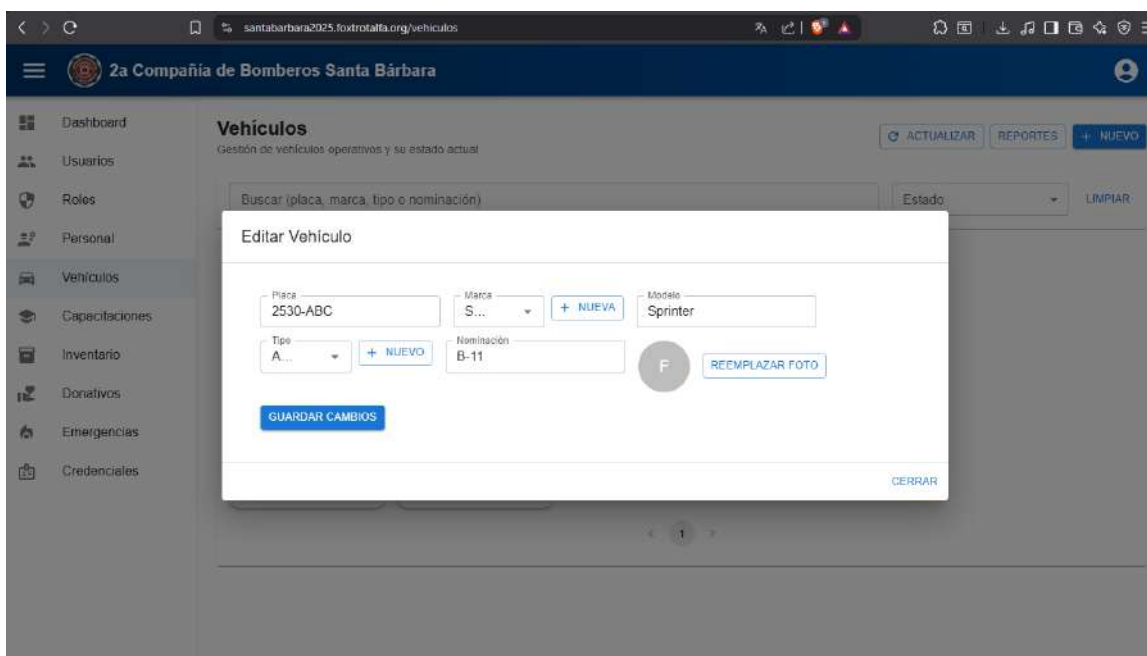


Figura 215. Evidencia del Caso QA-VEH-02: Edición de ficha técnica del vehículo y actualización de estado.

Fuente: Elaboración Propia.

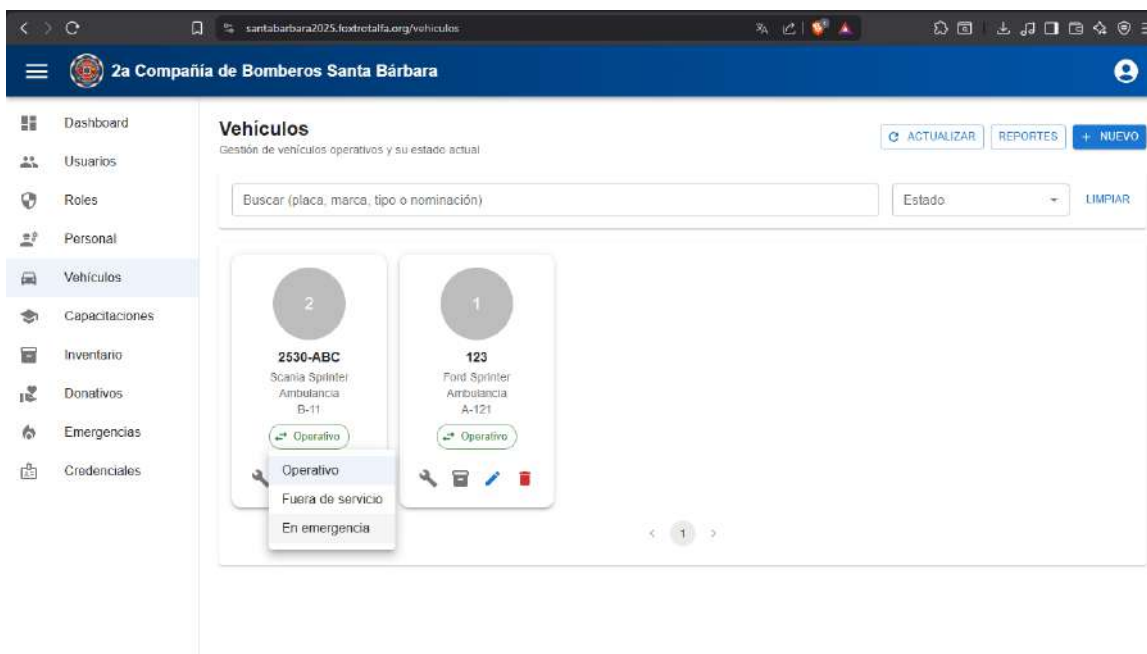


Figura 216. Evidencia del Caso QA-VEH-03: Edición de ficha técnica del vehículo y actualización de estado.

Fuente: Elaboración Propia.

Figura 217. Evidencia del Caso QA-VEH-04: Registro de mantenimiento preventivo asociado a un vehículo.

Fuente: Elaboración Propia.

II.1.2.6.19.1.5 Modulo 5: Gestión de Emergencias

Figura 218. Evidencia del Caso QA- EME-01: Formulario de registro de emergencia con geolocalización en mapa.

Fuente: Elaboración Propia.

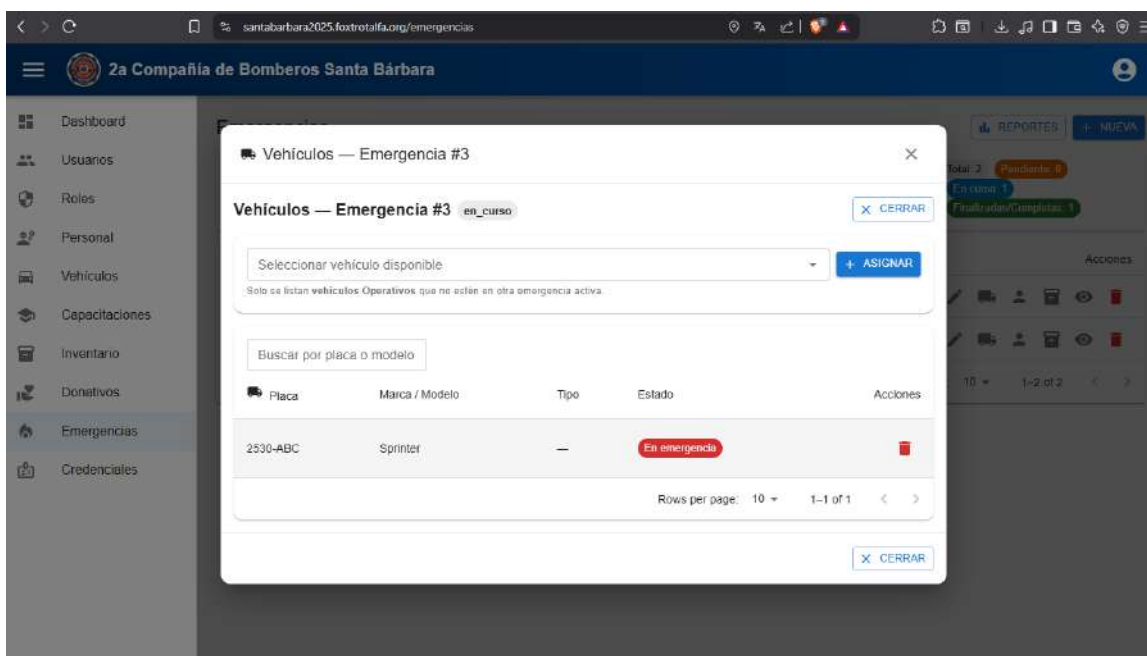


Figura 219. Evidencia del Caso QA- EME-02: Asignación de vehículos disponibles a la emergencia en curso.

Fuente: Elaboración Propia.

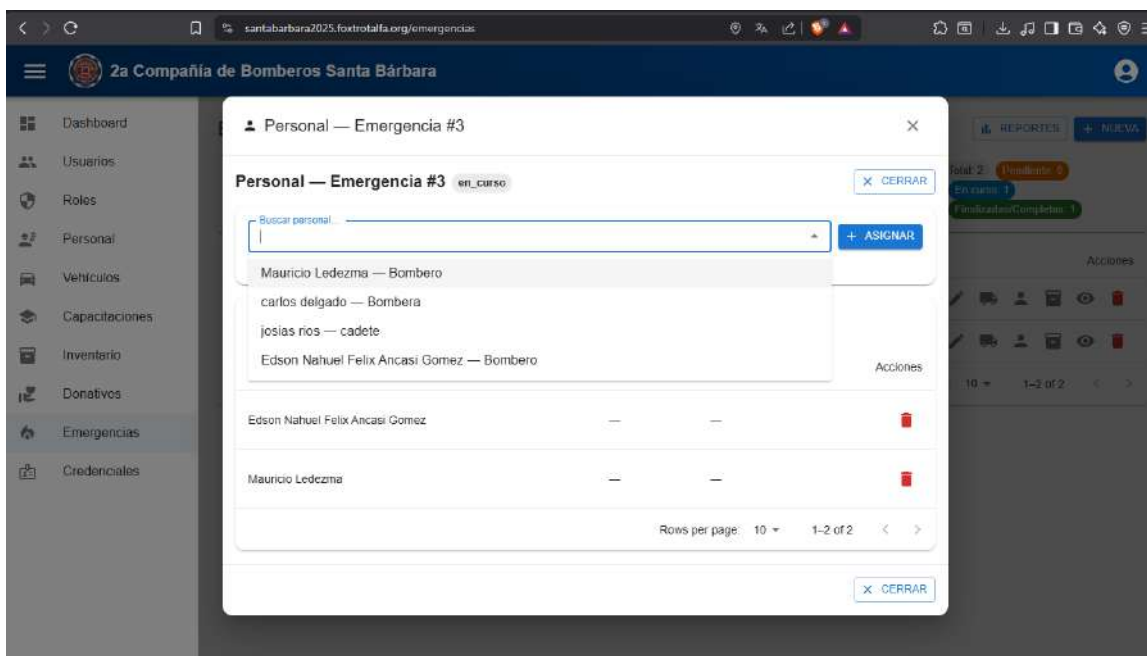


Figura 220. Evidencia del Caso QA- EME-03: Asignación de personal operativo a la emergencia.

Fuente: Elaboración Propia.

Inventario — Emergencia #3

Inventario usado — Emergencia #3

en_curso

Vehículo asignado: 2530-ABC —... Material del vehículo: Cantidad usada: Fecha movimiento: 18/11/2025 02:05

Debe coincidir con el día de la emergencia (2025-11-18)

☐ Dado de baja + AGREGAR

Pendientes

Vehículo	Material	Cantidad	Estado	Fecha movimiento
2530-ABC — Sprinter	Suero Fisiológico	2	Uso	2025-11-18 02:05

REGISTRAR LOTE

Buscar por material

Material	Vehículo	Cantidad	Descargo	Fecha movimiento	Acciones

CERRAR

Figura 221. Evidencia del Caso QA- EME-04: Registro de consumo de materiales e inventario durante la operación.
Fuente: Elaboración Propia.

2a Compañía de Bomberos Santa Bárbara

Dashboard
Usuarios
Roles
Personal
Vehículos
Capacitaciones
Inventario
Donativos
Emergencias
Credenciales

Mapa de la emergencia

Latitud: -21.545617 Longitud: -64.7223

Dirección (aproximada):

Fecha y estado

Fecha y hora: 19/11/2025 02:05 Estado: Finalizada

Documento de solvencia

ADJUNTAR / ACTUALIZAR DOCUMENTO

Archivo: SantaBarbara2025 - public 2 pdf

GUARDAR CAMBIOS

Figura 222. Evidencia del Caso QA- EME-05: Finalización de la emergencia y liberación automática de recursos.
Fuente: Elaboración Propia.

II.1.2.6.19.1.6 Modulo 6: Gestión de Inventario

Editar ítem de inventario

Nombre: CAMBIAR NOMBRE Cantidad: Estado: Descripción:

Usa el botón para cambiar el nombre

Categoría: + Unidad: + Ubicación (n/a): + Proveedor: +

Especificaciones

Sugeridos por categoría (elige cuáles aplicar):

☒ Talla ☐ Talla L ☐ Talla M ☐ none

Talla: X

+ NUEVO ATRIBUTO (CATEGORÍA) + CAMPO PERSONALIZADO

Figura 223. Evidencia del Caso QA- INV-01: Alta de nuevo ítem en inventario con categorización.
Fuente: Elaboración Propia.

Movimientos de Inventario

Buscar (ítem, tipo, motivo, fecha)

Fecha: 8/11/2025, 20:13:34
8/11/2025, 20:12:44

Nuevo movimiento

Ítem:

Stock actual: 100 unidad(es)

Tipo de movimiento:

Cantidad:

Motivo (opcional):

Figura 224. Evidencia del Caso QA- INV-02: Registro de movimiento de entrada (Abastecimiento) actualizando el stock.
Fuente: Elaboración Propia.

2a Compañía de Bomberos Santa Bárbara

Movimientos de Inventario

Buscar (Item, tipo, motivo, fecha)

Fecha

18/11/2025, 22:37:19

8/11/2025, 20:13:34

8/11/2025, 20:12:44

Nuevo movimiento

Suero Fisicoologico

Stock actual: 125 unidad(es)

Tipo de movimiento

Salida

Cantidad

50

Motivo (opcional)

Practicas

CANCELAR GUARDAR

Figura 225. Evidencia del Caso QA- INV-03: Registro de movimiento de salida (Dotación) y rebaja de stock.

Fuente: Elaboración Propia.

2a Compañía de Bomberos Santa Bárbara

Movimientos de Inventario

Buscar (Item, tipo, motivo, fecha)

Fecha

18/11/2025, 22:39:19

18/11/2025, 22:37:53

18/11/2025, 22:37:19

8/11/2025, 20:13:34

8/11/2025, 20:12:44

Stock insuficiente

Solo hay 150 unidades disponibles

OK

Cantidad

Motivo

0 dotacion

0 Practicas

0 Donación

0 j

0 practicas

Rows per page: 10 1-5 of 5

Figura 226. Evidencia del Caso QA- INV-04: Validación de lógica de negocio: Alerta de "Stock Insuficiente".

Fuente: Elaboración Propia.

← VOLVER Asignaciones — Guantes de Baqueta (Stock: 150)

Asignar este ítem a un vehículo

Vehículo: 2530-ABC — Scania Sprinter Cantidad: 25 + ASIGNAR

Buscar por placa, marca o modelo

Vehículo	Marca / Modelo	Cantidad	Acciones
2530-ABC	Scania Sprinter	60	

Rows per page: 10 1-1 of 1

Figura 227. Evidencia del Caso QA- INV-05: Asignación de equipamiento a una unidad vehicular específica.

Fuente: Elaboración Propia.

II.1.2.6.19.1.7 Modulo 7: Gestión de Donativos

Dashboard Gestión de Donativos

REPORTES + NUEVO DONATIVO

Crear Nuevo Donativo

Nuevo donativo

Tipo de donativo: Material +

Material (seleccione o escriba): Guantes de Baqueta

CANCELAR GUARDAR

Crear Donante

Tipo: Institución Razón social: Universidad Mayor de San

Documento (tipo): Documento (número):

Email: Teléfono:

Dirección:

CANCELAR CREAR

CERRAR

Figura 228. Evidencia del Caso QA-DON-01: Registro de donativo monetario creando un nuevo donante desde el mismo formulario ("al vuelo").

Fuente: Elaboración Propia.

The screenshot shows a web application interface for '2a Compañía de Bomberos Santa Bárbara'. The main menu on the left includes Dashboard, Usuarios, Roles, Personal, Vehículos, Capacitaciones, Inventario, Donativos, Emergencias, and Credenciales. The 'Gestión de Donativos' section is active, displaying a modal form titled 'Crear Nuevo Donativo'.

The modal form contains the following fields and controls:

- Tipo de donativo:** A dropdown menu with 'Material' selected.
- Fecha:** A date input field showing '18/11/2025'.
- Estado:** A dropdown menu with 'Aprobado' selected.
- Donante:** A search input field with a placeholder 'Q. G...' and a plus icon.
- Material (selecciona o escribe):** A text input field containing 'Guantes de Baqueta'.
- Cantidad:** A numeric input field with a value of '200'.
- Definir atributos de inventario para:** A checkbox labeled 'Guantes de Baqueta' with a help icon.
- Buttons:** 'CANCELAR' and 'GUARDAR DONATIVO' at the bottom left, and 'CERRAR' at the bottom right.

Figura 229. Evidencia del Caso QA-DON-02: Integración Automática: Donativo de material que crea un nuevo ítem en el Inventario.
Fuente: Elaboración Propia.

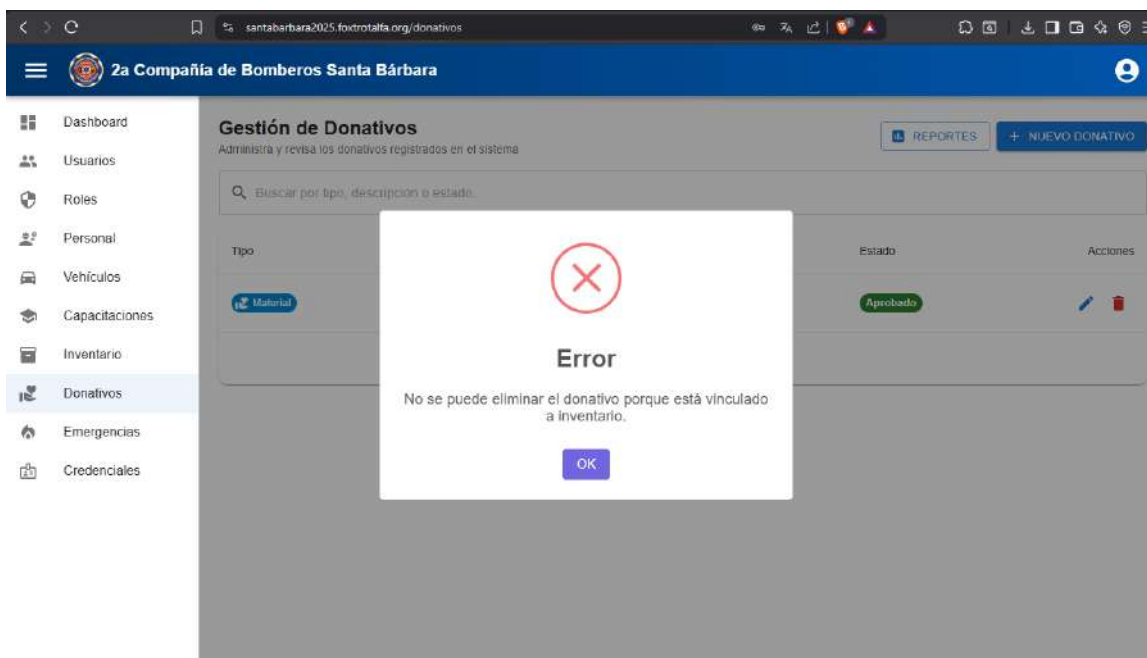


Figura 230. Evidencia del Caso QA-DON-05: Validación de integridad referencial: Error al intentar eliminar un donativo ya vinculado al inventario.
Fuente: Elaboración Propia.

II.1.2.6.19.1.8 Modulo 8: Modulo Capacitaciones y Certificados

Nueva Capacitación

Curso: Interna | Tema: Primeros Auxilios | Fecha de solicitud: 19/11/2025 | Fecha de inicio: 22/11/2025 | Fecha de fin: 22/11/2025

Solicitante: JCapacitaciones | [ADJUNTAR DOCUMENTO \(PDF\)](#) | [CREAR CAPACITACIÓN](#)

Archivo: SantaBarbara2025 - public.pdf

Resumen
Vista previa de la capacitación:

Curso	Tema	Fechas	Solicitante
Interna	Primeros Auxilios	Solicitud: 2025-11-19 Inicio: 2025-11-22 Fin: 2025-11-22	JCapacitaciones

Documento: SantaBarbara2025 - public.pdf

[CERRAR](#)

Figura 231. Evidencia del Caso QA-CAP-05: Formulario de registro de un nuevo curso de capacitación con carga de material digital.
Fuente: Elaboración Propia.

Participantes (Personal)
Interna

No hay personal asignado

[+ ASIGNAR](#)

Asignar personal

Selección personal

- Edson Nahuel Felix Ancasi Gomez — CI: 6595886
- carlos delgado — CI: 6436067
- Mauricio Ledezma — CI: 78528544
- Josias rios — CI: 8301937

Figura 232. Evidencia del Caso QA-CAP-05: Inscripción de participantes internos y externos al curso.
Fuente: Elaboración Propia.

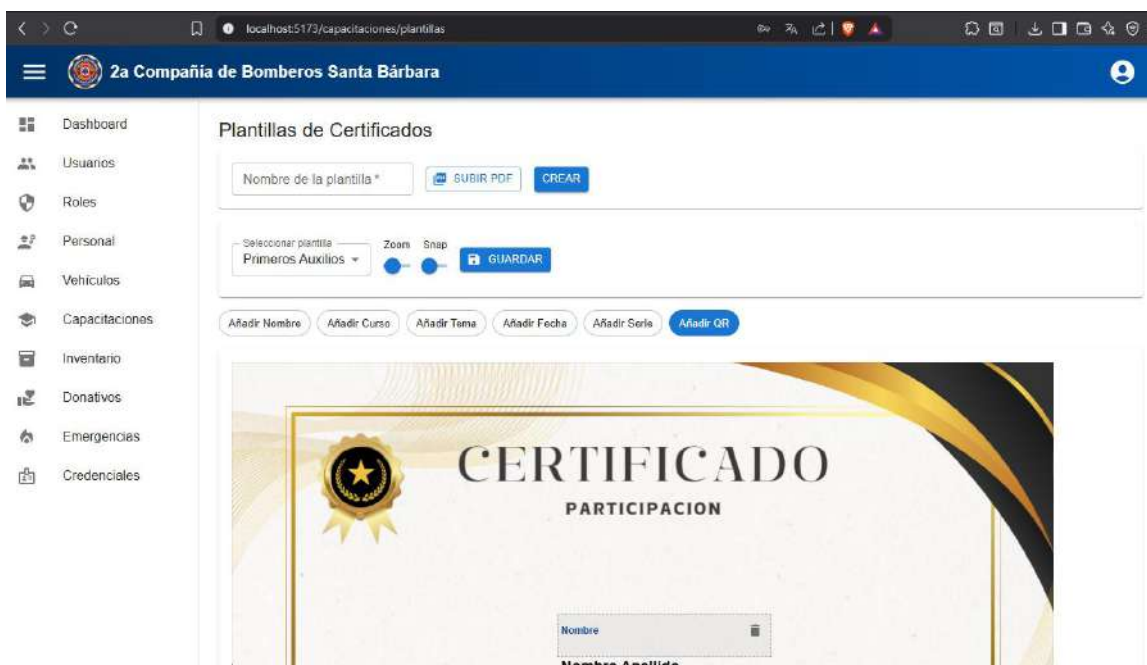


Figura 233. Evidencia del Caso QA-CAP-05: Gestión de plantillas de certificados.
Fuente: Elaboración Propia.

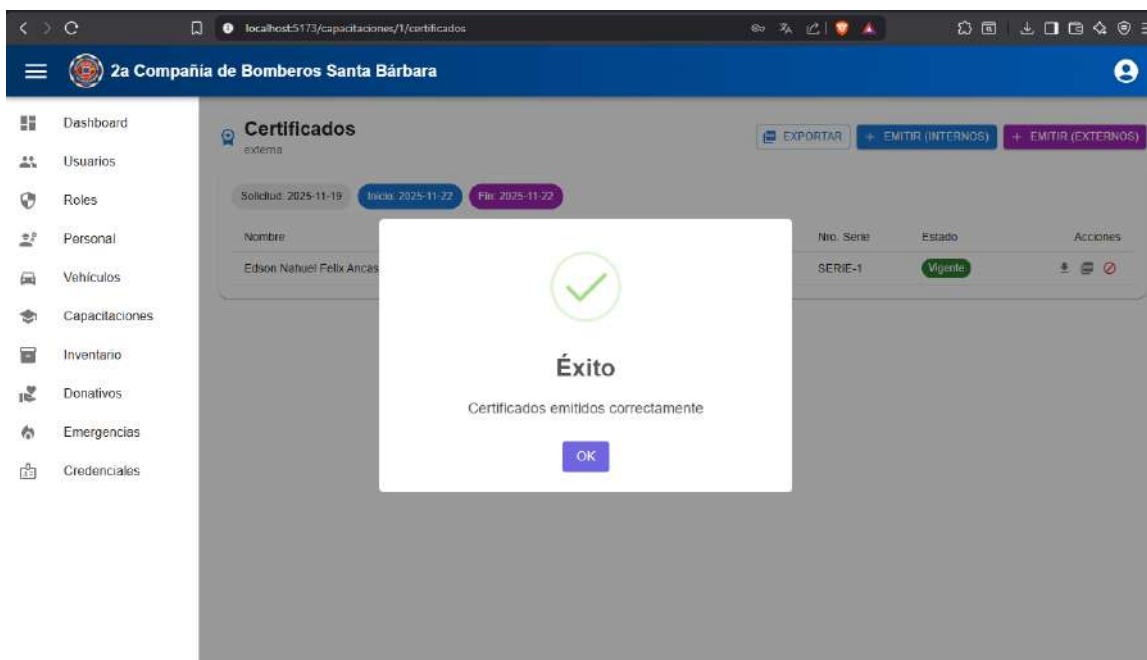


Figura 234. Evidencia del Caso QA-CAP-05: Emisión exitosa y visualización del Certificado PDF con los datos del participante.
Fuente: Elaboración Propia.

II.1.2.6.19.1.9 Modulo 9: Credenciales QR

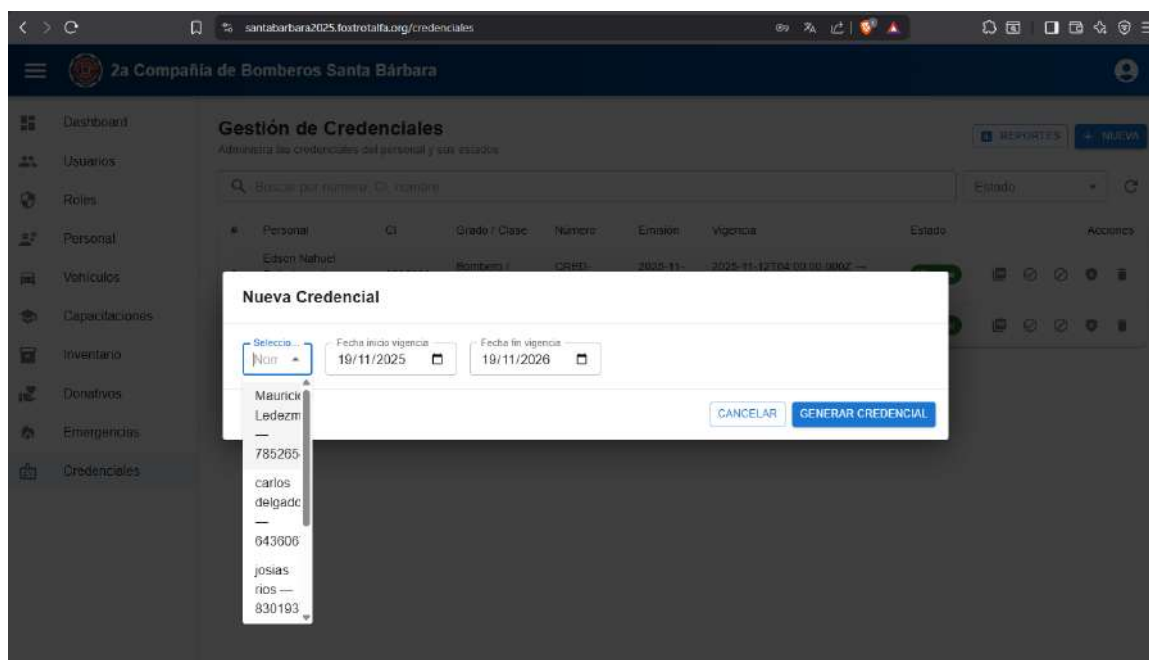


Figura 235. Evidencia del Caso QA-CRE-01: Generación de credencial digital institucional para un bombero activo.

Fuente: Elaboración Propia.



Figura 236. Evidencia del Caso QA- CRE -02: Verificación Pública: Pantalla que ve un ciudadano al escanear el QR (Estado: VIGENTE).

Fuente: Elaboración Propia.

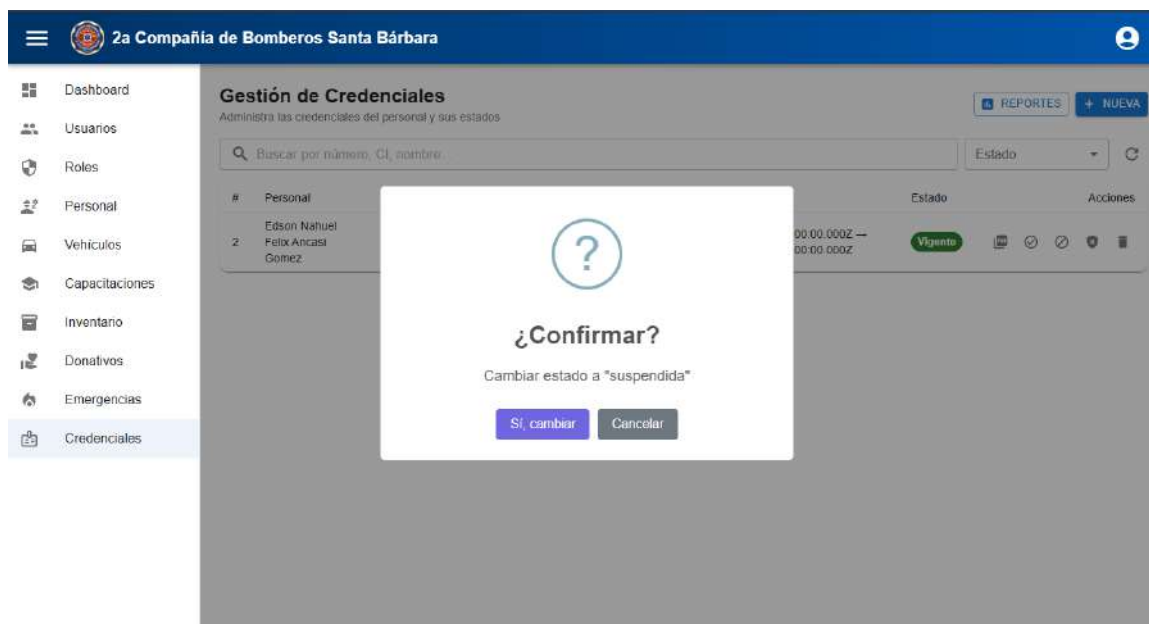


Figura 237. Evidencia del Caso QA- CRE -03: Proceso administrativo de revocación de credencial por baja o sanción.
Fuente: Elaboración Propia.



Figura 238. Evidencia del Caso QA- CRE -04: Verificación Pública de seguridad: Alerta de "CREDENCIAL NO VÁLIDA" al escanear un QR revocado.
Fuente: Elaboración Propia.

II.1.2.6.19.2 Fragmento de Código Fuente

A continuación, se presentan ejemplos representativos del código desarrollado en los componentes frontend y backend, con el fin de evidenciar la implementación técnica del sistema.

a) Backend – Node.js / Express (API de autenticación)

```
const Usuario = require('../models/usuario.model');
const bcrypt = require('bcrypt');
const jwt = require('jsonwebtoken');
const db = require('../config/db');

const AuthController = {
  async login(req, res) {
    const { nombreUsuario, contraseña } = req.body;

    const usuario = await Usuario.findByUsername(nombreUsuario);
    if (!usuario) return res.status(401).json({ error: 'Credenciales inválidas' });

    if (!usuario.estado) {
      return res.status(403).json({ error: 'Usuario inactivo. Contacte al administrador.' });
    }

    const intentos = await db.query(
      'SELECT * FROM intentos_login WHERE idusuario = $1',
      [usuario.idusuario]
    );

    if (intentos.rowCount > 0 && intentos.rows[0].intentos >= 3) {
      return res.status(403).json({ error: 'Usuario bloqueado por múltiples intentos fallidos' });
    }

    const valido = await bcrypt.compare(contraseña, usuario.contraseña);
    if (!valido) {
      if (intentos.rowCount === 0) {
        await db.query(
          'INSERT INTO intentos_login (idusuario, intentos) VALUES ($1, 1)',
          [usuario.idusuario]
        );
      } else {

```

```

    const nuevosIntentos = intentos.rows[0].intentos + 1;

    await db.query(
      'UPDATE intentos_login SET intentos = $1 WHERE idusuario = $2',
      [nuevosIntentos, usuario.idusuario]
    );

    if (nuevosIntentos >= 3) {
      await db.query(
        'UPDATE usuarios SET estado = FALSE WHERE idusuario = $1',
        [usuario.idusuario]
      );
      return res.status(403).json({ error: 'Usuario bloqueado por múltiples intentos fallidos' });
    }

    return res.status(401).json({ error: 'Credenciales inválidas' });
  }

  await db.query('DELETE FROM intentos_login WHERE idusuario = $1', [usuario.idusuario]);

  const token = jwt.sign(
    { idUsuario: usuario.idusuario, idRol: usuario.idrol },
    process.env.JWT_SECRET,
    { expiresIn: '8h' }
  );

  res.json({ token });
};

module.exports = AuthController;

```

b) Frontend – React (Componente de login)

```

import {
  Box,
  Card,
  CardContent,
  Typography,
  TextField,
  Button,

```

```

    InputAdornment,
    IconButton,
    Container,
    Alert,
    CircularProgress,
    useMediaQuery,
  } from "@mui/material";
import { AccountCircle, Lock, Visibility, VisibilityOff } from
"@mui/icons-material";
import { useForm } from "react-hook-form";
import { useNavigate } from "react-router-dom";
import { useTheme } from "@mui/material/styles";
import { useState } from "react";
import API from "../services/axios";
import { useAuth } from "../context/AuthContext";
import escudo from "../assets/img/EscudoSantaBarbara.jpg";

function LoginPage() {
  const {
    register,
    handleSubmit,
    formState: { errors },
  } = useForm();

  const navigate = useNavigate();
  const { login } = useAuth();
  const [loading, setLoading] = useState(false);
  const [errorMsg, setErrorMsg] = useState("");
  const [showPassword, setShowPassword] = useState(false);

  const theme = useTheme();
  const isSmallScreen = useMediaQuery(theme.breakpoints.down("sm"));

  const handleClickShowPassword = () => setShowPassword((prev) => !prev);

  const onSubmit = async (data) => {
    setLoading(true);
    setErrorMsg("");
    try {
      const res = await API.post("/auth/login", data);
      login(res.data.token);
      navigate("/dashboard");
    } catch (err) {
      const mensaje =

```



```

        err.response?.data?.error || "Error desconocido. Intente más
tarde.";
        setErrorMsg(mensaje);
    } finally {
        setLoading(false);
    }
};

return (
    <Box
        sx={{
            position: "relative",
            width: "100vw",
            height: "100vh",
            backgroundImage: `url(${escudo})`,
            backgroundRepeat: "no-repeat",
            backgroundSize: "cover",
            backgroundPosition: "center",
            backgroundAttachment: "fixed",
            display: "flex",
            alignItems: "center",
            justifyContent: "center",
            px: 2,
        }}
    >
        { /* Overlay para contraste */ }
        <Box
            sx={{
                position: "absolute",
                inset: 0,
                background: "rgba(0, 0, 0, 0.45)",
            }}
        />

        <Container
            maxWidth={isSmallScreen ? "xs" : "sm"}
            sx={{
                position: "relative",
                zIndex: 1,
            }}
        >
            <Card
                elevation={8}
                sx={{
                    p: isSmallScreen ? 3 : 5,

```

```

borderRadius: 4,
backdropFilter: "blur(10px)",
backgroundColor: "rgba(255, 255, 255, 0.9)",
}}
>
<CardContent>
  {/* Encabezado */}
  <Box sx={{ textAlign: "center", mb: 3 }}>
    <img
      src={escudo}
      alt="Escudo institucional"
      style={{
        width: isSmallScreen ? 60 : 80,
        height: isSmallScreen ? 60 : 80,
        borderRadius: "50%",
        marginBottom: 8,
      }}
    />
    <Typography
      variant={isSmallScreen ? "h6" : "h5"}
      sx={{
        fontWeight: 700,
        color: "primary.main",
        lineHeight: 1.2,
      }}
    />
    <Typography
      variant="subtitle2"
      color="text.secondary"
      sx={{ fontSize: isSmallScreen ? "0.8rem" : "0.9rem" }}
    />
    <Typography>
      Sistema de Gestión
    </Typography>
    <Typography>
      Municipalidad de Santa Bárbara
    </Typography>
  </Box>

  {/* Alerta de error */}
  {errorMsg && (
    <Alert severity="error" sx={{ mb: 2 }}>
      {errorMsg}
    </Alert>
  )}

  {/* Formulario */}

```

```

<form onSubmit={handleSubmit(onSubmit)} noValidate>
  <TextField
    label="Nombre de Usuario"
    fullWidth
    margin="normal"
    {...register("nombreUsuario", {
      required: "Campo obligatorio",
    })}
    error={!errors.nombreUsuario}
    helperText={errors.nombreUsuario?.message}
    InputProps={{
      startAdornment: (
        <InputAdornment position="start">
          <AccountCircle />
        </InputAdornment>
      ),
    }}
    inputProps={{ "aria-label": "nombre de usuario" }}
    autoFocus
  />

  <TextField
    label="Contraseña"
    type={showPassword ? "text" : "password"}
    fullWidth
    margin="normal"
    {...register("contraseña", {
      required: "Campo obligatorio",
    })}
    error={!errors.contraseña}
    helperText={errors.contraseña?.message}
    InputProps={{
      startAdornment: (
        <InputAdornment position="start">
          <Lock />
        </InputAdornment>
      ),
      endAdornment: (
        <InputAdornment position="end">
          <IconButton
            onClick={handleClickShowPassword}
            edge="end"
            aria-label="mostrar u ocultar contraseña"
          />
        </InputAdornment>
      ),
    }}
  />

```

```

        {showPassword ? <VisibilityOff /> : <Visibility
/>}

        </IconButton>
      </InputAdornment>
    ),
  }}
/>

<Button
  type="submit"
  fullWidth
  variant="contained"
  color="primary"
  disabled={loading}
  sx={{
    mt: 3,
    py: 1.3,
    fontWeight: 700,
    fontSize: isSmallScreen ? "0.9rem" : "1rem",
    borderRadius: 2,
    boxShadow: 3,
    textTransform: "none",
  }}
>
  {loading ? (
    <CircularProgress size={24} color="inherit" />
  ) : (
    "Iniciar sesión"
  )}
</Button>
</form>

{/* Pie de página */}
<Typography
  variant="caption"
  display="block"
  align="center"
  sx={{
    mt: 3,
    color: "text.secondary",
    fontSize: isSmallScreen ? "0.7rem" : "0.8rem",
  }}
>
  © 2025 Santa Bárbara
</Typography>

```

```

        </CardContent>
      </Card>
    </Container>
  </Box>
);
}

export default LoginPage;

```

c) Base de Datos – PostgreSQL

```

CREATE TABLE IF NOT EXISTS public.atributocategoria
(
  idatributo serial NOT NULL,
  idcategoria integer NOT NULL,
  nombre character varying(120) COLLATE pg_catalog."default" NOT NULL,
  tipo character varying(20) COLLATE pg_catalog."default" NOT NULL,
  opciones jsonb,
  activo boolean DEFAULT true,
  obligatorio boolean DEFAULT false,
  CONSTRAINT atributocategoria_pkey PRIMARY KEY (idatributo),
  CONSTRAINT uq_attr_categoria_nombre UNIQUE (idcategoria, nombre)
);

CREATE TABLE IF NOT EXISTS public.capacitacion_institucion
(
  idcapacitacion integer NOT NULL,
  idinstitucion integer NOT NULL,
  CONSTRAINT capacitacion_institucion_pkey PRIMARY KEY (idcapacitacion,
idinstitucion)
);

CREATE TABLE IF NOT EXISTS public.capacitacion_persona
(
  idcapacitacion integer NOT NULL,
  idpersona integer NOT NULL,
  CONSTRAINT capacitacion_persona_pkey PRIMARY KEY (idcapacitacion,
idpersona)
);

CREATE TABLE IF NOT EXISTS public.capacitacion_personal
(
  idcapacitacion integer NOT NULL,
  idpersonal integer NOT NULL,
  CONSTRAINT capacitacion_personal_pkey PRIMARY KEY (idcapacitacion,
idpersonal)
);

CREATE TABLE IF NOT EXISTS public.capacitacion_tema

```

```
(
  idcapacitacion integer NOT NULL,
  idtema integer NOT NULL,
  CONSTRAINT capacitacion_tema_pkey PRIMARY KEY (idcapacitacion,
idtema)
);
```

```
CREATE TABLE IF NOT EXISTS public.capacitaciones
```

```
(
  idcapacitacion serial NOT NULL,
  titulo character varying(100) COLLATE pg_catalog."default" NOT NULL,
  tema text COLLATE pg_catalog."default" NOT NULL,
  fecha date NOT NULL,
  documentosolicitud text COLLATE pg_catalog."default",
  idusuario integer,
  idcurso integer,
  titulopersonalizado character varying(150) COLLATE pg_catalog."default",
  fechasolicitud date,
  fechainicio date,
  fechafin date,
  CONSTRAINT capacitaciones_pkey PRIMARY KEY (idcapacitacion)
);
```

```
CREATE TABLE IF NOT EXISTS public.categorias
```

```
(
  idcategoria serial NOT NULL,
  nombre character varying(120) COLLATE pg_catalog."default" NOT NULL,
  activo boolean DEFAULT true,
  CONSTRAINT categorias_pkey PRIMARY KEY (idcategoria),
  CONSTRAINT categorias_nombre_key UNIQUE (nombre)
);
```

```
CREATE TABLE IF NOT EXISTS public.certificado
```

```
(
  idcertificado serial NOT NULL,
  idcapacitacion integer NOT NULL,
  tipoparticipante character varying(16) COLLATE pg_catalog."default" NOT
NULL,
  idreferencia integer NOT NULL,
  nombre_impreso text COLLATE pg_catalog."default" NOT NULL,
  idplantilla integer,
  titulo text COLLATE pg_catalog."default",
  tema text COLLATE pg_catalog."default",
  fecha date,
  metadata jsonb,
  nroserie character varying(100) COLLATE pg_catalog."default",
  archivopdf text COLLATE pg_catalog."default",
  archivoqr text COLLATE pg_catalog."default",
  CONSTRAINT certificado_pkey PRIMARY KEY (idcertificado),
  CONSTRAINT certificado_nroserie_key UNIQUE (nroserie)
);
```

```
CREATE TABLE IF NOT EXISTS public.certificadoplantilla
(
    idplantilla serial NOT NULL,
    nombre character varying(120) COLLATE pg_catalog."default" NOT NULL,
    archivopdf text COLLATE pg_catalog."default",
    activo boolean DEFAULT true,
    orientacion character varying(10) COLLATE pg_catalog."default",
    dpi integer,
    ancho_px integer,
    alto_px integer,
    campos jsonb,
    CONSTRAINT certificadoplantilla_pkey PRIMARY KEY (idplantilla)
);
```

```
CREATE TABLE IF NOT EXISTS public.clases
(
    idclase serial NOT NULL,
    gestion integer NOT NULL,
    etiqueta character varying(50) COLLATE pg_catalog."default",
    activo boolean DEFAULT true,
    CONSTRAINT clases_pkey PRIMARY KEY (idclase),
    CONSTRAINT uq_clases_gestion UNIQUE (gestion)
);
```

```
CREATE TABLE IF NOT EXISTS public.consumo_general
(
    idconsumo serial NOT NULL,
    iditem integer NOT NULL,
    cantidad integer NOT NULL,
    motivo text COLLATE pg_catalog."default",
    fecha timestamp without time zone DEFAULT CURRENT_TIMESTAMP,
    CONSTRAINT consumo_general_pkey PRIMARY KEY (idconsumo)
);
```

```
CREATE TABLE IF NOT EXISTS public.contenidoscurso
(
    idcontenido serial NOT NULL,
    descripcion text COLLATE pg_catalog."default" NOT NULL,
    archivopdf text COLLATE pg_catalog."default",
    idcapitacion integer,
    nombreak original text COLLATE pg_catalog."default",
    mime character varying(100) COLLATE pg_catalog."default",
    size integer,
    checksum text COLLATE pg_catalog."default",
    CONSTRAINT contenidoscurso_pkey PRIMARY KEY (idcontenido)
);
```

```
CREATE TABLE IF NOT EXISTS public.credencial_historial
(
    idhistorial serial NOT NULL,
```

```

    idcredencial integer NOT NULL,
    evento character varying(20) COLLATE pg_catalog."default" NOT NULL,
    detalle text COLLATE pg_catalog."default",
    actor_id integer,
    created_at timestamp without time zone NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    CONSTRAINT credencial_historial_pkey PRIMARY KEY (idhistorial)
);

```

```

CREATE TABLE IF NOT EXISTS public.credenciales
(
    idcredencial serial NOT NULL,
    codigoqr character varying(255) COLLATE pg_catalog."default" NOT NULL,
    fechaemision date NOT NULL,
    estado character varying(20) COLLATE pg_catalog."default",
    idpersonal integer,
    fecha_inicio_vigencia date,
    fecha_fin_vigencia date,
    numero character varying(50) COLLATE pg_catalog."default",
    motivo_suspension_revocacion text COLLATE pg_catalog."default",
    deleted_at timestamp with time zone,
    CONSTRAINT credenciales_pkey PRIMARY KEY (idcredencial),
    CONSTRAINT credenciales_codigoqr_key UNIQUE (codigoqr)
);

```

```

CREATE TABLE IF NOT EXISTS public.cursos
(
    idcurso serial NOT NULL,
    codigo character varying(40) COLLATE pg_catalog."default",
    nombre character varying(120) COLLATE pg_catalog."default" NOT NULL,
    version character varying(20) COLLATE pg_catalog."default",
    horas integer,
    descripcion text COLLATE pg_catalog."default",
    activo boolean DEFAULT true,
    CONSTRAINT cursos_pkey PRIMARY KEY (idcurso)
);

```

```

CREATE TABLE IF NOT EXISTS public.descripcionemergencia
(
    iddescripcion serial NOT NULL,
    idtipoemergencia integer,
    texto character varying(200) COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT descripcionemergencia_pkey PRIMARY KEY (iddescripcion),
    CONSTRAINT descripcionemergencia_texto_key UNIQUE (texto)
);

```

```

CREATE TABLE IF NOT EXISTS public.donante
(
    iddonante bigserial NOT NULL,
    tipo text COLLATE pg_catalog."default" NOT NULL,
    nombres text COLLATE pg_catalog."default",

```



```

    apellidos text COLLATE pg_catalog."default",
    razon_social text COLLATE pg_catalog."default",
    documento_tipo text COLLATE pg_catalog."default",
    documento_numero text COLLATE pg_catalog."default",
    email text COLLATE pg_catalog."default",
    telefono text COLLATE pg_catalog."default",
    direccion text COLLATE pg_catalog."default",
    activo boolean NOT NULL DEFAULT true,
    created_at timestamp without time zone NOT NULL DEFAULT now(),
    CONSTRAINT donante_pkey PRIMARY KEY (iddonante)
);

```

```

CREATE TABLE IF NOT EXISTS public.donativo_donante
(
    iddonativo integer NOT NULL,
    iddonante bigint NOT NULL,
    principal boolean NOT NULL DEFAULT false,
    aporte_monto numeric(14, 2),
    aporte_desc text COLLATE pg_catalog."default",
    created_at timestamp without time zone NOT NULL DEFAULT now(),
    CONSTRAINT donativo_donante_pkey PRIMARY KEY (iddonativo, iddonante)
);

```

```

CREATE TABLE IF NOT EXISTS public.donativo_tipo
(
    idtipodonativo serial NOT NULL,
    codigo character varying(64) COLLATE pg_catalog."default" NOT NULL,
    nombre character varying(120) COLLATE pg_catalog."default" NOT NULL,
    descripcion text COLLATE pg_catalog."default",
    activo boolean NOT NULL DEFAULT true,
    created_at timestamp without time zone NOT NULL DEFAULT now(),
    CONSTRAINT donativo_tipo_pkey PRIMARY KEY (idtipodonativo),
    CONSTRAINT donativo_tipo_codigo_key UNIQUE (codigo),
    CONSTRAINT donativo_tipo_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.donativos
(
    iddonativo serial NOT NULL,
    tipo character varying(50) COLLATE pg_catalog."default",
    descripcion text COLLATE pg_catalog."default",
    cantidad integer,
    fecha date,
    estado character varying(20) COLLATE pg_catalog."default",
    idtipodonativo integer,
    iddonante bigint,
    CONSTRAINT donativos_pkey PRIMARY KEY (iddonativo)
);

```

```

CREATE TABLE IF NOT EXISTS public.emergencia_inventario
(

```

```

    idemergencia integer NOT NULL,
    iditem integer NOT NULL,
    idvehiculo integer,
    cantidadusada integer DEFAULT 0,
    dadodebaja boolean DEFAULT false,
    fechamovimiento timestamp without time zone,
    CONSTRAINT emergencia_inventario_pkey PRIMARY KEY (idemergencia,
    iditem)
);

```

```

CREATE TABLE IF NOT EXISTS public.emergencia_personal
(
    idemergencia integer NOT NULL,
    idpersonal integer NOT NULL,
    CONSTRAINT emergencia_personal_pkey PRIMARY KEY (idemergencia,
    idpersonal)
);

```

```

CREATE TABLE IF NOT EXISTS public.emergencia_vehiculo
(
    idemergencia integer NOT NULL,
    idvehiculo integer NOT NULL,
    CONSTRAINT emergencia_vehiculo_pkey PRIMARY KEY (idemergencia,
    idvehiculo)
);

```

```

CREATE TABLE IF NOT EXISTS public.emergencias
(
    idemergencia serial NOT NULL,
    ubicacion text COLLATE pg_catalog."default" NOT NULL,
    fechahora timestamp without time zone NOT NULL,
    descripcion text COLLATE pg_catalog."default",
    documentosolvencia text COLLATE pg_catalog."default",
    idtipoemergencia integer,
    estado character varying(20) COLLATE pg_catalog."default" DEFAULT
'pendiente'::character varying,
    iddescripcionemergencia integer,
    lat numeric(9, 6),
    lng numeric(9, 6),
    direcciontexto text COLLATE pg_catalog."default",
    zonaadmin jsonb,
    CONSTRAINT emergencias_pkey PRIMARY KEY (idemergencia)
);

```

```

CREATE TABLE IF NOT EXISTS public.grados
(
    idgrado serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    activo boolean DEFAULT true,
    CONSTRAINT grados_pkey PRIMARY KEY (idgrado),
    CONSTRAINT uq_grados_nombre UNIQUE (nombre)
);

```

```
);
```

```
CREATE TABLE IF NOT EXISTS public.institucion
(
    idinstitucion serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    responsable character varying(100) COLLATE pg_catalog."default",
    contacto character varying(100) COLLATE pg_catalog."default",
    direccion text COLLATE pg_catalog."default",
    tipo character varying(40) COLLATE pg_catalog."default",
    correo character varying(120) COLLATE pg_catalog."default",
    telefono character varying(30) COLLATE pg_catalog."default",
    activo boolean DEFAULT true,
    idpersonaresponsable integer,
    CONSTRAINT institucion_pkey PRIMARY KEY (idinstitucion)
);
```

```
CREATE TABLE IF NOT EXISTS public.intentos_login
(
    idusuario integer NOT NULL,
    intentos integer DEFAULT 1,
    bloqueado boolean DEFAULT false,
    CONSTRAINT intentos_login_pkey PRIMARY KEY (idusuario)
);
```

```
CREATE TABLE IF NOT EXISTS public.inventario
(
    iditem serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    descripcion text COLLATE pg_catalog."default",
    cantidad integer NOT NULL,
    estado character varying(20) COLLATE pg_catalog."default",
    idcategoria integer,
    idunidad integer,
    idubicacion integer,
    idproveedor integer,
    especificaciones jsonb,
    CONSTRAINT inventario_pkey PRIMARY KEY (iditem)
);
```

```
CREATE TABLE IF NOT EXISTS public.inventario_donativo
(
    iditem integer NOT NULL,
    iddonativo integer NOT NULL,
    CONSTRAINT inventario_donativo_pkey PRIMARY KEY (iditem, iddonativo)
);
```

```
CREATE TABLE IF NOT EXISTS public.invmovimiento
(
    idmovimiento serial NOT NULL,
    tipo character varying(16) COLLATE pg_catalog."default" NOT NULL,
```

```

    fecha timestamp without time zone NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    motivo text COLLATE pg_catalog."default",
    referencia text COLLATE pg_catalog."default",
    idusuario integer,
    idvehiculo integer,
    srcubicacion integer,
    dstubicacion integer,
    idmotivobaja integer,
    CONSTRAINT invmovimiento_pkey PRIMARY KEY (idmovimiento)
);

```

```

CREATE TABLE IF NOT EXISTS public.invmovimientolinea
(
    idlinea serial NOT NULL,
    idmovimiento integer NOT NULL,
    iditem integer NOT NULL,
    cantidad integer NOT NULL,
    costounitario numeric(14, 2),
    CONSTRAINT invmovimientolinea_pkey PRIMARY KEY (idlinea)
);

```

```

CREATE TABLE IF NOT EXISTS public.logsisistema
(
    idlog serial NOT NULL,
    idusuario integer,
    fechahora timestamp without time zone DEFAULT CURRENT_TIMESTAMP,
    accion text COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT logsisistema_pkey PRIMARY KEY (idlog)
);

```

```

CREATE TABLE IF NOT EXISTS public.mantenimientovehiculo
(
    idmantenimiento serial NOT NULL,
    fecha date NOT NULL,
    descripcion text COLLATE pg_catalog."default" NOT NULL,
    archivopdf text COLLATE pg_catalog."default",
    idvehiculo integer,
    CONSTRAINT mantenimientovehiculo_pkey PRIMARY KEY (idmantenimiento)
);

```

```

CREATE TABLE IF NOT EXISTS public.marcas
(
    idmarca serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT marcas_pkey PRIMARY KEY (idmarca),
    CONSTRAINT marcas_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.motivobaja
(

```

```

    idmotivobaja serial NOT NULL,
    nombre text COLLATE pg_catalog."default" NOT NULL,
    activo boolean NOT NULL DEFAULT true,
    CONSTRAINT motivobaja_pkey PRIMARY KEY (idmotivobaja),
    CONSTRAINT motivobaja_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.permisos
(
    idpermiso serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    descripcion text COLLATE pg_catalog."default",
    CONSTRAINT permisos_pkey PRIMARY KEY (idpermiso),
    CONSTRAINT permisos_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.personal
(
    idpersonal serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    apellido character varying(100) COLLATE pg_catalog."default" NOT NULL,
    ci character varying(20) COLLATE pg_catalog."default" NOT NULL,
    fechanacimiento date NOT NULL,
    telefono character varying(20) COLLATE pg_catalog."default",
    foto text COLLATE pg_catalog."default",
    filedocumento text COLLATE pg_catalog."default",
    idclase integer NOT NULL,
    idgrado integer NOT NULL,
    activo boolean DEFAULT true,
    CONSTRAINT personal_pkey PRIMARY KEY (idpersonal),
    CONSTRAINT personal_ci_key UNIQUE (ci)
);

```

```

CREATE TABLE IF NOT EXISTS public.personaparticular
(
    idpersona serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    ci character varying(20) COLLATE pg_catalog."default" NOT NULL,
    correo character varying(100) COLLATE pg_catalog."default",
    CONSTRAINT personaparticular_pkey PRIMARY KEY (idpersona)
);

```

```

CREATE TABLE IF NOT EXISTS public.proveedores
(
    idproveedor serial NOT NULL,
    nombre character varying(150) COLLATE pg_catalog."default" NOT NULL,
    contacto character varying(100) COLLATE pg_catalog."default",
    telefono character varying(30) COLLATE pg_catalog."default",
    email character varying(120) COLLATE pg_catalog."default",
    activo boolean DEFAULT true,
    CONSTRAINT proveedores_pkey PRIMARY KEY (idproveedor),

```

```

    CONSTRAINT proveedores_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.rol_permiso
(
    idrol integer NOT NULL,
    idpermiso integer NOT NULL,
    CONSTRAINT rol_permiso_pkey PRIMARY KEY (idrol, idpermiso)
);

```

```

CREATE TABLE IF NOT EXISTS public.roles
(
    idrol serial NOT NULL,
    nombre character varying(50) COLLATE pg_catalog."default" NOT NULL,
    descripcion text COLLATE pg_catalog."default",
    CONSTRAINT roles_pkey PRIMARY KEY (idrol)
);

```

```

CREATE TABLE IF NOT EXISTS public.temas
(
    idtema serial NOT NULL,
    nombre character varying(120) COLLATE pg_catalog."default" NOT NULL,
    activo boolean DEFAULT true,
    CONSTRAINT temas_pkey PRIMARY KEY (idtema),
    CONSTRAINT temas_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.tiposemergencia
(
    idtipoemergencia serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    descripcion text COLLATE pg_catalog."default",
    CONSTRAINT tiposemergencia_pkey PRIMARY KEY (idtipoemergencia),
    CONSTRAINT tiposemergencia_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.tiposvehiculo
(
    idtipo serial NOT NULL,
    nombre character varying(100) COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT tiposvehiculo_pkey PRIMARY KEY (idtipo),
    CONSTRAINT tiposvehiculo_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.ubicaciones
(
    idubicacion serial NOT NULL,
    idpadre integer,
    nombre character varying(120) COLLATE pg_catalog."default" NOT NULL,
    activo boolean DEFAULT true,
    CONSTRAINT ubicaciones_pkey PRIMARY KEY (idubicacion),

```

```

    CONSTRAINT uq_ubicacion_nombre_por_padre UNIQUE (idpadre, nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.unidades
(
    idunidad serial NOT NULL,
    nombre character varying(80) COLLATE pg_catalog."default" NOT NULL,
    activo boolean DEFAULT true,
    CONSTRAINT unidades_pkey PRIMARY KEY (idunidad),
    CONSTRAINT unidades_nombre_key UNIQUE (nombre)
);

```

```

CREATE TABLE IF NOT EXISTS public.usuario_rol
(
    idusuario integer NOT NULL,
    idrol integer NOT NULL,
    CONSTRAINT usuario_rol_pkey PRIMARY KEY (idusuario, idrol)
);

```

```

CREATE TABLE IF NOT EXISTS public.usuarios
(
    idusuario serial NOT NULL,
    nombrequero character varying(50) COLLATE pg_catalog."default" NOT
NULL,
    "contraseña" character varying(255) COLLATE pg_catalog."default" NOT
NULL,
    estado boolean DEFAULT true,
    idrol integer,
    idpersonal integer,
    CONSTRAINT usuarios_pkey PRIMARY KEY (idusuario),
    CONSTRAINT usuarios_nombreusuario_key UNIQUE (nombrequero)
);

```

```

CREATE TABLE IF NOT EXISTS public.vehiculo_inventario
(
    idvehiculo integer NOT NULL,
    iditem integer NOT NULL,
    cantidad integer NOT NULL DEFAULT 1,
    CONSTRAINT vehiculo_inventario_pkey PRIMARY KEY (idvehiculo, iditem)
);

```

```

CREATE TABLE IF NOT EXISTS public.vehiculos
(
    idvehiculo serial NOT NULL,
    placa character varying(20) COLLATE pg_catalog."default" NOT NULL,
    modelo character varying(50) COLLATE pg_catalog."default",
    estado character varying(20) COLLATE pg_catalog."default" DEFAULT
'Operativo':character varying,
    nominacion character varying(10) COLLATE pg_catalog."default" NOT NULL,
    foto text COLLATE pg_catalog."default",
    idmarca integer,

```

```

    idtipo integer,
    CONSTRAINT vehiculos_pkey PRIMARY KEY (idvehiculo)
);

```

```

ALTER TABLE IF EXISTS public.atributocategoria
    ADD CONSTRAINT atributocategoria_idcategoria_fkey FOREIGN KEY
(idcategoria)
    REFERENCES public.categorias (idcategoria) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;

```

```

ALTER TABLE IF EXISTS public.capacitacion_institucion
    ADD CONSTRAINT capacitacion_institucion_idcapacitacion_fkey FOREIGN
KEY (idcapacitacion)
    REFERENCES public.capacitaciones (idcapacitacion) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS idx_cap_ins_cap
    ON public.capacitacion_institucion(idcapacitacion);

```

```

ALTER TABLE IF EXISTS public.capacitacion_institucion
    ADD CONSTRAINT capacitacion_institucion_idinstitucion_fkey FOREIGN KEY
(idinstitucion)
    REFERENCES public.institucion (idinstitucion) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE RESTRICT;

```

```

ALTER TABLE IF EXISTS public.capacitacion_persona
    ADD CONSTRAINT capacitacion_persona_idcapacitacion_fkey FOREIGN
KEY (idcapacitacion)
    REFERENCES public.capacitaciones (idcapacitacion) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_cap_per_ext_cap
    ON public.capacitacion_persona(idcapacitacion);

```

```

ALTER TABLE IF EXISTS public.capacitacion_persona
    ADD CONSTRAINT capacitacion_persona_idpersona_fkey FOREIGN KEY
(idpersona)
    REFERENCES public.personaparticular (idpersona) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.capacitacion_personal
    ADD CONSTRAINT capacitacion_personal_idcapacitacion_fkey FOREIGN
KEY (idcapacitacion)

```



```

REFERENCES public.capacitaciones (idcapacitacion) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_cap_per_int_cap
ON public.capacitacion_personal(idcapacitacion);

```

```

ALTER TABLE IF EXISTS public.capacitacion_personal
ADD CONSTRAINT capacitacion_personal_idpersonal_fkey FOREIGN KEY
(idpersonal)
REFERENCES public.personal (idpersonal) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_cap_per_personal
ON public.capacitacion_personal(idpersonal);

```

```

ALTER TABLE IF EXISTS public.capacitacion_tema
ADD CONSTRAINT capacitacion_tema_idcapacitacion_fkey FOREIGN KEY
(idcapacitacion)
REFERENCES public.capacitaciones (idcapacitacion) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS idx_cap_tema_cap
ON public.capacitacion_tema(idcapacitacion);

```

```

ALTER TABLE IF EXISTS public.capacitacion_tema
ADD CONSTRAINT capacitacion_tema_idtema_fkey FOREIGN KEY (idtema)
REFERENCES public.temas (idtema) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE RESTRICT;

```

```

ALTER TABLE IF EXISTS public.capacitaciones
ADD CONSTRAINT capacitaciones_idcurso_fkey FOREIGN KEY (idcurso)
REFERENCES public.cursos (idcurso) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.capacitaciones
ADD CONSTRAINT capacitaciones_idusuario_fkey FOREIGN KEY (idusuario)
REFERENCES public.usuarios (idusuario) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.certificado
ADD CONSTRAINT certificado_idcapacitacion_fkey FOREIGN KEY
(idcapacitacion)

```

```

REFERENCES public.capacitaciones (idcapacitacion) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS idx_cert_idcap
ON public.certificado(idcapacitacion);

```

```

ALTER TABLE IF EXISTS public.certificado
ADD CONSTRAINT certificado_idplantilla_fkey FOREIGN KEY (idplantilla)
REFERENCES public.certificadoplantilla (idplantilla) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.consumo_general
ADD CONSTRAINT consumo_general_iditem_fkey FOREIGN KEY (iditem)
REFERENCES public.inventario (iditem) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_consumo_item
ON public.consumo_general(iditem);

```

```

ALTER TABLE IF EXISTS public.contenidoscurso
ADD CONSTRAINT contenidoscurso_idcapacitacion_fkey FOREIGN KEY
(idcapacitacion)
REFERENCES public.capacitaciones (idcapacitacion) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_contenido_cap
ON public.contenidoscurso(idcapacitacion);

```

```

ALTER TABLE IF EXISTS public.credencial_historial
ADD CONSTRAINT credencial_historial_idcredencial_fkey FOREIGN KEY
(idcredencial)
REFERENCES public.credenciales (idcredencial) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS idx_cred_hist_cred
ON public.credencial_historial(idcredencial);

```

```

ALTER TABLE IF EXISTS public.credenciales
ADD CONSTRAINT credenciales_idpersonal_fkey FOREIGN KEY (idpersonal)
REFERENCES public.personal (idpersonal) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS uq_cred_vigente_por_persona_active
ON public.credenciales(idpersonal);

```

```

ALTER TABLE IF EXISTS public.descripcionemergencia
  ADD CONSTRAINT descripcionemergencia_idtipoemergencia_fkey FOREIGN
  KEY (idtipoemergencia)
    REFERENCES public.tiposemergencia (idtipoemergencia) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE SET NULL;

```

```

ALTER TABLE IF EXISTS public.donativo_donante
  ADD CONSTRAINT donativo_donante_iddonante_fkey FOREIGN KEY
  (iddonante)
    REFERENCES public.donante (iddonante) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE RESTRICT;
CREATE INDEX IF NOT EXISTS ix_donativo_donante__iddonante
  ON public.donativo_donante(iddonante);

```

```

ALTER TABLE IF EXISTS public.donativo_donante
  ADD CONSTRAINT donativo_donante_iddonativo_fkey FOREIGN KEY
  (iddonativo)
    REFERENCES public.donativos (iddonativo) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;

```

```

ALTER TABLE IF EXISTS public.donativos
  ADD CONSTRAINT donativos_iddonante_fkey FOREIGN KEY (iddonante)
    REFERENCES public.donante (iddonante) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE SET NULL;
CREATE INDEX IF NOT EXISTS ix_donativos__iddonante
  ON public.donativos(iddonante);

```

```

ALTER TABLE IF EXISTS public.donativos
  ADD CONSTRAINT fk_donativos_idtipodonativo FOREIGN KEY
  (idtipodonativo)
    REFERENCES public.donativo_tipo (idtipodonativo) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_donativos_idtipodonativo
  ON public.donativos(idtipodonativo);

```

```

ALTER TABLE IF EXISTS public.emergencia_inventario
  ADD CONSTRAINT emergencia_inventario_idemergencia_fkey FOREIGN
  KEY (idemergencia)
    REFERENCES public.emergencias (idemergencia) MATCH SIMPLE
    ON UPDATE NO ACTION

```

ON DELETE NO ACTION;

```
ALTER TABLE IF EXISTS public.emergencia_inventario
  ADD CONSTRAINT emergencia_inventario_iditem_fkey FOREIGN KEY
(iditem)
  REFERENCES public.inventario (iditem) MATCH SIMPLE
  ON UPDATE NO ACTION
  ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.emergencia_inventario
  ADD CONSTRAINT emergencia_inventario_idvehiculo_fkey FOREIGN KEY
(idvehiculo)
  REFERENCES public.vehiculos (idvehiculo) MATCH SIMPLE
  ON UPDATE NO ACTION
  ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.emergencia_personal
  ADD CONSTRAINT emergencia_personal_idemergencia_fkey FOREIGN KEY
(idemergencia)
  REFERENCES public.emergencias (idemergencia) MATCH SIMPLE
  ON UPDATE NO ACTION
  ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_emerg_per_emerg
  ON public.emergencia_personal(idemergencia);
```

```
ALTER TABLE IF EXISTS public.emergencia_personal
  ADD CONSTRAINT emergencia_personal_idpersonal_fkey FOREIGN KEY
(idpersonal)
  REFERENCES public.personal (idpersonal) MATCH SIMPLE
  ON UPDATE NO ACTION
  ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_emerg_per_personal
  ON public.emergencia_personal(idpersonal);
```

```
ALTER TABLE IF EXISTS public.emergencia_vehiculo
  ADD CONSTRAINT emergencia_vehiculo_idemergencia_fkey FOREIGN KEY
(idemergencia)
  REFERENCES public.emergencias (idemergencia) MATCH SIMPLE
  ON UPDATE NO ACTION
  ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_emergencia_vehiculo_emerg
  ON public.emergencia_vehiculo(idemergencia);
```

```
ALTER TABLE IF EXISTS public.emergencia_vehiculo
```

```

ADD CONSTRAINT emergencia_vehiculo_idvehiculo_fkey FOREIGN KEY
(idvehiculo)
REFERENCES public.vehiculos (idvehiculo) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_emergencia_vehiculo_veh
ON public.emergencia_vehiculo(idvehiculo);

```

```

ALTER TABLE IF EXISTS public.emergencias
ADD CONSTRAINT emergencias_iddescripcionemergencia_fkey FOREIGN
KEY (iddescripcionemergencia)
REFERENCES public.descripcionemergencia (iddescripcion) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.emergencias
ADD CONSTRAINT emergencias_idtipoemergencia_fkey FOREIGN KEY
(idtipoemergencia)
REFERENCES public.tiposemergencia (idtipoemergencia) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_emergencias_tipo
ON public.emergencias(idtipoemergencia);

```

```

ALTER TABLE IF EXISTS public.institucion
ADD CONSTRAINT institucion_idpersonaresponsable_fkey FOREIGN KEY
(idpersonaresponsable)
REFERENCES public.personaparticular (idpersona) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_institucion_idpersona_resp
ON public.institucion(idpersonaresponsable);

```

```

ALTER TABLE IF EXISTS public.intentos_login
ADD CONSTRAINT intentos_login_idusuario_fkey FOREIGN KEY (idusuario)
REFERENCES public.usuarios (idusuario) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS intentos_login_pkey
ON public.intentos_login(idusuario);

```

```

ALTER TABLE IF EXISTS public.inventario
ADD CONSTRAINT inventario_idcategoria_fkey FOREIGN KEY (idcategoria)
REFERENCES public.categorias (idcategoria) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;

```

```
CREATE INDEX IF NOT EXISTS idx_inventario_idcategoria
ON public.inventario(idcategoria);
```

```
ALTER TABLE IF EXISTS public.inventario
ADD CONSTRAINT inventario_idproveedor_fkey FOREIGN KEY (idproveedor)
REFERENCES public.proveedores (idproveedor) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_inventario_idproveedor
ON public.inventario(idproveedor);
```

```
ALTER TABLE IF EXISTS public.inventario
ADD CONSTRAINT inventario_idubicacion_fkey FOREIGN KEY (idubicacion)
REFERENCES public.ubicaciones (idubicacion) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_inventario_idubicacion
ON public.inventario(idubicacion);
```

```
ALTER TABLE IF EXISTS public.inventario
ADD CONSTRAINT inventario_idunidad_fkey FOREIGN KEY (idunidad)
REFERENCES public.unidades (idunidad) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_inventario_idunidad
ON public.inventario(idunidad);
```

```
ALTER TABLE IF EXISTS public.inventario_donativo
ADD CONSTRAINT inventario_donativo_iddonativo_fkey FOREIGN KEY
(iddonativo)
REFERENCES public.donativos (iddonativo) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.inventario_donativo
ADD CONSTRAINT inventario_donativo_iditem_fkey FOREIGN KEY (iditem)
REFERENCES public.inventario (iditem) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.invmovimiento
ADD CONSTRAINT invmovimiento_dstubicacion_fkey FOREIGN KEY
(dstubicacion)
REFERENCES public.ubicaciones (idubicacion) MATCH SIMPLE
ON UPDATE NO ACTION
```

```

ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_inv_mov_dst
ON public.invmovimiento(dstubicacion);

```

```

ALTER TABLE IF EXISTS public.invmovimiento
ADD CONSTRAINT invmovimiento_idmotivobaja_fkey FOREIGN KEY
(idmotivobaja)
REFERENCES public.motivobaja (idmotivobaja) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_invmov_idmotivobaja
ON public.invmovimiento(idmotivobaja);

```

```

ALTER TABLE IF EXISTS public.invmovimiento
ADD CONSTRAINT invmovimiento_idusuario_fkey FOREIGN KEY (idusuario)
REFERENCES public.usuarios (idusuario) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.invmovimiento
ADD CONSTRAINT invmovimiento_idvehiculo_fkey FOREIGN KEY
(idvehiculo)
REFERENCES public.vehiculos (idvehiculo) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.invmovimiento
ADD CONSTRAINT invmovimiento_srcubicacion_fkey FOREIGN KEY
(srcubicacion)
REFERENCES public.ubicaciones (idubicacion) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_inv_mov_src
ON public.invmovimiento(srcubicacion);

```

```

ALTER TABLE IF EXISTS public.invmovimientolinea
ADD CONSTRAINT invmovimientolinea_iditem_fkey FOREIGN KEY (iditem)
REFERENCES public.inventario (iditem) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_inv_mov_linea_item
ON public.invmovimientolinea(iditem);

```

```

ALTER TABLE IF EXISTS public.invmovimientolinea

```

```

    ADD CONSTRAINT invmovimientolinea_idmovimiento_fkey FOREIGN KEY
(idmovimiento)
    REFERENCES public.invmovimiento (idmovimiento) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS idx_inv_mov_linea_mov
    ON public.invmovimientolinea(idmovimiento);

```

```

ALTER TABLE IF EXISTS public.logsistema
    ADD CONSTRAINT logsistema_idusuario_fkey FOREIGN KEY (idusuario)
    REFERENCES public.usuarios (idusuario) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.mantenimientovehiculo
    ADD CONSTRAINT mantenimientovehiculo_idvehiculo_fkey FOREIGN KEY
(idvehiculo)
    REFERENCES public.vehiculos (idvehiculo) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.personal
    ADD CONSTRAINT personal_idclase_fkey FOREIGN KEY (idclase)
    REFERENCES public.clases (idclase) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_personal_idclase
    ON public.personal(idclase);

```

```

ALTER TABLE IF EXISTS public.personal
    ADD CONSTRAINT personal_idgrado_fkey FOREIGN KEY (idgrado)
    REFERENCES public.grados (idgrado) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_personal_idgrado
    ON public.personal(idgrado);

```

```

ALTER TABLE IF EXISTS public.rol_permiso
    ADD CONSTRAINT rol_permiso_idpermiso_fkey FOREIGN KEY (idpermiso)
    REFERENCES public.permisos (idpermiso) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_rol_permiso_permiso
    ON public.rol_permiso(idpermiso);

```



```

ALTER TABLE IF EXISTS public.rol_permiso
  ADD CONSTRAINT rol_permiso_idrol_fkey FOREIGN KEY (idrol)
    REFERENCES public.roles (idrol) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_rol_permiso_rol
  ON public.rol_permiso(idrol);

```

```

ALTER TABLE IF EXISTS public.ubicaciones
  ADD CONSTRAINT ubicaciones_idpadre_fkey FOREIGN KEY (idpadre)
    REFERENCES public.ubicaciones (idubicacion) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE SET NULL;

```

```

ALTER TABLE IF EXISTS public.usuario_rol
  ADD CONSTRAINT usuario_rol_idrol_fkey FOREIGN KEY (idrol)
    REFERENCES public.roles (idrol) MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE RESTRICT;
CREATE INDEX IF NOT EXISTS idx_usuario_rol_rol
  ON public.usuario_rol(idrol);

```

```

ALTER TABLE IF EXISTS public.usuario_rol
  ADD CONSTRAINT usuario_rol_idusuario_fkey FOREIGN KEY (idusuario)
    REFERENCES public.usuarios (idusuario) MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS idx_usuario_rol_usuario
  ON public.usuario_rol(idusuario);

```

```

ALTER TABLE IF EXISTS public.usuarios
  ADD CONSTRAINT usuarios_idpersonal_fkey FOREIGN KEY (idpersonal)
    REFERENCES public.personal (idpersonal) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.usuarios
  ADD CONSTRAINT usuarios_idrol_fkey FOREIGN KEY (idrol)
    REFERENCES public.roles (idrol) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

```

```

ALTER TABLE IF EXISTS public.vehiculo_inventario
  ADD CONSTRAINT vehiculo_inventario_iditem_fkey FOREIGN KEY (iditem)
    REFERENCES public.inventario (iditem) MATCH SIMPLE

```

```

ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_vi_item
ON public.vehiculo_inventario(iditem);

```

```

ALTER TABLE IF EXISTS public.vehiculo_inventario
ADD CONSTRAINT vehiculo_inventario_idvehiculo_fkey FOREIGN KEY
(idvehiculo)
REFERENCES public.vehiculos (idvehiculo) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_vi_vehiculo
ON public.vehiculo_inventario(idvehiculo);

```

```

ALTER TABLE IF EXISTS public.vehiculos
ADD CONSTRAINT vehiculos_idmarca_fkey FOREIGN KEY (idmarca)
REFERENCES public.marcas (idmarca) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_vehiculos_idmarca
ON public.vehiculos(idmarca);

```

```

ALTER TABLE IF EXISTS public.vehiculos
ADD CONSTRAINT vehiculos_idtipo_fkey FOREIGN KEY (idtipo)
REFERENCES public.tiposvehiculo (idtipo) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION;
CREATE INDEX IF NOT EXISTS idx_vehiculos_idtipo
ON public.vehiculos(idtipo);

```

II.1.2.6.19.3 Entorno de Desarrollo y Herramientas Utilizadas

El sistema fue desarrollado e implementado en un entorno multiplataforma con las siguientes herramientas y tecnologías:

Categoría	Herramienta / Tecnología	Descripción de uso
Lenguaje de programación	JavaScript (ES6)	Lógica principal del frontend y backend
Framework frontend	React.js	Desarrollo de interfaz web dinámica
Framework backend	Node.js + Express	Construcción de API REST y manejo de rutas
Base de datos	PostgreSQL v15	Almacenamiento estructurado de datos institucionales
ORM	Sequelize	Mapeo objeto-relacional entre Node.js y PostgreSQL
Entorno de desarrollo	Visual Studio Code	Codificación, depuración y control de versiones
Pruebas de API	Postman	Validación de endpoints del servidor
Gestión de base de datos	PgAdmin 4	Administración y consultas SQL
Control de versiones	Git / GitHub	Seguimiento y respaldo del código fuente
Servidor de despliegue local	Node.js v20	Pruebas locales antes del despliegue institucional

Tabla 85. Evidencias Visuales y Técnicas: Entorno de desarrollo y Herramientas utilizadas
Fuente: Elaboración Propia.

II.1.2.6.19.4 Reportes del Sistema y Salidas Exportadas

El sistema incluye una funcionalidad para la generación y exportación de reportes en formatos PDF, lo que permite la documentación y respaldo de la información institucional. Los principales reportes generados son:

- Reporte de usuarios registrados (exportable en PDF).
- Reporte de emergencias atendidas (con filtros por fechas y estado).
- Reporte de inventario (con stock disponible y observaciones).
- Reporte de capacitaciones (incluye responsables, fechas y participantes).
- Credenciales institucionales con código QR (generadas en formato PDF individual).

Cada reporte fue validado conforme al plan de pruebas, garantizando que la información exportada corresponda a los datos almacenados en la base institucional.

Conclusión General:

Las evidencias visuales y técnicas presentadas respaldan la implementación exitosa del sistema web desarrollado. La integración coherente entre código, base de datos y entorno de ejecución demuestra la solidez técnica del proyecto y su conformidad con los lineamientos definidos por la metodología RUP y la norma IEEE 830.

II.2 Componente II: Capacitacion a la 2a Compañía de Bomberos Santa Barbara SAR Bolivia- La Paz .

II.2.1 Introduccion

La capacitación constituye una etapa fundamental dentro del proceso de implementación del sistema web para el mejoramiento de la gestión operativa de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, ya que garantiza que el personal involucrado adquiera los conocimientos necesarios para el uso correcto, eficiente y seguro de la herramienta tecnológica desarrollada.

El programa de capacitación fue diseñado con un enfoque práctico y participativo, orientado a fortalecer las competencias digitales del personal operativo, administrativo y de mando. A través de sesiones teóricas virtuales y prácticas presenciales, se busca asegurar que los usuarios comprendan el funcionamiento integral del sistema, sus módulos y la importancia de su correcta aplicación en las operaciones institucionales.

El plan formativo contempla la instrucción en los diez módulos funcionales que componen el sistema web:

- Gestión de Inicio de Sesión,
- Gestión de Usuarios,
- Gestión de Roles,
- Gestión de Personal,
- Gestión de Emergencias,
- Gestión de Inventario,
- Gestión de Donativos,
- Gestión de Vehículos,
- Gestión de Capacitaciones, y
- Gestión de Credenciales Digitales con Código QR.

Cada módulo fue abordado desde una perspectiva teórico-práctica, con ejemplos reales de la operatividad de la compañía, simulando escenarios cotidianos de respuesta a emergencias, registro de recursos y administración de personal. Este enfoque permitió no solo el aprendizaje técnico, sino también la comprensión del impacto que la digitalización tiene sobre la eficiencia institucional y la toma de decisiones.

La capacitación fue planificada bajo la metodología Rational Unified Process (RUP), siguiendo las fases de Transición y Validación, asegurando que el producto entregado sea plenamente funcional, documentado y comprendido por sus usuarios finales. Asimismo, se elaboraron materiales de apoyo como el manual de usuario, guía de instalación y presentaciones digitales que facilitan la comprensión de las funcionalidades del sistema.

En síntesis, la capacitación no solo busca instruir en el manejo del sistema, sino también fomentar una cultura institucional basada en el uso responsable de las Tecnologías de la Información y Comunicación (TIC), impulsando la transformación digital dentro de la compañía y garantizando la sostenibilidad del sistema a largo plazo.

II.2.2 Proposito

El propósito de este programa de capacitación es asegurar la correcta adopción y uso operativo del sistema web desarrollado para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, garantizando que todos los usuarios —desde la comandancia hasta el personal operativo— comprendan las funcionalidades, alcances y beneficios de la herramienta tecnológica implementada.

Esta capacitación busca que el personal institucional pueda utilizar cada uno de los módulos del sistema de manera autónoma, responsable y eficiente, fortaleciendo la gestión de emergencias, el control de inventarios, la administración del personal y la transparencia institucional mediante el uso de credenciales digitales verificables.

De forma más específica, el propósito central es transferir el conocimiento técnico y funcional del sistema a los usuarios finales, de modo que sean capaces de:

- Navegar e interactuar correctamente con la interfaz del sistema web.
- Ejecutar las operaciones esenciales de los módulos según sus roles asignados (registro, edición, consulta y generación de reportes).
- Aplicar buenas prácticas de seguridad digital en el uso de credenciales, contraseñas y roles.
- Contribuir al mantenimiento y actualización continua de la información institucional dentro del sistema.

El desarrollo de esta capacitación, por tanto, no se limita al entrenamiento técnico, sino que promueve la integración del sistema como parte del trabajo cotidiano de la compañía, potenciando la eficiencia operativa, la comunicación interna y la trazabilidad de todas las actividades que realiza la institución.

II.2.3 Objetivos

II.2.3.1 Objetivo General

Brindar al personal de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz las competencias necesarias para el uso eficiente, seguro y responsable del sistema web de gestión operativa, fortaleciendo la digitalización de los procesos institucionales mediante la aplicación de Tecnologías de la Información y Comunicación (TIC).

Este objetivo busca asegurar que cada usuario, de acuerdo con su rol (comandante, jefe de área o personal operativo), sea capaz de utilizar las funcionalidades del sistema para optimizar la administración de emergencias, recursos humanos, materiales y donativos, garantizando una gestión moderna, ágil y transparente.

II.2.3.2 Objetivos Especificos

- Capacitar al personal institucional en el uso de los diez módulos funcionales del sistema web: Inicio de Sesión, Usuarios, Roles, Personal, Emergencias, Inventario, Donativos, Vehículos, Capacitaciones y Credenciales QR.
- Promover la adopción tecnológica en la compañía, fomentando el uso de herramientas digitales como parte de la rutina operativa y administrativa.
- Instruir en el manejo de los manuales y materiales de apoyo, garantizando que cada usuario pueda consultar, instalar y actualizar el sistema de forma independiente.

- Fortalecer las buenas prácticas en la gestión de la información, promoviendo la seguridad de los datos, el acceso controlado y la trazabilidad de las operaciones internas.
- Consolidar la transformación digital de la compañía, integrando el sistema web como herramienta institucional para mejorar la eficiencia, coordinación y comunicación en la atención de emergencias.

II.2.4 Contexto

La capacitación fue diseñada y ejecutada en el marco del proyecto “Mejoramiento de la gestión operativa para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz a través de las TIC”, con el propósito de garantizar una correcta apropiación tecnológica por parte del personal institucional.

El proceso formativo se desarrolló en dos fases complementarias:

II.2.4.1 Fase Virtual (Teórica):

En esta etapa se impartieron sesiones en línea orientadas a la presentación general del sistema web, su arquitectura funcional y los objetivos de digitalización institucional. Los participantes conocieron la estructura modular del sistema, la relación entre sus componentes y las buenas prácticas en el uso de credenciales digitales, contraseñas y roles de usuario.

II.2.5 Fase Presencial (Práctica):

Durante esta etapa, el personal participó activamente en la manipulación directa del sistema desde equipos informáticos de la compañía. Se realizaron ejercicios guiados en los diferentes módulos del sistema —como el registro de emergencias, la gestión de inventario y la emisión de credenciales QR— aplicando casos reales y simulaciones operativas.

Este enfoque mixto (virtual-presencial) permitió combinar la teoría con la práctica, asegurando una comprensión integral del sistema y su aplicación inmediata en las tareas diarias de la institución.

La capacitación se organizó considerando los niveles jerárquicos y funciones dentro de la compañía:

- Nivel Ejecutivo: Comandancia y jefaturas, responsables de la toma de decisiones estratégicas y supervisión general del sistema.
- Nivel Operativo: Personal voluntario y operativo, encargado de registrar y mantener actualizada la información en los módulos del sistema.
- Nivel de Apoyo: Personal administrativo o logístico, encargado del uso de los módulos de inventario, donativos y capacitaciones.

Asimismo, se elaboraron guías de usuario, presentaciones digitales y ejercicios prácticos, alineados con la metodología RUP (Rational Unified Process) en su fase de Transición, enfocada en la validación, entrenamiento y despliegue del sistema en un entorno real.

El desarrollo de la capacitación fomentó un entorno colaborativo entre los participantes, promoviendo el intercambio de experiencias, la identificación de mejoras y la adopción de una cultura tecnológica institucional, esencial para la sostenibilidad del sistema en el tiempo.

II.2.6 Propuesta Pedagógica

La propuesta pedagógica de la capacitación se fundamenta en un enfoque expositivo-participativo, que combina la transmisión estructurada de conocimientos teóricos con la aplicación práctica en entornos reales de trabajo. Este enfoque busca que los participantes aprendan haciendo, internalizando el uso del sistema web a través de la experiencia directa y contextualizada en las actividades operativas de la compañía.

El diseño de la capacitación consideró tres ejes metodológicos principales:

II.2.6.1 Enfoque Cognitivo:

Centrado en el desarrollo del conocimiento conceptual sobre el sistema. Durante la etapa virtual, los participantes comprendieron la estructura general del software, los objetivos del proyecto y la función de cada módulo dentro de la gestión operativa institucional.

II.2.6.2 Enfoque Constructivista:

Basado en la construcción del conocimiento a partir de la interacción del usuario con el sistema. Los participantes resolvieron ejercicios prácticos, como el registro de emergencias, la gestión de inventario o la emisión de credenciales QR, favoreciendo la comprensión autónoma del proceso.

II.2.6.3 Enfoque Tecno pedagógico:

Integró las Tecnologías de la Información y Comunicación (TIC) en el proceso de enseñanza, aprovechando recursos digitales como videoconferencias, tutoriales, simulaciones en línea y material audiovisual explicativo. Esto facilitó la comprensión del funcionamiento del sistema y la apropiación tecnológica de los participantes.

La metodología se organizó de manera mixta, combinando sesiones virtuales teóricas con talleres presenciales prácticos. Esta modalidad permitió adaptar el proceso formativo a la disponibilidad del personal operativo, garantizando la participación de todos los miembros de la compañía, incluso aquellos con turnos activos o fuera de la sede.

Cada sesión fue diseñada con una duración promedio de 2 horas, incluyendo momentos de exposición, demostración, práctica y evaluación. Se emplearon estrategias didácticas activas, tales como:

- Demostraciones guiadas por el instructor utilizando el sistema en tiempo real.
- Ejercicios de simulación, replicando casos reales de emergencias, asignaciones de personal y control de inventario.
- Trabajo colaborativo, donde los participantes resolvieron situaciones prácticas en grupos pequeños.
- Retroalimentación inmediata, corrigiendo errores y reforzando el aprendizaje.

El rol del instructor fue actuar como facilitador del aprendizaje, orientando el uso correcto del sistema y fomentando la confianza de los participantes en el manejo de la herramienta tecnológica.

La propuesta pedagógica se alinea con los objetivos del proyecto, contribuyendo al fortalecimiento de las competencias digitales del personal SAR y asegurando una transición efectiva del sistema desde su desarrollo hacia su uso institucional sostenible.

II.2.7 Contenido de la Capacitación

El contenido de la capacitación fue diseñado en función de los diez módulos principales del sistema web desarrollado para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, con el propósito de garantizar un conocimiento integral del software y su aplicación en los procesos operativos, administrativos y de gestión institucional.

El programa se estructuró de manera progresiva y modular, iniciando con los fundamentos básicos del uso del sistema y avanzando hacia la administración de información crítica, la generación de reportes y la seguridad digital.

Cada módulo incluyó una parte teórica —que abarcó los conceptos, objetivos y funciones— y una parte práctica, donde los participantes realizaron ejercicios reales en el sistema, fortaleciendo su comprensión y destreza tecnológica.

II.2.7.1 Módulo 1: Autenticación y Acceso al Sistema

- Introducción a la plataforma web y sus componentes principales.
- Proceso de inicio de sesión, autenticación mediante JWT y seguridad de contraseñas.
- Reconocimiento de la interfaz inicial y menús de navegación.
- Buenas prácticas en la gestión de credenciales y protección de datos personales.

II.2.7.2 Módulo 2: Administración de Usuarios

- Creación, modificación y eliminación de usuarios.
- Asignación de roles y permisos diferenciados según jerarquía institucional.
- Control de acceso y validación de identidad de los usuarios registrados.
- Procedimientos para la recuperación o desbloqueo de cuentas.

II.2.7.3 Módulo 3: Gestión de Roles y Permisos

- Concepto de roles institucionales dentro del sistema.
- Configuración y asignación de permisos a cada módulo del software.
- Restricciones y validaciones para el uso seguro de la información.
- Supervisión del acceso a recursos críticos.

II.2.7.4 Módulo 4: Gestión de Vehículos

- Registro y actualización de información de unidades operativas.
- Asignación de vehículos a emergencias y control de disponibilidad.
- Registro de mantenimiento preventivo y correctivo.
- Reportes automáticos de estado y uso de unidades.

II.2.7.5 Módulo 5: Gestión de Personal

- Registro del personal operativo y voluntario.
- Asignación de grados, clases, funciones y disponibilidad.
- Control de participación en emergencias y capacitaciones.
- Emisión automática de credenciales y actualización de estado.

II.2.7.6 Módulo 6: Gestión de Capacitaciones

- Registro de cursos internos y externos, contenidos y responsables.
- Inscripción de participantes y generación de certificados digitales.
- Consulta de historial de capacitaciones y estadísticas de participación.
- Procedimientos para actualizar nuevos programas de formación.

II.2.7.7 Módulo 7: Gestión de Donativos

- Registro de donaciones materiales y monetarias.
- Asignación de recursos donados a vehículos o emergencias.
- Control de distribución y generación de reportes de uso.
- Visualización de estadísticas por tipo, fuente o periodo.

II.2.7.8 Módulo 8: Gestión de Emergencias

- Registro de emergencias, localización geográfica y descripción del evento.
- Asignación de recursos humanos, materiales y vehiculares.
- Control de estados de emergencia (pendiente, en curso, finalizada).
- Generación de reportes operativos y estadísticos.

II.2.7.9 Módulo 9: Gestión de Inventario

- Registro de materiales, equipos y herramientas institucionales.
- Control de stock, movimientos de entrada/salida y alertas de niveles bajos.
- Integración con módulos de donativos y emergencias.
- Reportes detallados de inventario y seguimiento de existencias.

II.2.7.10 Módulo 10: Gestión de Credenciales

- Generación de credenciales digitales con código QR único.
- Proceso de verificación pública de credenciales mediante escaneo.
- Control de vigencia, suspensión y revocación de credenciales.
- Mantenimiento de historial institucional de acreditaciones.

Cada módulo fue acompañado de material audiovisual, manual de usuario, diapositivas y guías prácticas, los cuales fueron elaborados como parte de los entregables del proyecto. El contenido fue validado con la Comandancia y la Jefatura de la 2ª Cía. Santa Bárbara SAR, garantizando su pertinencia, aplicabilidad y alineación con las operaciones reales de la institución.

II.2.8 Plan de Clases

La capacitación fue planificada bajo una modalidad mixta, combinando dos jornadas virtuales teóricas y una jornada presencial práctica. Esta estructura permitió abordar de manera progresiva los contenidos del sistema web, garantizando la comprensión conceptual antes de la aplicación operativa en campo.

Cada sesión se desarrolló siguiendo el formato metodológico expositivo-participativo, con el apoyo de material audiovisual, guías de usuario, ejercicios prácticos y asistencia técnica en tiempo real.

II.2.8.1 Fase Virtual (Sesiones Teóricas)

Duración: 2 días consecutivos

Horario: de 19:00 a 22:00

Modalidad: En línea (videoconferencia + material digital compartido)

Objetivo general de la fase virtual:

Introducir al personal en los fundamentos del sistema web, su estructura modular, navegación, y las buenas prácticas de gestión digital dentro del entorno institucional.

II.2.8.1.1 Día 1 – Introducción y configuración inicial del sistema

Duración: 3 horas

Contenidos:

Presentación general del sistema y objetivos del proyecto.

Proceso de autenticación, gestión de usuarios y roles.

Introducción al panel principal y estructura de menús.

Ejemplo guiado de uso: creación y gestión de un usuario operativo.

Actividades:

Demostración práctica del ingreso al sistema.

Ejercicio de registro y edición de un usuario de prueba.

Debate guiado sobre la importancia de la trazabilidad digital.

Material de apoyo:

Manual digital de usuario.

Video tutorial: navegación inicial y Login seguro.

II.2.8.1.2 Día 2 – Gestión operativa de módulos institucionales

Duración: 3 horas

Contenidos:

Introducción a los módulos operativos: emergencias, personal, vehículos e inventario.

Integración entre módulos: ejemplo de asignación de recursos a una emergencia.

Módulos administrativos: capacitaciones, donativos y credenciales digitales.

Seguridad y respaldo de la información.

Actividades:

Simulación guiada de registro de emergencia.

Ejercicio práctico: control de inventario y generación de reportes.

Ronda de preguntas y retroalimentación grupal.

Material de apoyo:

Presentaciones digitales.

Guía PDF de ejercicios prácticos.

II.2.8.2 Fase Presencial (Sesión Práctica Intensiva)

Duración: 1 día

Horario: 08:00 a 18:00 (con pausas de descanso y almuerzo)

Modalidad: Presencial – Aula tecnológica / Sala de informática de la Compañía

Objetivo general de la fase presencial:

Consolidar las competencias adquiridas en la fase virtual mediante la ejecución práctica de los procesos operativos reales dentro del sistema web institucional.

- **Bloque 1 – 08:00 a 10:30**

Tema: Inicio de sesión, roles y seguridad de acceso.

Actividad:

Cada participante ingresa al sistema con su usuario asignado. Se realiza una práctica supervisada de creación de roles y gestión de accesos.

- **Bloque 2 – 10:45 a 12:30**

Tema: Gestión de emergencias y asignación de recursos.

Actividad: Simulación completa de una emergencia (registro, asignación de personal, vehículos e inventario).

Evaluación práctica: Validación de flujo correcto del proceso.

- **Pausa de almuerzo – 12:30 a 14:00**

- **Bloque 3 – 14:00 a 16:00**

Tema: Gestión administrativa y documentación.

Actividades:

Registro de capacitaciones.

Emisión de certificados digitales.

Generación y verificación de credenciales QR.

Evaluación: Revisión individual del cumplimiento de pasos del proceso.

- **Bloque 4 – 16:15 a 18:00**

Tema: Cierre, reportes y retroalimentación.

Actividades:

Generación de reportes globales (emergencias, inventario, personal).

Retroalimentación grupal sobre desempeño.

Encuesta de satisfacción y registro de observaciones para mejoras futuras.

La fase presencial finalizó con la entrega simbólica de certificados de participación, validando que el personal capacitado domina los procedimientos del sistema y puede aplicarlos en su labor operativa cotidiana.

II.2.9 Resultados Esperados

La capacitación al personal de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia– La Paz tiene como propósito principal garantizar la adopción efectiva y sostenida del sistema web de gestión operativa, fortaleciendo las competencias digitales del personal e incrementando la eficiencia en los procesos internos de la institución.

Los resultados esperados se estructuran en tres niveles: técnico, operativo e institucional, asegurando un impacto integral en la organización.

II.2.9.1 Resultados Técnicos

- Dominio funcional del sistema web: Los participantes serán capaces de utilizar correctamente los diez módulos principales (autenticación, usuarios, roles, vehículos, personal, capacitaciones, donativos, emergencias, inventario y credenciales).

- Manejo de información digital segura: El personal comprenderá la importancia del registro, almacenamiento y protección de datos en una plataforma institucional.
- Uso autónomo del sistema: Los usuarios podrán resolver incidencias básicas, realizar configuraciones menores y gestionar sus módulos sin requerir asistencia técnica constante.
- Reducción de errores operativos: Gracias a la práctica supervisada, se espera una disminución significativa en los errores de registro y duplicación de información.

II.2.9.2 Resultados Operativos

- Optimización de procesos administrativos y de campo: La capacitación permitirá una ejecución más ágil de tareas como el registro de emergencias, asignación de personal o control de inventario.
- Estandarización de la gestión interna: Se logrará un manejo uniforme de la información en todos los departamentos, garantizando coherencia y trazabilidad institucional.
- Generación de reportes automáticos: El personal capacitado podrá generar y analizar reportes en tiempo real para la toma de decisiones, sin depender de registros manuales.
- Incremento de la transparencia: Las operaciones del sistema quedarán registradas mediante auditorías y bitácoras, fortaleciendo la rendición de cuentas ante la comunidad y las autoridades.

II.2.9.3 Resultados Institucionales

- Fortalecimiento de la capacidad institucional: La capacitación impulsará una cultura digital dentro de la compañía, promoviendo el uso responsable de las TIC en la gestión operativa.
- Autonomía tecnológica: La 2ª Cía. Santa Bárbara SAR reducirá su dependencia de soporte externo para la gestión de su información, consolidando su capacidad de autogestión digital.
- Replicabilidad del modelo: El conocimiento adquirido permitirá que el personal capacitado actúe como multiplicador, entrenando a nuevos miembros y posibilitando la adopción del sistema en otras compañías SAR.
- Mejora en la respuesta ante emergencias: La integración del sistema en la operativa diaria optimizará la asignación de recursos humanos y materiales, permitiendo una reacción más rápida y coordinada frente a emergencias reales.

En síntesis, se espera que la capacitación no solo contribuya a dominar la herramienta tecnológica, sino que también transforme la gestión institucional de la compañía, consolidando un modelo moderno, eficiente y transparente de administración operativa, sustentado en la transformación digital y el uso de las TIC.

II.2.10 Medios de Verificación

Los medios de verificación son los instrumentos y evidencias utilizadas para comprobar que la capacitación fue efectivamente ejecutada, que los objetivos planteados fueron alcanzados y que los participantes adquirieron las competencias necesarias para operar el sistema web institucional.

Estos medios permiten evaluar el proceso formativo tanto en su dimensión técnica (aprendizaje y desempeño), como en su dimensión operativa (aplicación práctica y mejora institucional).

II.2.10.1 Documentación y registros administrativos

- Listas de asistencia digital y presencial: evidencian la participación activa de los miembros de la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia – La Paz durante las sesiones virtuales y la jornada presencial.
- Acta de inicio y cierre de capacitación: firmadas por la comandancia, instructores y participantes, validan formalmente la realización del evento.
- Cronograma oficial de actividades: detalla los horarios, contenidos, responsables y modalidad de cada sesión, asegurando trazabilidad del proceso formativo.
- Reportes de conectividad virtual: generados por la plataforma de videoconferencia, registran la participación en las sesiones teóricas en línea.

II.2.10.2 Instrumentos de evaluación

- Evaluaciones diagnósticas y finales: aplicadas antes y después de la capacitación para medir el nivel de conocimiento y las competencias adquiridas por los participantes.
- Pruebas prácticas en el sistema: cada participante ejecutó tareas reales (registro de emergencias, generación de credenciales, actualización de inventario, etc.), evaluadas mediante rúbricas de desempeño.
- Encuestas de satisfacción: utilizadas para recopilar opiniones sobre la metodología, claridad de los contenidos y utilidad del sistema en la labor diaria.

II.2.10.3 Productos Generados Durante la Capacitación

- Reportes institucionales reales: creados por los participantes durante las prácticas de los módulos de inventario, emergencias y donativos, utilizados como evidencia del dominio funcional del sistema.
- Credenciales y certificados generados: documentos digitales elaborados por el propio personal durante la capacitación, que demuestran el uso efectivo de los módulos de emisión automatizada.
- Bitácora de actividades: registro detallado del desarrollo de cada sesión, con observaciones sobre desempeño y participación.

II.2.10.4 Documentación Técnica Complementaria

- Manual de usuario del sistema web: entregado digitalmente como material de referencia para consultas posteriores.
- Guía práctica de capacitación: documento que incluye el contenido teórico, ejemplos de ejercicios y procedimientos paso a paso para cada módulo.

- Encuesta de satisfacción: instrumento aplicado al finalizar la capacitación para recoger la opinión de los participantes respecto al contenido, metodología y utilidad del sistema.

II.2.10.5 Evidencias Institucionales Posteriores

- Informe de validación del sistema: emitido por la Comandancia y Jefatura, confirmando la correcta implementación del sistema y la capacidad del personal para operarlo de manera autónoma.
- Registros de uso real del sistema: estadísticas de acceso, operaciones ejecutadas y reportes generados durante las semanas posteriores a la capacitación, que demuestran la adopción efectiva de la herramienta.
- Carta de conformidad: documento firmado por la institución, que certifica la recepción del sistema, la capacitación del personal y la operatividad comprobada de los módulos.

II.2.10.6 Evidencia Gráfica del Proceso de Capacitación

Como respaldo visual del desarrollo de la capacitación, se adjuntarán imágenes y capturas de pantalla correspondientes a las sesiones virtuales y presenciales. Estas evidencias muestran la participación activa del personal, la interacción con el sistema web en tiempo real y las prácticas realizadas en los diferentes módulos (gestión de emergencias, inventario, credenciales, entre otros). El material fotográfico y digital servirá como complemento documental que respalda la efectividad y ejecución del proceso formativo, así como la integración del sistema dentro de la dinámica operativa de la compañía.

En conjunto, estos medios de verificación garantizan la trazabilidad del proceso de formación, la validez de los resultados alcanzados y la sostenibilidad del proyecto tecnológico dentro de la institución, demostrando que la 2ª Cía. de Bomberos Santa Bárbara SAR Bolivia cuenta con el conocimiento y las herramientas necesarias para operar plenamente el sistema desarrollado.

II.2.11 Conclusiones

La capacitación desarrollada para el personal de la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz representó una etapa fundamental dentro del proceso de implementación del sistema web de gestión operativa.

A través de un enfoque mixto (virtual y presencial), se logró no solo transmitir conocimientos teóricos, sino también consolidar la competencia práctica en el uso de la herramienta tecnológica que ahora forma parte integral de la gestión institucional.

Durante las sesiones virtuales, los participantes adquirieron las bases conceptuales sobre el funcionamiento general del sistema, la importancia de la seguridad de la información, el uso responsable de las credenciales digitales y los principios de trazabilidad operativa. Posteriormente, en la jornada presencial, se realizó la aplicación práctica en entorno real, donde el personal utilizó el sistema para registrar emergencias, asignar recursos, generar reportes y administrar inventarios, reforzando la comprensión de cada módulo a partir de la experiencia directa.

El proceso permitió evidenciar una elevada predisposición y participación activa del personal, lo cual demuestra el compromiso institucional con la modernización digital y la mejora continua de sus procesos internos.

El equipo de capacitación constató que, tras la formación, los participantes alcanzaron un dominio operativo satisfactorio del sistema web, pudiendo ejecutar de manera autónoma las funciones principales y comprender las ventajas de la automatización frente a los procesos manuales previos.

Asimismo, la capacitación fomentó una cultura organizacional orientada al uso de las TIC, fortaleciendo la transparencia, la eficiencia y la capacidad de respuesta de la institución ante emergencias y gestiones administrativas.

Se generó además un núcleo multiplicador de conocimiento, compuesto por los usuarios más experimentados, quienes podrán capacitar a nuevos miembros o reforzar el aprendizaje continuo dentro de la compañía.

En términos generales, la capacitación cumplió con los objetivos planteados, logrando:

- La apropiación efectiva del sistema web por parte del personal operativo y administrativo.
- La mejora sustancial de la gestión interna mediante la digitalización de procesos clave.
- La creación de una base sólida para futuras actualizaciones tecnológicas y replicación del modelo en otras unidades SAR.

Finalmente, puede afirmarse que la capacitación no solo consolidó el éxito técnico del proyecto, sino que también fortaleció la autonomía institucional y el liderazgo tecnológico de la 2ª Cía. Santa Bárbara SAR Bolivia – La Paz, marcando un precedente en la transformación digital de los cuerpos de bomberos voluntarios del país.

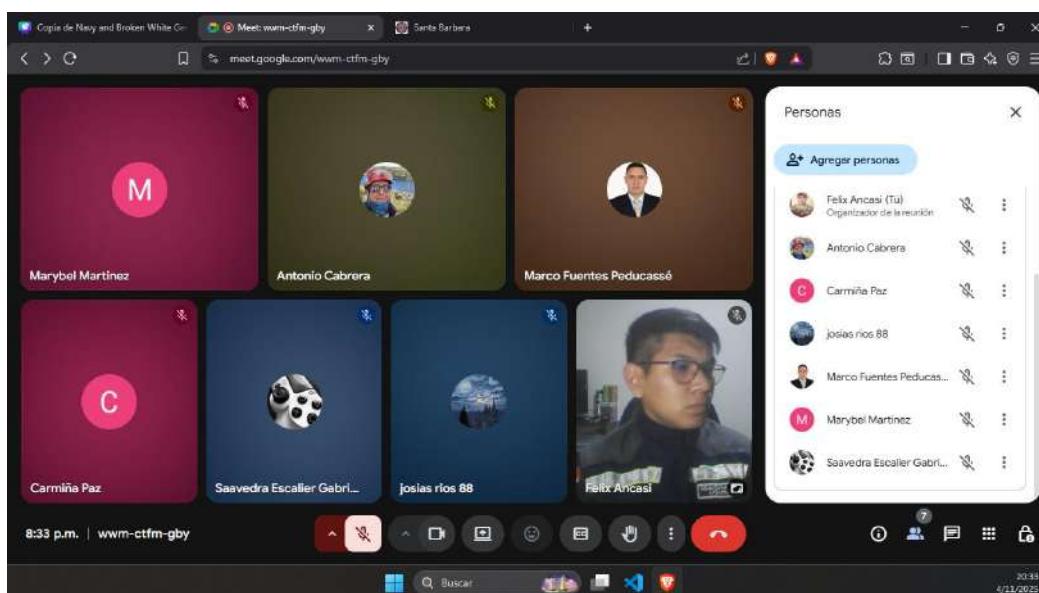


Figura 239. Foto de Capacitación al Personal de la 2ª Cía. Santa Barbara SAR Bolivia-La Paz
Fuente: Elaboración Propia.

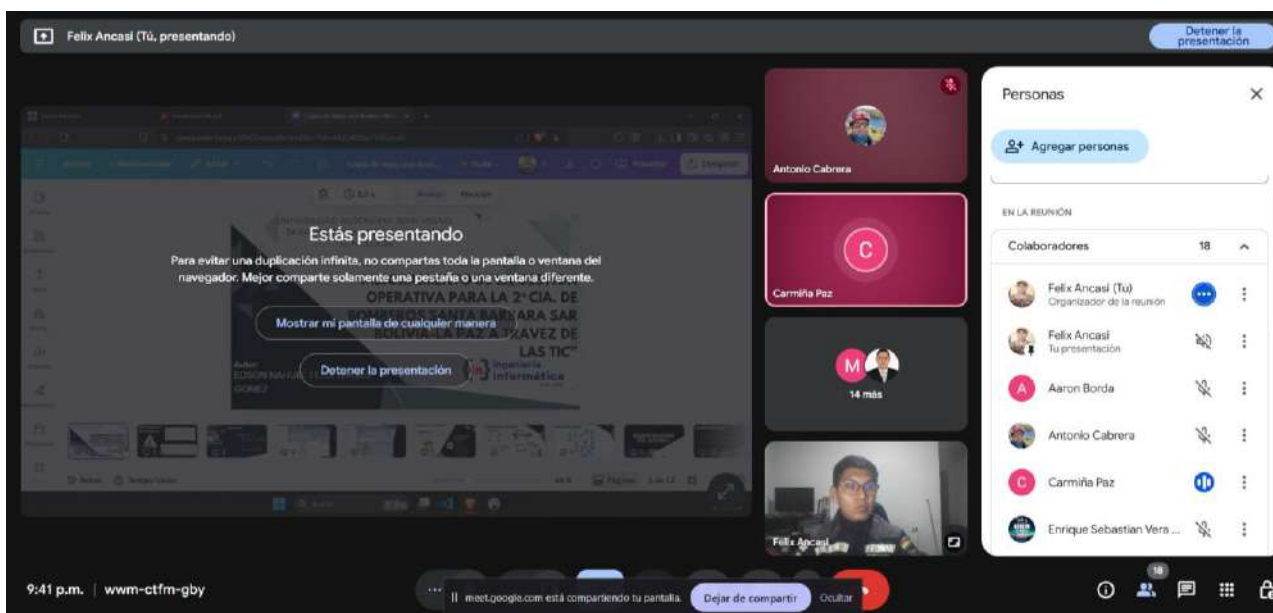


Figura 240. Foto de Capacitación al Personal de la 2ª Cía. Santa Barbara SAR Bolivia-La Paz
Presentación de Diapositivas
Fuente: Elaboración Propia.



Figura 241. Foto de Capacitación al Personal de la 2ª Cía. Santa Barbara SAR Bolivia-La Paz
Planteamiento del Problema
Fuente: Elaboración Propia.

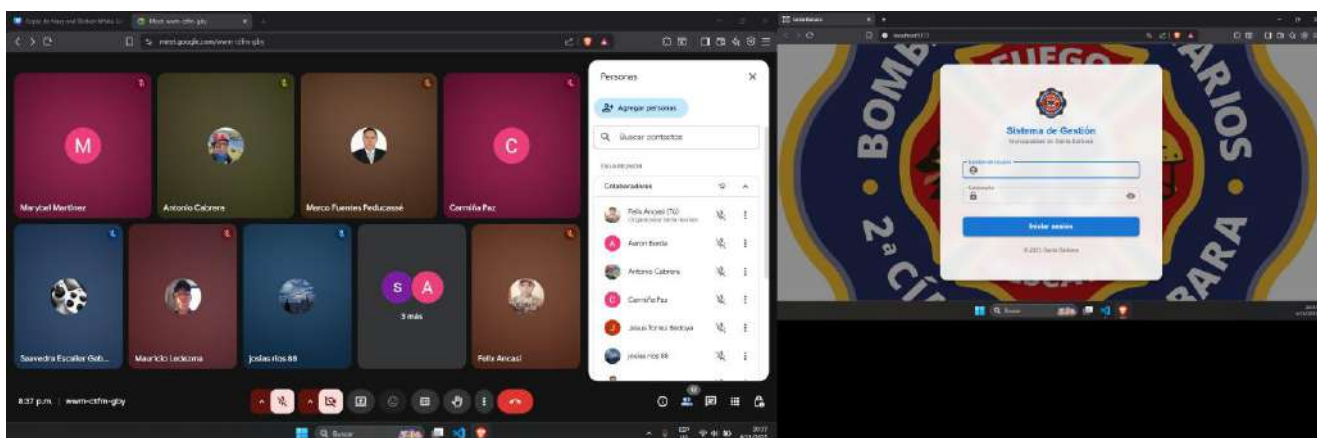


*Figura 242. Foto de Capacitación al Personal de la 2ª Cía. Santa Barbara SAR Bolivia-La Paz
presentación del Sistema
Fuente: Elaboración Propia.*

CAPÍTULO III

CONCLUSIONES Y

RECOMENDACIONES

III Conclusiones y recomendaciones

III.1 Conclusiones

El presente proyecto titulado “Mejoramiento de la gestión operativa para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz a través de las TIC”, constituye un aporte significativo al fortalecimiento institucional mediante la aplicación de herramientas tecnológicas orientadas a la modernización de los procesos administrativos y operativos.

El desarrollo del sistema web de gestión operativa permitió automatizar los principales procesos internos de la compañía, tales como la gestión de emergencias, personal, inventario, donativos, capacitaciones y credenciales digitales, consolidando en una sola plataforma la información que anteriormente se manejaba de forma dispersa y manual.

La implementación de este sistema ha demostrado su efectividad al optimizar la coordinación de recursos humanos y materiales durante las operaciones de respuesta, garantizando un flujo de información más rápido, preciso y seguro. Gracias a ello, la institución cuenta ahora con una herramienta que mejora la toma de decisiones, la trazabilidad de sus operaciones y la transparencia de su gestión.

Asimismo, la adopción de la metodología RUP (Rational Unified Process) permitió estructurar el desarrollo en fases iterativas y controladas, asegurando la validación constante de los requerimientos y la producción de artefactos documentales conforme a la norma IEEE 830. Este enfoque metodológico aportó rigor técnico, trazabilidad y control de calidad en todas las etapas del proyecto.

La aplicación de tecnologías de código abierto como React, Node.js, Express.js, Sequelize y PostgreSQL permitió construir un sistema escalable, seguro y de fácil mantenimiento, reduciendo los costos de desarrollo y garantizando compatibilidad multiplataforma.

Por otro lado, el componente de capacitación dirigido al personal operativo, jefaturas y administración de la 2ª Cía. Santa Bárbara, facilitó la apropiación del sistema y el fortalecimiento de competencias digitales entre los miembros de la institución. Gracias a estas sesiones impartidas de forma virtual y presencial los usuarios adquirieron los conocimientos necesarios para operar el sistema, gestionar la información y aprovechar las funcionalidades implementadas de manera eficiente.

En conjunto, el proyecto logró cumplir los objetivos propuestos, integrando tecnología, metodología y formación en un mismo marco de transformación institucional. Este esfuerzo contribuye directamente a la profesionalización del cuerpo de bomberos voluntarios, promoviendo una cultura digital que mejora la eficacia en la gestión de emergencias y la confianza ciudadana en su labor.

III.2 Recomendaciones

Con base en los resultados obtenidos durante el desarrollo e implementación del sistema web de gestión operativa para la 2ª Compañía de Bomberos Santa Bárbara SAR Bolivia – La Paz, se formulan las siguientes recomendaciones orientadas a la sostenibilidad, mejora continua y apropiación institucional del sistema:

III.2.1 Recomendaciones técnicas

- Se recomienda mantener actualizada la base de datos y realizar copias de seguridad periódicas (backups) con el fin de evitar la pérdida de información ante posibles fallos del sistema o cortes de energía.
- Implementar mecanismos de auditoría automática y supervisión de logs para garantizar la trazabilidad de las operaciones realizadas por cada usuario.
- Asegurar que el servidor y el entorno de ejecución (Node.js, PostgreSQL) se mantengan con las versiones estables más recientes, incorporando parches de seguridad y actualizaciones críticas.
- Se sugiere habilitar un sistema de respaldo remoto o en la nube, que permita restaurar la información en caso de desastres físicos o ciberataques.
- Continuar con el uso de herramientas de código abierto para futuras mejoras, promoviendo la independencia tecnológica y la sostenibilidad económica de la institución.

III.2.2 Recomendaciones operativas

- Designar un responsable técnico interno dentro de la compañía, encargado de administrar el sistema, gestionar usuarios, validar la información registrada y coordinar los respaldos.
- Estandarizar los procedimientos internos para el registro de emergencias, inventario, capacitaciones y credenciales, asegurando que toda la información se incorpore correctamente al sistema.
- Fomentar el uso cotidiano del sistema en todas las áreas (comandancia, logística, operaciones e instrucción), evitando el retorno a los registros manuales.
- Se recomienda realizar evaluaciones periódicas de desempeño del sistema mediante indicadores de uso, tiempos de respuesta y nivel de satisfacción de los usuarios.

III.2.3 Recomendaciones de capacitación

- Mantener un programa de formación continua, tanto para el personal nuevo como para los miembros antiguos, asegurando que todos los usuarios conozcan las funcionalidades actualizadas del sistema.
- Desarrollar sesiones prácticas semestrales donde se refuercen los módulos clave: emergencias, inventario, donativos, capacitaciones y credenciales.

- Promover la retroalimentación de los usuarios para identificar mejoras y requerimientos que puedan incorporarse en futuras versiones del sistema.
- Elaborar manuales actualizados de usuario y administración, tanto en formato digital como impreso, para garantizar un aprendizaje autónomo.

III.2.4 Recomendaciones institucionales

- Incluir el uso del sistema dentro del reglamento operativo interno de la compañía, formalizando su adopción como herramienta oficial de gestión.
- Gestionar la posibilidad de extender el uso del sistema a otras compañías SAR y unidades de bomberos voluntarios del país, fortaleciendo la cooperación interinstitucional.
- Promover la actualización tecnológica progresiva de los equipos de cómputo y la infraestructura de red, garantizando compatibilidad, velocidad y acceso confiable.