



PROYECTO DE GRADO

Desarrollo de un sistema integral para la automatización de la gestión de usuarios, inventario, recaudo, facturación y administración de dispositivos MikroTik con monitoreo en tiempo real en la empresa NetLine Colombia ISP

Corporación Universitaria Remington.

Facultad de Ingeniería.

Programa Ingeniería de Sistemas.

Eduard Bayardo Insuasty Insuasty

Deivid Andrés Melo Cerón.

Dr. Nelson Ricardo Zambrano Martínez

Asesor

Proyecto de Grado

2026

Dedicatoria

A mi mayor fuente de inspiración y amor: mi hijo Gerónimo Insuasty. Eres la luz que ha dado sentido a mi vida. Sé que, mientras atravesábamos esta aventura, tuve que reemplazar el juego con libros y estudio, pero quiero que sepas que este tiempo no fue en vano, fue una semilla para tu futuro. Y cada minuto que estuviste lejos será más que compensado, se utilizará como herramientas para crear un mejor camino para ti, te hará sentir que podré darte tiempo de calidad con tranquilidad y el verdadero ejemplo de que los grandes sueños se harán realidad al final. Pongo ese logro profesional en tus manos para mostrarte que la educación es el motor para hacer nuestra realidad. ¡Siempre serás mi mayor proyecto!

Mis queridos padres Libardo Melo y Nubia Cerón ¡que son el fundamento de mi vida y mis pasos! Mi logro no es el resultado solo de mí, sino por sus noches en vela, sus palabras amorosas en tiempos difíciles, su amor incondicional y su fe en mí. Iba a la biblioteca y no me quedaba en casa estudiando mucho y los extrañaba. Su refugio seguro y su aliento para nunca rendirme, siento que por eso asumo este título profesional, estoy aquí tan agradecido con ustedes.

Agradecimientos

En primer lugar, nuestro más sincero y sentido agradecimiento a la Corporación Universitaria Remington y en particular al programa de Ingeniería de Sistemas. Gracias por brindarnos un espacio de excelencia académica, por los recursos proporcionados y por ser la institución que sentó las bases de nuestra formación profesional.

Queremos expresar nuestro más profundo agradecimiento al asesor y supervisor del proyecto, el Dr. Nelson Ricardo Zambrano Martínez, por toda la orientación que nos brindó, su paciencia y su constante compromiso con este esfuerzo. Estamos agradecidos por la generosidad que nos mostró, compartiendo su experiencia y conocimiento con nosotros. El apoyo que nos brindó ha sido crucial no solo para abordar los desafíos técnicos y metodológicos que planteó esta investigación, sino también para implementar todo ese conocimiento y crear un sistema concreto y exitoso para la empresa ISP NetLine Colombia. Tal logro profesional simboliza sin duda la excelencia de su vocación docente.

Por último, y lo más importante, nos gustaría agradecer a todas las personas, colegas y familiares que, tanto directa como indirectamente, nos brindaron su apoyo y motivación a lo largo de nuestro Proyecto de Graduación.

Tabla de Contenidos

Resumen.....	12
Palabras clave:	13
1 Introducción, marco teórico o de referencia	13
2 Planteamiento del problema.....	15
3 Objetivos	16
3.1 Objetivo General	16
3.2 Objetivos Específicos.....	16
4 Metodología	17
4.1 Metodología de la investigación	17
4.2 Metodología de Desarrollo	18
5 Resultados y Discusión	22
5.1 Procesos Administrativos y Operativos de Netline Colombia ISP	22
5.2 Requerimientos Funcionales y no Funcionales del Sistema.	23
5.3 Requerimientos no funcionales (RNF)	24
5.4 Diseño de la Arquitectura del Software: Estructura, Tecnologías, Base de Datos e Integración con Dispositivos Mikrotik	25
5.5 Arquitectura Cliente-Servidor.....	32
5.6 Tecnologías orientadas al servidor (Backend).....	32

5.7	Tecnologías del lado del cliente (Frontend).....	33
5.8	Bibliotecas y APIs para la gestión de colas	34
5.9	Desarrollo e Implementación de los Módulos del Sistema AdminISP	35
5.10	Módulo de clientes y usuarios.....	35
5.11	Módulo de planes	39
5.12	Módulo de facturación	42
5.13	Módulo de recaudos.....	46
5.14	Módulo de ingresos y gastos.....	50
5.15	Módulo de inventario de activos.....	52
5.16	Módulo de Integración con Mikrotik (Queues)	56
5.17	Módulo de monitoreo de red.....	59
5.18	Portal de clientes	69
5.19	Evaluación de la Calidad del Software AdminISP	70
5.20	Pruebas de usabilidad.....	80
5.21	Pruebas de Seguridad y Acceso.	91
Conclusiones		95
Referencias.....		97
Anexos		100

Lista de tablas

Tabla 1 <i>Resultados SUS - Personal Administrativo y Técnico</i>	87
Tabla 2 <i>Resultados SUS - Clientes.....</i>	88
Tabla 3 <i>Control de Incidencias y Corrección de Errores.....</i>	90

Lista de figuras

Figura 1 <i>Diagrama caso de uso - General</i>	25
Figura 2 <i>Diagrama Entidad Relación</i>	26
Figura 3 <i>Diagrama de clase (clientes – facturación)</i>	27
Figura 4 <i>Diagrama UML de clases (clientes e inventario)</i>	28
Figura 5 <i>Diagrama de secuencia (habilitación del servicio)</i>	29
Figura 6 <i>Módulo de clientes y usuarios – interfaz gráfica para registrar y editar</i>	37
Figura 7 <i>Fragmento de código - Registro de un usuario de gestión o cliente</i>	38
Figura 8 <i>Fragmento de código - Lógica para el Registro y Validación de Planes de Servicio</i> .	40
Figura 9 <i>Módulo de Planes – interfaz grafica</i>	40
Figura 10 <i>Fragmento de código - Lógica para la Actualización y Modificación de Planes de servicio</i>	41
Figura 11 <i>Fragmento de código -Lógica para la Eliminación de Planes de Servicio</i>	41
Figura 12 <i>Módulo de Facturación – interfaz grafica</i>	43
Figura 13 <i>Ventana prefacturación – interfaz grafica</i>	44
Figura 14 <i>Fragmento de código - Lógica para la Ejecución y Registro de Facturación Masiva</i>	45
Figura 15 <i>Editar el valor de una factura generada.</i>	45
Figura 16 <i>Fragmento de código - Lógica para la Actualización de Montos en Facturas Individuales</i>	46
Figura 17 <i>Módulo de Recaudo – interfaz grafica</i>	48
Figura 18 <i>Historial de facturas Pagadas/Anuladas</i>	49

Figura 19 <i>Fragmento de código – Actualizar el estado de la factura</i>	49
Figura 20 <i>Módulo de Ingresos y Gastos – Interfaz Grafica</i>	51
Figura 21 <i>Fragmento de código: Modulo de ingresos y gastos - Lógica de Backend para el Inicio de Sesión y Autenticación de Usuarios</i>	52
Figura 22 <i>Módulo de Inventario de Activos – Interfaz Gráfica</i>	54
Figura 23 <i>Tipo de activo</i>	54
Figura 24 <i>Asignar un activo</i>	55
Figura 25 <i>Agregar nuevo tipo de activo y adicionar cantidades</i>	55
Figura 26 <i>Fragmento de código - Adicionar un Activo al inventario - Lógica de Backend para el Cierre de Sesión y Finalización de Actividad</i>	56
Figura 27 <i>Módulo de Integración con MikroTik (Queues) – interfaz Gráfica</i>	57
Figura 28 <i>Fragmento de código - Módulo de Queues - Función de Utilidad para la Verificación de Conectividad (Ping)</i>	58
Figura 29 <i>fragmento de código de acciones de las Queues simple - Lógica para la Administración de Queue Simple y Queue Tree</i>	59
Figura 30 <i>Módulo de Monitoreo de Red – interfaz gráfica (mapa de red)</i>	60
Figura 31 <i>Ventana Emergente de Autenticación para la Administración del Panel</i>	61
Figura 32 <i>Panel de Administración de Dispositivos y Configuración de Conexiones</i>	61
Figura 33 <i>Interfaz del Panel de Administración de Clientes y Edición de Dispositivos</i>	62
Figura 34 <i>Interfaz del Panel para la Administración y Edición de Conexiones de Red</i>	63
Figura 35 <i>Fragmento de código - Lógica para la Obtención y Ordenamiento de Dispositivos de Red</i>	64
Figura 36 <i>Visualización del Mapa de Red Durante el Proceso de Sincronización Activa</i>	64

Figura 37 Fragmento de código para agregar dispositivos	65
Figura 38 Fragmento de código para eliminar dispositivos.....	65
Figura 39 Fragmento de código para agregar conexiones - Lógica para la Actualización de Dispositivos de Red	66
Figura 40 Fragmento de código para editar conexiones	66
Figura 41 Fragmento de código para eliminar conexiones.....	67
Figura 42 Código del RouterOS API.....	67
Figura 43 Interfaz web para la interacción con el usuario	69
Figura 44 Interfaz web para la consulta y pago de facturas.....	70
Figura 45 Formulario de registro de un nuevo cliente en el sistema	71
Figura 46 Persistencia del registro del cliente en la base de datos	72
Figura 47 Interfaz de Revisión y Edición de Facturación Masiva.....	72
Figura 48 Persistencia de los datos en la interfaz de usuario.....	73
Figura 49 Mensaje de confirmación.	73
Figura 50 Formulario de registro y alta de nueva antena en el inventario.....	75
Figura 51 Asignación de activo de telecomunicaciones al perfil del cliente.....	75
Figura 52 Actualización automática del stock en bodega tras la asignación y la persistencia en la base de datos	76
Figura 53 Interfaz para la creación, edición de queues.....	77
Figura 54 tabla de queues en el software y en el aplicativo winbox	78
Figura 55 Mapa de red después del reinicio	79
Figura 56 Mapa de red antes del reinicio e interfaz de reinicio del dispositivo.	79
Figura 57 interfaz del módulo de ingresos y gastos.....	81

Figura 58 Interfaz del módulo de recaudo.	81
Figura 59 Modal de cambio de estado y recibo de pago.	82
Figura 60 Interfaz de inicio de sesión a un mikrotik e interfaz de administración de las queues.	83
Figura 61 Interfaz de administración de las queues.....	84
Figura 62 Módulo de monitoreo e interfaz para la edición de dispositivos.....	85
Figura 63 Formulario para la edición de los datos y mapa de red.	86
Figura 64 Bloqueo de acceso por rol insuficiente en el Módulo de Finanzas.	93
Figura 65 Módulo de Búsqueda de Usuarios.....	94

Lista de Anexos

Anexo 1 Cuestionario de Entrevista a Personal Técnico y Administrativo de Netline Colombia ISP	100
Anexo 2 Resultado del Cuestionario de Entrevista a Personal Técnico y Administrativo de Netline Colombia ISP	101
Anexo 3 Cuestionario SUS para AdminISP – Para Administrativos y Técnicos.....	103
Anexo 4 Cuestionario SUS para AdminISP – Para Clientes	104

Resumen

Esta investigación se enmarca en un enfoque pragmático ya que su objetivo principal no es solo proporcionar nuevo conocimiento teórico, sino abordar la gestión práctica y efectiva de los problemas organizacionales en la microempresa Netline Colombia ISP. En este contexto, el enfoque y la relevancia del estudio se basan en la utilidad práctica de la solución (Creswell y Creswell, 2018) para eliminar las principales ineficiencias que pueden ocurrir por la gestión manual a través de la automatización y centralización. En línea con el objetivo general establecido, el estudio es una investigación aplicada con diseño de desarrollo tecnológico y proyectivo. Hurtado de Barrera (2012) dice que este tipo de investigación se basa en utilizar el conocimiento de la comunidad científica y técnica para desarrollar una solución concreta para una necesidad ambiental. En este caso, el producto real es el sistema "AdminISP" (que funciona con módulos de facturación, cobro, control de inventario y comunicación bidireccional automatizada con la API de RouterOS (MikroTik)). Como vemos, la investigación no se detiene en la etapa de diagnóstico; es la arquitectura, construcción y validación de una tecnología operativa. Se utilizó el enfoque cualitativo para desarrollar el trabajo de campo y la evaluación del sistema con métodos cualitativos y cuantitativos (Hernández Sampieri et al., 2014).

La parte cualitativa se aplicó en la primera etapa de recopilación de datos con entrevistas al personal administrativo, técnico y gerencial. En la segunda etapa del análisis, se aplicó el análisis deductivo para traducir los problemas operativos en los discursos en especificaciones y requisitos de software. El aspecto cuantitativo se observó en la última etapa de evaluación de la calidad del software, que fue respaldada por la norma internacional ISO/IEC 25010 (ISO/IEC, 2011) para evaluar la operabilidad y la participación del usuario en el software. Al aplicar el cuestionario estandarizado System Usability Scale (SUS) propuesto por Brooke (1996), se

recopilaron datos que, tras la tabulación algorítmica, proporcionaron índices cuantitativos sobre la curva de aprendizaje y la usabilidad de la interfaz, confirmando así el éxito del proyecto en términos empíricos y estadísticos.

Palabras clave:

Proveedor de servicios de internet (ISP), automatización de procesos, sistemas de información, monitoreo de red, usabilidad de software.

1 Introducción, marco teórico o de referencia

Actualmente, Internet es un servicio clave para el desarrollo socioeconómico, educativo y tecnológico de las comunidades. Los Proveedores de Servicios de Internet (ISP) son la columna vertebral del ecosistema global, conectando a los usuarios finales a Internet (Kurose y Ross, 2017). Pero a medida que los grandes actores construyen su infraestructura en las grandes ciudades, los ISP regionales y rurales están en condiciones de cerrar la brecha digital en las áreas de difícil acceso. A pesar de su relevancia, las microempresas en esta área a menudo enfrentan un gran problema de brecha de madurez digital y tienen que recurrir a procesos manuales debido a los altos costos económicos y técnicos. Por esta razón, el modelo organización-tecnología-entorno (TOE), desarrollado por Tornatzky y Fleischer (1990), se basa en el hecho de que son muy importantes, en los recursos de la organización de servicios/testigos, la tecnología disponible, el entorno y las presiones o demandas del entorno, que son críticas para la adopción e implementación de innovaciones técnicas en una empresa. En cuanto a los ISP rurales actuales, las soluciones comerciales en el mercado internacional como SmartISP, WisPRO o WispHub son completamente inviables debido al alto costo de licencias y la complejidad técnica de su

infraestructura, por lo tanto, es difícil hacer viable el negocio para una microempresa (Tornatzky y Fleischer, 1990). Este problema internacional es aún peor en Colombia debido a la infraestructura desconectada en el país. La implementación de Tecnologías de la Información y la Comunicación (TIC) en Colombia ha demostrado tener aún un retraso en la sistematización de las operaciones centrales de las micro, pequeñas y medianas empresas (MIPYMES), según Gálvez Albarracín et al. (2014). La ausencia de herramientas centralizadas en las organizaciones colombianas también es una limitación importante porque la gestión no puede lograr una visión clara y efectiva del negocio en tiempo real. En el contexto de los Sistemas de Información, el modelo de éxito de DeLone y McLean (2003) establece que la calidad de la información y la calidad del sistema son los factores que impactan la satisfacción del usuario y, por lo tanto, el uso neto de la plataforma. La dependencia de bases de datos empíricas en hojas de cálculo y facturación básica lleva a ineficiencias que obstaculizan la calidad del servicio que el ISP tiene en marcha; si no tienen una plataforma centralizada o un portal de autogestión, los clientes pueden recurrir a canales informales (llamadas o comunicación instantánea) para reportar problemas técnicos u obtener estados de cuenta. Esto lleva a tiempos de retraso, exceso de personal y la erosión de la confianza del consumidor (DeLone y McLean, 2003). En la infraestructura técnica, la falta de políticas de seguridad bien establecidas hace que las configuraciones de red a nivel de servicio de la red estén desequilibradas. Sandhu et al. (1996) argumentan que el Control de Acceso Basado en Roles (RBAC) en la administración de redes y sistemas debe implementarse, de modo que cada desarrollador de software (técnico, administrador o cobrador) tenga el menor privilegio para asegurar que no cambie el software y, por lo tanto, la red siempre esté estable. Según el Gobierno de Nariño (2020), fortalecer la conectividad en los municipios es una parte esencial del desarrollo regional, pero las empresas locales a menudo no tienen recursos. Netline Colombia ISP

es una microempresa ubicada en el municipio de Yacuanquer. Con una base de suscriptores en crecimiento y una infraestructura de red física (nodos, antenas, enrutadores), la empresa ahora está experimentando algunos problemas reales con la gestión empírica y manual de sus actividades. Los procesos en papel de registro de usuarios, facturación masiva, cobro y control de inventario han creado un cuello de botella administrativo que genera ineficiencias, errores de facturación y omisiones en las tarifas y no puede siquiera ver la salud financiera y técnica del negocio. La gestión de fallas es reactiva desde el lado técnico y no se lleva a cabo en presencia de un sistema de monitoreo proactivo, lo que resulta en que la calidad del servicio no sea buena y los suscriptores no confíen en él. Las soluciones comerciales como SmartISP o WisPRO simplemente no son económicamente viables o están sobrevaloradas para esta microempresa. Basado en el modelo TOE y la teoría del éxito de los sistemas de información, Netline Colombia ISP no se trata de comprar software comercial genérico, sino de desarrollar una tecnología diseñada para la empresa con requisitos específicos de la organización. El desafío surge de la siguiente pregunta:

¿Cómo ayudará el desarrollo e implementación de un gran sistema de software personalizado en la automatización y optimización de las operaciones, la gestión financiera y la gestión de la red de NetLine Colombia ISP?

2 Planteamiento del problema

Basado en el diagnóstico, la declaración del problema se resume en las principales ineficiencias y vulnerabilidades sufridas por Netline Colombia ISP como resultado de una gestión principalmente empírica basada en procesos manuales, hojas de cálculo aisladas y registros en papel. Esto resulta en una mala gestión organizativa de los suscriptores, errores y fallos en la comunicación en los procesos de facturación y cobro masivo, e incertidumbre logística en el

almacenamiento de productos de telecomunicaciones. A nivel técnico, la ausencia de un sistema de monitoreo automatizado y proactivo lleva a la organización a un esquema de soporte reactivo en caso de fallos en los nodos de la red, así como la falta de políticas de Control de Acceso Basado en Roles (RBAC) para los routers MikroTik, dejando la infraestructura vulnerable a fallos de configuración o modificaciones no autorizadas. Con el mercado actual ofreciendo costos muy bajos y desconectados de la microempresa, la solución debe ser una solución tecnológica que integre todo, automatice los flujos de trabajo y emplee controles de seguridad para brindar estabilidad en el servicio y escalabilidad del proveedor a escala local.

3 Objetivos

3.1 Objetivo General

Diseñar e implementar un software de gestión centralizado que permita automatizar y optimizar los procesos de gestión de clientes, facturación, recaudo, inventario, finanzas y monitoreo en tiempo real de la red y dispositivos MikroTik para la empresa NetLine Colombia ISP en el municipio de Yacuanquer, Nariño, durante el año 2026.

3.2 Objetivos Específicos

- Analizar los procesos administrativos y operativos de NetLine Colombia ISP con el fin de identificar y documentar los requerimientos funcionales y no funcionales del sistema.
- Diseñar la arquitectura del software, definiendo la estructura, tecnologías, base de datos e integración con dispositivos MikroTik, acorde con las necesidades de la empresa.
- Desarrollar e implementar los módulos del sistema en una plataforma centralizada para la

gestión de clientes, facturación, recaudo, inventario y monitoreo en tiempo real de la red, incluyendo una página web de autogestión para la consulta y pago en línea de facturas.

- Evaluar la calidad del software mediante la norma ISO/IEC 25010, enfocándose en la característica de usabilidad (Capacidad de Interacción), específicamente en los atributos de operabilidad e involucración del usuario.

4 Metodología

4.1 Metodología de la investigación

Esta investigación se enmarca en un enfoque pragmático ya que su objetivo principal no es solo proporcionar nuevo conocimiento teórico, sino abordar la gestión práctica y efectiva de los problemas organizacionales en la microempresa Netline Colombia ISP. En este contexto, el enfoque y la relevancia del estudio se basan en la utilidad práctica de la solución (Creswell y Creswell, 2018) para eliminar las principales ineficiencias que pueden ocurrir por la gestión manual a través de la automatización y centralización. En línea con el objetivo general establecido, el estudio es una investigación aplicada con diseño de desarrollo tecnológico y proyectivo. Hurtado de Barrera (2012) dice que este tipo de investigación se basa en utilizar el conocimiento de la comunidad científica y técnica para desarrollar una solución concreta para una necesidad ambiental. En este caso, el producto real es el sistema "AdminISP" (que funciona con módulos de facturación, cobro, control de inventario y comunicación bidireccional automatizada con la API de RouterOS (MikroTik)). Como vemos, la investigación no se detiene en la etapa de diagnóstico; es la arquitectura, construcción y validación de una tecnología operativa. Se utilizó el enfoque cualitativo para desarrollar el trabajo de campo y la evaluación del sistema con métodos cualitativos y cuantitativos (Hernández Sampieri et al., 2014). La parte cualitativa se aplicó en la primera etapa de recopilación de datos con entrevistas al personal administrativo, técnico y

gerencial. En la segunda etapa del análisis, se aplicó el análisis deductivo para traducir los problemas operativos en los discursos en especificaciones y requisitos de software. El aspecto cuantitativo se observó en la última etapa de evaluación de la calidad del software, que fue respaldada por la norma internacional ISO/IEC 25010 (ISO/IEC, 2011) para evaluar la operabilidad y la participación del usuario en el software. Al aplicar el cuestionario estandarizado System Usability Scale (SUS) propuesto por Brooke (1996), se recopilaron datos que, tras la tabulación algorítmica, proporcionaron índices cuantitativos sobre la curva de aprendizaje y la usabilidad de la interfaz, confirmando así el éxito del proyecto en términos empíricos y estadísticos.

4.2 Metodología de Desarrollo

El enfoque de la metodología ágil se empleará para desarrollar el software “AdminISP” con prácticas de Programación Extrema (XP) para convertir el gran objetivo del proyecto en una serie de ciclos de desarrollo cortos y adaptables. En cada período, la dirección estará en construir y entregar una pieza de software totalmente funcional, como el módulo de cliente o la facturación. Las prácticas de XP, especialmente el Desarrollo Guiado por Pruebas, servirán como un riguroso proceso de control de calidad personal para certificar que cada nueva funcionalidad sea fuerte y no cause traspiés en el sistema, accediendo así un progreso incremental, flexible y razonable hacia el producto final.

4.2.1 Fases

Fase 1: Planificación y preparación (fundamentos del proyecto).

Objetivo: Establecer las bases técnicas y de diseño antes de escribir la primera línea de código funcional.

Tecnología confirmada.

Lenguaje de backend: PHP, Base de datos: MySQL, Frontend: HTML, CSS, Python y JavaScript. Configuración del entorno de desarrollo. Así que debes configurar un entorno de servidor local (XAMPP o WAMP SERVER) con Apache, PHP, MySQL y carpetas de proyecto (css, js, img). Diseña el diagrama de entidad-relación. Configura las tablas principales en MySQL: usuarios, planes, facturas, inventario, gastos. Define las relaciones entre las tablas y las claves primarias y foráneas

.Fase 2: Creación de módulos iniciales (Gestión de usuarios, Clientes y Planes).

Objetivo: Proporcionar funcionalidad básica para gestionar los usuarios de gestión que serán los administradores del software y los clientes del ISP con los planes ofrecidos.

Módulo de gestión y usuarios clientes. Crear el formulario de inicio de sesión y la lógica con PHP y sesiones. Crear el CRUD (Crear, Leer, Actualizar, Eliminar) para usuarios administrativos (técnicos, administradores). Implementar un sistema básico de roles y permisos para restringir el acceso a ciertos módulos.

Desarrollar el formulario para registrar y editar clientes. Crear la vista para mostrar la lista de clientes con búsqueda y filtros. Implementar la funcionalidad para asignar un plan a un cliente. Módulo de Planes: Crear la interfaz (HTML/CSS/JAVASCRIPT) y la lógica (PHP/MySQL) para

agregar, editar y eliminar planes de internet.

Fase 3: Construcción de los módulos financieros (Facturación, Recaudo, Ingresos/Gastos).

Objetivos: Automatizar el ciclo de facturación, cobranza y mantener las finanzas de la empresa bajo control.

Módulo de Facturación y Recaudo:

Crear la lógica para generar manualmente una factura. Y crear la lógica para generar toda la facturación con un solo botón. Crear la vista para facturas pendientes y visualización a través de una ventana emergente de facturas marcadas como pagadas/canceladas y sus respectivas funcionalidades (búsqueda por ID o nombre, filtrado por rango de fechas y el botón para exportar datos a una hoja de cálculo). Desarrollar una ventana emergente que se abra en la pantalla con el botón que puede cambiar el estado de una factura de Pendiente a Pagada/Cancelada y mantener el historial de facturas. Módulo de Ingresos y Gastos. Desarrollar la lógica para ver el número de facturas pagadas y la suma de los montos que serían los ingresos y una sección para los gastos concurrentes y diarios que se registran diariamente. Donde se realiza la operación respectiva (ingresos - gastos = Ganancia/Pérdida).

Fase 4: Desarrollo del módulo de inventario.

Diseño del módulo de inventario; agregando diferentes tipos de activos y agregando o eliminando unidades, con una sección donde se enumeran todos los tipos de activos con cantidades a asignar y finalmente una sección donde se enumeran todos los activos asignados a clientes o para uso de la empresa.

Fase 5: Producción de módulos de gestión de colas y monitoreo de red.

Objetivo: Crear las herramientas técnicas clave para la gestión de redes y dispositivos.

Módulo de gestión de colas (Integración con Mikrotik): Escribimos una biblioteca en Python que se conecta con la API de RouterOS (Mikrotik), crear una interfaz de inicio de sesión para completar los datos del dispositivo Mikrotik. La interfaz para el personal técnico para crear, actualizar y eliminar las colas simples de un cliente desde el software. Crear una interfaz para que el personal técnico vea las rutas que cada cliente está tomando para acceder a internet. Módulo de monitoreo de red. Crear un script en PHP para verificar el estado (en línea/fuera de línea) de los dispositivos registrados en la red usando ping. Crear un mapa en el frontend (usando bibliotecas de JavaScript) que pueda proporcionar visualización en tiempo real del estado de los dispositivos Subida (Verde) y Descarga (Rojo). Diseñar un panel para gestionar los dispositivos registrados y sus respectivas conexiones y un botón para iniciar/detener la sincronización de dispositivos.

Fase 6: Construcción del portal del cliente.

Crear una interfaz web donde el usuario pueda interactuar directamente con la empresa. Crear una página web donde el usuario pueda acceder a toda la información de la empresa y los servicios ofrecidos: Crear un área separada para clientes. Desarrollar la vista donde el cliente pueda ver el valor y el estado de su factura. Integrar una pasarela de pago en línea.

Fase 7: Pruebas, evaluación y despliegue

Objetivo: Asegurar la calidad del software y ponerlo en producción.

Pruebas y depuración. Realizar pruebas funcionales de todos los módulos, Corregir todos los errores (bugs) encontrados.

Evaluación de usabilidad (ISO 25010). Realizar sesiones de prueba con el personal de Netline para evaluar la Operabilidad y la Involucración del Usuario, para medir la Inclusión del Usuario.

Despliegue en producción, contratar y configurar un servidor web compatible con PHP y MySQL. Subir los archivos del proyecto al servidor y realizar pruebas finales en el entorno de producción.

5 Resultados y Discusión

5.1 *Procesos Administrativos y Operativos de Netline Colombia ISP*

Esta investigación se enmarca en un enfoque pragmático ya que su objetivo principal no es solo proporcionar nuevo conocimiento teórico, sino abordar la gestión práctica y efectiva de los problemas organizacionales en la microempresa Netline Colombia ISP. En este contexto, el enfoque y la relevancia del estudio se basan en la utilidad práctica de la solución (Creswell y Creswell, 2018) para eliminar las principales ineficiencias que pueden ocurrir por la gestión manual a través de la automatización y centralización. En línea con el objetivo general establecido, el estudio es una investigación aplicada con diseño de desarrollo tecnológico y proyectivo. Hurtado de Barrera (2012) dice que este tipo de investigación se basa en utilizar el conocimiento de la comunidad científica y técnica para desarrollar una solución concreta para una necesidad ambiental. En este caso, el producto real es el sistema "AdminISP" (que funciona con módulos de facturación, cobro, control de inventario y comunicación bidireccional automatizada con la API de RouterOS (MikroTik)). Como vemos, la investigación no se detiene en la etapa de diagnóstico; es la arquitectura, construcción y validación de una tecnología operativa. Se utilizó el enfoque cualitativo para desarrollar el trabajo de campo y la evaluación del sistema con métodos cualitativos y cuantitativos (Hernández Sampieri et al., 2014). La parte cualitativa se aplicó en la primera etapa de recopilación de datos con entrevistas al personal administrativo, técnico y gerencial. En la segunda etapa del análisis, se aplicó el análisis deductivo para traducir los

problemas operativos en los discursos en especificaciones y requisitos de software. El aspecto cuantitativo se observó en la última etapa de evaluación de la calidad del software, que fue respaldada por la norma internacional ISO/IEC 25010 (ISO/IEC, 2011) para evaluar la operabilidad y la participación del usuario en el software. Al aplicar el cuestionario estandarizado System Usability Scale (SUS) propuesto por Brooke (1996), se recopilaron datos que, tras la tabulación algorítmica, proporcionaron índices cuantitativos sobre la curva de aprendizaje y la usabilidad de la interfaz, confirmando así el éxito del proyecto en términos empíricos y estadísticos.

5.2 Requerimientos Funcionales y no Funcionales del Sistema.

5.2.1 Requerimientos funcionales (RF)

RF-01 Gestión de usuarios y clientes. El sistema debe permitir el registro, actualización y consulta centralizada de datos de clientes y usuarios de gestión (CRUD)

RF-02 Gestión de planes y tarifas. El sistema debe permitir crear planes de servicio con tarifas predefinidas para evitar errores de cobro manual.

RF-03 Facturación masiva y automática. El sistema debe generar facturas automáticamente de forma mensual para todos los clientes activos.

RF-04 Módulo de recaudo. El sistema debe permitir registrar pagos manuales y conciliar pagos en línea de manera intuitiva.

RF-05 Dashboard financiero. El sistema debe generar reportes en tiempo real sobre ingresos, cartera vencida y flujo de caja.

RF-06 Gestión de Inventario. El sistema debe controlar el stock de equipos (antenas, routers, cables), registrando entradas, salidas y asignación a clientes.

RF-07 Alertas de Stock Mínimo. El sistema debe notificar cuando el inventario de un equipo crítico esté por debajo del umbral definido.

RF-08 Integración con API Mikrotik. El sistema debe comunicarse con los equipos Mikrotik para aplicar cortes de servicio (por no pago), reconexiones y gestión de ancho de banda.

RF-09 Monitoreo de Nodos. El sistema debe visualizar el estado (online/offline) de los nodos principales y alertar sobre tiempos de indisponibilidad.

RF-10 Portal de Autogestión. El sistema debe contar con una interfaz web donde el cliente pueda consultar su saldo y realizar pagos (integración con pasarela de pagos).

RF-11 Gestión de Solicitudes (PQR). El sistema debe permitir a los clientes y empleados crear tickets de soporte técnico o nuevas instalaciones.

5.3 Requerimientos no funcionales (RNF)

Definen cómo debe ser el sistema (calidad, rendimiento, seguridad). RNF-01 Seguridad y Control de Acceso. El sistema debe manejar roles y permisos (Administrador, Técnico, Cajero) para restringir quién puede modificar configuraciones en los equipos y en el sistema.

RNF-02 Disponibilidad. El sistema debe estar disponible 24/7 (99.9% de optime), especialmente el portal de clientes y el monitoreo.

RNF-03 Usabilidad. La interfaz debe ser intuitiva y "responsive" (adaptable a móviles), facilitando su uso por personal no técnico.

RNF-04 Interoperabilidad. El sistema debe ser capaz de intercambiar datos correctamente con el protocolo de la API de RouterOS (Mikrotik).

RNF-05 Rendimiento. El módulo de monitoreo debe procesar el estado de los dispositivos

en tiempo casi real (latencia menor a 1 minuto para alertas).

RNF-06 Escalabilidad. La base de datos y la arquitectura deben soportar el crecimiento del número de abonados sin degradar la velocidad del sistema.

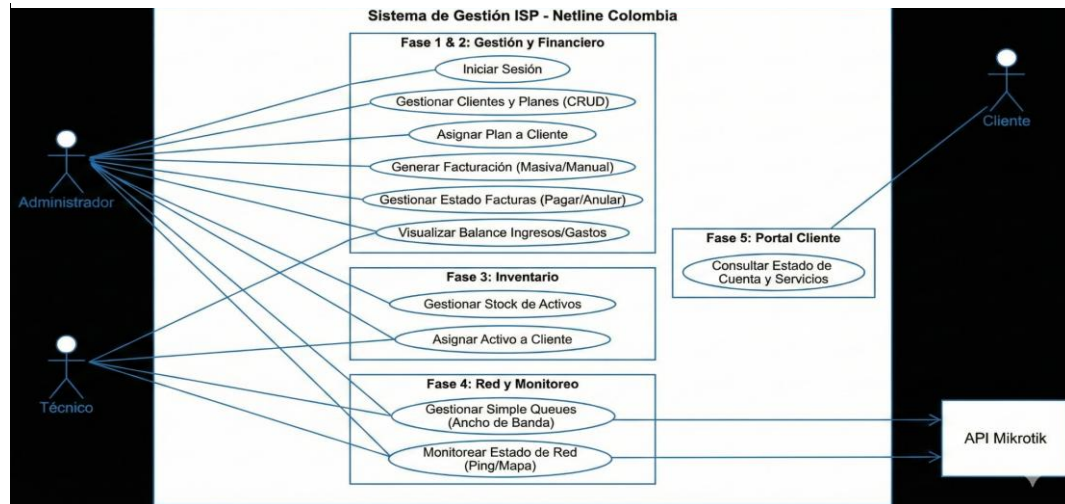
5.4 Diseño de la Arquitectura del Software: Estructura, Tecnologías, Base de Datos e Integración con Dispositivos Mikrotik

En esta etapa de aseguramiento de calidad, ahora pasamos a la validación de alto nivel para garantizar que el sistema cumpla con todos los requisitos funcionales de AdminISP. Realizamos una auditoría del diagrama de casos de uso general para asegurar que las interacciones entre los diferentes actores y módulos del sistema coincidieran con los procesos operativos presentados para Netline Colombia (Figura. 1).

5.4.1 Diagrama caso de uso

Figura 1

Diagrama caso de uso - General



Nota. El diagrama detalla la arquitectura funcional del sistema dividida en fases operativas. Se valida la interacción de los roles Administrador, Técnico y Cliente con los módulos de facturación, inventario y gestión de red, destacando la integración con la API Mikrotik para el

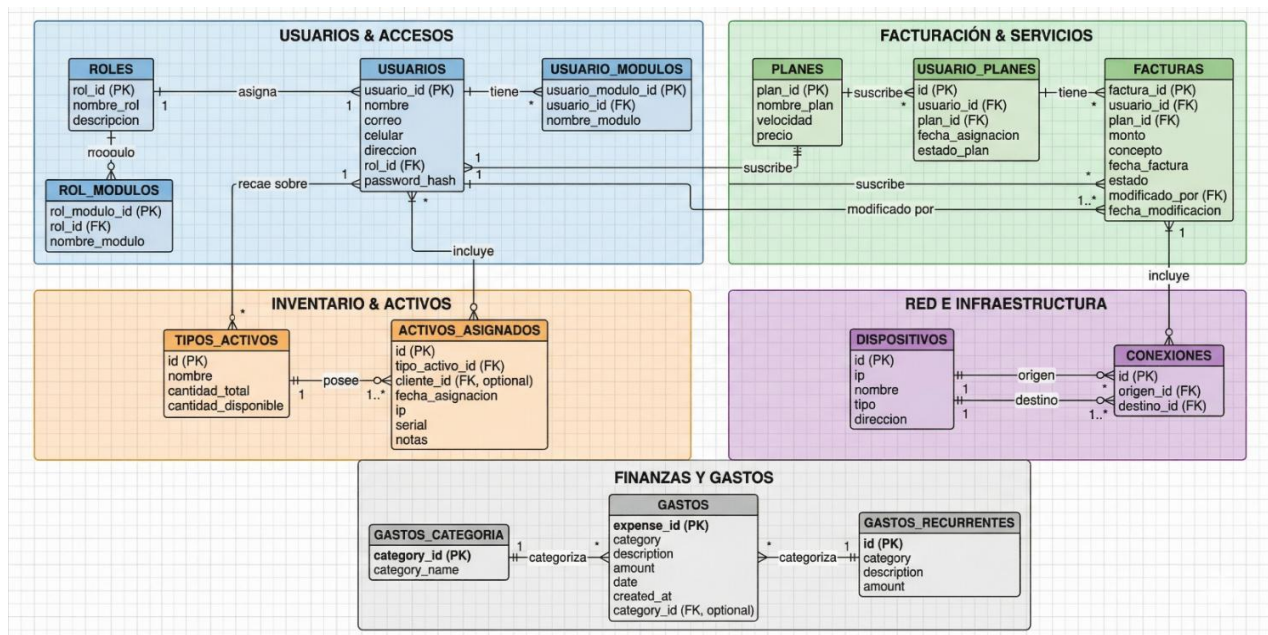
control de ancho de banda y monitoreo en tiempo real.

5.5 Diagramas de clase, bases de datos

Modelo de Datos y Arquitectura de Persistencia, el sistema para Netline Colombia se basa en un modelo Entidad-Relación segmentado en cinco núcleos: Usuarios, Facturación, Inventario, Red y Finanzas. Esta estructura garantiza la integridad referencial y la trazabilidad de operaciones críticas, como la vinculación de planes de navegación y la asignación técnica de activos. La normalización aplicada facilita una gestión auditable y escalable, permitiendo al sistema responder eficientemente al crecimiento de suscriptores e infraestructura en la región, como se observa en la figura 2

Figura 2

Diagrama Entidad Relación



Nota. El diagrama muestra una estructura robusta y normalizada. Se validaron puntos clave como la centralización de la tabla usuarios y sus relaciones con planes y roles, la gestión de inventario a través de activos_asignados, y el control financiero mediante las tablas de facturas y gastos. Esta

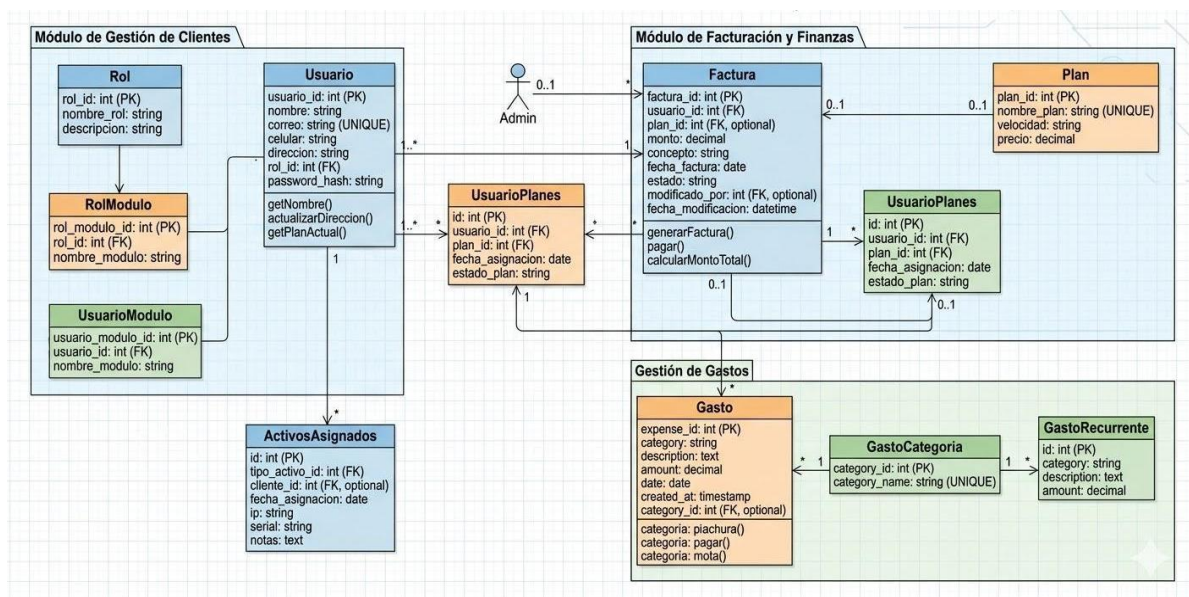
estructura garantiza la integridad referencial y la escalabilidad del sistema ante el crecimiento de datos.

5.5.1 Modelo de clases y lógica de negocio

El sistema se estructura mediante un diagrama de clases que define la interacción entre los módulos de gestión de clientes, facturación y control de gastos. Esta arquitectura lógica permite la administración de usuarios, la suscripción a planes y el procesamiento financiero mediante métodos operativos específicos como se observa en la figura 3

Figura 3

Diagrama de clase (clientes – facturación)



Nota. El diagrama representa la estructura estática del software, detallando los atributos y métodos que permiten la interoperabilidad entre la gestión de usuarios, la facturación de servicios y el control de egresos del sistema.

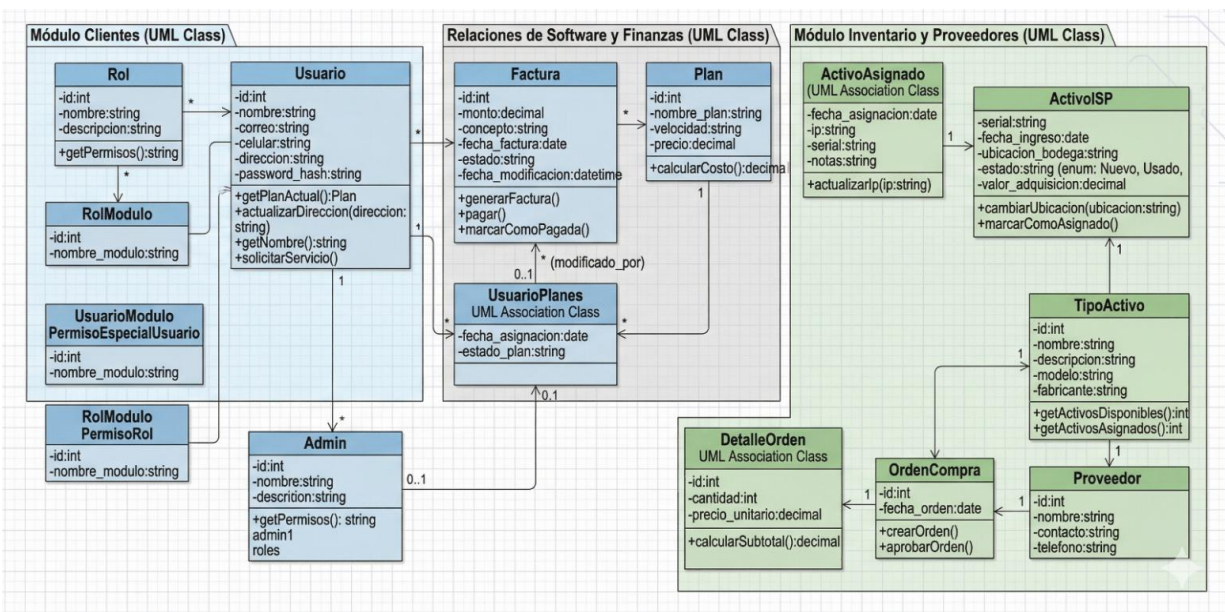
5.5.2 Arquitectura de gestión de recursos e inventario

El sistema integra la administración de clientes con el control de inventarios y la relación con proveedores. Esta estructura UML permite coordinar la asignación de activos tecnológicos

con el flujo financiero de la empresa, asegurando que cada servicio contratado cuente con el respaldo físico necesario. La cohesión entre estos módulos facilita una gestión operativa centralizada, optimizando el seguimiento de activos desde su adquisición hasta su entrega final al usuario, como se observa en la figura 4.

Figura 4

Diagrama UML de clases (clientes e inventario)



Nota. El diagrama describe la interoperabilidad entre la gestión de usuarios, el control de activos de AdminISP y el proceso de órdenes de compra con proveedores, garantizando la trazabilidad técnica y económica del sistema.

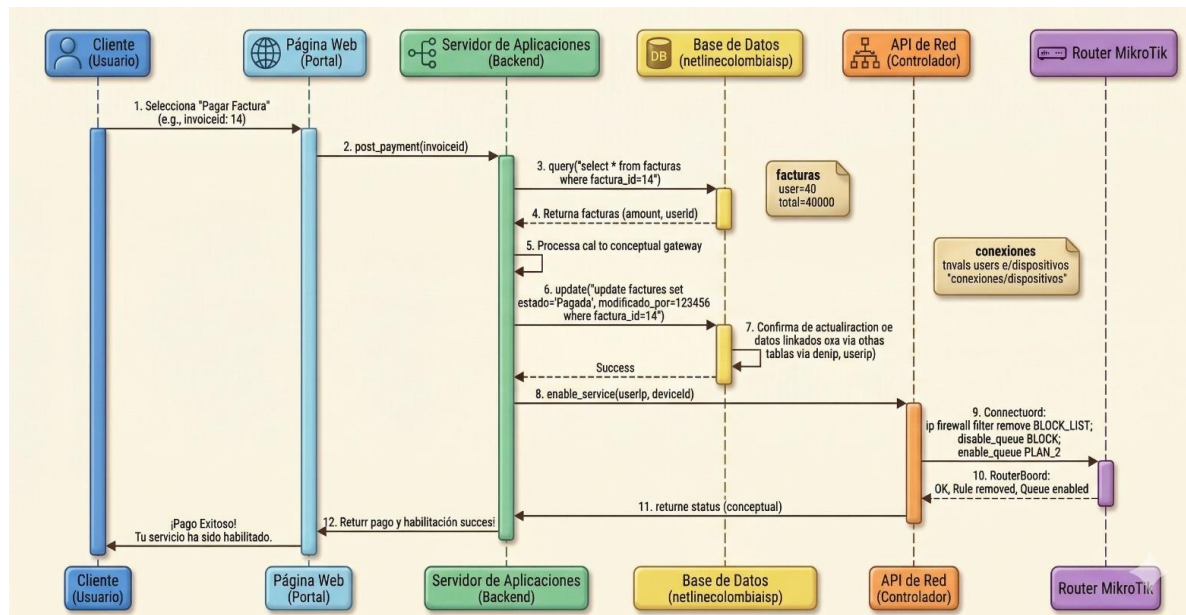
5.5.3 Automatización del flujo de pago y provisión del servicio

El sistema integra un flujo de procesos dinámicos que automatiza la habilitación del servicio de internet tras la validación de un pago. Este proceso coordina la interacción entre la interfaz de usuario, el servidor de aplicaciones, la base de datos y la infraestructura de red a través de una API de control. La arquitectura permite que, una vez confirmada la transacción financiera,

se envíen comandos automáticos al enrutador MikroTik para modificar las reglas de firewall y las colas de tráfico (queues), garantizando una respuesta inmediata al cliente sin intervención manual del administrador como se observa en la figura 5.

Figura 5

Diagrama de secuencia (habilitación del servicio)



Nota. El diagrama describe la secuencia de eventos entre los componentes del sistema para la reconexión del servicio. Destaca la comunicación bidireccional entre el backend y el controlador de red para la actualización de estados de pago y la ejecución de comandos técnicos en el hardware MikroTik.

5.5.4 Modelo de desarrollo ágil e incremental

Para la construcción de la plataforma de gestión de NetLine Colombia ISP, se empleó un enfoque de desarrollo ágil e incremental. Esta metodología permitió adaptar el proyecto a los

requerimientos del cliente y entregar valor de forma continua, a diferencia de los modelos tradicionales que entregan el producto completo al final. La implementación se llevó a cabo en las siguientes fases interrelacionadas:

Con los requisitos definidos, se procedió a diseñar la arquitectura de la solución. En esta fase se establecieron los planos del sistema antes de escribir la primera línea de código. Se crearon los siguientes artefactos:

5.5.5 Arquitectura Lógica.

El sistema se describe como cliente-servidor y la interfaz web se ejecuta en el navegador (cliente) mientras que la lógica de negocio, el procesamiento de datos y la interacción con la base de datos se realizan en el servidor.

5.5.6 Diseño de Base de Datos.

La estructura de la base de datos fue modelada mediante un Diagrama de Entidad-Relación (ERD). El diagrama proporcionó las tablas (entidades) utilizadas para mantener clientes, facturas, planes, inventario y las relaciones entre ellos.

5.5.7 Gestión de Base de Datos.

La gestión se ejecuta utilizando el sistema de gestión de bases de datos MySQL (versión 9.1.0) y el motor InnoDB para transacciones y restricciones de integridad referencial. Los datos están organizados en tablas normalizadas para mantener los datos del ISP

5.5.8 Consistentes Diseño y Estructura de la Base de Datos.

La siguiente tabla proporciona descripciones técnicas de cada tabla según el esquema:

usuarios: Contiene información personal del usuario como detalles de contacto (nombre, correo electrónico, número de teléfono, dirección) y el hash de la contraseña del usuario para

acceder al sistema.

roles: Describe los privilegios de acceso ('Cliente' y 'Usuario de Gestión'), que están conectados a la tabla de usuarios a través de `role_id`.

planes: Especifica la oferta comercial del ISP por nombre del plan, velocidad (por ejemplo, 15M/15M) y precio mensual.

Planes_de_usuarios: Maneja las conexiones activas entre clientes y servicios contratados y anota la fecha de asignación y el estado de un plan.

Facturas: Registra las colecciones de dinero recaudado y conecta cada cargo a un usuario y plan, y rastrea su estado ('Pendiente' o 'Pagado') y quién realizó la última modificación.

Gastos: Rastrea los gastos operativos incurridos, registrando el monto, fecha y descripción, y opcionalmente puede estar vinculado a una categoría (por ejemplo, `category_id`).

Categoría_de_gastos: Es el catálogo maestro de gastos de la empresa (ALQUILER, ENERGÍA, INTERNET) que representa las principales categorías de gastos.

Gastos_recurrentes: Representa precios fijos mensuales (alquiler y pagos al proveedor de fibra) para que los cargos de mantenimiento o regulares puedan ser automatizados o estimados.

Tipos_de_activos: Una lista de hardware disponible (por ejemplo, LiteBeam M5-23, Routers Mikrotik), la cantidad total y la cantidad disponible para ser almacenada.

Activos_asignados: Rastrea el envío del equipo a ciertos clientes o su uso en nodos, donde se almacenan direcciones IP, números de serie y notas técnicas.

Dispositivos: Inventario técnico de la infraestructura de red, incluyendo clase de equipo

(repetidor, router, switch, etc.) y asignación de sus direcciones IP únicas.

5.6 Arquitectura Cliente-Servidor

La arquitectura del sistema se basa en un modelo cliente-servidor, donde las tareas se comparten entre dos componentes principales que están interconectados a través de una red. Permite que la lógica de negocio y el manejo de datos se unifiquen, y la interfaz de usuario es accesible a través de un navegador.

5.6.1 El cliente.

Es la interfaz a la que acceden los usuarios finales y el personal administrativo desde un navegador web. Su objetivo es enviar solicitudes al servidor (como cargar una página, iniciar sesión o crear un cliente) y producir la respuesta que recibe en forma de una solicitud informativa.

5.6.2 El servidor.

Es entonces el núcleo del sistema. Funciona como un servidor, alojado en un ordenador remoto, para procesar las llamadas del lado del cliente y procesar la funcionalidad de la aplicación (por ejemplo, automatización de facturación) e interactuar con bases de datos y sistemas internos. Un aspecto clave de esta arquitectura es la comunicación directa entre los dispositivos de red internos (enrutadores Mikrotik) y el propio servidor. El módulo de colas y el módulo de monitoreo, por otro lado, están diseñados para operar exclusivamente en la red local de la empresa y, por lo tanto, proporcionan una fuerte seguridad y evitan exponer los dispositivos internos al tráfico de Internet.

5.7 Tecnologías orientadas al servidor (Backend)

El módulo del servidor o Backend es responsable de la lógica de negocio, la gestión de

datos y la comunicación con otros sistemas. En un desarrollo del sistema ISP de NetLine Colombia se emplean las siguientes tecnologías:

5.7.1 Lenguaje de Programación (PHP).

Fue el lenguaje de programación del backend. Se seleccionó por su madurez, amplia cooperación comunitaria y también por el fácil acceso a bases de datos y la API de Mikrotik. PHP funciona como el motor para todos los trabajos de la aplicación, desde la autenticación de usuarios hasta la generación de facturas.

5.7.2 Servidor Web (Apache o Nginx)

Estos servidores web son responsables de alojar la aplicación y de aceptar solicitudes de clientes. Son responsables de que las solicitudes HTTP se transmitan desde el navegador al intérprete de PHP para ejecutar el código de la aplicación. Se han seleccionado en función de su capacidad, velocidad, rendimiento, fiabilidad del sistema y la facilidad de uso de la aplicación con el entorno de desarrollo.

5.7.3 API de Mikrotik.

Se implementó una integración directa con la API de los enrutadores Mikrotik. El software permitió que el servidor ejecutara funciones de forma remota, incluyendo la creación y modificación de colas, sin interactuar directamente con comandos manuales.

5.8 Tecnologías del lado del cliente (Frontend).

La capa del cliente, o frontend, es la capa donde el administrador y el cliente final tienen una interacción directa. El propósito principal de esto es presentar la información de manera clara y facilitar la entrada de datos. Esta capa se construyó utilizando el siguiente conjunto de tecnologías web estándar:

5.8.1 HTML5.

Se utilizó como el lenguaje de marcado esencial y se usó para definir los contenidos de todas las páginas de la aplicación. HTML5 es el lenguaje de marcado que sirve como columna vertebral para proporcionar una estructura arquitectónica para elementos: Formularios, botones, tablas, texto, etc. Estos son todos los lugares donde se establecerá una base semántica segura y moderna para la aplicación.

5.8.2 CSS3.

Fue el estilo estético de diseño que proporcionó la plataforma. Aunque la interfaz fue altamente personalizada usando CSS3, fue fácil de usar. Luego se implementó para gestionar el diseño, colores, fuentes y espaciado, de modo que los usuarios vieran un diseño limpio y profesional.

5.8.3 JAVASCRIPT.

Este lenguaje de programación facilitó la interacción y la adición de dinamismo a la interfaz de usuario. Para la validación de formularios en el navegador, animaciones, manipulación de la interfaz y solicitudes asíncronas al servidor (AJAX), JavaScript asume la responsabilidad en el diseño de la página. Esto también ayuda a que las diversas secciones se actualicen sin cargar toda la página.

5.9 Bibliotecas y APIs para la gestión de colas

Arquitectura Híbrida (PHP + Python): El sistema utiliza un entorno donde PHP (a través de `queues.php` y `actions.php`) gestiona la interfaz de usuario y la sesión, mientras que la comunicación directa con el hardware Mikrotik se delega a un script especializado en Python (`manager.py`). Esta separación permite utilizar los servicios web que tiene PHP y las bibliotecas

de red que se pueden usar con eficiencia si se utiliza Python.

5.9.1 API del fabricante (RouterOS API).

La comunicación con los dispositivos Mikrotik se realiza a través de su API nativa. El script de Python utiliza la librería `routeros_api`, la cual permite una conexión estructurada y segura con el RouterOS para realizar operaciones en tiempo real sobre las colas de tráfico (Simple Queues y Queue Tree).

5.10 Desarrollo e Implementación de los Módulos del Sistema AdminISP

Esta fue la etapa de construcción, donde se tradujeron los diseños en código funcional el cual se encuentra disponible en el repositorio de GitHub al cual puede acceder mediante el siguiente link: <https://github.com/einsuasty/AdminISP-Netline>

El desarrollo se realizó de manera incremental, entregando funcionalidades en ciclos. Los módulos clave se construyeron en el siguiente orden de prioridad.

5.11 Módulo de clientes y usuarios

El módulo de Gestión de Usuarios y Clientes representa una parte importante y de alto nivel del Panel de Administración del Sistema para centralizar el control sobre todos los usuarios que interactúan con la plataforma a nivel administrativo, operativo y comercial.

Desempeña un papel fundamental en la gestión del flujo de trabajo general de los usuarios internos del sistema y de los clientes finales, en el que se puede gestionar de manera ordenada la información, el acceso y los servicios relacionados con cada perfil. En primer lugar, Usuarios de Gestión: estos usuarios gestionarán el sistema de la plataforma administrándolo, operándolo y

monitoreándolo. En esta área, se debe registrar información importante, incluyendo ID de usuario, nombre, correo electrónico, detalles de contacto, dirección y rol asignado. Este rol es muy importante porque define el nivel de acceso entre cada usuario y el sistema y las acciones que cada usuario debe realizar en el sistema. Los usuarios también pueden editar/eliminar estos usuarios para que esta información se mantenga actualizada y aquellos que acceden al panel administrativo puedan saber quién puede acceder a la página a través del sistema.

Por otro lado, el módulo es para la gestión de los Clientes que son los usuarios finales del servicio. Se proporciona una lista que indica la identidad del cliente, nombre completo, correo electrónico, teléfono celular, dirección y el rol del cliente. Esta parte del módulo tiene un componente crucial de la asignación del plan principal que indica qué tipo de servicios se están contratando (planes residenciales que tienen diferentes velocidades).

De esta manera, el cliente puede comunicarse directamente con el servicio que utiliza, permitiendo la gestión comercial y técnica. También vienen con herramientas de búsqueda, exportación de datos y acciones específicas de usuario y cliente específicas para cada registro, como editar o eliminar usuarios y clientes.

El módulo incluye la facilidad de exportación en estructura de Excel, ayudan en la gestión rápida pero organizada de la información, asisten en el proceso de administración (Generación de informes, Control Interno, Auditorías, etc.), como se muestra en las figuras 6 y 7.

Figura 6

Módulo de clientes y usuarios – interfaz gráfica para registrar y editar

Panel de Administración Bienvenido(a), Eduard Insuasty (Usuario de Gestión) U

Gestión de Usuarios y Clientes Buscar por ID o Nombre **Buscar** **Limpiar** **+ Agregar Usuario**

Editar Usuario

Rol:

ID de Usuario:

Nombre:

Correo:

Número de Celular:

Dirección:

Planes Asignados

Plan Residencial 15Mbps (15M/15M - \$35.000) (Activo) Cambiar Estado Eliminar

Asignar Nuevo Plan: **Asignar**

Guardar Usuario **Cancelar**

Cerrar Sesión

Módulo de Usuarios

- Módulo de Planes
- Módulo de Facturación
- Módulo de Recaudo
- Módulo de Ingresos y Gastos
- Módulo de Inventario
- Módulo de Queues
- Módulo de Monitoreo

Nota. Se observan campos para datos personales, la asignación de planes de navegación y la matriz de permisos de módulos, garantizando una administración centralizada y segura de los accesos al sistema.

Figura 7

Fragmento de código - Registro de un usuario de gestión o cliente

```
switch ($action) {
    case 'create_user':
        $nombre = trim($data['nombre'] ?? '');
        $correo = filter_var(trim($data['correo'] ?? ''), FILTER_VALIDATE_EMAIL);
        $celular = trim($data['celular'] ?? null);
        $direccion = trim($data['direccion'] ?? null);
        $rol_id = filter_var($data['rol_id'] ?? null, FILTER_VALIDATE_INT);
        $new_password = $data['new_password'] ?? '';
        $allowed_modules = $data['allowed_modules'] ?? [];
        $plan_id = filter_var($data['plan_id'] ?? null, FILTER_VALIDATE_INT);

        $stmt_rol_name = $pdo->prepare("SELECT nombre_rol FROM roles WHERE rol_id = ?");
        $stmt_rol_name->execute([$rol_id]);
        $current_rol_name = $stmt_rol_name->fetchColumn();

        if (empty($nombre) || $correo === false || empty($rol_id) || $rol_id <= 0 || $current_rol_name === false) {
            http_response_code(400);
            echo json_encode(['error' => 'Nombre, correo válido y rol son requeridos.']);
            exit;
        }
        if ($current_rol_name !== 'Cliente') {
            if (empty($new_password) || strlen($new_password) < 6) {
                http_response_code(400);
                echo json_encode(['error' => 'Contraseña de al menos 6 caracteres es requerida para este rol.']);
                exit;
            }
        } else {
            $new_password = null;
            if (empty($plan_id)) {
                http_response_code(400);
            }
        }

        try {
            $pdo->beginTransaction();
            $usuario_id_to_insert = isset($data['usuario_id']) && $data['usuario_id'] != '' ? (int)$data['usuario_id'] : null;
            $hashed_password = ($new_password != null) ? password_hash($new_password, PASSWORD_DEFAULT) : null;

            if ($usuario_id_to_insert) {
                $stmt_check_id = $pdo->prepare("SELECT COUNT(*) FROM usuarios WHERE usuario_id = ?");
                $stmt_check_id->execute([$usuario_id_to_insert]);
                if ($stmt_check_id->fetchColumn() > 0) {
                    http_response_code(409);
                    echo json_encode(['error' => 'El ID de usuario especificado ya existe.']);
                    $pdo->rollback();
                    exit;
                }
                $stmt = $pdo->prepare("INSERT INTO usuarios (usuario_id, nombre, correo, celular, direccion, rol_id, password_hash) VALUES (?, ?, ?, ?, ?, ?, ?)");
                $stmt->execute([$usuario_id_to_insert, $nombre, $correo, $celular, $direccion, $rol_id, $hashed_password]);
            } else {
                $stmt = $pdo->prepare("INSERT INTO usuarios (nombre, correo, celular, direccion, rol_id, password_hash) VALUES (?, ?, ?, ?, ?, ?)");
                $stmt->execute([$nombre, $correo, $celular, $direccion, $rol_id, $hashed_password]);
                $usuario_id_to_insert = $pdo->lastInsertId();
            }

            if ($plan_id) {
                $stmt_assign = $pdo->prepare("INSERT INTO usuario_planes (usuario_id, plan_id, fecha_asignacion, estado_plan) VALUES (?, ?, NOW(), 'Activo')");
                $stmt_assign->execute([$usuario_id_to_insert, $plan_id]);
            }

            if ($current_rol_name === 'Usuario de Gestión') {
                updateUserPermissions($pdo, $usuario_id_to_insert, $allowed_modules);
            }

            $pdo->commit();
            http_response_code(201);
            echo json_encode(['message' => 'Usuario y plan creados exitosamente.']);
        } catch (PDOException $e) {
            $pdo->rollback();
            if ($e->getCode() == 23000) {
                http_response_code(409);
                echo json_encode(['error' => 'El correo o ID ya está registrado.']);
            } else {
                http_response_code(500);
                error_log("Error de BD al crear usuario: " . $e->getMessage());
                echo json_encode(['error' => 'Error al crear usuario.']);
            }
        }
        break;
    }
}
```

Nota. Proceso de creación de usuarios en el sistema. Incluye la validación de datos de entrada mediante filtros de seguridad, la verificación de roles mediante consultas preparadas a la base de datos y la implementación de transacciones PDO para asegurar la integridad de la información al insertar registros simultáneos en las tablas de usuarios y planes. Además, se observa el manejo de excepciones y el uso de funciones de hash para la protección de contraseñas.

5.12 Módulo de planes

Módulo de Gestión de Planes Parte del Panel de Administración, el Módulo de Gestión de Planes permite a los proveedores crear, consultar, modificar o eliminar planes de servicio ofrecidos por la plataforma. La importancia de este módulo surge de la gestión comercial clave del sistema, ya que determina las principales características de los servicios que luego se asignan a los clientes, como la velocidad y el valor del plan.

En la vista principal, una lista de planes registrados proporciona al usuario información esencial como el identificador del plan, el nombre comercial, la velocidad ofrecida y el valor correspondiente. Esta tabla ofrece al administrador una vista clara y organizada de los planes en la lista, permitiendo la toma de decisiones y el control sobre la oferta de servicios. Cada plan tiene opciones de acción para editar o eliminar información cuando ya no aplica. El módulo también proporciona una opción para agregar un nuevo plan, lo cual es posible a través de un formulario donde se ingresan los datos básicos del servicio (el nombre, la velocidad y el valor del plan).

El sistema permite guardar el plan después de que se haya completado la información detallada para que se incluya en la lista general o se cancele la operación si es necesario. Este método de registro mantiene un registro estructurado de cada plan. Finalmente, el módulo añade un botón de exportación en formato Excel para descargar los detalles del plan registrado con fines administrativos, análisis comercial o respaldo de datos (como se ve en las figuras 8 a 11).

Figura 8

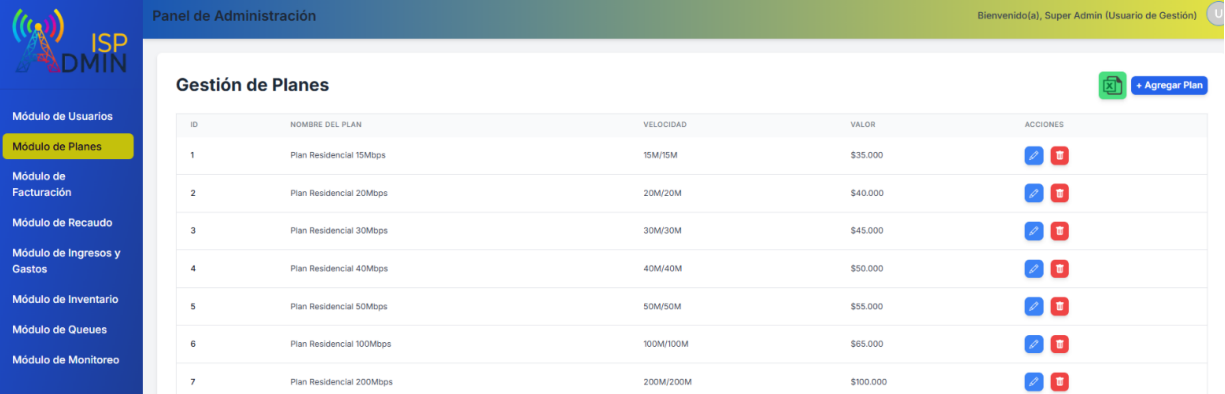
Fragmento de código - Lógica para el Registro y Validación de Planes de Servicio

```
switch ($action) {
    case 'create_plan':
        if (empty($data['nombre_plan']) || empty($data['velocidad']) || !isset($data['precio']) || !is_numeric($data['precio'])) { // Validación más estricta para precio
            http_response_code(400); // Bad Request
            echo json_encode(['error' => 'Nombre del plan, velocidad y un precio numérico son requeridos.']);
            exit;
        }
        try {
            $stmt = $pdo->prepare("INSERT INTO planes (nombre_plan, velocidad, precio) VALUES (?, ?, ?)");
            $stmt->execute([trim($data['nombre_plan']), trim($data['velocidad']), (float)$data['precio']]);
            http_response_code(201); // Created
            echo json_encode(['message' => 'Plan creado exitosamente.']);
        } catch (PDOException $e) {
            if ($e->getCode() == 23000) { // Código para violación de unicidad (nombre_plan UNIQUE)
                http_response_code(409); // Conflict
                echo json_encode(['error' => 'El nombre de plan ya existe.']);
            } else {
                http_response_code(500); // Internal Server Error
                error_log("Error de BD al crear plan en api_planes.php: " . $e->getMessage());
                echo json_encode(['error' => 'Error al crear plan', 'detalle' => (ini_get('display_errors') == 1 ? $e->getMessage() : 'Error interno del servidor.')]);
            }
        }
        break;
}
```

Nota. Implementación para la creación de planes. Se destaca la validación estricta de campos obligatorios (nombre, velocidad y precio numérico) y el manejo de excepciones de base de datos para prevenir la duplicidad de nombres mediante códigos de estado HTTP (409 Conflict), asegurando la integridad de la oferta comercial del ISP.

Figura 9

Módulo de Planes – interfaz grafica



ID	NOMBRE DEL PLAN	VELOCIDAD	VALOR	ACCIONES
1	Plan Residencial 15Mbps	15M/15M	\$35.000	 
2	Plan Residencial 20Mbps	20M/20M	\$40.000	 
3	Plan Residencial 30Mbps	30M/30M	\$45.000	 
4	Plan Residencial 40Mbps	40M/40M	\$50.000	 
5	Plan Residencial 50Mbps	50M/50M	\$55.000	 
6	Plan Residencial 100Mbps	100M/100M	\$65.000	 
7	Plan Residencial 200Mbps	200M/200M	\$100.000	 

Nota. El listado de planes residenciales configurados en el sistema, detallando el nombre del plan, la velocidad de navegación (Mbps) y el valor comercial. Se observa la disponibilidad de herramientas para la edición, eliminación y exportación de datos, permitiendo una administración ágil de la oferta de servicios del ISP.

Figura 10*Fragmento de código - Lógica para la Actualización y Modificación de Planes de servicio*

```

case 'update_plan':
    if (empty($data['plan_id']) || !is_numeric($data['plan_id']) || empty($data['nombre_plan']) || empty($data['velocidad']) || !isset($data['precio']) || !is_numeric($data['precio'])) {
        http_response_code(400); // Bad Request
        echo json_encode(['error' => 'ID, nombre del plan, velocidad y un precio numérico son requeridos para actualizar.']);
        exit;
    }
    $plan_id_to_update = (int)$data['plan_id'];
    try {
        $stmt = $pdo->prepare("UPDATE planes SET nombre_plan = ?, velocidad = ?, precio = ? WHERE plan_id = ?");
        $stmt->execute([trim($data['nombre_plan']), trim($data['velocidad']), (float)$data['precio'], $plan_id_to_update]);

        if ($stmt->rowCount() > 0) {
            http_response_code(200); // OK
            echo json_encode(['message' => 'Plan actualizado exitosamente.']);
        } else {
            // Si rowCount es 0, puede que el plan no exista o no hubo cambios
            // Se devuelve 200 OK con un mensaje informativo
            http_response_code(200);
            echo json_encode(['message' => 'Plan no encontrado o no se realizaron cambios.']);
        }
    } catch (PDOException $e) {
        if ($e->getCode() == 23000) { // Código para violación de unicidad (nombre_plan UNIQUE)
            http_response_code(409); // Conflict
            echo json_encode(['error' => 'El nombre de plan ya existe.']);
        } else {
            http_response_code(500); // Internal Server Error
            error_log("Error de BD al actualizar plan en api_plans.php: " . $e->getMessage());
            echo json_encode(['error' => 'Error al actualizar plan', 'detalle' => (ini_get('display_errors') == 1 ? $e->getMessage() : 'Error interno del servidor.')] );
        }
    }
    break;

```

Nota. El código gestiona la eliminación de planes utilizando transacciones SQL para borrar registros relacionados en las tablas de usuarios y facturas, garantizando que no queden datos huérfanos y aplicando un rollback en caso de error.

Figura 11*Fragmento de código -Lógica para la Eliminación de Planes de Servicio*

```

case 'delete_plan':
    if (empty($data['plan_id']) || !is_numeric($data['plan_id'])) {
        http_response_code(400); // Bad Request
        echo json_encode(['error' => 'ID de plan requerido para eliminar.']);
        exit;
    }
    $plan_id_to_delete = (int)$data['plan_id'];
    try {
        $pdo->beginTransaction();
        // Eliminar registros relacionados en 'usuario_planes'
        $stmt = $pdo->prepare("DELETE FROM usuario_planes WHERE plan_id = ?");
        $stmt->execute([$plan_id_to_delete]);
        // Eliminar registros relacionados en 'facturas'
        $stmt = $pdo->prepare("DELETE FROM facturas WHERE plan_id = ?");
        $stmt->execute([$plan_id_to_delete]);

        // Finalmente, eliminar el plan
        $stmt = $pdo->prepare("DELETE FROM planes WHERE plan_id = ?");
        $stmt->execute([$plan_id_to_delete]);
        $pdo->commit();
        http_response_code(200); // OK
        echo json_encode(['message' => 'Plan eliminado exitosamente.']);
    } catch (PDOException $e) {
        $pdo->rollBack();
        http_response_code(500); // Internal Server Error
        error_log("Error de BD al eliminar plan en api_plans.php: " . $e->getMessage());
        echo json_encode(['error' => 'Error al eliminar plan', 'detalle' => (ini_get('display_errors') == 1 ? $e->getMessage() : 'Error interno del servidor.')] );
    }
    break;

default:
    http_response_code(400); // Bad Request
    echo json_encode(['error' => 'Acción POST no reconocida.']);
    break;

```

Nota. El código implementa la función de actualización de datos en la base de datos, incluyendo la validación de campos obligatorios, el manejo de errores por nombres duplicados y la confirmación de cambios realizados.

5.13 Módulo de facturación

El módulo de generación y manejo de facturas está formulado como una pieza administrativa de un sistema que proporciona gestión de facturación al cliente y control del sistema de facturación de manera consolidada. Hay un resumen en la parte superior con categorías de facturas: generadas, pagadas, pendientes y canceladas. Las categorías de archivos se proporcionan con el número y costo de los documentos, por lo tanto, pueden proporcionar rápida y precisamente una visión general del comportamiento financiero de la organización.

En esta sección también se presenta un panel de control que proporciona indicadores vitales para la toma de decisiones y el monitoreo de la gestión contable. También puedes generar tus propias facturas individuales en la interfaz. Esto se hace utilizando un campo desplegable en el que se elige el tipo de facturación a aplicar. En cambio, esta función está adaptada para ciertas situaciones, como cuando se emiten facturas personalizadas o se corrigen notas de una manera no incluida en su proceso de facturación estándar. Así, el sistema no solo automatiza grandes procesos, sino que también ofrece flexibilidad en cómo manejar los casos específicos. La sección de detalle de facturas se encuentra en el corazón del módulo y lista todos los documentos emitidos. Todos los registros se almacenan en columnas que consisten en ID de factura, ID de usuario, nombre del cliente, plan o concepto de facturación, monto, fecha de emisión, estado actual y qué acciones se implementaron.

Estos campos desempeñan roles complementarios para permitir la identificación del cliente, el conocimiento del servicio o artículo facturado, la verificación del valor económico, el control del estado de pago y las acciones de gestión (intentos de editar, cancelar o reenviar la factura). Para mejorar el proceso de búsqueda y análisis, el sistema incluye configuraciones de filtro para encontrar diferentes facturas por ID de usuario o fecha. Esto es particularmente útil en

auditoría, conciliación o servicio al cliente porque permite localizar datos exactos más rápido en una gran cantidad de datos. El módulo también incluye un mecanismo para generar facturas masivas para la creación automatizada de documentos para todos los clientes que tienen planes activos.

Debido a que tienes una contraseña para confirmación, esto añade una capa de seguridad a las operaciones que no pueden deshacerse. Exportarlo en formato Excel también está incluido en el módulo para que las personas puedan descargar información en Excel, como se muestra en las figuras 12 a 16.

Figura 12

Módulo de Facturación – interfaz grafica

Panel de Administración Bienvenido(a), Super Admin (Usuario de Gestión) U

Generar y Gestionar Facturas [Generar Facturas Masivas](#)

Resumen de Facturas

Facturas Generadas: 61 Suma Total: \$2.215.000	Facturas Pagadas: 2 Suma Total: \$60.000	Facturas Pendientes: 59 Suma Total: \$2.155.000	Facturas Anuladas: 0 Suma Total: \$0
---	---	--	---

Generar Factura Individual

Tipo de Facturación:
-- Seleccione un tipo de facturación --

Detalle de Facturas (Todas)

ID FACTURA	ID USUARIO	USUARIO	PLAN / CONCEPTO	MONTO	FECHA	ESTADO	ACCIONES
061	84	LUIS RODRIGUEZ	Plan Residencial 20Mbps	\$20.000	2025-09-16	Pagada	Ver Eliminar
060	25	ALBERTO INSUASTY	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	Ver Eliminar

Nota. Ilustra el panel de control de facturación, el cual incluye un resumen estadístico de estados (pagadas, pendientes y anuladas). Se detallan las herramientas para la generación masiva de facturas, la creación de registros individuales y el listado histórico con información de usuarios, montos y fechas de vencimiento para un control financiero exhaustivo.

Figura 13

Ventana prefacturación – interfaz grafica

The screenshot displays a web application interface for generating and managing invoices. A modal dialog box titled 'Confirmación Requerida' is centered on the screen, prompting the user to enter a password to confirm the generation of invoices for all active clients. The background interface includes a summary of invoice statistics, a section for generating individual invoices, and a table of invoice details.

Generar y Gestionar Facturas

Resumen de Facturas

Facturas Generadas:	Facturas Pendientes:	Facturas Anuladas:
61	2	0
Suma Total: \$2.215.000	Suma Total: \$2.155.000	Suma Total: \$0

Generar Factura Individual

Tipo de Facturación:

-- Seleccione un tipo de facturación --

Detalle de Facturas (Todas)

ID FACTURA	ID USUARIO	USUARIO	MONTO	FECHA	ESTADO	ACCIONES
061	84	LUIS RODRIGUEZ	\$20.000	2025-09-16	Pagada	[Edit] [Delete]
060	25	ALBERTO INSUASTY	\$40.000	2025-09-16	Pendiente	[Edit] [Delete]
059	83	WILLIAM ANDRADE	\$35.000	2025-09-16	Pendiente	[Edit] [Delete]

Confirmación Requerida

Está a punto de generar la facturación para **TODOS** los clientes con planes activos. Esta acción no se puede deshacer.

Para confirmar, por favor escriba la **contraseña** en el campo de abajo.

Confirmar y Generar Facturas

Cancelar

Nota. Son módulos de seguridad y edición. La primera parte destaca el protocolo de confirmación requerida mediante contraseña para acciones globales, mientras que la segunda ilustra la herramienta de revisión masiva, donde se validan los nombres de clientes, planes asociados y montos finales antes de la generación del lote de facturas.

Figura 14

Fragmento de código - Lógica para la Ejecución y Registro de Facturación Masiva

```
// --- ACCIÓN: EJECUTAR GENERACIÓN FINAL DE FACTURAS EDITADAS ---
case 'execute_bulk_invoices':
    $invoices_to_insert = $data['invoices'] ?? [];
    $fecha_factura_bulk = trim($data['fecha'] ?? date('Y-m-d'));

    if (empty($invoices_to_insert)) {
        http_response_code(400);
        echo json_encode(['error' => 'No se recibieron facturas para generar.']);
        exit;
    }

    // Vuelve a verificar si ya se facturó este mes por seguridad
    $stmt_check = $pdo->prepare("SELECT COUNT(*) FROM facturas WHERE MONTH(fecha_factura) = MONTH(?) AND YEAR(fecha_factura) = YEAR(?)");
    $stmt_check->execute([$fecha_factura_bulk, $fecha_factura_bulk]);
    if ($stmt_check->fetchColumn() > 0) {
        http_response_code(400); // Conflict
        echo json_encode(['error' => 'Ya se generaron facturas para este mes. No se puede duplicar la facturación masiva.']);
        exit;
    }
}
```

Nota. Ventana emergente destinada a la modificación de valores económicos. Se destaca la validación del ID de factura y la previsualización del concepto actual, permitiendo al administrador ingresar un nuevo monto y actualizar la información de manera instantánea en el

Figura 15

Editar el valor de una factura generada.

The screenshot displays a web interface with a table titled "Detalle de Facturas (Todas)". The table has columns for ID FACTURA, ID USUARIO, USUARIO, PLAN / CONCEPTO, MONTO, FECHA, ESTADO, and ACCIONES. A modal window titled "Editar Monto de Factura" is open, showing the details for invoice ID 060. The modal includes fields for Factura ID (060), Concepto Actual (Plan Residencial 20Mbps), and Nuevo Monto (40000). There are "Cancelar" and "Guardar Monto" buttons at the bottom of the modal.

ID FACTURA	ID USUARIO	USUARIO	PLAN / CONCEPTO	MONTO	FECHA	ESTADO	ACCIONES
061	84	LUIS RODRIGUEZ	Plan Residencial 20Mbps	\$20.000	2025-09-16	Pagada	[Iconos]
060	25	ALBERTO INSUASTY	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	[Iconos]
059	83	WILLIAM ANDRADE	Plan Residencial 20Mbps	\$35.000	2025-09-16	Pendiente	[Iconos]
058	82	VIVIANA PORTILLO	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	[Iconos]
057	82	VIVIANA PORTILLO	Plan Residencial 20Mbps	\$35.000	2025-09-16	Pendiente	[Iconos]
056	81	SEBASTIAN RIASCOS CO	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	[Iconos]
055	80	SEBASTIAN INSUASTY	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	[Iconos]
054	79	ROSIO INSUASTY	Plan Residencial 15Mbps	\$35.000	2025-09-16	Pendiente	[Iconos]

listado general de facturación.

Nota. Implementa la generación final de facturas mediante transacciones SQL, incluyendo una verificación de seguridad para evitar duplicidad de cobros en el mismo periodo y el procesamiento en lote de los montos editados.

Figura 16

Fragmento de código - Lógica para la Actualización de Montos en Facturas Individuales

```
// --- NUEVA ACCIÓN: ACTUALIZAR SOLO MONTO DE FACTURA ---
case 'update_invoice_monto':
    $factura_id = filter_var($data['factura_id'] ?? null, FILTER_VALIDATE_INT);
    $monto = filter_var($data['monto'] ?? null, FILTER_VALIDATE_FLOAT);
    $modificado_por_usuario_id = $_SESSION['user_id'];

    if (!$factura_id || $factura_id <= 0 || $monto === null || $monto <= 0) {
        http_response_code(400);
        echo json_encode(['error' => 'ID de factura y monto positivo requeridos para actualizar.']);
        exit;
    }

    try {
        // Solo se actualiza el monto, y se registra la modificación.
        $stmt = $pdo->prepare("
            UPDATE facturas
            SET monto = ?, modificado_por = ?, fecha_modificacion = NOW()
            WHERE factura_id = ? AND estado = 'Pendiente'
        ");
        $stmt->execute([$monto, $modificado_por_usuario_id, $factura_id]);

        if ($stmt->rowCount() > 0) {
            http_response_code(200);
            echo json_encode(['message' => 'Monto de factura actualizado exitosamente.']);
        } else {
            http_response_code(400);
            echo json_encode(['error' => 'No se pudo actualizar el monto. La factura no existe o su estado no es Pendiente.']);
        }
    } catch (PDOException $e) {
        http_response_code(500);
        error_log("Error de BD al actualizar monto de factura: " . $e->getMessage());
        echo json_encode(['error' => 'Error al actualizar monto de factura', 'detalle' => (ini_get('display_errors') == 1 ? $e->getMessage() : 'Error interno del servidor.')]);
    }
    break;
// --- FIN NUEVA ACCIÓN: ACTUALIZAR SOLO MONTO DE FACTURA ---
```

Nota. El código permite modificar el valor de una factura específica en estado "Pendiente", registrando automáticamente el ID del administrador que realiza el cambio y la fecha exacta de la modificación para mantener la trazabilidad financiera.

5.14 Módulo de recaudos

El módulo de recaudo de la interfaz está principalmente involucrado con la gestión de pagos de usuarios recolectados a través de un sistema financiero o administrativo. La parte superior muestra un resumen que clasifica las facturas en dos estados principales: pagadas y pendientes. Un resumen del conteo de cada categoría y su monto total le da al administrador un fácil acceso al estado de la colección y una advertencia temprana para posibles retrasos o incumplimientos. Luego, vemos las "Facturas de Cobro Pendientes" - mostrando documentos aún no pagados por los usuarios. La tabla contiene columnas que incluyen ID de factura, ID de usuario, nombre de usuario, plan o concepto facturado, monto, fecha, estado y acciones que se pueden

tomar.

Cada celda realiza una función que es nombrar al cliente, registrar el monto adeudado, indicar la fecha de emisión, o para acciones administrativas, como el registro del pago, cancelación de la factura o verificación del historial. Esta arquitectura permite un control detallado sobre los detalles de cada transacción y ofrece toma de decisiones operativas. En particular, el sistema tiene filtros que permiten búsquedas de cada factura por su ID, nombre de usuario o rango de fechas. Este mecanismo puede ser particularmente útil para auditorías, para responder a demandas de clientes, o para verificar el estado de una factura en fechas particulares.

Además, hay un botón para ver las facturas que ya han sido pagadas o canceladas, ampliando el espectro desde el análisis de cobros actuales hasta una vista completa del historial de cobros. Por último, el módulo muestra una ventana emergente y pide al usuario seleccionar el estado de las facturas que desea consultar, ya sea "Pagadas" o "Canceladas".

Esto permite un acceso segmentado, de modo que acelera el proceso de búsqueda y ayuda a mantener un historial preciso. El sistema también incluye una opción de exportación en Excel que descarga la información del propio módulo, como se muestra en las Figuras 17-19.

Figura 17

Módulo de Recaudo – interfaz grafica

Panel de Administración Bienvenido(a), Super Admin (Usuario de Gestión)

Módulo de Recaudo

Resumen de Facturas

Facturas Pagadas: **2**
Suma Total: \$60.000

Facturas Pendientes: **59**
Suma Total: \$2.155.000

Facturas Pendientes de Recaudo

ID FACTURA	ID USUARIO	USUARIO	PLAN / CONCEPTO	MONTO	FECHA	ESTADO	ACCIONES
060	25	ALBERTO INSUASTY	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	<button>Cambiar Estado</button>
059	83	WILLIAM ANDRADE	Plan Residencial 15Mbps	\$35.000	2025-09-16	Pendiente	<button>Cambiar Estado</button>
058	82	VIVIANA PORTILLO	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	<button>Cambiar Estado</button>
057	82	VIVIANA PORTILLO	Plan Residencial 15Mbps	\$35.000	2025-09-16	Pendiente	<button>Cambiar Estado</button>
056	81	SEBASTIAN RIASCOS CORD	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	<button>Cambiar Estado</button>
055	80	SEBASTIAN INSUASTY	Plan Residencial 20Mbps	\$40.000	2025-09-16	Pendiente	<button>Cambiar Estado</button>
054	79	ROSIO INSUASTY	Plan Residencial 15Mbps	\$35.000	2025-09-16	Pendiente	<button>Cambiar Estado</button>

Cambiar Estado de Factura

Realizado por: Super Admin

Estado Actual: Pendiente

Nuevo Estado: Pagada

Guardar Cancelar

Nota. El panel de gestión de recaudo, donde se presenta un resumen estadístico de saldos y el listado de facturas pendientes. Se destaca la ventana modal para el cambio de estado, que registra al administrador responsable y permite la actualización a "Pagada", facilitando el control de ingresos y la auditoría de transacciones financieras.

Figura 18*Historial de facturas Pagadas/Anuladas*

Facturas - Pagadas

TOTAL: \$10.655.000

Buscar por ID o Nombre de Usuario

-- Filtrar por Modificador --

30/12/2025 20/01/2026 Filtrar Limpiar

ID FACTURA	ID USUARIO	USUARIO	PLAN / CONCEPTO	MONEDA	FECHA	MODIFICADO POR	FECHA MODIFICACIÓN	RECIBO
315	85	DANIELA APRAEZ	Servicio Plan Residencial 15Mbps - Enero 2026	\$35,000	2026-01-01	Yanira Insuasty	7/1/2026, 6:23:32 p. m.	Recibo
314	86	LEIDY INSUASTY CAEZ	Servicio Plan Residencial 15Mbps - Enero 2026	\$35,000	2026-01-01	Yanira Insuasty	10/1/2026, 2:20:25 p. m.	Recibo
313	46	HAROL CADENA	Servicio Plan Economico - Enero 2026	\$30,000	2026-01-01	Eduard Insuasty	11/1/2026, 10:19:38 a. m.	Recibo
312	55	JUAN CARLOS GONZALEZ	Servicio Plan Residencial 20Mbps - Enero 2026	\$40,000	2026-01-01	Yanira Insuasty	14/1/2026, 9:23:46 a. m.	Recibo
311	80	SEBASTIAN INSUASTY	Servicio Plan Economico - Enero 2026	\$30,000	2026-01-01	Eduard Insuasty	10/1/2026, 6:56:57 p. m.	Recibo
310	49	JAVIER CABRERA	Servicio Plan Residencial 20Mbps - Enero 2026	\$40,000	2026-01-01	Eduard Insuasty	8/1/2026, 3:48:18 p. m.	Recibo

Nota. Se presenta la ventana del historial financiero, donde se consolida el recaudo total y se listan los comprobantes emitidos. Se destaca la columna de "Modificado por", que vincula cada registro con el administrador responsable

Figura 19*Fragmento de código – Actualizar el estado de la factura*

```
// --- Actualizar Estado de una Factura ---
case 'update_invoice_status':

    if (!isset($_SESSION['user_id']) || empty($_SESSION['user_id'])) {
        http_response_code(401); // Unauthorized
        echo json_encode(['error' => 'Sesión no válida o expirada. Por favor, inicie sesión de nuevo.']);
        exit;
    }

    $factura_id = $_data['factura_id'] ?? null;
    $estado = $_data['estado'] ?? null;
    $modificado_por = $_SESSION['user_id'];

    if (!$factura_id || !$estado) {
        http_response_code(400);
        echo json_encode(['error' => 'ID de factura y nuevo estado son requeridos.']);
        exit;
    }

    $stmt = $pdo->prepare("UPDATE facturas SET estado = ?, modificado_por = ?, fecha_modificacion = NOW() WHERE factura_id = ?");
    $stmt->execute([$estado, $modificado_por, $factura_id]);

    if ($stmt->rowCount() > 0) {
        echo json_encode(['message' => 'Estado de la factura actualizado exitosamente.']);
    } else {
        echo json_encode(['message' => 'No se realizaron cambios o la factura no fue encontrada.']);
    }
    break;
}
```

Nota. El código implementa un motor de búsqueda y filtrado que procesa parámetros como ID de factura, nombre de usuario y fechas.

5.15 Módulo de ingresos y gastos

El módulo de gestión de ingresos y gastos proporciona una vista clara y estructurada del estado financiero del ISP. En la parte superior de la interfaz se encuentra el "Estado de Resultados" que ilustra visualmente los ingresos totales y los gastos totales para ser comparados y generar automáticamente la ganancia o pérdida. Esta sección permite evaluar el desempeño económico durante un período definido, lo que podría llevar a la toma de decisiones estratégicas.

Se añaden filtros de fecha con los que el análisis puede restringirse a un cierto rango (bueno para usar mensualmente, trimestralmente o incluso anualmente) al software. Justo al lado está la sección de "Gastos Mensuales (Recurrentes)" donde se registran los gastos. Cada gasto se coloca en una categoría específica, se describe, se asigna un costo particular y se ofrece una opción para editar o eliminar, proporcionando control personal de las obligaciones financieras.

El sistema permite añadir un nuevo gasto a través de un botón dedicado y registrar un gasto separado para cualquier momento seleccionado en el tiempo. Esto permite contabilizar todos los gastos a tiempo en ese período. La interfaz amigable también fomenta la interacción del usuario con los datos, permitiendo una gestión eficiente y ordenada. Por ejemplo, el módulo de historial de gastos registrados permite consultar todos los gastos que ya están ingresados.

Esta sección añade filtros de fecha y muestra una tabla con columnas para el ID del gasto, categoría, descripción, monto, fecha y las acciones disponibles. Al eliminar registros directamente desde esta vista, se pueden evitar errores o depurar información. También tiene exportación en formato Excel; puede descargar los datos para tareas administrativas como la generación de informes, control interno y auditorías (Figuras 20 y 21).

Figura 20

Módulo de Ingresos y Gastos – Interfaz Gráfica

Panel de Administración Bienvenido(a), Super Admin (Usuario de Gestión)

Gestión de Ingresos y Gastos

Estado de Resultados (Ingresos - Gastos)

Ingresos Totales:
\$60.000

Gastos Totales:
\$769.050

Ganancia/Pérdida:
\$-709.050

Desde: dd/mm/aaaa Hasta: dd/mm/aaaa Filtrar Limpiar

Gastos Mensuales (Recurrentes)

+ Añadir Gasto a la Lista

Fecha para registrar los gastos seleccionados:
20/01/2026 Registrar Gastos Seleccionados

☐	CATEGORÍA	DESCRIPCIÓN	MONTO	ACCIONES
☐	ARRIENDO	ANDREA INSUASTY	\$55.000	✎ ✖
☐	ARRIENDO	BEATRIZ INSUASTY	\$25.000	✎ ✖

Historial de Gastos Registrados

Buscar por Categoría o ID

ID	CATEGORÍA	DESCRIPCIÓN	MONTO	FECHA	ACCIONES
36	INTERNET	CONEXION FIBRA SAS	\$589.050	2026-02-18	✖
28	ARRIENDO	DIEGO INSUASTY	\$20.000	2026-02-02	✖
35	INTERNET	CONECTIVA SAS	\$80.000	2026-01-25	✖
31	ARRIENDO	ANDREA INSUASTY	\$200.000	2026-01-22	✖
30	ARRIENDO	BEATRIZ INSUASTY	\$25.000	2026-01-16	✖
29	INTERNET	CONEXION FIBRA SAS	\$589.050	2026-01-16	✖
27	ARRIENDO	DIEGO INSUASTY	\$20.000	2026-01-15	✖

Nota. Panel de gestión de egresos, donde se listan cronológicamente las transacciones financieras de salida. Se observa la categorización de gastos y la descripción de conceptos operativos, incluyendo herramientas para la eliminación de registros y la supervisión de montos, lo que facilita el control presupuestario y la auditoría de costos fijos y variables.

Figura 21

Fragmento de código: Modulo de ingresos y gastos - Lógica de Backend para el Inicio de Sesión y Autenticación de Usuarios

```

$data = [];
if ($method === 'POST') {
    $input = file_get_contents('php://input');
    if (!empty($input)) {
        $data = json_decode($input, true);
        if (json_last_error() !== JSON_ERROR_NONE) {
            http_response_code(400);
            echo json_encode(['error' => 'Formato JSON inválido en la solicitud.', 'detalle' => json_last_error_msg()]);
            exit;
        }
    }
}

$action = $_GET['action'] ?? ($data['action'] ?? '');

try {
    switch ($action) {
        // --- ACCIONES PARA GASTOS REGISTRADOS (HISTORIAL) ---
        case 'get_expenses':
            $start_date = trim($_GET['start_date'] ?? '');
            $end_date = trim($_GET['end_date'] ?? '');

            $sql = "SELECT expense_id, category, description, amount, date FROM gastos";
            $conditions = [];
            $params = [];

            $stmt = $pdo->prepare($sql);
            $stmt->execute($params);
            $expenses = $stmt->fetchAll(PDO::FETCH_ASSOC);
            echo json_encode($expenses);
            break;

        case 'delete_expense': // Eliminar del historial de gastos
            $id = filter_var($data['expense_id'] ?? null, FILTER_VALIDATE_INT);
            if (!$id || $id <= 0) {
                http_response_code(400);
                echo json_encode(['error' => 'ID de gasto inválido para eliminar.']);
                exit();
            }
            $stmt = $pdo->prepare("DELETE FROM gastos WHERE expense_id = ?");
            $stmt->execute([$id]);
            if ($stmt->rowCount() > 0) {
                echo json_encode(['message' => 'Gasto del historial eliminado exitosamente.']);
            } else {
                http_response_code(404);
                echo json_encode(['error' => 'Gasto no encontrado en el historial.']);
            }
            break;
    }
}

```

Nota. El código gestiona la autenticación mediante la verificación de credenciales y el estado activo del usuario. Utiliza la función `password_verify` para comparar el hash de la contraseña y, tras una validación exitosa, inicializa la sesión de PHP almacenando el rol y los permisos necesarios para el control de acceso al sistema.

5.16 Módulo de inventario de activos

El módulo de Inventario y Asignación de Activos se ocupa de los recursos, incluyendo la gestión del equipo y materiales almacenados en el sistema y cómo los recursos existentes se distribuyen a los clientes y para el uso interno de la empresa. Desde este módulo, se manejan los activos físicos para crear un sistema que asegure el control sobre el inventario de existencias, la

disponibilidad y el historial de uso, útil para la gestión técnica y administrativa con precisión.

La sección Agregar Activo al Inventario proporciona la capacidad de registrar nuevos activos; los usuarios seleccionan un tipo existente e ingresan la cantidad de activos a agregar, con medidas de control que facilitan el aumento o disminución de unidades. Esto ayuda a mantener el inventario actualizado cuando se agregan nuevas máquinas o recursos, de modo que las cantidades no sean diferentes cuando se implementen nuevos equipos o materiales. La opción de Inventario Disponible para Asignación incluye una lista de los activos registrados, incluyendo su cantidad total y disponible, para la detección rápida de lo que se puede usar como recursos.

Los activos pueden ser asignados directamente o eliminados del inventario y utilizados como un punto de datos exportado a Excel a través de esta tabla para control, análisis o para respaldo administrativo. Por último, esta parte del módulo tendrá la sección de Activos Asignados donde se mostrará el historial de asignaciones realizadas. Esta tabla muestra a quién se asignó cada activo, la fecha, dirección IP, número de serie o MAC, y observaciones relevantes realizadas.

Permite el mapeo y monitoreo completo del equipo, auditorías y gestión adecuada de las instalaciones tecnológicas propiedad de la empresa, como se evidencia en los datos que muestran el equipo (Figuras 22 a 26).

Figura 22*Módulo de Inventario de Activos – Interfaz Gráfica*

Panel de Administración Bienvenido(a), Super Admin (Usuario de Gestión)

Adicionar Activo al Inventario

Tipo de Activo: Cantidad:

Inventario Disponible para Asignar

NOMBRE DEL ACTIVO	CANTIDAD TOTAL	CANTIDAD DISPONIBLE	ACCIONES
LiteBeam M5-23	60	57	Asignar Eliminar
Ponchadora RJ45	1	1	Asignar Eliminar
Repetidora LiteGPS 90	3	2	Asignar Eliminar
Route Wifi	30	30	Asignar Eliminar
Router Mikrotik	3	0	Asignar Eliminar

Activos Asignados

Buscar por Nombre, IP, Serial...

ID ASIG.	ACTIVO	ASIGNADO A	FECHA	IP	SERIAL/MAC	NOTAS	ACCIONES
11	Router Mikrotik	Empresa (Uso Interno)	2025-09-12	192.168.4.1	N/A	El Rosario	Editar Eliminar
12	LiteBeam M5-23	JUAN CARLOS GONZALEZ	2025-09-12	192.168.6.123	N/A	El Rosario	Editar Eliminar
13	LiteBeam M5-23	JAVIER CABRERA	2025-09-12	192.168.6.174	N/A	San Jose	Editar Eliminar

Nota. La interfaz integra tres funciones: adición de equipos mediante formulario, monitoreo de stock disponible en bodega y el registro detallado de activos asignados con sus respectivos seriales e IPs para control técnico.

Figura 23*Tipo de activo*

Adicionar Activo al Inventario

Tipo de Activo: Cantidad:

-- Agregar Nuevo Tipo de Activo --

- LiteBeam M5-23
- Ponchadora RJ45
- Repetidora LiteGPS 90
- Route Wifi
- Router Mikrotik

Nota. Se visualiza el menú desplegable para la categorización de hardware en el inventario. Esta función permite estandarizar el ingreso de equipos como routers, antenas LiteBeam o repetidoras, optimizando la organización del stock en bodega.

Figura 24*Asignar un activo*

Nota. Muestra las ventanas para la entrega de activos, diferenciando entre el uso interno de la empresa y la asignación directa a clientes. Se registran parámetros críticos como direcciones IP, números de serie y georreferenciación de la instalación.

Figura 25*Agregar nuevo tipo de activo y adicionar cantidades*

Adicionar Activo al Inventario

Nota. La imagen ilustra los dos procesos principales del módulo de inventario: en la sección superior se muestra la creación de un nuevo tipo de activo (ej. Switch de 8 puertos), mientras que en la sección inferior se detalla el incremento de stock para un activo ya registrado (ej. LiteBeam M5-23) mediante el selector de cantidades.

Figura 26

Fragmento de código - Adicionar un Activo al inventario - Lógica de Backend para el Cierre de Sesión y Finalización de Actividad

```

case 'get_tipos_activos':
    $stmt = $pdo->query("SELECT id, nombre, cantidad_total, cantidad_disponible FROM tipos_activos ORDER BY nombre ASC");
    $tipos_activos = $stmt->fetchAll(PDO::FETCH_ASSOC);
    echo json_encode($tipos_activos);
    break;

case 'add_stock':
    $tipo_id = $data['tipo_id'] ?? null;
    $nuevo_nombre = trim($data['nuevo_nombre'] ?? '');
    $cantidad = filter_var($data['cantidad'] ?? 0, FILTER_VALIDATE_INT);

    if ($cantidad <= 0) {
        http_response_code(400);
        echo json_encode(['error' => 'La cantidad debe ser un número positivo.']);
        exit;
    }

    $pdo->beginTransaction();

    if (!empty($nuevo_nombre)) {
        $stmt = $pdo->prepare("INSERT INTO tipos_activos (nombre, cantidad_total, cantidad_disponible) VALUES (?, ?, ?)");
        $stmt->execute([$nuevo_nombre, $cantidad, $cantidad]);
    } elseif (!empty($tipo_id)) {
        $stmt = $pdo->prepare("UPDATE tipos_activos SET cantidad_total = cantidad_total + ?, cantidad_disponible = cantidad_disponible + ? WHERE id = ?");
        $stmt->execute([$cantidad, $cantidad, $tipo_id]);
    } else {
        $pdo->rollBack();
        http_response_code(400);
        echo json_encode(['error' => 'Debe seleccionar un activo existente o proporcionar un nombre para uno nuevo.']);
        exit;
    }

    $pdo->commit();
    echo json_encode(['message' => 'Inventario actualizado exitosamente.']);
    break;

```

Note. El código ejecuta la terminación segura de la sesión mediante la limpieza del arreglo `$_SESSION` y la destrucción del identificador en el servidor, redirigiendo automáticamente al usuario hacia la interfaz de acceso principal.

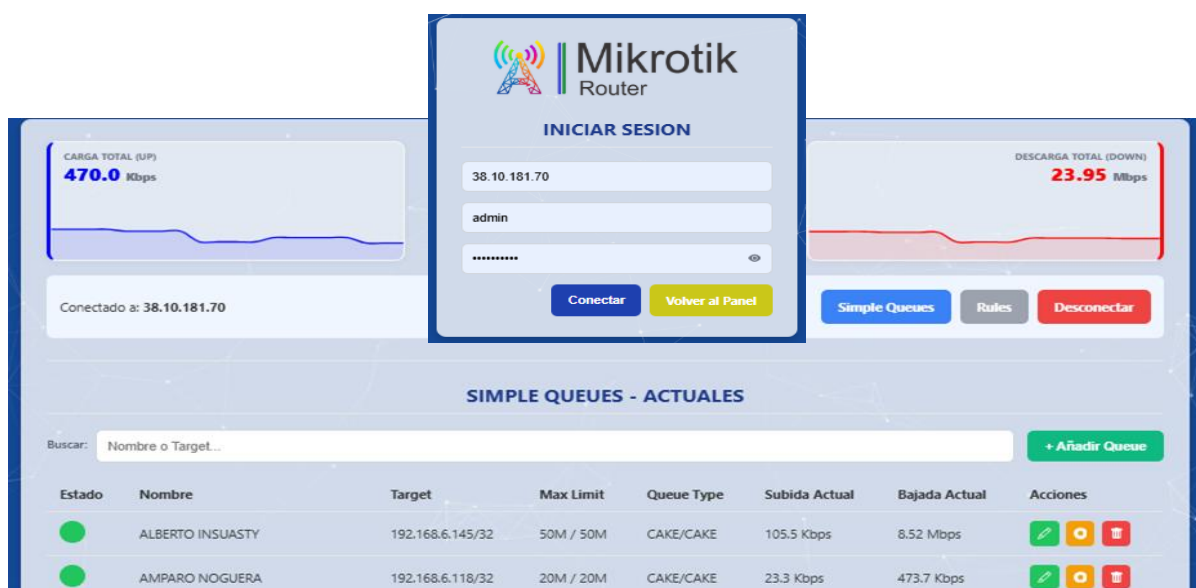
5.17 Módulo de Integración con Mikrotik (Queues)

Existen varios elementos del Módulo de Gestión de Colas de Mikrotik para gestionar las conexiones de red de los clientes en Mikrotik y el router, y controlar el ancho de banda, permitiendo a los usuarios asignar restricciones de carga y descarga individualmente según su conexión al router. Desde esta interfaz, se puede iniciar una conexión directa con el router Mikrotik, lo que permite un control eficiente de las colas: una vista para gestionar las colas desde lejos y hacerlo de manera centralizada sin necesidad de ingresar directamente a la consola del

dispositivo. En la sección Añadir Nueva Cola, el administrador puede agregar una nueva regla para el control de tráfico e introducir nueva información como el nombre del cliente, la dirección IP objetivo, el límite en archivos subidos y descargados, la prioridad y el tipo de cola. Los usuarios pueden utilizar esta función para aplicar políticas de velocidad basadas en el plan que el cliente contrató para distribuir su ancho de banda de manera uniforme y controlada. Las Colas Simples actuales consisten en listas de colas existentes y configuradas en el router, que indican el estado, nombre, dirección IP, límite máximo configurado, tipo de cola y consumo actual de carga y descarga. Además, se ofrecen opciones para editar, pausar o eliminar colas, lo que permite monitorear y controlar el tráfico de red de manera continua (figuras 27-29).

Figura 27

Módulo de Integración con MikroTik (Queues) – interfaz Gráfica



Nota. Presenta el panel de control para la gestión de ancho de banda mediante MikroTik. Se observa el formulario de inicio de sesión al router, gráficas de tráfico en tiempo real (carga y descarga) y la tabla de "Simple Queues" donde se monitorea el estado, límite máximo y consumo actual de los usuarios conectados a la red.

Figura 28

Fragmento de código - Módulo de Queues - Función de Utilidad para la Verificación de Conectividad (Ping)

```
// --- Lógica para la vista de QUEUES ---
if ($view === 'queues') {
    $command_types = "$python_path $script_path $ip $user $pass get_queue_types";
    $response_types_json = shell_exec($command_types);
    if ($response_types_json) {
        $response_types = json_decode($response_types_json, true);
        if ($response_types && $response_types['status'] === 'success') {
            $queue_types = $response_types['data'];
        }
    }
}

if (isset($_GET['edit_id'])) {
    $edit_id = escapeshellarg($_GET['edit_id']);
    $command = "$python_path $script_path $ip $user $pass get_one $edit_id";
    $response_json = shell_exec($command);
    if ($response_json) {
        $response = json_decode($response_json, true);
        if ($response && $response['status'] === 'success' && $response['data']) {
            $editing_queue = $response['data'];
            if (isset($editing_queue['max-limit']) && strpos($editing_queue['max-limit'], '/') !== false) {
                list($upload, $download) = explode('/', $editing_queue['max-limit']);
                $editing_queue['max_limit_upload'] = format_bytes($upload);
                $editing_queue['max_limit_download'] = format_bytes($download);
            } else {
                $editing_queue['max_limit_upload'] = '0';
                $editing_queue['max_limit_download'] = '0';
            }
        }
    }
}
```

Nota. El código implementa la función pingadles, la cual utiliza el comando del sistema operativo para verificar la latencia y disponibilidad de una dirección IP. Retorna un valor booleano tras procesar la respuesta del comando, permitiendo al sistema determinar si un nodo de la red está activo.

Figura 29

fragmento de código de acciones de las Queues simple - Lógica para la Administración de Queue Simple y Queue Tree

```
// Lógica para acciones de QUEUES
if ($action === 'add' && $_SERVER['REQUEST_METHOD'] === 'POST') {
    $name = escapeshellarg($_POST['name']);
    $target = escapeshellarg($_POST['target']);
    $max_limit = escapeshellarg($_POST['max_limit_upload'] . '/' . $_POST['max_limit_download']);
    $priority = escapeshellarg($_POST['priority'] . '/' . $_POST['priority']);
    $queue_type = escapeshellarg($_POST['queue_type']);
    $command = "$python_path $script_path $ip $user $pass add $name $target $max_limit $priority $queue_type";
}
elseif ($action === 'edit' && $_SERVER['REQUEST_METHOD'] === 'POST') {
    $id = escapeshellarg($_POST['id']);
    $name = escapeshellarg($_POST['name']);
    $target = escapeshellarg($_POST['target']);
    $max_limit = escapeshellarg($_POST['max_limit_upload'] . '/' . $_POST['max_limit_download']);
    $priority = escapeshellarg($_POST['priority'] . '/' . $_POST['priority']);
    $queue_type = escapeshellarg($_POST['queue_type']);
    $command = "$python_path $script_path $ip $user $pass edit $id $name $target $max_limit $priority $queue_type";
}
elseif ($action === 'delete' && isset($_GET['id'])) {
    $id = escapeshellarg($_GET['id']);
    $command = "$python_path $script_path $ip $user $pass delete $id";
}
elseif ($action === 'suspend' && isset($_GET['id'])) {
    $id = escapeshellarg($_GET['id']);
    $command = "$python_path $script_path $ip $user $pass suspend $id";
}
elseif ($action === 'activate' && isset($_GET['id'])) {
    $id = escapeshellarg($_GET['id']);
    $command = "$python_path $script_path $ip $user $pass activate $id";
}
```

Nota. El código procesa las solicitudes de creación, actualización y eliminación de colas de tráfico. Utiliza la ejecución de scripts externos para interactuar con la API de MikroTik, permitiendo la gestión dinámica del ancho de banda y la prioridad de los servicios directamente desde el sistema.

5.18 Módulo de monitoreo de red

Este es el sistema de vigilancia proactiva. Un script automatizado (cron Job) ejecuta un comando ping cada 60 segundos a cada nodo y dispositivo hijo registrado. La respuesta se almacena en la base de datos y se visualiza en un mapa interactivo:

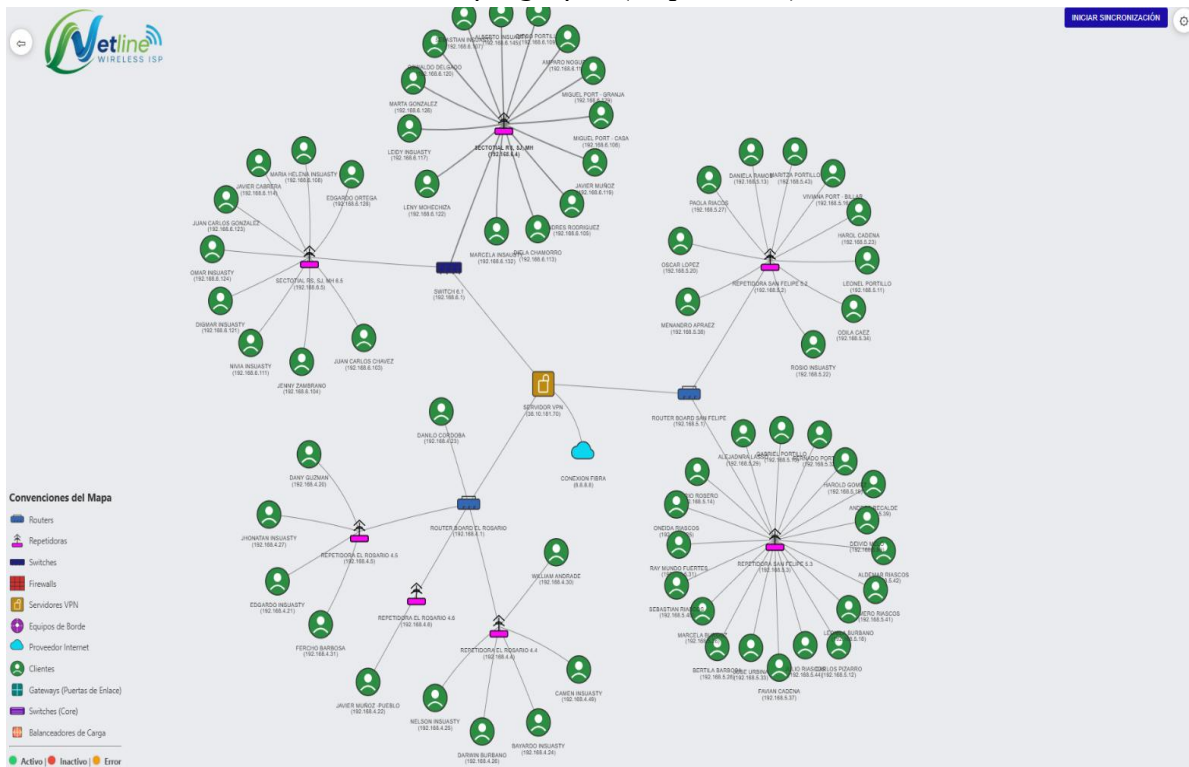
Verde: El dispositivo está en línea (UP).

Rojo: El dispositivo no responde (DOWN).

Esta visualización permite al equipo técnico identificar y localizar fallos de manera instantánea, como se observa en las figuras 30 al 42

Figura 30

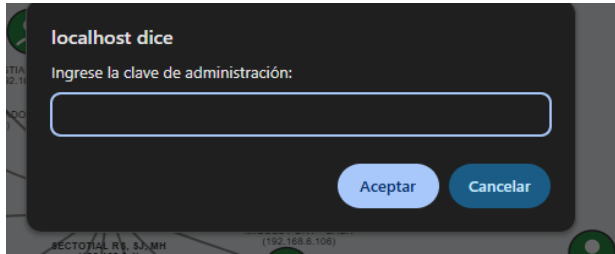
Módulo de Monitoreo de Red – interfaz gráfica (mapa de red)



Nota. Muestra una representación jerárquica de la infraestructura de red de Netline Wireless ISP. Se detalla la interconexión entre enrutadores, repetidoras y clientes finales mediante nodos; además, incluye una leyenda de convenciones en el costado izquierdo que permite identificar el estado operativo (activo, inactivo o error) de los diferentes dispositivos y servidores VPN dentro de la topología.

Figura 31

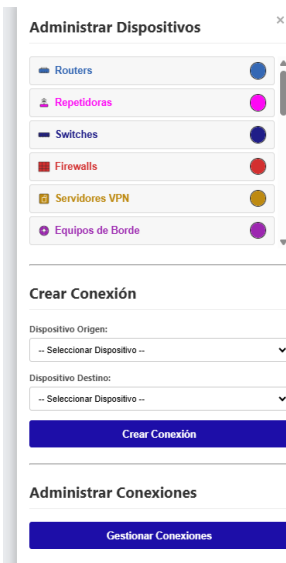
Ventana Emergente de Autenticación para la Administración del Panel



Nota. Cuadro de diálogo de validación de seguridad del sistema. Este mecanismo actúa como una capa de protección adicional que solicita una clave de administración antes de permitir el acceso a funciones sensibles o de configuración global dentro del panel de control del servidor local.

Figura 32

Panel de Administración de Dispositivos y Configuración de Conexiones



Nota. Menú lateral de gestión de red, el cual se divide en tres categorías operativas: la administración de hardware por tipos (Routers, Switches, Firewalls, etc.), un formulario dinámico para la creación de enlaces lógicos mediante la selección de dispositivos de origen y destino, y el acceso directo al módulo de gestión de conexiones existentes.

Figura 33

Interfaz del Panel de Administración de Clientes y Edición de Dispositivos

The screenshot displays a web application interface for managing clients and devices. The main section is titled 'Administrar Clientes' and contains a 'Lista Actual' (Current List) of clients. Each client entry includes their name and IP address, followed by edit and delete icons. Below the list is a section titled 'Editando Dispositivo' (Editing Device) with fields for Name, IP Address, Physical Address, and Observations. A 'Guardar Cambios' (Save Changes) button is at the bottom.

Nombre	Dirección IP	Acciones
DANY GUZMAN	(192.168.4.20)	[Edit] [Delete]
EDGARDO INSUASTY	(192.168.4.21)	[Edit] [Delete]
JAVIER MUÑOZ -PUEBLO	(192.168.4.22)	[Edit] [Delete]
DANILO CORDOBA	(192.168.4.23)	[Edit] [Delete]
BAYARDO INSUASTY	(192.168.4.24)	[Edit] [Delete]
NELSON INSUASTY	(192.168.4.25)	[Edit] [Delete]
DARWIN BURBANO	(192.168.4.26)	[Edit] [Delete]
JHONATAN INSUASTY	(192.168.4.27)	[Edit] [Delete]
WILLIAM ANDRADE	(192.168.4.30)	[Edit] [Delete]
FERCHO BARBOSA	(192.168.4.31)	[Edit] [Delete]

Editando Dispositivo

Nombre:

Dirección IP:

Dirección:

Observaciones:

Guardar Cambios

Nota. Módulo de gestión de usuarios, el cual incluye una lista detallada de clientes con sus respectivas direcciones IP y opciones de edición o eliminación. En la parte inferior, se muestra el formulario de edición que permite modificar campos específicos como el nombre, dirección IP, dirección física y observaciones técnicas del dispositivo vinculado al cliente.

Figura 34

Interfaz del Panel para la Administración y Edición de Conexiones de Red

Administrar Conexiones

Lista de Conexiones

Filtrar por Origen/Destino:

CONEXION FIBRA ↔ SERVIDOR VPN		
REPETIDORA EL ROSARIO 4.4 ↔ BAYARDO INSUASTY		
REPETIDORA EL ROSARIO 4.4 ↔ CAMEN INSUASTY		
REPETIDORA EL ROSARIO 4.4 ↔ DARWIN BURBANO		
REPETIDORA EL ROSARIO 4.4 ↔ NELSON INSUASTY		
REPETIDORA EL ROSARIO 4.4 ↔ WILLIAM ANDRADE		
REPETIDORA EL ROSARIO 4.5 ↔ DANY GUZMAN		
REPETIDORA EL ROSARIO 4.5 ↔ EDGARDO INSUASTY		
REPETIDORA EL ROSARIO 4.5 ↔ FERCHO BARBOSA		
REPETIDORA EL ROSARIO 4.5 ↔ JHONATAN INSUASTY		

Editar Conexión

Dispositivo Origen:

Dispositivo Destino:

Guardar Cambios

Nota. En la sección superior se presenta la lista de conexiones activas entre servidores, repetidoras y clientes, permitiendo filtrar por origen o destino; en la sección inferior, se despliega el formulario para editar la topología mediante la reasignación de dispositivos de origen y destino a través de menús desplegables.

Figura 35

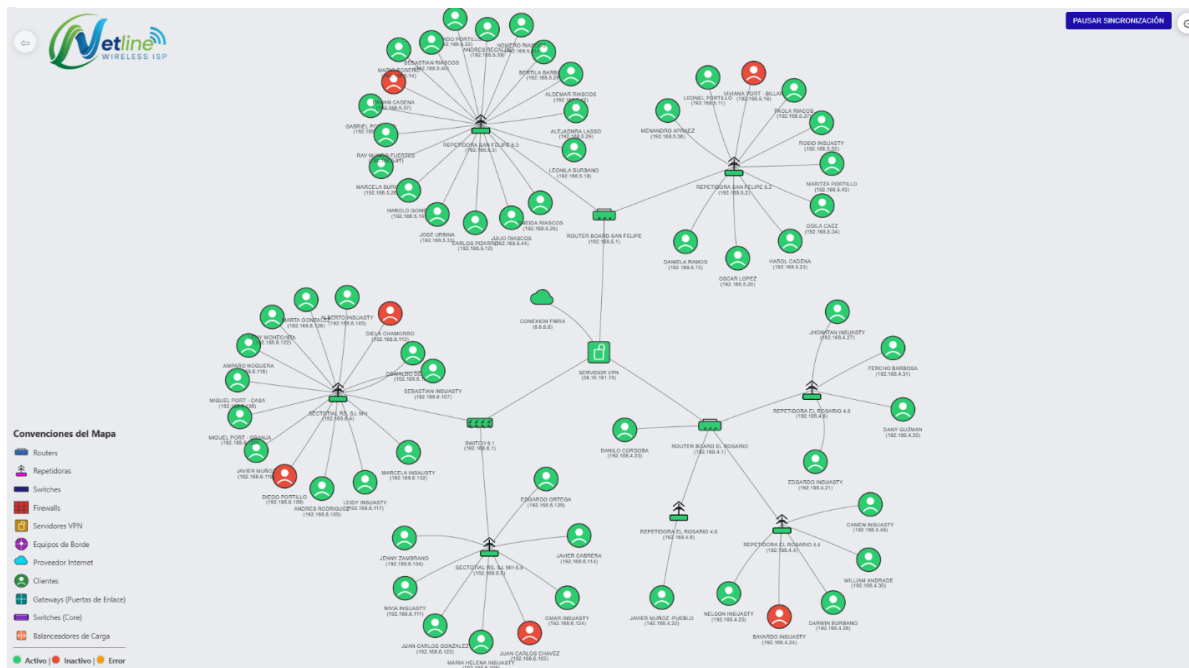
Fragmento de código - Lógica para la Obtención y Ordenamiento de Dispositivos de Red

```
case 'obtener_todos_dispositivos':
$stmt = $pdo->query("SELECT id, ip, nombre FROM dispositivos ORDER BY INET_ATON(ip)");
echo json_encode($stmt->fetchAll(PDO::FETCH_ASSOC));
break;
```

Nota. El código ejecuta una consulta SQL para recuperar el listado completo de dispositivos registrados. Utiliza la función INET_ATON (ir) para asegurar que el ordenamiento por dirección IP sea numéricamente correcto, facilitando la administración jerárquica de los nodos en la infraestructura del ISP.

Figura 36

Visualización del Mapa de Red Durante el Proceso de Sincronización Activa



Nota. El estado de la topología de red una vez iniciada la función de sincronización. En este estado, el sistema realiza un escaneo en tiempo real de los nodos, permitiendo identificar mediante alertas visuales en color rojo aquellos dispositivos que presentan fallos de conexión o estados de error, mientras que el botón superior derecho cambia su estado para permitir pausar el monitoreo.

Figura 37

Fragmento de código para agregar dispositivos

```
case 'agregar_dispositivo':
    $ip = trim($_POST['ip'] ?? '');
    $nombre = trim($_POST['nombre'] ?? '');
    $tipo = $_POST['tipo'] ?? '';
    $direccion = trim($_POST['direccion'] ?? null);
    $observaciones = trim($_POST['observaciones'] ?? null);
```

Nota. El código procesa la actualización de permisos mediante una transacción que elimina los privilegios previos y registra los nuevos módulos autorizados para un usuario específico, garantizando la consistencia de los niveles de acceso en el sistema.

Figura 38

Fragmento de código para eliminar dispositivos

```
case 'eliminar_dispositivo':
    $id = $_POST['id'] ?? 0;
    if ($id) {
        $pdo->prepare("DELETE FROM dispositivos WHERE id = ?")->execute([$id]);
        $pdo->prepare("DELETE FROM conexiones WHERE origen_id = ? OR destino_id = ?")->execute([$id, $id]);
        echo json_encode(['success' => true]);
    } else {
        echo json_encode(['success' => false, 'message' => 'ID no proporcionado.']);
    }
    break;
```

Nota. El código procesa la eliminación definitiva de un nodo de la infraestructura. Para mantener la integridad de la base de datos, el sistema ejecuta una eliminación en cadena que remueve tanto el registro del dispositivo como cualquier conexión lógica (origen o destino) vinculada a su identificador.

Figura 39

Fragmento de código para agregar conexiones - Lógica para la Actualización de Dispositivos de Red

```
case 'agregar_conexion':
    $origen_id = $_POST['origen_id'] ?? 0;
    $destino_id = $_POST['destino_id'] ?? 0;
    if ($origen_id && $destino_id && $origen_id != $destino_id) {
        $stmt = $pdo->prepare("INSERT INTO conexiones (origen_id, destino_id) VALUES (?, ?)");
        $stmt->execute([$origen_id, $destino_id]);
        echo json_encode(['success' => true, 'message' => 'Conexión creada correctamente.']);
    } else {
        echo json_encode(['success' => false, 'message' => 'Error en datos de conexión.']);
    }
    break;
```

Nota. El código procesa la edición de la información técnica de los dispositivos (nodos) en la base de datos. Valida la integridad de la dirección IP y el nombre del equipo, ejecutando una actualización mediante sentencias preparadas para asegurar la persistencia de los cambios en la topología de red.

Figura 40

Fragmento de código para editar conexiones

```
case 'editar_conexion':
    $id = $_POST['id'] ?? 0;
    $origen_id = $_POST['origen_id'] ?? 0;
    $destino_id = $_POST['destino_id'] ?? 0;
    if ($id && $origen_id && $destino_id && $origen_id != $destino_id) {
        $stmt = $pdo->prepare("UPDATE conexiones SET origen_id = ?, destino_id = ? WHERE id = ?");
        $stmt->execute([$origen_id, $destino_id, $id]);
        echo json_encode(['success' => true, 'message' => 'Conexión actualizada.']);
    } else {
        echo json_encode(['success' => false, 'message' => 'Error al editar conexión.']);
    }
    break;
```

Nota. Procesa la creación de enlaces lógicos entre dispositivos mediante una transacción SQL. Verifica que no existan conexiones duplicadas entre los mismos nodos y registra los parámetros de interfaz y estado, asegurando la integridad de la topología en la base de datos.

Figura 41

Fragmento de código para eliminar conexiones

```
case 'eliminar_conexion':
    $id = $_POST['id'] ?? 0;
    if ($id) {
        $stmt = $pdo->prepare("DELETE FROM conexiones WHERE id = ?");
        $stmt->execute([$id]);
        echo json_encode(['success' => true]);
    } else {
        echo json_encode(['success' => false]);
    }
    break;
```

Nota. El código gestiona la eliminación de enlaces entre dispositivos mediante el identificador único de la conexión. Implementa una validación previa del ID y ejecuta la sentencia SQL de borrado, notificando al administrador el éxito de la operación o la inexistencia del registro.

Figura 42

Código del RouterOS API

```
import sys
import json
import routeros_api

def get_sorted_queues(api):
    """
    Obtiene y ordena las colas del árbol de forma jerárquica (padre seguido de hijos).
    """
    resource_queue_tree = api.get_resource('/queue/tree')
    queues = resource_queue_tree.get()

    # 1. Crear un mapa de IDs a objetos de cola para referencia rápida.
    queue_map = {q['id']: q for q in queues}

    # 2. Clasificar colas en base a su padre:
    parent_queues_list = []
    children_map = {}

    for q in queues:
        parent_id = q.get('parent')

        # Un padre "lógico" es aquel cuyo padre es otra Queue Tree (su ID existe en queue_map)
        is_logical_child = parent_id and parent_id in queue_map

        if is_logical_child:
            if parent_id not in children_map:
                children_map[parent_id] = []
            children_map[parent_id].append(q)
        else:
            # Es un elemento de nivel superior (padre de WinBox, con parent=interface o no parent)
            parent_queues_list.append(q)
```

Nota El script automatiza la obtención y organización de las colas de tráfico desde el servidor. Implementa una estructura de mapeo para transformar la lista plana de recursos del RouterOS en un árbol jerárquico, facilitando la administración de prioridades y límites de ancho de banda de forma organizada.

Ejecución de comandos y automatización. En lugar de depender de extensiones nativas de PHP, el servidor web invoca el intérprete de Python mediante funciones de ejecución de sistema (`shell_exec`), pasando las credenciales de forma segura. Esto facilita tareas complejas como.

Gestión jerárquica. El uso de Python permite organizar y ordenar las colas de forma lógica (padres e hijos) antes de mostrarlas en la web.

Monitorización en tiempo real. Se procesan datos de tráfico (bytes por segundo) que Python extrae del router y entrega a la interfaz web en formato JSON para su visualización dinámica.

Protocolo ICMP (Ping). El protocolo ICMP (Protocolo de Mensajes de Control de Internet), comúnmente conocido como Ping, se utiliza en el módulo de monitoreo de redes para verificar la conectividad y el estado de los dispositivos en tiempo real.

Cómo se utiliza. El módulo de monitoreo, a través de su código PHP (`monitor.php`), ejecuta comandos a nivel de sistema operativo para enviar paquetes ICMP a las direcciones IP de los dispositivos de la red.

Verificación de estado. Se envía una solicitud de eco (ping) a la dirección IP de un dispositivo. Si el dispositivo está encendido y conectado a la red, responderá con un paquete de respuesta de eco. Esto le permite al sistema determinar si un equipo está en línea u fuera de línea.

Medición de latencia. El módulo mide el tiempo que tarda en recibir la respuesta del ping. Este valor, conocido como latencia, es un indicador clave del rendimiento de la conexión y la calidad de la red.

En resumen, el protocolo ICMP es la herramienta principal que permite al módulo de monitoreo obtener información en tiempo real sobre la disponibilidad y la calidad de la conexión de cada uno de los dispositivos en la red.

5.19 Portal de clientes

Diseñado para brindarle el control total de sus servicios de conectividad desde la comodidad de su hogar. A través de esta plataforma rápida y segura, usted podrá consultar en tiempo real el estado de su cuenta, descargar su facturación y realizar pagos en línea sin complicaciones. Además, ponemos a su disposición un canal directo de atención para resolver sus solicitudes y brindarle el soporte técnico oportuno que garantice su mejor experiencia digital.", como se observa en las figuras 43 al 44.

Figura 43

Interfaz web para la interacción con el usuario



Nota. Captura de pantalla de la página de inicio del proveedor de servicios de internet NetLine Colombia ISP, mostrando la sección de "Servicios Disponibles" para Internet Hogar e Internet Corporativo, con precios de planes residenciales, consulta y pago de facturas y atención.

Figura 44

Interfaz web para la consulta y pago de facturas

Nota. Captura de pantalla de la página de la interfaz de consulta y pago de las facturas pendientes, la cual remite a una pasarela de pagos online con el operador Wompi.

5.20 Evaluación de la Calidad del Software AdminISP

El proceso de validación y verificación del sistema AdminISP se estructuró en tres fases fundamentales, diseñadas para garantizar la estabilidad técnica, la correcta comunicación con el hardware de red y la idoneidad de la interfaz para los usuarios finales de NetLine Colombia ISP.

5.20.1 Pruebas funcionales (Enfoque de Caja Negra)

La primera fase del aseguramiento de calidad se centró en las pruebas funcionales bajo la modalidad de caja negra; que de acuerdo con Pressman (2010) y Sommerville (2011) este enfoque evalúa el sistema desde la perspectiva del usuario final, analizando exclusivamente las entradas y salidas para descubrir errores funcionales, de interfaz o de rendimiento, omitiendo intencionalmente la estructura lógica interna del código. Por su parte, Myers & Badgett (2011) refuerzan la efectividad de esta metodología al aportar técnicas rigurosas de ejecución, tales como la partición de equivalencia y el análisis de valores límite. Finalmente, desde una perspectiva de

estandarización contemporánea, Black (2012) complementa estas ideas afirmando que las pruebas de caja negra constituyen el paso estratégico preliminar para validar el cumplimiento de los requerimientos del negocio antes de proceder con evaluaciones estructurales más profundas. En este marco de referencia, el objetivo principal de esta etapa fue validar que las interacciones del usuario con la interfaz, tales como el uso de botones y el diligenciamiento de formularios, produjeran los resultados esperados sin evaluar el código interno.

Para comenzar, uno de los integrantes del equipo de desarrollo de software evaluó el Módulo de Clientes y Facturación. El procedimiento consistió en simular el alta de un nuevo suscriptor, introduciendo sus datos personales y de servicio. Tras guardar la información, se procedió a verificar la persistencia de los datos, comprobando que el registro se reflejara con total fidelidad en la base de datos y apareciera correctamente renderizado en la lista visual del panel de administración, este mismo proceso se lo realizó para llevar a cabo la facturación masiva como se observa en la figura 45 al 49.

Figura 45

Formulario de registro de un nuevo cliente en el sistema

The screenshot displays the 'Gestión de Usuarios y Clientes' interface. On the left is a sidebar with navigation links: 'Módulo de Usuarios' (highlighted), 'Módulo de Planes', 'Módulo de Facturación', 'Módulo de Recaudo', 'Módulo de Ingresos y Gastos', 'Módulo de Inventario', 'Módulo de Queues', and 'Módulo de Monitoreo'. The main content area is titled 'Gestión de Usuarios y Clientes' and includes a search bar 'Buscar por ID o Nombre' with 'Buscar' and 'Limpiar' buttons. Below this is the 'Agregar Nuevo Usuario' form. The form contains the following fields: 'Rol' (dropdown menu set to 'Cliente'), 'ID de Usuario' (text field with '1087959656'), 'Nombre' (text field with 'ANDRES BELALCAZAR'), 'Correo' (text field with 'andresb@netline.com'), 'Número de Celular' (text field with '3145654434'), and 'Dirección' (text field with 'Vereda el Rosario'). At the bottom of the form is the 'Planes Asignados' section, which includes a dropdown menu for 'Asignar Nuevo Plan' showing 'Plan Residencial 20Mbps (20M/20M - \$40.000)' and an 'Asignar' button. There are also 'Guardar Usuario' and 'Cancelar' buttons at the bottom of the form.

Nota. Interfaz del Módulo de Clientes utilizada para simular el alta de un nuevo suscriptor e ingresar sus datos personales y de servicio durante las pruebas funcionales.

Figura 46

Persistencia del registro del cliente en la base de datos

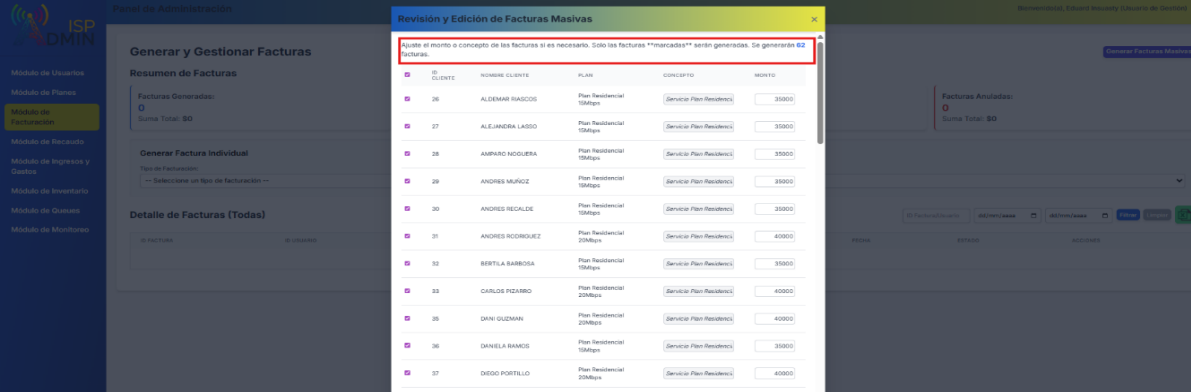


ID	NOMBRE CLIENTE	EMAIL	TELEFONO	DIRECCION	STATUS
75	OSCAR LOPEZ	usuario75@netline.com	315222272	yacuanquer	1
76	OSWALDO DELGADO	usuario76@netline.com	315222273	yacuanquer	1
77	PAOLA RIASCOS	usuario77@netline.com	315222274	yacuanquer	1
78	RAY MUNDO FUERTES	usuario78@netline.com	315222275	yacuanquer	1
79	ROSIO INSUASTY	usuario79@netline.com	315222276	yacuanquer	1
80	SEBASTIAN INSUASTY	usuario80@netline.com	315222277	yacuanquer	1
81	SEBASTIAN RIASCOS CORD	usuario81@netline.com	315222278	yacuanquer	1
82	VIVIANA PORTILLO	usuario82@netline.com	315222279	yacuanquer	1
83	WILLIAM ANDRADE	usuario84@netline.com	315222281	yacuanquer	1
84	LUIS RODRIGUEZ	usuario85@netline.com	3158584434	Mohechiza	1
85	DANIELA APRAEZ	usuario86@netline.com	314666756	San Felipe	1
86	LEIDY INSUASTY CAEZ	usuario87@netline.com	3177765337	El Rosario	1
87	CARMEN INSUASTY MH	usuario88@netline.com	3125555555	Mohechiza	1
123456	Eduard Insuasty	administrador@netline.com	3156513179	Yacuanquer - Nariño	2
27548656	Yanira Insuasty	yanira@netline.com			2
1087959629	DANIEL CHAVES	daniel@netline.com	3138556249	San Felipe	1
1087959656	ANDRES BELALCAZAR	andresb@netline.com	3145654434	Vereda el Rosario	1

Nota. Evidencia de la correcta inserción y almacenamiento de los datos del nuevo suscriptor en la base de datos tras enviar el formulario.

Figura 47

Interfaz de Revisión y Edición de Facturación Masiva.

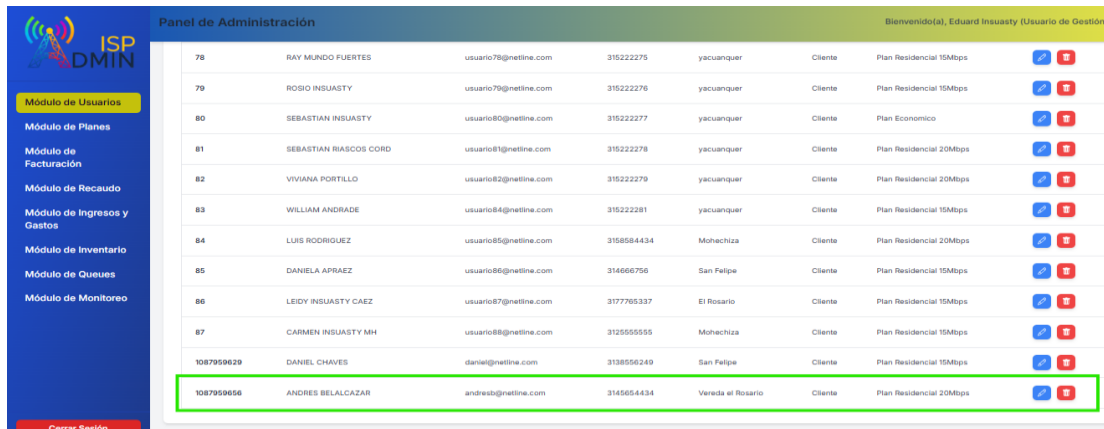


ID CLIENTE	NOMBRE CLIENTE	PLAN	CONCEPTO	MONTOS
26	ALEJANDRA RIASCOS	Plan Residencial 10Mbps	Service Plan Residencial	35000
27	ALEJANDRA LASSO	Plan Residencial 10Mbps	Service Plan Residencial	35000
28	AMBARO RODRIGUEZ	Plan Residencial 10Mbps	Service Plan Residencial	35000
29	ANDRES MUÑOZ	Plan Residencial 10Mbps	Service Plan Residencial	35000
30	ANDRES RECALDE	Plan Residencial 10Mbps	Service Plan Residencial	35000
31	ANDRES RODRIGUEZ	Plan Residencial 10Mbps	Service Plan Residencial	40000
32	BERTHA SARRIOSA	Plan Residencial 10Mbps	Service Plan Residencial	35000
33	CARLOS PIZABO	Plan Residencial 20Mbps	Service Plan Residencial	40000
35	DANI GUZMAN	Plan Residencial 20Mbps	Service Plan Residencial	40000
36	DANIELA RAMOS	Plan Residencial 10Mbps	Service Plan Residencial	35000
37	DIEGO PORTILLO	Plan Residencial 20Mbps	Service Plan Residencial	40000

Nota. Previsualización antes de la emisión de comprobantes. El sistema permite al administrador verificar el listado de clientes seleccionados, sus planes y montos correspondientes, ofreciendo la posibilidad de ajustar conceptos o valores de manera individual antes de ejecutar la generación masiva de facturas mediante el botón de confirmación

Figura 48

Persistencia de los datos en la interfaz de usuario



Panel de Administración

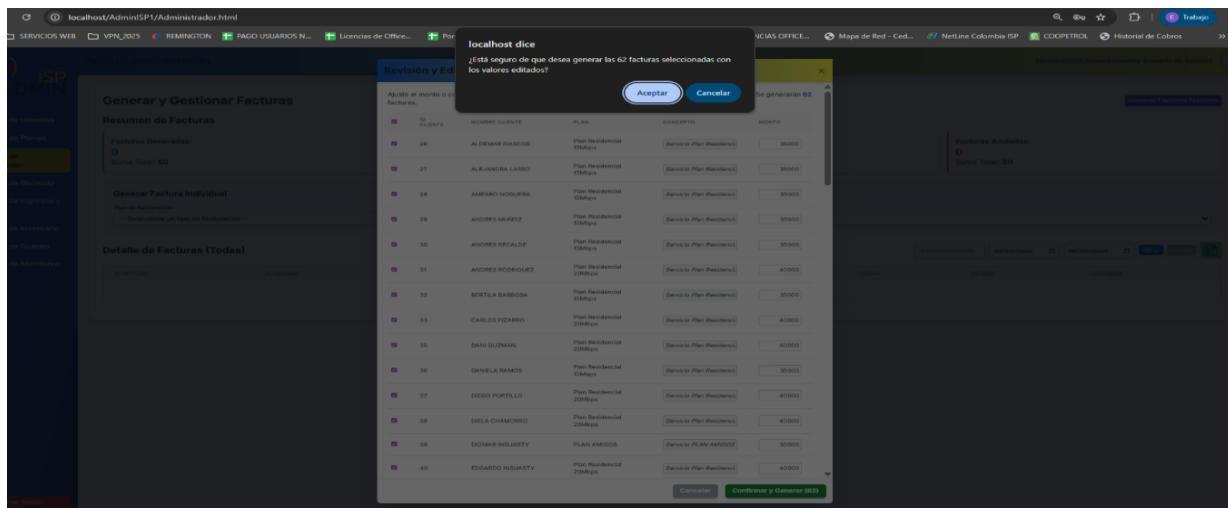
Bienvenido(a), Eduard Insuasty (Usuario de Gestión)

ID	NOMBRE	EMAIL	CELULAR	USUARIO	ROL	PLAN	ACCIONES
78	RAY MUNDO FUERTES	usuario78@netline.com	31522275	yacuanquer	Ciente	Plan Residencial 15Mbps	[Iconos]
79	ROSO INSUASTY	usuario79@netline.com	31522276	yacuanquer	Ciente	Plan Residencial 15Mbps	[Iconos]
80	SEBASTIAN INSUASTY	usuario80@netline.com	31522277	yacuanquer	Ciente	Plan Residencial 15Mbps	[Iconos]
81	SEBASTIAN RASCOS CORD	usuario81@netline.com	31522278	yacuanquer	Ciente	Plan Residencial 20Mbps	[Iconos]
82	VIVIANA PORTILLO	usuario82@netline.com	31522279	yacuanquer	Ciente	Plan Residencial 20Mbps	[Iconos]
83	WILLIAM ANDRADE	usuario83@netline.com	31522281	yacuanquer	Ciente	Plan Residencial 15Mbps	[Iconos]
84	LUIS RODRIGUEZ	usuario84@netline.com	3158584434	Mohachiza	Ciente	Plan Residencial 20Mbps	[Iconos]
85	DANIELA APRAEZ	usuario85@netline.com	314666756	San Felipe	Ciente	Plan Residencial 15Mbps	[Iconos]
86	LEIDY INSUASTY CAEZ	usuario86@netline.com	3177765337	El Rosario	Ciente	Plan Residencial 15Mbps	[Iconos]
87	CARMEN INSUASTY MH	usuario87@netline.com	3125555555	Mohachiza	Ciente	Plan Residencial 15Mbps	[Iconos]
1087950629	DANIEL CHAVES	daniel@netline.com	3138556240	San Felipe	Ciente	Plan Residencial 15Mbps	[Iconos]
1087950656	ANDRES BELALCAZAR	andres@netline.com	3145654434	Varela el Rosario	Ciente	Plan Residencial 20Mbps	[Iconos]

Nota. Verificación de la actualización de la interfaz en el panel de administración, confirmando que la información guardada se refleja correctamente para el usuario final.

Figura 49

Mensaje de confirmación.



Nota. Verificación de la emisión de la facturación masiva, confirmando que la facturación emitida se refleja correctamente para el usuario final y los administradores de la aplicación.

5.20.2 Gestión de inventario.

Finalmente, en lo que respecta a la Gestión de Inventario, el equipo de pruebas ejecutó una simulación del ciclo de vida de un equipo de telecomunicaciones para garantizar la integridad transaccional entre los activos físicos y su asignación lógica. El procedimiento inició con el registro de una nueva unidad en el sistema, específicamente una antena, detallando sus especificaciones técnicas y asegurando su disponibilidad inicial en la bodega central.

Una vez el activo estuvo disponible, se procedió a su asignación mediante el Módulo de Clientes, vinculando el equipo al perfil de un suscriptor activo. Esta acción permitió validar que el sistema disparara correctamente la transición de estado del hardware de "disponible" a "asignado", asegurando la trazabilidad del movimiento.

La prueba se consideró exitosa al auditar el panel de inventario y constatar que el contador de existencias en bodega disminuyó automáticamente en una unidad. Esta verificación, confirmó que la base de datos mantiene una sincronización exacta entre los módulos de inventario y servicios, previniendo discrepancias entre el stock reportado y los activos entregados al usuario final., como se observa en las figuras 50 a 52.

Figura 50

Formulario de registro y alta de nueva antena en el inventario

Adicionar Activo al Inventario

Tipo de Activo: Cantidad:

Inventario Disponible para Asignar

NOMBRE DEL ACTIVO	CANTIDAD TOTAL	CANTIDAD DISPONIBLE	ACCIONES
LiteBeam M5-23	55	48	Asignar
Ponchadora RJ45	2	1	Asignar
Repetidora LiteGPS 90	1	0	Asignar
RJ45	100	3	Asignar
Route Wifi	20	23	Asignar
Router Mikrotik	4	1	Asignar

Activos Asignados

Buscar por Nombre, IP, Serial...

ID ASIG.	ACTIVO	CANTIDAD	ASIGNADO A	FECHA	IP	SERIAL/MAC	NOTAS	ACCIONES
20	LiteBeam M5-23	1	BERTILA BARBOSA	2026-02-18	192.168.4.88	N/A	N/A	Asignar
21	LiteBeam M5-23	1	DIEGO PORTELLO	2026-02-18	192.168.6.129	N/A	N/A	Asignar

Nota. Interfaz del Módulo de Inventario utilizada para dar de alta un nuevo activo en la bodega central, estableciendo el stock inicial.

Figura 51

Asignación de activo de telecomunicaciones al perfil del cliente

Asignar: LiteBeam M5-23

Tipo de Asignación:

Cantidad a Entregar:

Asignar a Cliente:

Dirección IP (Opcional):

Número de Serie o MAC (Opcional):

Notas (Opcional):

localhost dice
Activo(s) asignado(s) exitosamente.

Nota. Evidencia del proceso de vinculación de la antena al suscriptor, validando la comunicación Inter modular entre Inventario y Clientes.

Figura 52

Actualización automática del stock en bodega tras la asignación y la persistencia en la base de

Inventario Disponible para Asignar

NOMBRE DEL ACTIVO	CANTIDAD TOTAL	CANTIDAD DISPONIBLE	ACCIONES
LiteBeam M5-23	55	47	Asignar
Ponchadora RJ45	2	1	Asignar
Repetidora LiteGPS 90	1	0	Asignar
RJ45	100	3	Asignar
Route Wifi	30	23	Asignar
Router Mikrotik	4	1	Asignar

```
SELECT * FROM `tipos_activos`
```

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

☐ Mostrar todo | Número de filas: 25 Filtrar filas: Ordenar según la

Opciones extra

				id	nombre	cantidad_total	cantidad_disponible
☐	Editar	Copiar	Borrar	1	LiteBeam M5-23	55	47
☐	Editar	Copiar	Borrar	2	Route Wifi	30	23
☐	Editar	Copiar	Borrar	3	Repetidora LiteGPS 90	1	0
☐	Editar	Copiar	Borrar	4	Router Mikrotik	4	1
☐	Editar	Copiar	Borrar	5	Ponchadora RJ45	2	1
☐	Editar	Copiar	Borrar	9	RJ45	100	3

Nota. Verificación de la integridad transaccional; el sistema refleja automáticamente la baja en bodega, garantizando que el stock visual coincida con la persistencia en la base de datos.

5.20.3 Pruebas de integración (Comunicación MikroTik API).

La segunda fase abordó el componente más crítico de la infraestructura tecnológica: las pruebas de integración. Estas evaluaciones fueron diseñadas para certificar que la comunicación en el backend (desarrollado en lenguajes como PHP y Python) y los routers de borde operara de manera fluida y en tiempo real.

El primer hito evaluado fue la Sincronización de Queues (Colas de ancho de banda). Se configuró desde el software AdminISP un plan de navegación de 100MB y se le asignó a un cliente de prueba. Para validar la integración, el equipo técnico accedió directamente al equipo MikroTik utilizando la herramienta Winbox, confirmando que la Simple Queue correspondiente se hubiese aprovisionado automáticamente y respetara de manera estricta los límites de velocidad establecidos en el software. Seguidamente, se puso a prueba la Suspensión de Servicio. Al oprimir el botón de "Suspender" en el perfil de un usuario moroso dentro de AdminISP, se verificó que el sistema enviara la instrucción al router y que la navegación del cliente quedara bloqueada de forma inmediata, garantizando un control de acceso eficiente. Como se observa en las figuras 53 y 54.

Figura 53

Interfaz para la creación, edición de queues

The screenshot displays the WinBox interface for a MikroTik Router. At the top, there are status bars for 'CARGA TOTAL (UP)' at 3.48 Mbps and 'DESCARGA TOTAL (DOWN)' at 53.77 Mbps. A central confirmation dialog box with a green checkmark states '¡Operación Exitosa!' and 'Operación procesada satisfactoriamente.' Below this, the 'Añadir Nueva Queue' form is visible, containing the following fields:

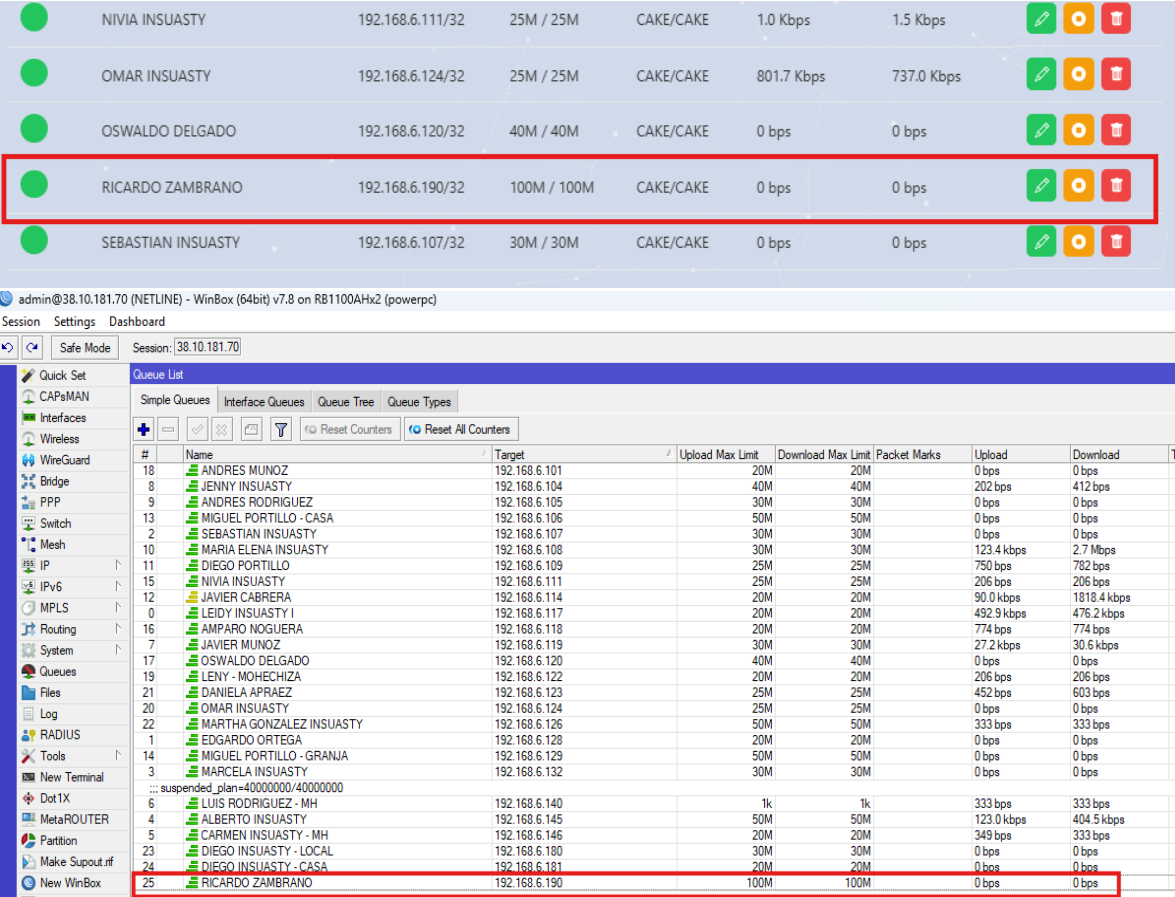
Nombre:	Target (IP):	Límite de Subida:	Límite de Bajada:	Prioridad:	Queue Type:
RICARDO ZAMBRANO	192.168.6.190/32	100M	100M	1/1	CAKE

Buttons for 'Añadir' (Add) and 'Cancelar' (Cancel) are located at the bottom of the form.

Nota. Verificación de registro exitoso de una queue, constatando el éxito de la operación y la comunicación con el router board mikrotik.

Figura 54

tabla de queues en el software y en el aplicativo winbox



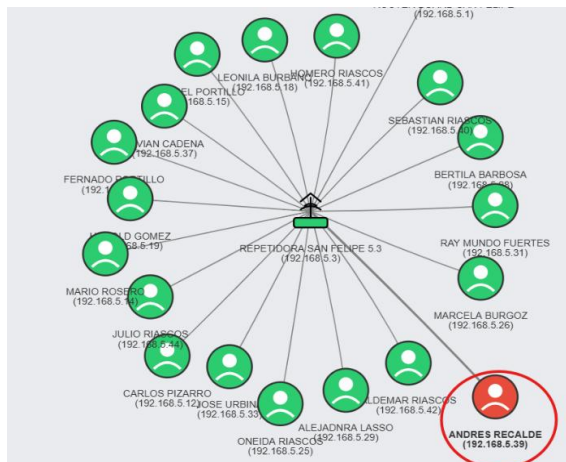
#	Name	Target	Upload Max Limit	Download Max Limit	Packet Marks	Upload	Download	To
18	ANDRES MUNOZ	192.168.6.101	20M	20M	0 bps	0 bps		
8	JENNY INSUASTY	192.168.6.104	40M	40M	202 bps	412 bps		
9	ANDRES RODRIGUEZ	192.168.6.105	30M	30M	0 bps	0 bps		
13	MIGUEL PORTILLO - CASA	192.168.6.106	50M	50M	0 bps	0 bps		
2	SEBASTIAN INSUASTY	192.168.6.107	30M	30M	0 bps	0 bps		
10	MARIA ELENA INSUASTY	192.168.6.108	30M	30M	123.4 kbps	2.7 Mbps		
11	DIEGO PORTILLO	192.168.6.109	25M	25M	750 bps	782 bps		
15	NIVIA INSUASTY	192.168.6.111	25M	25M	206 bps	206 bps		
12	JAVIER CABRERA	192.168.6.114	20M	20M	90.0 kbps	1818.4 kbps		
0	LEIDY INSUASTY I	192.168.6.117	20M	20M	492.9 kbps	476.2 kbps		
16	AMPARO NOGUERA	192.168.6.118	20M	20M	774 bps	774 bps		
7	JAVIER MUNOZ	192.168.6.119	30M	30M	27.2 kbps	30.6 kbps		
17	OSWALDO DELGADO	192.168.6.120	40M	40M	0 bps	0 bps		
19	LENY - MOHECHIZA	192.168.6.122	20M	20M	206 bps	206 bps		
21	DANIELA APRAEZ	192.168.6.123	25M	25M	452 bps	603 bps		
20	OMAR INSUASTY	192.168.6.124	25M	25M	0 bps	0 bps		
22	MARTHA GONZALEZ INSUASTY	192.168.6.126	50M	50M	333 bps	333 bps		
1	EDGARDO ORTEGA	192.168.6.128	20M	20M	0 bps	0 bps		
14	MIGUEL PORTILLO - GRANJA	192.168.6.129	50M	50M	0 bps	0 bps		
3	MARCELA INSUASTY	192.168.6.132	30M	30M	0 bps	0 bps		
...	suspended_plan=40000000/40000000							
6	LUIS RODRIGUEZ - MH	192.168.6.140	1k	1k	333 bps	333 bps		
4	ALBERTO INSUASTY	192.168.6.145	50M	50M	123.0 kbps	404.5 kbps		
5	CARMEN INSUASTY - MH	192.168.6.146	20M	20M	349 bps	333 bps		
23	DIEGO INSUASTY - LOCAL	192.168.6.180	30M	30M	0 bps	0 bps		
24	DIEGO INSUASTY - CASA	192.168.6.181	20M	20M	0 bps	0 bps		
25	RICARDO ZAMBRANO	192.168.6.190	100M	100M	0 bps	0 bps		

Nota. Tabla de queues en el aplicativo y herramienta de gestión de mikrotik (winbox), en la cual se evidencia el registro exitoso de la queues con los parámetros establecidos en el formulario de la figura 38.

Finalmente se evaluó el Monitoreo de Red mediante Ping. Se forzó la desconexión física de un dispositivo de prueba para simular una caída del servicio. Al observar el mapa topológico de red dentro de AdminISP, se constató que el sistema de monitoreo detectó la pérdida de paquetes, cambiando el indicador visual de estado de color verde a rojo en un lapso inferior al límite de tolerancia de 60 segundos. Como se observa en las figuras 55 y 56.

Figura 55

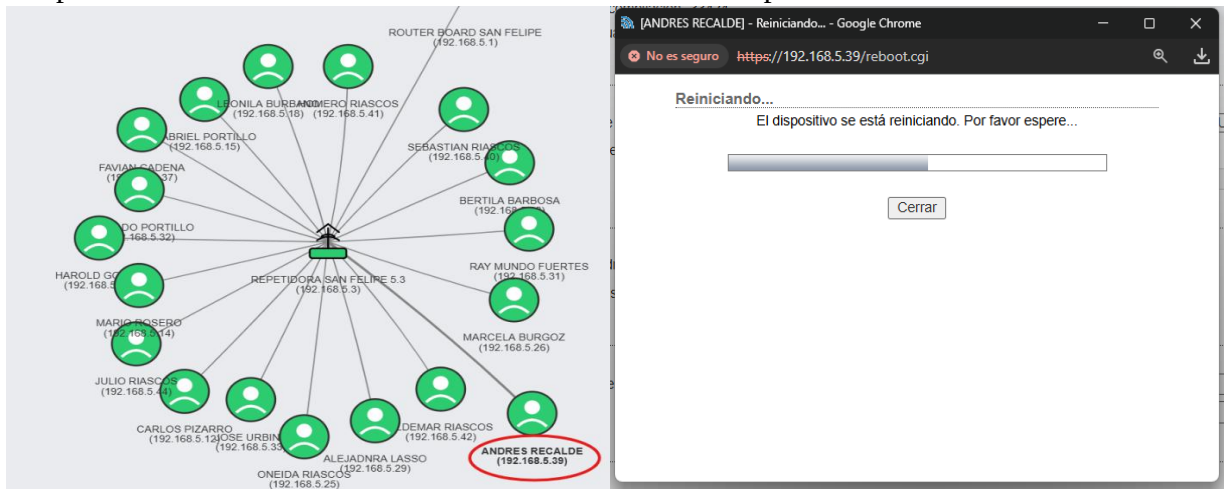
Mapa de red después del reinicio



Nota. Mapa de red en la que se visualiza el estado DOWN del cliente ANDRES RECALDE después de sufrir un reinicio evidenciando el cambio de color al perder paquetes al momento de lanzar el ping.

Figura 56

Mapa de red antes del reinicio e interfaz de reinicio del dispositivo.



Nota. Mapa de red en la que se visualiza el estado UP del cliente ANDRES RECALDE, y la interfaz de administración de la antena.

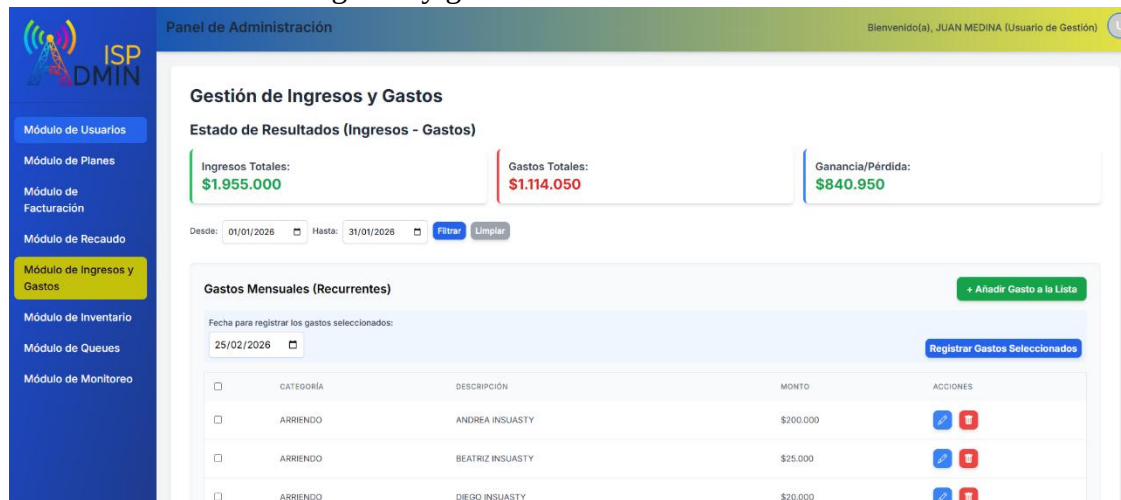
5.21 Pruebas de usabilidad

La fase final se orientó a cumplir con el objetivo específico de evaluar la operabilidad e involucración del usuario, apoyándose en los estándares de calidad de software de la norma ISO/IEC 25010 (2011). Esta etapa permitió medir la curva de aprendizaje y la intuición del diseño de la plataforma AdminISP. En este sentido, la evaluación trascendió la simple verificación técnica para enfocarse en la experiencia del usuario bajo condiciones de uso real. De acuerdo con los lineamientos de la norma ISO/IEC 25010 (2011) y las perspectivas de autores como Bevan et al. (2015) y Piattini y Rodríguez (2023), la verdadera calidad del software se manifiesta cuando el sistema permite una interacción tan natural que la carga cognitiva se reduce al mínimo, fomentando una involucración positiva del trabajador con la herramienta. Por ello, se diseñó una fase de experimentación controlada que permitiera observar de forma directa cómo los diferentes roles de la organización se apropiaban de las funciones del sistema sin mediación técnica previa.

Para garantizar resultados apegados a la realidad operativa de NetLine Colombia ISP, se seleccionó una muestra representativa de usuarios finales: dos personas del área administrativa y dos técnicos de campo. La metodología consistió en asignar a cada perfil una tarea fundamental de su día a día, con la directriz estricta de ejecutarla sin recibir asistencia ni manuales externos. Desde la perspectiva gerencial, se le solicitó al dueño de la empresa que utilizara el módulo financiero para consolidar y generar un reporte detallado de ingresos y gastos correspondientes al mes de enero de 2026, como se observa en las figuras 57 al 59.

Figura 57

interfaz del módulo de ingresos y gastos.

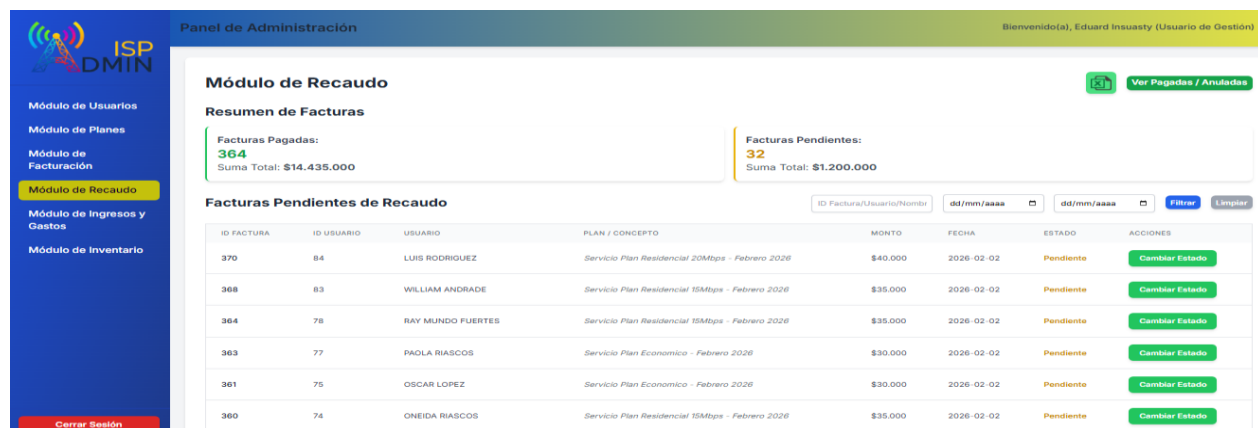


Nota. Se evidencia que se realizó el filtro de fecha desde el 01/01/2026 hasta 31/01/2026 en donde se detalla el total de los ingresos y gastos dentro de esa vigencia.

Por otra parte, en el área de cartera, se le asignó a la recaudadora Yanira Insuasty la tarea de procesar la recepción de un pago. Ella debió localizar la factura de un cliente, registrar el ingreso del dinero en el sistema y culminar el proceso imprimiendo el comprobante físico o digital de la transacción.

Figura 58

Interfaz del módulo de recaudo.



Nota. Interfaz recaudo y lista de tabla de facturas pendientes de pago.

Figura 59

Modal de cambio de estado y recibo de pago.

Cambiar Estado de Factura

Realizado por:
YANIRA INSUASTY

Estado Actual:
Pendiente

Nuevo Estado:
Pagada

Guardar Cancelar

localhost dice
Estado de la factura actualizado exitosamente.
Aceptar

Recibo Generado
El recibo de pago ha sido generado. ¿Qué deseas hacer?

Imprimir Descargar

RECIBO DE PAGO
NETLINE COLOMBIA ISP
Conectando Hogares

Recibo #: 244
Fecha de Pago: 24/2/2026

Datos del Cliente:
Nombre: WILLIAM ANDRADE (ID: 83)
Dirección: yacuquer
Correo: usuario84@netline.com
Celular: 315222281

Concepto	Monto
Servicio Plan Residencial 15Mbps - Febrero 2026	\$35.000

Total Pagado: \$35.000

Imprimir 1 página

Destino: Guardar como PDF

Páginas: Todas

Páginas por hoja: 1

Nota. Modal para el cambio de estado de pendiente a pagado/anulado, con la validación de éxito del proceso realizado y su respectivo comprobante de pago.

En el área técnica, las pruebas se dividieron según las especialidades. Al **Técnico 1**, quien es el encargado de administrar las políticas de tráfico (Queues), se le pidió localizar el perfil de un cliente específico y modificar su configuración de ancho de banda para realizar un aumento de velocidad, como se evidencia en las figuras 60 al 63

Figura 60

Interfaz de inicio de sesión a un mikrotik e interfaz de administración de las queues.

The image shows two screenshots of the Mikrotik Router web interface. The top screenshot is the login page, and the bottom screenshot is the queue management dashboard.

Top Screenshot: Login Page

Panel de Administración Bienvenido(a), Técnico Operativo Queues (Usuario de Gestión) U

Módulo de Queues

Mikrotik Router

INICIAR SESION

38.10.181.70

admin

.....

[Conectar](#) [Volver al Panel](#)

Bottom Screenshot: Queue Management Dashboard

CARGA TOTAL (UP) **13.68 Mbps**

DESCARGA TOTAL (DOWN) **45.64 Mbps**

Mikrotik Router

Conectado a: 38.10.181.70

[Simple Queues](#) [Rules](#) [Desconectar](#)

SIMPLE QUEUES - ACTUALES

Buscar: Nombre o Target... [+ Añadir Queue](#)

Estado	Nombre	Target	Max Limit	Queue Type	Subida Actual	Bajada Actual	Acciones
●	ALBERTO INSUASTY	192.168.6.145/32	50M / 50M	CAKE/CAKE	286.3 Kbps	3.57 Mbps	Editar Queue Eliminar Reiniciar
●	AMPARO NOGUERA	192.168.6.118/32	20M / 20M	CAKE/CAKE	24.0 Kbps	2.44 Mbps	Editar Queue Eliminar Reiniciar
●	ANDRES MUNOZ	192.168.6.101/32	20M / 20M	CAKE/CAKE	27.2 Kbps	1.74 Mbps	Editar Queue Eliminar Reiniciar

Nota. Se insertan los datos de ingreso al routerboard 38.10.181.70 para el iniciar la gestión. Evidenciando el éxito del login, se toma como muestra al usuario AMPARO NOGUERA para la respectiva prueba de edición.

Figura 61

Interfaz de administración de las queues.

The figure consists of two screenshots of the Mikrotik Router Queue Manager interface.

Top Screenshot: Editar Queue

The interface shows the Mikrotik Router logo and status information. The top bar displays "CARGA TOTAL (UP)" at 11.58 Mbps and "DESCARGA TOTAL (DOWN)" at 55.84 Mbps. The connection status is "Conectado a: 38.10.181.70". The "Editar Queue" form is visible, with fields for "Nombre" (AMPARO NOGUERA), "Target (IP)" (192.168.6.118/32), "Límite de Subida" (50M), "Límite de Bajada" (50M), "Prioridad" (8/8), and "Queue Type" (CAKE). A red box highlights the "Límite de Bajada" field.

Bottom Screenshot: SIMPLE QUEUES - ACTUALES

The interface shows the "SIMPLE QUEUES - ACTUALES" table. The top bar displays "CARGA TOTAL (UP)" at 3.72 Mbps and "DESCARGA TOTAL (DOWN)" at 32.60 Mbps. The connection status is "Conectado a: 38.10.181.70". The table lists the queue configuration for "AMPARO NOGUERA". A red box highlights the "Max Limit" column, which shows "50M / 50M".

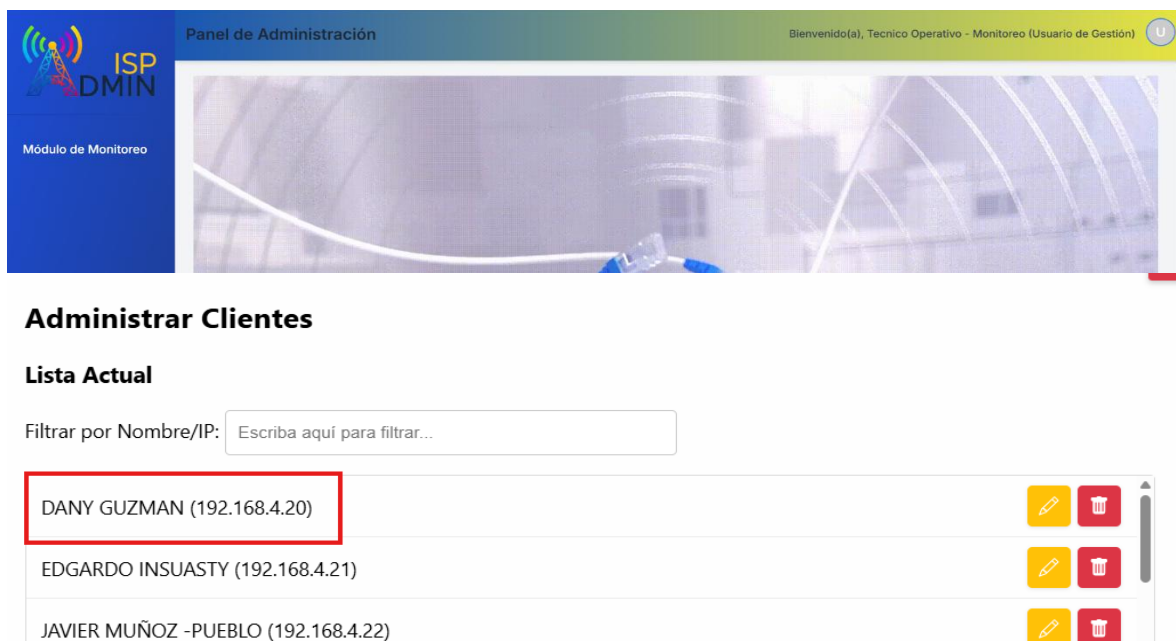
Estado	Nombre	Target	Max Limit	Queue Type	Subida Actual	Bajada Actual	Acciones
●	AMPARO NOGUERA	192.168.6.118/32	50M / 50M	CAKE/CAKE	268.8 Kbps	10.62 Mbps	✎ ⏸ 🗑

Nota. Se realiza el cambio de plan a 50M y se evidencia la persistencia del cambio en la interfaz del usuario seleccionado.

Finalmente, al Técnico 2, responsable del sistema de monitoreo, se le indicó que el cliente Dani Guzmán requería un cambio lógico en su infraestructura. Su tarea consistió en ingresar al sistema, localizar la antena asignada a dicho cliente y actualizar su dirección IP a la nueva asignación 192.168.4.35.

Figura 62

Módulo de monitoreo e interfaz para la edición de dispositivos.



ISP ADMIN
Módulo de Monitoreo

Panel de Administración
Bienvenido(a), Técnico Operativo - Monitoreo (Usuario de Gestión)

Administrar Clientes

Lista Actual

Filtrar por Nombre/IP:

DANY GUZMAN (192.168.4.20)	 
EDGARDO INSUASTY (192.168.4.21)	 
JAVIER MUÑOZ -PUEBLO (192.168.4.22)	 

Nota. Se realiza el cambio de la IP 192.168.4.35 al cliente DANI GUZMAN,

Figura 63

Formulario para la edición de los datos y mapa de red.

Editando Dispositivo

Nombre:

Dirección IP:

Dirección:

Observaciones:

Nota. Se realiza el cambio de la IP 192.168.4.35 al cliente DANI GUZMAN y se evidencia el cambio en el mapa de red.

Como parte de la fase de verificación y validación del sistema AdminISP, se llevó a cabo una evaluación de calidad centrada en la característica de Usabilidad, definida por la norma internacional ISO/IEC 25010 (2011). El objetivo fue medir el grado en que el software puede ser utilizado por perfiles específicos para alcanzar sus objetivos con eficacia, eficiencia y satisfacción.

Para cuantificar esta métrica, se aplicó la metodología System Usability Scale (SUS). Este instrumento estandarizado consta de un cuestionario de 10 ítems (5 positivos y 5 negativos). La evaluación se dividió en dos grupos de estudio principales: el personal operativo de NetLine Colombia ISP (Administradores y Técnicos) y una muestra de los Clientes finales.

5.21.1 Evaluación de Usabilidad: Perfil Administrador y Técnico.

Se diseñó un cuestionario SUS (System Usability Scale) que, de acuerdo con Brooke (1996), constituye una herramienta estandarizada, rápida y confiable para la evaluación subjetiva

de la usabilidad. Diversos investigadores, como Bangor et al. (2008) y Sauro (2011), han utilizado y validado empíricamente la efectividad de este método para medir la operabilidad en sistemas de software, demostrando su alta fiabilidad para diagnosticar barreras de interacción en perfiles técnicos y corporativos. En el marco de estas ideas planteadas, el cuestionario (Anexo 3) se adaptó al flujo de trabajo interno de NetLine, evaluando factores como la integración de funciones, la curva de aprendizaje y la complejidad del sistema en tareas diarias. La prueba fue aplicada a 4 miembros del personal estratégico de la empresa. Los resultados obtenidos tras la tabulación de las respuestas —multiplicando el total por 2.5 para obtener la escala estandarizada sobre 100, tal como lo establece la metodología original de Brooke (1996), como se muestran en la tabla 1.

Tabla 1

Resultados SUS - Personal Administrativo y Técnico

ADMINISTRADOR/TÉCNICO	R/IMPARES	R/PARES	TOTAL	2.5
ANDY STEVEN ROJAS	19	19	38	95
LEONARDO JIMENEZ	20	17	37	92.5
JUAN ALEXANDER	20	20	40	100
YANIRA INSUASTY	20	19	39	97.5
			154	385
PROMEDIO DE LA SUMATORIA			96.25	
CALIFICACION GRADO			A (EXCELENTE)	

Nota. Esta tabla nos muestra el promedio de la sumatoria general para el personal de administración y técnicos se situó en 96.25 sobre 100. Bajo la escala de calificación del modelo SUS, cualquier puntaje superior a 80.3 se clasifica en la categoría "A (Excelente)". Esto demuestra que el sistema presenta una curva de aprendizaje mínima y garantiza una alta operabilidad,

validando que la integración de la API de MikroTik y los módulos de facturación no generan una sobrecarga cognitiva en los empleados.

5.21.2 Evaluación de usabilidad: perfil cliente (Portal de Autogestión).

Para evaluar la interfaz pública, se adaptó el cuestionario SUS (Anexo 4) enfocándolo en la claridad de los servicios, el proceso de pago mediante PSE y la navegación general de la página web (consulta de facturas y contacto).

La prueba se aplicó a una muestra de 10 usuarios finales de NetLine Colombia ISP, obteniendo los resultados que se muestran en la tabla 2.

Tabla 2

Resultados SUS - Clientes

USUARIO	R/ IMPARES	R/ PARES	TOTAL	2,5
JAVIER MUÑOZ	20	19	39	97,5
MIGUEL PORTILLO	19	18	37	92,5
JAVIER CABRERA	18	18	36	90,0
SEBASTIAN RIASCOS	20	20	40	100,0
ANDRES RODRIGUEZ	16	15	31	77,5
CARMEN INSUASTY	20	18	38	95,0
ANDRES RECALDE	18	19	37	92,5
DIELA CHAMORRO	19	18	37	92,5
WILLIAM ANDRADE	19	18	37	92,5
JUAN CARLOS GONZALEZ	20	20	40	100,0
			372	930

PROMEDIO DE LA SUMATORIA

93,00

CALIFICACION GRADO:

A (EXCELENTE)

Nota. Esta tabla evidencia que el promedio consolidado para la muestra de clientes finales se situó en 93.00 sobre 100. Según la escala de calificación del modelo SUS, este puntaje ubica al portal de autogestión en la categoría "A (Excelente)". Esto certifica que la plataforma web pública cumple con altos estándares de intuición e involucración del usuario, demostrando que los clientes pueden consultar sus estados de cuenta, procesar pagos en línea y navegar por los servicios ofrecidos con gran facilidad. En términos operativos, este resultado garantiza una excelente autonomía por parte del cliente, reduciendo significativamente la carga de soporte técnico y administrativo para la empresa.

5.21.3 Análisis de resultados (Grupo Clientes).

El promedio consolidado para los clientes finales fue de 93.00 sobre 100, lo cual también posiciona al portal de autogestión en la categoría "A (Excelente)". A excepción de un caso aislado (Andrés Rodríguez, con 77.5, puntaje que sigue siendo considerado aceptable o grado 'B'), el 90% de la muestra superó los 90 puntos. Esto certifica ante la norma ISO/IEC 25010 (2011) que la plataforma pública cumple con altos estándares de intuición, generando confianza en los procesos de recaudo digital y mitigando la dependencia del soporte telefónico.

5.21.4 Cuadro Comparativo de Errores Encontrados vs. Corregidos (Testan)

Como parte del ciclo de Aseguramiento de Calidad (QA) y la aplicación de prácticas de Programación Extrema (XP), el software AdminISP fue sometido a rigurosas pruebas de estrés, validación de caja negra y pruebas de integración de red.

Durante las iteraciones de desarrollo, se identificaron y documentaron diversas vulnerabilidades y fallos lógicos. A continuación, se presenta el cuadro de control de incidencias, detallando los errores críticos encontrados y las soluciones técnicas implementadas para garantizar la estabilidad final del sistema entregado a NetLine Colombia ISP como se muestra en la tabla 3.

Tabla 3

Control de Incidencias y Corrección de Errores

ID	MODULO AFECTADO	DESCRIPCION DEL ERROR ENCONTRADO	NIVEL DE SEVERIDAD
ERR-01	Facturación Masiva	Duplicidad de cobros: Si el administrador hacía doble clic rápidamente en el botón "Generar Facturas", el sistema creaba la facturación dos veces para el mismo mes.	ALTA
ERR-02	Seguridad y Acceso	Vulnerabilidad a Inyección SQL (SQLi): Al ingresar comillas simples o la sentencia ' OR 1=1 en el buscador de usuarios, el sistema arrojaba errores de sintaxis de la base de datos o exponía registros no autorizados.	CRITICA
ERR-03	Integración MikroTik	Fallo al crear Queues con caracteres especiales: Al intentar asignar un plan a un cliente cuyo nombre contenía tildes o espacios dobles, el script de Python se bloqueaba y no enviaba el comando a la API de RouterOS.	MEDIA
ERR-04	Monitoreo (Ping)	Colapso por Timeout: Si un equipo de borde se desconectaba por completo (corte de energía), el script de ping se quedaba esperando respuesta indefinidamente, congelando la	ALTA

renderización del mapa de red en el frontend.

ERR-05	Inventario	Stock negativo: Al asignar manualmente equipos a los técnicos o clientes, el sistema permitía ingresar cantidades superiores a las disponibles en bodega, dejando el contador en números negativos.	MEDIA
--------	------------	---	-------

Nota. La presente tabla detalla el registro de incidencias (bug tracking) identificadas durante la ejecución de las pruebas funcionales (caja negra) y de integración del sistema AdminISP.

En esta tabla se exponen cinco vulnerabilidades y fallos lógicos encontrados en los módulos operativos, clasificados según su nivel de severidad (Crítica, Alta y Media). Entre los hallazgos destacan riesgos vitales para la seguridad y estabilidad de la plataforma, tales como la vulnerabilidad a inyección SQL, la duplicidad en el procesamiento de facturación masiva y el colapso del mapa de red por timeout. La detección detallada de estos errores evidencia el rigor del control de calidad aplicado bajo la metodología ágil (XP), lo cual fue fundamental para aplicar las correcciones técnicas pertinentes antes de autorizar el despliegue del software en el entorno de producción de NetLine Colombia ISP.

5.22 Pruebas de Seguridad y Acceso.

En esta segunda fase, el enfoque de calidad se desplazó hacia la robustez del sistema y la protección de la integridad de los datos, pilares fundamentales en la ingeniería de seguridad de la información, tal como lo establecen Stallings y Brown (2018). Dada la arquitectura cliente-

servidor del proyecto, se ejecutaron pruebas específicas para validar los mecanismos de defensa ante accesos no autorizados, fundamentando estas evaluaciones empíricas en los modelos de Control de Acceso Basado en Roles (RBAC) propuestos por Sandhu et al. (1996). Asimismo, en concordancia con los estándares internacionales de evaluación de vulnerabilidades web descritos por la OWASP Foundation (2021), se diseñaron escenarios de estrés para identificar y mitigar posibles brechas críticas, garantizando la resistencia del software frente a ataques por inyección de código (SQLi).

5.22.1 Paso 1. Validación de control de roles.

El procedimiento consistió en verificar la eficacia de la Matriz de Permisos, un componente crítico para garantizar la integridad de los procesos corporativos. Se realizó una prueba de acceso cruzado fundamentada en el modelo de Control de Acceso Basado en Roles (RBAC), el cual, según autores como Sandhu et al. (1996) y Ferraiolo et al. (2001), representa el estándar de la industria para asegurar que los usuarios interactúen únicamente con los recursos autorizados según su perfil. En esta evaluación, un usuario autenticado con el rol de "Técnico" intentó ingresar de forma directa al Módulo de Finanzas; este tipo de pruebas de cumplimiento de políticas de seguridad, recomendadas por Sommerville (2011), permitió validar que el sistema

restringe el acceso exclusivamente a perfiles administrativos, protegiendo así la información financiera de la organización como se evidencia en la figura 64.

Figura 64

Bloqueo de acceso por rol insuficiente en el Módulo de Finanzas.



Nota. Interfaz de seguridad que evidencia la restricción de navegación para usuarios con nivel de acceso "Técnico", redirigiendo o notificando la falta de privilegios.

5.22.2 paso 2. Pruebas de resistencia a inyección SQL (SQLi).

Posteriormente, se evaluó la vulnerabilidad de los puntos de entrada de datos en el Módulo de Búsqueda de Usuarios. El objetivo fue intentar manipular la consulta dinámica a la base de datos MySQL mediante la inserción de sentencias lógicas maliciosas como se muestra en la figura 65. Este procedimiento se alinea con las directrices de ciberseguridad establecidas por la OWASP Foundation (2021), la cual clasifica la inyección de código como una de las amenazas más críticas y recurrentes en las aplicaciones web. Asimismo, la metodología de prueba empleada para evaluar la robustez del motor relacional responde a los principios de seguridad e integridad de bases de datos descritos por Silberschatz et al. (2020), y a los modelos de clasificación de vulnerabilidades

documentados por Halfond et al. (2006). Esta evaluación garantiza que el sistema posea los mecanismos de filtrado y parametrización necesarios para neutralizar cualquier alteración no autorizada en la estructura de las consultas SQL. Como se evidencia en la figura 65

Figura 65

Módulo de Búsqueda de Usuarios.

The screenshot displays the 'Gestión de Usuarios y Clientes' module. At the top, there's a search bar with the text 'OR 1=1' entered. To the right of the search bar are buttons for 'Buscar' and 'Limpiar', and a green button labeled '+ Agregar Usuario'. Below the search bar, there are two sections: 'Usuarios de Gestión' and 'Clientes'. Each section has a table header with columns: ID, NOMBRE, CORREO, CELULAR, DIRECCIÓN, ROL, and ACCIONES. Below each table header, a message states 'No hay usuarios de gestión registrados que coincidan con la búsqueda.' and 'No hay clientes registrados que coincidan con la búsqueda.' respectively. On the left side, there's a sidebar with a menu containing items like 'Módulo de Usuarios', 'Módulo de Planes', 'Módulo de Facturación', 'Módulo de Recaudos', 'Módulo de Ingresos y Gastos', 'Módulo de Inventario', 'Módulo de Queues', and 'Módulo de Monitoreo'. The top of the page has a header with 'Panel de Administración' and a welcome message 'Bienvenido(a), Eduard Insuasty (Usuario de Gestión)'.

Nota. Al intentar vulnerar el formulario de búsqueda ingresando la sentencia relacional ' OR 1=1, el sistema no ejecuta la instrucción. Gracias al uso de sentencias preparadas (PDO) en el backend (PHP), la cadena es tratada como texto plano, arrojando cero coincidencias y protegiendo la integridad y confidencialidad de la base de datos.

Conclusiones

La culminación del proyecto de investigación permitió alcanzar con éxito el objetivo general de diseñar e implementar AdminISP como una solución tecnológica centralizada, transformando radicalmente la operatividad de la empresa NetLine Colombia ISP. A través del desarrollo de este software a la medida, se logró la transición efectiva de un modelo de gestión empírico y descentralizado —basado en el uso de herramientas manuales propensas al error— hacia una plataforma robusta e integrada. Esta transformación no solo erradicó ineficiencias críticas en los procesos de facturación e inventarios, sino que también elevó el estándar de profesionalización de la microempresa, dotándola de herramientas competitivas para el mercado de las telecomunicaciones.

Desde la perspectiva metodológica, la adopción del marco de trabajo ágil Programación Extrema (XP) y la técnica de Diseño Guiado por Pruebas (TDD) resultó ser una decisión estructural acertada. Este enfoque iterativo facilitó una supervisión de calidad constante, permitiendo la detección temprana y corrección de vulnerabilidades de seguridad críticas, como intentos de inyección SQL, y fallos lógicos graves, como la duplicidad en la facturación masiva. Asimismo, la arquitectura cliente-servidor, fundamentada en un ecosistema backend híbrido de PHP y Python, demostró una alta confiabilidad para procesar de forma simultánea la persistencia de datos en MySQL y la comunicación compleja con el hardware de red.

El mayor logro técnico e innovador del proyecto radicó en la integración bidireccional con la API de MikroTik RouterOS. Mediante la automatización del control de ancho de banda a través de Simple Queues, se eliminó la necesidad de intervención manual directa sobre los enrutadores de borde, mitigando riesgos por errores de configuración y protegiendo la estabilidad del servicio. Complementariamente, la implementación del módulo de monitoreo topológico en tiempo real,

apalancado en el protocolo ICMP, transformó la capacidad de respuesta de la empresa; al transitar de un soporte puramente reactivo a un modelo proactivo capaz de visualizar incidencias en un margen inferior a 60 segundos, se optimizó drásticamente la calidad del servicio percibida por los suscriptores.

Finalmente, la evaluación de calidad bajo la norma ISO/IEC 25010 (2011) confirmó que el sistema AdminISP es una herramienta altamente usable y eficiente. Los resultados obtenidos mediante el modelo System Usability Scale (SUS), con calificaciones de grado "A (Excelente)" tanto para el personal interno (96.25/100) como para los clientes finales (93.00/100), certifican que la interfaz es intuitiva y genera una baja carga cognitiva. Específicamente, el éxito del portal web de autogestión empoderó a los usuarios y demostró que es factible reducir los cuellos de botella administrativos mediante soluciones de comercio electrónico bien implementadas. En definitiva, AdminISP no solo resolvió el caos operativo actual, sino que sentó las bases tecnológicas para un modelo de negocio escalable y sostenible, demostrando que el software a la medida es el motor de desarrollo clave para los proveedores de internet en zonas rurales.

Referencias

- Bangor, A., Kortum, P. T., y Miller, J. T. (2008). An empirical evaluation of the System Usability Scale. *International Journal of Human-Computer Interaction*, 24(6), 574–594. <https://doi.org/10.1080/10447310802205776>
- Bevan, N., Carter, J., y Harker, S. (2015). The practical application of ISO 9241-11 and ISO/IEC 25010. En *Proceedings of the 17th International Conference on Human-Computer Interaction* (pp. 268–276). Springer. https://doi.org/10.1007/978-3-319-20901-2_25
- Black, R. (2012). *Foundations of software testing: ISTQB certification* (3.^a ed.). Cengage Learning.
- Brooke, J. (1996). SUS: A "quick and dirty" usability scale. En P. W. Jordan, B. Thomas, B. A. Weerdmeester, y A. L. McClelland (Eds.), *Usability evaluation in industry* (pp. 189–194). Taylor & Francis.
- Creswell, J. W., y Creswell, J. D. (2018). *Research design: Qualitative, quantitative, and mixed methods approaches* (5.^a ed.). SAGE Publications.
- DeLone, W. H., y McLean, E. R. (2003). The DeLone and McLean model of information systems success: A ten-year update. *Journal of Management Information Systems*, 19(4), 9–30. <https://doi.org/10.1080/07421222.2003.11045748>
- Ferraiolo, D. F., Sandhu, R., Gavrila, S., Kuhn, D. R., y Chandramouli, R. (2001). Proposed NIST standard for role-based access control. *ACM Transactions on Information and System Security (TISSEC)*, 4(3), 224–274. <https://doi.org/10.1145/501978.501980>
- Gálvez Albarracín, E. J., Riascos Erazo, S. C., y Contreras Paz, F. (2014). Influencia de las

- tecnologías de la información y comunicación en el rendimiento de las micro, pequeñas y medianas empresas colombianas. *Estudios Gerenciales*, 30(133), 355–364. <https://doi.org/10.1016/j.estger.2014.06.006>
- Gobernación de Nariño. (2020). Plan de Desarrollo Departamental "Mi Nariño, en defensa de lo nuestro" 2020-2023. <https://xn--nario-rta.gov.co/>
- Halfond, W. G. J., Viegas, J., y Orso, A. (2006). A classification of SQL-injection attacks and countermeasures. En *Proceedings of the International Symposium on Secure Software Engineering (ISSSE 2006)*. IEEE.
- Hernández Sampieri, R., Fernández Collado, C., y Baptista Lucio, P. (2014). *Metodología de la investigación* (6.^a ed.). McGraw-Hill Interamericana.
- Hurtado de Barrera, J. (2012). *Metodología de la investigación: Guía para la comprensión holística de la ciencia* (4.^a ed.). Quirón Ediciones.
- ISO/IEC. (2011). *ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*. Organización Internacional de Normalización.
- Kurose, J. F., y Ross, K. W. (2017). *Redes de computadoras: Un enfoque descendente* (7.^a ed.). Pearson Educación.
- MikroTik. (2023). RouterOS Documentation. <https://help.mikrotik.com/docs/>
- Myers, G. J., Sandler, C., y Baggett, T. (2011). *The art of software testing* (3.^a ed.). John Wiley & Sons.
- OWASP Foundation. (2021). OWASP Top 10:2021. <https://owasp.org/Top10/>

- Piattini, M., y Rodríguez, M. (2023). Calidad de sistemas de información (4.^a ed.). Ra-Ma.
- Pressman, R. S. (2010). Ingeniería de software: Un enfoque práctico (7.^a ed.). McGraw-Hill InterAmerican.
- Sandhu, R. S., Coyne, E. J., Feinstein, H. L., y Youman, C. E. (1996). Role-based access control models. *IEEE Computer*, 29(2), 38–47. <https://doi.org/10.1109/2.485845>
- Sauro, J. (2011). A practical guide to the System Usability Scale: Background, benchmarks & best practices. Measuring Usability LLC.
- Silberschatz, A., Korth, H. F., y Sudarshan, S. (2020). Fundamentos de bases de datos (7.^a ed.). McGraw-Hill Interamericana.
- Sommerville, I. (2011). Ingeniería de software (9.^a ed.). Pearson Educación.
- Stallings, W., y Brown, L. (2018). Computer security: Principles and practice (4.^a ed.). Pearson.
- Tornatzky, L. G., y Fleischer, M. (1990). The processes of technological innovation. Lexington Books.

Anexos

Anexo 1

Cuestionario de Entrevista a Personal Técnico y Administrativo de Netline Colombia ISP

Gestión Administrativa y Facturación (Personal Administrativo)

1. ¿Cómo gestionan actualmente la información de los suscriptores y los planes que tienen contratados?

2. ¿Cómo se lleva a cabo el proceso de generación de facturas y el control de los pagos mensuales?

Operación de Red e Inventario (Personal Técnico)

3. ¿Cómo realizan el control de la velocidad (ancho de banda) y los cortes de servicio a los clientes morosos?

4. ¿Cómo monitorean el estado actual de la red y las antenas?

5. ¿Llevan algún registro estructurado de los equipos, antenas y routers de la empresa?

Experiencia del Cliente y Seguridad (Gerencia)

6. ¿Cómo interactúa el cliente con la empresa para consultar su deuda, reportar un fallo o realizar un pago?

7. ¿Qué medidas de seguridad y acceso esperan que tenga la nueva plataforma?

Anexo 2

Resultado del Cuestionario de Entrevista a Personal Técnico y Administrativo de Netline Colombia ISP

Gestión Administrativa y Facturación (Personal Administrativo)

Pregunta 1: ¿Cómo gestionan actualmente la información de los suscriptores y los planes que tienen contratados?

Respuesta: Todo se maneja de forma manual usando hojas de cálculo y documentos en papel. Es un proceso desorganizado que frecuentemente nos lleva a cometer errores en los cobros porque no tenemos la información centralizada ni vinculada de forma clara con las tarifas de cada plan.

Pregunta 2: ¿Cómo se lleva a cabo el proceso de generación de facturas y el control de los pagos mensuales?

Respuesta: Es un trabajo repetitivo y lento. Tenemos que generar cada cobro a mano y luego registrar si nos pagaron o no. Necesitamos una herramienta que genere las facturas de todos los clientes al mismo tiempo cada mes, que nos permita registrar pagos o conciliar pagos por internet, y que nos muestre un resumen gráfico del dinero que entra, lo que se debe y los gastos.

Operación de Red e Inventario (Personal Técnico).

Pregunta 3: ¿Cómo realizan el control de la velocidad (ancho de banda) y los cortes de servicio a los clientes morosos?

Respuesta: Tenemos que entrar manualmente a la configuración de cada equipo MikroTik para cambiar las reglas o suspender el servicio. Es tedioso y se presta para cometer errores de configuración. Nos gustaría que el sistema se comunicara solo con los routers y aplicara los cortes, reconexiones y límites de velocidad automáticamente.

Pregunta 4: ¿Cómo monitorean el estado actual de la red y las antenas?

Respuesta: Actualmente, nos damos cuenta de que un nodo principal se cayó porque los clientes empiezan a llamar a quejarse. Es una reacción tardía. Necesitamos ver el estado de los equipos en vivo, saber quién está conectado y recibir alertas si algo se desconecta.

Pregunta 5: ¿Llevan algún registro estructurado de los equipos, antenas y routers de la empresa?

Respuesta: No, el control de la bodega es muy impreciso. A veces vamos a hacer una instalación y nos damos cuenta de que no hay equipos. Queremos registrar qué entra, qué sale, a quién se le instaló cada equipo y que el sistema nos avise cuando nos estemos quedando sin stock.

Experiencia del Cliente y Seguridad (Gerencia)

Pregunta 6: ¿Cómo interactúa el cliente con la empresa para consultar su deuda, reportar un fallo o realizar un pago?

Respuesta: Los clientes dependen totalmente de nosotros; tienen que llamarnos o escribirnos. Queremos darles una página web donde ellos mismos puedan ver su saldo, pagar en línea y crear solicitudes de soporte, para así quitarnos esa carga operativa de encima.

Pregunta 7: ¿Qué medidas de seguridad y acceso esperan que tenga la nueva plataforma?

Respuesta: No todos los empleados deben ver todo. Necesitamos que existan perfiles: el técnico no debe ver la contabilidad, y el cajero no debe poder meterse a desconfigurar la red. Además, el sistema debe ser fácil de usar en celulares, estar siempre en línea y estar preparado para no ponerse lento cuando tengamos muchos más clientes en el futuro.

Anexo 3*Cuestionario SUS para AdminISP – Para Administrativos y Técnicos*

1. ¿Cree que el sistema es fácil de usar para sus tareas diarias?
2. ¿Encontró que el sistema es innecesariamente complejo?
3. ¿Considera que las diversas funciones del sistema están muy bien integradas?
4. ¿Siente que necesitó el apoyo de un técnico o administrativo para poder realizar la tarea?
5. ¿Imagina que la gente aprendería a usar este sistema muy rápidamente?
6. ¿Le pareció que el sistema es muy pesado o complejo de usar en el flujo de trabajo?
7. ¿Se sintió seguro y confiado al usar la aplicación?
8. ¿Sintió que la inducción suministrada fue insuficiente y requiero reinducción para operar la aplicación?
9. ¿El sistema es intuitivo y la inducción suministrada fue suficiente para operar sin problemas?
10. ¿El sistema le resultó tan confuso que necesitaría ayuda de otra persona para usarlo correctamente?

Anexo 4
Cuestionario SUS para AdminISP – Para Clientes

1. ¿Los servicios ofrecidos por NetLine Colombia ISP en su página web son claros y fáciles de comprender?
2. ¿Me resultó muy complicado y confuso intentar consultar el estado de mis facturas pendientes en la página web?
3. ¿Pude pagar mis facturas pendientes por medio del botón PSE con gran facilidad?
4. ¿El proceso de pago online me pareció inseguro o frustrante, por lo que preferiría evitar usarlo en el futuro?
5. ¿Pude identificar rápidamente los canales de contacto (teléfono, correo, dirección) en caso de presentar una falla técnica?
6. ¿Sentí que la información y los íconos en la sección "¿Por qué elegir NetLine?" son puro relleno y no me generaron confianza?
7. ¿Al hacer clic en el botón "¿VER PLANES", fui dirigido correctamente a la sección de precios sin ninguna confusión en mi navegación?
8. ¿Fue muy difícil, tedioso y me tomó mucho tiempo encontrar los precios de los planes de internet en la página?
9. ¿El apartado de "Contáctanos" me ayudó claramente a saber qué hacer o a dónde acudir al tener una duda?
10. ¿Sentí que la página web es muy compleja o desordenada, y que necesitaría ayuda de otra persona para poder usarla correctamente?