

# GESTIÓN DE GIMNASIOS

## SQL -SLayers

Barón Salinas Mauricio  
Valentín

Cristóbal Silverio Cristian  
Gómez Daniel Aimar Jair  
Hernández Reyes Magaly  
Sánchez Carrasco Monserrat  
Ramírez Suárez Gerardo



Sistema  
Distribuido



# Tabla de Contenido

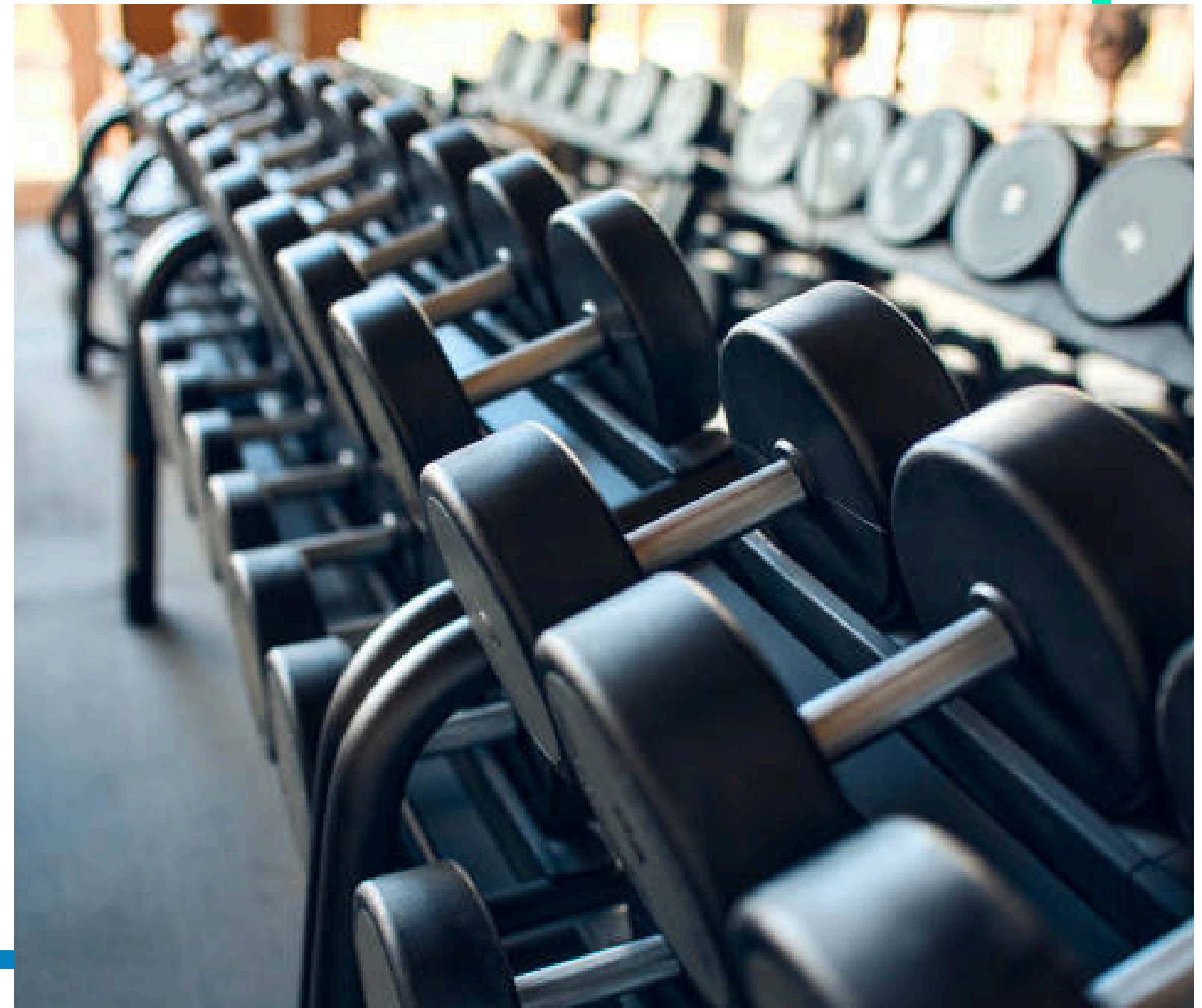
01	Introducción
02	Objetivos y misión
03	Definición y conceptos
04	Ventajas y desafíos
05	Propuesta de proyecto
06	Casos de éxito
07	Proceso de desarrollo
08	Tendencias del mercado
09	Equipo de trabajo
10	Conclusiones



# Definición del Proyecto

Este proyecto consiste en desarrollar una base de datos que permita gestionar eficientemente las operaciones de una cadena de gimnasios.

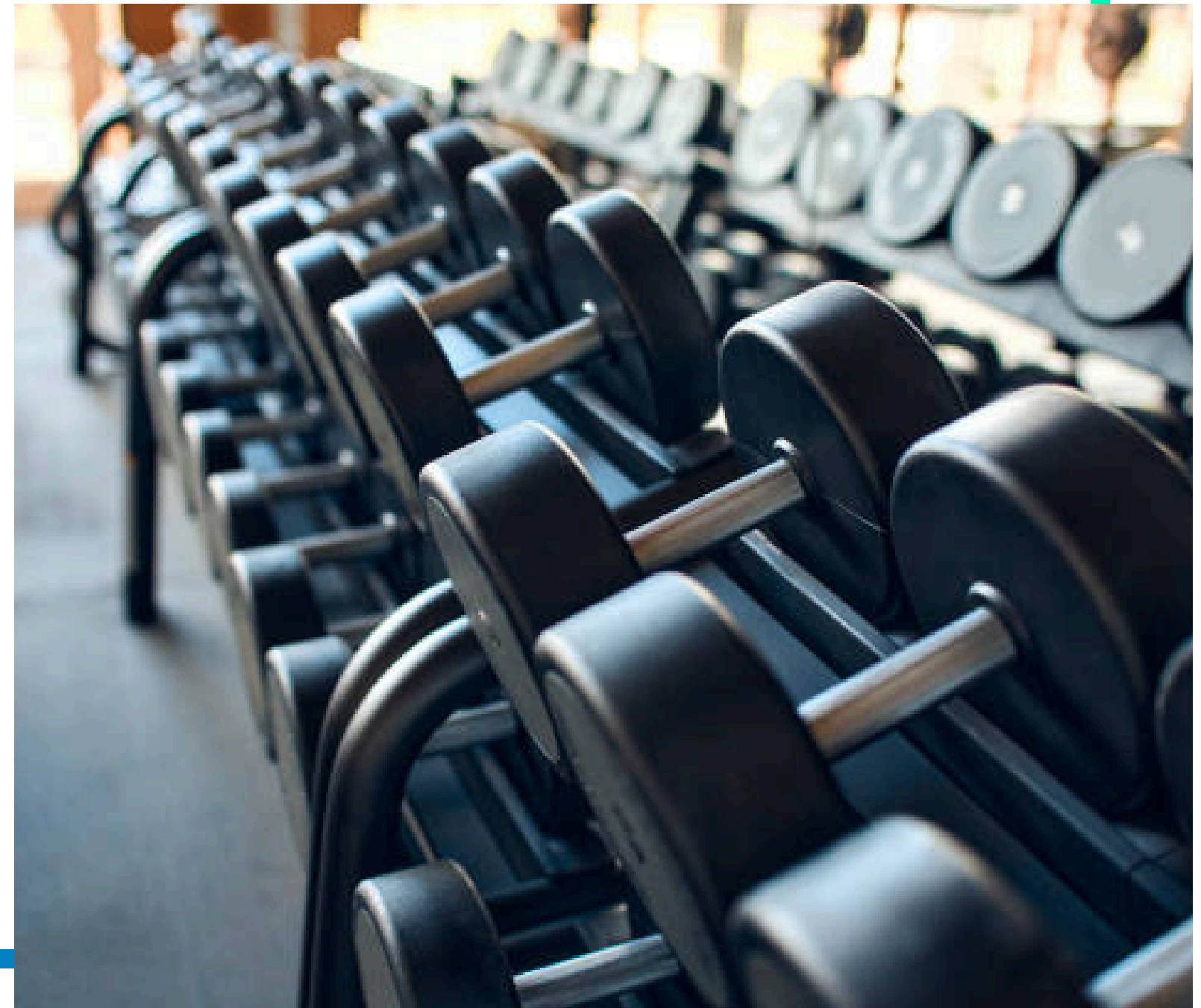
Su objetivo principal es optimizar la administración diaria y mejorar la experiencia de los usuarios mediante un sistema integral para manejar membresías, pagos, empleados e inventario.



# Definición del Proyecto

Además, busca asegurar una gestión eficiente en todas las áreas operativas y adaptarse al crecimiento del negocio.

- Gestionar datos de múltiples sedes con distribución geográfica.
- Sincronizar y garantizar consistencia en la información entre sedes.
- Escalar el sistema según el crecimiento del negocio.
- Proteger los datos mediante medidas de seguridad.
- Asegurar resiliencia y recuperación ante desastres.








# Introducción

La eficiencia de un gimnasio no solo depende de sus instalaciones o personal, sino también de la adecuada gestión de los datos de sus miembros y operaciones.

La cantidad de información que manejan, como control de membresías, pagos y datos personales, hace necesario contar con un sistema bien organizado para evitar problemas y mejorar la experiencia del usuario.

Este proyecto busca desarrollar una base de datos que optimice la administración diaria, mejore el servicio al cliente y aumente la eficiencia operativa del gimnasio.



# Plantamiento del Problema

El gimnasio enfrenta dificultades para gestionar membresías, pagos y datos de usuarios. La falta de un sistema integral genera desorganización, pérdida de información y problemas en el seguimiento de transacciones, afectando la experiencia del cliente y la rentabilidad del negocio.



# Obejtivo General

Desarrollar una base de datos escalable que optimice la administración y el control de las áreas operativas, incluyendo gestión de membresías, inventario de equipos y productos, y seguimiento de empleados en cada sucursal.

La base de datos garantizará integridad y disponibilidad de la información en tiempo real, facilitando decisiones estratégicas, mejorando la eficiencia operativa y asegurando un control efectivo de los recursos conforme el negocio crezca.





# Objetivos Especificos

01

Desarrollar una base de datos distribuida que permita el almacenamiento de la información de los clientes, entrenadores, rutinas de entrenamiento, y pagos

02

Garantizar la comunicación entre los nodos de la base de datos para que la información esté sincronizada y disponible en todo momento, independientemente de la localización geográfica de los gimnasios o sucursales.

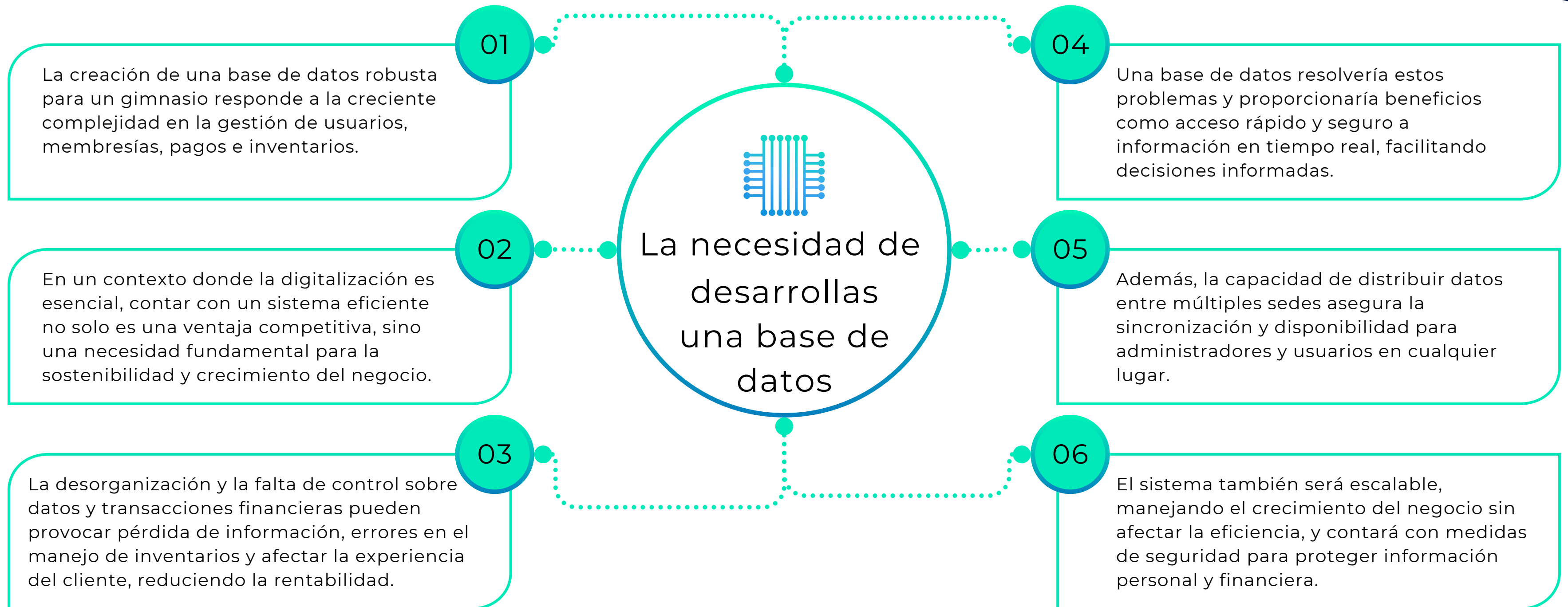
03

Desarrollar una interfaz de usuario amigable, utilizando HTML, CSS y JavaScript, que permita a los usuarios acceder, visualizar y modificar datos.





# Justificación



# Alcances y Limitaciones

## Alcances



### Distribución Geográfica de Datos:

Implementación de una base de datos distribuida que permita almacenar y gestionar datos en múltiples ubicaciones físicas—en nuestro caso, las sedes de gimnasio gestionadas por el cliente—para mejorar la disponibilidad y la redundancia.

### Sincronización y Consistencia de Datos:

Desarrollo de mecanismos que garanticen que los datos sean consistentes entre las distintas ubicaciones y que cualquier actualización se propague adecuadamente.

### Escalabilidad:

Diseño de la base de datos para que pueda escalar horizontalmente, añadiendo más nodos según sea necesario para manejar el aumento del volumen de datos y la carga de trabajo.

# Alcances y Limitaciones

## Alcances



### Seguridad de Datos:

Implementación de medidas de seguridad que protejan los datos distribuidos contra accesos no autorizados y posibles ataques cibernéticos.

### Resiliencia y Recuperación ante Desastres:

Establecimiento de procedimientos y tecnologías para asegurar la recuperación rápida y efectiva en caso de fallos o desastres.

# Alcances y Limitaciones

## Limitaciones



### Latencia de Red:

La latencia de la red entre los nodos puede afectar el rendimiento del sistema, especialmente si los nodos están ubicados en regiones geográficas distantes.

### Complejidad de Gestión:

La administración y el mantenimiento de una base de datos distribuida pueden ser significativamente más complejos en comparación con una base de datos centralizada.

### Costos de Infraestructura:

Los costos asociados con el hardware, el software y la red para soportar una base de datos distribuida pueden ser elevados.



# Alcances y Limitaciones

## Limitaciones



### Consistencia Eventual:

Algunas arquitecturas distribuidas optan por la consistencia eventual, lo que puede llevar a periodos en los que los datos no están sincronizados en todos los nodos.

### Problemas de Sincronización:

La sincronización de datos entre nodos puede enfrentar problemas, especialmente en redes con alta latencia o en escenarios de particionamiento de red

# Marco Teórico

El marco teórico guía el desarrollo estructurado del proyecto, desde la recopilación de información hasta la implementación de herramientas necesarias para cada etapa. Se basa en enfoques metodológicos y técnicas específicas para garantizar una ejecución eficiente y ordenada.

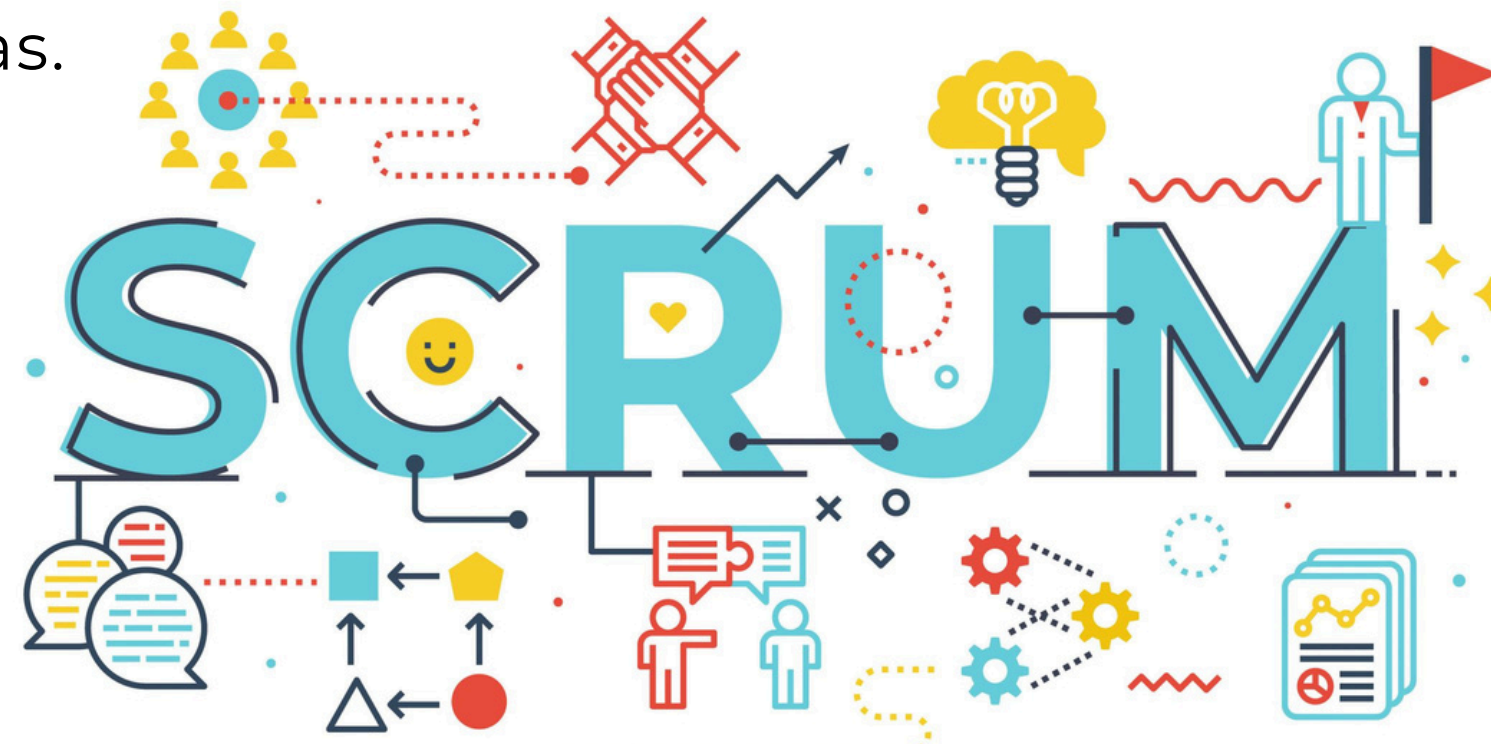
01

## SCRUM

SCRUM es un marco ágil utilizado para la gestión y desarrollo de proyectos de software, basado en la colaboración, autoorganización y entregas incrementales.

Este método incluye:

- Roles: Scrum Master, Product Owner y equipo de desarrollo.
- Sprints: Unidades de trabajo con objetivos definidos y un tiempo preestablecido, generalmente de 30 días.
- Reuniones diarias: Breves sesiones para revisar avances, identificar obstáculos y planificar tareas.



# Marco Téorico

02

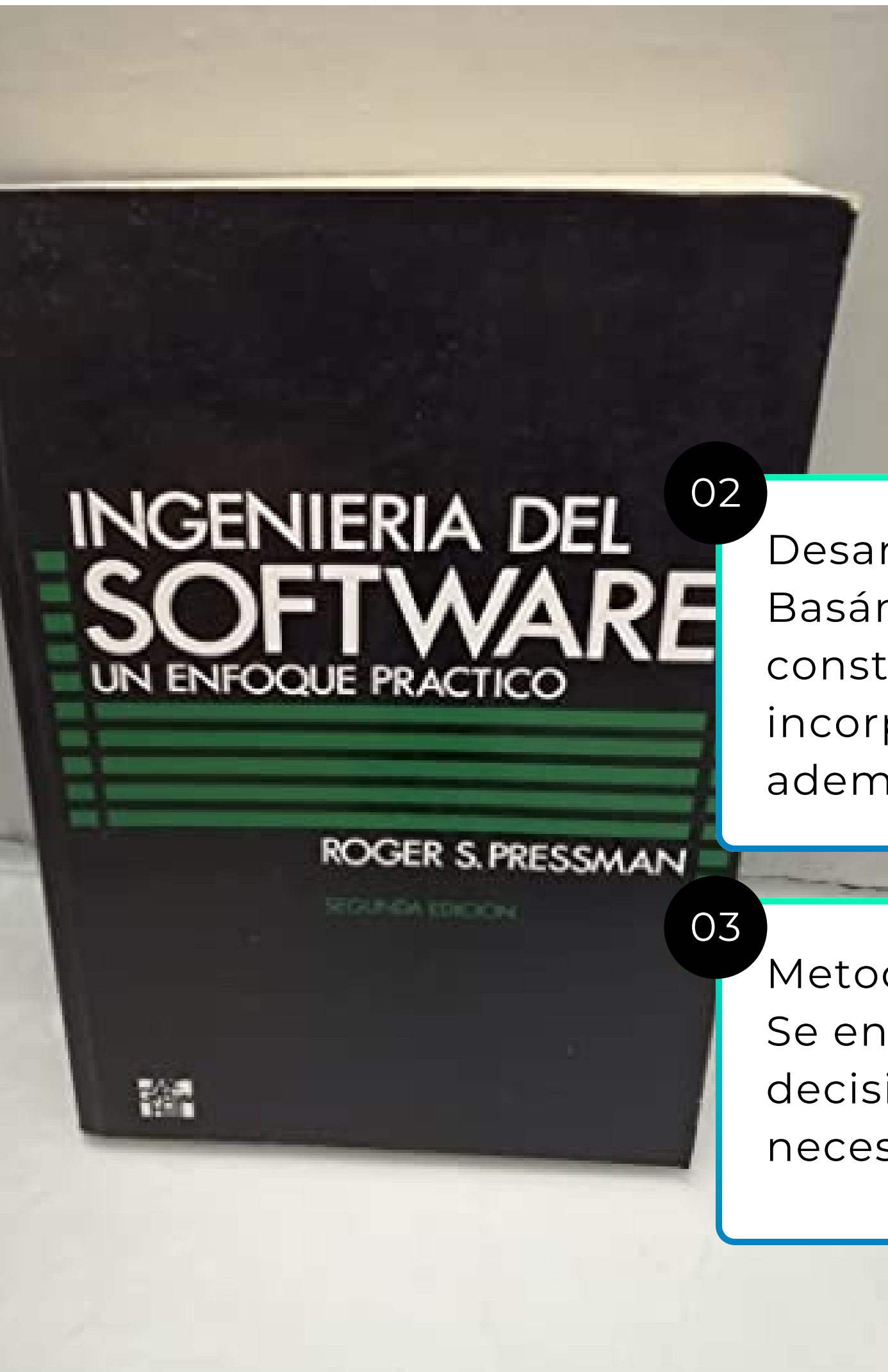
## Desarrollo Iterativo e Incremental

Basándonos en Pressman, se aplicará un enfoque iterativo para construir y mejorar gradualmente la base de datos distribuida, incorporando prácticas de gestión, estimación y control del proyecto, además de la detección temprana y corrección de errores.

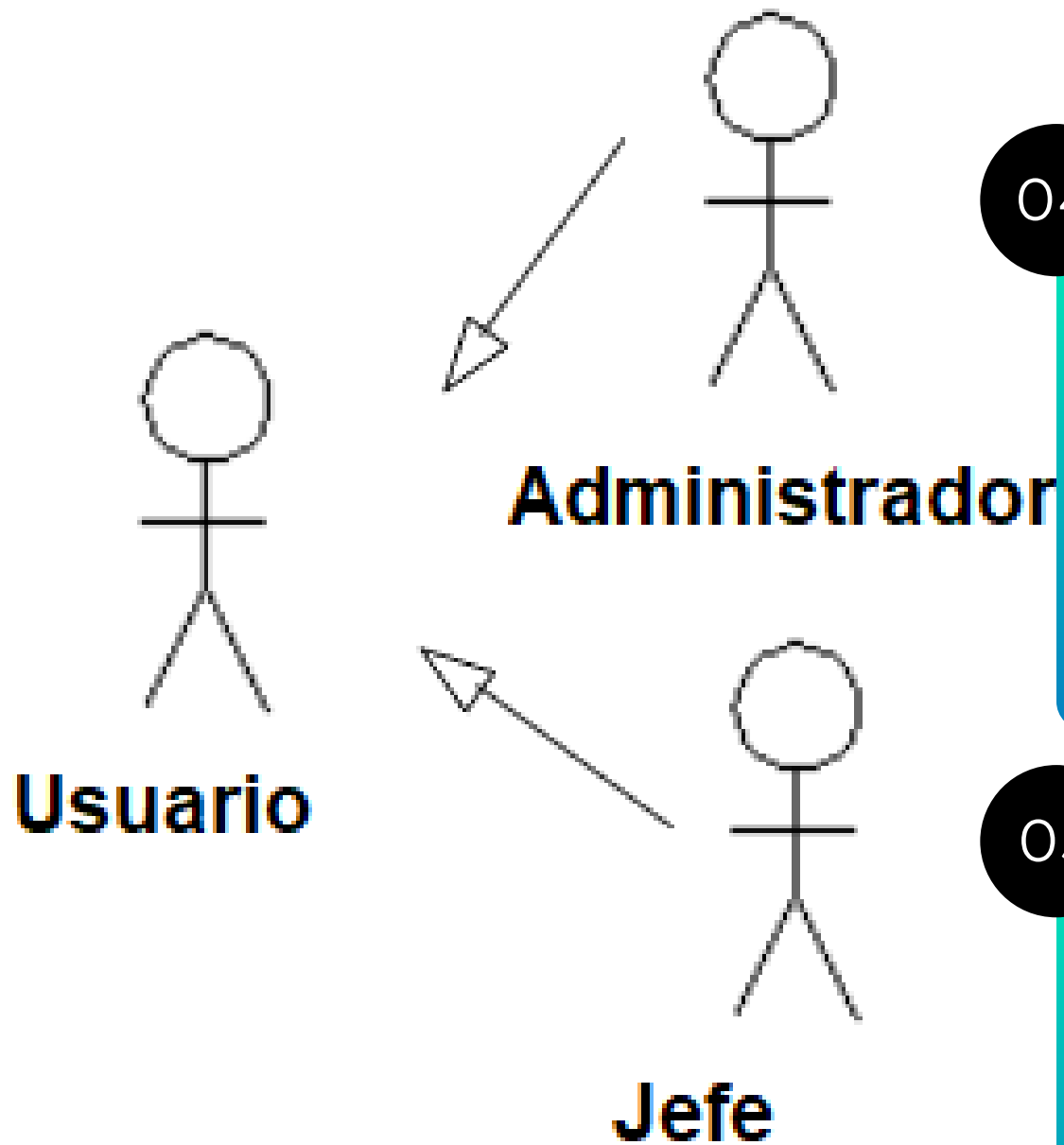
03

## Metodología de Análisis

Se enfoca en la recopilación y evaluación de información para tomar decisiones informadas. Se analizan documentos, entrevistas y necesidades para resolver problemas de forma estructurada.



# Marco Téorico



04

## Casos de Uso

Describen cómo interactúan los usuarios con el sistema en diferentes condiciones, ilustrando el comportamiento del sistema desde el punto de vista del usuario final.

05

## Historias de Usuario

Narrativas breves que definen necesidades concretas desde la perspectiva de un usuario, detallando interacciones o tareas específicas.



# Marco Téorico

06

Entrevista de Requerimientos  
A través de reuniones individuales o grupos de enfoque, se identifican necesidades, motivaciones y aspectos clave de los usuarios para garantizar que el sistema se adapte a sus requerimientos.



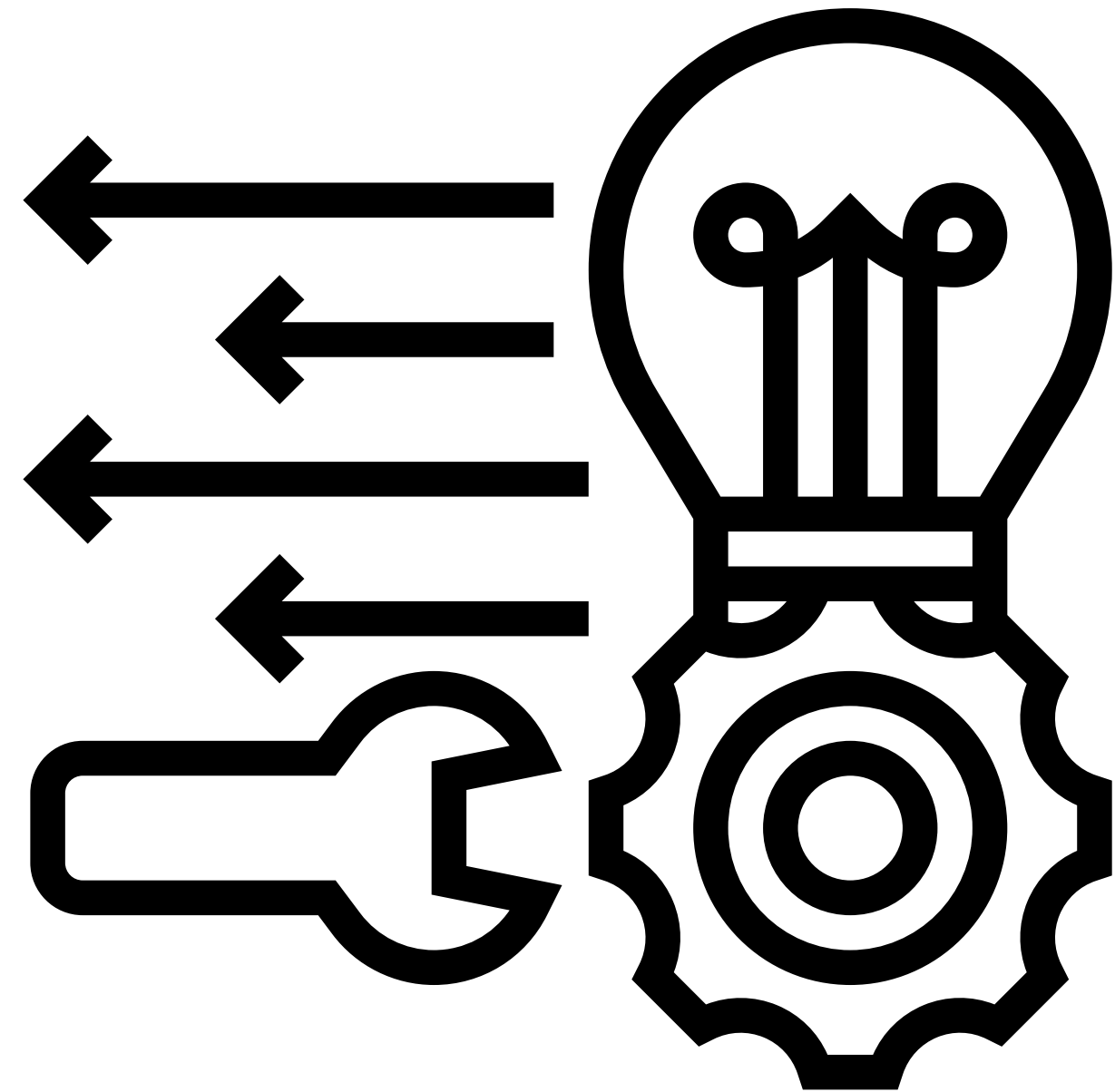
# Metodología de Diseño

Guía el proceso creativo y la resolución de problemas, desde la evaluación de diversas posibles soluciones. Su salida debe consistir en un activo potencial para el desarrollo del sistema, a partir de la generación de una estructura más sólida para el entendimiento del proyecto.



# Definición del Problema

- Definición clara del problema a abordar
- Identificación de las necesidades y expectativas de los usuarios
  - Definición de los objetivos del diseño



# Desarrollo de Conceptos

01 Proposición de ideas.

02 Selección de las ideas más prometedoras

03 Desarrollo en conceptos más concretos.

04 Esbozo de prototipos iniciales o maquetas para visualizar y evaluar las soluciones propuestas.

x x x





# Evaluación y Pruebas

- Los conceptos desarrollados se prueban y evalúan, a menudo mediante feedback de usuarios.
- Se identifican fallos y áreas de mejora, permitiendo iteraciones en el diseño



## Metodología de Desarrollo

Proporciona un marco estructurado en el que ya se puede planificar, ejecutar y gestionar el proyecto de manera eficiente.

# Planificación

01

Definición del alcance del proyecto

02

Identificación de los objetivos

03

Elaboración de un cronograma

04

Definición de los recursos necesarios

05

Asignación de roles y responsabilidades al equipo



01

## Implementación

Una vez que el software ha sido probado y validado, se despliega en el entorno de producción. Esto puede incluir la migración de datos y la capacitación de usuarios, asegurando una transición fluida

02

## Mantenimiento

Después de la implementación, se requiere un seguimiento continuo para corregir errores, realizar actualizaciones y mejorar el sistema en función del feedback de los usuarios. Este componente es esencial para asegurar la longevidad y la relevancia del producto.

03

## Evaluación y Mejora Continua

Se lleva a cabo una revisión del proceso y los resultados del proyecto. Esta evaluación permite identificar lecciones aprendidas y oportunidades de mejora, lo que contribuye a la optimización de futuras metodologías de desarrollo

# Resultados



El proyecto logró cumplir con varios objetivos clave establecidos desde su inicio.

- Gestionar múltiples sedes de gimnasios a través de una arquitectura distribuida, a partir de la implementación de un sistema de bases de datos distribuidas utilizando tecnologías como MongoDB.
- Garantizar la consistencia de los datos a través de las distintas ubicaciones a partir de la sincronización en tiempo real utilizando mecanismos de replicación en MongoDB y técnicas de consistencia eventual.







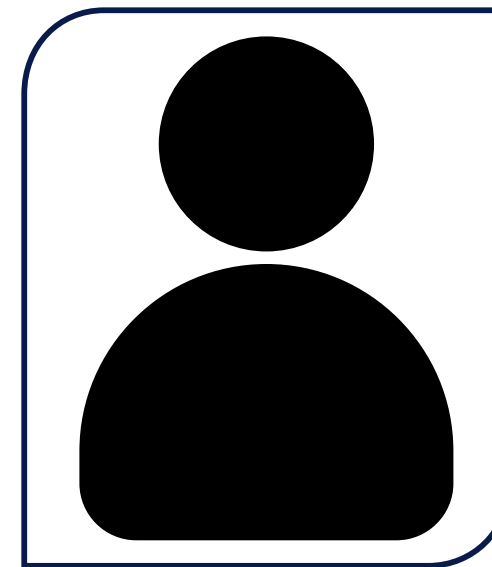
# Conclusiones

El proyecto ha cumplido con sus objetivos principales, proporcionando una solución robusta, escalable y segura para gestionar de manera eficiente los gimnasios distribuidos. El sistema está preparado para afrontar el crecimiento del negocio y para ofrecer una experiencia de usuario mejorada, todo mientras garantiza la seguridad y la disponibilidad continua de los datos.

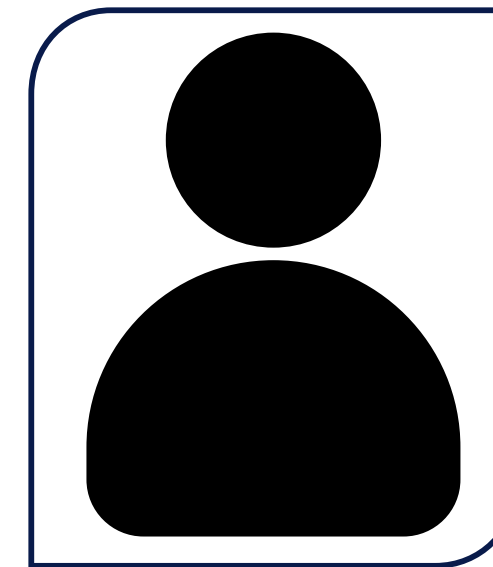


# Nuestro Equipo

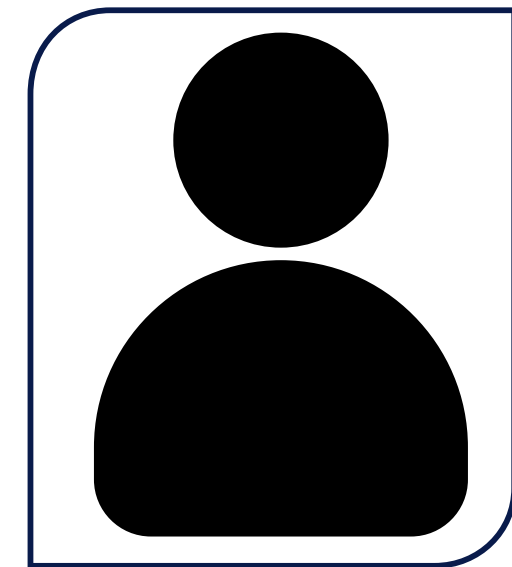
SQL MANAMEGENT



Barón Salinas  
Mauricio Valentín



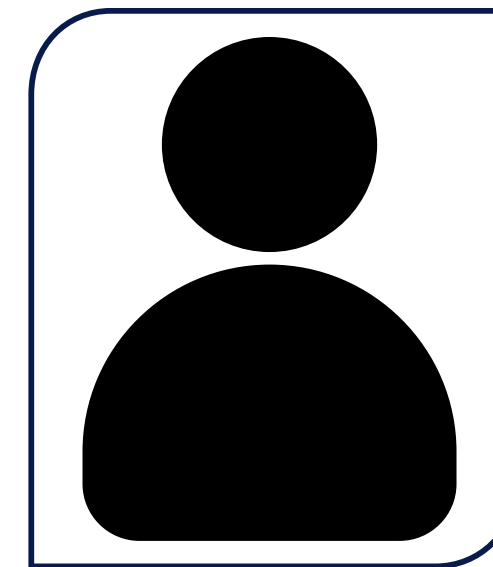
Cristobal Silverio  
Cristian



Gomez Daniel  
Aimar Jair



Hernandez Reyes  
Magaly



Ramirez Suarez  
Gerardo



Sanchez Carrasco  
Montserrat



Muchas Gracias