TRABAJO DE GRADO

Proyecto de Grado

SeguriLog: Sistema de gestión y análisis de incidentes en unidades residenciales en Medellín

Corporación Universitaria Remington.

Facultad de Ingeniería

Tecnología en Desarrollo de Software

Laura Daniela Prado Pérez

Docente: Mónica María Córdoba Castrillón

Medellín, Colombia

2026

Tabla de contenido

Lista de tablas

Lista de figuras

Resumen

Este trabajo presenta el desarrollo de SeguriLog, una plataforma web creada para centralizar el registro y el análisis de incidentes de seguridad en conjuntos residenciales de Medellín. El punto de partida fue una realidad concreta: la mayoría de estos espacios aún depende de cuadernos físicos u hojas de cálculo sin estructura para documentar sus novedades, lo que impide convertir esa información en insumo útil para quienes deben tomar decisiones.

La investigación adoptó un enfoque cuantitativo, de tipo aplicado y con alcance descriptivo. Se recurrió a la observación, la revisión documental y una encuesta dirigida a residentes. Los hallazgos mostraron que la comunidad está dispuesta a adoptar herramientas digitales y que valora especialmente la posibilidad de consultar gráficos e indicadores, no solo registrar datos.

Con base en esos resultados, se diseñó e implementó una aplicación web organizada en capas, construida sobre ASP.NET Core, SQL Server y Power BI. El sistema cubre el ciclo completo: desde el registro inicial del incidente hasta su consulta, exportación y análisis visual mediante dashboards interactivos.

Los resultados obtenidos muestran que SeguriLog logra unificar la información dispersa, hace visible la trazabilidad de cada evento y revela patrones que antes pasaban desapercibidos. Esto le entrega a las empresas de vigilancia privada una herramienta concreta para decidir con criterio y anticiparse a los riesgos.

En definitiva, SeguriLog demuestra que es posible profesionalizar la gestión de la seguridad residencial usando tecnología accesible, y que hacerlo tiene un impacto directo en la eficiencia del servicio y la calidad de vida de quienes habitan esos espacios.

Palabras clave

Palabras clave: análisis de datos, seguridad privada, incidentes, inteligencia de negocios, visualización de datos, sistema de información, toma de decisiones basada en datos.

Abstract

This paper describes the development of SeguriLog, a web platform built to centralize the recording and analysis of security incidents in residential complexes in Medellín, Colombia. The starting point was a persistent problem: most of these settings still rely on notebooks or unstructured spreadsheets to document security events, which prevents turning that information into actionable insight. The study followed a quantitative, applied, and descriptive approach, drawing on observation, documentary review, and a resident survey. Findings revealed strong openness to digital tools and a clear demand for features that go beyond simple registration, including visual analysis through charts and indicators. In response, a layered web application was designed and deployed using ASP.NET Core, SQL Server, and Power BI. The platform covers the full cycle: from incident logging and filtered retrieval to data export and interactive dashboard visualization. Outcomes show that SeguriLog consolidates previously scattered information, improves event traceability, and surfaces patterns that were invisible before, equipping private security companies with a practical tool for data-driven, proactive decision-making.

Keywords: data analysis, private security, incidents, business intelligence, data visualization, information systems, data-driven decision making.

Introducción

Vivir en un conjunto residencial en Medellín implica compartir espacios, servicios e incertidumbres. El crecimiento acelerado de la ciudad ha multiplicado la cantidad de unidades residenciales y con ellas la demanda de seguridad privada. Pero contar con vigilantes en el turno no es garantía suficiente: en muchos conjuntos, los incidentes se siguen acumulando sin que nadie los analice ni los use para mejorar el servicio.

Los tipos de eventos que se repiten son conocidos: ingresos no autorizados, hurtos en zonas comunes, fallas en los controles de acceso, disputas entre vecinos. A todo eso se suma, con frecuencia, una comunicación fragmentada entre el vigilante que lleva el turno y la administración que debería actuar. Según la Secretaría de Seguridad y Convivencia de Medellín (2023), los hurtos en unidades residenciales ocupan un lugar destacado entre los delitos de mayor reporte en la ciudad, lo que indica que las estrategias vigentes todavía no alcanzan.

El fondo del problema está en cómo se maneja la información. Cuadernos físicos, hojas de cálculo improvisadas, columnas sin nombre claro: así se registran hoy los incidentes en buena parte de los conjuntos. El resultado es previsible: datos incompletos, registros perdidos y, sobre todo, la imposibilidad de preguntar cosas tan básicas como en qué zonas se concentran más los eventos o a qué horas son más frecuentes. Sin esa información, actuar con anticipación es casi imposible.

Las herramientas de análisis de datos ofrecen una respuesta práctica a ese vacío. Cuando los incidentes se registran de forma ordenada y se visualizan a través de dashboards, el panorama cambia por completo: en lugar de reaccionar al problema ya ocurrido, es posible identificar patrones, anticipar riesgos y distribuir mejor los recursos.

Con ese objetivo, este proyecto propone una aplicación web para el registro y análisis de incidentes en unidades residenciales de Medellín, conectada con dashboards construidos en Microsoft Power BI. La solución usa SQL Server como motor de base de datos y ASP.NET Core para el backend, y está pensada para ser robusta, escalable y fácil de adoptar. El propósito final es poner información de calidad en manos de quienes tienen que decidir qué hacer con ella.

Justificación

En los conjuntos residenciales de Medellín existe información que se genera todos los días pero que rara vez llega a aprovecharse. Cada vigilante llena un registro durante su turno; ese registro queda guardado en un formato que nadie analiza, que no permite comparaciones y que no se convierte en aprendizaje. Ese es el punto que este proyecto busca transformar.

Pertinencia académica

Desde lo académico, este proyecto pone a prueba de manera integrada los aprendizajes del programa de Tecnología en Análisis de Datos: diseño de bases de datos relacionales, construcción de aplicaciones web, inteligencia de negocios y visualización. No se trata de un ejercicio teórico sobre un caso ficticio, sino de resolver un problema real con herramientas reales, lo que convierte cada decisión técnica en una oportunidad de aprendizaje genuina.

Pertinencia social y práctica

En términos sociales, el impacto es tangible y directo. Una herramienta que organice y analice los incidentes le permite a las empresas de vigilancia privada saber exactamente qué está ocurriendo, cuándo y dónde. Eso se traduce en decisiones más sólidas, en una distribución más eficiente del personal y, en última instancia, en una mejora real en la seguridad de quienes viven en esos espacios.

Pertinencia tecnológica

Aunque la transformación digital ha avanzado en casi todos los sectores, la seguridad privada residencial sigue siendo, en muchos casos, un campo donde dominan las prácticas manuales. Incorporar herramientas como Power BI, SQL Server y ASP.NET Core no es únicamente una decisión técnica: es un paso hacia la profesionalización de un servicio del que

dependen miles de familias en la ciudad. El sistema fue construido con criterios de escalabilidad, por lo que puede crecer y adaptarse a medida que el servicio evolucione.

En su conjunto, este proyecto no se limita a responder a una necesidad puntual: también muestra que el análisis de datos puede generar valor en contextos donde todavía no se ha explorado su potencial.

Marco teórico

Análisis de datos en entornos organizacionales

El análisis de datos en las organizaciones dejó hace tiempo de ser una práctica reservada para grandes empresas con equipos especializados. Hoy, instituciones de todos los tamaños y sectores reconocen que la diferencia entre decidir bien y decidir mal pasa, en gran medida, por la calidad de la información disponible. No alcanza con recolectar datos: hay que limpiarlos, estructurarlos, interpretarlos y presentarlos de una forma que resulte útil para quien debe actuar.

Provost y Fawcett (2018) sostienen que el análisis de datos hace visibles patrones que a simple vista no se detectan, y que el conocimiento así generado tiene respaldo en evidencia empírica. En la misma dirección, el McKinsey Global Institute (2021) documentó que las organizaciones que utilizan datos de forma sistemática superan consistentemente en productividad y competitividad a las que no lo hacen. Aunque estos hallazgos provienen del mundo empresarial, sus implicaciones son igualmente válidas para la seguridad privada.

En el contexto de los conjuntos residenciales, la situación es clara: los incidentes se registran a diario y contienen información con valor real, pero esa información nunca llega a procesarse. Los patrones existen —en los horarios, en los tipos de eventos, en las zonas más afectadas— pero permanecen ocultos porque no hay un sistema que los haga visibles. Este

proyecto surge precisamente de ese desfase: transformar un registro disperso en conocimiento útil para mejorar el servicio.

Tipos de análisis de datos

Existen cuatro formas de abordar el análisis de datos según el tipo de pregunta que se quiera responder. El análisis descriptivo se ocupa de lo que ya ocurrió, a partir del historial disponible. El diagnóstico profundiza en las causas: por qué pasó lo que pasó. El predictivo anticipa lo que podría ocurrir bajo ciertas condiciones. Y el prescriptivo va un paso más allá: no solo anticipa, sino que sugiere qué hacer ante ese escenario.

Sharda et al. (2019) explican que estos niveles de análisis permiten que una organización avance desde una lectura básica de sus datos hasta un uso estratégico de los mismos. En este proyecto, el foco está en los dos primeros: describir cómo se comportan los incidentes e identificar qué factores están detrás de su ocurrencia son los pasos necesarios para empezar a gestionar la seguridad con información real.

Inteligencia de negocios

La inteligencia de negocios —o Business Intelligence (BI)— engloba un conjunto de herramientas y procesos orientados a convertir datos en información que sirva para tomar decisiones. No es una tecnología puntual, sino un enfoque que articula la captura de datos, su integración desde múltiples fuentes, el análisis mediante métodos estadísticos y la presentación a través de visualizaciones comprensibles para el usuario.

Según Sharda et al. (2019), el BI fortalece el desempeño organizacional al poner a disposición de los tomadores de decisiones información actualizada, confiable y presentada de forma clara. Herramientas como Microsoft Power BI han bajado significativamente la barrera de

entrada a estas soluciones, permitiendo construir dashboards interactivos sin necesidad de conocimientos avanzados de programación.

En este proyecto, la inteligencia de negocios actúa como el eje articulador del análisis: los incidentes registrados en la aplicación web se convierten en el insumo que alimenta los dashboards de Power BI, donde los datos adquieren sentido a través de gráficas, indicadores clave y filtros que facilitan la interpretación por parte de supervisores y administradores.

Componentes de un sistema de Business Intelligence

Para que un sistema de BI opere correctamente, varios elementos deben funcionar de manera coordinada. La fuente de datos es el origen de todo: en este caso, el formulario web donde los vigilantes documentan los incidentes. Los procesos de transformación se encargan de preparar esa información —limpiándola y estructurándola— antes de que llegue al análisis. El almacenamiento, implementado aquí con SQL Server, organiza los datos de forma que puedan consultarse con eficiencia. Y las herramientas de visualización, representadas por Power BI, son las que convierten todo ese trabajo previo en algo comprensible de un vistazo.

Kimball y Ross (2013) señalan que la solidez de un sistema de BI depende de que todos sus componentes estén bien articulados entre sí. Si los datos se capturan mal, si la base de datos dificulta las consultas o si los dashboards no responden a las preguntas correctas, el sistema pierde utilidad. Por eso, en el desarrollo de SeguriLog se prestó atención específica a cada uno de esos eslabones.

Visualización de datos y dashboards

Uno de los obstáculos más frecuentes para que el análisis de datos llegue a quienes más lo necesitan es la forma en que se presentan los resultados. Las tablas repletas de números, los informes extensos y los reportes técnicos tienen valor para quien los elabora, pero pocas veces logran

comunicar algo claro a quienes deben actuar. La visualización de datos nació precisamente para cerrar esa brecha.

Knaflic (2015) plantea que el valor de una visualización no está en su apariencia, sino en su capacidad de comunicar con claridad, precisión y sin elementos que distraigan. Few (2013) complementa esa visión advirtiendo que un dashboard mal diseñado —uno que intenta mostrar todo a la vez— puede ser tan inútil como no tener información: la sobrecarga visual paraliza la toma de decisiones igual que la ausencia de datos.

García y López (2021) documentaron, en el ámbito de la seguridad privada, que las organizaciones que implementan dashboards de monitoreo logran acortar sus tiempos de respuesta ante incidentes y asignar mejor sus recursos, porque los supervisores pueden ver en tiempo real dónde y cuándo se están concentrando los eventos. Ese es exactamente el tipo de impacto que se busca con los dashboards de SeguriLog.

El papel de los dashboards en la toma de decisiones

Un dashboard bien construido concentra la información más relevante en una sola vista y le permite al usuario tomar decisiones sin tener que procesar manualmente grandes volúmenes de datos. Gartner (2023) señala que las organizaciones que adoptan este tipo de herramientas mejoran su capacidad analítica y reducen el tiempo que dedican a generar reportes manuales.

En la seguridad residencial eso se traduce en algo muy concreto: poder ver de un vistazo cuántos incidentes ocurrieron el mes pasado, en qué zonas del conjunto se concentraron, a qué horas fueron más frecuentes y cuántos permanecen sin resolver. Esa información no resuelve los problemas por sí sola, pero le da a quien debe actuar un punto de partida sólido para tomar mejores decisiones.

Sistemas de información en la gestión organizacional

Un sistema de información va mucho más allá del software que lo soporta. Es la articulación entre personas, procesos y tecnología para que la información correcta llegue a quien la necesita, cuando la necesita. Así entendido, resulta lógico que su implementación tenga un efecto directo en la eficiencia de cualquier organización.

Laudon y Laudon (2016) sostienen que estos sistemas son fundamentales para mejorar la eficiencia organizacional porque no solo automatizan tareas, sino que eliminan los errores que surgen cuando los procesos dependen exclusivamente del criterio humano en cada paso. O'Brien y Marakas (2019) añaden que un sistema bien implementado libera al personal de las tareas rutinarias y le permite concentrarse en lo que realmente aporta valor.

En SeguriLog, el sistema reemplaza el registro manual por una aplicación estructurada que garantiza que cada evento quede almacenado de forma completa, ordenada y disponible para su análisis posterior. Puede parecer un cambio técnico menor, pero en la práctica transforma la manera en que se gestiona la información de seguridad dentro del conjunto residencial.

Bases de datos relacionales

El modelo relacional que Codd (1970) formalizó hace más de cinco décadas sentó un principio que sigue vigente: organizar la información en tablas vinculadas entre sí mediante identificadores comunes, de modo que los datos sean coherentes e íntegros por diseño. Aunque las tecnologías han evolucionado enormemente desde entonces, este modelo sigue siendo el estándar dominante para el almacenamiento de información estructurada en entornos empresariales.

La base de datos de SeguriLog, diseñada en SQL Server, está conformada por siete tablas normalizadas que almacenan información sobre usuarios, unidades residenciales, tipos de incidente, ubicaciones y los incidentes propiamente dichos. Además, se construyó la vista

VW_Incidentes_PowerBI, una consulta predefinida que presenta los datos en un formato plano con columnas de tiempo calculadas —año, mes, semana, día y hora—, lo que permite conectar directamente con Power BI sin transformaciones adicionales.

Arquitectura de software

La arquitectura de un sistema de software determina cómo se organizan sus piezas y cómo se comunican entre ellas. Es una decisión que se toma antes de escribir la primera línea de código y que tiene consecuencias duraderas: una arquitectura bien concebida facilita el mantenimiento, permite incorporar nuevas funcionalidades y reduce el riesgo de que un cambio en un módulo rompa otros.

Fowler (2002) describe la arquitectura en capas como uno de los patrones más adecuados para sistemas empresariales, justamente porque separa responsabilidades con claridad: una capa gestiona la interfaz con el usuario, otra contiene las reglas del negocio y una tercera se ocupa del acceso a los datos. Cada capa cumple su función sin necesitar conocer los detalles internos de las demás.

En SeguriLog esa separación se materializó con ASP.NET Core 8 en el servidor, donde los controladores REST, los servicios de negocio y el acceso a datos mediante Entity Framework Core están claramente delimitados. La interfaz —HTML, CSS y JavaScript— corresponde a la capa de presentación, y Power BI opera como una capa externa de análisis y visualización que se conecta directamente a la base de datos.

Toma de decisiones basada en datos

Toda decisión implica algún grado de incertidumbre. Decidir con datos no elimina esa incertidumbre, pero sí la acota de forma concreta. Davenport (2018) argumenta que las organizaciones que incorporan el análisis de datos en sus procesos de decisión obtienen resultados

consistentemente mejores, porque la evidencia permite anticipar consecuencias que de otro modo serían difíciles de prever.

En el ámbito de la seguridad privada, pasar de una gestión reactiva —responder cuando el problema ya ocurrió— a una gestión proactiva —identificar patrones de riesgo y anticiparse— requiere información estructurada. Y esa información ya existe: está en los registros que los vigilantes completan cada día. El problema no es la ausencia de datos; es la falta de un sistema que los haga utilizables.

Los dashboards de SeguriLog son precisamente ese puente: toman registros dispersos y los convierten en indicadores claros que orientan las decisiones diarias de supervisores y administradores, desde la planificación de turnos hasta la identificación de zonas críticas dentro del conjunto.

Problemática en la gestión de incidentes en unidades residenciales

La seguridad en los conjuntos residenciales de Medellín enfrenta un desafío que no es reciente, pero que sigue sin resolverse. El crecimiento sostenido de la ciudad, con más unidades residenciales y mayor densidad poblacional, ha aumentado tanto la demanda del servicio de vigilancia como la complejidad de gestionarlo de forma efectiva.

La CEPAL (2022) señala que en los países latinoamericanos persiste un uso muy limitado de tecnologías de análisis de datos en sectores como la seguridad privada, lo que restringe la capacidad de las organizaciones para fundamentar sus decisiones en información objetiva. Esa limitación se hace especialmente visible cuando los incidentes se anotan en cuadernos o en hojas de cálculo sin estructura, ya que esos formatos imposibilitan los análisis comparativos y el seguimiento de tendencias a lo largo del tiempo.

Los efectos son tangibles: registros incompletos o perdidos, supervisores que no pueden identificar qué zonas o qué horarios concentran más eventos, y las decisiones que terminan apoyándose en la experiencia personal del vigilante de turno en lugar de en datos verificables. SeguriLog apunta directamente a cambiar esa dinámica: centraliza el registro, estructura la información y la pone al alcance de quienes necesitan analizarla para actuar.

Planteamiento del problema

En los conjuntos residenciales de Medellín, el personal de vigilancia privada documenta a diario una amplia variedad de situaciones: intentos de ingreso no autorizado, hurtos en zonas comunes, conflictos entre vecinos, fallas en los sistemas de control de acceso y otras novedades operativas. Toda esa información se genera y se acumula. El problema es lo que ocurre —o más bien, lo que no ocurre— con ella después: los registros terminan en cuadernos físicos o en hojas de cálculo sin estructura, lo que hace casi imposible consultarlos, analizarlos o usarlos para decidir.

El resultado de ese vacío es una gestión esencialmente reactiva: se actúa cuando el problema ya ocurrió, sin posibilidad de anticiparse porque no existe información que permita identificar dónde o cuándo es más probable que ocurra el próximo evento. Las zonas de mayor riesgo, los horarios críticos, los tipos de incidentes que se repiten: todo eso está latente en los registros, pero nunca llega a procesarse.

Esta no es una situación marginal. La Secretaría de Seguridad y Convivencia de Medellín (2023) reporta que los incidentes en conjuntos residenciales representan una fracción significativa de los eventos de seguridad ciudadana en la ciudad, lo que evidencia la dimensión real del problema y la insuficiencia de las estrategias actuales.

¿Cómo desarrollar un sistema que permita registrar y analizar los incidentes en unidades residenciales de Medellín, con el fin de mejorar la toma de decisiones en las empresas de seguridad privada?

Objetivos

Objetivo General

Desarrollar un sistema de registro y análisis de incidentes en unidades residenciales en Medellín, mediante una aplicación web y dashboards, con el fin de que esto sirva de apoyo en la toma de decisiones en empresas de seguridad privada.

Objetivos específicos

- Diseñar una base de datos relacional en Microsoft SQL Server para el almacenamiento estructurado, íntegro y trazable de la información relacionada con los incidentes registrados en unidades residenciales.
- Desarrollar una aplicación web en ASP.NET Core para la gestión de incidentes, incluyendo registro, consulta, edición y control de acceso diferenciado por roles de usuario (Vigilante, Supervisor y Administrador).
- Implementar un proceso de organización y transformación de datos mediante vistas SQL y su integración con Power BI para el análisis de la información registrada.
- Crear dashboards interactivos en Microsoft Power BI para la visualización y el análisis de indicadores clave de los incidentes, con el fin de la identificación de patrones que apoyen la toma de decisiones en las empresas de seguridad privada.

Metodología

Enfoque de la investigación

El proyecto se desarrolló bajo un enfoque cuantitativo, ya que parte de la recolección, el procesamiento y el análisis de datos relacionados con los incidentes registrados en unidades residenciales. Este enfoque permite trabajar con información objetiva y medible, condición indispensable para identificar patrones, detectar tendencias y extraer conclusiones basadas en evidencia.

La elección de este enfoque también es coherente con el propósito central del proyecto: no solo observar la problemática, sino analizarla de forma estructurada para apoyar la toma de decisiones con datos. El análisis cuantitativo convierte la información recolectada en resultados concretos y comprensibles para quienes deben actuar a partir de ellos.

Tipo de investigación

La investigación es de tipo aplicado porque está orientada a resolver una problemática concreta mediante el desarrollo de una herramienta tecnológica: un sistema que mejore el registro y el análisis de incidentes en unidades residenciales.

El estudio tiene además un alcance descriptivo: su foco está en analizar cómo se comportan los incidentes a partir de los datos disponibles. No se pretende intervenir en las condiciones en que ocurren los eventos, sino comprenderlos mejor para generar información útil que oriente las decisiones.

Diseño de la investigación

El diseño de la investigación es no experimental, ya que no se manipulan variables ni se interviene directamente en el entorno donde ocurren los hechos. En cambio, se trabaja con información que ya existe o que se recolecta tal como se presenta en la realidad.

Asimismo, el diseño es de tipo transversal, debido a que los datos se recopilan en un periodo determinado de tiempo, lo que permite realizar un análisis puntual del comportamiento de los incidentes durante ese intervalo.

Población y muestra

La población objeto de estudio está conformada por los incidentes registrados en unidades residenciales que cuentan con servicio de vigilancia privada en la ciudad de Medellín.

Por su parte, la muestra está compuesta por los registros de incidentes recopilados durante un periodo específico, los cuales sirven como base para el análisis de la información.

Adicionalmente, se incluye un grupo de mínimo 20 residentes de unidades residenciales, quienes participan en la aplicación de una encuesta con el fin de evaluar la aceptación del sistema propuesto.

La selección de esta muestra permite obtener tanto datos cuantitativos sobre los incidentes como la percepción de los usuarios frente a la solución planteada.

Técnicas de recolección de información

Para la recolección de la información se utilizan diferentes técnicas que permiten obtener datos desde distintas perspectivas:

- **Observación directa:** se emplea para identificar cómo se llevan actualmente los procesos de registro de incidentes en las unidades residenciales.

- **Revisión documental:** consiste en analizar registros previos y formatos utilizados para documentar incidentes.
- **Registro de información:** se realiza a través del sistema propuesto, permitiendo estructurar los datos de manera organizada.
- **Encuesta:** aplicada a los residentes, con el fin de conocer su percepción sobre el sistema y las funcionalidades que consideran más importantes.

El uso de estas técnicas permite contar con información más completa y confiable para el desarrollo del proyecto.

Instrumento de recolección de información: Encuesta

Como parte del proceso de recolección de información, se diseñó una encuesta estructurada orientada a evaluar la aceptación del sistema propuesto.

La encuesta está compuesta por cinco preguntas cerradas de selección múltiple, lo que facilita el análisis de los resultados al trabajar con datos cuantificables. Este instrumento fue aplicado a una muestra mínima de 20 personas que residen en unidades residenciales.

Las preguntas están enfocadas en conocer la opinión de los usuarios sobre la importancia de contar con herramientas tecnológicas para la gestión de incidentes, así como su disposición a utilizar una aplicación que permita registrar y analizar la información mediante gráficos y reportes. Esto permite validar si la solución propuesta realmente responde a una necesidad del entorno.

Procesamiento y análisis de datos

El procesamiento de los datos se realiza utilizando herramientas tecnológicas que permiten su almacenamiento, organización y posterior análisis.

En una primera etapa, los datos son almacenados en una base de datos gestionada con Microsoft SQL Server, lo que asegura su integridad, organización y disponibilidad. Posteriormente, esta información es transformada y analizada mediante Microsoft Power BI, herramienta que facilita la creación de dashboards interactivos para visualizar los datos de manera clara.

El análisis se enfoca principalmente en identificar aspectos como:

- Frecuencia de incidentes
- Tipos de incidentes
- Horarios en los que ocurren con mayor frecuencia
- Comportamientos o patrones repetitivos

Este proceso permite convertir los datos en información útil para apoyar la toma de decisiones.

Métodos estadísticos utilizados

Para el análisis de la información se emplean métodos estadísticos descriptivos, los cuales permiten organizar y resumir los datos de manera sencilla.

Entre los principales métodos utilizados se encuentran:

- Frecuencias absolutas y relativas
- Tablas de distribución de datos
- Representaciones gráficas (barras, líneas y gráficos circulares)

Estos recursos facilitan la interpretación de los datos y permiten comunicar los resultados de forma clara y comprensible.

Desarrollo del sistema

El sistema se desarrolla como una aplicación web orientada al registro y gestión de incidentes en unidades residenciales.

Para su implementación se utiliza ASP.NET como tecnología principal, junto con Microsoft SQL Server para la gestión de la base de datos. La aplicación se organiza bajo una arquitectura en capas, lo que permite separar la lógica del sistema, la interfaz de usuario y el acceso a los datos. Este tipo de estructura facilita tanto el mantenimiento como la escalabilidad del sistema, permitiendo realizar mejoras futuras sin afectar su funcionamiento general.

Fases del proyecto

El desarrollo del proyecto se lleva a cabo a través de varias fases que permiten organizar el proceso de manera clara:

1. **Análisis:** identificación de la problemática y definición de requerimientos.
2. **Diseño:** estructuración de la arquitectura del sistema y la base de datos.
3. **Desarrollo:** construcción de la aplicación web.
4. **Pruebas:** verificación del correcto funcionamiento del sistema.
5. **Implementación:** puesta en marcha de la solución desarrollada.
6. **Evaluación:** análisis de resultados y validación mediante la encuesta aplicada.

Estas fases permiten llevar un control ordenado del proyecto y asegurar que se cumplan los objetivos planteados.

Encuesta

A continuación, se detallarán las preguntas que se formularon para recopilar la información necesaria para este proyecto.

Link de la encuesta:

https://docs.google.com/forms/d/e/1FAIpQLSe7rJuJRhVIshx7QfJW0o32hr5qG-D1Vc8RDMllKas1ndcRBA/viewform?usp=header

Figura 1 Descripción de la encuesta

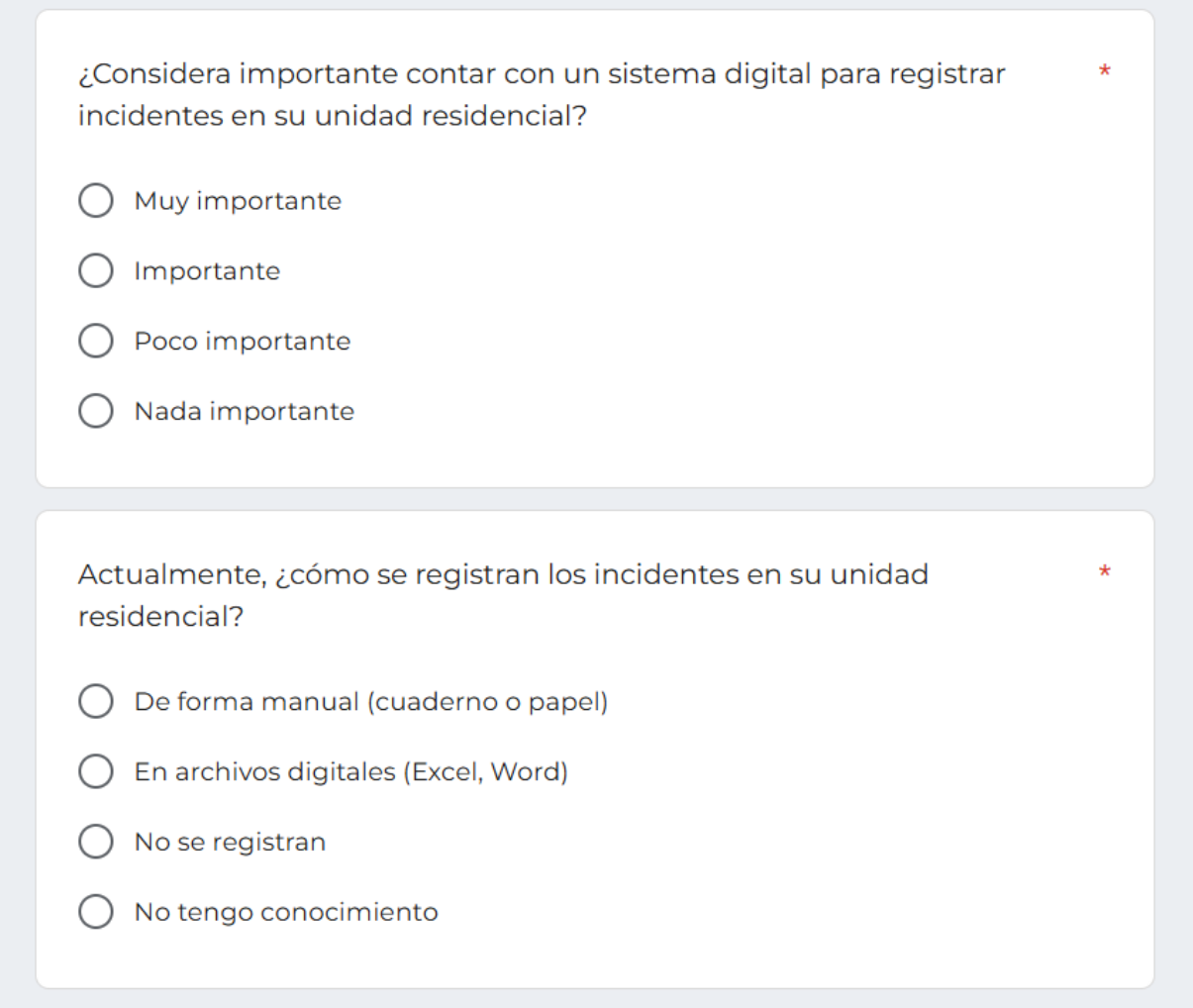


Figura 2 Preguntas 1 y 2

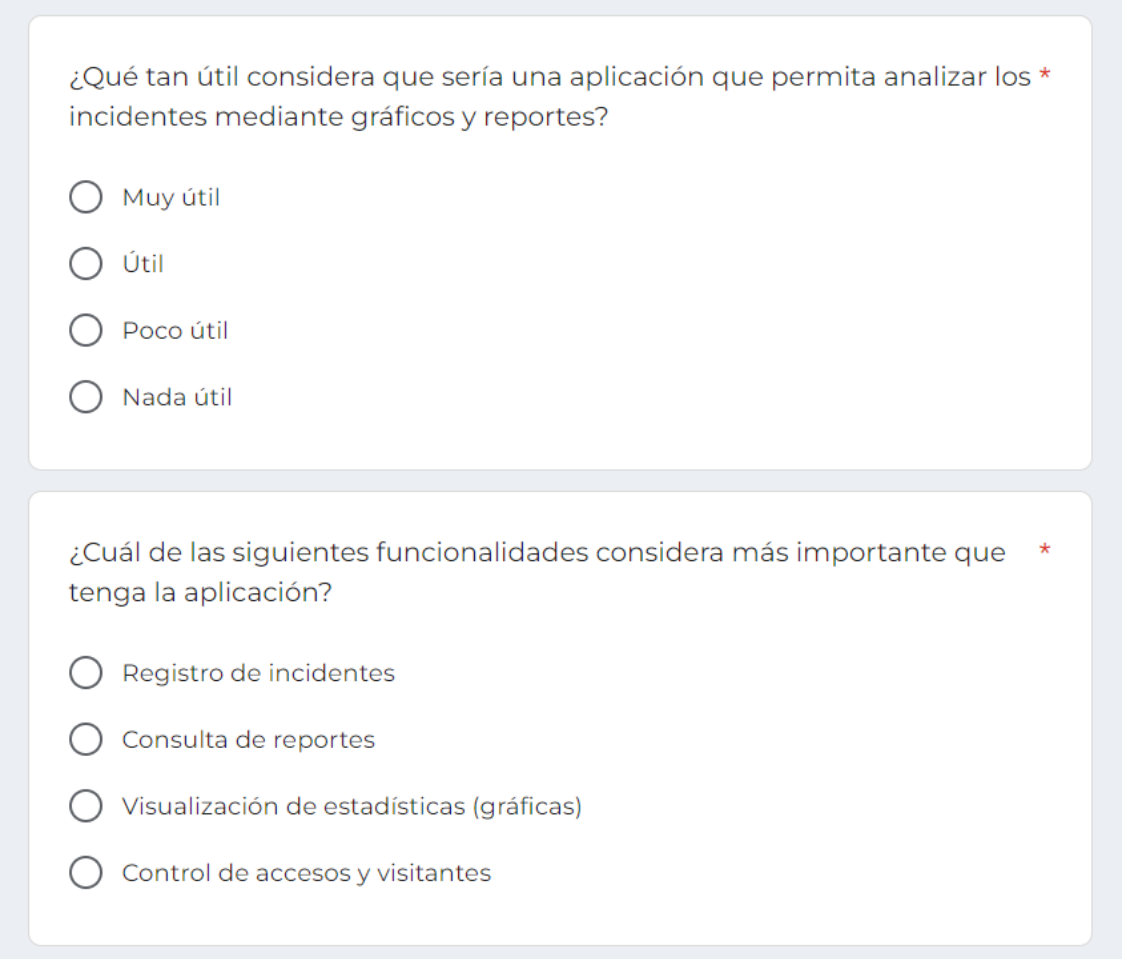


Figura 3 Preguntas 3 y 4

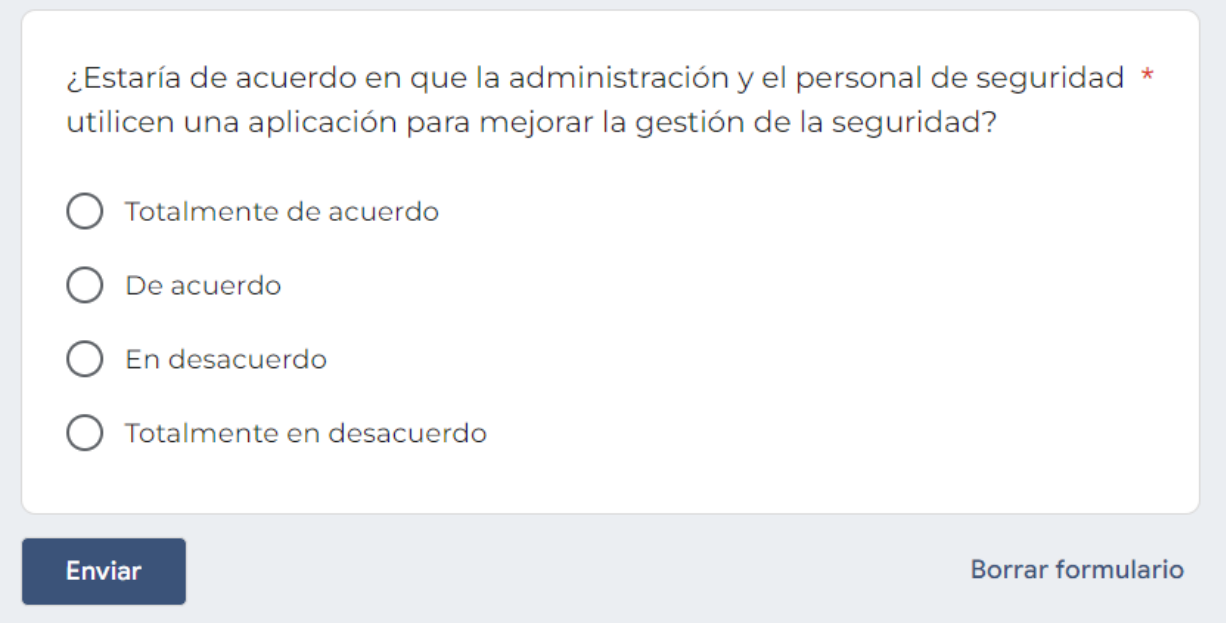


Figura 4 Pregunta 5

Resultados de la encuesta.

La encuesta estructurada se aplicó a 20 potenciales usuarios de unidades residenciales, con el objetivo de validar la necesidad de una plataforma digital para la gestión de incidentes de seguridad y obtener métricas clave para la definición de los requisitos funcionales y no funcionales del sistema.

• Pregunta y resultado clave: Importancia de un sistema digital

El 70% de los encuestados considera muy importante contar con un sistema digital para registrar incidentes, mientras que el 30% lo considera importante.

Implicación para el proyecto:

Estos resultados evidencian que el 100% de los encuestados reconoce la importancia de implementar una solución digital, lo que valida la necesidad del sistema propuesto. Este alto

nivel de aceptación demuestra que los métodos actuales no satisfacen completamente las necesidades de los usuarios, justificando el desarrollo de una plataforma tecnológica orientada a mejorar la gestión de incidentes.

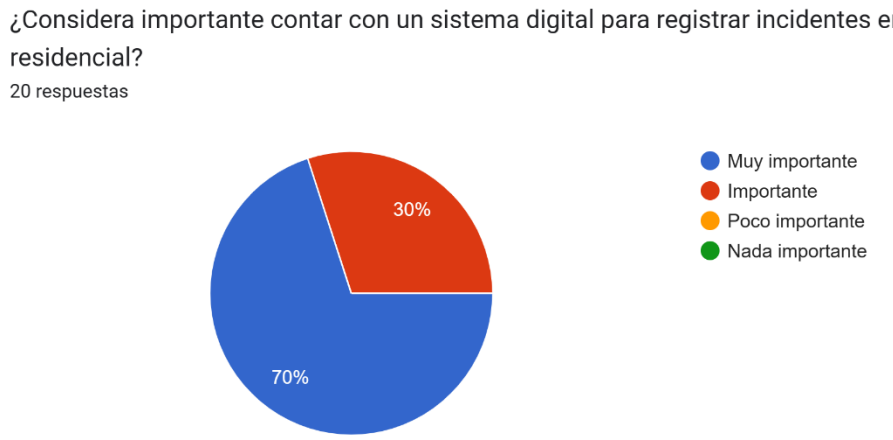


Figura 5 Resultado encuesta pregunta #1

• Pregunta y resultado clave: Forma actual de registro de incidentes

El 25% de los encuestados indica que los incidentes se registran de forma manual, el 25% en archivos digitales (Excel o Word), el 20% afirma que no se registran, y el 30% no tiene conocimiento del proceso.

Implicación para el proyecto:

Los resultados evidencian una falta de estandarización en los procesos de registro, así como debilidades en el control de la información. El hecho de que el 50% (no se registran o no tienen conocimiento) refleja una problemática crítica en la gestión de la seguridad, lo que justifica la implementación de un sistema centralizado que permita mejorar la trazabilidad, organización y acceso a la información.

Figura 6 Resultado encuesta pregunta #2

• Pregunta y resultado clave: Utilidad de la aplicación con gráficos y reportes

El 70% de los encuestados considera que la aplicación sería muy útil, mientras que el 30% la considera útil.

Implicación para el proyecto:

El 100% de aceptación confirma que los usuarios no solo requieren registrar incidentes, sino también analizarlos mediante herramientas visuales. Esto valida la inclusión de funcionalidades como dashboards, reportes y análisis de datos dentro del sistema, permitiendo mejorar la toma de decisiones en temas de seguridad.

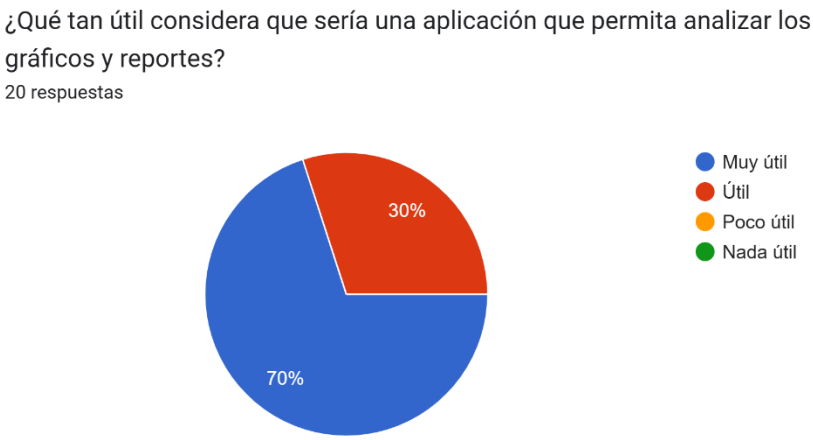


Figura 7 Resultado encuesta pregunta #3

• Pregunta y resultado clave: Funcionalidad más importante

El 75% de los encuestados considera que la funcionalidad más importante es el registro de incidentes, el 15% la visualización de estadísticas, y el 10% la consulta de reportes.

Implicación para el proyecto:

Estos resultados indican que el sistema debe centrarse principalmente en el registro eficiente de incidentes como funcionalidad principal. Sin embargo, también se evidencia la necesidad de incluir herramientas complementarias de análisis y consulta, lo que respalda el desarrollo de una solución integral.

Figura 8 Resultado encuesta pregunta #4

• Pregunta y resultado clave: Disposición de uso de la aplicación

El 85% de los encuestados está totalmente de acuerdo y el 15% está de acuerdo en que la administración y el personal de seguridad utilicen una aplicación para mejorar la gestión de la seguridad.

Implicación para el proyecto:

El 100% de aceptación demuestra que existe una alta disposición al uso de herramientas tecnológicas, lo que reduce significativamente la resistencia al cambio y fortalece la viabilidad de implementación del sistema en un entorno real.

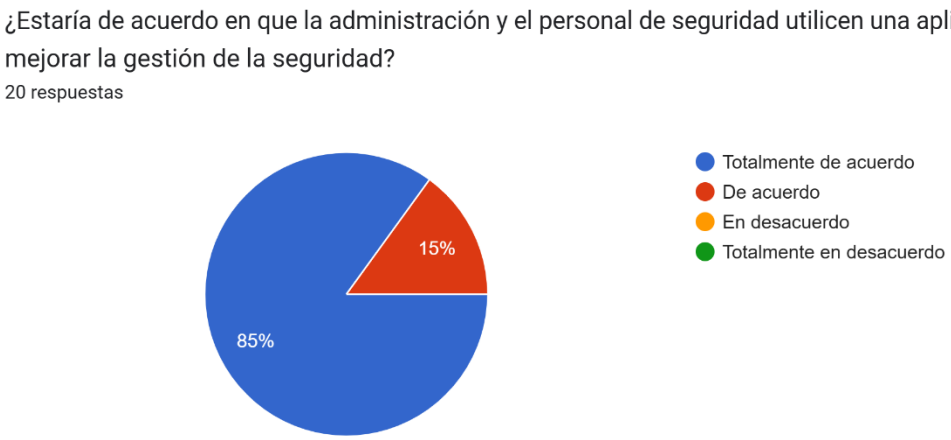


Figura 9 Resultado encuesta pregunta #5

Conclusión del análisis

Los resultados obtenidos evidencian que actualmente existen deficiencias en el registro y control de incidentes en las unidades residenciales, debido al uso de métodos manuales, herramientas básicas o la ausencia total de registros.

Asimismo, se identifica una alta aceptación hacia la implementación de una plataforma digital, así como la necesidad de funcionalidades clave como el registro de incidentes y la visualización de información.

En este sentido, el desarrollo del sistema SeguriLog se presenta como una solución viable y necesaria, orientada a mejorar la gestión de la seguridad, optimizar los procesos y apoyar la toma de decisiones basada en datos.

Objetivo específico 1

Diseñar una base de datos relacional en SQL Server que permita almacenar de forma estructurada y confiable la información de los incidentes registrados en unidades residenciales.

Actividad 1.1 — Identificación de las entidades del sistema

El punto de partida de cualquier base de datos bien diseñada no es la tecnología ni las herramientas: es entender qué información necesita guardar el sistema y de qué manera esa información se relaciona. En el caso de SeguriLog, esta etapa implicó revisar con detalle el problema que se quería resolver y traducirlo en una estructura de datos que lo reflejara fielmente.

A partir del análisis de los requerimientos, se identificaron siete entidades que representan los elementos fundamentales del sistema. La primera es el incidente en sí mismo, que es la razón de ser del proyecto: cada evento que ocurre en un conjunto residencial debe quedar documentado con suficiente detalle para que pueda analizarse después. Pero para que ese registro tenga sentido, necesita estar conectado con otras entidades: el usuario que lo reportó, la unidad residencial donde ocurrió, el tipo de evento que fue y el lugar específico dentro de esa unidad.

Las otras entidades —roles, tipos de incidente, ubicaciones específicas y tokens de recuperación de contraseña— sirven de soporte a esas relaciones principales. Los roles definen qué puede hacer cada persona dentro del sistema. Los catálogos de tipos de incidente y de ubicaciones estandarizan la forma en que se clasifica la información, lo que es fundamental para poder hacer análisis comparativos. Y los tokens de recuperación de contraseña son la estructura que permite gestionar el proceso de restablecimiento de acceso de forma segura.

La Tabla 1 presenta los componentes de la arquitectura del sistema, incluyendo la capa de datos con las tecnologías y elementos que la integran.

Tabla 1 Componentes de arquitectura

Capa	Componente	Descripción	Tecnología
Presentación	Login	Pantalla donde los usuarios ingresan su correo y contraseña para acceder al sistema.	HTML, CSS, JavaScript
	Dashboard	Pantalla principal con indicadores del sistema: total de incidentes, abiertos, cerrados e incidentes del día. Incluye gráficas y actividad reciente.	HTML, Chart.js
	Registro de incidentes	Formulario para documentar un nuevo incidente. Permite seleccionar el tipo, la unidad, la ubicación y escribir una descripción.	HTML, CSS, JavaScript
	Consulta y filtros	Módulo para buscar incidentes usando filtros por fecha, tipo, unidad y estado. Permite ver el detalle, editar y exportar a Excel.	HTML, JavaScript

	Panel de administración	Sección exclusiva para administradores donde se gestionan usuarios, unidades residenciales y se consulta el historial de accesos.	HTML, CSS, JavaScript
Lógica de negocio	AuthService	Se encarga de verificar las credenciales al iniciar sesión, generar el token de seguridad y registrar la última vez que el usuario entró.	C#, ASP.NET Core 8
	IncidenteService	Maneja todas las operaciones sobre los incidentes: registrar, consultar, editar y eliminar, respetando los permisos de cada usuario.	C#, Entity Framework Core
	RecuperacionService	Gestiona el proceso de recuperación de contraseña enviando un código de verificación por correo con una duración de 10 minutos.	C#, ASP.NET Core

	ExportService	Genera el archivo de exportación en formato CSV compatible con Excel y Power BI.	C#, ASP.NET Core
	EmailService	Envía correos electrónicos al sistema, como el mensaje con el código de recuperación de contraseña.	C#, SMTP
Acceso a datos	AppDbContext	Componente que conecta el sistema con la base de datos y traduce las operaciones del sistema en consultas SQL.	C#, EF Core 8
	AdminController	Punto de acceso de la API exclusivo para administradores: gestión de usuarios, unidades y log de accesos.	C#, ASP.NET Core
Datos	SQL Server	Base de datos donde se almacena toda la información del sistema: 7 tablas, vistas y procedimientos de exportación.	SQL Server 2019 Express
	VW_Incidentes_PowerBI	Vista especial de la base de datos con columnas de tiempo (año, mes,	T-SQL, SQL Server

		día, hora) lista para conectarse directamente con Power BI.	
Análisis	Power BI Desktop	Herramienta externa de análisis conectada a la base de datos para generar reportes y dashboards interactivos con 4 páginas de análisis.	Power BI Desktop, DAX

Tabla 2 Herramientas de proceso y colaboración.

Categoría	Herramienta	Uso en el proyecto	Licencia
Desarrollo backend	Visual Studio 2022	Editor principal para escribir y depurar el código del servidor (backend) en C#.	Community — gratuita
	.NET 8 SDK	Plataforma que permite ejecutar y construir la aplicación del servidor.	MIT — gratuita
	Entity Framework Core	Librería que permite trabajar con la base de datos usando código C# en lugar de escribir SQL directamente.	MIT — gratuita
	BCrypt.Net-Next	Librería para cifrar las contraseñas de los usuarios de forma segura	MIT — gratuita

		antes de guardarlas en la base de datos.	
	Swagger	Genera automáticamente una página de documentación que permite probar los servicios del sistema desde el navegador.	Apache 2.0 — gratuita
Desarrollo frontend	Visual Studio Code	Editor de código utilizado para desarrollar las páginas web del sistema (HTML, CSS y JavaScript).	MIT — gratuita
	Chart.js	Librería para crear las gráficas del dashboard, como las barras de incidentes por mes.	MIT — gratuita
	Google Fonts	Fuentes tipográficas usadas en el diseño visual del sistema (Syne para títulos y Mulish para el texto del cuerpo).	OFL — gratuita
Base de datos	SQL Server 2019 Express	Motor de base de datos donde se almacena toda la información del sistema.	Express — gratuita

	SSMS 19	Herramienta gráfica para administrar, consultar y visualizar la base de datos de SQL Server.	Gratuita
Análisis de datos	Power BI Desktop	Aplicación de Microsoft para crear dashboards y reportes visuales conectados directamente a la base de datos del sistema.	Gratuita
	DAX	Lenguaje de fórmulas propio de Power BI, usado para calcular indicadores como la tasa de cierre y el promedio de horas de resolución.	Integrado en Power BI
Colaboración	Git / GitHub	Sistema de control de versiones para llevar un historial de cambios del proyecto y hacer copias de seguridad del código.	Git 2.x — gratuita

Actividad 1.2 — Construcción del modelo entidad-relación

Una vez identificadas las entidades, el siguiente paso fue definir cómo se relacionan entre sí. Eso es lo que representa el modelo entidad-relación (MER): un mapa visual que muestra qué elementos guarda el sistema y cómo se conectan, sin entrar todavía en los detalles técnicos de la implementación.

En SeguriLog, las relaciones más importantes son del tipo uno a muchos, lo que significa que un elemento de una tabla puede estar relacionado con varios elementos de otra, pero no al revés. Un rol puede estar asignado a muchos usuarios, pero cada usuario tiene un solo rol. Una unidad residencial puede tener muchos incidentes registrados, pero cada incidente pertenece a una sola unidad. Un tipo de incidente puede clasificar muchos eventos, pero cada evento tiene un único tipo. Y un vigilante puede registrar muchos incidentes a lo largo de su tiempo en el servicio, mientras que cada incidente queda vinculado al vigilante específico que lo documentó.

Esta estructura de relaciones no es arbitraria: responde directamente al problema que el sistema busca resolver. Si los incidentes no estuvieran conectados con los tipos, no se podría saber cuáles son los eventos más frecuentes. Si no estuvieran conectados con las unidades, no se podría identificar en qué conjuntos se concentran más los problemas. Y si no estuvieran conectados con los usuarios, no habría trazabilidad sobre quién registró qué y cuándo. Kimball y Ross (2013) destacan precisamente que el diseño de las relaciones entre entidades es lo que determina qué tipo de análisis será posible hacer después.

La Figura 10 muestra el diagrama MER completo del sistema, con las siete entidades, sus atributos principales, las llaves primarias (PK) y las llaves foráneas (FK) que establecen las relaciones.

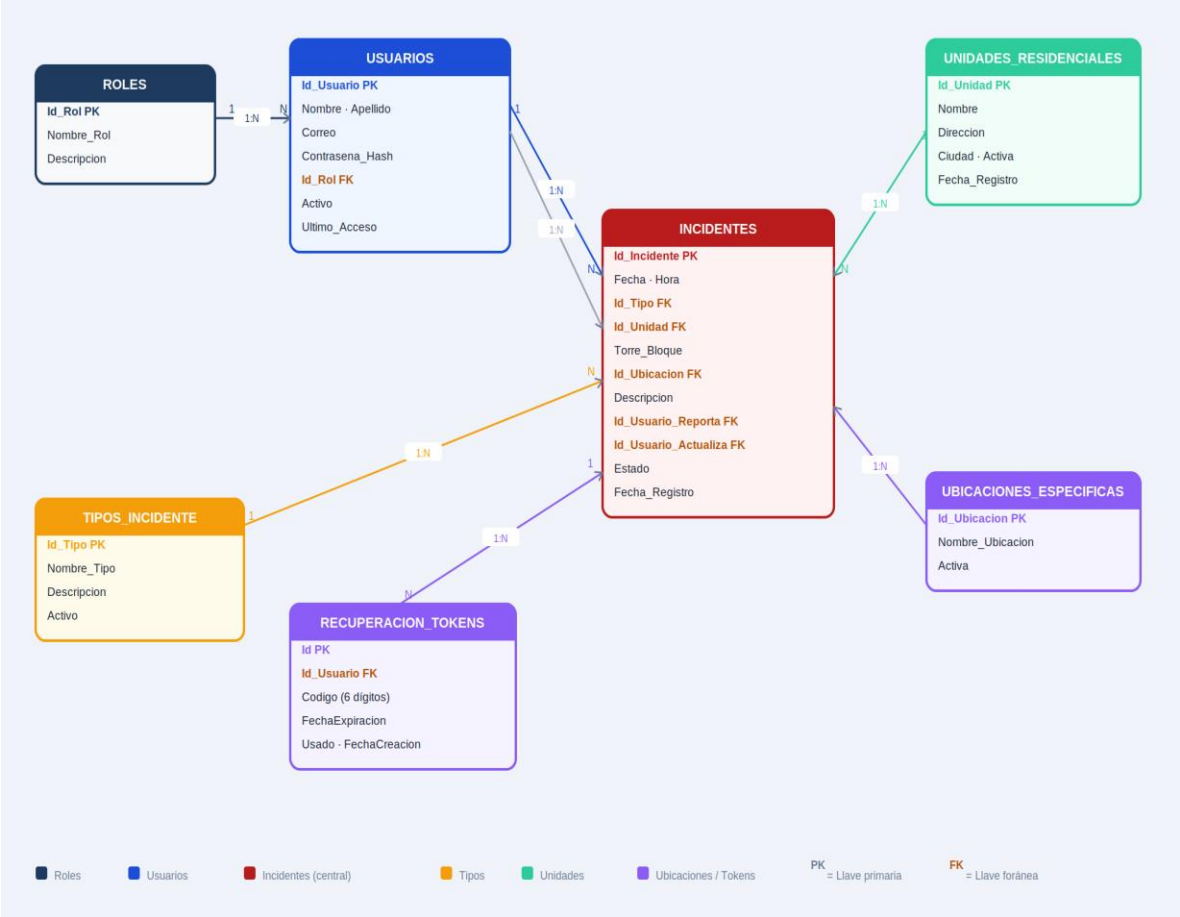


Figura 10 Modelo entidad-relación del sistema

Tabla 3 Convenciones del diagrama MER

USUARIOS	PK = Llave primaria FK = Llave foránea	Gestiona credenciales y permisos
INCIDENTES	PK = Llave primaria FK = Llave foránea	Entidad central del sistema
UNIDADES	PK = Llave primaria FK = Llave foránea	Conjuntos residenciales

TIPOS	PK = Llave primaria FK = Llave foránea	Categorías de incidentes
UBICACIONES	PK = Llave primaria FK = Llave foránea	Lugares específicos
TOKENS OTP	PK = Llave primaria FK = Llave foránea	Recuperación de contraseña

Actividad 1.3 — Descripción de las tablas de la base de datos

Con el modelo entidad-relación como guía, se implementaron las siete tablas en SQL Server. El proceso de implementación implicó definir el tipo de dato de cada campo, establecer las restricciones de integridad y configurar los índices necesarios para que las consultas sean eficientes. A continuación, se describe el propósito de cada tabla y las decisiones de diseño más relevantes.

La tabla Roles es la más sencilla del sistema, pero cumple una función crítica: centraliza los tres niveles de acceso (Vigilante, Supervisor y Administrador) en un único lugar. Gracias a eso, si en el futuro se quisiera agregar un nuevo rol o modificar los permisos de uno existente, el cambio solo afectaría a esta tabla y no a todo el sistema.

La tabla Usuarios almacena la información de cada persona que tiene acceso al sistema: nombre, apellido, correo, contraseña cifrada con BCrypt, el rol asignado, si la cuenta está activa y la fecha del último acceso. El campo de contraseña nunca guarda el texto legible, sino el resultado del algoritmo de cifrado, lo que garantiza que incluso si alguien accediera directamente a la base de datos no podría conocer las contraseñas reales.

La tabla Unidades_Residenciales registra los conjuntos residenciales donde opera el servicio de vigilancia, con su nombre, dirección, ciudad y si está activa. Las unidades desactivadas no desaparecen del sistema, sino que simplemente dejan de mostrarse en los formularios de registro, lo que conserva el historial de incidentes asociados a ellas.

Las tablas Tipos_Incidente y Ubicaciones_Especificas son catálogos: listas controladas de valores que estandarizan la forma en que se clasifica la información. Usar catálogos en lugar de texto libre garantiza que el análisis posterior sea confiable, porque todos los registros que corresponden a "hurto", por ejemplo, estarán clasificados de la misma manera y podrán contarse y compararse sin ambigüedad.

La tabla Incidentes es el corazón del sistema. Cada fila representa un evento registrado y contiene referencias a todas las demás tablas: el tipo de incidente, la unidad donde ocurrió, la ubicación específica dentro de esa unidad, el vigilante que lo reportó y el supervisor que lo actualizó. Además, guarda la fecha, la hora, la descripción del evento, el estado (Abierto o Cerrado) y las fechas de registro y última actualización. Esta riqueza de datos es la que hace posible el análisis multidimensional en los dashboards.

La tabla Recuperacion_Tokens guarda temporalmente los códigos de verificación de seis dígitos que se generan cuando un usuario solicita recuperar su contraseña. Cada código tiene una fecha de expiración de diez minutos y se invalida automáticamente después de usarse una vez, lo que garantiza que el proceso de recuperación sea seguro y no pueda ser aprovechado por terceros.

Además de las tablas, se creó la vista VW_Incidentes_PowerBI. Esta no almacena datos, sino que es una consulta predefinida que combina la información de todas las tablas relevantes y la presenta en un formato plano con columnas adicionales de tiempo: año, nombre del mes, número de semana, día de la semana y hora del día. La existencia de esta vista es lo que hace posible

conectar la base de datos con Power BI sin necesidad de transformaciones adicionales en la herramienta de análisis.

Actividad 1.4 — Justificación de las decisiones de diseño

Diseñar una base de datos supone tomar decisiones que tienen consecuencias directas en el rendimiento, la seguridad y las posibilidades de análisis del sistema. En este proyecto, tres decisiones merecen una explicación particular porque no son evidentes a primera vista, pero resultan fundamentales para que el sistema funcione bien.

La primera decisión fue normalizar las tablas siguiendo la tercera forma normal. La normalización garantiza que cada dato se almacene en un único lugar y que los cambios no generen inconsistencias. Un ejemplo concreto: si el nombre de una unidad residencial cambia, basta con actualizar un solo registro en la tabla Unidades_Residenciales. Todos los incidentes vinculados a esa unidad reflejarán el cambio automáticamente, porque no guardan el nombre directamente, sino una referencia (la llave foránea) a ese registro. Codd (1970), quien propuso el modelo relacional, estableció precisamente este principio como el fundamento de la integridad de los datos.

La segunda decisión fue separar la base de datos transaccional de la vista de análisis. Las tablas del sistema están optimizadas para registrar información de manera rápida y segura durante la operación diaria. La vista VW_Incidentes_PowerBI, en cambio, está optimizada para consultar y analizar. Conectar Power BI directamente a las tablas operativas habría generado consultas lentas y potencialmente habría afectado el rendimiento del sistema durante su uso. La vista actúa como un intermediario que protege la operación y facilita el análisis.

La tercera decisión fue crear índices en los campos de búsqueda más utilizados: Fecha, Id_Unidad, Id_Tipo y Estado. Los índices son estructuras adicionales en la base de datos que aceleran las consultas de la misma manera que un índice al final de un libro permite encontrar un

tema sin leer todo el contenido. Esta decisión responde directamente al requisito no funcional RNF-07, que establece que las consultas con filtros deben ejecutarse en menos de dos segundos incluso cuando la base de datos tiene miles de registros.

Objetivo específico 2

Desarrollar una aplicación web en ASP.NET Core que permita registrar, consultar, editar y gestionar incidentes, con acceso diferenciado según el rol de cada usuario.

Actividad 2.1 — Definición de la arquitectura del sistema

La arquitectura de un sistema de software es, en términos simples, la decisión sobre cómo organizar sus piezas y cómo van a comunicarse entre sí. Es una decisión que se toma antes de escribir código y que determina qué tan fácil será mantener el sistema, agregar nuevas funcionalidades o corregir errores en el futuro sin que todo lo demás se rompa.

Para SeguriLog se adoptó una arquitectura en capas, que es uno de los modelos más utilizados en el desarrollo de aplicaciones empresariales. Fowler (2002) la describe como una estructura donde cada capa tiene una responsabilidad clara y solo se comunica con la capa inmediatamente adyacente. Esto garantiza que los cambios en una parte del sistema no generen efectos inesperados en otras.

La capa de presentación es lo que el usuario ve y con lo que interactúa: las páginas web del sistema, construidas con HTML, CSS y JavaScript. Esta capa no sabe nada de cómo funciona la base de datos; solo sabe cómo mostrar información y cómo enviar las acciones del usuario al servidor.

La capa de lógica de negocio es el corazón del sistema. Aquí residen las reglas que determinan qué puede hacer cada tipo de usuario, cómo se validan los datos antes de guardarse,

cómo se gestionan las sesiones y cómo se procesa cada solicitud. Esta capa está implementada en C# usando ASP.NET Core 8, y se organiza en servicios con responsabilidades bien definidas: AuthService para la autenticación, IncidenteService para la gestión de eventos, RecuperacionService para el flujo de contraseñas, EmailService para el envío de correos y ExportService para la generación de archivos CSV.

La capa de acceso a datos es el puente entre la lógica del sistema y la base de datos. Se implementó con Entity Framework Core 8, que permite trabajar con los datos usando código C# en lugar de escribir consultas SQL directamente. Esto hace que el código sea más legible y mantenible, y reduce el riesgo de errores.

La capa de datos es SQL Server, que almacena toda la información del sistema de forma persistente.

Adicionalmente, Power BI actúa como una capa externa de análisis y visualización: se conecta a la base de datos a través de la vista VW_Incidentes_PowerBI y genera los dashboards interactivos que soportan la toma de decisiones.

La Figura 11 presenta el diagrama de arquitectura lógica del sistema, con las cuatro capas internas, los componentes de cada una y las tecnologías utilizadas.

Figura 11 Diagrama de arquitectura lógica

Tabla 4 Arquitectura lógica

Capa	Tecnología	Responsabilidad
Presentación	HTML5 · CSS3 · JavaScript · Chart.js	Interfaz de usuario. Interacción con el usuario, validaciones en el cliente, visualización de datos y llamadas a la API REST.
Lógica de negocio	ASP.NET Core 8 · C# · JWT · BCrypt	Lógica de aplicación. Autenticación, autorización por roles, CRUD de incidentes, envío de correos y exportación de datos.

Acceso a datos	Entity Framework Core 8 · Fluent API	Abstracción de la base de datos mediante ORM. Mapeo objeto-relacional, relaciones entre entidades y consultas LINQ.
Datos	SQL Server 2019 Express	Persistencia. 7 tablas normalizadas, vista VW_Incidentes_PowerBI para análisis, triggers de auditoría y stored procedures.
Análisis (ext.)	Power BI Desktop · DAX	Componente externo de análisis. Se conecta directamente a la vista SQL para generar dashboards y reportes interactivos.

Actividad 2.2 — Definición de los casos de uso

Los casos de uso describen qué puede hacer cada tipo de usuario dentro del sistema. Son una herramienta de análisis que permite asegurarse de que el sistema responde a las necesidades reales de quienes lo van a usar, antes de comenzar el desarrollo.

En SeguriLog se identificaron tres actores con niveles de acceso diferentes. El Vigilante es el usuario operativo del sistema: su función principal es documentar los eventos que ocurren durante su turno de trabajo y consultarlos cuando lo necesite. No puede modificar lo que ya se registró ni eliminar información, lo que garantiza que los datos sean confiables y no puedan alterarse posteriormente. El Supervisor tiene las mismas capacidades del Vigilante y además puede editar los incidentes existentes, cambiar su estado de Abierto a Cerrado y exportar los datos a un archivo CSV para análisis externos. El Administrador tiene acceso total al sistema: puede hacer todo lo que hacen los otros roles y además gestionar los usuarios (crear, editar, activar o desactivar cuentas), gestionar las unidades residenciales y revisar el historial de accesos al sistema.

Esta jerarquía de permisos no es una decisión arbitraria: responde a la realidad operativa de las empresas de seguridad privada, donde existe una distinción clara entre quienes ejecutan el servicio en campo, quienes lo supervisan y quienes lo administran. Laudon y Laudon (2016) señalan que los sistemas de información bien diseñados deben reflejar esa estructura organizacional para ser adoptados efectivamente por sus usuarios.

La Figura 12 presenta el diagrama de casos de uso del sistema. Los colores diferencian las acciones según el nivel mínimo de acceso requerido: verde para las funciones disponibles para todos los usuarios, azul para las que requieren rol de Supervisor o superior, y morado para las exclusivas del Administrador.

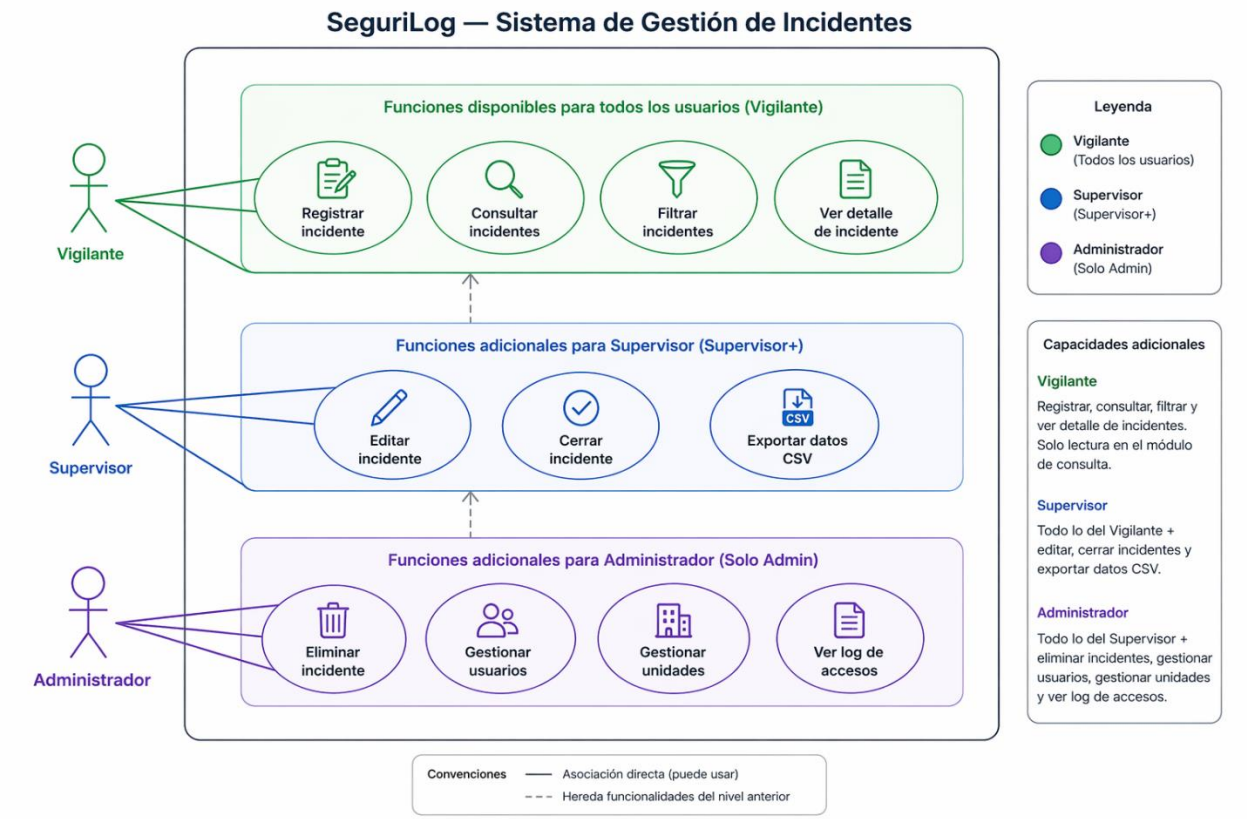


Figura 12 Diagrama de casos de uso

Tabla 5 Casos de uso

Actor	Color en el diagrama	Capacidades adicionales
Vigilante	Verde (todos los usuarios)	Registrar, consultar, filtrar y ver detalle de incidentes. Solo lectura en el módulo de consulta.
Supervisor	Azul (Supervisor+)	Todo lo del Vigilante + editar, cerrar incidentes y exportar datos CSV.

Administrador	Morado (solo Admin)	Todo lo del Supervisor + eliminar incidentes, gestionar usuarios, gestionar unidades y ver log de accesos.

La Tabla 6 presenta el catálogo completo de requisitos funcionales del sistema, que describe con mayor detalle cada una de las funcionalidades identificadas en los casos de uso, junto con el actor responsable y el nivel de prioridad de cada una. En total se definieron 17 requisitos distribuidos en cinco módulos: autenticación, incidentes, exportación, dashboard y administración.

Tabla 6 Requisitos funcionales del sistema

ID	Módulo	Descripción	Actor	Prioridad
RF-01	Autenticación	Los usuarios ingresan con correo y contraseña. Las contraseñas se guardan cifradas.	Todos	**Alta**
RF-02	Autenticación	El sistema ofrece recuperación de contraseña en 3 pasos: ingresar el correo, verificar un código enviado por correo y establecer la nueva contraseña.	Todos	**Alta**
RF-03	Autenticación	Si el usuario no ha iniciado sesión o su sesión venció, el sistema lo redirige automáticamente a la pantalla de login.	Sistema	**Alta**

RF-04	Incidentes	El vigilante puede registrar un incidente seleccionando el tipo, la unidad, la ubicación y escribiendo una descripción. El sistema agrega automáticamente quién lo registró.	Vigilante, Supervisor, Admin	**Alta**
RF-05	Incidentes	Se pueden buscar incidentes usando filtros por fecha, estado, unidad y tipo. Los resultados se muestran de 12 en 12.	Todos	**Alta**
RF-06	Incidentes	Los supervisores y administradores pueden modificar la descripción, la ubicación y el estado (abierto o cerrado) de un incidente ya registrado.	Supervisor, Admin	**Alta**
RF-07	Incidentes	Solo el administrador puede eliminar un incidente de forma permanente. El sistema solicita confirmación antes de hacerlo.	Administrador	**Media**
RF-08	Incidentes	Al hacer clic en una fila de la tabla, se puede ver el detalle completo del incidente sin abrir una página nueva.	Todos	**Media**

RF-09	Exportación	El sistema permite descargar los incidentes en un archivo Excel (.csv) con opción de filtrar por rango de fechas.	Supervisor, Admin	**Alta**
RF-10	Exportación	La base de datos tiene una vista especial con la información organizada por año, mes, día y hora, lista para conectarse con Power BI.	Sistema	**Alta**
RF-11	Dashboard	La pantalla principal muestra cuatro indicadores en tiempo real: total de incidentes, abiertos, cerrados e incidentes del día.	Todos	**Alta**
RF-12	Dashboard	El dashboard incluye una gráfica de barras con los incidentes de los últimos 6 meses, diferenciando los abiertos de los cerrados.	Todos	**Media**
RF-13	Administración	El administrador puede crear, editar, activar y desactivar usuarios del sistema. El rol asignado define qué puede hacer cada uno.	Administrador	**Alta**

RF-14	Administración	El administrador puede cambiar la contraseña de cualquier usuario sin necesidad de saber cuál es la contraseña actual.	Administrador	**Alta**
RF-15	Administración	El administrador puede crear y gestionar las unidades residenciales del sistema. Las unidades desactivadas no aparecen al registrar incidentes.	Administrador	**Media**
RF-16	Administración	El sistema guarda un registro de los últimos accesos: quién entró, cuándo y con qué rol.	Administrador	**Baja**
RF-17	Control de acceso	Cada usuario tiene permisos según su rol: el vigilante solo registra y consulta; el supervisor también edita; el administrador tiene acceso total.	Sistema	**Alta**

Actividad 2.3 — Descripción de los flujos de actividad

Los diagramas de flujo de actividad complementan los casos de uso añadiendo una dimensión temporal: no solo describen qué puede hacer cada usuario, sino cómo ocurre ese proceso paso a paso, qué hace el sistema en respuesta a cada acción y qué pasa cuando algo no sale como se esperaba. Son especialmente útiles para comunicar los procesos a personas que no tienen

formación técnica, porque representan visualmente la secuencia de eventos de una manera intuitiva.

Flujo de autenticación — inicio de sesión

El proceso de inicio de sesión comienza cuando el usuario accede a la pantalla de login e ingresa su correo y contraseña. El sistema valida en el cliente que los campos no estén vacíos y que el correo tenga el formato correcto. Si la validación pasa, las credenciales se envían al servidor mediante una petición segura.

En el servidor, el sistema busca al usuario por su correo en la base de datos. Si no existe o su cuenta está desactivada, retorna un error sin especificar cuál de los dos casos se cumple, lo que evita dar información que podría usarse para identificar usuarios registrados. Si el usuario existe y está activo, se verifica la contraseña usando el algoritmo BCrypt: se compara lo que el usuario escribió con el hash almacenado, sin necesidad de descifrar nada.

Si las credenciales son correctas, el sistema genera un token JWT (JSON Web Token) con una duración de ocho horas, registra la fecha y hora del acceso en la tabla Usuarios y retorna el token al navegador. El navegador guarda ese token localmente y lo incluye automáticamente en todas las peticiones posteriores, lo que permite que el usuario navegue por el sistema sin volver a autenticarse hasta que su sesión venza o cierre sesión manualmente.

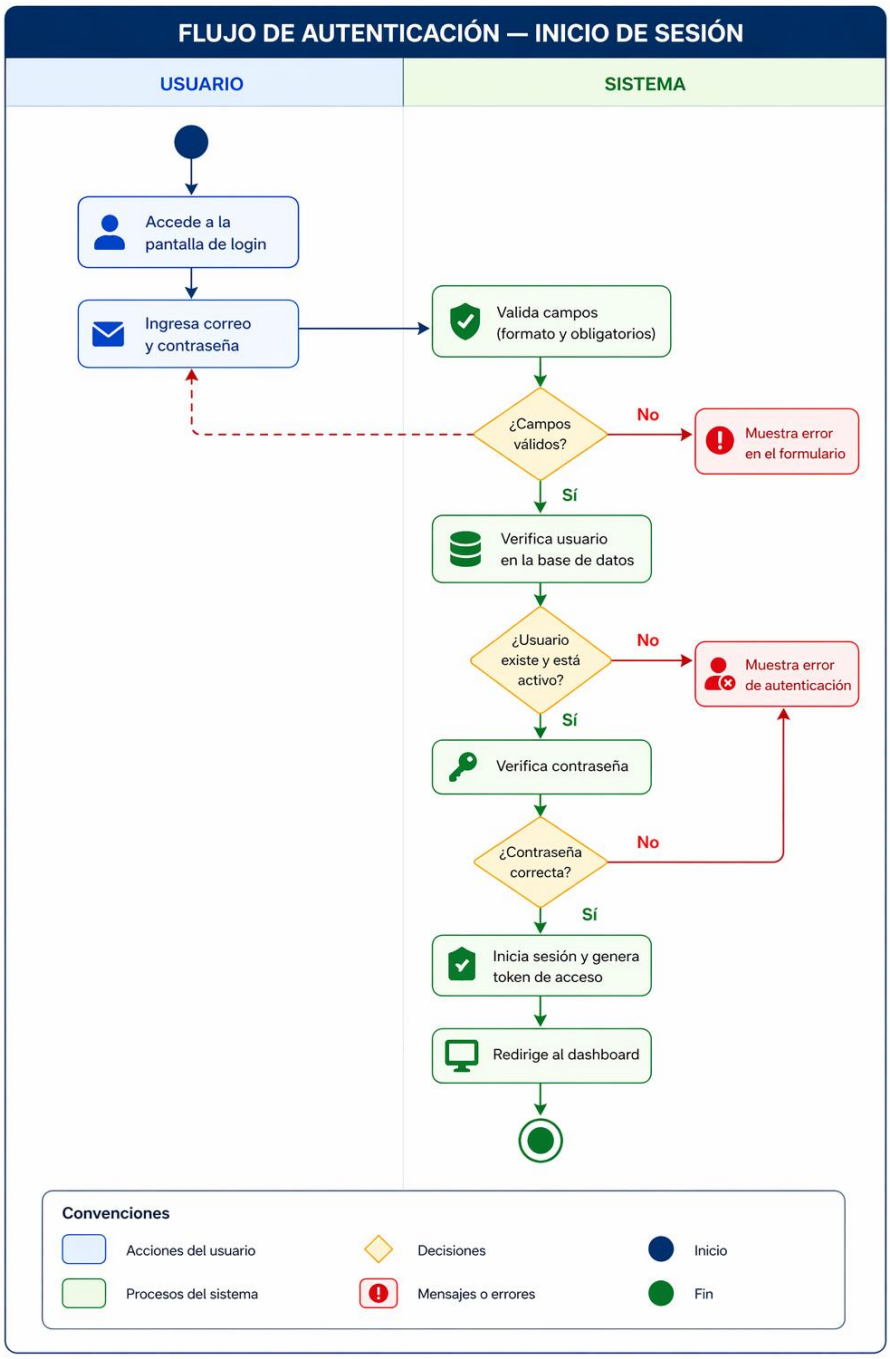


Figura 13 Diagrama de flujo del proceso de autenticación en el sistema

Flujo de registro de incidentes

Cuando un vigilante necesita documentar un evento, accede al módulo de registro desde el menú lateral. Al cargar la pantalla, el sistema consulta automáticamente los catálogos de tipos de

incidente, unidades residenciales y ubicaciones específicas, que se muestran como opciones en los campos del formulario. La fecha y la hora se precargaron con los valores actuales, aunque el vigilante puede modificarlos si el evento no está siendo registrado en el momento exacto en que ocurrió.

El vigilante selecciona el tipo de incidente mediante tarjetas visuales, completa los campos de unidad, ubicación, torre o bloque y escribe una descripción del evento. Al hacer clic en "Guardar", el sistema valida en el cliente que todos los campos obligatorios estén completos y con el formato correcto. Si alguno falta, muestra un mensaje de error junto al campo correspondiente sin borrar lo que el usuario ya escribió.

Una vez superada la validación, los datos se envían al servidor. El servidor extrae el identificador del vigilante del token JWT que acompaña la petición, crea el registro con estado "Abierto" y retorna una confirmación. El navegador muestra un mensaje de éxito y limpia el formulario para que el vigilante pueda registrar otro incidente si es necesario.

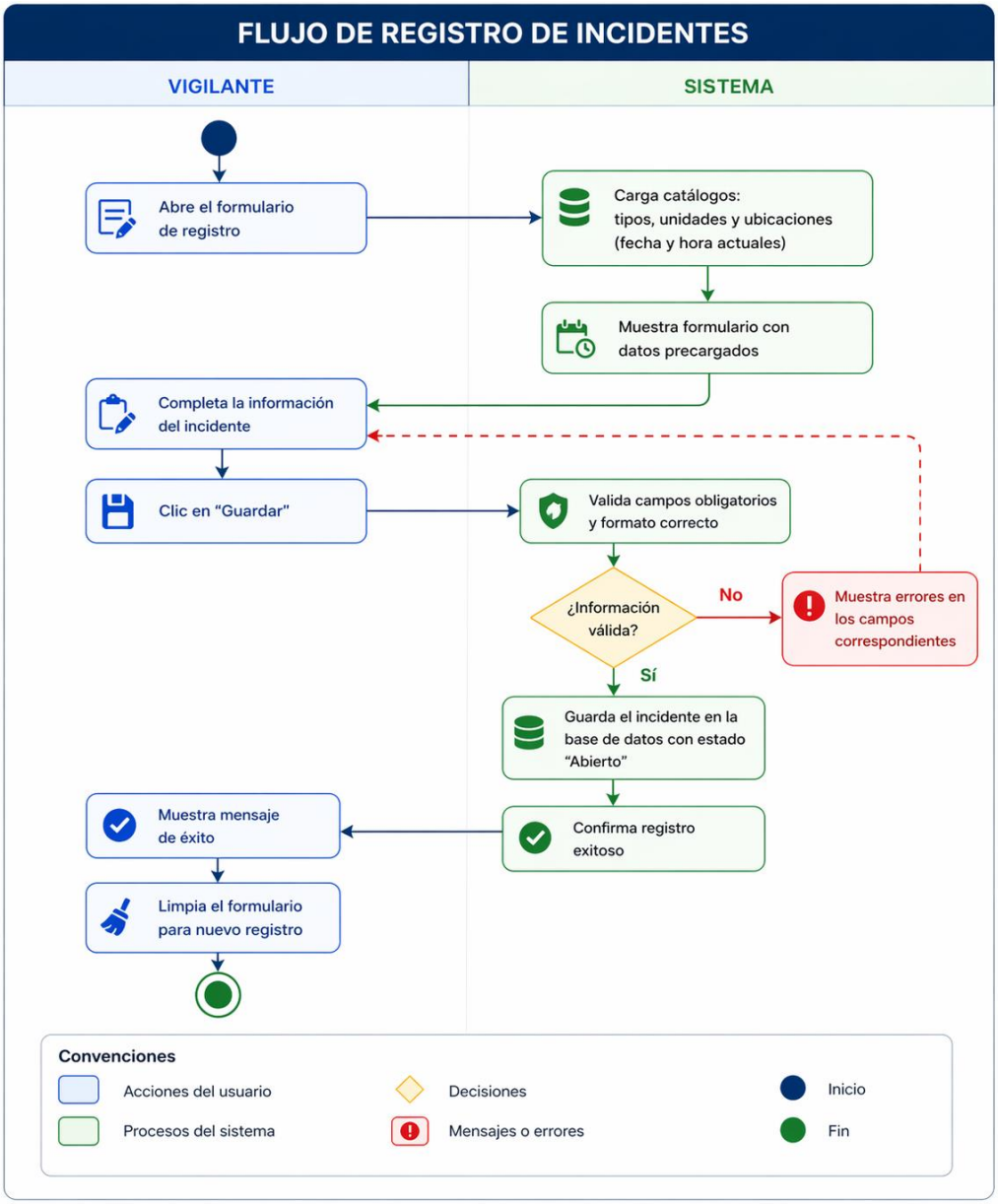


Figura 14 Diagrama de flujo del proceso de registro de incidentes

Flujo de recuperación de contraseña

El proceso de recuperación de contraseña se diseñó en tres pasos para garantizar que solo el titular de la cuenta pueda restablecerla, sin necesidad de intervención de un administrador.

En el primer paso, el usuario ingresa su correo registrado en el sistema. El servidor verifica que ese correo corresponda a una cuenta activa. Si lo es, genera un código aleatorio de seis dígitos, lo guarda en la tabla Recuperacion_Tokens con una fecha de expiración de diez minutos e invalida cualquier código anterior que pudiera existir para ese usuario. Luego envía el código al correo del usuario mediante una plantilla HTML.

En el segundo paso, el usuario ingresa el código recibido. El servidor verifica que el código sea correcto, que no haya expirado y que no haya sido usado anteriormente. Si alguna de esas condiciones no se cumple, retorna un error específico que indica qué falló, sin revelar información adicional.

En el tercer paso, el usuario escribe su nueva contraseña y la confirma. El sistema verifica que ambas coincidan y que cumplan los criterios mínimos de seguridad. Si todo es correcto, actualiza el hash en la base de datos, marca el token como usado y redirige al usuario a la pantalla de inicio de sesión con un mensaje de confirmación.

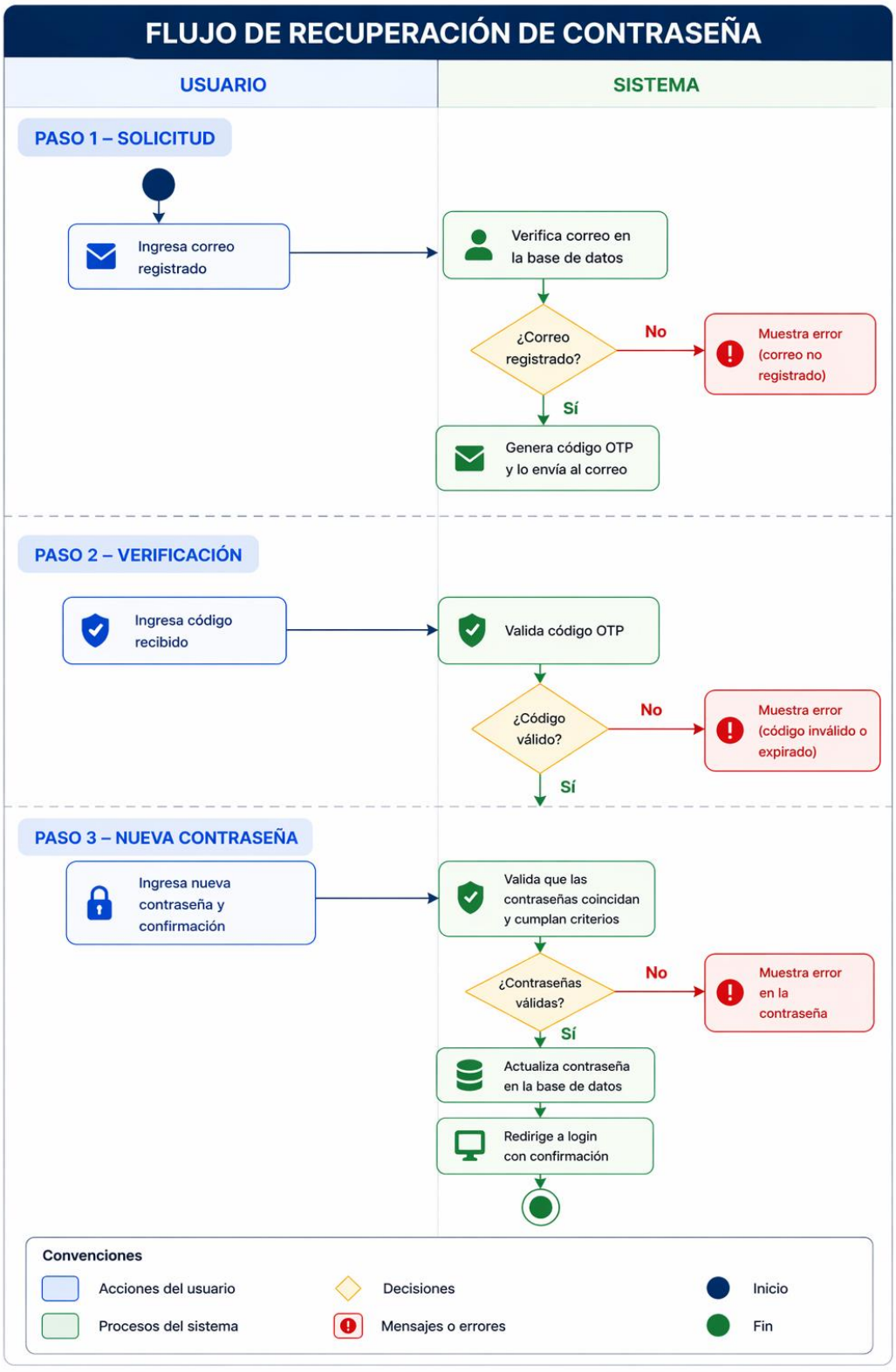


Figura 15 Diagrama de flujo del proceso de recuperación de contraseña

Actividad 2.4 — Presentación de las pantallas desarrolladas

La aplicación web resultante cuenta con cinco módulos principales, cada uno diseñado para responder a las necesidades de un tipo específico de usuario. A continuación, se describe cada módulo y se presenta una captura de pantalla que ilustra la versión final implementada.

Pantalla de login

La pantalla de acceso al sistema está diseñada con dos secciones diferenciadas. La izquierda presenta una descripción del sistema con sus características principales y una barra de estado que indica si el servicio está activo. La derecha contiene el formulario de inicio de sesión con los campos de correo y contraseña, y un enlace de acceso rápido al proceso de recuperación. El diseño responde al requisito funcional RF-01, que establece que el sistema debe permitir el inicio de sesión mediante credenciales y gestionar la sesión mediante token JWT.

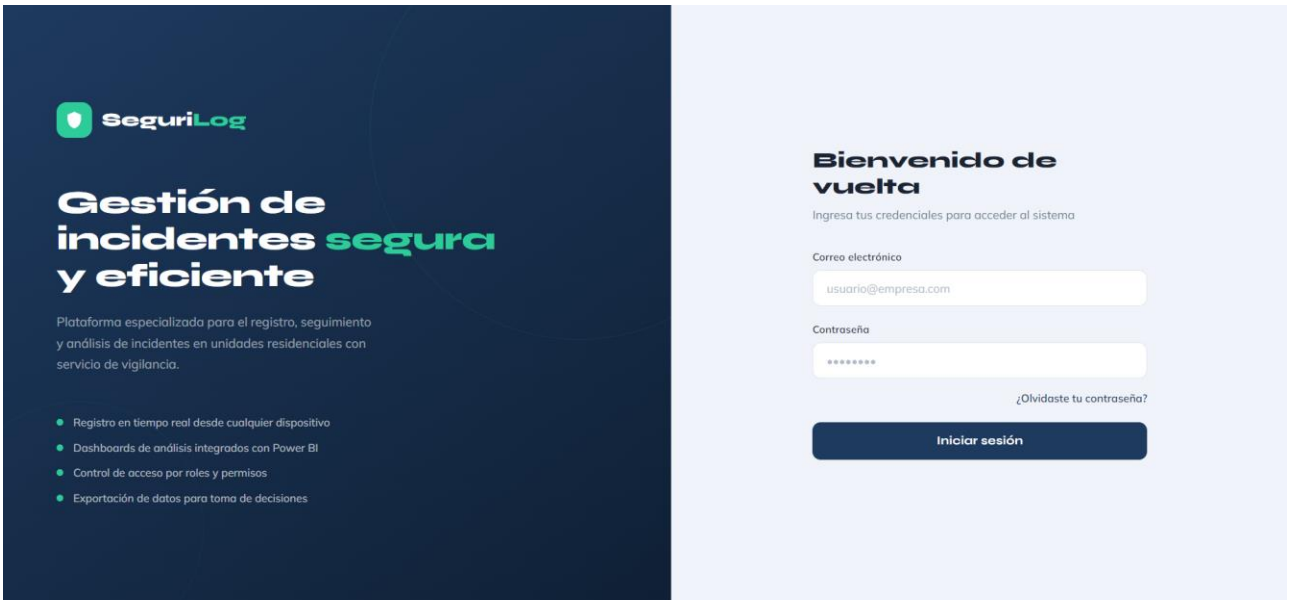


Figura 16 Login SeguriLog

Dashboard principal

El dashboard es la primera pantalla que ve el usuario al iniciar sesión y está diseñado para ofrecer una visión inmediata del estado del sistema. En la parte superior muestra cuatro tarjetas con indicadores en tiempo real: el total de incidentes registrados, los que están abiertos, los que han sido cerrados y los del día actual. Cada tarjeta incluye un indicador de tendencia que compara el valor actual con el período anterior.

En la sección central aparece una gráfica de barras agrupadas con los incidentes de los últimos seis meses, diferenciando los abiertos de los cerrados. A la derecha, una gráfica de líneas muestra la tendencia acumulada del total de incidentes. En la parte inferior se presenta la distribución por tipo de incidente y un listado de la actividad más reciente. El módulo del dashboard responde a los requisitos funcionales RF-11 y RF-12.

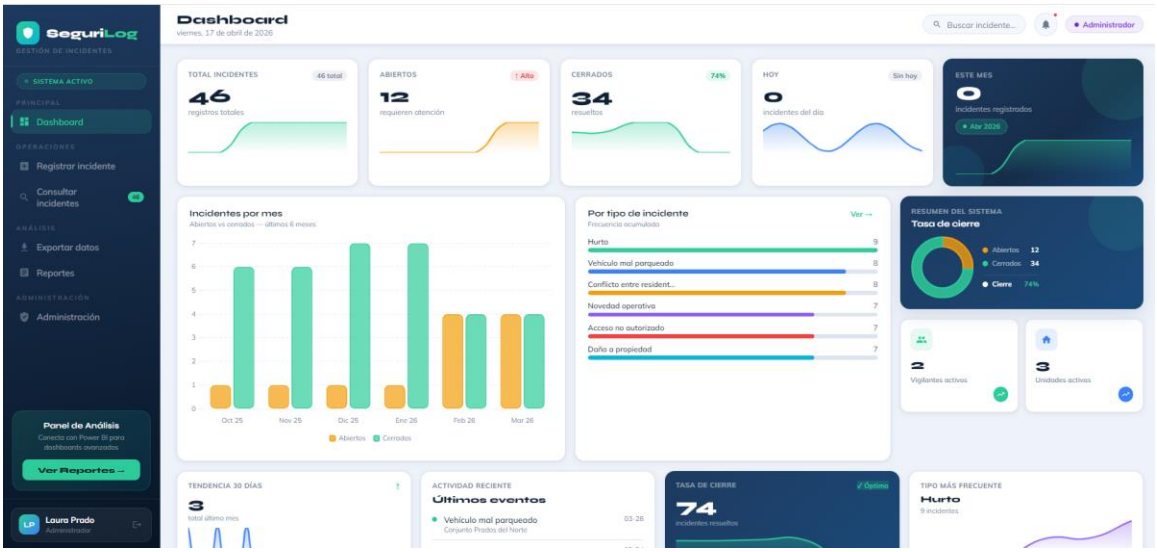


Figura 17 Pantalla Dashboard SeguriLog

Módulo de registro de incidentes

El formulario de registro está organizado para guiar al vigilante de manera intuitiva. Lo primero que debe seleccionar es el tipo de incidente, que se presenta como un conjunto de tarjetas visuales con el nombre e icono de cada categoría. Una vez seleccionado el tipo, aparecen los demás campos: unidad residencial, ubicación específica, torre o bloque, fecha, hora y descripción del evento. Los campos de catálogo se cargan dinámicamente desde el servidor, lo que garantiza que las opciones disponibles sean siempre las activas en el sistema. El módulo responde al requisito funcional RF-04.

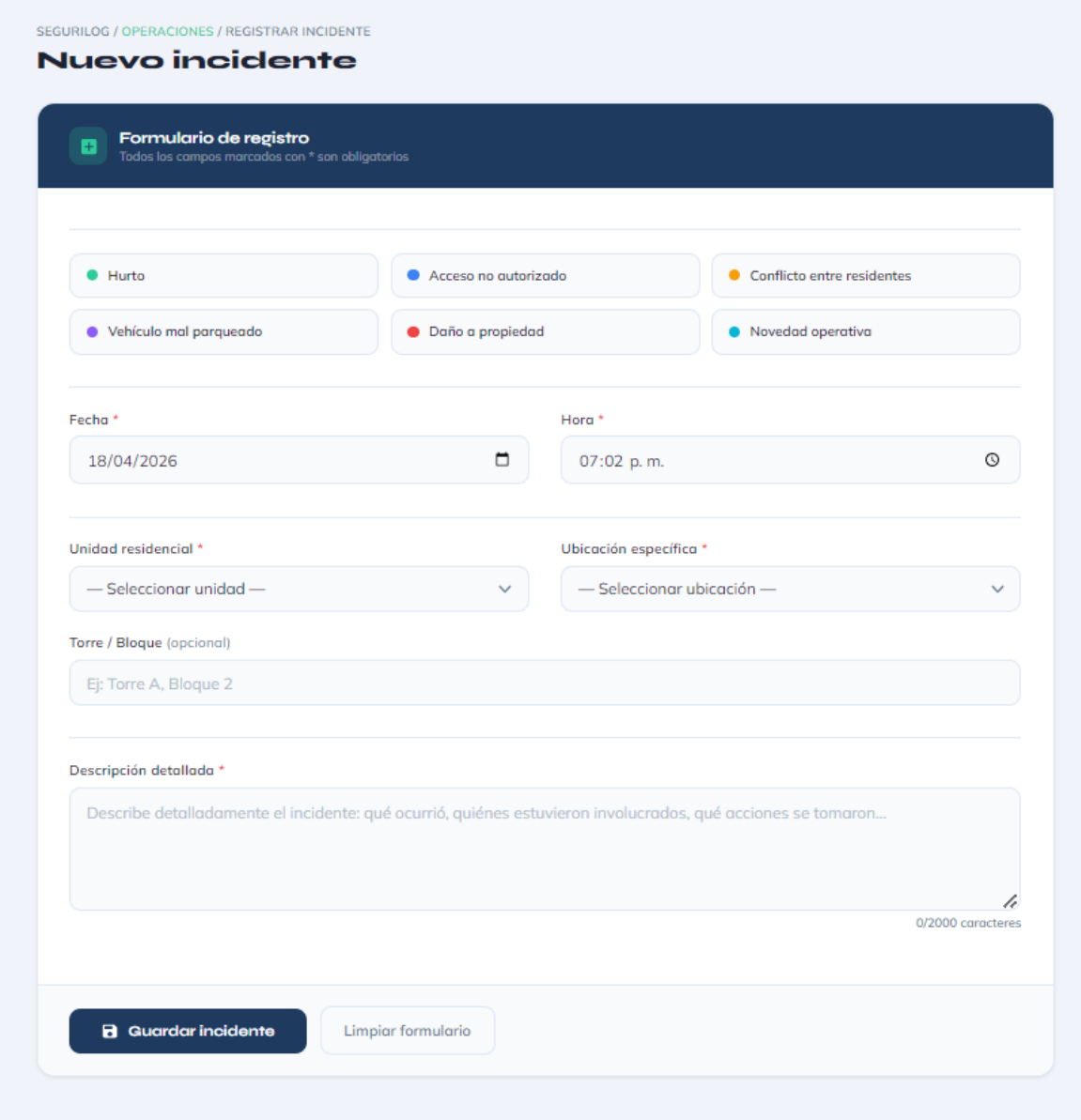


Figura 18 Módulo de registro de incidentes

Módulo de consulta y filtros

Este módulo permite buscar y revisar los incidentes registrados usando filtros combinables: rango de fechas, estado, unidad residencial y tipo de incidente. Los resultados se muestran en una tabla paginada de doce en doce. Al hacer clic en cualquier fila, se despliega el detalle completo del incidente sin necesidad de abrir una nueva pantalla. Los chips de filtros activos muestran en todo

momento qué criterios de búsqueda están aplicados, con la opción de eliminar cada uno individualmente.

Desde este módulo, los supervisores y administradores pueden editar un incidente haciendo clic en el botón correspondiente, que abre un modal con los campos modificables. Los vigilantes ven el mismo módulo, pero sin la opción de edición, lo que implementa el control de acceso basado en roles definido en el requisito RF-17. El módulo responde a los requisitos funcionales RF-05, RF-06, RF-07, RF-08 y RF-09.

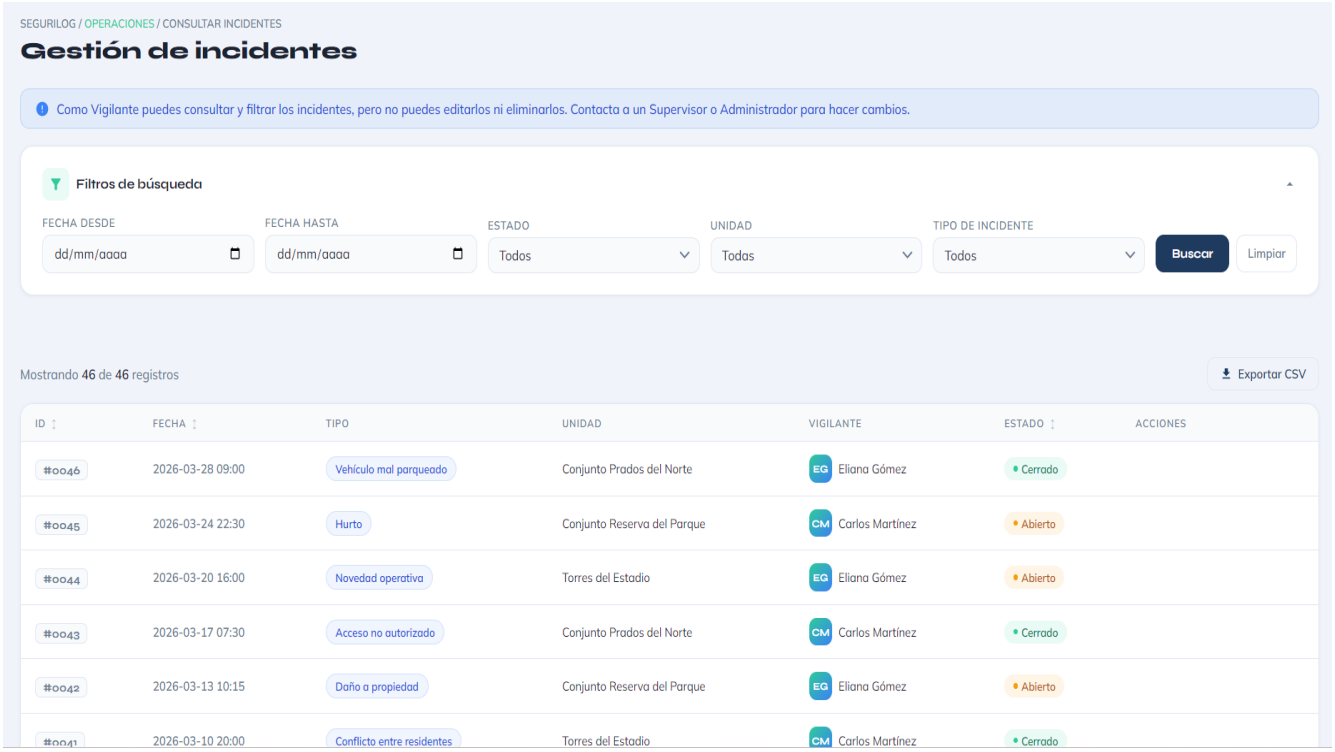


Figura 19 Módulo de consultas y filtros

Panel de administración

El panel de administración es exclusivo del rol Administrador y está organizado en tres pestañas. La primera pestaña, Usuarios, muestra la lista de todos los usuarios del sistema con su nombre, correo, rol, estado y fecha del último acceso. Desde aquí se pueden crear nuevos usuarios,

editar los existentes, activarlos o desactivarlos y cambiar sus contraseñas. La segunda pestaña, Unidades, permite gestionar los conjuntos residenciales registrados en el sistema. La tercera pestaña, Log de accesos, muestra los últimos cincuenta ingresos al sistema con el detalle de quién entró, cuándo y con qué rol. El módulo responde a los requisitos funcionales RF-13, RF-14, RF-15 y RF-16.

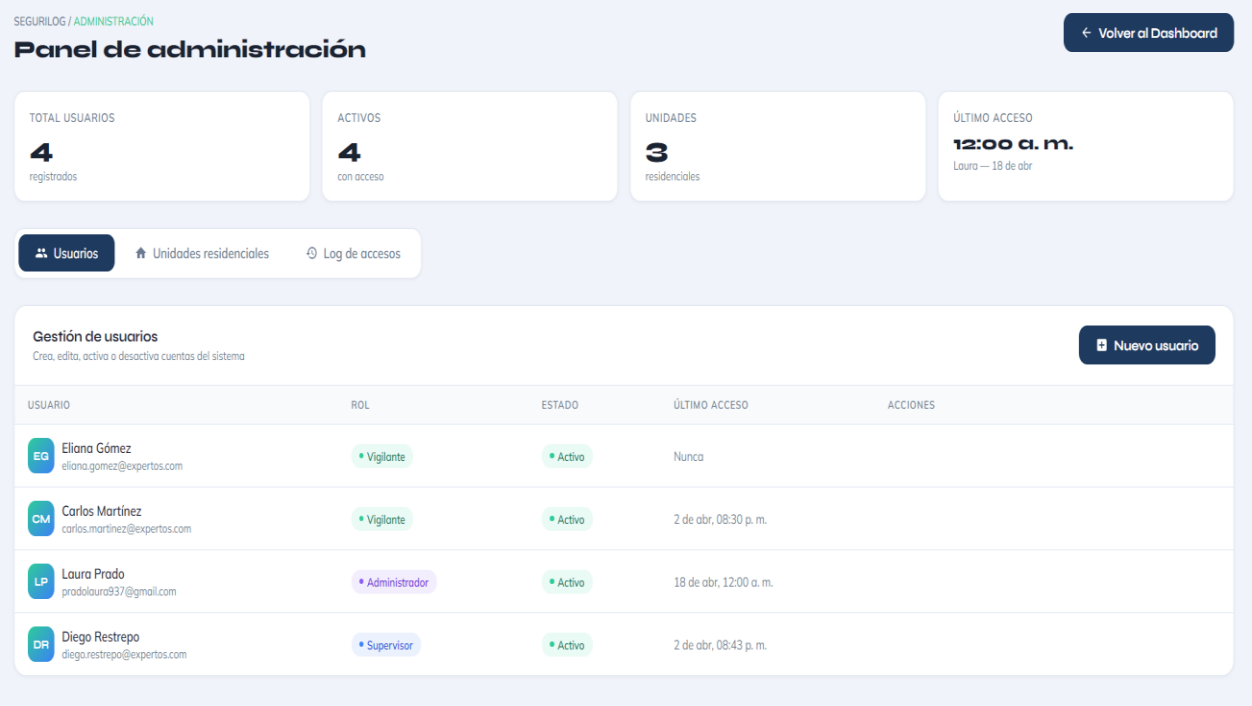


Figura 20 Panel de administración

Objetivo específico 3

Implementar un proceso de organización y transformación de datos mediante vistas SQL conectadas directamente a Power BI, para facilitar su análisis posterior.

Actividad 3.1 — Descripción del flujo de datos

Para que los dashboards de Power BI puedan mostrar información útil y actualizada, es necesario que los datos recorran un camino claro desde que se originan hasta que se visualizan. Ese recorrido,

conocido habitualmente como pipeline o flujo de datos, define cómo se captura la información, cómo se almacena y cómo se prepara para el análisis.

En SeguriLog, el flujo comienza en el formulario de registro de la aplicación web. Cuando el vigilante documenta un incidente y hace clic en guardar, el navegador envía los datos al servidor (ASP.NET Core 8). El servidor los valida, extrae el identificador del vigilante del token JWT de sesión y los almacena en la base de datos SQL Server 2019. En ese punto, el registro queda disponible de inmediato en el módulo de consulta de la aplicación. Las fechas se guardan en hora local colombiana (UTC-5, zona SA Pacific Standard Time) para evitar desfases horarios en los registros.

La segunda parte del flujo ocurre cuando Power BI necesita actualizar los dashboards. En lugar de conectarse directamente a las tablas operativas del sistema, Power BI accede a la vista VW_Incidentes_PowerBI, que es una consulta predefinida que combina la información de todas las tablas relevantes y la organiza en un formato plano optimizado para el análisis. Esta separación entre la base de datos transaccional y la fuente de datos analítica es una práctica recomendada por Kimball y Ross (2013), quienes señalan que mezclar las operaciones de registro con las de análisis puede degradar el rendimiento de ambas.

Esta arquitectura garantiza que el sistema operativo no se vea afectado por las consultas de análisis, y que los dashboards siempre reflejen la información más reciente disponible en la base de datos.

Actividad 3.2 — Estructura de la vista VW_Incidentes_PowerBI

La vista VW_Incidentes_PowerBI es el elemento técnico que hace posible el análisis en Power BI. Se construyó como una consulta SQL que une la información de las tablas de incidentes, usuarios,

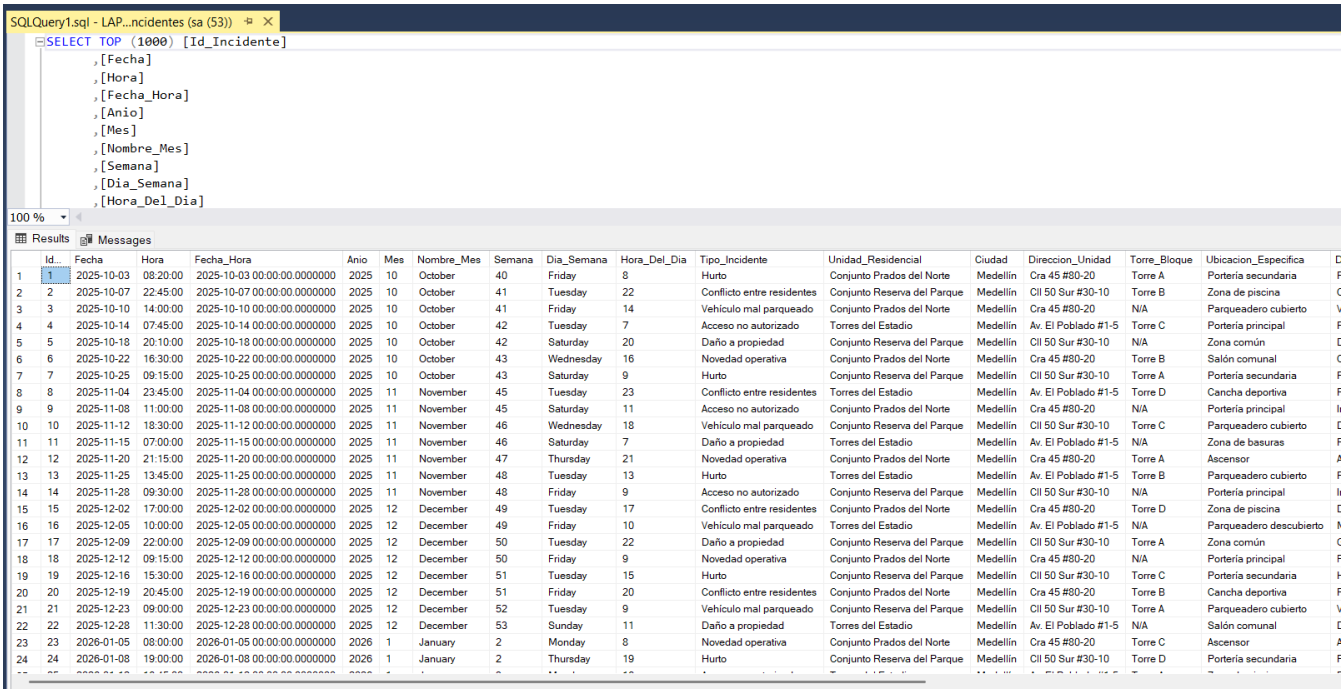
unidades residenciales, tipos de incidente y ubicaciones específicas, produciendo un conjunto de datos en formato plano donde cada fila representa un incidente con todos sus atributos descriptivos resueltos: en lugar del identificador de la unidad, aparece el nombre de la unidad; en lugar del identificador del tipo, aparece el nombre del tipo; en lugar del identificador del vigilante, aparece su nombre completo.

Las columnas calculadas de tiempo son la parte más importante de la vista, ya que se derivan automáticamente del campo Fecha de cada incidente. El año (Anio) permite filtrar por período anual. El mes en número (Mes) y en nombre (Nombre_Mes) permite el análisis mensual y la ordenación cronológica en las gráficas. El número de semana del año (Semana) facilita el análisis de tendencias semanales. El día de la semana (Dia_Semana) permite identificar qué días concentran más eventos. Y la hora del día (Hora_Del_Dia), extraída del campo Hora, permite analizar los patrones horarios de los incidentes.

La vista también incluye la columna Horas_Para_Cierre, que calcula el tiempo transcurrido entre el registro y la actualización de cada incidente cuando su estado es "Cerrado". Esta columna es la que alimenta el indicador de promedio de horas de resolución en los dashboards. Durante el desarrollo se identificó que esta columna devolvía valores NULL para los incidentes cerrados porque el trigger de la tabla interfería con el mecanismo de actualización de Entity Framework Core. La solución implicó configurar el contexto de datos para reconocer el trigger mediante el método HasTrigger(), lo que permitió que las actualizaciones se guardaran correctamente.

Estas columnas calculadas son fundamentales para que Power BI pueda construir los análisis temporales sin necesidad de transformaciones adicionales en la herramienta de visualización. Corresponden al requisito funcional RF-10, que establece que el sistema debe

organizar la información de los incidentes con columnas de tiempo para su uso directo en Power BI.



SQLQuery1.sql - LAP...ncidentes (sa (53))

```
SELECT TOP (1000) [Id_Incidente]
      ,[Fecha]
      ,[Hora]
      ,[Fecha_Hora]
      ,[Anio]
      ,[Mes]
      ,[Nombre_Mes]
      ,[Semana]
      ,[Dia_Semana]
      ,[Hora_Del_Dia]
```

100 %

Results | Messages

	Id...	Fecha	Hora	Fecha_Hora	Anio	Mes	Nombre_Mes	Semana	Dia_Semana	Hora_Del_Dia	Tipo_Incidente	Unidad_Residencial	Ciudad	Direccion_Unidad	Torre_Bloque	Ubicacion_Especifica
1	1	2025-10-03	08:20:00	2025-10-03 00:00:00.0000000	2025	10	October	40	Friday	8	Hurto	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	Torre A	Portería secundaria
2	2	2025-10-07	22:45:00	2025-10-07 00:00:00.0000000	2025	10	October	41	Tuesday	22	Conflicto entre residentes	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	Torre B	Zona de piscina
3	3	2025-10-10	14:00:00	2025-10-10 00:00:00.0000000	2025	10	October	41	Friday	14	Vehículo mal parqueado	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	N/A	Parqueadero cubierto
4	4	2025-10-14	07:45:00	2025-10-14 00:00:00.0000000	2025	10	October	42	Tuesday	7	Acceso no autorizado	Torres del Estadio	Medellín	Av. El Poblado #1-5	Torre C	Portería principal
5	5	2025-10-18	20:10:00	2025-10-18 00:00:00.0000000	2025	10	October	42	Saturday	20	Daño a propiedad	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	N/A	Zona común
6	6	2025-10-22	16:30:00	2025-10-22 00:00:00.0000000	2025	10	October	43	Wednesday	16	Novedad operativa	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	Torre B	Salón comunal
7	7	2025-10-25	09:15:00	2025-10-25 00:00:00.0000000	2025	10	October	43	Saturday	9	Hurto	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	Torre A	Portería secundaria
8	8	2025-11-04	23:45:00	2025-11-04 00:00:00.0000000	2025	11	November	45	Tuesday	23	Conflicto entre residentes	Torres del Estadio	Medellín	Av. El Poblado #1-5	Torre D	Cancha deportiva
9	9	2025-11-08	11:00:00	2025-11-08 00:00:00.0000000	2025	11	November	45	Saturday	11	Acceso no autorizado	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	N/A	Portería principal
10	10	2025-11-12	18:30:00	2025-11-12 00:00:00.0000000	2025	11	November	46	Wednesday	18	Vehículo mal parqueado	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	Torre C	Parqueadero cubierto
11	11	2025-11-15	07:00:00	2025-11-15 00:00:00.0000000	2025	11	November	46	Saturday	7	Daño a propiedad	Torres del Estadio	Medellín	Av. El Poblado #1-5	N/A	Zona de basuras
12	12	2025-11-20	21:15:00	2025-11-20 00:00:00.0000000	2025	11	November	47	Thursday	21	Novedad operativa	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	Torre A	Ascensor
13	13	2025-11-25	13:45:00	2025-11-25 00:00:00.0000000	2025	11	November	48	Tuesday	13	Hurto	Torres del Estadio	Medellín	Av. El Poblado #1-5	Torre B	Parqueadero cubierto
14	14	2025-11-28	09:30:00	2025-11-28 00:00:00.0000000	2025	11	November	48	Friday	9	Acceso no autorizado	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	N/A	Portería principal
15	15	2025-12-02	17:00:00	2025-12-02 00:00:00.0000000	2025	12	December	49	Tuesday	17	Conflicto entre residentes	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	Torre D	Zona de piscina
16	16	2025-12-05	10:00:00	2025-12-05 00:00:00.0000000	2025	12	December	49	Friday	10	Vehículo mal parqueado	Torres del Estadio	Medellín	Av. El Poblado #1-5	N/A	Parqueadero descubierto
17	17	2025-12-09	22:00:00	2025-12-09 00:00:00.0000000	2025	12	December	50	Tuesday	22	Daño a propiedad	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	Torre A	Zona común
18	18	2025-12-12	09:15:00	2025-12-12 00:00:00.0000000	2025	12	December	50	Friday	9	Novedad operativa	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	N/A	Portería principal
19	19	2025-12-16	15:30:00	2025-12-16 00:00:00.0000000	2025	12	December	51	Tuesday	15	Hurto	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	Torre C	Portería secundaria
20	20	2025-12-19	20:45:00	2025-12-19 00:00:00.0000000	2025	12	December	51	Friday	20	Conflicto entre residentes	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	Torre B	Cancha deportiva
21	21	2025-12-23	09:00:00	2025-12-23 00:00:00.0000000	2025	12	December	52	Tuesday	9	Vehículo mal parqueado	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	Torre A	Parqueadero cubierto
22	22	2025-12-28	11:30:00	2025-12-28 00:00:00.0000000	2025	12	December	53	Sunday	11	Daño a propiedad	Torres del Estadio	Medellín	Av. El Poblado #1-5	N/A	Salón comunal
23	23	2026-01-05	08:00:00	2026-01-05 00:00:00.0000000	2026	1	January	2	Monday	8	Novedad operativa	Conjunto Prados del Norte	Medellín	Cra 45 #80-20	Torre C	Ascensor
24	24	2026-01-08	19:00:00	2026-01-08 00:00:00.0000000	2026	1	January	2	Thursday	19	Hurto	Conjunto Reserva del Parque	Medellín	Cll 50 Sur #30-10	Torre D	Portería secundaria

Figura 21 Columnas de la vista VW_Incidentes_PowerBI

Actividad 3.3 — Medidas DAX creadas en Power BI

Una vez conectada la vista a Power BI, se crearon las medidas de análisis usando DAX (Data Analysis Expressions), que es el lenguaje de fórmulas nativo de Power BI. Las medidas son cálculos que se actualizan en tiempo real cada vez que el usuario aplica un filtro en el dashboard, lo que permite que todos los indicadores respondan simultáneamente a las selecciones del usuario sin necesidad de recargar la página. Todas las medidas se organizaron en una tabla independiente llamada _Medidas para mantener el modelo de datos ordenado.

El primer grupo contiene los indicadores base del sistema, que responden a las preguntas más inmediatas sobre el estado de los incidentes:

1. Total Incidentes: cuenta el número total de registros en el período y los filtros seleccionados.
2. Total Abiertos: filtra y cuenta solo los incidentes cuyo estado es "Abierto".
3. Total Cerrados: filtra y cuenta solo los incidentes cuyo estado es "Cerrado".
4. Tasa de Cierre %: calcula el porcentaje de incidentes resueltos sobre el total, expresado como número entre 0 y 100.
5. Promedio Horas Cierre: calcula el tiempo promedio, en horas, que transcurre desde el registro de un incidente hasta que se marca como cerrado. La medida filtra exclusivamente los incidentes en estado "Cerrado" con valores mayores a cero en Horas_Para_Cierre para evitar que los registros abiertos distorsionen el promedio. El resultado se redondea al entero más cercano.
6. Promedio Horas Texto: convierte el promedio de horas en texto con el sufijo "h" para mostrarse en las tarjetas KPI de la página 4.
7. Tasa Cierre Texto: convierte la tasa de cierre en texto con el símbolo "%" para mostrarse en las tarjetas KPI.
8. Abiertos Más 48h: cuenta los incidentes abiertos cuyas horas transcurridas superan las 48 horas, lo que indica casos que requieren atención prioritaria.
9. Incidentes Hoy: cuenta los incidentes cuya fecha coincide con la fecha del día actual en que se consulta el dashboard.

El segundo grupo contiene medidas de comparación temporal, que utilizan la tabla Calendario para funcionar correctamente:

10. Incidentes Mes Reciente: calcula el total de incidentes del último mes con datos disponibles en el sistema, usando MAXX con ALL para ignorar cualquier filtro de contexto.
11. Incidentes Mes Anterior: calcula el total del mes inmediatamente anterior al mes con más datos recientes, usando EDATE para retroceder un mes exacto.
12. Variación Mes %: compara el mes reciente con el anterior y expresa la diferencia como porcentaje. Devuelve BLANK si no hay datos del mes anterior para evitar errores de división.
13. Meta Cierre: valor constante de 70, que representa la meta mínima de tasa de cierre establecida para el servicio.
14. Meta 24h: valor constante de 24, que representa la meta máxima de horas de resolución por incidente.

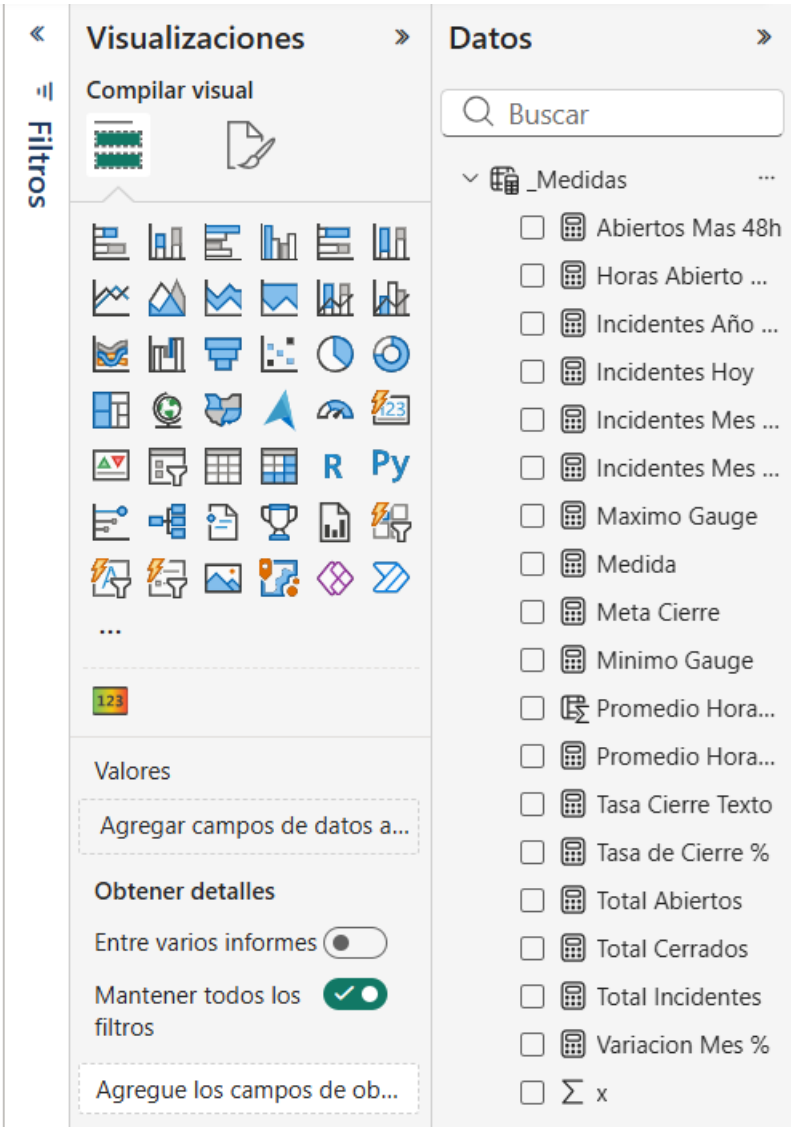


Figura 22 Medidas DAX organizadas en la tabla _Medidas en Power BI Desktop

Actividad 3.4 — Conexión entre SQL Server y Power BI

La conexión entre la base de datos y Power BI se configuró usando el conector nativo de SQL Server disponible en Power BI Desktop. El proceso implicó especificar el nombre del servidor de base de datos (localhost\SQLEXPRESS), el nombre de la base de datos (GestionIncidentes) y seleccionar la vista VW_Incidentes_PowerBI como fuente de datos principal.

Se eligió el modo de importación, que consiste en que Power BI carga una copia de los datos en su propia memoria y los actualiza cuando el usuario hace clic en el botón "Actualizar". Esta decisión garantiza que los dashboards respondan con rapidez incluso en equipos con recursos limitados, porque los cálculos se realizan sobre datos ya cargados en memoria y no dependen de una conexión constante al servidor. La alternativa, el modo DirectQuery, consulta la base de datos en tiempo real con cada interacción del usuario, lo cual es más exigente en términos de infraestructura.

Adicionalmente, se creó en Power BI una tabla Calendario mediante una fórmula DAX que genera automáticamente un rango de fechas desde el primer hasta el último incidente registrado, incluyendo columnas calculadas de año, mes, nombre del mes en español, número de semana, día de la semana en español y trimestre. Esta tabla se relacionó con la vista a través del campo Fecha con una cardinalidad de uno a muchos, y es la que habilita las medidas de comparación temporal. Sin la tabla Calendario, Power BI no puede realizar cálculos que impliquen comparar períodos distintos.

Durante el desarrollo se aplicó también el tema visual de SeguriLog, un archivo JSON con la paleta de colores del sistema (azul marino #1E3A5F y verde teal #2ECC9A como colores principales) que se cargó desde la opción Examinar temas del menú Vista, garantizando coherencia visual entre la aplicación web y los dashboards de análisis.

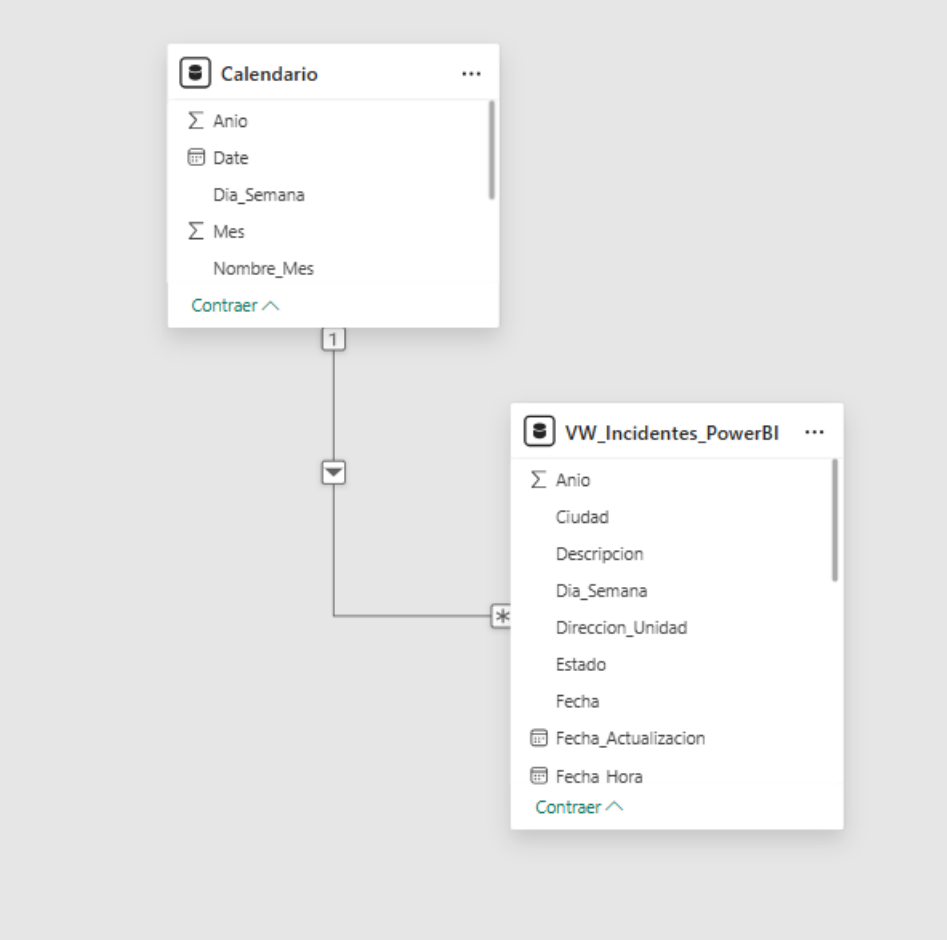


Figura 23 Relación entre la tabla Calendario y la vista VW_Incidentes_PowerBI.

Objetivo específico 4

Crear dashboards interactivos en Power BI que muestren indicadores clave sobre los incidentes: frecuencia, tipo, horario, ubicación y evolución en el tiempo.

Actividad 4.1 — Diseño y distribución de los dashboards

Antes de construir cualquier visualización, es necesario definir qué preguntas deben responder los dashboards y a quién van dirigidos. Few (2013) advierte que el error más común en el diseño de dashboards es querer mostrar todo lo que hay disponible en lugar de enfocarse en la información

que realmente importa para quien toma decisiones. Esa advertencia fue el punto de partida del diseño de los reportes de SeguriLog.

Se definió una estructura de cuatro páginas temáticas, cada una orientada a un tipo específico de análisis. La primera página ofrece una visión ejecutiva del sistema, diseñada para quien necesita entender el estado general de los incidentes en pocos segundos. La segunda página profundiza en la naturaleza de los eventos: qué tipos ocurren con más frecuencia y en qué zonas. La tercera página analiza el comportamiento temporal de los incidentes: a qué horas y en qué días de la semana se concentran. Y la cuarta página está orientada al seguimiento operativo: qué incidentes siguen abiertos, cuánto tiempo llevan sin resolverse y cómo está el desempeño del servicio.

Para el diseño visual se aplicó el tema de color de SeguriLog, que usa el azul marino oscuro (#1E3A5F) para los encabezados y elementos de énfasis, y el verde teal (#2ECC9A) como color principal de los gráficos y elementos positivos. El fondo de todas las páginas se configuró en gris azulado claro (#E8EFF6) para reducir la fatiga visual. Todos los visuales de cada página están conectados entre sí mediante filtros cruzados, lo que permite que al hacer clic en cualquier elemento de una gráfica, todos los demás visuales se actualicen automáticamente.

Actividad 4.2 — Descripción de cada página del reporte

Página 1 — Resumen ejecutivo

La primera página está diseñada para dar una fotografía instantánea del estado del sistema. En la parte superior se ubica el logo de SeguriLog junto a tres tarjetas KPI que muestran el total de incidentes abiertos, el total de cerrados y el total general de incidentes registrados. En la sección central se ubican dos gráficas complementarias: a la izquierda, un gráfico de barras agrupadas

muestra mes a mes los incidentes por estado (abiertos, cerrados y total), diferenciados por color; a la derecha, una gráfica de líneas muestra la tendencia acumulada del total de incidentes por mes.

En la parte inferior se presenta una gráfica de dona con la distribución de incidentes por estado (abiertos vs cerrados) con sus porcentajes, una tabla resumen por tipo de incidente con el total, los abiertos y la tasa de cierre de cada categoría, y tres segmentadores que permiten filtrar toda la página por año, mes y unidad residencial. Los segmentadores se configuraron en estilo mosaico con los colores del sistema para que los botones seleccionados se destaquen en verde teal. La página corresponde a los requisitos funcionales RF-11 y RF-12.

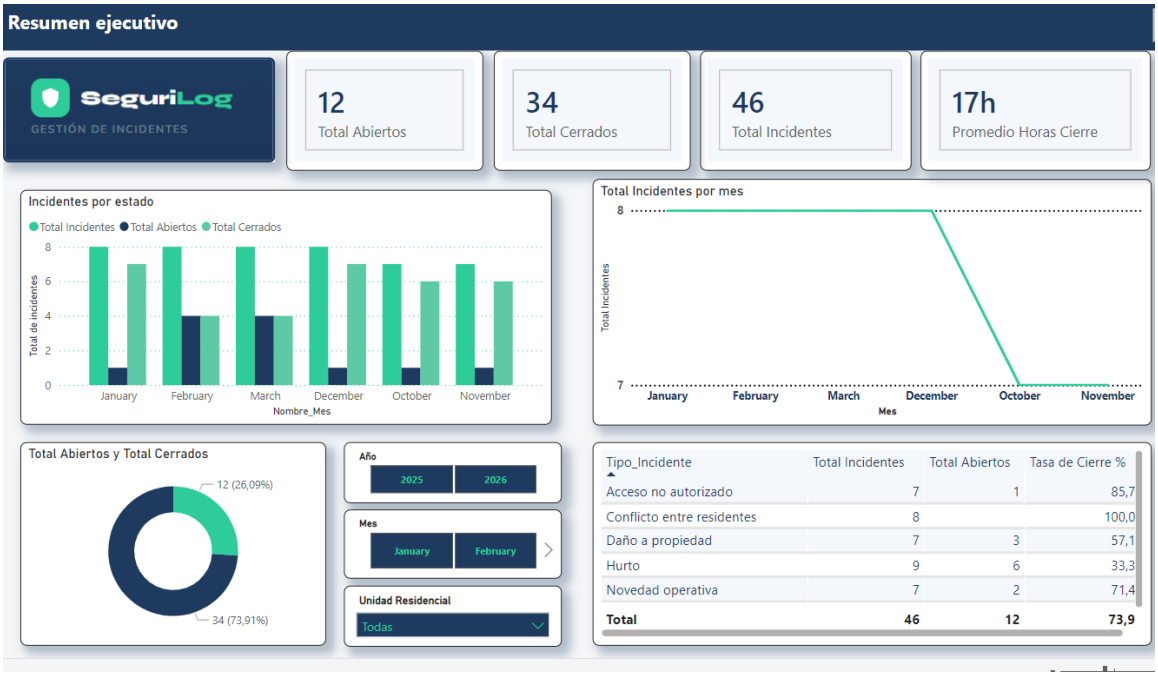


Figura 24 Página de resumen ejecutivo

Página 2 — Análisis por tipo y ubicación

La segunda página responde a las preguntas sobre la naturaleza y la distribución espacial de los incidentes. Un gráfico de barras horizontales presenta los tipos de incidente ordenados de mayor a menor frecuencia. Un gráfico de barras apiladas cruza los tipos de incidente con las unidades residenciales, mostrando cómo se distribuyen los distintos tipos de eventos en cada conjunto.

Un mapa de árbol (treemap) presenta la frecuencia por ubicación específica dentro del conjunto: parqueadero, portería, pasillos, salón comunal, entre otros. El tamaño de cada bloque es proporcional al número de incidentes, lo que hace muy evidente cuáles son los espacios de mayor riesgo. Una gráfica de dona muestra la participación porcentual de cada unidad residencial en el total de incidentes del período.

Figura 25 Página de análisis por tipo y ubicación

Página 3 — Análisis temporal

La tercera página responde a la pregunta de cuándo ocurren los incidentes. Un gráfico de barras con las 24 horas del día muestra en cuáles se concentra la mayor actividad. Un gráfico similar por día de la semana muestra el patrón semanal. Durante el análisis se identificó que los días martes y viernes concentran la mayor cantidad de incidentes, y que el horario de mayor ocurrencia corresponde a las 8, 9 y 22 horas.

Una matriz de calor cruza los meses del período con los tipos de incidente, usando una escala de colores donde el teal más oscuro indica mayor frecuencia. En la parte derecha de la página, cuatro tarjetas muestran el comparativo entre el mes con datos más recientes y el mes anterior, incluyendo la variación porcentual y el total acumulado del período. La página incluye segmentadores de año para filtrar el análisis por período.

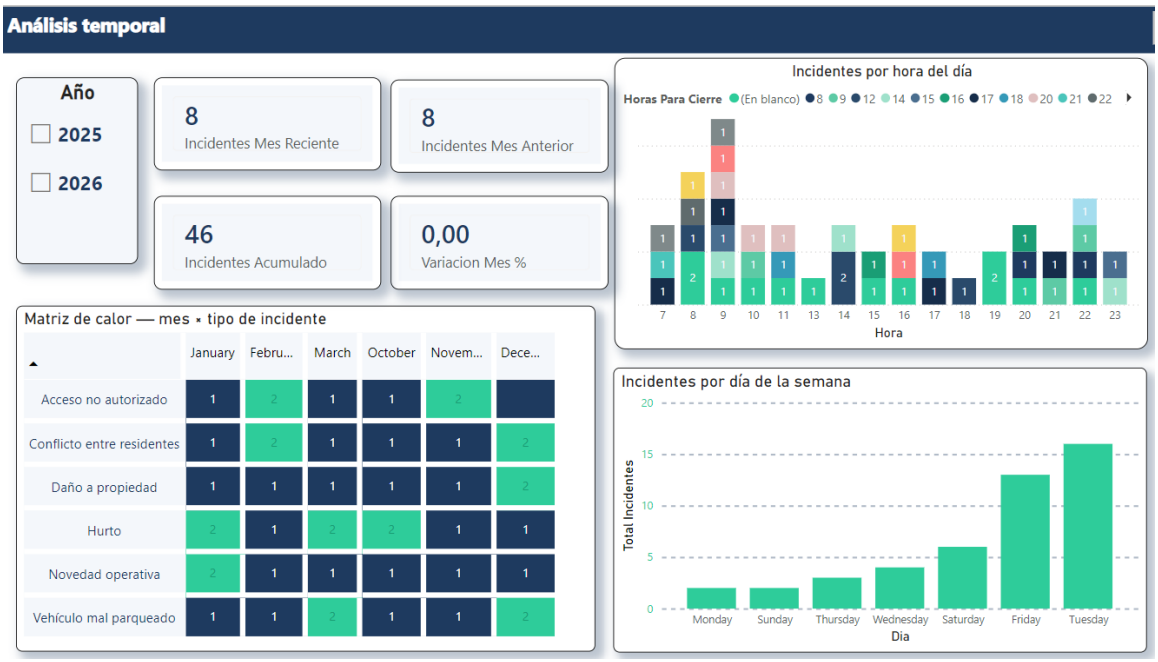


Figura 26 Página de análisis temporal

Página 4 — Gestión y seguimiento

La cuarta página está diseñada para el uso operativo diario del supervisor o administrador. En la parte superior, un logo de SeguriLog flanquea tres tarjetas KPI que muestran el promedio de horas de cierre con el sufijo "h" (17h), la tasa de cierre con el símbolo "%" (74%) y el total de incidentes abiertos. Las dos primeras tarjetas incluyen una indicación de la meta establecida: el promedio de horas debe ser menor o igual a 24 horas, y la tasa de cierre debe ser mayor o igual al 70%.

Un visual de medidor (gauge) muestra de manera gráfica la relación entre la tasa de cierre actual (74%) y la meta establecida (70%), con una línea de referencia punteada que marca el umbral. Un gráfico de barras horizontales muestra cuántos incidentes ha registrado cada vigilante durante el período, diferenciando los abiertos de los cerrados.

Una tabla detallada lista los incidentes que siguen abiertos, con las columnas ID, tipo, unidad residencial y horas transcurridas desde su registro. La columna de horas tiene formato condicional: los incidentes con más de 48 horas sin resolver se resaltan en rojo, los que llevan entre 24 y 48 horas en ámbar, y los recientes en verde. Un gráfico de barras horizontales por tipo de incidente muestra el promedio de horas de resolución de cada categoría, donde el conflicto entre residentes presenta el mayor tiempo promedio y el vehículo mal parqueado el menor. Esta página facilita la toma de decisiones inmediatas y el seguimiento del servicio en tiempo real.

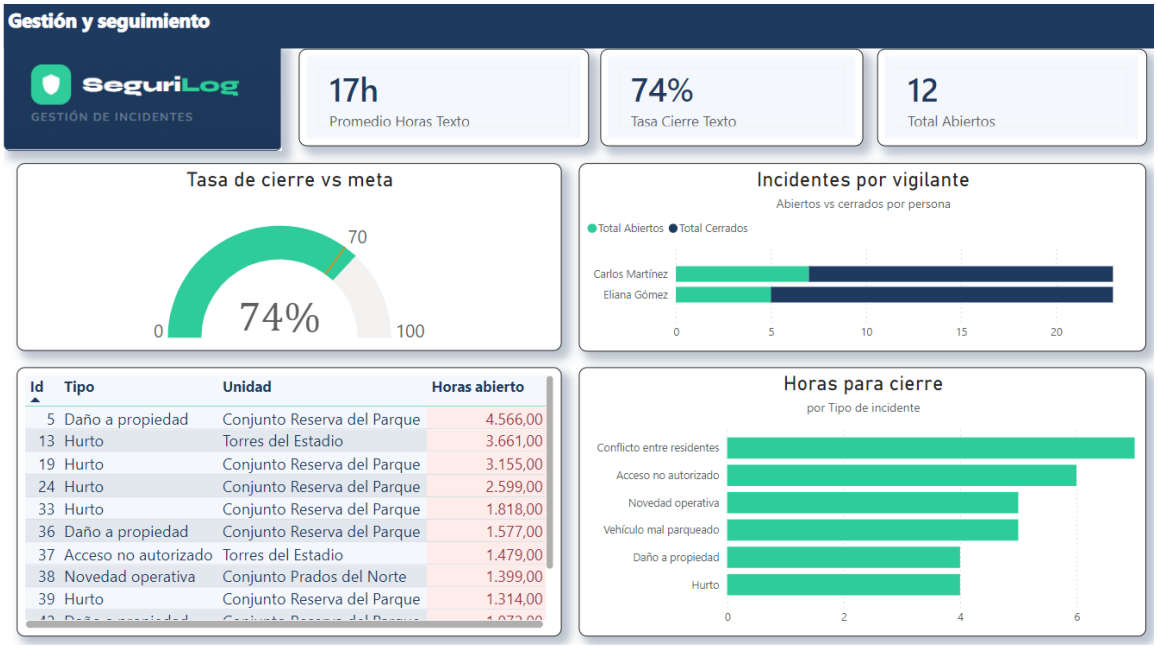


Figura 27 Página de gestión y seguimiento

Actividad 4.3 — Verificación del cumplimiento de los requisitos

Una vez construidos los dashboards, se verificó que lo entregado respondiera efectivamente a los requisitos definidos al inicio del proyecto. Esta verificación es importante porque cierra el ciclo entre lo que se propuso y lo que se construyó, y demuestra que el sistema no solo funciona técnicamente, sino que cumple con las expectativas establecidas.

En cuanto a los requisitos funcionales, los dashboards de Power BI dan respuesta directa a RF-11, que establece que el sistema debe mostrar indicadores KPI principales; a RF-12, que establece que debe incluir una gráfica de barras mensual diferenciando abiertos y cerrados; a RF-10, que establece que la información debe estar organizada con columnas de tiempo para Power BI; y a RF-09, que establece que el sistema debe permitir exportar los datos para análisis externo. El resto de los requisitos funcionales corresponden a la aplicación web y fueron verificados en las actividades del Objetivo 2.

En cuanto a los requisitos no funcionales, los dashboards se verificaron contra RNF-05 (tiempo de respuesta de las consultas menor a dos segundos), RNF-08 (correcta visualización en distintas resoluciones de pantalla) y RNF-16 (exportación de archivos sin errores de codificación en caracteres especiales del español). La Tabla 7 presenta el catálogo completo de requisitos no funcionales con su criterio de verificación correspondiente.

Tabla 7 Requisitos no funcionales

ID	Atributo	Categoría	Descripción	Cómo se verifica
RNF-01	Seguridad	Contraseñas	Las contraseñas se almacenan cifradas con BCrypt. Nunca se guardan como texto legible.	No existe ninguna contraseña en texto plano en la base de datos.
RNF-02	Seguridad	Acceso a la API	Todos los servicios del sistema requieren que el usuario haya iniciado sesión, excepto el login y la recuperación de contraseña.	Un intento sin sesión activa retorna error 401. Un rol sin permisos retorna error 403.
RNF-03	Seguridad	Código de verificación	El código enviado por correo para recuperar la contraseña expira en 10	Intentar usar un código ya usado o expirado muestra

			minutos y solo puede usarse una vez.	un mensaje de error.
RNF-04	Seguridad	Acceso desde otros sitios	El sistema solo acepta peticiones desde la dirección web autorizada del frontend, bloqueando accesos desde otros orígenes.	Las peticiones desde orígenes no autorizados son rechazadas automáticamente.
RNF-05	Rendimiento	Velocidad de respuesta	Las consultas de incidentes responden en menos de 2 segundos para bases de datos con hasta 10.000 registros.	Pruebas locales muestran tiempos de respuesta menores a 2 segundos.
RNF-06	Rendimiento	Paginación	Los resultados de la consulta se muestran de 12 en 12 para no cargar demasiada información a la vez.	Cada página muestra máximo 12 incidentes.
RNF-07	Rendimiento	Índices de búsqueda	La base de datos tiene índices en los campos más usados para buscar	Las consultas filtradas usan los índices y no hacen

			(fecha, unidad, tipo), lo que hace las consultas más rápidas.	una búsqueda completa de la tabla.
RNF-08	Usabilidad	Adaptabilidad	La interfaz se ajusta correctamente a pantallas de computador (1280px o más) y tabletas (768px).	Todas las funciones son accesibles sin desplazamiento horizontal en pantallas de 1280px.
RNF-09	Usabilidad	Mensajes al usuario	Después de guardar, editar, eliminar o exportar, el sistema muestra un mensaje informando si la acción fue exitosa o si hubo un error.	El mensaje aparece en menos de 500 milisegundos después de la acción.
RNF-10	Usabilidad	Validación de formularios	Los formularios avisan al usuario si un campo tiene un error antes de enviarlo, sin necesidad de recargar la página.	No es posible enviar un formulario con campos vacíos o

				con información inválida.
RNF-11	Mantenibilidad	Organización del código	El sistema está organizado en capas separadas: una para la interfaz, otra para las reglas de negocio y otra para el acceso a los datos.	Cada parte del sistema tiene una responsabilidad clara y no mezcla funciones de otras capas.
RNF-12	Mantenibilidad	Separación de datos	La información que se envía al usuario desde la API no expone directamente la estructura interna de la base de datos.	Las tablas internas de la base de datos no se muestran directamente en los servicios del sistema.
RNF-13	Disponibilidad	Manejo de errores	Cuando ocurre un error, el sistema responde con un mensaje claro y no muestra información técnica al usuario.	Ningún error genera una pantalla en blanco ni un mensaje de sistema incomprensible para el usuario.

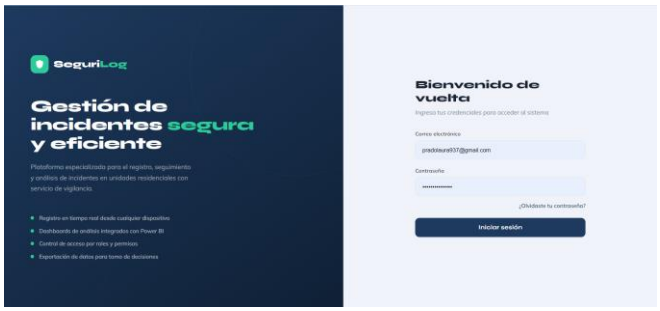
RNF-14	Disponibilidad	Reconexión automática	Si la conexión con la base de datos falla momentáneamente, el sistema intenta reconectarse hasta 3 veces antes de mostrar un error.	El sistema se recupera automáticamente de interrupciones breves de conexión.
RNF-15	Compatibilidad	Navegadores	El sistema funciona correctamente en las versiones actuales de Chrome, Edge y Firefox, sin necesidad de instalar extensiones adicionales.	Todas las funciones se verificaron en Chrome 120+, Edge 120+ y Firefox 120+.
RNF-16	Compatibilidad	Archivos exportados	Los archivos CSV descargados muestran correctamente los caracteres especiales del español (tildes, ñ) al abrirlos en Excel y Power BI.	El archivo abierto en Excel muestra todos los caracteres del español sin errores de codificación.

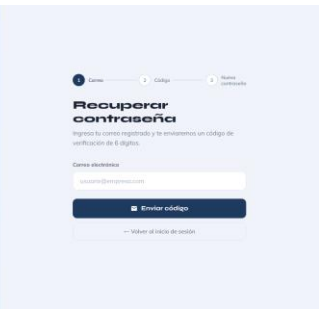
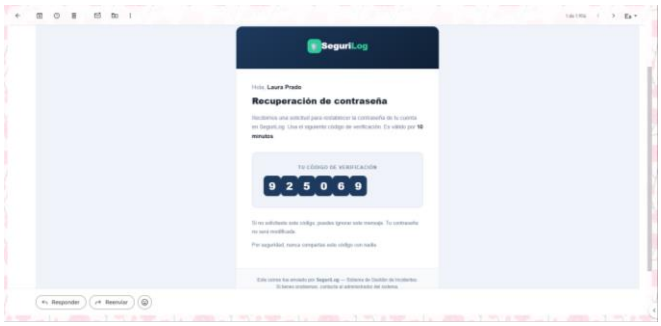
Pruebas funcionales del sistema

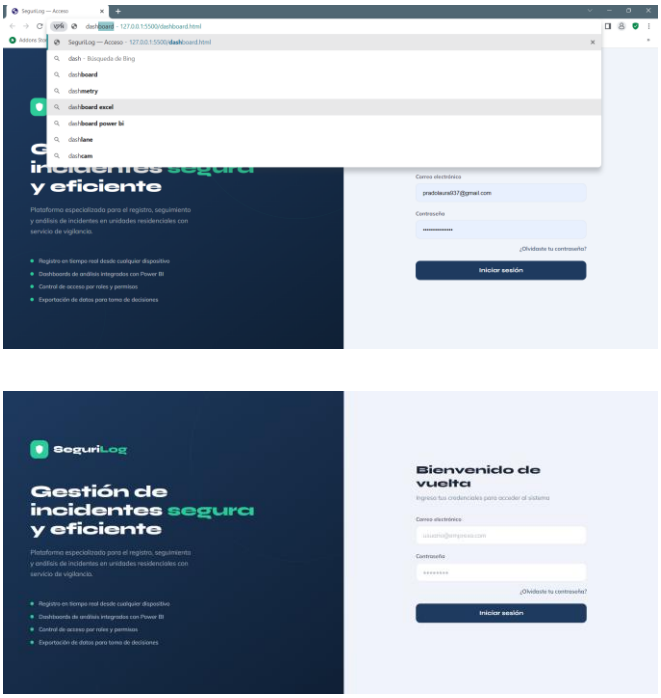
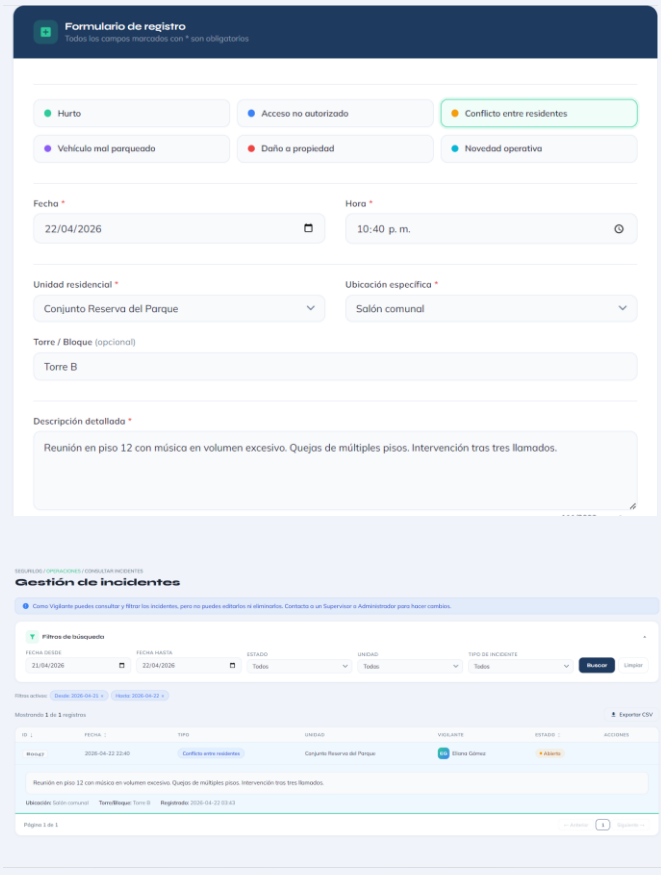
Estas pruebas tuvieron como objetivo verificar que cada uno de los requisitos funcionales definidos en el diseño del sistema SeguriLog se cumpliera correctamente. Para cada prueba se documenta el módulo al que pertenece, el requisito funcional (RF) que valida, la descripción de los pasos ejecutados, el resultado esperado, la evidencia fotográfica obtenida y el estado de la prueba.

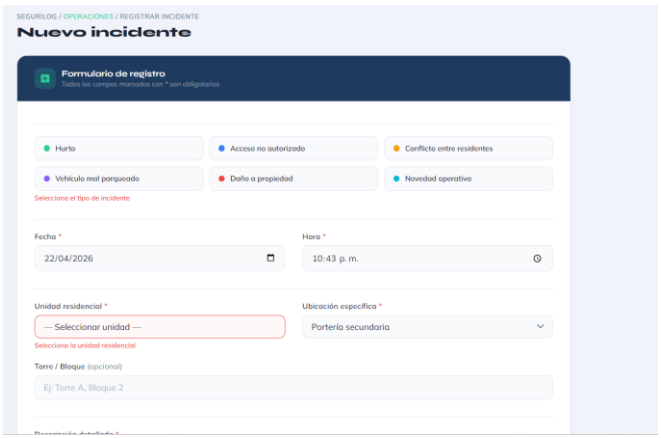
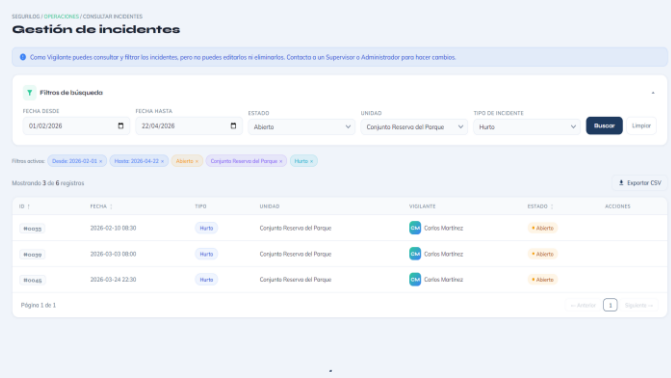
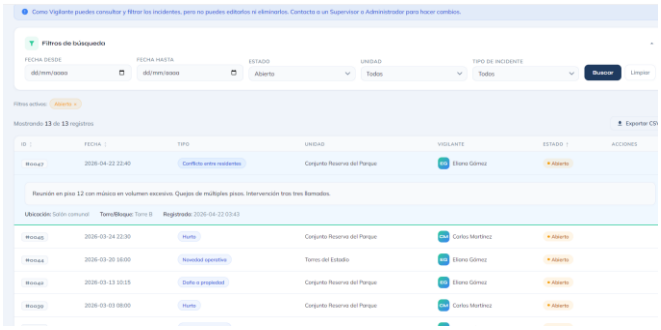
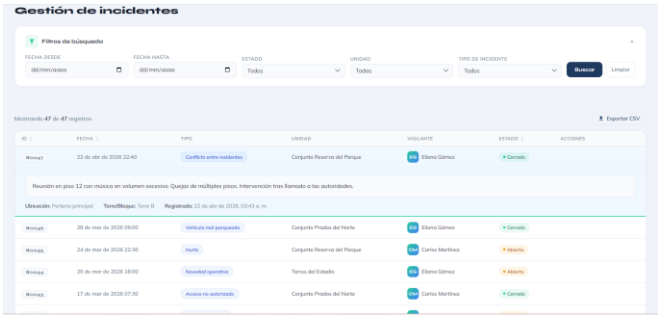
En total se realizaron 21 pruebas distribuidas en seis módulos: autenticación, gestión de incidentes, exportación de datos, dashboard, administración y control de acceso por roles. Todas las pruebas se ejecutaron en entorno local con datos de demostración. El estado ✓ indica que la prueba fue superada satisfactoriamente.

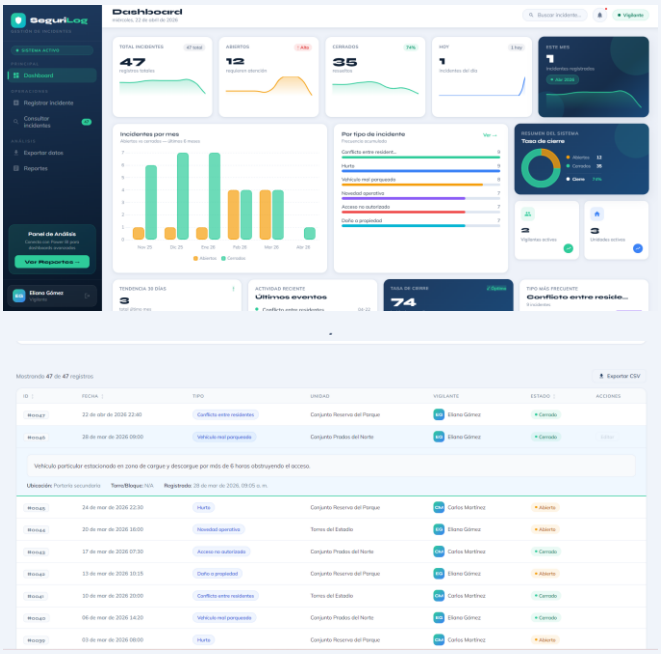
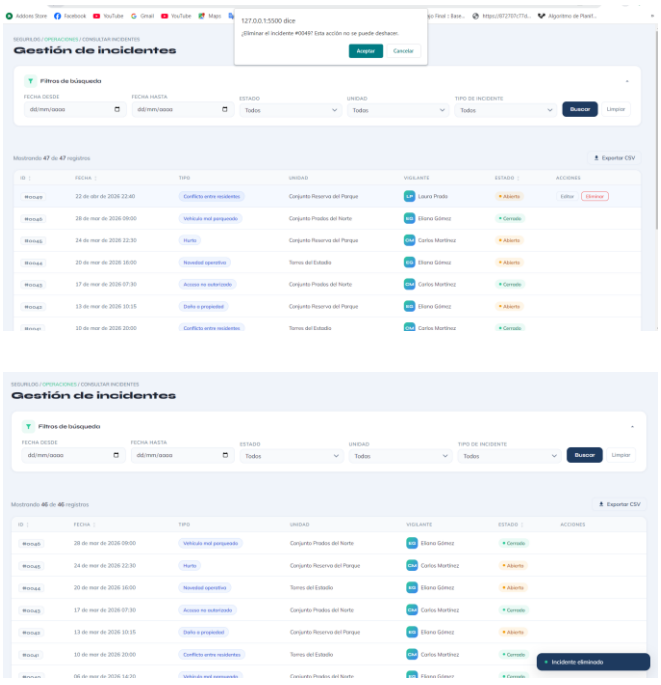
Tabla 8 Pruebas funcionales del sistema SeguriLog

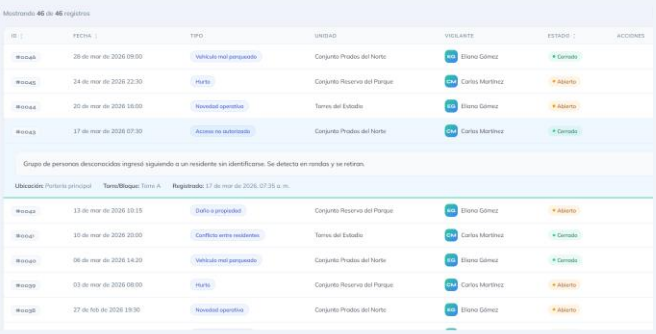
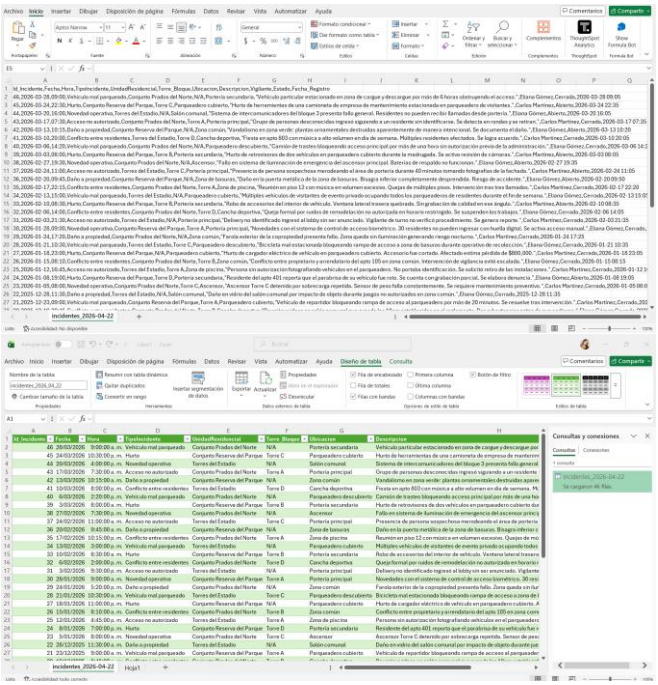
ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
1	RF-01	**Inicio de sesión exitoso** Ingresar correo y contraseña correctos y hacer clic en "Ingresar".	El sistema valida las credenciales, genera el token de sesión y redirige automáticamente al dashboard. El nombre del usuario y su rol aparecen visibles en la barra superior.	 	✓

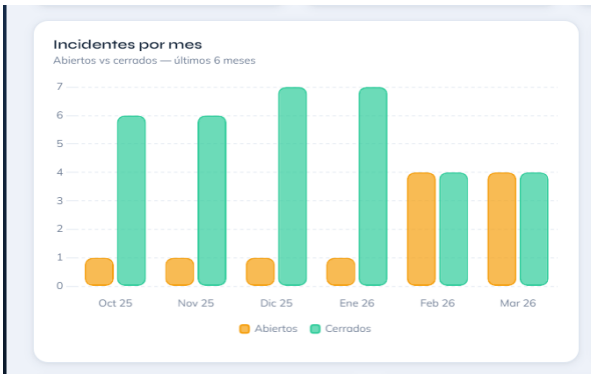
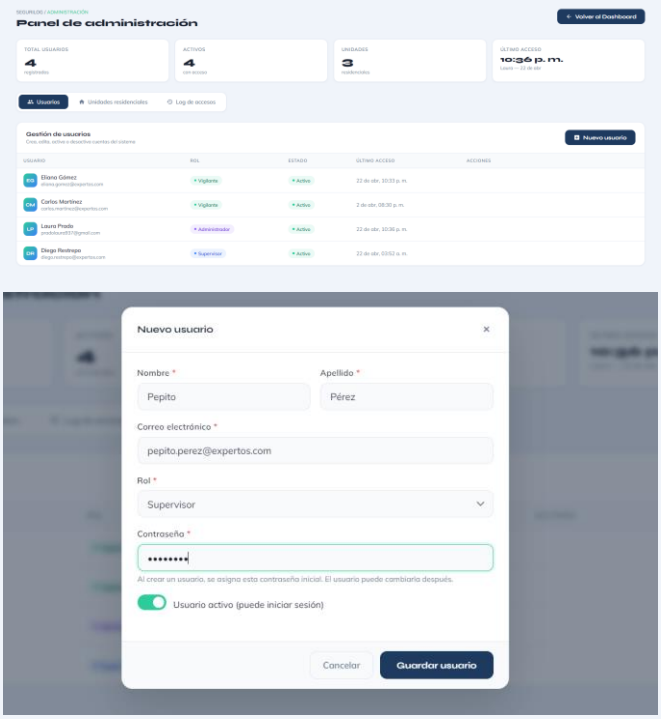
ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
2	RF-01	**Inicio de sesión con credenciales incorrectas** Ingresar una contraseña incorrecta para un usuario registrado.	El sistema muestra un mensaje de error en la pantalla. No permite el acceso al sistema ni revela si el error está en el correo o la contraseña.	 	✓
3	RF-02	**Recuperación de contraseña — flujo completo** Hacer clic en "¿Olvidaste tu contraseña?", ingresar el correo, verificar el código OTP de 6 dígitos recibido por correo y establecer una nueva contraseña.	El sistema envía el código al correo en menos de un minuto. Acepta el código correcto dentro de los 10 minutos de vigencia. Actualiza la contraseña y permite iniciar sesión con la nueva clave.	  	✓
4	RF-02	**Código OTP expirado o ya usado** Ingresar un código OTP que ya fue utilizado o que superó los 10 minutos de vigencia.	El sistema rechaza el código y muestra un mensaje indicando que es inválido o que expiró. No permite avanzar al paso de nueva contraseña.	 	✓

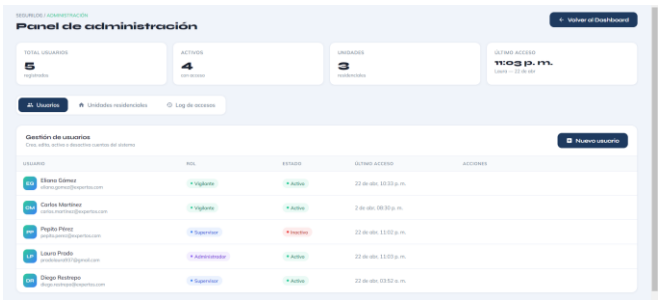
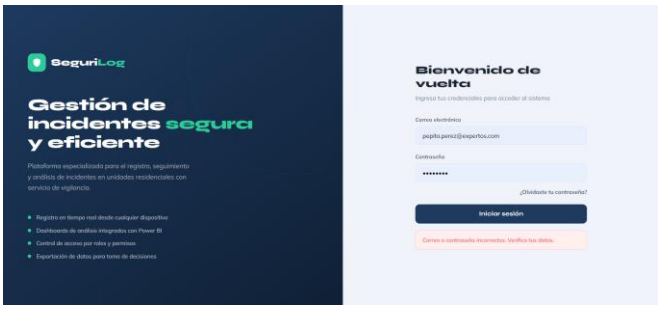
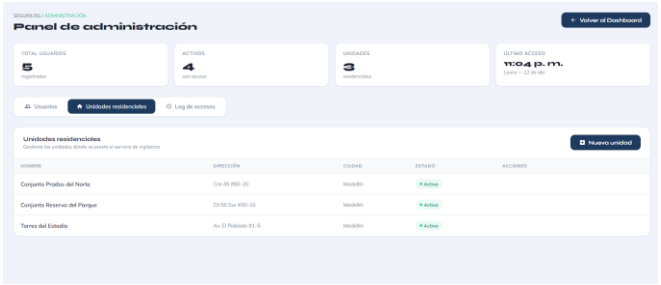
ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
5	RF-03	**Redirección por sesión no iniciada** Intentar acceder directamente a la URL del dashboard sin haber iniciado sesión.	El sistema detecta que no hay sesión activa y redirige automáticamente a la pantalla de login. No se muestra ningún contenido protegido.		✓
6	RF-04	**Registro de incidente completo** Completar todos los campos del formulario (tipo, unidad, ubicación, fecha, hora y descripción) y hacer clic en "Guardar incidente".	El incidente queda registrado con estado "Abierto" y con el nombre del vigilante que lo reportó. El sistema muestra un mensaje de confirmación y limpia el formulario.		✓

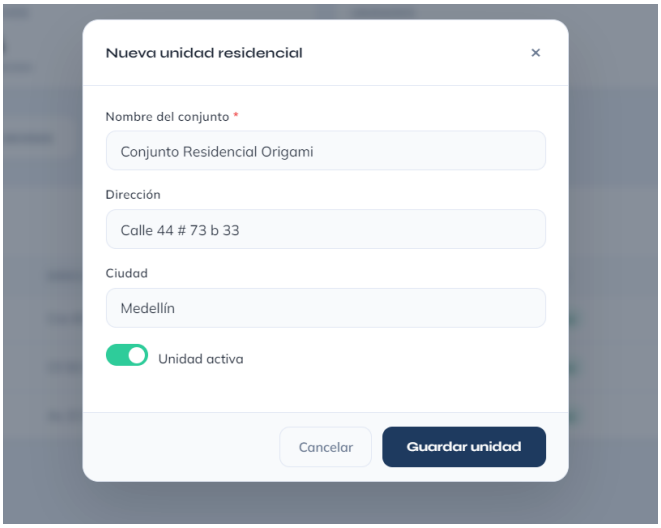
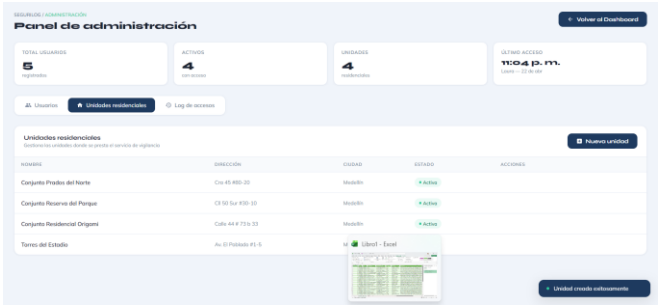
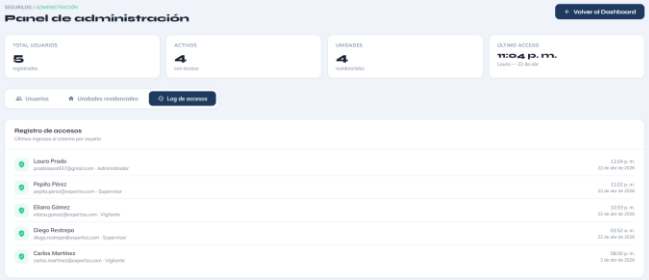
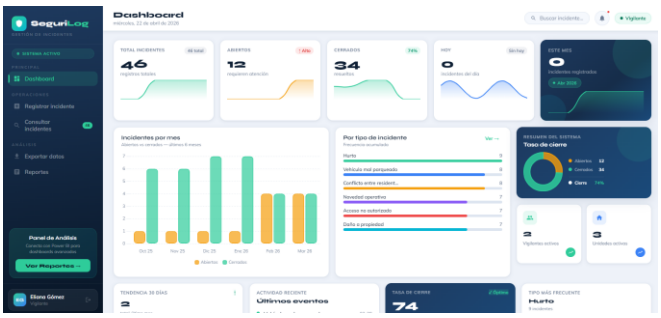
ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
7	RF-04	**Validación de campos obligatorios** Intentar guardar un incidente dejando campos requeridos vacíos.	El sistema muestra mensajes de error junto a cada campo incompleto. No envía el formulario ni pierde la información ya ingresada.		✓
8	RF-05	**Consulta con filtros combinados** Aplicar simultáneamente filtros de fecha, estado y tipo de incidente en el módulo de consulta.	La tabla muestra únicamente los incidentes que cumplen todos los criterios aplicados. Los chips de filtros activos son visibles y cada uno puede eliminarse individualmente.		✓
9	RF-06	**Edición de incidente — Supervisor** Iniciar sesión como Supervisor, seleccionar un incidente abierto y modificar la descripción y el estado a "Cerrado".	Los cambios quedan guardados correctamente. El incidente aparece con estado "Cerrado" en la tabla de consulta. El sistema muestra un toast de confirmación.	 	✓

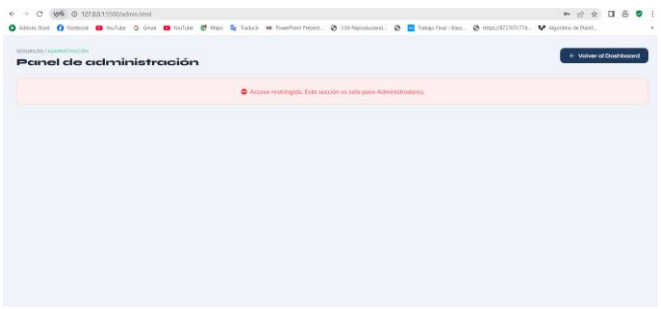
ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
10	RF-06	**Intento de edición por Vigilante — acceso denegado** Iniciar sesión como Vigilante e intentar editar un incidente.	El botón de editar no aparece en la tabla para el rol Vigilante. El sistema no permite realizar ninguna modificación sobre los registros.		✓
11	RF-07	**Eliminación de incidente — Administrador** Iniciar sesión como Administrador, seleccionar un incidente y hacer clic en eliminar.	El sistema solicita confirmación antes de eliminar. Al confirmar, el incidente desaparece de la tabla y no puede recuperarse. El sistema muestra un mensaje de éxito.		✓

ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
12	RF-08	**Ver detalle completo de un incidente** Hacer clic en una fila de la tabla de consulta para expandirla.	La fila se expande mostrando la descripción completa, la ubicación específica, la torre/bloque y la fecha de registro. No se abre ninguna página nueva.		✓
13	RF-09	**Exportación de datos a CSV** Aplicar un filtro de fechas en el módulo de consulta y hacer clic en "Exportar datos".	Se descarga un archivo CSV con los incidentes del filtro activo. Al abrir el archivo en Excel, los caracteres especiales del español (tildes, ñ) se muestran correctamente sin errores de codificación.		✓
14	RF-11	**Visualización de KPIs en el dashboard** Iniciar sesión con cualquier rol y acceder al dashboard principal.	Las cuatro tarjetas KPI muestran valores actualizados: total de incidentes, abiertos, cerrados e incidentes del día. Cada tarjeta incluye un indicador de tendencia comparado con el período anterior.		✓

ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
15	RF-12	**Gráfica de incidentes por mes** Acceder al dashboard y verificar la gráfica de barras mensual.	La gráfica muestra los incidentes de los últimos seis meses con barras diferenciadas por color para abiertos y cerrados. Los datos se actualizan automáticamente en cada carga de la página.		✓
16	RF-13	**Crear nuevo usuario** Desde el panel de administración, crear un usuario con rol Supervisor completando todos los campos requeridos.	El usuario aparece en la lista con el rol asignado y estado activo. El nuevo usuario puede iniciar sesión con las credenciales asignadas y accede con los permisos de Supervisor.		✓

ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
17	RF-13	**Desactivar cuenta de usuario** Desde el panel de administración, desactivar la cuenta de un usuario activo haciendo clic en el toggle de estado.	El usuario aparece como "Inactivo" en la lista. Al intentar iniciar sesión con esa cuenta, el sistema muestra un error y deniega el acceso.	 	✓
18	RF-14	**Cambiar contraseña de un usuario** Desde el panel de administración, cambiar la contraseña de un usuario sin necesidad de conocer la contraseña actual.	El cambio se aplica correctamente. El usuario puede iniciar sesión con la nueva contraseña de inmediato.		✓
19	RF-15	**Crear unidad residencial** Desde la pestaña de unidades en el panel de administración, crear una nueva unidad residencial.	La unidad aparece en la lista del panel de administración. La nueva unidad también aparece disponible en el selector del formulario de registro de incidentes.		✓

ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
				 	
20	RF-16	**Consultar log de accesos** Desde el panel de administración, acceder a la pestaña de log de accesos.	El sistema muestra los últimos 50 accesos con el nombre del usuario, correo, rol y fecha/hora. Los registros están ordenados del más reciente al más antiguo.		✓
21	RF-17	**Control de acceso — menú oculto para Vigilante** Iniciar sesión como Vigilante y revisar el menú lateral del sistema.	La opción "Administración" no aparece en el menú lateral. Al intentar acceder directamente por URL al panel de administración, el sistema niega el acceso.		✓

ID	RF	DESCRIPCIÓN	RESULTADO ESPERADO	EVIDENCIA	ESTADO
					

Conclusiones

El desarrollo de SeguriLog confirmó algo que el diagnóstico inicial ya sugería: en la gestión de la seguridad residencial, el problema no es la falta de datos, sino la ausencia de un sistema que los haga utilizables. Los métodos tradicionales de registro —cuadernos, hojas de cálculo sin estructura— generan información que se acumula sin aprovecharse, lo que mantiene a las organizaciones en un modo reactivo que limita su capacidad de anticiparse a los riesgos.

La encuesta aplicada a residentes mostró una disposición muy alta hacia el uso de herramientas digitales, con un 100% de aceptación. Más relevante aún fue la demanda explícita de funcionalidades que vayan más allá del simple registro: los usuarios quieren poder analizar la información, visualizarla en gráficos e identificar patrones. Eso validó directamente el enfoque del proyecto.

Desde el punto de vista técnico, la arquitectura en capas demostró ser la elección correcta: permitió construir un sistema organizado, mantenible y con capacidad de escalar. La integración con Power BI fue el puente que convirtió los registros operativos en información analítica, haciendo visibles patrones que antes permanecían ocultos en tablas sin estructura.

En términos generales, SeguriLog demuestra que incorporar tecnologías de información en la gestión de la seguridad privada no es un lujo ni una complicación: es un cambio de enfoque que transforma la toma de decisiones, reduce la dependencia del criterio individual y abre la puerta a una gestión más eficiente, más documentada y orientada a la prevención.

Referencias

Agile Alliance. (2001). *Manifesto for agile software development*. https://agilemanifesto.org/

Al-Shehri, M. (2024). Big data challenge for monitoring quality using business intelligence dashboards. *Journal of Electronic Science and Technology*.

Cabral, L., Pinto, R., & Gonçalves, G. (2025). AI-powered learning analytics dashboards: A systematic review. *Discover Education*. https://doi.org/10.1007/s44217-025-00964-y

Cardona-Román, D. M., Colmenares, A., & Guina, E. (2024). Business intelligence dashboard as a technological innovation for analysis on digital transformation.

CEPAL. (2022). *Seguridad y tecnología en América Latina*. Comisión Económica para América Latina y el Caribe.

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*.

Davenport, T. H. (2018). *Analytics at work: Smarter decisions, better results*. Harvard Business Review Press.

Few, S. (2013). *Information dashboard design: Displaying data for at-a-glance monitoring* (2.ª ed.). Analytics Press.

Fowler, M. (2002). *Patterns of enterprise application architecture*. Addison-Wesley.

García, M., & López, R. (2021). Dashboards de monitoreo en seguridad privada: impacto en tiempos de respuesta y asignación de recursos. *Revista Latinoamericana de Seguridad Ciudadana*, *14*(2), 45–62.

Gartner. (2023). *Magic quadrant for analytics and business intelligence platforms*. Gartner Research. https://www.gartner.com/en/documents/4218175

Kimball, R., & Ross, M. (2013). *The data warehouse toolkit: The definitive guide to dimensional modeling* (3rd ed.). Wiley.

Knaflic, C. N. (2015). *Storytelling with data: A data visualization guide for business professionals*. Wiley.

Laudon, K. C., & Laudon, J. P. (2016). *Management information systems: Managing the digital firm*. Pearson.

McKinsey Global Institute. (2021). *The state of data-driven organizations*.

O'Brien, J. A., & Marakas, G. M. (2019). *Management information systems*. McGraw-Hill.

Provost, F., & Fawcett, T. (2018). *Data science for business*. O'Reilly Media.

Ramírez-Aristizábal, J., Mejía-Triana, C., & Zapata-Madrigal, D. (2024). Business intelligence in organizational decision-making: A bibliometric analysis of research trends and gaps (2014–2024). *Discover Sustainability*. https://doi.org/10.1007/s43621-024-00692-7

Schmidt, R., Brünken, R., & Scheiter, K. (2024). Organizational decision making and analytics: An experimental study on dashboard visualizations. *Information & Management*, *61*(6). https://doi.org/10.1016/j.im.2024.104011

Secretaría de Seguridad y Convivencia de Medellín. (2023). *Informe de seguridad ciudadana: Hurtos en unidades residenciales*. Alcaldía de Medellín.

Sharda, R., Delen, D., & Turban, E. (2019). *Business intelligence, analytics, and data science: A managerial perspective*. Pearson.

Singh, P., Rathore, B., & Sharma, B. (2025). *Business intelligence tools in 2024: A comparative analysis and market insights*.

Anexo 1

Video de sustentación: https://www.youtube.com/watch?v=SQygiT2BuP0