

Documento de requerimientos de software

***DESARROLLO DE UNA APLICACIÓN WEB CON LA
METODOLOGÍA ÁGIL SCRUM PARA LA GESTIÓN DE
PROCESOS DE NEGOCIO DE LA ASOCIACIÓN SAN PEDRO DE
LICTO***

Fecha: [13/05/2025]

Tabla de contenido

Historial de Versiones.....	3
Información del Proyecto	3
Aprobaciones.....	3
1. Propósito	4
2. Alcance del producto / Software.....	4
3. Referencias	6
4. Funcionalidades del producto.....	6
5. Clases y características de usuarios.....	7
6. Entorno operativo.....	10
7. Requerimientos funcionales	10
9.1. (Nombre de la funcionalidad 1).....	¡Error! Marcador no definido.
9.2. (Nombre de la funcionalidad 2).....	¡Error! Marcador no definido.
9.3. (Nombre de la funcionalidad N)	¡Error! Marcador no definido.
8. Reglas de negocio	15
9. Requerimientos de interfaces externas.....	16
9.1. Interfaces de usuario	16
9.2. Interfaces de hardware	16
9.3. Interfaces de software.....	17
9.4. Interfaces de comunicación.....	17
10. Requerimientos no funcionales	18

11.	Otros requerimientos.....	19
12.	Glosario.....	19

Historial de Versiones

Fecha	Versión	Autor	Organización	Descripción
2024-04-15	1.0	Jennifer Yambay Lemache	ESPOCH	Versión Inicial del documento de Requerimientos
2024 - 05- 13	2.0	Jennifer Yambay Lemache	ESPOCH	Segunda versión del documento de Requerimientos

Información del Proyecto

Empresa / Organización	Asociación de Productores San Pedro de Licto
Proyecto	Desarrollo de una aplicación web con la metodología ágil SCRUM para optimizar los procesos de negocio de la Asociación de Productores San Pedro de Licto
Fecha de preparación	13/ 05 / 2025
Cliente	Asociación de Productores San Pedro De Licto
Patrocinador principal	Jennifer Yambay Lemache
Gerente / Líder de Proyecto	Jennifer Yambay Lemache
Gerente / Líder de Análisis de negocio y requerimientos	Jennifer Yambay Lemache

Aprobaciones

Nombre y Apellido	Cargo	Departamento u Organización	Fecha	Firma
-------------------	-------	-----------------------------	-------	-------

Jennifer Yambay Lemache	Proponente	ESPOCH	15/04/2025	
----------------------------	------------	--------	------------	--

1. Propósito

Este documento describe los requerimientos funcionales y no funcionales del sistema denominado “Sistema Web de Gestión Productiva San Pedro”, versión 2.0. El alcance de este documento cubre la totalidad del sistema, incluyendo los siguientes componentes: autenticación, gestión de entregas, pagos, gestión de usuarios, generación de reportes, predicción de producción y análisis de datos. Esta solución tiene como objetivo modernizar y optimizar los procesos operativos de la Asociación de Productores San Pedro de Licto.

2. Alcance del producto / Software

El presente proyecto tiene como propósito principal el desarrollo de una aplicación web orientada a automatizar la gestión diaria de entregas de leche, cálculo de pagos y generación de reportes para la Asociación de Productores. Este sistema se alinea con la necesidad de modernizar los procesos operativos y administrativos, brindando una herramienta tecnológica que mejore la eficiencia organizacional y respalde la toma de decisiones con datos confiables y actualizados.

Objetivo general del software

Automatizar y optimizar el control de entregas de leche, la liquidación de pagos a los socios productores, y la generación de reportes gerenciales y operativos, a través de una solución web accesible, segura y trazable.

Beneficios para el área de negocio y la organización

- Eficiencia operativa: Reducción del tiempo y esfuerzo en el procesamiento manual de datos y cálculos de pagos.

- Transparencia y confianza: Generación automática de reportes verificables que mejoran la confianza entre la directiva y los productores.
- Trazabilidad: Registro sistemático e histórico de entregas, pagos y estadísticas de producción.
- Toma de decisiones basada en datos: Acceso a indicadores clave y visualización de datos en tiempo real para respaldar decisiones estratégicas.

Objetivos y metas

Los objetivos funcionales del software están estrechamente vinculados con los objetivos estratégicos de la Asociación, entre los cuales se destacan:

- Modernización de procesos internos, mediante la digitalización de las actividades clave de recolección y pago.
- Incremento de la competitividad, facilitando un control más eficiente sobre la calidad y cantidad de leche entregada.
- Fortalecimiento de la gestión directiva, a través de reportes consolidados que faciliten auditorías internas y externas.

Estas metas se enmarcan dentro de la línea de investigación de Sistemas de Información, lo cual permite una integración entre el desarrollo tecnológico y los enfoques académicos y metodológicos establecidos.

Documentos relacionados

El desarrollo de este sistema está sustentado en los siguientes documentos y normativas:

- Anteproyecto del trabajo de grado, que define el problema, justificación, objetivos, metodología y cronograma de desarrollo.
- ISO/IEC 25010, estándar internacional utilizado como referencia para establecer criterios de calidad del software, asegurando

características como usabilidad, mantenibilidad, seguridad y eficiencia.

- Scrum Guide 2020, como marco de trabajo ágil para la gestión del desarrollo del proyecto, permitiendo entregas incrementales y la validación continua de avances con los usuarios finales.

Además, se contempla como documento complementario el acta de constitución del proyecto, en la que se establece formalmente el compromiso institucional con la iniciativa, los recursos disponibles y los responsables del seguimiento.

3. Referencias

- Romero Cangas, S. (2023). Desarrollo de aplicativo web para el manejo ganadero. ESPOCH.
- Ponte, R. (2024). Uso de Machine Learning en educación superior. Universidad César Vallejo.
- Julián, A. M. (2022). Análisis sistemático de la norma ISO 25010. ResearchGate.
- The Scrum Guide (2020). scrumguides.org.
- Bejarano & Rey (2022). Sitio Web: estructura y navegación.

4. Funcionalidades del producto

A continuación, se presenta la lista de funcionalidades principales que conforman el sistema de gestión para la Asociación de Productores, agrupadas según su propósito general dentro del flujo operativo:

- Módulo de autenticación de usuarios

Permite el inicio de sesión seguro mediante credenciales personalizadas, validando la identidad de los usuarios registrados en el sistema.

- Módulo de gestión de usuarios y roles

Gestión integral de usuarios, incluyendo la creación, edición y eliminación, así como la asignación de roles y permisos según el perfil (administrador, productor, operador, etc.).

- Registro de entregas de leche por productor

Registro diario de la cantidad de leche entregada por cada productor, con datos asociados como fecha, hora, calidad y observaciones.

- Gestión automática de pagos

Cálculo automático de pagos a productores según parámetros configurables (precio por litro, bonificaciones, penalizaciones), y generación de comprobantes o liquidaciones.

- Predicción de producción con algoritmos de Machine Learning

Implementación de modelos predictivos que analicen el historial de entregas para estimar la producción futura de leche por productor o por periodo.

- Generación de reportes y gráficos estadísticos

Creación automática de reportes mensuales, trimestrales o personalizados-.

- Panel administrativo con indicadores clave

Interfaz centralizada para administradores, que muestra indicadores clave de desempeño, alertas operativas y datos relevantes para la toma de decisiones.

- Recuperación de contraseña

Funcionalidad para que los usuarios puedan recuperar o restablecer su contraseña de manera segura mediante verificación por correo electrónico u otro mecanismo seguro.

5. Clases y características de usuarios

A continuación, se describen los diferentes tipos de usuarios que interactuarán con el sistema, clasificados según la frecuencia de uso, nivel de acceso, experiencia y funcionalidades del sistema a las que acceden. Esta segmentación permite definir adecuadamente los permisos, roles de seguridad y prioridades de diseño en la interfaz de usuario.

1. Productores

- Frecuencia de uso: Alta (uso diario o frecuente).
- Nivel de experiencia técnica: Bajo a medio. Se espera una interfaz intuitiva y amigable.
- Funcionalidades relevantes:
 - Registro de entregas de leche (Funcionalidad 3)
 - Consulta de pagos y comprobantes (Funcionalidad 4)
 - Recuperación de contraseña (Funcionalidad 8)
- Descripción: Son los principales usuarios operativos del sistema. Su objetivo es registrar las entregas de leche y verificar el estado de sus pagos. Su experiencia tecnológica puede variar, por lo que la interfaz para este perfil debe ser sencilla y de fácil navegación.

2. Administradores

- Frecuencia de uso: Alta (uso diario o frecuente).
- Nivel de experiencia técnica: Medio a alto.
- Funcionalidades relevantes:
 - Gestión de usuarios y roles (Funcionalidad 2)
 - Validación y supervisión de registros de entrega (Funcionalidad 3)
 - Gestión automática de pagos (Funcionalidad 4)
 - Generación de reportes (Funcionalidad 6)

- Panel administrativo (Funcionalidad 7)
- Recuperación de contraseña (Funcionalidad 8)
- Descripción: Encargados de supervisar y controlar el funcionamiento del sistema. Tienen privilegios amplios para acceder y modificar información crítica, administrar perfiles de usuario y revisar indicadores clave. Utilizan el sistema para tareas operativas y de gestión.

3. Analistas

- Frecuencia de uso: Moderada (uso periódico).
- Nivel de experiencia técnica: Alto. Requieren acceso a funciones avanzadas.
- Funcionalidades relevantes:
 - Acceso a reportes y análisis estadístico (Funcionalidad 6)
 - Uso del módulo de predicción con ML (Funcionalidad 5)
- Descripción: Profesionales encargados de realizar análisis de datos para identificar patrones de producción y generar proyecciones. Usan el sistema como herramienta de apoyo a la toma de decisiones estratégicas.

4. Visitantes

- Frecuencia de uso: Baja (uso ocasional).
- Nivel de experiencia técnica: Variable.
- Funcionalidades relevantes:
 - Acceso informativo a datos públicos o institucionales (funcionalidades básicas o módulo informativo fuera de autenticación)

- Descripción: Usuarios externos sin necesidad de autenticación. Su acceso está limitado a información pública que la asociación desee compartir, como noticias, visión institucional o informes generales.

6. Entorno operativo

- Hardware: Laptop HP Envy, Smartphone Redmi 13C, impresora EPSON L3560.
- Software: Windows 10, Django, React, PostgreSQL, Visual Studio Code, R Studio.
- Navegadores compatibles: Chrome, Firefox, Edge.

7. Requerimientos funcionales

Los siguientes requerimientos describen las funcionalidades específicas que el sistema deberá implementar. Cada requisito funcional está clasificado por módulo o categoría funcional.

- 7.1. Autenticación de usuarios

REQ-01: Ingreso con credenciales y validación

El sistema debe permitir a los usuarios ingresar mediante un formulario de autenticación que valide el nombre de usuario y la contraseña registrados en la base de datos. Si las credenciales son correctas, se permitirá el acceso al sistema según los privilegios correspondientes.

REQ-02: Gestión de sesiones según rol

El sistema debe gestionar las sesiones de los usuarios autenticados, estableciendo permisos y accesos específicos de acuerdo con su rol asignado (administrador, productor, analista). Cada tipo de usuario tendrá una interfaz personalizada con funcionalidades habilitadas o restringidas.

REQ-03: Recuperación de contraseña

El sistema debe ofrecer un mecanismo para que los usuarios puedan recuperar o restablecer su contraseña en caso de olvido, a través de un

enlace de recuperación enviado por correo electrónico u otro método de verificación seguro.

- 7.2. Registro de entregas

REQ-04: Registro diario por productor

El sistema debe permitir a los productores ingresar diariamente la cantidad de leche entregada, asociando la entrada con su usuario, fecha y hora. Este registro debe ser almacenado de forma estructurada para su posterior análisis.

REQ-05: Validación automática de cantidades

Durante el ingreso de datos, el sistema debe validar que la cantidad de leche registrada sea numérica, positiva y no exceda los límites establecidos por la asociación para una sola entrega. Se deben mostrar mensajes de error si se ingresan valores inválidos.

- 7.3. Gestión de pagos

REQ-06: Cálculo mensual automático

El sistema debe calcular automáticamente el pago mensual correspondiente a cada productor, en función del volumen entregado, precio por litro, bonificaciones o descuentos aplicables. Los resultados deben quedar almacenados y ser accesibles para los usuarios autorizados.

REQ-07: Histórico de pagos por usuario

Cada productor debe tener acceso a un historial detallado de sus pagos mensuales, con información sobre cantidades entregadas, precios aplicados, deducciones y pagos netos. El administrador podrá visualizar el historial de todos los usuarios.

- 7.4. Predicción de producción

REQ-08: Entrenamiento de modelo ML

El sistema debe integrar un módulo de aprendizaje automático (Machine Learning) que entrene modelos predictivos a partir del historial de

entregas. El entrenamiento debe ser actualizable periódicamente con nuevos datos recolectados.

REQ-09: Proyección de volumen mensual

Con base en el modelo entrenado, el sistema debe generar proyecciones de la producción mensual por productor o en conjunto. Estas predicciones deben ser accesibles para administradores y analistas a través de un módulo gráfico.

- 7.5. Reportes y estadísticas

REQ-10: Reportes PDF con filtros personalizados

El sistema debe generar reportes en formato PDF, configurables mediante filtros por fecha, productor, volumen entregado, entre otros criterios. Los reportes deben poder ser descargados y almacenados por el usuario.

REQ-11: Visualización gráfica de datos

El sistema debe permitir la visualización de datos a través de gráficos interactivos (líneas, barras, pastel, etc.) que representen indicadores clave como producción por día, evolución mensual, comparativas entre productores, entre otros.

Identificación	Sub identificación	Descripción del requisito	Versión	Estado actual	Última fecha estado registrado	Criterios de aceptación	Nivel de complejidad	Necesidad, oportunidades u objetivos de negocio	Objetivo del proyecto	Entregables (EDT)	Diseño del producto	Desarrollo del producto	Estrategia y escenarios de pruebas	Interesado (Stakeholder) dueño del requisito	Nivel de prioridad
----------------	--------------------	---------------------------	---------	---------------	--------------------------------	-------------------------	----------------------	-------------------------------------------------	-----------------------	-------------------	---------------------	-------------------------	------------------------------------	----------------------------------------------	--------------------

REQ-04	REQ-03	REQ-02	REQ-01
7.2	7.1	7.1	7.1
Registro diario por productor	Recuperación de contraseña	Gestión de sesiones según rol	Ingreso con credenciales y validación
1.0	1.0	1.0	1.0
Propuesto	Propuesto	Propuesto	Propuesto
Registro por fecha y usuario productor	Envío de enlace seguro para restablecer contraseña	Sesiones diferenciadas por rol (admin, productor)	Usuario puede iniciar sesión con credenciales válidas
Media	Media	Alta	Media
Control de entregas diarias	Mejora de usabilidad y continuidad del servicio	Seguridad y personalización de experiencia	Control de acceso seguro
Seguimiento de producción	Facilitar recuperación de acceso	Personalizar funcionalidades según roles	Asegurar acceso restringido y autenticado
Módulo de entregas	Función de recuperación	Sistema de roles	Módulo de login
Formulario de registro	Pantalla y lógica de recuperación	Lógica de sesión por rol	Pantalla de login
CRUD de entregas	API de recuperación de credenciales	Middleware de control de acceso	Backend de autenticación
Pruebas de inserción y edición de entregas	Flujo de recuperación exitoso y fallido	Pruebas por tipo de rol	Pruebas de login válido/inválido
Productores	Usuario final	Administrador de sistema	Equipo de desarrollo
Alta	Media	Alta	Alta

REQ-08	REQ-07	REQ-06	REQ-05
7.4	7.3	7.3	7.2
Entrenamiento de modelo ML	Histórico de pagos por usuario	Cálculo mensual automático	Validación automática de cantidades
1.0	1.0	1.0	1.0
Propuesto	Propuesto	Propuesto	Propuesto
Modelo entrenado con datos históricos	Visualización de pagos anteriores	Generación automática del monto total a pagar	Alerta si la cantidad ingresada no es válida
Alta	Media	Alta	Alta
Implementar IA para mejorar predicciones	Transparencia en pagos	Automatización de procesos	Reducción de errores manuales
Automatización de predicción de producción	Seguimiento de historial financiero	Eficientar cálculos contables	Validar datos ingresados automáticamente
Modelo ML	Historial de pagos	Módulo de pagos	Lógica de validación
Diseño de arquitectura ML	Interfaz de historial	Algoritmo de cálculo	Reglas de negocio
Entrenamiento con dataset	Consultas a base de datos	Script de procesamiento	Validaciones en backend
Validación cruzada y métricas de error	Pruebas de visualización y paginación	Pruebas de cálculo por mes y usuario	Casos con errores y excepciones
Data Scientist	Usuario productor	Contabilidad	Productores / QA
Alta	Media	Alta	Alta

REQ-11	REQ-10	REQ-09
7.5	7.5	7.4
Visualización gráfica de datos	Reportes PDF con filtros personalizados	Proyección de volumen mensual
1.0	1.0	1.0
Propuesto	Propuesto	Propuesto
Dashboard interactivo con gráficos	Generación de PDFs según filtros seleccionados	Predicción mostrada al usuario
Media	Alta	Alta
Mejora de análisis visual	Mejorar gestión documental	Planeación de la producción
Visualización de estadísticas clave	Facilitar exportación de reportes	Asistencia a la toma de decisiones
Dashboard	Reportes PDF	Módulo de predicción
Diseño visual con gráficos	Interfaz de filtros	Gráfica de predicción
Lógica de gráficos interactivos	Generador de PDF dinámico	Endpoint de proyección
Pruebas de carga y visualización dinámica	Pruebas con múltiples combinaciones de filtros	Comparación con datos reales
Gerencia / Producción	Gerencia	Administrador / Productores
Media	Alta	Alta

https://docs.google.com/spreadsheets/d/1_8FGV6z_SpVekoTEzTs08FubiAzVl08S/edit?usp=drive_link&ouid=115763589927341337064&rtpof=true&sd=true

8. Reglas de negocio

- Solo administradores pueden modificar datos.
- Los productores solo acceden a su información.
- Los pagos se generan los primeros 5 días del mes.

- Los datos históricos no pueden eliminarse.

9. Requerimientos de interfaces externas

9.1. Interfaces de usuario

La interfaz de usuario del sistema será responsiva, asegurando su correcto funcionamiento en distintos dispositivos y tamaños de pantalla (computadoras de escritorio, tablets y dispositivos móviles). Se garantizará la accesibilidad conforme a las pautas WCAG 2.1, permitiendo el uso a personas con diferentes tipos de discapacidad.

Las pantallas estarán organizadas mediante paneles intuitivos, empleando menús horizontales y verticales que permitan una navegación fluida. Se aplicarán estándares de interfaz gráfica de usuario (GUI) que contemplan:

- Consistencia en el uso de colores, íconos y tipografía.
- Botones claramente identificables con retroalimentación visual.
- Ubicación fija de botones principales como "Guardar", "Cancelar" y "Volver".
- Campos de entrada con validaciones automáticas y mensajes de ayuda contextual.

9.2. Interfaces de hardware

El sistema será compatible con diversos dispositivos, incluyendo:

- Computadoras personales (Windows, macOS, Linux).
- Dispositivos móviles (Android, iOS).
- Impresoras conectadas vía red o USB, para la generación de reportes en PDF y documentos administrativos.

Se soportarán protocolos de comunicación estándar como USB, Bluetooth (en caso de ser necesario) y Wi-Fi para la interacción entre dispositivos.

Se definirá una capa de abstracción que permita la interoperabilidad del software con distintos controladores de hardware según el sistema operativo.

9.3. Interfaces de software

El sistema integrará los siguientes componentes de software:

- Back-end: desarrollado con el framework Django, que manejará la lógica de negocio y las operaciones del lado del servidor.
- Front-end: implementado en React.js, permitiendo una experiencia de usuario dinámica y moderna mediante el uso de componentes reutilizables
- Base de datos: gestionada por PostgreSQL, sistema robusto y escalable que almacena la información estructurada del sistema.

La integración entre estos componentes se realizará mediante API RESTful. Además, se garantizará la compatibilidad con sistemas operativos basados en Unix/Linux y Windows Server, bibliotecas específicas de Python, herramientas de visualización (Chart.js, Recharts) y posibles integraciones con otros servicios web externos mediante JSON y/o XML.

9.4. Interfaces de comunicación

El sistema utilizará el protocolo seguro HTTPS para todas las comunicaciones entre el cliente y el servidor, garantizando la cifrado de datos mediante certificados SSL/TLS. Los servicios de mensajería electrónica se realizarán mediante SMTP con autenticación segura, permitiendo el envío de notificaciones, alertas y correos de confirmación.

Los protocolos admitidos incluirán:

- HTTP/HTTPS para el acceso web.
- SMTP/IMAP/POP3 para servicios de correo electrónico.
- REST/JSON para servicios web.

- FTP/SFTP, en caso de requerir intercambio de archivos.

10. Requerimientos no funcionales

1. Seguridad:

- El sistema debe implementar un mecanismo de autenticación de dos factores (2FA) utilizando código enviado por SMS o correo electrónico para verificar la identidad del usuario.
- Toda la comunicación entre el cliente y el servidor debe estar cifrada mediante el protocolo HTTPS con certificados SSL actualizados.
- Las contraseñas deben almacenarse utilizando algoritmos de hash seguros como bcrypt o Argon2.
- Los accesos a datos sensibles deben ser auditables y registrar el historial de modificaciones (logs).

2. Usabilidad:

- El sistema debe contar con una interfaz intuitiva y coherente que facilite su uso por parte de usuarios sin experiencia técnica.
- Se debe aplicar un enfoque de diseño centrado en el usuario (DCU), realizando pruebas de usabilidad en fases tempranas del desarrollo.
- El tiempo promedio de aprendizaje para tareas básicas no debe superar los 10 minutos para un nuevo usuario.

3. Portabilidad:

- La aplicación debe ser compatible con navegadores modernos (Chrome, Firefox, Edge, Safari) y funcionar correctamente en dispositivos móviles y de escritorio.
- El sistema debe adaptarse automáticamente a diferentes tamaños de pantalla mediante diseño web responsivo.

- No debe requerir instalaciones adicionales para su funcionamiento en plataformas estándar (Android, iOS, Windows, Linux, macOS).

4. Escalabilidad:

- El sistema debe ser modular, permitiendo la integración de nuevos componentes sin afectar el núcleo de la aplicación.
- Debe estar preparado para operar adecuadamente con hasta 100.000 usuarios con sesiones concurrentes.
- La arquitectura debe permitir la escalabilidad horizontal (agregando nuevos servidores) y vertical (aumentando los recursos del servidor actual).

5. Rendimiento:

- El sistema debe responder a cualquier acción del usuario en un tiempo inferior a 2 segundos en el 95% de los casos.
- Las operaciones críticas como autenticación, consulta de datos y generación de reportes deben completarse en menos de 3 segundos bajo carga máxima.
- El sistema debe mantener un uso eficiente de los recursos, utilizando menos del 75% de CPU y memoria bajo condiciones normales de operación.

11. Otros requerimientos

- Internacionalización: Soporte futuro multilenguaje.
- Cumplimiento legal: Normativa de datos vigente.
- Compatibilidad futura: APIs RESTful.

12. Glosario

- Seguridad:

Conjunto de medidas técnicas y organizativas implementadas para proteger la integridad, confidencialidad y disponibilidad de los datos del sistema. Incluye mecanismos como autenticación multifactor (2FA), cifrado de extremo a extremo mediante TLS/SSL, almacenamiento seguro de credenciales (por ejemplo, hash con bcrypt), y control de acceso basado en roles (RBAC).

- Usabilidad:

Grado en que el sistema permite a los usuarios alcanzar objetivos específicos con eficacia, eficiencia y satisfacción en un contexto de uso definido. En el desarrollo de software, se traduce en la aplicación de principios de interacción humano-computadora (HCI), accesibilidad, navegación intuitiva y diseño centrado en el usuario (UCD).

- Portabilidad:

Capacidad del sistema para ejecutarse correctamente en diversas plataformas hardware o software sin requerir modificaciones significativas. Implica compatibilidad con múltiples sistemas operativos, navegadores web y dispositivos (desktop, tablet, smartphone), y está relacionada con el uso de tecnologías multiplataforma y diseño responsivo.

- Escalabilidad:

Propiedad arquitectónica del sistema que permite aumentar su capacidad de procesamiento y almacenamiento sin comprometer la estabilidad ni el rendimiento. Puede lograrse mediante escalabilidad horizontal (distribución de carga entre múltiples instancias) o vertical (mejora de los recursos en una instancia única). Es clave en sistemas con crecimiento progresivo de usuarios y datos.

- Rendimiento:

Capacidad del sistema para ejecutar operaciones en tiempos óptimos bajo condiciones normales y de carga máxima. Se mide comúnmente mediante métricas como tiempo de respuesta, latencia, throughput y uso de

recursos (CPU, memoria, ancho de banda). Un buen rendimiento es fundamental para garantizar la experiencia del usuario y la eficiencia operativa.