



GESTIÓN DE
GIMNASIOS

2024

Sistema Distribuido

SQL-Slayers

Barón Salinas Mauricio Valentín

Cristóbal Silverio Cristian

Gómez Daniel Aimar Jair

Hernández Reyes Magaly

Sánchez Carrasco Monserrat

Ramírez Suárez Gerardo

Índice

1. Definición de Proyecto	4
1.1. Introducción	4
1.2. Planteamiento del problema	4
1.3. Objetivo General	4
1.4. Objetivos Específicos	4
1.5. Justificación	4
1.6. Alcances y Limitaciones	4
1.6.1. Alcances del Proyecto	4
1.6.2. Limitaciones del Proyecto	4
2. Marco teorico	5
2.1. Descripción	5
2.1.1. Metodología de Analisis	5
3. Historias de usuario y Casos de uso	6
3.1. Casos de Uso	6
3.2. Historias de Usuario	6
3.3. Entrevista de requerimientos	6
3.4. Analisis	6
3.4.1. Entrevista	6
3.4.2. Registro de miembros	6
3.4.3. Seguimiento Administrativo	7
3.4.4. Accesos al sistema	7
3.4.5. Integración	7
3.4.6. Soporte y mantenimiento	8
3.4.7. Últimas palabras	8
3.5. Historias de Usuario	8
3.6. Diseño	8
3.7. Desarrollo	8
3.8. Prioridad de historias de usuario	9
3.9. Vision Board	9

4. Cronograma de actividades	10
4.1. Definición de proyecto	10

1. Definición de Proyecto

El proyecto en cuestión plantea el desarrollo de una base de datos para gestionar de manera eficiente las operaciones de una serie de gimnasios. Su enfoque principal está en la optimización de la administración diaria y en la mejora de la experiencia de los usuarios a través de un sistema que maneje de manera integral la información de membresías, pagos, empleados y el inventario de productos o equipos. El propósito final de este sistema es garantizar que todas las áreas operativas del gimnasio sean gestionadas de manera eficiente, con la capacidad de escalar a medida que el negocio crezca.

Se destacan cinco objetivos específicos que buscan cubrir aspectos claves del funcionamiento del gimnasio:

1. **Distribución geográfica** de los datos para gestionar múltiples sedes.
2. **Sincronización y consistencia** de la información entre esas sedes.
3. **Escalabilidad** del sistema para manejar el crecimiento del negocio.
4. **Implementación** de medidas de seguridad para proteger los datos.
5. **Garantizar la resiliencia y recuperación** en caso de desastres.

A su vez, el proyecto identifica algunas limitaciones, tales como la latencia de la red entre las distintas ubicaciones, la complejidad en la gestión de un sistema distribuido, los costos de infraestructura asociados, y posibles problemas de sincronización en redes de alta latencia.

1.1. Introducción

En la actualidad, la eficiencia de un gimnasio depende no solo de la calidad de sus instalaciones o de su personal, si no también de la correcta gestión de los datos de sus miembros, participantes y operaciones diarias. Los gimnasios de hoy en día manejan una enorme cantidad de información que contiene el control de membresías, el seguimiento de sus pagos e información personal de sus integrantes. Por ello, sin un sistema bien organizado es fácil perder el control sobre los aspectos antes mencionados y ocasiona una experiencia negativa en los usuarios.

Este proyecto tiene la tarea de crear una base de datos que permita la excelente gestión de un gimnasio para optimizar la administración diaria, mejorar el servicio al cliente y aumentar la eficiencia operativa.

1.2. Planteamiento del problema

El gimnasio enfrenta dificultades para gestionar eficientemente las membresías, pagos y datos de usuarios. La falta de un sistema integral provoca desorganización, pérdida de información y dificultades en el seguimiento de las transacciones, lo que puede afectar la experiencia del cliente y la rentabilidad del negocio.

1.3. Objetivo General

Desarrollar una base de datos para un gimnasio, buscando una capacidad de escalabilidad, que nos permita optimizar la administración y el control de todas las áreas operativas, incluyendo gestión de mem-

bresías, el manejo de inventario de equipos y productos como suplementos de igual manera el seguimiento de los empleados de cada sucursal.

La base de datos nos dará la integridad y disponibilidad de la información en tiempo real para facilitar la toma de decisiones estratégicas, mejorar la eficiencia operativa y asegurar un control efectivo de los recursos a medida que el negocio crezca.

1.4. Objetivos Específicos

- Desarrollar una base de datos distribuida que permita el almacenamiento de la información de los clientes, entrenadores, rutinas de entrenamiento, y pagos.
- 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.
- Desarrollar una interfaz de usuario amigable, utilizando HTML, CSS y JavaScript, que permita a los usuarios acceder, visualizar y modificar datos.

1.5. Justificación

La necesidad de desarrollar una base de datos robusta para la gestión de un gimnasio surge de la creciente complejidad en la administración de los datos relacionados con los usuarios, sus membresías, los pagos y el manejo de inventario. En el contexto actual, donde la digitalización de los servicios se ha convertido en un estándar, contar con un sistema de gestión que permita optimizar las operaciones diarias no solo es una ventaja competitiva, sino también una necesidad fundamental para la sostenibilidad y crecimiento de un gimnasio.

Uno de los problemas principales que enfrentan los gimnasios es la desorganización y falta de control sobre los datos de sus miembros y transacciones financieras. La ausencia de un sistema centralizado de información puede dar lugar a la pérdida de datos, dificultades en la trazabilidad de las operaciones y errores en la gestión de inventarios, lo que puede perjudicar la experiencia del cliente y reducir la rentabilidad del negocio.

Por lo tanto, la implementación de una base de datos no solo resolvería estos problemas, sino que también ofrecería una serie de beneficios adicionales. Por ejemplo, permitiría un acceso rápido y seguro a la información en tiempo real, lo que facilitaría la toma de decisiones informadas. Además, la posibilidad de contar con un sistema que distribuya los datos geográficamente en múltiples ubicaciones permitiría gestionar de manera efectiva las distintas sedes de un gimnasio, asegurando que toda la información esté sincronizada y disponible para los administradores y usuarios en cualquier momento y lugar.

Asimismo, la escalabilidad del sistema es un aspecto crítico en este proyecto, ya que a medida que el gimnasio crezca y aumente la cantidad de miembros, empleados y equipos, la base de datos debe ser capaz de manejar este crecimiento sin comprometer la eficiencia operativa. La seguridad también es un factor esencial, especialmente cuando se trata de la protección de información personal y financiera de los usuarios. Las medidas de seguridad que se implementarán estarán destinadas a prevenir accesos no autorizados y garantizar la integridad de los datos.

Otro punto a considerar es la resiliencia del sistema. Dado que un gimnasio maneja información crítica de sus operaciones, es indispensable que la base de datos esté preparada para recuperar la información de manera rápida y efectiva en caso de cualquier eventualidad, como fallos técnicos o desastres naturales. Esto asegurará la continuidad de las operaciones sin afectar el servicio al cliente.

1.6. Alcances y Limitaciones

1.6.1. Alcances

1. **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.
2. **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.
3. **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.
4. **Seguridad de Datos:** Implementación de medidas de seguridad que protejan los datos distribuidos contra accesos no autorizados y posibles ataques cibernéticos.
5. **Resiliencia y Recuperación ante Desastres:** Establecimiento de procedimientos y tecnologías para asegurar la recuperación rápida y efectiva en caso de fallos o desastres.

1.6.2. Limitaciones

1. **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.
2. **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.
3. **Costos de Infraestructura:** Los costos asociados con el hardware, el software y la red para soportar una base de datos distribuida pueden ser elevados.
4. **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.
5. **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.

2. Marco teorico

2.1. Metodología de Analisis

2.1.1. ¿Qué es metodología Scrum?

Es un pequeño equipo de personas, un equipo Scrum. El equipo Scrum consta de:

- un Scrum Master,
- un propietario de producto (Product Owner) y
- desarrolladores.

Dentro de un equipo de Scrum, no hay sub-equipos ni jerarquías. Es una unidad cohesionada de profesionales enfocada en un objetivo a la vez, el objetivo del Producto.

Scrum (nombre que proviene de cierta jugada que tiene lugar durante un partido de rugby) es un método de desarrollo ágil de software concebido por Jeff Sutherland y su equipo de desarrollo a principios de la década de 1990.

Retraso: lista de prioridades de los requerimientos o características del proyecto que dan al cliente un valor del negocio. Es posible agregar en cualquier momento otros aspectos al retraso (ésta es la forma en la que se introducen los cambios). El gerente del proyecto evalúa el retraso y actualiza las prioridades según se requiera. Sprints: consiste en unidades de trabajo que se necesitan para alcanzar un requerimiento definido en el retraso que debe ajustarse en una caja de tiempo predefinida (lo común son 30 días). Durante el sprint no se introducen cambios (por ejemplo, aspectos del trabajo retrasado). Así, el sprint permite a los miembros del equipo trabajar en un ambiente de corto plazo pero estable. Reuniones Scrum: son reuniones breves (de 15 minutos, por lo general) que el equipo Scrum efectúa a diario. Hay tres preguntas clave que se pide que respondan todos los miembros del equipo.

- ¿Qué hiciste desde la última reunión del equipo?
- ¿Qué obstáculos estás encontrando?
- ¿Qué planeas hacer mientras llega la siguiente reunión del equipo? Un líder del equipo, llamado maestro Scrum, dirige la junta y evalúa las respuestas de cada persona.

2.1.2. ¿Qué es una Metodología de análisis?

Nosotros enfocamos un marco teorico que influye en el proceso estructurado para guiar el desarrollo desde los elementos de entrada(información recopilada), hasta las herramientas que se utilizaran para la elaboración de ellas. Asi facilitando su implementación en las diferentes etapas del proceso. Pressman promueve un enfoque iterativo e incremental en el desarrollo de software, en este caso lo utilizaremos en el desarrollo de nuestra base de datos distribuida en base al proyecto del GYM, esto para que se contruya y mejore partes del proyecto de manera gradual. Esto tambien tomando en cuenta los aspectos de gestión de proyectos, estimación, planificación y control, asi tambien la detección temprana de los defectos y su corrección inmediata.

3. Historias de usuario y Casos de uso

3.1. Casos de Uso

Analiza el comportamiento del sistema en distintas condiciones en las que el sistema responde a una petición de alguno de sus participantes. En esencia, un caso de uso narra una historia estilizada sobre cómo interactúa un usuario final (que tiene cierto número de roles posibles) con el sistema en circunstancias específicas. Un caso de uso ilustra el software o sistema desde el punto de vista del usuario final.

3.2. Historias de Usuario

Las historia de usuario consta de ser un texto narrativo, un lineamiento de tareas o interacciones, una descripción basada en un formato o una representación diagramática. Sin importar su forma

3.3. Entrevista de requerimientos

Éste es el enfoque más directo, los miembros del equipo de software se reúnen con los usuarios para entender mejor sus necesidades, motivaciones, cultura laboral y una multitud de aspectos adicionales. Esto se logra en reuniones individuales o a través de grupos de enfoque.

3.4. Analisis

3.4.1. Entrevista

Entrevistador: Buenas tardes. Gracias por tomarse el tiempo para esta entrevista. Para empezar, ¿puede darme una visión general de su negocio y qué es lo que espera lograr con este nuevo sistema?

Administrador: Buenas tardes. Claro, tengo varios gimnasios en diferentes ubicaciones, y actualmente gestionamos todo manualmente o con sistemas muy básicos. Lo que realmente necesito es un sistema que me permita gestionar las inscripciones de los miembros, el seguimiento de su actividad y productividad, y también la administración de los empleados y horarios.

Entrevistador: Entiendo. Vamos a desglosar esto en partes. Primero, ¿puede describir cómo maneja actualmente el registro de miembros y qué aspectos le gustaría mejorar?

Administrador: En este momento, usamos un sistema de hojas de cálculo para el registro de nuevos miembros y el seguimiento de pagos. Esto es bastante ineficiente y propenso a errores. Me gustaría que el nuevo sistema permitiera registrar automáticamente nuevos miembros y gestionar sus pagos.

3.4.2. Registro de miembros

Entrevistador: Perfecto. Para el registro de miembros, ¿qué información específica necesita capturar?

Administrador: Necesitamos capturar información básica como:

- nombre,
- número de teléfono principal y
- número de teléfono alternativo.

También necesitamos registrar:

- el tipo de membresía,
- la fecha de inicio y
- los pagos realizados.

3.4.3. Seguimiento Administrativo

Entrevistador: En relación a la administración de empleados y horarios, ¿qué funcionalidades específicas está buscando?

Administrador: Necesito un módulo para gestionar:

- los horarios de los empleados, incluyendo:
 - las asignaciones de turno y
 - las solicitudes de vacaciones.

También sería útil poder hacer un seguimiento de:

- sus horas trabajadas y
- sus pagos.

3.4.4. Accesos al sistema

Entrevistador: Entiendo. ¿Qué tipo de acceso y permisos deberían tener los diferentes usuarios del sistema? Por ejemplo, ¿qué pueden ver o hacer los administradores frente a los empleados o los entrenadores?

Administrador:

- Los administradores deberían tener acceso completo a todas las funciones del sistema, incluyendo: la gestión de miembros, empleados, informes y configuraciones.
- Los empleados y entrenadores solo deberían tener acceso a funciones relacionadas con su trabajo, como ver sus horarios y gestionar sus propias clases.

3.4.5. Integración

Entrevistador: Muy bien. En términos de integración, ¿hay otros sistemas o herramientas con los que el nuevo sistema debería integrarse, como sistemas de contabilidad o plataformas de marketing?

Administrador: Sí, el sistema debería integrarse con nuestro software de contabilidad para que los pagos se registren automáticamente.

3.4.6. Soporte y mantenimiento

Entrevistador: Perfecto. Para finalizar, ¿hay algún otro requerimiento o aspecto específico que no hayamos cubierto y que sea importante para usted?

Administrador: Creo que lo hemos cubierto bastante bien. Solo me gustaría:

- que el sistema fuera fácil de usar
- una interfaz llamativa.

La facilidad de uso es importante para que tanto los empleados como los miembros puedan adaptarse rápidamente.

3.4.7. Últimas palabras

Entrevistador: Entendido. Muchas gracias por la información detallada. Con estos requisitos, podemos empezar a definir el alcance del proyecto y trabajar en el diseño del sistema. ¿Hay algo más que quiera agregar antes de que terminemos?

Administrador: No, eso es todo. Gracias a usted.

Entrevistador: De nada. Le mantendremos informado sobre el avance del proyecto. Que tenga un buen día.

Administrador: Igualmente. Que tenga un buen día.

4. Historias de Usuario

Registro de Miembros Como administrador, quiero poder registrar nuevos miembros en el sistema, Para automatizar el proceso de inscripción y tener un registro preciso de los datos de los miembros.

- El sistema debe permitir ingresar información básica del miembro: nombre y número de teléfono
- El sistema debe registrar el tipo de membresía, fecha de inicio, fecha final y pagos realizados.

Gestión de Pagos Como administrador, quiero poder gestionar los pagos de los miembros dentro del sistema, para asegurarme de que los pagos se registren de manera precisa y automática.

- El sistema debe registrar pagos realizados por los miembros.
- El sistema debe permitir consultar el estado de cuenta y el historial de pagos.

Gestión de Empleados y Horarios Como administrador, quiero poder gestionar los horarios de los empleados, incluyendo turnos, para llevar un orden y supervisar las actividades de los trabajadores.

- El sistema debe hacer seguimiento de las horas trabajadas por cada empleado.
- El sistema debe gestionar los pagos de los empleados.

Gestión de inventario de suplementos Como administrador, quiero poder llevar un registro de mi inventario de suplementos para poder monitorear la venta de alguno y ver la disponibilidad de éstos.

- El sistema debe de permitir controlar el inventario de mis suplementos
- El sistema debe registrar las ganancias y precios por los suplementos.

5. Diagrama de Casos de uso

5.1. Diseño

- Arquitectura de Sistema de Base de Datos Distriuido
- Esquema Conceptual Global
- Esquema Conceptual Local

5.2. Desarrollo

- Implementacion del SBDD

5.2.1. Estrategía top-down

- Implementacion De Interfaces de Usuario
- Pruebas Unitarias

5.3. Vision Board

Registro de Miembros

Como **administrador**,
Quiero **poder registrar nuevos miembros en el sistema**,
Para **automatizar el proceso de inscripción y tener un registro preciso de los datos de los miembros**.

Gestión de Pagos

Como **administrador**,
Quiero **poder gestionar los pagos de los miembros dentro del sistema**,
Para **asegurarme de que los pagos se registren de manera precisa y automática**.

Seguimiento Administrativo

Como **administrador**,
Quiero **tener acceso a informes sobre la asistencia y frecuencia de uso de las instalaciones por parte de los miembros**,
Para **monitorear la utilización de las instalaciones y mejorar la gestión de recursos**.

Gestión de Empleados y Horarios

Como **administrador**,
Quiero **poder gestionar los horarios de los empleados, incluyendo turnos y solicitudes de vacaciones**,
Para **facilitar la administración del personal y asegurar una cobertura adecuada en los gimnasios**.

Control de Acceso y Permisos

Como **administrador**,
Quiero **definir diferentes niveles de acceso y permisos** para los usuarios del sistema,
Para **asegurar que cada usuario tenga acceso solo a las funciones relevantes para su rol.**

Integración con Otros Sistemas

Como **administrador**,
Quiero **integrar el sistema con el software de contabilidad y plataformas de marketing**,
Para **facilitar la gestión financiera y la comunicación con los miembros.**

Facilidad de Uso y Soporte Técnico

Como **usuario del sistema**,
Quiero **que el sistema sea fácil de usar y cuente con buen soporte técnico**,
Para **poder adaptarme rápidamente y recibir ayuda cuando sea necesario.**

6. Cronograma de actividades

6.1. Definición de proyecto

