

Instituto tecnológico superior del occidente del estado de hidalgo

Ingeniería en Tecnologías de la Información y la Comunicación

JLabs RepairSuite

Nombres: Sebastian Hernandez Angeles

Jose Manuel Cortez Ceron

Asuzena Reyes García

Bryan Fuentes Perez

Matricula: 230110313

Materia: Programacion 2

Nombre del docente: Maria Guadalupe Tolentino Cruz

Fecha: 28/05/2025

Indice

Resumen Ejecutivo.....	3
Introduccion.....	3
Objetivo General.....	4
Objetivos Específicos.....	4
Metodología de Desarrollo	5
Metodología en Espiral	5
Introduccion.....	10
Descripción de la problemática y la organización	11
Requerimientos de la organización con respecto a la red	11
Topología Lógica y Física	12
Listado de protocolos y estándares empleados en el proyecto.....	14
Esquema de direccionamiento IP (IPv4 e IPv6) y justificación.....	14
Descripción de la implementación de seguridad en los equipos de red	16
Descripción de las configuraciones básicas en los equipos de red.....	17
Pruebas de conectividad de red.....	19
Equipos de red	20
Resultados	20
Trabajos Futuros	21
Conclusiones.....	21

Resumen Ejecutivo

Introducción

En el marco de la transformación digital y ante el crecimiento sostenido de JLaboratories —una empresa dedicada a la reparación y mantenimiento de equipos de cómputo— surge la necesidad de contar con una solución de software que permita gestionar de manera ágil y confiable toda la operación diaria. Hasta ahora, el registro de clientes, equipos y órdenes se llevaba en cuadernos físicos, lo cual ha generado pérdidas de información, dificultades para priorizar reparaciones urgentes y falta de trazabilidad en los tiempos de entrega y los cobros asociados.

El propósito de este proyecto es diseñar e implementar una aplicación de escritorio basada en Windows Forms y .NET, respaldada por una base de datos SQL Server 2022, que centralice la gestión de clientes, equipos, órdenes de servicio, reparaciones, notas de reparación. Con una arquitectura en capas (Data, Business, Presentation y Entity), el sistema ofrecerá interfaces intuitivas para registrar alta y edición de clientes, incorporación de nuevos equipos, generación de órdenes de servicio, registro de procesos de reparación y notificaciones en el sistema.

La relevancia de esta solución radica en la mejora inmediata de la eficiencia operativa de JLaboratories:

- Reducción de errores y pérdidas de datos al sustituir anotaciones manuales por registros electrónicos validados.
- Priorización automática de órdenes urgentes.
- Escalabilidad para soportar el crecimiento del negocio, gracias a una base de datos normalizada y una arquitectura modular que facilita futuras extensiones.

De este modo, el proyecto no solo atiende las necesidades actuales de control y seguimiento de JLaboratories, sino que sienta las bases para un crecimiento ordenado y una mejor experiencia tanto para los clientes como para el equipo de trabajo.

Objetivo General

Desarrollar un sistema de software de escritorio bajo una arquitectura cliente-servidor para JLaboratories que permita digitalizar y optimizar el proceso de registro de clientes y equipos, la gestión de reparaciones (urgentes y no urgentes), la generación de tickets de órdenes, y el control eficiente de fechas de entrega mediante una interfaz intuitiva y una base de datos.

Objetivos Específicos

1. Analizar y documentar los requerimientos del cliente mediante entrevistas para identificar las necesidades reales del negocio y los puntos críticos del proceso actual.
2. Diseñar una base de datos relacional optimizada que permita almacenar, consultar y gestionar de manera eficiente la información de clientes, equipos y reparaciones utilizando normalizaciones, índices etc.
3. Desarrollar una aplicación de escritorio en .NET con Windows Forms, que proporcione una interfaz intuitiva para el registro y seguimiento de los servicios prestados.
4. Implementar un sistema de alertas y notificaciones para reparaciones urgentes.
5. Desplegar la solución en una red cliente-servidor, asegurando la integridad y disponibilidad de la información.
6. Realizar pruebas funcionales y documentar el sistema, garantizando su calidad y facilidad de uso para futuras actualizaciones.

Metodología de Desarrollo

Metodología en Espiral

El modelo en espiral es un enfoque iterativo e incremental que combina las ventajas de los modelos de cascada y prototipado, poniendo especial énfasis en el análisis continuo de riesgos. Cada vuelta de la espiral consta de cuatro actividades principales y produce artefactos que alimentan la siguiente iteración. A continuación, describe cómo lo aplicaremos al proyecto de JLaboratories.

1. Primera iteración: Definición de objetivos y alcance

1. Planificación

- Objetivo: Comprender el negocio, recopilar requisitos fundamentales y validar la viabilidad inicial.
- Actividades:
 - Entrevista con el cliente (JLaboratories) para documentar procesos actuales.
 - Identificación de actores (Recepcionista, Técnico, Cliente).
 - Lista inicial de casos de uso (gestión de clientes, registro de equipos, creación de órdenes).
 - Entrega: Documento de requisitos de alto nivel y bocetos de pantallas principales.

2. Análisis de riesgos

- Objetivo: Detectar riesgos técnicos y organizativos que puedan comprometer el proyecto.
- Actividades:
 - Evaluar madurez de la infraestructura
 - Verificar compatibilidad de herramientas (.NET 8, Windows Forms, Visual Studio 2022).
 - Analizar complejidad de integraciones

3. Ingeniería (prototipado rápido)

- Objetivo: Validar las pantallas y flujos críticos con un prototipo básico.
- Actividades:
 - Crear maquetas de los formularios de login y dashboard.
 - Implementar un CRUD mínimo de “Clientes” usando datos en memoria.
 - Entrega: Prototipo ejecutable; feedback del cliente sobre usabilidad.

4. Evaluación del cliente

- Objetivo: Recoger feedback para ajustar requisitos y validar dirección.
- Actividades:
 - Demostración del prototipo.
 - Recoger comentarios sobre usabilidad y workflow.
 - Priorizar nuevos casos de uso o cambios.
 - Entrega: Acta de reunión con lista de ajustes y features para la siguiente iteración.

2. Segunda iteración: Diseño detallado y base de datos

1. Planificación

- Objetivo: Refinar requisitos y planear diseño de datos y arquitectura en capas.
- Actividades:
 - Definir entidades y relaciones (Users, Clients, Equipment...).
 - Especificar esquemas de tabla y restricciones (FK, índices).
 - Entrega: Diagrama ER y scripts de creación de BD.

2. Análisis de riesgos

- Objetivo: Asegurar robustez del modelo de datos y performance.
- Actividades:
 - Simular consultas de carga (órdenes urgentes con índice en IsUrgent).
 - Entrega: Informe de performance y ajustes de índices

3. Ingeniería

- Objetivo: Generar el esquema físico en SQL Server y la capa de infraestructura.
- Actividades:
 - Ejecutar scripts de creación de BD en entorno de pruebas.
 - Implementar JLabSdbContext y repositorios EF Core.
 - Entrega: Proyecto Infrastructure con conexión a BD y repositorios operativos.

4. Evaluación

- Objetivo: Verificar correctamente persistencia y recuperación de datos.
- Actividades:
 - Tests manuales de inserción/lectura/borrado de datos.
 - Ajustar tipos de datos y tamaños de columnas según resultados.
 - Entrega: Informe de pruebas de persistencia y esquema finalizado.

3. Tercera iteración: Lógica de negocio y servicios

1. Planificación

- Objetivo: Definir los casos de uso concretos y la capa de aplicación.
- Actividades:
 - Mapear cada caso de uso a un servicio application-layer (ClientService, OrderService...).
 - Documentar contratos (DTOs y métodos públicos).
 - Entrega: Lista de servicios y firma de sus métodos.

2. Análisis de riesgos

- Objetivo: Garantizar coherencia y testabilidad de la lógica de negocio.
- Actividades:
 - Revisar posibles duplicados de lógica (cálculo de recargos, descuentos).
 - Plan de pruebas unitarias para cada servicio.
 - Entrega: Plan de pruebas unitarias y cobertura mínima.

3. Ingeniería

- Objetivo: Implementar los servicios y validaciones de negocio.
- Actividades:
 - Escribir tests unitarios (xUnit/NUnit) para cada servicio.
 - Entrega: Proyecto Application con servicios testeados

4. Evaluación

- Objetivo: Validar lógicas de negocio contra casos reales.
- Actividades:
 - Ejecutar test suite y corregir fallos.
 - Revisar escenarios de borde (cliente sin dirección).
 - Entrega: Informe de resultados y bugs corregidos.

4. Cuarta iteración: Integración UI y despliegue

1. Planificación

- Objetivo: Plan de integración, despliegue y capacitación.
- Actividades:
 - Definir pasos de despliegue (instalador, configuración de cadena de conexión).
 - Preparar manual de usuario y guía rápida.
 - Entrega: Plan de despliegue y documentación de usuario.

2. Análisis de riesgos

- Objetivo: Minimizar downtime y errores en producción.
- Actividades:
 - Ensayar rollback de BD.
 - Verificar permisos de SQL Server y rutas de instalación.
 - Entrega: Procedimiento de recuperación ante fallo.

3. Ingeniería

- Objetivo: Empaquetar la solución final y preparar el instalador.
- Actividades:
 - Compilar Release, generar instalador (MSI/ClickOnce).
 - Configurar CI/CD si procede (Azure DevOps, GitHub Actions).
 - Entrega: Paquete de instalación y pipeline configurado.

4. Evaluación

- Objetivo: Hacer el despliegue piloto y recoger feedback.
- Actividades:
 - Instalar en un equipo de prueba en la empresa.
 - Capacitar a usuarios clave (repcionista, técnico).
 - Recoger incidencias de uso real.
 - Entrega: Listado de mejoras y plan para la versión 1.1.

Introduccion

En el marco de la transformación digital y ante el crecimiento sostenido de JLaboratories —una empresa dedicada a la reparación y mantenimiento de equipos de cómputo— surge la necesidad de contar con una solución de software que permita gestionar de manera ágil y confiable toda la operación diaria. Hasta ahora, el registro de clientes, equipos y órdenes se llevaba en cuadernos físicos, lo cual ha generado pérdidas de información, dificultades para priorizar reparaciones urgentes y falta de trazabilidad en los tiempos de entrega y los cobros asociados.

El propósito de este proyecto es diseñar e implementar una aplicación de escritorio basada en Windows Forms y .NET, respaldada por una base de datos SQL Server 2022, que centralice la gestión de clientes, equipos, órdenes de servicio, reparaciones, notas de reparación. Con una arquitectura en capas (Data, Business, Presentation y Entity), el sistema ofrecerá interfaces intuitivas para registrar alta y edición de clientes, incorporación de nuevos equipos, generación de órdenes de servicio, registro de procesos de reparación y notificaciones en el sistema.

La relevancia de esta solución radica en la mejora inmediata de la eficiencia operativa de JLaboratories:

- Reducción de errores y pérdidas de datos al sustituir anotaciones manuales por registros electrónicos validados.
- Priorización automática de órdenes urgentes.
- Escalabilidad para soportar el crecimiento del negocio, gracias a una base de datos normalizada y una arquitectura modular que facilita futuras extensiones.

De este modo, el proyecto no solo atiende las necesidades actuales de control y seguimiento de JLaboratories, sino que sienta las bases para un crecimiento ordenado y una mejor experiencia tanto para los clientes como para el equipo de trabajo.

Descripción de la problemática y la organización

Actualmente, JLaboratories no cuenta con una infraestructura de red formal que permita una gestión centralizada y eficiente de la información. Las estaciones de trabajo operan de manera aislada, sin un sistema de interconexión que facilite el acceso compartido a datos, lo que genera duplicidad de información, demoras en la atención al cliente y dificultades para el trabajo colaborativo entre técnicos.

La ausencia de una red local impide el uso de una base de datos común y limita la posibilidad de implementar soluciones digitales que requieran acceso concurrente, como el sistema de gestión propuesto. Además, no existen medidas de seguridad en la transmisión o almacenamiento de datos, lo que expone a la empresa a pérdidas de información por errores humanos, fallos técnicos o accesos no autorizados.

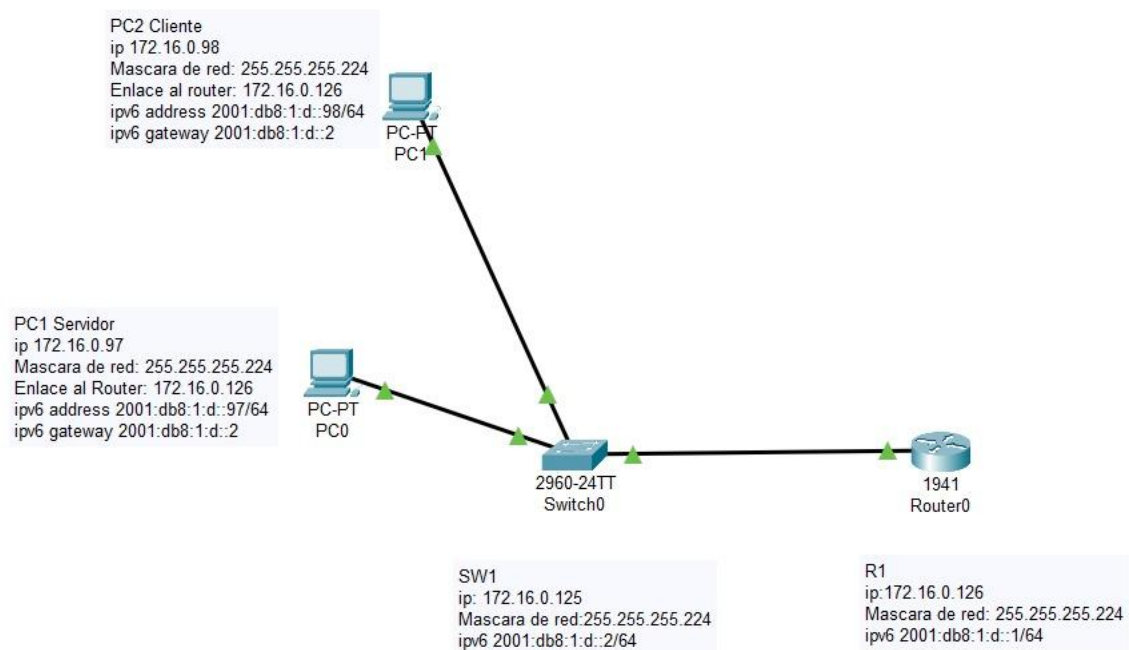
La situación actual refleja una necesidad urgente de establecer una red de computadoras estructurada que permita la centralización, seguridad y escalabilidad del sistema, como base tecnológica para soportar el crecimiento y la transformación digital de la organización.

Requerimientos de la organización con respecto a la red

<i>Requerimiento de red</i>	<i>Descripción</i>
<i>Ancho de banda</i>	Se requiere un ancho de banda mínimo de 100 Mbps para soportar la transferencia de datos entre estaciones de trabajo, acceso al servidor y futuras ampliaciones.
<i>Cantidad de usuarios</i>	La red debe admitir al menos 10 usuarios concurrentes en la fase inicial, con capacidad de escalar hasta 25 sin necesidad de rediseñar la infraestructura.
<i>Topología de red</i>	Se sugiere una topología en estrella, con un switch central que interconecte todas las estaciones de trabajo y el servidor principal.
<i>Tipo de red</i>	Red de área local (LAN) cableada, con posibilidad de agregar puntos de acceso inalámbrico (Wi-Fi) para dispositivos móviles o laptops de soporte técnico.
<i>Servidor central</i>	El servidor SQL Server 2022 deberá estar conectado por cable a la red, con acceso restringido mediante credenciales y perfiles de usuario.
<i>Seguridad de red</i>	Se implementarán firewalls, segmentación de red y autenticación mediante contraseñas seguras. Se contemplará el uso de antivirus centralizado.
<i>Acceso remoto</i>	Se prevé la futura incorporación de acceso remoto mediante VPN para permitir la gestión fuera del sitio, especialmente para técnicos en campo.
<i>Respaldo de datos</i>	Se requiere un sistema de respaldo automático en red, con copias diarias almacenadas localmente y semanalmente en la nube o medio externo seguro.

Disponibilidad	La red debe garantizar una disponibilidad mínima del 99%, asegurando operación continua durante el horario laboral sin interrupciones por fallos de red.
Escalabilidad	La red debe permitir la fácil integración de nuevos equipos, terminales o sucursales sin afectar el rendimiento ni requerir una reestructuración completa.

Topología Lógica y Física



Topología física

La red se basa en una **topología en estrella**, donde todos los dispositivos (PCs y servidor) están conectados a un switch central Cisco 2960-24TT, que a su vez está conectado a un router Cisco 1941. Esta disposición permite una estructura sencilla y escalable que centraliza el tráfico de red, facilitando la administración y el mantenimiento.

Dispositivos incluidos:

Dispositivo	Función	Dirección IPv4	Dirección IPv6
PC0 (Servidor)	Almacenamiento y servicios	172.16.0.97	2001:db8:1:d::97/64
PC1 (Cliente)	Estación de trabajo	172.16.0.98	2001:db8:1:d::98/64
Router0 (R1)	Enlace a red externa	172.16.0.126	2001:db8:1:d::1/64
SW1 (Interfaz)	Configuración lógica	172.16.0.125	2001:db8:1:d::2/64

Topología lógica

Desde el punto de vista lógico, la red pertenece al segmento **172.16.0.96/27**, con una máscara de subred **255.255.255.224**, lo que permite hasta 30 hosts útiles por subred. Se implementa también direccionamiento IPv6 para soporte moderno y escalabilidad futura.

Características lógicas clave:

- Subred: 172.16.0.96/27
- Gateway predeterminado: 172.16.0.126 (Router0)
- Prefijo IPv6: 2001:db8:1:d::/64
- Enlace predeterminado IPv6: 2001:db8:1:d::1 (Router0)

Justificación de la elección

La topología seleccionada para la red de JLaboratories es una topología en estrella, la cual fue elegida por su alta disponibilidad y facilidad de mantenimiento. Esta estructura garantiza que una falla en una estación de trabajo no afecte el funcionamiento del resto de la red, permitiendo así una operación continua y estable.

El uso de un switch centralizado facilita la administración del tráfico interno entre los dispositivos conectados, y permite una expansión futura del número de nodos sin necesidad de realizar un rediseño completo de la red. Esta característica es esencial considerando el crecimiento proyectado de la organización.

La incorporación de un router dedicado aporta escalabilidad, ya que permite extender la red hacia Internet o hacia futuras sucursales remotas. Además, este dispositivo puede gestionar políticas de seguridad y enrutar tráfico tanto en IPv4 como en IPv6, lo que incrementa la flexibilidad y el control de la red.

Finalmente, se ha optado por implementar compatibilidad con el protocolo IPv6, con el objetivo de preparar la infraestructura para el futuro. Esta decisión garantiza que la red será compatible con nuevos dispositivos y se ajustará a los estándares tecnológicos internacionales, asegurando así su vigencia a largo plazo.

Listado de protocolos y estándares empleados en el proyecto

Capa OSI	Protocolo / Estándar	Descripción
1. Física	Ethernet (IEEE 802.3)	Estándar de comunicaciones por cable de par trenzado (100BASE-TX) utilizado en la LAN.
2. Enlace	Ethernet MAC (IEEE 802.3)	Control de acceso al medio, direccionamiento MAC y detección de colisiones.
3. Red	IPv4 (RFC 791)	Protocolo de direccionamiento lógico y enrutamiento de paquetes.
	IPv6 (RFC 8200)	Versión de IP con direcciones de 128 bits, soportada en la LAN para futuro crecimiento.
	ICMP (RFC 792)	Protocolo de mensajes de control, usado para pruebas de ping y diagnóstico de red.
4. Transporte	TCP (RFC 793)	Protocolo de transporte fiable, garantiza entrega y orden de los datos (utilizado por SQL Server).
5. Sesión	TDS (Tabular Data Stream)	Protocolo de sesión de Microsoft SQL Server para intercambio de datos entre cliente y servidor.
6. Presentación	T-SQL (ANSI SQL)	Lenguaje de definición y manipulación de datos en SQL Server (procedimientos, vistas, triggers).
7. Aplicación	Microsoft.Data.SqlClient	Librería .NET que implementa TDS y gestiona la conexión y comandos hacia SQL Server desde C#.

Esquema de direccionamiento IP (IPv4 e IPv6) y justificación

Descripción general

Para el proyecto de red de JLaboratories se ha implementado un esquema de direccionamiento mixto que contempla tanto **IPv4** como **IPv6**, con el fin de garantizar compatibilidad con la infraestructura actual y al mismo tiempo preparar la red para futuras necesidades tecnológicas.

En cuanto a **IPv4**, se utilizó el bloque **172.16.0.96/27**, que proporciona 30 direcciones IP útiles, suficientes para el tamaño actual de la red. Este bloque pertenece a la clase B privada y es adecuado para redes empresariales pequeñas o medianas. Se utilizó una **máscara de subred 255.255.255.224**, que facilita el control y segmentación de los dispositivos.

En cuanto a **IPv6**, se empleó el prefijo **2001:db8:1:d::/64**, reservado para documentación, pero representativo de un esquema estructurado y escalable. Este direccionamiento permite una gran cantidad de dispositivos únicos y asegura compatibilidad con estándares modernos.

Tabla de distribución de direcciones IP

<i>Dispositivo</i>	<i>Interfaz</i>	<i>Dirección IPv4</i>	<i>Máscara de Subred</i>	<i>Gateway Predeterminado (IPv4)</i>	<i>Dirección IPv6</i>	<i>Gateway IPv6</i>
<i>PC0 (Servidor)</i>	FastEthernet0	172.16.0.9 7	255.255.255. 224	172.16.0.12 6	2001:db8:1:d::9 7/64	2001:db8:1:d::1
<i>PC1 (Cliente)</i>	FastEthernet0	172.16.0.9 8	255.255.255. 224	172.16.0.12 6	2001:db8:1:d::9 8/64	2001:db8:1:d::1
<i>SW1 (Switch L3)</i>	VLAN1	172.16.0.1 25	255.255.255. 224	172.16.0.12 6	2001:db8:1:d::2/ 64	2001:db8:1:d::1
<i>R1 (Router)</i>	G0/0	172.16.0.1 26	255.255.255. 224	-	2001:db8:1:d::1/ 64	-

Justificación de la elección

- **IPv4 (172.16.0.96/27):** Se seleccionó este rango privado por su adecuación a redes corporativas y por permitir control eficiente del direccionamiento con una cantidad razonable de hosts.
- **IPv6 (2001:db8:1:d::/64):** Este prefijo garantiza escalabilidad, soporte para autoconfiguración, mejor rendimiento en redes grandes y compatibilidad con tecnologías emergentes.
- **Segmentación lógica:** La máscara aplicada (en IPv4) limita el dominio de broadcast, aumentando la eficiencia de la red.
- **Uniformidad:** El uso del mismo gateway (IPv4 e IPv6) en todos los nodos garantiza facilidad de administración.

Descripción de la implementación de seguridad en los equipos de red

Para garantizar la integridad, confidencialidad y disponibilidad de los dispositivos de red en la infraestructura de JLaboratories, se han aplicado una serie de medidas de seguridad esenciales en los routers y switches de la red. Estas medidas permiten restringir el acceso no autorizado, proteger las credenciales y mantener una administración segura.

1. Sesiones seguras mediante SSH

Se ha habilitado el protocolo **SSH (Secure Shell)** en lugar de Telnet para el acceso remoto a los dispositivos. SSH cifra toda la información transmitida entre el administrador y el equipo de red, protegiendo las credenciales de autenticación y comandos sensibles frente a posibles interceptaciones. Además, se ha restringido el acceso VTY para aceptar exclusivamente conexiones SSH mediante el comando `transport input ssh`.

2. Configuración de contraseñas de acceso

Se han establecido contraseñas en distintos niveles de acceso:

- **Consola:** Para el acceso físico al equipo, mediante el comando `line console 0`.
- **VTY (líneas virtuales):** Para el acceso remoto, aplicando `line vty 0 4`.
- **Modo EXEC privilegiado:** A través de `enable secret`, que utiliza una forma encriptada de almacenamiento más segura que `enable password`.

3. Autenticación local

Se ha creado un usuario local con nombre de usuario y contraseña encriptada (`username admin secret admin123`), el cual es utilizado para la autenticación en sesiones SSH. Esto proporciona control de acceso más granular y seguro.

4. Desactivación de funciones innecesarias

Se ha desactivado la búsqueda de DNS (`no ip domain-lookup`) para evitar demoras innecesarias ante errores de comando, que podrían ser explotadas por actores maliciosos.

5. Mensajes de advertencia (banner MOTD)

Se ha configurado un **mensaje de advertencia legal (banner MOTD)** visible antes de acceder al equipo. Este mensaje notifica a los usuarios que el acceso está restringido, y que cualquier uso indebido será sancionado. Esto también tiene validez jurídica en caso de accesos no autorizados.

6. Cifrado de contraseñas

Se ha utilizado el comando `enable secret` y `username secret`, que almacenan contraseñas cifradas con algoritmos más robustos que los métodos tradicionales de texto plano (`service password-encryption` puede añadirse para mayor protección de contraseñas simples si se usan).

Descripción de las configuraciones básicas en los equipos de red

<i>Dispositivo</i>	<i>Comando</i>	<i>Comentario</i>
Switch SW1	hostname SW1	Cambia el nombre del switch para facilitar su identificación en la red.
Switch SW1	banner motd # *** ACCESO RESTRINGIDO *** #	Configura un mensaje que se muestra al ingresar al equipo, advertencia para usuarios no autorizados.
Switch SW1	no ip domain-lookup	Desactiva la búsqueda DNS para evitar retrasos cuando se escribe mal un comando.
Switch SW1	enable secret cisco123	Establece una contraseña encriptada para acceso al modo privilegiado, mejorando la seguridad.
Switch SW1	line console 0 password consola123 login	Configura la contraseña para acceso físico por consola y habilita la autenticación.
Switch SW1	line vty 0 4 password vty123 login	Establece la contraseña para acceso remoto por Telnet/SSH y habilita la autenticación.
Switch SW1	ip domain-name ejemplo.com	Define el nombre de dominio, necesario para la generación de claves RSA para SSH.
Switch SW1	username admin secret admin123	Crea un usuario con privilegios para autenticación local SSH, aumentando la seguridad remota.
Switch SW1	crypto key generate rsa 1024	Genera las claves RSA necesarias para cifrado SSH con un tamaño seguro de 1024 bits.
Switch SW1	line vty 0 4 transport input ssh login local	Restringe acceso remoto solo por SSH y usa la base de datos local para autenticación.
Switch SW1	sdm prefer dual-ipv4-and-ipv6 default	Cambia el modo SDM para soportar tanto IPv4 como IPv6 (necesita reinicio para aplicar).
Switch SW1	interface vlan 1 ip address 172.16.0.126 255.255.255.224	Asigna IP para gestión del switch en VLAN 1 para administración remota.
Switch SW1	ipv6 address 2001:db8:1:d::2/64 ipv6 address fe80::2 link-local	Configura direcciones IPv6 global y link-local para administración y compatibilidad IPv6.

Switch SW1	no shutdown	Activa la interfaz VLAN para que el switch pueda ser gestionado remotamente.
---------------	-------------	--

Dispositivo	Comando	Comentario
Router RD	hostname RD	Cambia el nombre del router para identificarlo fácilmente en la red.
Router RD	banner motd # *** ACCESO RESTRINGIDO *** #	Muestra un mensaje de bienvenida o advertencia a los usuarios que acceden al router.
Router RD	no ip domain-lookup	Desactiva la búsqueda DNS para evitar retrasos al ingresar comandos incorrectos.
Router RD	enable secret cisco123	Define una contraseña cifrada para el modo privilegiado, aumentando la seguridad del acceso.
Router RD	line console 0 password consola123 login	Establece la contraseña para acceso físico por consola y habilita la autenticación.
Router RD	line vty 0 4 password vty123 login	Define la contraseña para acceso remoto por Telnet o SSH y activa la autenticación.
Router RD	ip domain-name ejemplo.com	Configura el nombre de dominio necesario para generar las claves RSA para SSH.
Router RD	username admin secret admin123	Crea un usuario con contraseña cifrada para autenticación local en accesos remotos.
Router RD	crypto key generate rsa 1024	Genera claves RSA para cifrado SSH con tamaño recomendado de 1024 bits para mayor seguridad.
Router RD	line vty 0 4 transport input ssh login local	Restringe el acceso remoto solo a conexiones SSH y usa la base de datos local para autenticación.
Router RD	ipv6 unicast-routing	Habilita el enrutamiento de paquetes IPv6 en el router (necesario para redes IPv6).
Router RD	interface g0/1 description "to LAND" ip address 172.16.0.125 255.255.255.224	Configura la interfaz GigabitEthernet 0/1 con descripción y dirección IPv4 para comunicación en LAN.

<i>Router RD</i>	ipv6 address 2001:db8:1:d::1/64 ipv6 address fe80::1 link-local	Asigna direcciones IPv6 global y link-local a la interfaz para conectividad IPv6.
<i>Router RD</i>	no shutdown	Activa la interfaz para que esté operativa y pueda transmitir tráfico.

Pruebas de conectividad de red

<pre> C:\>ping 2001:db8:1:d::2 Pinging 2001:db8:1:d::2 with 32 bytes of data: Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Ping statistics for 2001:DB8:1:D:2: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms C:\>ping 2001:db8:1:d::1 Pinging 2001:db8:1:d::1 with 32 bytes of data: Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Ping statistics for 2001:DB8:1:D:1: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms </pre>		<pre> Cisco Packet Tracer PC Command Line 1.0 C:\>ping 172.16.0.98 Pinging 172.16.0.98 with 32 bytes of data: Reply from 172.16.0.98: bytes=32 time=11ms TTL=128 Reply from 172.16.0.98: bytes=32 time<1ms TTL=128 Reply from 172.16.0.98: bytes=32 time=1ms TTL=128 Reply from 172.16.0.98: bytes=32 time=6ms TTL=128 Ping statistics for 172.16.0.98: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 11ms, Average = 4ms C:\>ping 172.16.0.125 Pinging 172.16.0.125 with 32 bytes of data: Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Ping statistics for 172.16.0.125: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms C:\>ping 172.16.0.126 Pinging 172.16.0.126 with 32 bytes of data: Request timed out. Reply from 172.16.0.126: bytes=32 time<1ms TTL=255 Reply from 172.16.0.126: bytes=32 time<1ms TTL=255 Reply from 172.16.0.126: bytes=32 time<1ms TTL=255 Ping statistics for 172.16.0.126: Packets: Sent = 4, Received = 3, Lost = 1 (25% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 1ms, Average = 0ms </pre>
<pre> C:\>ping 2001:db8:1:d::2 Pinging 2001:db8:1:d::2 with 32 bytes of data: Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Ping statistics for 2001:DB8:1:D:2: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms </pre>		<pre> Cisco Packet Tracer PC Command Line 1.0 C:\>ping 172.16.0.97 Pinging 172.16.0.97 with 32 bytes of data: Reply from 172.16.0.97: bytes=32 time<1ms TTL=128 Reply from 172.16.0.97: bytes=32 time<1ms TTL=128 Reply from 172.16.0.97: bytes=32 time<1ms TTL=128 Reply from 172.16.0.97: bytes=32 time<1ms TTL=128 Ping statistics for 172.16.0.97: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms C:\>ping 172.16.0.125 Pinging 172.16.0.125 with 32 bytes of data: Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Reply from 172.16.0.125: bytes=32 time<1ms TTL=255 Ping statistics for 172.16.0.125: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms C:\>ping 172.16.0.126 Pinging 172.16.0.126 with 32 bytes of data: Reply from 172.16.0.126: bytes=32 time<1ms TTL=255 Reply from 172.16.0.126: bytes=32 time<1ms TTL=255 Reply from 172.16.0.126: bytes=32 time<1ms TTL=255 Reply from 172.16.0.126: bytes=32 time<1ms TTL=255 Ping statistics for 172.16.0.126: Packets: Sent = 4, Received = 3, Lost = 1 (25% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms </pre>
<pre> C:\>ping 2001:db8:1:d::2 Pinging 2001:db8:1:d::2 with 32 bytes of data: Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Ping statistics for 2001:DB8:1:D:2: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms </pre>		<pre> C:\>ping 2001:db8:1:d::2 Pinging 2001:db8:1:d::2 with 32 bytes of data: Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:2: bytes=32 time<1ms TTL=255 Ping statistics for 2001:DB8:1:D:2: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms C:\>ping 2001:db8:1:d::1 Pinging 2001:db8:1:d::1 with 32 bytes of data: Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Reply from 2001:DB8:1:D:1: bytes=32 time<1ms TTL=255 Ping statistics for 2001:DB8:1:D:1: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms </pre>

Equipos de red

Dispositivo	Marca	Modelo	Puertos / Interfaces	Características principales
HP ProBook 440 G8	HP	ProBook 440 G8	Ethernet 10/100/1000 Mbps; Wi-Fi 6 (802.11ax); Bluetooth 5.0	11th Gen Intel® Core™ i5-1135G7; 8 GB DDR4; 256 GB NVMe SSD; 14" FHD antirreflejo
HP EliteBook 440 G8	HP	EliteBook 440 G8	Ethernet 10/100/1000 Mbps; Wi-Fi 6 (802.11ax); Bluetooth 5.0	11th Gen Intel® Core™ i7-1165G7; 16 GB DDR4; 512 GB NVMe SSD; 14" FHD IPS (400 nits)
Switch SW1	Cisco	Catalyst 2960-X-24TS-L	24×1 Gbps RJ-45; 2×1 Gbps SFP	Switching 216 Gbps; Layer 2 gestionado (VLAN/QoS/STP); CLI/Web/SNMP
Router R1	Cisco	ISR 4321	2×Gigabit Ethernet WAN; 2×Gigabit Ethernet LAN	Throughput hasta 50 Mbps; VPN IPsec/DMVPN; IPv4/IPv6; Firewall; SSH/CLI

Resultados

- Despliegue completo cliente-servidor en LAN, con conexión remota a SQL Server por TCP/1433.
- Autenticación segura mediante usuario SQL y validación de credenciales en WinForms.
- Funcionalidad CRUD para Clientes, PCs y Órdenes, con listas, filtros y edición en tiempo real.
- Generación automática de tickets al crear órdenes y exportación a PDF desde la aplicación.
- Dashboard dinámico que muestra el número de órdenes Pending, Urgent y Completed, y tabla de órdenes recientes ordenadas cronológicamente.
- Procedimientos almacenados y triggers en la base de datos que encapsulan la lógica de inserción, actualización y cascada de fechas.
- Pruebas de rendimiento: uso de memoria RAM hasta 78 MB y pico de CPU en 27 %.

Trabajos Futuros

1. Migración a la nube (Azure/AWS) para alta disponibilidad y escalabilidad global.
2. Integración de autenticación con Active Directory o Azure AD.
3. Desarrollo de clientes web y mobile (Blazor, React Native).
4. Reportería avanzada: exportación a Excel, dashboards y gráficas de tendencias.
5. Cifrado de datos en reposo y en tránsito (TLS, columnas sensibles).
6. Módulo de notificaciones (email, WhatsApp vía API).

Conclusiones

- Se entregó una plataforma integral para la gestión de servicios de reparación, aplicando todas las fases del modelo en espiral y buenas prácticas de ingeniería de software.
- La arquitectura en capas y el uso de SPs y triggers garantizan mantenibilidad, seguridad y facilidad de evolución.
- El proyecto demostró la aplicación efectiva de competencias de bases de datos, programación, redes y arquitectura de computadores.
- JLaboratories está lista para ampliarse con nuevos módulos y desplegarse en entornos más grandes, ofreciendo robustez y flexibilidad.