

Universidad de Concepción
Departamento de Ingeniería Informática y Ciencias de la Computación

Usabilidad y visualización de datos para la asignación y seguimiento de rutinas de ejercicio

Nicolás Herrera Vega

Memoria presentada para la obtención del título de Ingeniero Civil Informático

Patrocinante: Gonzalo Rojas Durán
Comisión Evaluadora: José Fuentes Sepúlveda, Geoffrey Hecht

Enero de 2026

Índice

Índice	2
1. Introducción	5
1.1. Antecedentes Generales del Problema	5
1.2. Solución Propuesta y Alcances	5
1.3. Objetivo General	6
1.4. Objetivos Específicos	6
1.5. Metodología	7
1.6. Estructura del informe	7
2. Marco Teórico	8
2.1. Usabilidad en aplicaciones móviles	8
2.2. Visualización de datos en el contexto deportivo	9
2.3. Estado del arte en soluciones de gestión de entrenamiento	10
2.4. Síntesis crítica	12
3. Descripción de la Solución	13
3.1. Contexto de aplicación	13
3.1.1 Levantamiento de requerimientos	13
3.2. Requerimientos del sistema	14
3.2.1. Requerimientos funcionales	14
3.2.2. Requerimientos no funcionales	16
3.2.3. Reglas de negocio	16
3.2.4. Supuestos y restricciones	17
3.3. Arquitectura del sistema	17
3.3.1. Arquitectura por capas de la aplicación móvil	18
3.3.1.1. Capa de Presentación	19
3.3.1.2. Capa de Lógica y Datos	20
3.3.1.3. Capa de Infraestructura Externa	20
3.3.1.4. Justificación de la arquitectura propuesta	20
3.3.2. Aplicación móvil	20
3.3.3. Backend en la nube	21
3.3.4. Dashboard web	21
3.3.5. Flujo de datos y conectividad	22
3.3.6. Decisiones de diseño	22
3.4. Modelo de datos	22
3.4.1. Descripción de Relaciones	24
3.4.2. Justificación del Diseño: Planificación vs. Ejecución	25
3.4.3. Reglas de Integridad y Lógica de Negocio	25
3.4.4. Seguridad y Control de Acceso	26
4. Implementación y Operación	26
4.1. Tecnologías y Herramientas	26

4.1.1. Aplicación móvil	26
4.1.2. Backend y servicios en la nube	28
4.1.3. Dashboard web	29
4.1.4. Conjunto tecnológico y coherencia de la solución	29
4.2. Desarrollo de funcionalidades clave	30
4.2.1. Planificación de sesiones por etapas	30
4.2.2. Repetir sesiones	31
4.2.3. Compartir sesiones	32
4.2.4. Cierre de sesión con validaciones	37
4.2.5. Historial y calendario de asistencia	38
4.2.6. Visualización analítica en dashboard web	40
4.3. Integración frontend backend	41
4.3.1. Inicialización y conexión	42
4.3.2. Autenticación y seguridad	42
4.3.3. Operaciones de datos	42
4.3.4. Manejo de imágenes	42
4.3.5. Manejo de estado y actualización de la interfaz	43
4.3.6. Manejo de errores y validaciones	43
4.4. Manual y condiciones de operación	43
4.4.1. Requisitos de operación	44
4.4.2. Flujo operativo diario	44
4.4.3. Capacitación del usuario	44
4.4.4. Mantenimiento del sistema	45
4.4.5. Manejo ante fallas	45
4.4.6. Seguridad y privacidad en operación	45
4.4.7. Límites operativos	45
5. Diseño y ejecución de pruebas	46
5.1. Diseño experimental y métricas	46
5.1.1. Objetivo de la evaluación	46
5.1.2. Diseño experimental	46
5.1.3. Tareas evaluadas	47
5.1.4. Métricas y herramientas de medición	47
5.1.5. Criterios de éxito	48
5.2. Planificación y obstáculos previstos	48
5.2.1. Recursos y contexto de ejecución	48
5.2.2. Obstáculos previstos y estrategias de mitigación	48
5.2.3. Riesgos fuera de control	49
5.3. Ejecución de pruebas y hallazgos	49
5.4. Pruebas manuales durante el desarrollo	50
6. Resultados y Análisis	50
6.1. Resultados obtenidos	50

6.2. Análisis de resultados	51
7. Discusión y tendencias	53
7.1. Contrastación con el estado del arte	53
7.2. Alineación con tendencias emergentes	54
7.2.1. Salud digital y entrenamiento basado en datos	54
7.2.2. Modelo de fitness híbrido	54
7.2.3. Preparación para ecosistemas conectados	54
7.3. Impacto del proyecto	55
7.3.1. Impacto económico y productividad	55
7.3.2. Impacto ambiental y eficiencia tecnológica	55
7.3.3. Implicancias éticas y protección de datos	56
7.4. Limitaciones del estudio	56
8. Conclusiones y trabajo futuro	57
8.1. Conclusiones generales	57
8.2. Aprendizaje autónomo y desafíos técnicos	58
8.3. Trabajo futuro	58
9. Bibliografía	59
10. Anexos	60

1. Introducción

1.1. Antecedentes Generales del Problema

En la actualidad, la creciente demanda de gimnasios y centros de entrenamiento ha generado la necesidad de ofrecer atención personalizada a los clientes. La diversidad de estados físicos iniciales, la presencia de lesiones o enfermedades previas, y la variedad de objetivos de entrenamiento requieren un enfoque individualizado que no siempre es posible de garantizar con los métodos tradicionales.

Uno de los principales desafíos radica en la asignación de rutinas de ejercicio. Si bien muchos centros de entrenamiento utilizan rutinas estandarizadas para diferentes clientes, estas pueden no ser óptimas para todos, lo que podría derivar en falta de progreso, desmotivación o incluso en el riesgo de lesiones por sobrecarga o ejecución inadecuada. Además, la alta demanda de atención simultánea a múltiples clientes dificulta el seguimiento detallado de cada persona por parte de los entrenadores, lo que compromete la corrección y adaptación de los ejercicios según la evolución de cada usuario.

Otro problema relevante es la falta de control efectivo sobre las rutinas ya realizadas. Sin una gestión adecuada, los clientes pueden repetir ejercicios de forma no deseada, omitir algunos que resultan esenciales o sobrecargarse innecesariamente, aumentando el riesgo de fatiga o lesión[1]. Asimismo, el seguimiento a lo largo del tiempo se vuelve complejo, dificultando la adaptación progresiva de los entrenamientos según el avance del usuario o la aparición de dificultades como lesiones o interrupciones en la actividad física.

Si bien existen soluciones tecnológicas orientadas a mejorar estos procesos, la mayoría se enfocan en plataformas de escritorio, que no se ajustan a la dinámica ágil de un centro de entrenamiento, o en aplicaciones para entrenamientos autónomos, lo que implica riesgos para personas sin la supervisión de un experto. Esto evidencia la necesidad de una solución tecnológica que permita gestionar eficazmente las rutinas de entrenamiento, optimizando la personalización, el control y el seguimiento del progreso de los clientes dentro de un entorno de entrenamiento supervisado.

1.2. Solución Propuesta y Alcances

La solución propuesta consiste en el desarrollo de una aplicación móvil orientada a entrenadores de centros de entrenamiento, cuyo objetivo principal es facilitar la asignación de rutinas personalizadas y organizar la información de los clientes en un entorno dinámico. A diferencia de otras soluciones disponibles en el mercado, el sistema se centra en el rol activo del entrenador durante las sesiones

presenciales, permitiéndole mantener el control del proceso y evitando que la gestión recaiga únicamente en los alumnos.

La propuesta considera la implementación de un modelo de datos flexible que permite almacenar usuarios, rutinas, ejercicios y sesiones realizadas, asegurando la trazabilidad del progreso de cada alumno. Asimismo, incluye una interfaz gráfica sencilla y enfocada en la usabilidad, que permite al entrenador gestionar alumnos, asignar rutinas y consultar sesiones en curso de manera rápida y eficaz. El alcance de este trabajo se limita al desarrollo de un prototipo funcional que responda a las necesidades de un centro de entrenamiento real, dejando fuera de esta etapa la integración con sistemas de pago, plataformas externas o módulos avanzados de análisis, los cuales se consideran posibles líneas de extensión en trabajos futuros.

1.3. Objetivo General

Optimizar la gestión y el análisis del entrenamiento en centros deportivos, mediante el desarrollo de un sistema software que comprenda: una interfaz gráfica eficiente para dispositivos móviles; la definición de representaciones visuales que faciliten el seguimiento del progreso de los clientes; y la implementación de un modelo de datos flexible que permita la asignación y consulta estructurada de rutinas de ejercicio en distintos tipos de entrenamiento.

1.4. Objetivos Específicos

1. Proponer un modelo conceptual de datos para especificar, almacenar y consultar rutinas de ejercicios, que soporte eficientemente la asignación de rutinas a clientes y la consulta de su progreso, de un modo extensible que considere, si se requiere, la especificación de distintos tipos o centros de entrenamiento (pesas, crossfit, aeróbico).
2. Generar una propuesta de interfaz gráfica de usuario para dispositivos móviles, que favorezca la usabilidad en el despliegue de información personalizada y la asignación de rutinas de ejercicio, sin interrumpir la dinámica de un centro de entrenamiento.
3. Definir un conjunto de gráficos relevantes para facilitar el análisis del progreso del entrenamiento de clientes en el tiempo, comparando objetivos iniciales y estado actual, en base a rutinas realizadas y eventuales incidencias en el periodo consultado.

1.5. Metodología

Se efectuó un estudio de caso en un centro de entrenamiento real, específicamente en el HU ATHLETICS, con el fin de modelar el negocio y levantar los requerimientos. De forma paralela, se analizaron aplicaciones existentes que brindan apoyo al entrenamiento personal y al seguimiento de los avances a lo largo del tiempo.

Para la propuesta del modelo, se analizaron dichos requerimientos junto con los de otros centros de entrenamiento. Se procedió a generar un modelo de datos, el cual fue implementado mediante un sistema gestor de bases de datos y un prototipo con operaciones CRUD.

En cuanto a los objetivos restantes, se adoptó una metodología de desarrollo iterativa e incremental, con adaptaciones de propuestas ágiles a un entorno de desarrollo unipersonal, y se contó con la participación de actores relevantes para obtener retroalimentación temprana.

1.6. Estructura del informe

El Capítulo 2 “Marco teórico” presenta los fundamentos conceptuales que sustentan el desarrollo del proyecto. En este capítulo se abordan los conceptos de usabilidad en aplicaciones móviles, visualización de datos en contextos deportivos. Asimismo, se incluye un análisis del estado del arte, donde se revisan soluciones existentes en el mercado y se identifican sus limitaciones frente al escenario de entrenamiento presencial centrado en el rol activo del entrenador.

El Capítulo 3 “Descripción de la solución” describe la propuesta desarrollada. Se presenta el contexto de aplicación en el centro de entrenamiento HU Athletics, el alcance del sistema y los requerimientos identificados. Además, se expone la arquitectura general de la solución y el modelo de datos que soporta la planificación, ejecución y registro de sesiones de entrenamiento.

El Capítulo 4 “Implementación y operación” detalla el desarrollo de la solución desde una perspectiva técnica. Se describen las tecnologías y herramientas utilizadas, las funcionalidades clave implementadas en la aplicación móvil y la integración entre el frontend y el backend. También se presentan las condiciones de operación del sistema, considerando los requisitos de uso en el entorno real del centro de entrenamiento.

El Capítulo 5 “Diseño y ejecución de pruebas” expone el proceso de evaluación realizado sobre la aplicación móvil. En este capítulo se describe el diseño experimental, las métricas utilizadas para evaluar

la usabilidad, la planificación de las pruebas y su ejecución con un usuario real. Se documentan los principales hallazgos obtenidos durante la evaluación.

El Capítulo 6 “Resultados y análisis” presenta los resultados derivados de la evaluación de usabilidad. Se muestran los valores obtenidos a partir de las métricas aplicadas y se realiza un análisis de estos resultados en relación con los objetivos planteados para el proyecto, identificando fortalezas y oportunidades de mejora del sistema.

El Capítulo 7 “Discusión y tendencias” contrasta la solución propuesta con el estado del arte revisado previamente, analiza su alineación con tendencias actuales relevantes para la ingeniería informática aplicada al deporte e integra el análisis del impacto del proyecto con la discusión académica de la solución desarrollada. Además, se exponen las principales limitaciones del estudio y se delimitan los alcances de los resultados obtenidos.

El Capítulo 8 “Conclusiones” presenta las conclusiones finales del trabajo. En este capítulo se resume el cumplimiento de los objetivos, se reflexiona sobre los aprendizajes técnicos y metodológicos adquiridos durante el desarrollo del proyecto y se proponen líneas de trabajo futuro coherentes con la solución implementada.

El Capítulo 9 “Bibliografía” reúne las referencias utilizadas a lo largo del informe.

Finalmente, el Capítulo 10 “Anexos” incluye material complementario que respalda el desarrollo del proyecto.

2. Marco Teórico

2.1. Usabilidad en aplicaciones móviles

La usabilidad constituye un factor crítico en el diseño de sistemas de software orientados a la interacción humana. De acuerdo con la norma ISO 9241-11, se define como el grado en que un producto puede ser utilizado por usuarios específicos para lograr objetivos determinados con efectividad, eficiencia y satisfacción en un contexto de uso concreto [2].

En aplicaciones móviles destinadas a la gestión de rutinas de entrenamiento, la usabilidad debe responder a escenarios de uso dinámicos, en los que el entrenador requiere realizar tareas con rapidez y bajo nivel de complejidad cognitiva. Estudios en usabilidad móvil señalan que interfaces diseñadas para interacciones

breves y recurrentes reducen errores y tiempos de operación en contextos con presión temporal y movilidad constante [3]. Principios como la claridad visual, la consistencia en los flujos de interacción y la prevención de errores resultan especialmente relevantes para evitar interrupciones en la dinámica del entrenamiento presencial [3].

El diseño también debe contemplar métricas objetivas y subjetivas de evaluación. Instrumentos estandarizados como el System Usability Scale (SUS) permiten medir la percepción global de usabilidad de forma sistemática y comparable entre sistemas, incluso en evaluaciones con muestras reducidas de usuarios expertos [4].

2.2. Visualización de datos en el contexto deportivo

La visualización de datos transforma información en representaciones gráficas que apoyan la comprensión y la toma de decisiones. En el ámbito deportivo, estas representaciones permiten al entrenador revisar el estado de cada alumno durante la sesión y evaluar su progreso en el tiempo sin revisar registros extensos.

Un ejemplo frecuente corresponde a los gráficos de línea y series temporales. Estos gráficos muestran la evolución de una variable a lo largo del tiempo, como el aumento progresivo de carga en un ejercicio de fuerza o la variación en el número de repeticiones completadas por sesión. Un entrenador puede identificar estancamientos cuando una línea se mantiene plana durante varias semanas o detectar sobrecarga cuando el aumento es abrupto entre sesiones consecutivas [5].

Los diagramas comparativos permiten contrastar metas iniciales con resultados obtenidos. Por ejemplo, un gráfico de barras puede mostrar el peso objetivo definido para un alumno frente al peso real alcanzado después de un ciclo de entrenamiento. Esta comparación facilita decidir si el objetivo se mantiene, se ajusta o se reemplaza por uno nuevo según el desempeño observado [6].

Los paneles de control agrupan varios indicadores en una sola vista. En un centro de entrenamiento, un panel puede mostrar cuántos alumnos entrenan en ese momento, qué rutina ejecuta cada uno y el estado de avance de la sesión. Este tipo de visualización apoya al entrenador cuando supervisa a varios alumnos en paralelo y necesita priorizar su atención sin interrumpir el flujo de la clase [6].

Como se ejemplifica en la Figura 1, el uso de indicadores visuales permite identificar rápidamente el estado de cada alumno durante la sesión, facilitando la toma de decisiones en tiempo real.

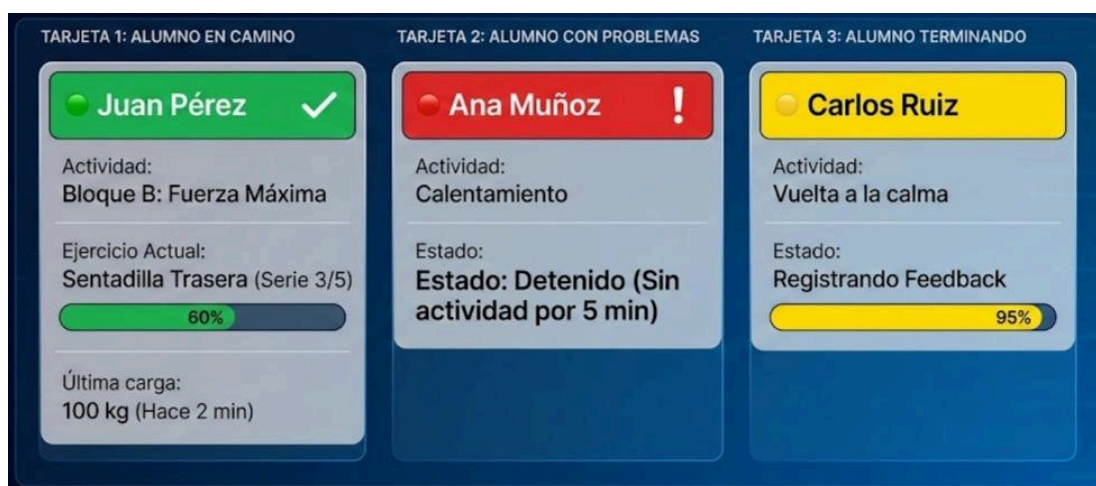


Figura 1. Prototipo de visualización del panel de control en tiempo real. El uso de indicadores de estado basados en colores (semáforo) y barras de progreso permite al entrenador identificar en segundos situaciones anómalas, como un alumno detenido por más tiempo del previsto (tarjeta roja), optimizando la gestión de la atención en sala.

El valor de la visualización no está en mostrar grandes volúmenes de datos, sino en presentar información clara y directamente relacionada con decisiones concretas. Si un gráfico no ayuda a ajustar una carga, modificar una rutina o detectar un problema durante la sesión, pierde utilidad para el entrenador [5].

2.3. Estado del arte en soluciones de gestión de entrenamiento

El análisis comparativo se centra en plataformas ampliamente usadas para la gestión del entrenamiento físico: Trainerize [7], TrueCoach [8], TrainingPeaks [9] y StrengthLog [10]. Estas soluciones se revisan a partir de su documentación oficial y del uso de sus versiones gratuitas, con el objetivo de evaluar su adecuación al trabajo presencial del entrenador en centros de entrenamiento.

Los criterios de comparación se definen según las necesidades identificadas en el problema abordado. Se consideran el rol del entrenador en sesiones presenciales, el enfoque principal de la aplicación, el registro de la ejecución real de la sesión, el soporte para trabajo simultáneo con varios alumnos, el costo por licencia y la escalabilidad según cantidad de clientes.

Solución	Rol del entrenador presencial	Enfoque principal	Registro de ejecución real	Soporte multiusuario presencial	Costo y escalabilidad

Trainerize [7]	Bajo. El entrenador planifica rutinas, pero no gestiona la sesión en tiempo real.	Entrenamiento remoto y seguimiento individual del alumno.	Limitado a reportes ingresados por el alumno.	No considera supervisión simultánea en sala.	Modelo de suscripción cuyo costo aumenta según la cantidad de clientes.
TrueCoach [8]	Bajo. El sistema se orienta al envío de planes y revisión posterior.	Coaching remoto y comunicación entrenador-alumno.	Basado en autoinforme del alumno.	No diseñado para uso presencial con múltiples alumnos activos.	Pago mensual por entrenador con límites en la cantidad de clientes.
TrainingPeaks [9]	Bajo. El entrenador analiza datos, pero no gestiona la sesión presencial.	Planificación y análisis de rendimiento, especialmente en deportes de resistencia.	Integrado a dispositivos externos y reportes del usuario.	No orientado a sesiones presenciales simultáneas.	Licencias diferenciadas para entrenadores y atletas, con costos elevados para centros pequeños.
StrengthLog [10]	Nulo. La aplicación se centra en el uso individual.	Registro personal de entrenamientos de fuerza.	Registro manual realizado por el usuario.	No existe gestión de múltiples alumnos.	Modelo freemium enfocado en usuarios individuales.

Tabla 1. Tabla de comparación de aplicaciones existentes

Del análisis se observa que todas las soluciones priorizan el trabajo remoto o el uso autónomo por parte del alumno. El entrenador cumple un rol secundario y no dispone de herramientas para gestionar la sesión presencial en tiempo real. El registro de la ejecución depende del alumno o de dispositivos externos, lo que limita el control directo durante el entrenamiento[7]–[10].

Ninguna de las plataformas analizadas está diseñada para supervisar a varios alumnos entrenando en paralelo dentro de un mismo espacio físico. Tampoco ofrecen una vista operativa que permita al entrenador conocer el estado de avance de cada sesión mientras se desarrolla la clase. A esto se suma que los modelos de licenciamiento dificultan su adopción en centros pequeños o medianos, ya que los costos aumentan con la cantidad de clientes gestionados.

Este análisis justifica el desarrollo de la solución propuesta. El sistema planteado aborda un escenario no cubierto por las aplicaciones existentes, centrado en la gestión del entrenamiento presencial, el rol activo del entrenador y el control de la ejecución real de las sesiones en sala. Esta diferencia define el aporte específico del proyecto frente a las alternativas disponibles en el mercado.

2.4. Síntesis crítica

El estado del arte muestra una oferta amplia de aplicaciones para entrenamiento físico, con foco en planificación remota, registro individual y análisis posterior. Estas soluciones resuelven necesidades del alumno y del entrenador en contextos no presenciales, pero no abordan la gestión operativa del entrenamiento en sala. La brecha no se relaciona con la ausencia de herramientas, sino con la falta de sistemas diseñados para apoyar al entrenador durante la sesión presencial[7]–[10].

Las plataformas analizadas reducen el rol del entrenador a la definición de planes y a la revisión diferida de resultados. No entregan soporte para supervisar a varios alumnos en paralelo ni para acceder a información clave mientras la sesión se desarrolla. El registro de la ejecución queda en manos del alumno o de dispositivos externos, lo que limita el control directo y oportuno del entrenamiento.

La usabilidad y la visualización de datos resultan determinantes en este escenario. En un entorno presencial, el entrenador necesita consultar información en segundos, interpretar estados de avance y decidir ajustes sin interrumpir la dinámica de la clase. Las soluciones existentes no priorizan este tipo de interacción ni ofrecen vistas operativas orientadas al trabajo en sala.

Esta revisión crítica establece una brecha clara entre las capacidades de las herramientas disponibles y las necesidades del entrenamiento presencial supervisado. La ausencia de sistemas centrados en la gestión en

tiempo real, con énfasis en usabilidad y visualización aplicada, justifica el desarrollo de la solución propuesta y define su aporte frente al estado actual del mercado.

3. Descripción de la Solución

3.1. Contexto de aplicación

El desarrollo de la solución surge a partir del análisis de un problema real observado en el funcionamiento cotidiano del centro de entrenamiento funcional HU ATHLETICS. Antes de la incorporación del sistema, el trabajo del entrenador se realizaba sin apoyo computacional, sin un registro estructurado de los entrenamientos y sin un historial persistente por alumno. La información asociada a las sesiones realizadas dependía exclusivamente de la memoria del entrenador y, en algunos casos, de anotaciones informales en cuadernos, lo que dificultaba la trazabilidad del progreso individual.

Este contexto generaba una pérdida progresiva de personalización del entrenamiento, aspecto que constituye uno de los principales factores diferenciadores del centro. A medida que aumentaba la cantidad de alumnos atendidos de forma simultánea, especialmente en horarios de alta demanda, el seguimiento individual se volvía más complejo y la planificación tendía a apoyarse en esquemas generales, reduciendo la capacidad de adaptar el entrenamiento según el avance, las interrupciones o las necesidades específicas de cada cliente.

La complejidad del problema identificado no se relacionaba únicamente con la ausencia de una herramienta tecnológica, sino con la falta de una estructura formal para organizar la información del entrenamiento. El desorden operativo observado impedía mantener continuidad entre sesiones, dificultaba la recuperación de antecedentes relevantes y aumentaba la carga cognitiva del entrenador durante la supervisión en sala.

Este escenario evidenció la necesidad de una solución que permitiera ordenar la información, preservar el historial de cada alumno y apoyar la toma de decisiones del entrenador en el contexto del entrenamiento presencial, sin interferir en la dinámica propia de la sesión.

3.1.1 Levantamiento de requerimientos

El levantamiento del problema y de los requerimientos se realizó mediante observación directa del funcionamiento del centro, conversaciones informales y continuas con el entrenador y dueño del

gimnasio, y participación activa del autor en sesiones de entrenamiento. Esta aproximación permitió comprender el flujo real de trabajo, la atención simultánea a varios alumnos y las limitaciones prácticas del registro manual utilizado hasta ese momento.

La participación directa en sesiones de entrenamiento permitió al autor experimentar el proceso desde la perspectiva del usuario final, identificando puntos críticos relacionados con la planificación, la supervisión en tiempo real y el seguimiento del progreso individual. Este acercamiento facilitó una comprensión profunda del dominio del problema y de las necesidades reales que debía cubrir la solución.

Durante el desarrollo del proyecto, el contacto con el entrenador se mantuvo de forma permanente, permitiendo ajustar y complementar los requerimientos a medida que surgían nuevas necesidades no identificadas inicialmente. Este proceso iterativo de levantamiento y validación aseguró que la solución respondiera al contexto real de uso y no a supuestos teóricos desvinculados de la operación diaria.

A partir de este análisis, se identificaron como problemáticas centrales la pérdida de seguimiento individual, la dependencia excesiva de la memoria del entrenador y la falta de una estructura formal para la planificación y el registro de las sesiones. Estas observaciones dieron origen a la definición del producto, al levantamiento de los requerimientos funcionales y no funcionales, al diseño del modelo de datos relacional y a la posterior implementación de la solución propuesta.

3.2. Requerimientos del sistema

Los requerimientos del sistema se definen considerando los dos componentes de la solución. La aplicación móvil orientada a la gestión del entrenamiento presencial y el dashboard web orientado a la visualización de información histórica. Ambos comparten la misma fuente de datos, pero cumplen funciones distintas.

3.2.1. Requerimientos funcionales

Código	Requerimiento funcional
RF-01	El sistema debe permitir registrar clientes con datos básicos como nombre, edad y observaciones.
RF-02	El sistema debe permitir visualizar un listado de clientes mediante desplazamiento y búsqueda por nombre.

RF-03	El sistema debe permitir crear sesiones de entrenamiento asociadas a un cliente y a una fecha específica.
RF-04	El sistema debe permitir planificar una sesión desde cero, incorporando una etapa de calentamiento y dos etapas generales.
RF-05	El sistema debe permitir agregar ejercicios a cada etapa, seleccionándolos desde un catálogo categorizado por tipo.
RF-06	El sistema debe permitir asignar a cada ejercicio su unidad de medida y valores como repeticiones, tiempo o peso.
RF-07	El sistema debe permitir editar una sesión planificada antes de su finalización.
RF-08	El sistema debe permitir repetir una sesión previamente realizada y reutilizar su estructura como base de una nueva sesión.
RF-09	El sistema debe permitir compartir una sesión planificada entre distintos clientes.
RF-10	El sistema debe mostrar en la vista de inicio a los clientes con sesiones activas y sus ejercicios organizados por etapa.
RF-11	El sistema debe permitir marcar una sesión como finalizada e indicar si fue completada o no.
RF-12	El sistema debe exigir una observación obligatoria cuando una sesión no se completa.
RF-13	El sistema debe registrar la duración y observaciones de cada sesión finalizada.
RF-14	El sistema debe permitir visualizar el historial completo de sesiones por cliente.
RF-15	El sistema debe permitir visualizar el historial mensual de sesiones mediante una vista de calendario.
RF-16	El sistema debe permitir que un dashboard web consulte los datos históricos almacenados para apoyar la visualización del entrenamiento y la asistencia.

Tabla 2. Tabla de requerimientos funcionales.

3.2.2. Requerimientos no funcionales

Código	Requerimiento no funcional
RNF-01	El sistema debe operar bajo el supuesto de conexión a internet activa durante su uso.
RNF-02	El tiempo de carga del listado de clientes y de la vista de inicio de la aplicación móvil no debe exceder los 3 segundos bajo condiciones normales de conectividad.
RNF-03	El sistema debe ejecutarse en dispositivos Android con versión mínima 5.0 y funcionar correctamente en versiones recientes del sistema operativo.
RNF-04	La interfaz de la aplicación móvil debe adaptarse al uso en teléfonos y tablets.
RNF-05	El sistema debe mantener la persistencia de datos en una base de datos centralizada para evitar la pérdida de información ante fallas del dispositivo.
RNF-06	El sistema debe permitir aprender el flujo básico de uso sin necesidad de capacitación formal.
RNF-07	El sistema debe minimizar la cantidad de acciones necesarias para la ejecución de tareas críticas durante la sesión presencial.
RNF-08	El sistema debe gestionar imágenes de perfil mediante almacenamiento remoto.
RNF-09	El dashboard web debe operar como un componente independiente, sin afectar el funcionamiento de la aplicación móvil.

Tabla 3. Tabla de requerimientos no funcionales.

3.2.3. Reglas de negocio

Código	Regla de negocio
RN-01	Cada cliente se identifica mediante un identificador único generado por la base de datos.
RN-02	Una sesión nueva debe contener obligatoriamente una etapa de calentamiento y dos etapas generales.

RN-03	El orden de las etapas y de los ejercicios dentro de una etapa debe mantenerse de forma única dentro de cada sesión.
RN-04	Una sesión finalizada no puede modificar su fecha.
RN-05	Si una sesión no se completa, el sistema debe exigir una observación para permitir su cierre.
RN-06	El sistema debe conservar el historial completo de sesiones realizadas por cada cliente.

Tabla 4. Tabla de reglas del negocio

3.2.4. Supuestos y restricciones

La solución no incorpora autenticación interna de usuarios. El control de acceso se delega a la seguridad del dispositivo utilizado, dado que el sistema está diseñado para un usuario único en el contexto operativo actual. No obstante, la arquitectura permite su adaptación a un escenario con múltiples usuarios, como asistentes de entrenamiento, mediante la incorporación de mecanismos de autenticación y control de roles. La versión evaluada no incluye funcionamiento sin conexión ni sincronización diferida. El dashboard web no reemplaza la aplicación móvil ni se utiliza durante el entrenamiento presencial. El sistema no almacena datos sensibles, limitándose a información básica de gestión.

Estos requerimientos establecen el marco funcional y técnico sobre el cual se diseña la arquitectura del sistema y se implementa el modelo de datos presentado en las secciones siguientes.

3.3. Arquitectura del sistema

La solución se basa en una arquitectura cliente servidor compuesta por tres componentes principales. Una aplicación móvil para la operación diaria en el centro de entrenamiento, un backend en la nube para la gestión y persistencia de datos, y un dashboard web independiente para la visualización y análisis de información histórica. Todos los componentes comparten una misma base de datos centralizada.

Como se muestra en la Figura 2, la arquitectura del sistema separa claramente las responsabilidades entre la capa de presentación, la capa de servicios en la nube y los componentes de análisis, permitiendo una interacción controlada entre los distintos módulos y facilitando la escalabilidad y el mantenimiento de la solución.

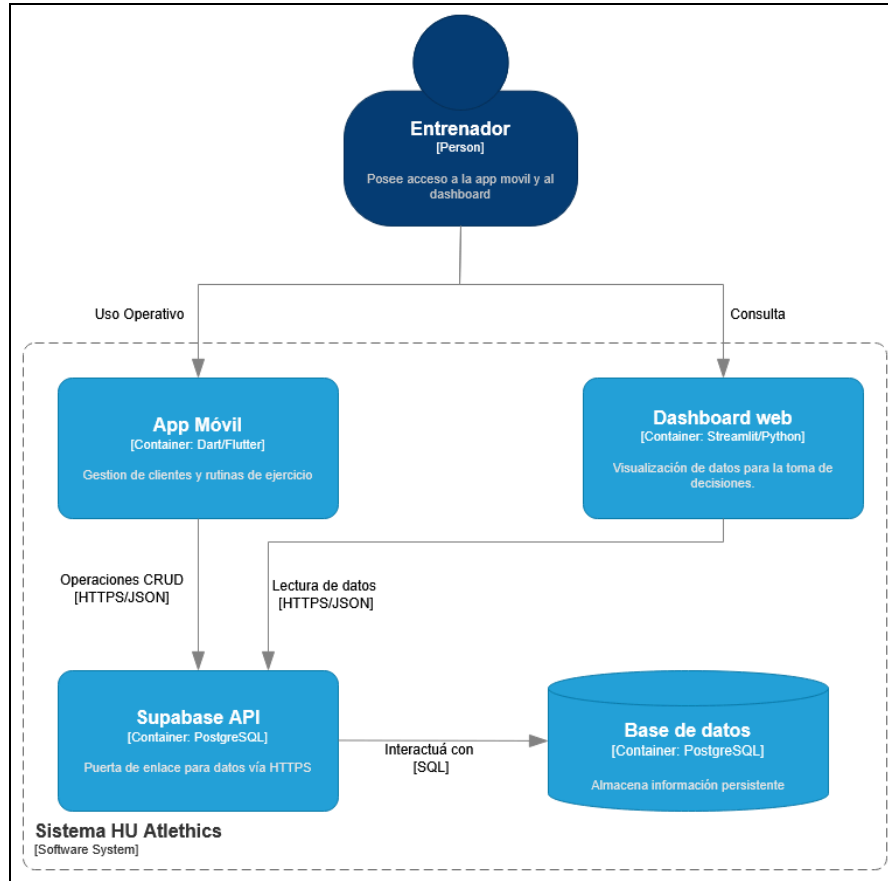


Figura 2. Modelo arquitectónico C4.

3.3.1. Arquitectura por capas de la aplicación móvil

La aplicación móvil se estructura bajo un patrón de arquitectura por capas (*Layered Architecture*), con el objetivo de separar responsabilidades y controlar la complejidad del sistema. Esta decisión permite desacoplar la interfaz de usuario de la lógica de negocio y de la infraestructura externa, facilitando la mantenibilidad, comprensión y evolución futura de la solución.

Como se observa en la Figura 3, la arquitectura se organiza en tres capas principales: presentación, lógica y datos, e infraestructura externa. Cada una cumple un rol específico dentro del sistema, permitiendo que los cambios en una capa no impacten directamente en las demás y asegurando una estructura modular y escalable.

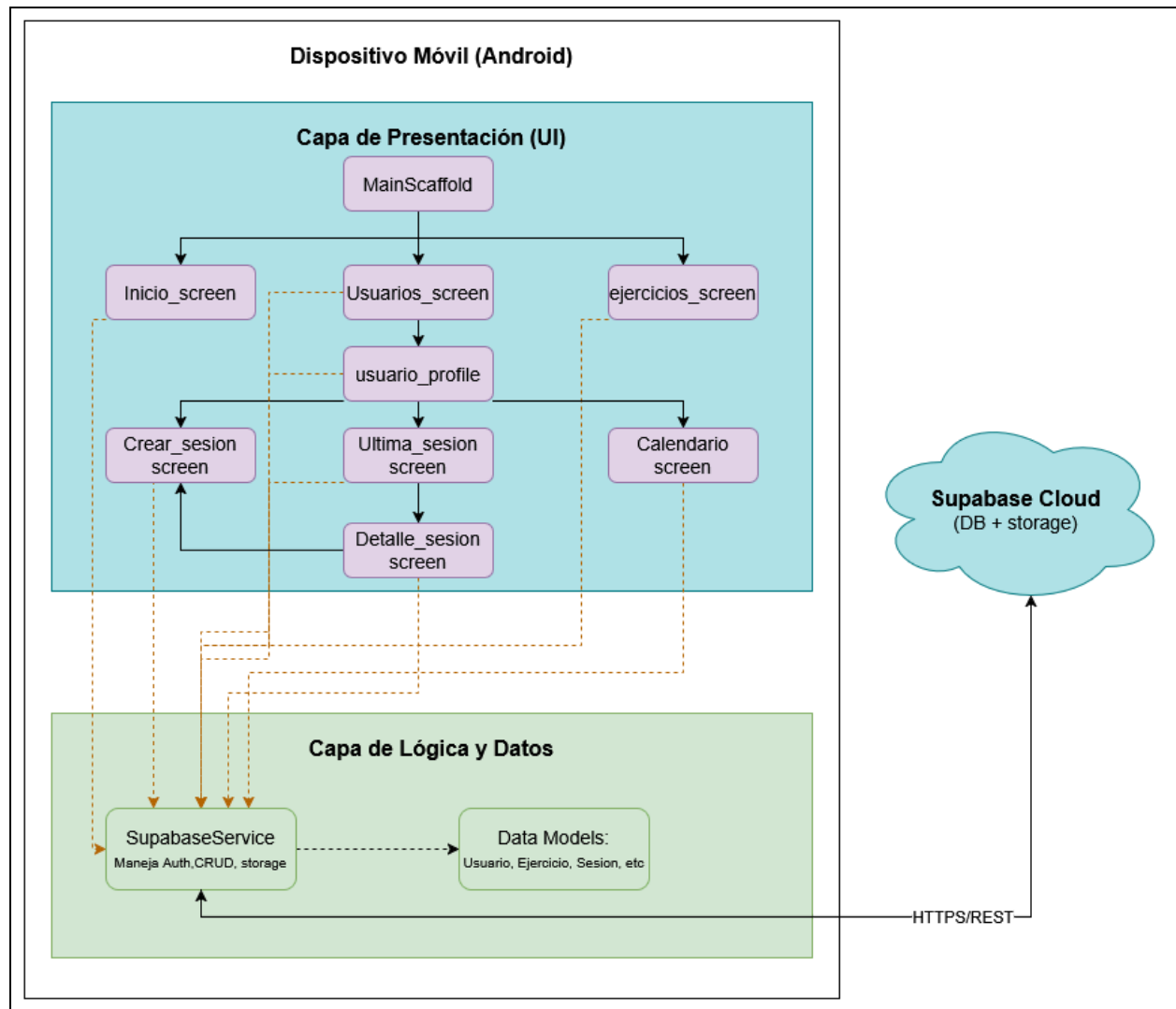


Figura 3. Diagrama de arquitectura por capas de la aplicación móvil.

3.3.1.1. Capa de Presentación

La capa de presentación agrupa todas las pantallas y componentes visuales de la aplicación móvil. Su responsabilidad es el renderizado de la interfaz gráfica y la captura de las interacciones del entrenador durante la operación en sala, incluyendo la navegación entre vistas como el listado de clientes, el perfil del cliente y las pantallas asociadas a la gestión de sesiones.

Esta capa no accede directamente a la base de datos ni ejecuta operaciones de persistencia. Se limita al manejo del estado efímero de la aplicación, como navegación y validaciones visuales, delegando cualquier interacción con datos persistentes a la capa de lógica y datos. Esta separación evita dependencias innecesarias entre la interfaz y la infraestructura.

3.3.1.2. Capa de Lógica y Datos

La capa de lógica y datos actúa como intermediaria entre la interfaz y el backend. En ella se centraliza la comunicación con Supabase mediante un servicio único que encapsula las operaciones de autenticación, consultas y almacenamiento, evitando la duplicación de lógica en las distintas pantallas.

Asimismo, esta capa incorpora los modelos de datos que representan las tablas de la base de datos relacional. Estos modelos permiten transformar respuestas crudas en objetos tipados, mejorando la seguridad de tipos y facilitando el mantenimiento. La centralización de la lógica responde al principio *Don't Repeat Yourself* y simplifica futuras modificaciones del sistema.

3.3.1.3. Capa de Infraestructura Externa

La capa de infraestructura externa corresponde a los servicios en la nube proporcionados por Supabase. En ella se gestionan la persistencia de los datos en PostgreSQL, el almacenamiento de archivos y los mecanismos de seguridad del sistema, utilizando comunicaciones seguras sobre HTTPS.

El uso de una plataforma Backend-as-a-Service permite delegar la disponibilidad, el respaldo de la información y el control de accesos a la infraestructura. Esto reduce la complejidad operativa del proyecto y garantiza la protección de los datos ante fallos del dispositivo local o accesos no autorizados.

3.3.1.4. Justificación de la arquitectura propuesta

La adopción de una arquitectura por capas responde a la necesidad de ordenar la complejidad del sistema y asegurar su mantenibilidad. La separación entre presentación, lógica y persistencia permite modificar la interfaz, ajustar el modelo de datos o extender la infraestructura sin afectar al resto del sistema.

Esta estructura establece una base técnica coherente y escalable, alineada con buenas prácticas de ingeniería de software. La arquitectura definida no solo satisface los requerimientos actuales del proyecto, sino que prepara la solución para futuras extensiones funcionales y tecnológicas.

3.3.2. Aplicación móvil

La aplicación móvil se desarrolla en Flutter y se ejecuta en dispositivos Android. Su propósito es apoyar la gestión del entrenamiento presencial. La aplicación concentra las funciones críticas para el entrenador durante la sesión.

Los módulos principales corresponden a la vista de inicio, la gestión de clientes, el catálogo de ejercicios, la planificación y edición de sesiones, y la visualización del historial de entrenamiento. La vista de inicio muestra a los alumnos con sesiones activas y sus rutinas organizadas por etapas. Desde esta vista el entrenador supervisa el trabajo en sala.

La gestión de clientes permite registrar, editar y consultar información básica. El catálogo de ejercicios se organiza por tipos definidos por el entrenador y se utiliza durante la planificación de sesiones. La planificación permite crear sesiones nuevas, repetir sesiones previas o compartir una sesión con otro alumno. El historial se presenta como un listado cronológico de sesiones y como una vista de calendario para revisar la asistencia mensual.

La aplicación se comunica con el backend mediante el SDK `supabase_flutter`. A través de esta capa se realizan operaciones de creación, lectura, actualización y eliminación de datos, además del almacenamiento de imágenes de perfil en el servicio de almacenamiento remoto.

3.3.3. Backend en la nube

El backend se implementa utilizando Supabase como plataforma Backend as a Service. La base de datos utiliza PostgreSQL como sistema gestor y centraliza toda la información de clientes, sesiones, ejercicios y registros de entrenamiento. El almacenamiento de imágenes de perfil se realiza mediante Supabase Storage.

El acceso a los datos se protege mediante políticas de seguridad a nivel de fila. Estas políticas restringen el acceso a la información según el contexto de uso del sistema y evitan exposiciones no deseadas. La persistencia centralizada permitió mantener la consistencia de datos y evitar pérdidas ante fallas del dispositivo móvil.

El proyecto se aloja en Supabase Cloud, utilizando la región `us east 1`. Esta infraestructura soporta la aplicación móvil y el dashboard web sin requerir servidores propios.

3.3.4. Dashboard web

El dashboard web se desarrolla de forma independiente utilizando Streamlit y Python. Su propósito es apoyar la visualización y el análisis de la información histórica almacenada en el backend. No se utiliza durante la sesión presencial y no reemplaza a la aplicación móvil.

El dashboard permite visualizar la evolución semanal de las sesiones de entrenamiento, calcular el porcentaje de adherencia y cumplimiento de los clientes, identificar los ejercicios más utilizados mediante rankings de frecuencia y monitorear la duración promedio de las sesiones. También incorpora funciones de búsqueda y filtrado por cliente y rango de fechas.

El acceso al dashboard está restringido mediante una URL privada y su uso está destinado al dueño y futuros entrenadores del centro. La conexión a la base de datos se realiza mediante la librería supabase-py y se limita a operaciones de lectura bajo demanda. El dashboard no modifica datos ni interviene en la operación diaria del sistema.

3.3.5. Flujo de datos y conectividad

La aplicación móvil y el dashboard web se comunican directamente con el backend en Supabase. La aplicación móvil realiza operaciones de lectura y escritura durante la planificación y cierre de sesiones. El dashboard consulta los datos cuando el usuario solicita una visualización específica.

El sistema asume conexión a internet activa. En ausencia de conectividad, la aplicación no puede operar y notifica el fallo al usuario. No se implementa almacenamiento local ni sincronización diferida en la versión evaluada.

3.3.6. Decisiones de diseño

La separación entre la aplicación móvil y el dashboard web responde a una decisión técnica y académica. La aplicación móvil se enfoca en la operación en sala y en tareas que requieren rapidez y baja carga cognitiva. El dashboard se orienta al análisis posterior y a la visualización agregada de datos.

Esta arquitectura permite cubrir distintas necesidades del centro de entrenamiento sin sobrecargar la aplicación móvil. También permite demostrar competencias en desarrollo móvil, backend en la nube y visualización de datos en un entorno web, manteniendo una base de datos única y coherente para toda la solución.

3.4. Modelo de datos

El componente de persistencia del sistema se implementa sobre un motor de base de datos relacional PostgreSQL, orquestado a través de la plataforma Supabase. El esquema lógico, representado en la Figura 4, consta de ocho entidades normalizadas diseñadas para segregar explícitamente los dominios de planificación (prescripción del entrenamiento) y ejecución (registro empírico de la sesión).

El modelo se organiza en torno a la relación cliente, que representa a cada alumno del centro. Un cliente puede tener múltiples sesiones asociadas, lo que permite registrar todo su historial de entrenamiento. Esta relación respalda el seguimiento longitudinal y evita la pérdida de información entre ciclos de trabajo.

La planificación del entrenamiento se representa mediante la relación sesion. Cada sesión define una estructura de trabajo prevista para una fecha determinada. Para descomponer la sesión en partes lógicas, el modelo incorpora la relación sesion_etapa, que vincula una sesión con una o más etapas y mantiene el orden de ejecución. Esta decisión permite modelar sesiones con calentamiento, bloques principales y cierres sin duplicar información.

La relación etapa describe cada bloque de trabajo dentro de la sesión. A su vez, etapa_ejercicio define los ejercicios asociados a una etapa específica, junto con parámetros como orden y cantidades. Esta estructura permite reutilizar ejercicios del catálogo y configurar distintas combinaciones según el objetivo del entrenamiento.

El catálogo de ejercicios se gestiona mediante las relaciones tipo y ejercicio. Tipo clasifica los ejercicios según su naturaleza, como fuerza o cardio, mientras que ejercicio almacena la información base de cada actividad disponible. Esta separación mantiene el catálogo independiente de las sesiones y facilita su extensión sin afectar el historial de entrenamiento.

La ejecución real del entrenamiento se registra en registro_sesion. Esta relación almacena el resultado de cada sesión realizada por el cliente, incluyendo el estado de cumplimiento y observaciones relevantes. De este modo, el sistema distingue explícitamente entre lo planificado y lo efectivamente realizado, permitiendo análisis posteriores y ajustes informados del entrenamiento.

Este modelo de datos permite al entrenador planificar sesiones estructuradas, registrar la ejecución real y analizar el progreso del alumno en el tiempo. Cada relación responde a una necesidad operativa concreta y se integra de forma coherente con la arquitectura y las funcionalidades implementadas en el sistema.

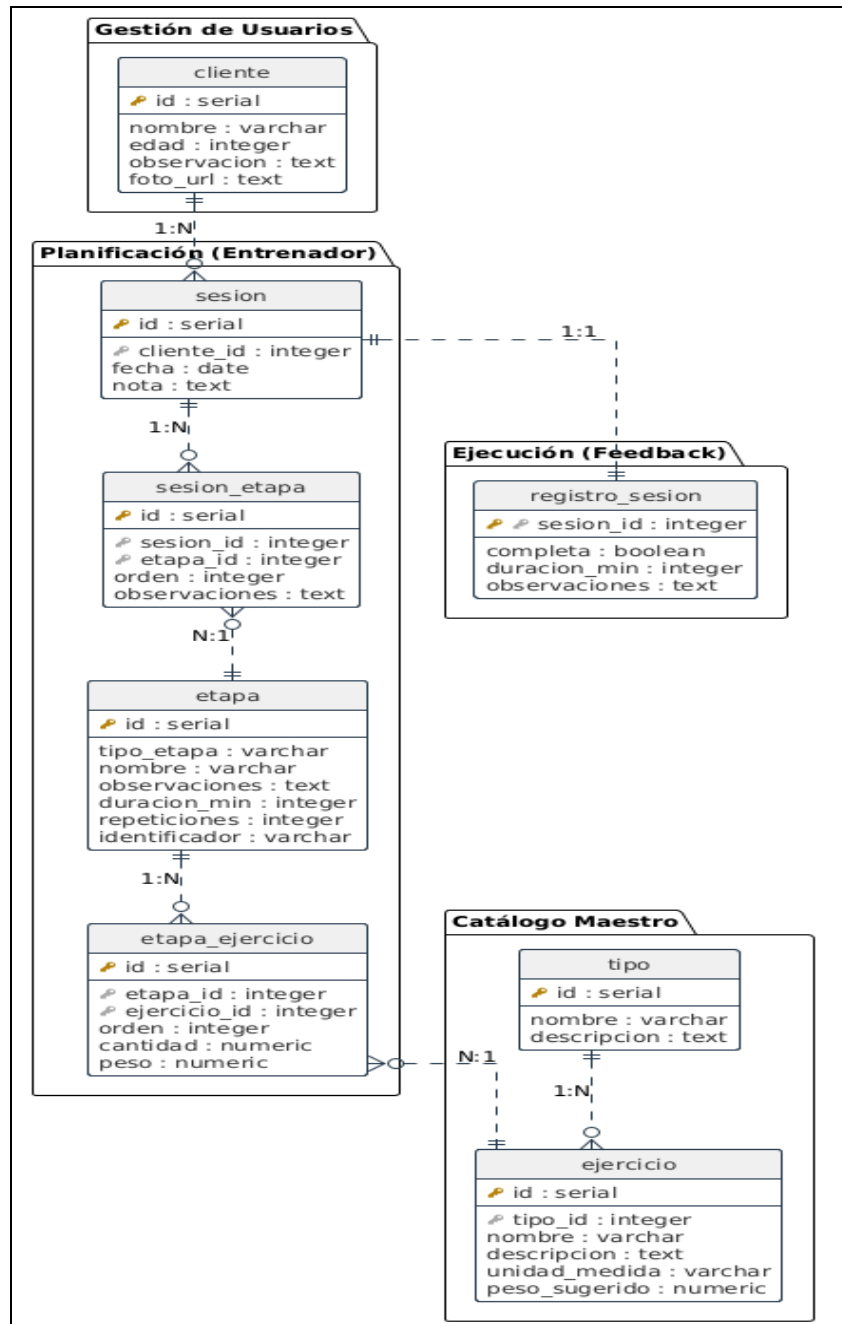


Figura 4. Esquema relacional de la base de datos.

3.4.1. Descripción de Relaciones

El modelo se estructura en torno a cuatro núcleos funcionales:

- Gestión de Usuarios (cliente): Relación central que representa al usuario final del sistema. Además de los atributos de identificación, almacena la referencia (URI) a los recursos

multimedia, como la fotografía de perfil, gestionados externamente en el almacenamiento de objetos (Object Storage), lo que optimiza el rendimiento de las consultas a la base de datos.

- Catálogo de Ejercicios (tipo, ejercicio): Se implementa un patrón de catálogo maestro para garantizar la reutilización y consistencia de los datos.
 - tipo: Categoriza los movimientos (ej. fuerza, cardio) asegurando la unicidad semántica.
 - ejercicio: Define el inventario de movimientos disponibles y estandariza las unidades de medida permitidas, lo cual es crítico para la validación de la entrada de datos.
- Estructura de la Sesión (sesion, sesion_etapa, etapa, etapa_ejercicio): La planificación del entrenamiento se modela mediante un conjunto de relaciones organizadas jerárquicamente. La relación `sesion` actúa como cabecera, mientras que `sesion_etapa` y `etapa_ejercicio` permiten la composición modular de rutinas, estableciendo un orden secuencial estricto mediante atributos de índice (orden). Esta estructura facilita la reutilización de etapas completas en distintas sesiones.
- Registro de Ejecución (registro_sesion): Relación vinculada mediante una correspondencia uno a uno (1:1) con `sesion`. Su creación es condicional a la finalización del entrenamiento y almacena las métricas reales de la ejecución, tales como duración efectiva, estado de completitud y observaciones cualitativas.

3.4.2. Justificación del Diseño: Planificación vs. Ejecución

Una de las decisiones arquitectónicas clave es el desacoplamiento entre la estructura planificada y el registro histórico.

La planificación (`sesion` y sus derivadas) responde al "qué se hará", mientras que la ejecución (`registro_sesion`) responde al "qué ocurrió". Esta separación ofrece dos ventajas técnicas fundamentales:

1. Integridad de Datos: Evita la proliferación de valores nulos (NULL) en la base de datos para sesiones futuras que aún no tienen métricas de ejecución.
2. Calidad del Historial: Permite realizar análisis comparativos precisos entre la prescripción teórica y el desempeño real, sin alterar la estructura original de lo planificado.

3.4.3. Reglas de Integridad y Lógica de Negocio

Para garantizar la consistencia de la información independientemente de la interfaz de usuario, se han implementado restricciones y validaciones directamente en el motor de base de datos:

- Integridad de Dominio: Se utilizan restricciones de control (CHECK constraints) en la relación `ejercicio` para limitar las unidades de medida a un conjunto finito de valores válidos. De igual

forma, el atributo `tipo_etapa` restringe los valores permitidos a 'calentamiento' o 'bloque', preservando la semántica del entrenamiento.

- **Lógica en Base de Datos (Functions & Triggers):** Se han implementado funciones almacenadas para validar reglas de negocio complejas. Por ejemplo, se impide la asignación de cargas de trabajo (repeticiones o peso) en etapas clasificadas como 'calentamiento', asegurando que la prescripción técnica sea coherente.
- **Integridad Referencial:** Se configuran claves foráneas (Foreign Keys) en todas las relaciones jerárquicas para impedir la existencia de registros huérfanos, asegurando que ninguna etapa o ejercicio exista sin su entidad padre correspondiente.

3.4.4. Seguridad y Control de Acceso

El diseño de seguridad adopta un enfoque de "defensa en profundidad". Además de la autenticación vía API Keys proporcionada por el backend, el modelo implementa políticas de seguridad a nivel de fila (Row Level Security - RLS) nativas de PostgreSQL. Estas políticas restringen las operaciones de lectura y escritura exclusivamente a los registros pertenecientes al inquilino (tenant) autenticado, garantizando el aislamiento de la información y protegiendo los datos sensibles frente a accesos no autorizados directos a la base de datos.

4. Implementación y Operación

4.1. Tecnologías y Herramientas

La implementación del sistema se apoya en un conjunto de tecnologías seleccionadas según su rol dentro de la solución. Cada herramienta cumple una función específica y delimitada, lo que permite mantener una arquitectura clara y coherente entre la aplicación móvil, el backend y el dashboard web.

4.1.1. Aplicación móvil

La aplicación móvil se desarrolla utilizando Flutter SDK versión 3.19.x o superior, junto con Dart SDK versión 3.3.x. Flutter se utiliza como framework principal para construir una aplicación nativa en Android, con una única base de código y soporte estable para dispositivos móviles y tablets.

La elección de Flutter responde principalmente a criterios de aprendizaje autónomo y viabilidad técnica en el contexto del proyecto. El autor contaba con una experiencia previa en el manejo del framework, lo

que facilitó una primera aproximación y permitió consolidar su dominio técnico durante el desarrollo de la memoria.

Flutter ofrece una curva de entrada razonable, una documentación amplia y una integración directa con herramientas modernas de backend, lo que resultó adecuado para el desarrollo de un prototipo funcional en un tiempo acotado. Adicionalmente, el enfoque de una sola base de código para distintos factores de forma permitió cubrir de manera consistente el uso en teléfonos y tablets, dispositivos habituales en el entorno del centro de entrenamiento.

Otro factor considerado fue la proyección futura de la solución. Si bien el proyecto se implementa y evalúa únicamente en Android, el uso de Flutter deja abierta la posibilidad técnica de portar la aplicación a iOS sin reescribir la lógica principal, lo que resulta coherente con una eventual evolución del sistema más allá del alcance de esta memoria.

El entorno de desarrollo principal corresponde a Visual Studio Code, utilizado para la escritura del código, depuración y gestión del proyecto. Para la compilación y generación del archivo APK se emplean las herramientas de Android Studio Build Tools y el SDK de Android.

Dentro de la aplicación móvil se utilizan las siguientes librerías relevantes:

- `supabase_flutter`, encargada de la comunicación con el backend, la ejecución de operaciones CRUD y la gestión de sesiones anónimas.
- `table_calendar`, utilizada para renderizar el calendario de asistencia y el historial mensual de sesiones.
- `image_picker`, utilizada para acceder a la cámara y galería del dispositivo al momento de asignar una foto de perfil a un cliente.
- `salomon_bottom_bar`, utilizada como componente de navegación principal entre las vistas de la aplicación.
- `intl`, utilizada para el formateo de fechas y la localización de textos, especialmente en la visualización del calendario.

Flutter se encarga exclusivamente de la capa de presentación y de la lógica local. Renderiza la interfaz gráfica, maneja la navegación entre vistas, ejecuta validaciones de entrada y controla el estado de la aplicación mientras se encuentra en ejecución. No almacena datos de forma permanente ni ejecuta lógica propia de servidor.

4.1.2. Backend y servicios en la nube

El backend del sistema se implementa utilizando Supabase, una plataforma Backend-as-a-Service que provee servicios de persistencia de datos, almacenamiento de archivos y acceso mediante API. Supabase se basa en PostgreSQL como motor de base de datos relacional, lo que permite definir un esquema estructurado, aplicar restricciones de integridad y mantener consistencia en los datos gestionados por el sistema.

La elección de Supabase se fundamenta en la eficiencia operativa y en la reducción de la complejidad asociada a la infraestructura de backend. Dado que el autor incursionaba por primera vez en plataformas en la nube, la tecnología permitió abordar el aprendizaje de una arquitectura moderna de manera progresiva, concentrando el esfuerzo en el diseño del modelo de datos sobre PostgreSQL y en la lógica de la aplicación, más que en la administración de servidores.

El uso de PostgreSQL como base relacional fue un factor relevante en la decisión, ya que facilita la representación estructurada del dominio del entrenamiento, el uso de claves foráneas, restricciones y funciones de validación. Adicionalmente, la integración directa con Flutter mediante el SDK oficial permitió simplificar la comunicación entre frontend y backend, evitando la implementación de una capa intermedia de servicios.

Se descartó el uso de una base de datos exclusivamente local debido a sus limitaciones en términos de escalabilidad y acceso remoto. Una solución local habría impedido la construcción del dashboard web y el acceso centralizado a la información histórica, elementos considerados relevantes para la supervisión y análisis posterior del entrenamiento.

En el contexto del proyecto, Supabase se utiliza como repositorio central de datos y como fuente única de verdad tanto para la aplicación móvil como para el dashboard web. La lógica de negocio se mantiene mayoritariamente en el cliente, mientras que el backend cumple un rol de persistencia, consulta y resguardo de la información, alineándose con un enfoque pragmático y con prácticas actuales en el desarrollo de aplicaciones modernas.

Supabase Storage se utiliza para el almacenamiento de archivos binarios, específicamente las fotos de perfil de los clientes. La aplicación móvil guarda únicamente la referencia al archivo, manteniendo liviana la persistencia local.

El sistema no implementa una interfaz de autenticación visible para el usuario. Para efectos del diseño final, se utiliza autenticación anónima en segundo plano junto con políticas de seguridad a nivel de fila, lo que permite controlar el acceso a los datos sin incorporar un flujo de inicio de sesión explícito.

4.1.3. Dashboard web

El dashboard web se implementa utilizando Streamlit, un framework basado en Python (versión 3.9 o superior). La elección de esta tecnología se fundamenta en su bajo costo de implementación y en la capacidad de aprovechar el ecosistema de análisis de datos de Python, con el cual el autor contaba con experiencia previa. Esto permitió desarrollar un prototipo funcional de visualización de manera declarativa, sin añadir complejidad innecesaria al sistema principal.

Las librerías principales utilizadas en el dashboard son:

- supabase, cliente oficial de Python utilizado para conectarse a la base de datos y ejecutar consultas de lectura.
- pandas, utilizado para transformar los datos obtenidos desde Supabase en estructuras tabulares, aplicar filtros por fecha y cliente, y calcular métricas como duración promedio y adherencia.
- altair, utilizada para generar gráficos de evolución temporal y visualizaciones interactivas basadas en series de tiempo.

Arquitectónicamente, el dashboard opera como una aplicación independiente de la solución móvil y se conecta directamente a Supabase realizando únicamente operaciones de lectura. Streamlit actúa como la capa de presentación, renderizando la interfaz y gestionando los filtros, delegando toda la persistencia de datos en el backend centralizado.

La aplicación se encuentra alojada en Streamlit Community Cloud con acceso restringido mediante URL privada. Su uso está orientado exclusivamente al dueño del centro y entrenadores para el análisis de información histórica, excluyéndose deliberadamente de la operación durante las sesiones presenciales para no interferir con el flujo de trabajo del entrenador. De este modo, la herramienta complementa la solución móvil bajo un criterio técnico y formativo, sin desplazar el foco del proyecto desde su objetivo principal: la gestión operativa del entrenamiento en sala.

4.1.4. Conjunto tecnológico y coherencia de la solución

La combinación de Flutter, Supabase y Streamlit permite separar claramente la operación en sala del análisis posterior. La aplicación móvil se enfoca en la gestión del entrenamiento presencial, el backend

centraliza la persistencia y la seguridad de los datos, y el dashboard web apoya la visualización y el análisis sin interferir con la operación diaria. Esta separación refuerza la mantenibilidad del sistema y facilita su evolución futura.

4.2. Desarrollo de funcionalidades clave

Este apartado describe las funcionalidades principales implementadas en la aplicación móvil. La descripción se enfoca en el flujo lógico de uso, las validaciones aplicadas y la relación directa con los requerimientos definidos previamente. No se presenta código fuente, ya que el objetivo es evidenciar el funcionamiento real del sistema.

4.2.1. Planificación de sesiones por etapas

La planificación de sesiones constituye el núcleo funcional de la aplicación. El flujo comienza cuando el entrenador selecciona un cliente desde el listado principal y decide crear una nueva sesión. El sistema solicita una fecha y una descripción opcional, y luego genera automáticamente la estructura base de la sesión.

Cada sesión se compone de tres etapas predefinidas. Una etapa de calentamiento y dos etapas generales. Esta estructura asegura consistencia en la planificación y evita omisiones en el diseño del entrenamiento. En cada etapa, el entrenador puede agregar ejercicios desde el catálogo, filtrar por tipo y asignar valores según la unidad de medida definida para cada ejercicio.

Durante la planificación, el sistema permite definir repeticiones, tiempo o distancia, junto con un peso estimado cuando corresponde. Los ejercicios se ordenan de forma secuencial dentro de cada etapa. La sesión queda disponible en la vista de inicio una vez finalizada la planificación.

Esta funcionalidad cumple los requerimientos RF-03, RF-04, RF-05, RF-06 y RF-07.

Como se observa en la Figura 5, la aplicación presenta una vista de planificación centrada en la estructura de la sesión de entrenamiento. En esta pantalla, el entrenador puede visualizar la sesión activa y organizarla en etapas diferenciadas, permitiendo agregar ejercicios, definir su orden y asignar los parámetros correspondientes según el tipo de actividad.

Esta vista concentra las acciones principales de planificación, facilitando que el entrenador construya la rutina de forma progresiva antes de iniciar la ejecución del entrenamiento, manteniendo una visión clara de la estructura completa de la sesión.

×

Planificando...

Cliente: Antonella Fuentes Sepúlveda

📅

3/1/2026

Nota general

entrenamiento cuerpo completo

🔥

Calentamiento General

Calentamiento

1 vuelta

✎

🗑

+

Sin ejercicios asignados (Requerido)

🔄

Bloque Principal 1

Bloque Principal

1 vuelta

✎

🗑

+

Sin ejercicios asignados (Requerido)

🔄

Bloque Principal 2

Bloque Principal

1 vuelta

✎

🗑

+

Sin ejercicios asignados (Requerido)

+ Agregar otro bloque

✓

TERMINAR PLANIFICACIÓN

Figura 5. Captura de pantalla de la aplicación desde la vista de planificación.

4.2.2. Repetir sesiones

La funcionalidad de repetir sesiones permite reutilizar sesiones previamente realizadas por un cliente. El entrenador accede al historial del cliente, selecciona una sesión anterior y elige la opción de repetición.

El sistema crea una nueva sesión copiando la estructura completa de la sesión original, incluyendo etapas, ejercicios y valores asignados. La nueva sesión queda asociada a la fecha actual y puede ser editada antes de su ejecución.

Este flujo reduce el tiempo de planificación en sesiones recurrentes y asegura continuidad en el entrenamiento. El entrenador mantiene control total sobre los ajustes necesarios antes de finalizar la planificación.

Esta funcionalidad cumple los requerimientos RF-08 y RF-07.

Como se observa en la Figura 6, la aplicación incorpora en la vista de una sesión previamente realizada un botón específico para repetir la rutina.

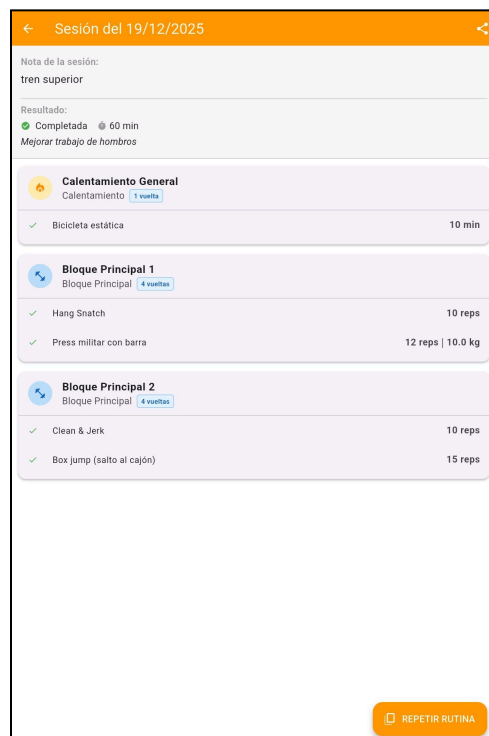


Figura 6. Captura de pantallas de la aplicación desde la vista de una sesión antigua con el botón “repetir rutina” visible.

4.2.3. Compartir sesiones

La aplicación permite compartir una sesión planificada entre distintos clientes, funcionalidad orientada a contextos donde varios alumnos realizan entrenamientos similares dentro de una misma jornada. El proceso mantiene la individualización del registro, evitando la duplicación de trabajo manual por parte del entrenador.

El entrenador inicia el proceso seleccionando una sesión previamente registrada y eligiendo un cliente destinatario. El sistema genera una copia independiente de la sesión para el nuevo cliente, conservando la estructura original y permitiendo realizar ajustes posteriores si se requiere.

Esta funcionalidad cumple el requerimiento RF-09.

El flujo completo para compartir una sesión se describe a continuación y se apoya en una secuencia de capturas de pantalla que ilustran cada paso del proceso.

Como se observa en la Figura 7, el proceso se inicia desde la vista de una sesión previamente realizada, donde el entrenador dispone de un botón específico para compartir la sesión. Esta acción permite reutilizar una planificación existente como base para otro cliente.

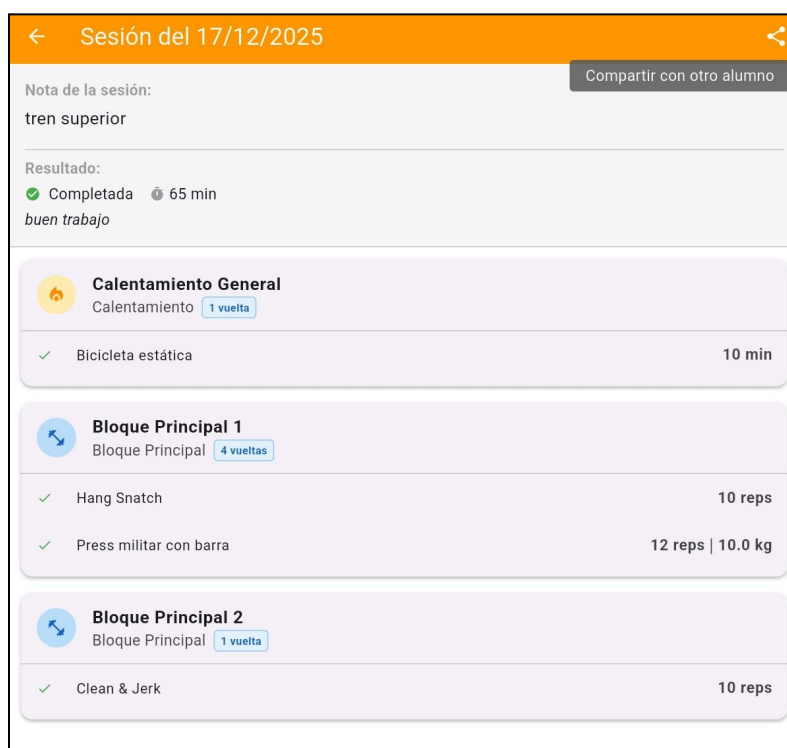


Figura 7. Captura de pantalla de la aplicación desde una vista de sesión antigua con el botón de “compartir” visible, Paso 1 para compartir sesión.

Como se muestra en la Figura 8, luego de seleccionar la opción para compartir la sesión, la aplicación solicita una confirmación explícita por parte del entrenador. Esta pantalla permite validar la acción antes

de generar la copia de la sesión, reduciendo el riesgo de operaciones involuntarias y reforzando el control del proceso de planificación.

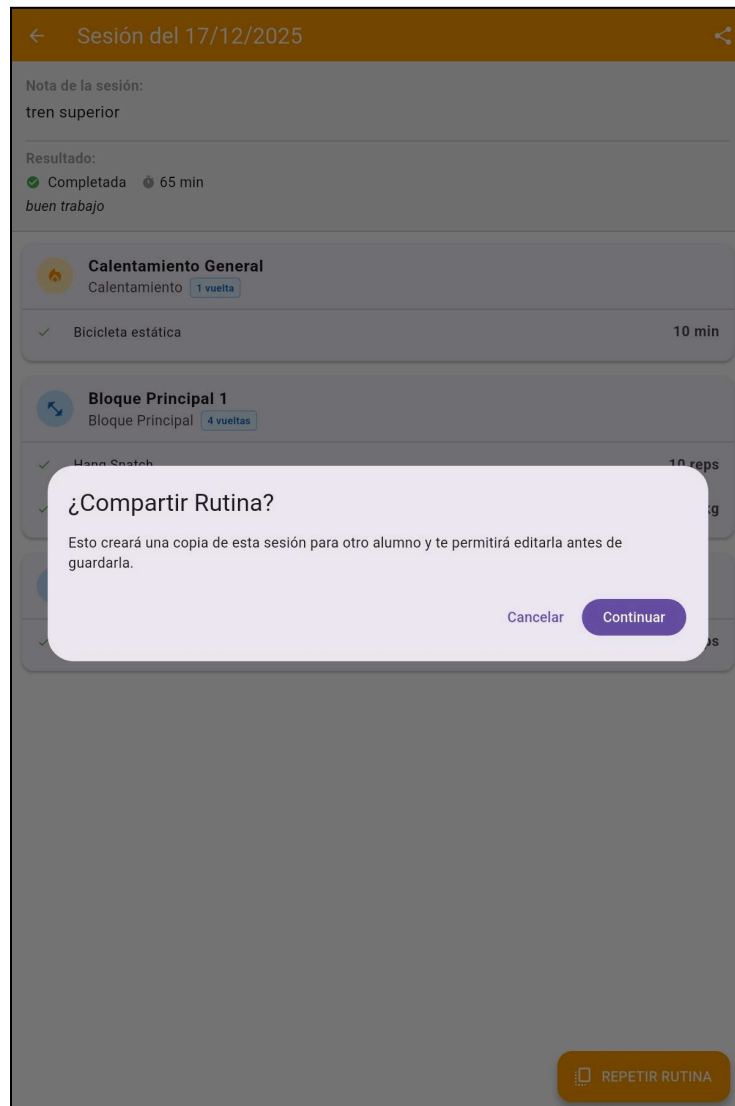


Figura 8. Captura de pantalla de la aplicación desde una vista de sesión antigua para pedir confirmación luego de apretar el botón de “compartir”. Paso 2 para compartir sesión.

Como se observa en la Figura 9, una vez confirmada la acción, la aplicación despliega un listado de clientes disponibles para seleccionar al destinatario de la sesión. Esta vista permite al entrenador identificar y elegir de forma directa al alumno correspondiente, manteniendo el control del proceso y evitando asignaciones incorrectas.

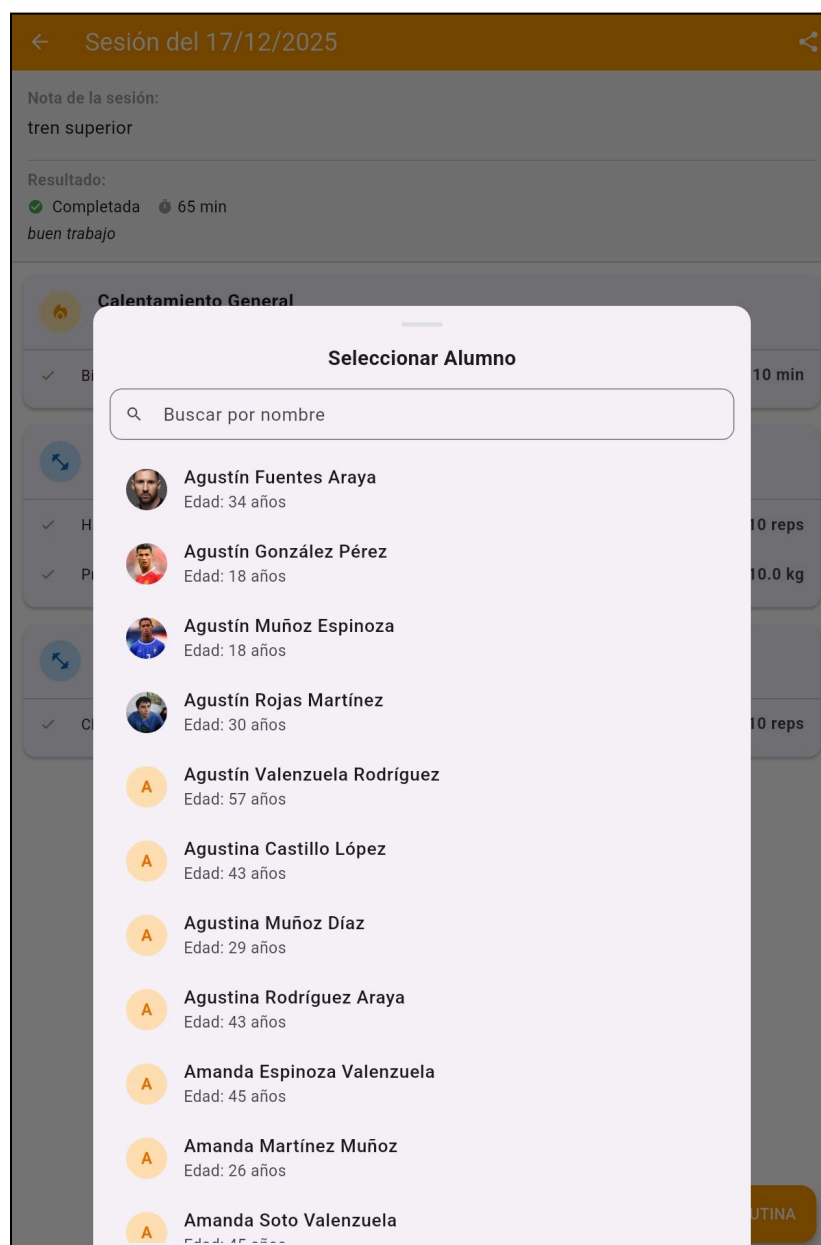


Figura 9. Captura de pantalla de la aplicación desde una vista de sesión antigua abriendo un listado de clientes para seleccionar a quien compartir la sesión. Paso 3 para compartir sesión.

Como se muestra en la Figura 10, una vez seleccionado el cliente destinatario, la aplicación presenta la vista de planificación de la sesión compartida en modo editable. En esta pantalla, el entrenador puede revisar la estructura heredada de la sesión original y realizar ajustes si son requeridos, tales como modificar ejercicios, orden o parámetros, antes de dar inicio a la ejecución del entrenamiento.

×

Planificando...

Cliente: Agustín Muñoz Espinoza

📅

3/1/2026

Nota general

🔥

Calentamiento General

Calentamiento 1 vuelta

✎

🗑

+

✓

Bicicleta estática

10 min

⋮

↺↻

Bloque Principal 1

Bloque Principal 4 vueltas

✎

🗑

+

✓

Hang Snatch

10 reps

⋮

✓

Press militar con barra

12 reps | 10.0 kg

⋮

↺↻

Bloque Principal 2

Bloque Principal 1 vuelta

✎

🗑

+

✓

Clean & Jerk

10 reps

⋮

+ Agregar otro bloque

✓

TERMINAR PLANIFICACIÓN

*Figura 10. Captura de pantalla de la aplicación desde la vista de planificación editable de la sesión compartida a otro cliente.
Paso 4 para compartir sesión.*

Como se observa en la Figura 11, una vez finalizado el proceso, la sesión compartida queda registrada correctamente en la vista de inicio del sistema, asociada al cliente destinatario. Esta confirmación visual permite al entrenador verificar de inmediato que la operación se realizó con éxito y que la sesión se encuentra disponible para su ejecución dentro de la jornada de entrenamiento.

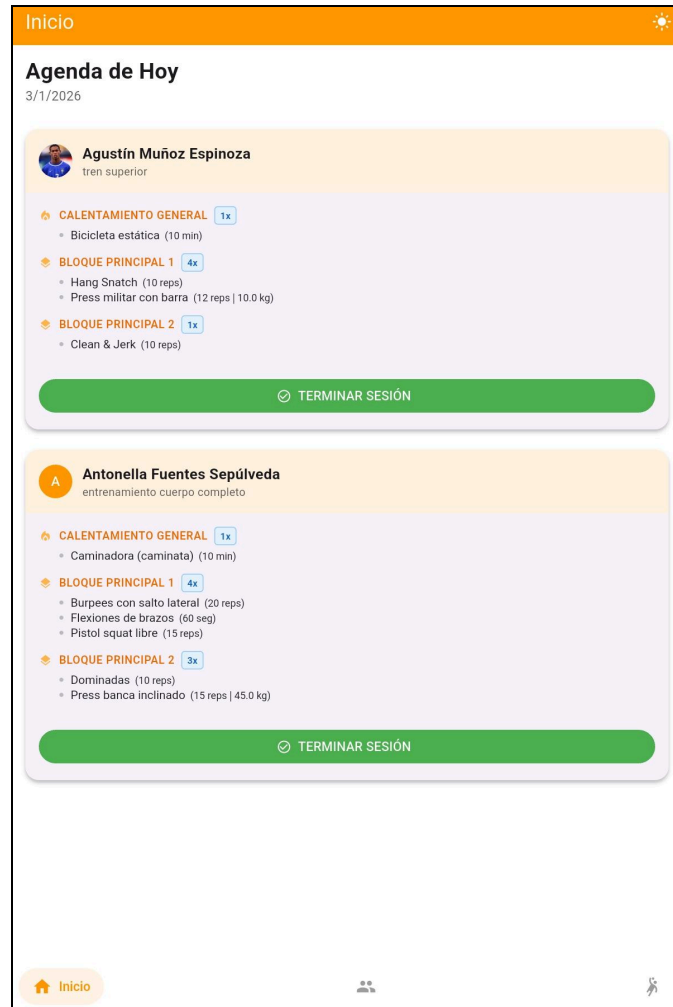


Figura 11. Captura de pantalla de la aplicación desde la vista de inicio, con la sesión compartida con éxito. Paso 5 para compartir sesión.

4.2.4. Cierre de sesión con validaciones

Al finalizar el entrenamiento, el entrenador accede a la sesión activa desde la vista de inicio y selecciona la opción de cierre. El sistema solicita confirmar si la sesión fue completada, registrar la duración total y agregar una observación.

Si el entrenador indica que la sesión no fue completada, el sistema exige de forma obligatoria el ingreso de una observación antes de permitir el cierre. Esta validación asegura que el historial conserve contexto sobre interrupciones, lesiones o problemas durante la ejecución.

Una vez cerrada la sesión, el sistema registra la información en el historial del cliente y elimina la sesión de la vista de inicio.

Esta funcionalidad cumple los requerimientos RF-11, RF-12 y RF-13.

Como se muestra en la Figura 12, al intentar finalizar una sesión marcada como no completada, la aplicación impide el cierre si no se ha ingresado una observación. Esta validación obliga al entrenador a registrar el motivo de la interrupción, asegurando que el historial del cliente conserve contexto relevante sobre situaciones como lesiones, pausas anticipadas o dificultades durante la ejecución del entrenamiento.

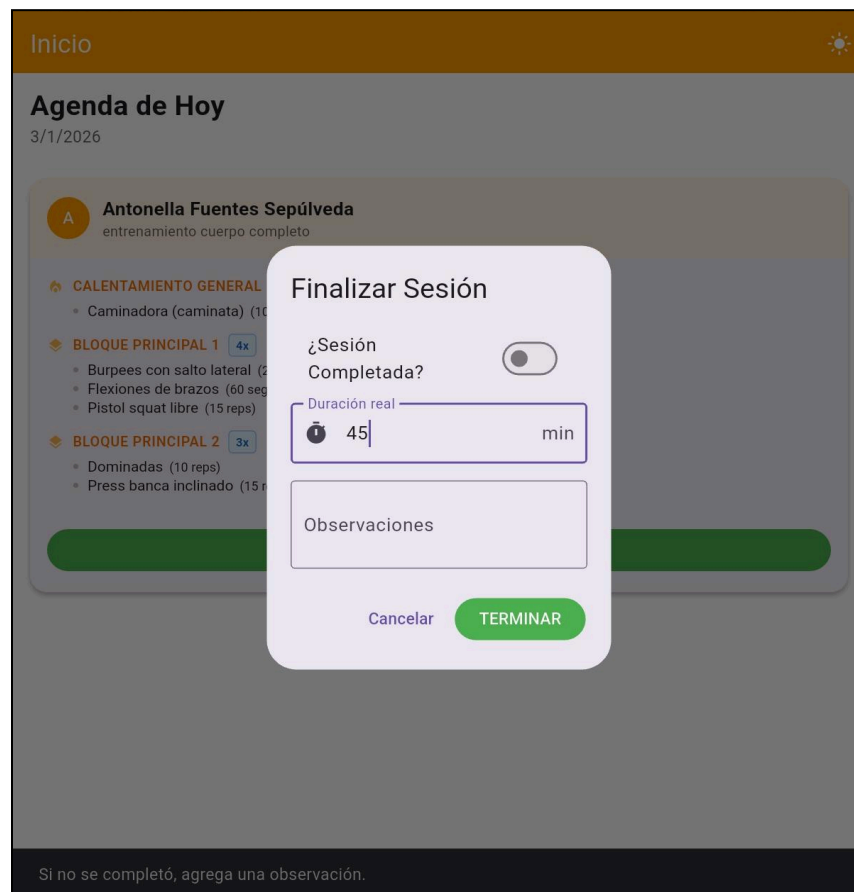


Figura 12. Captura de pantalla de la aplicación desde la vista de inicio, al momento de finalizar una sesión no completada sin observación.

4.2.5. Historial y calendario de asistencia

El sistema permite consultar el historial completo de sesiones por cliente mediante dos vistas complementarias. Una vista de listado cronológico y una vista de calendario mensual.

El listado muestra las sesiones realizadas con su fecha, estado de completitud y duración. La vista de calendario destaca los días con sesiones registradas, permitiendo identificar patrones de asistencia de forma visual.

Estas vistas permiten al entrenador revisar el progreso del cliente, identificar interrupciones y evaluar la regularidad del entrenamiento sin revisar registros manuales.

Esta funcionalidad cumple los requerimientos RF-14 y RF-15.

Como se observa en la Figura 13, la vista de historial en modo listado presenta las sesiones realizadas de forma cronológica, indicando para cada una la fecha, el estado de completitud y la duración registrada. Esta vista permite una revisión detallada de la actividad del cliente y facilita el análisis puntual de sesiones específicas.

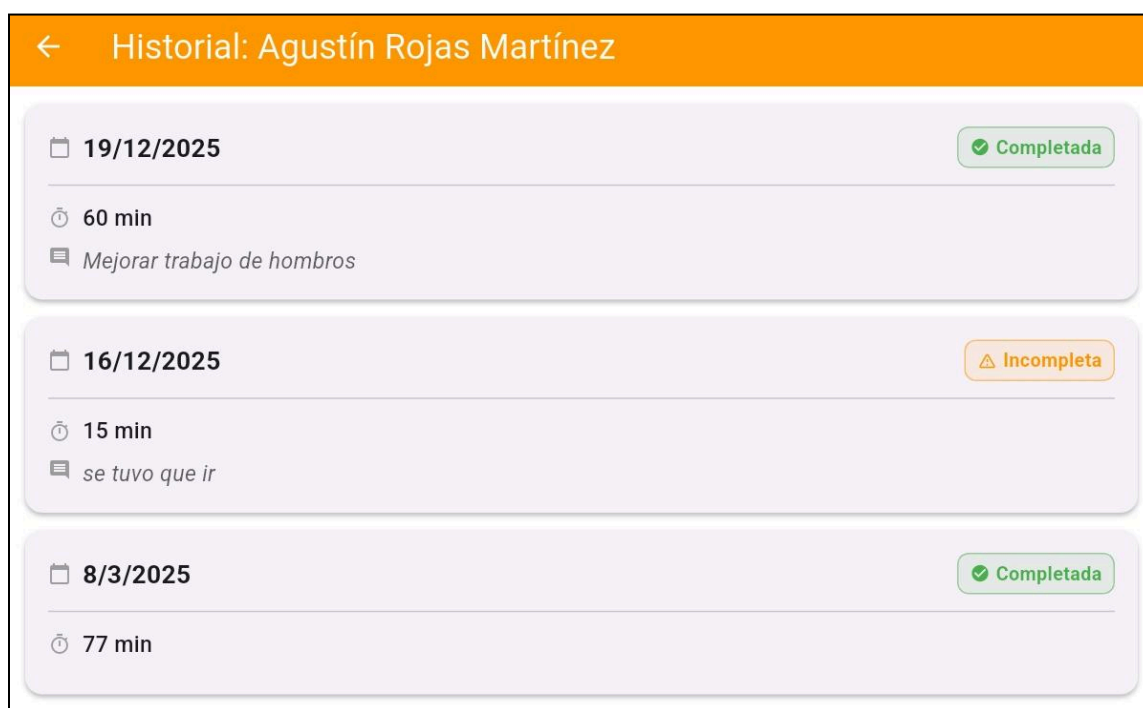


Figura 13. Captura de pantalla de la aplicación desde una vista del historial modo listado de sesiones completo del cliente.

De manera complementaria, como se muestra en la Figura 14, la vista de historial en modo calendario destaca visualmente los días en los que se registraron sesiones de entrenamiento. Esta representación permite identificar patrones de asistencia y regularidad en el tiempo, apoyando una evaluación rápida del compromiso del cliente con el entrenamiento.



Figura 14. Captura de pantalla de la aplicación desde la vista del historial modo calendario del cliente.

Con estas funcionalidades, la aplicación cubre los flujos operativos principales del entrenamiento presencial. Cada módulo se integra de forma coherente con los requerimientos definidos y demuestra la implementación efectiva del sistema en un entorno real.

4.2.6. Visualización analítica en dashboard web

El dashboard web constituye un componente complementario a la aplicación móvil, orientado a la visualización y análisis de la información histórica generada durante la operación del centro. Su propósito no es intervenir en la ejecución del entrenamiento presencial, sino apoyar la toma de decisiones a mediano y largo plazo a partir de datos consolidados.

A diferencia de la aplicación móvil, centrada en la acción inmediata y en la gestión de sesiones activas, este módulo permite al dueño del centro acceder a una vista agregada del funcionamiento general. La interfaz se diseña priorizando la claridad visual y la lectura rápida de indicadores relevantes, facilitando la interpretación sin necesidad de revisar registros individuales.

Como se observa en la Figura 15, el panel principal presenta métricas globales de alto nivel, tales como el número total de sesiones realizadas en un período determinado, la duración promedio de los entrenamientos y la regularidad de asistencia de los clientes. Estos indicadores permiten identificar patrones generales de uso y carga operativa del centro.

Adicionalmente, el dashboard incorpora controles de filtrado que permiten segmentar la información por rango temporal o por cliente específico. Esta funcionalidad habilita el análisis de la evolución del entrenamiento y apoya decisiones relacionadas con planificación, seguimiento y organización del trabajo, transformando los datos operativos recolectados por la aplicación móvil en información útil para la gestión del centro.

Esta funcionalidad cumple los requerimientos RF-16.

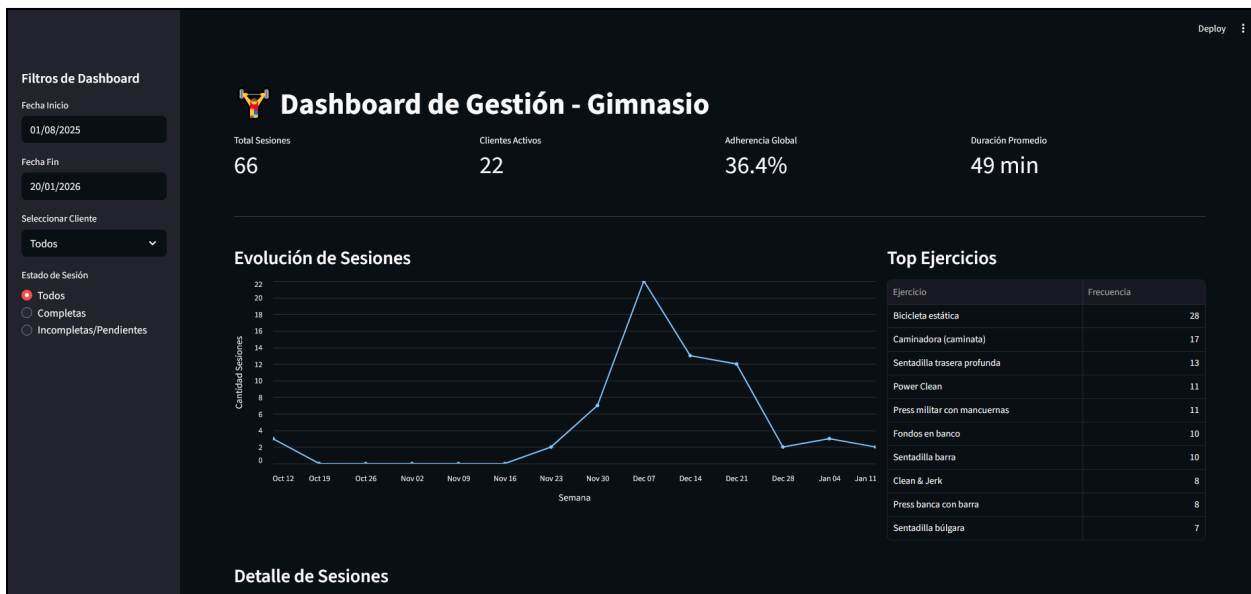


Figura 15. Captura de pantalla del Dashboard Web mostrando indicadores de rendimiento y filtros temporales.

4.3. Integración frontend backend

La integración entre la aplicación móvil y el backend se realiza de forma directa mediante los servicios provistos por Supabase. El diseño prioriza simplicidad operativa, coherencia con el alcance del proyecto y control centralizado de los datos.

4.3.1. Inicialización y conexión

La conexión con Supabase se ejecuta al inicio de la aplicación móvil. En el archivo principal se configura el cliente con la URL del proyecto y la clave anónima. Esta inicialización ocurre antes de cargar la interfaz, lo que garantiza que todas las vistas tengan acceso al backend desde el primer momento.

El uso de credenciales definidas en el código responde a que el sistema opera en un entorno controlado, sobre un dispositivo dedicado del centro de entrenamiento. Para el informe final, esta decisión se complementa con políticas de seguridad a nivel de fila que restringen el acceso a los datos.

4.3.2. Autenticación y seguridad

La aplicación utiliza autenticación anónima en segundo plano. Al iniciar, se ejecuta un inicio de sesión silencioso que genera un token válido. Este token permite cumplir las políticas de seguridad definidas en el backend sin requerir interacción del entrenador.

Las políticas de Row Level Security se definen para permitir acceso completo a los usuarios autenticados. Este enfoque se justifica porque el sistema corresponde a un administrador único y el acceso físico al dispositivo actúa como primera barrera de seguridad. La autenticación anónima impide accesos externos directos a la base de datos sin pasar por la aplicación.

4.3.3. Operaciones de datos

La comunicación entre Flutter y Supabase se realiza mediante el SDK `supabase_flutter`. La aplicación utiliza operaciones estándar de lectura y escritura. Se emplean consultas `select` para obtener clientes, sesiones e historial. Se utilizan `insert` y `update` para crear y modificar sesiones y registros de cierre. La eliminación se limita a casos controlados como la depuración de datos de prueba.

El catálogo de ejercicios, tipos y etapas se consume en modo solo lectura desde la aplicación móvil. Estos datos se consideran estables en la versión evaluada y no forman parte del flujo operativo diario del entrenador.

4.3.4. Manejo de imágenes

La gestión de imágenes de perfil integra el acceso al hardware del dispositivo con el almacenamiento remoto. El entrenador selecciona una imagen desde la cámara o galería. La aplicación procesa el archivo

y lo sube a un bucket en Supabase Storage. Una vez almacenado, se obtiene una URL pública que se guarda en la base de datos asociada al cliente.

Cada imagen se almacena con un nombre único generado a partir de la marca de tiempo. Esta estrategia evita conflictos de caché y permite actualizar la foto sin eliminar archivos previos. La aplicación siempre muestra la imagen referenciada en el registro del cliente.

4.3.5. Manejo de estado y actualización de la interfaz

La aplicación utiliza el manejo de estado nativo de Flutter mediante `setState`. Cada vista controla su propio estado de carga a través de variables booleanas. Las consultas al backend se ejecutan de forma asíncrona durante la inicialización de la vista o después de una acción del usuario.

Tras una operación de escritura, la aplicación vuelve a consultar los datos desde el backend. Este enfoque asegura consistencia entre la base de datos y la interfaz, evitando estados desincronizados. No se utiliza sincronización en tiempo real ni suscripciones por eventos.

4.3.6. Manejo de errores y validaciones

Las validaciones de negocio se implementan en la capa de presentación. Durante el cierre de una sesión, la aplicación valida que se ingrese una observación obligatoria cuando la sesión no se completa. Esta validación se realiza antes de enviar los datos al backend, evitando registros incompletos.

Los errores de conexión o fallos en las consultas se capturan mediante bloques `try catch`. El sistema informa al entrenador mediante mensajes visuales temporales. A nivel interno, los errores se registran en consola para facilitar la depuración durante el desarrollo.

Esta integración permite que la aplicación móvil opere como un cliente liviano, delegando la persistencia y la seguridad al backend. El resultado es un flujo estable, comprensible y coherente con el alcance del proyecto, que respalda el uso del sistema en un entorno real de entrenamiento presencial.

4.4. Manual y condiciones de operación

Este apartado describe las condiciones necesarias para operar el sistema en un centro de entrenamiento y el flujo básico de uso diario. El objetivo es establecer de forma clara qué se requiere para que la solución funcione de manera correcta y sostenida en el tiempo.

4.4.1. Requisitos de operación

El sistema se utiliza en una tablet Android dedicada al centro de entrenamiento. El dispositivo debe contar con sistema operativo Android versión 5.0 como mínimo. Se recomienda el uso de versiones recientes para asegurar compatibilidad y rendimiento.

La operación requiere conexión a internet activa mediante red WiFi del centro o datos móviles. El dispositivo es de uso exclusivo del entrenador o dueño del gimnasio. No se contempla el uso simultáneo desde varios dispositivos.

4.4.2. Flujo operativo diario

El uso del sistema durante una jornada típica sigue un flujo simple y repetible.

- El alumno llega al centro y se presenta con el entrenador.
- El entrenador busca al alumno en el listado de clientes y revisa de forma rápida su historial reciente.
- El entrenador asigna una sesión de entrenamiento. Puede repetir una sesión anterior o crear una nueva según el criterio profesional. Si es necesario, el entrenador edita la sesión antes de iniciar el entrenamiento.
- Durante la sesión, el entrenador supervisa la ejecución apoyándose en la vista de inicio.
- Al finalizar, el entrenador cierra la sesión indicando si fue completada. Si no se completa, ingresa la observación obligatoria.
- La sesión queda registrada en el historial del alumno para revisión posterior.

Este flujo se repite de forma independiente para cada alumno que entrena durante la jornada.

4.4.3. Capacitación del usuario

El sistema se diseña con foco en usabilidad y simplicidad. No requiere capacitación formal ni manuales extensos. La inducción inicial consiste en una explicación breve del flujo general y de las vistas principales, con una duración aproximada de 15 a 30 minutos.

Se considera suficiente un uso guiado durante los primeros días de operación. Tras este periodo, el entrenador puede operar el sistema de forma autónoma sin apoyo externo.

4.4.4. Mantenimiento del sistema

El mantenimiento del sistema se concentra en tareas de bajo esfuerzo.

- La base de datos cuenta con respaldo automático gestionado por la plataforma Supabase.
- Las actualizaciones de la aplicación móvil se realizan mediante la instalación manual de un nuevo archivo APK cuando se libera una versión.
- El catálogo de ejercicios se revisa y ajusta de forma ocasional según las necesidades del centro.
- No se considera soporte técnico permanente ni monitoreo externo.

4.4.5. Manejo ante fallas

El sistema no dispone de modo offline. Si no existe conexión a internet, la aplicación no puede operar ni registrar información. En caso de pérdida de conectividad durante el uso, el entrenador debe reintentar la operación una vez que la conexión se restablezca.

No se almacenan datos localmente para sincronización posterior. Esta limitación se declara explícitamente como parte del alcance del proyecto.

4.4.6. Seguridad y privacidad en operación

El acceso al sistema se protege mediante el PIN del dispositivo. La aplicación no implementa inicio de sesión propio. Se asume control físico del dispositivo por parte del entrenador.

El sistema no almacena datos sensibles de salud ni información legal del alumno. Los datos gestionados se limitan a información básica y registros de entrenamiento.

El dashboard web asociado al sistema se accede mediante una URL privada y su uso se restringe al dueño del centro.

4.4.7. Límites operativos

El sistema se diseña para operar con hasta 15 alumnos entrenando en paralelo durante horarios punta. Se espera uso intensivo en bloques de mañana y tarde.

Cada cliente mantiene una única imagen de perfil. El volumen de datos se mantiene controlado para no afectar el rendimiento ni la operación diaria.

Estas condiciones definen el marco operativo realista del sistema y establecen cómo debe utilizarse para cumplir su propósito dentro del centro de entrenamiento.

5. Diseño y ejecución de pruebas

5.1. Diseño experimental y métricas

5.1.1. Objetivo de la evaluación

La evaluación se centra en la usabilidad de la aplicación móvil, dado que esta constituye el principal medio de interacción del sistema y concentra las tareas críticas del entrenamiento presencial, tales como la planificación, supervisión y registro de sesiones. La evaluación fue concebida como una instancia de validación formativa, orientada a comprobar la adecuación del producto al contexto real de uso del centro de entrenamiento y a detectar posibles problemas de interacción que pudieran no haber sido evidentes durante el desarrollo, más que a obtener resultados estadísticamente generalizables.

El objetivo de la evaluación es determinar si la aplicación permite al entrenador realizar sus tareas de forma efectiva, eficiente y satisfactoria dentro de su entorno operativo habitual. El dashboard web se excluye de la evaluación de usabilidad, ya que fue diseñado como un componente complementario destinado a la validación de la arquitectura, la integración de datos y su visualización agregada, y no como parte del flujo operativo diario del entrenamiento presencial.

5.1.2. Diseño experimental

La evaluación se realiza con un único participante, correspondiente al dueño del gimnasio y usuario real del sistema móvil. El participante es responsable de la gestión y supervisión de rutinas y sesiones, y utiliza la aplicación como parte de su operación diaria. Este diseño se justifica porque el sistema no está orientado a un uso masivo ni a consumo general, sino a un contexto específico con un único usuario principal. El participante presenta experiencia media en el uso de herramientas digitales.

Se diseñó un protocolo de pruebas de usabilidad centrado en la ejecución de tareas representativas del flujo diario de trabajo del entrenador. Dado que el sistema fue evaluado por un único usuario experto en el dominio, correspondiente al dueño del gimnasio, se utilizó el cuestionario System Usability Scale (SUS) como instrumento estructurado de apoyo a una evaluación exploratoria de usabilidad, aplicado al finalizar el conjunto de tareas, siguiendo las recomendaciones de uso establecidas en la literatura especializada [4].

De manera complementaria, se incorporaron preguntas de facilidad por tarea mediante la métrica Single Ease Question (SEQ), junto con observación directa durante la ejecución de las actividades definidas, metodología habitualmente empleada para evaluar la percepción de esfuerzo y claridad de interacción en tareas específicas [4]. Este enfoque permitió identificar fortalezas y oportunidades de mejora del sistema sin buscar la generalización de los resultados, manteniendo coherencia con el carácter formativo y de validación temprana del prototipo desarrollado.

La evaluación contempla pruebas de usabilidad con usuario real y pruebas funcionales de caja negra sobre flujos críticos. Las pruebas se planifican en un entorno controlado, utilizando la aplicación móvil instalada en el dispositivo habitual del participante, con el fin de reproducir condiciones reales de uso sin interferir con la operación normal del centro.

5.1.3. Tareas evaluadas

El experimento somete al usuario a cuatro escenarios que cubren el ciclo de vida completo de una sesión de entrenamiento:

1. Registro de sesión: Inicio del flujo operativo.
2. Asignación de rutina: Estructuración de la carga de trabajo.
3. Registro de ejecución: Ingreso de datos críticos bajo carga cognitiva.
4. Consulta de historial: Apoyo a la toma de decisiones.

5.1.4. Métricas y herramientas de medición

La evaluación utiliza métricas cuantitativas y cualitativas para medir la usabilidad de la aplicación móvil. Para cada tarea se aplica la métrica SEQ. Al finalizar cada tarea, el participante responde una pregunta en una escala de 1 a 7 sobre la facilidad o dificultad percibida al completar la tarea en la aplicación móvil.

Al finalizar el conjunto de tareas se aplica la escala SUS para evaluar la experiencia global de uso. El participante responde considerando su experiencia general durante la ejecución de todas las tareas. Como complemento, el evaluador registra observaciones y comentarios cualitativos del participante.

Los datos se registran en una planilla. Los tiempos y resultados se anotan de forma manual. El evaluador documenta observaciones durante la ejecución de las tareas para apoyar el análisis posterior.

5.1.5. Criterios de éxito

Se define como criterio de éxito un puntaje SUS superior a 68, lo que indica un nivel de usabilidad adecuado para el contexto de uso. Para la métrica SEQ, se considera aceptable un puntaje promedio superior a 5 por tarea. Estos umbrales permiten determinar si la aplicación cumple con los objetivos de usabilidad definidos para el sistema.

5.2. Planificación y obstáculos previstos

Las pruebas se realizaron en un único día, con una duración estimada de entre 15 y 30 minutos. Esta planificación permitió ejecutar las tareas de forma acotada, registrar observaciones relevantes y minimizar interferencias con la operación normal del centro. En caso de presentarse alguna eventualidad o de requerirse la repetición de alguna actividad, se consideró la disponibilidad de un segundo día para completar la evaluación bajo las mismas condiciones.

5.2.1. Recursos y contexto de ejecución

Para la ejecución de las pruebas se requiere una tablet Android con la aplicación instalada y configurada, conexión a internet del gimnasio y una planilla de registro para anotar métricas y observaciones. El participante corresponde al entrenador del centro. El rol de evaluador recae en el autor del proyecto, quien observa la ejecución, registra tiempos y documenta hallazgos.

Las pruebas se planificaron fuera de los horarios de mayor demanda. Se realizan durante una jornada real de entrenamiento, pero con carga controlada de alumnos. Esta decisión permite evaluar la aplicación en un contexto realista sin someter al participante a presión operativa excesiva.

5.2.2. Obstáculos previstos y estrategias de mitigación

Se identificaron como obstáculos potenciales la baja disponibilidad del entrenador durante la jornada, las interrupciones propias del funcionamiento del gimnasio y las limitaciones de tiempo para completar todas las tareas en una sola sesión.

Para mitigar estos obstáculos se planificó realizar las pruebas en horarios de baja demanda, dividir las tareas en bloques cortos y permitir pausas entre tareas cuando sea necesario. Estas medidas buscan mantener la calidad de la evaluación y reducir la carga cognitiva durante la ejecución.

5.2.3. Riesgos fuera de control

Existen riesgos que no pueden ser mitigados desde el diseño de la prueba, como emergencias propias del centro de entrenamiento o cambios imprevistos en la agenda del entrenador. En estos casos, las pruebas se reprogramarán manteniendo la misma estructura y condiciones definidas.

Esta planificación establece un marco realista para la ejecución de las pruebas y permite anticipar dificultades operativas sin comprometer la validez de los resultados.

5.3. Ejecución de pruebas y hallazgos

La evaluación de usabilidad se llevó a cabo en el gimnasio HU Athletics, utilizando la aplicación móvil instalada en una tablet Samsung Galaxy Tab A8. La sesión de evaluación tuvo una duración aproximada de entre quince y veinte minutos y se realizó durante una jornada laboral con baja carga de alumnos, lo que permitió un contexto realista sin presión operativa excesiva.

Durante la ejecución de las tareas no se registraron interrupciones directas mientras el participante interactuaba con la aplicación. Las interrupciones propias del entorno del gimnasio ocurrieron entre tareas y no afectaron la continuidad ni la finalización de la evaluación. Ninguna tarea debió repetirse y todas se completaron dentro de lo esperado.

Respecto a los resultados por tarea, el registro de una nueva sesión de entrenamiento fue completado sin ayuda y obtuvo un puntaje SEQ de 7, lo que indica una percepción de alta facilidad de uso. La asignación de una rutina por etapas también fue completada sin asistencia, alcanzando un puntaje SEQ de 7, evidenciando claridad en la estructura de planificación propuesta por el sistema.

La tarea de registro de la ejecución del ejercicio presentó mayor dificultad relativa. El participante completó la tarea con apoyo del evaluador y asignó un puntaje SEQ de 4,5. La principal dificultad observada se relacionó con la selección del ejercicio adecuado desde el catálogo y la asignación del valor correspondiente según la unidad de medida definida. Este hallazgo sugiere una oportunidad de mejora en la presentación o asistencia visual durante este paso del flujo.

La revisión del historial de sesiones del cliente fue completada sin ayuda y obtuvo un puntaje SEQ de 7. El participante logró navegar y consultar sesiones anteriores sin inconvenientes, lo que refuerza la efectividad de las vistas de historial implementadas.

Al finalizar las tareas, se aplicó la escala SUS para evaluar la experiencia global de uso de la aplicación móvil. El puntaje obtenido fue de 92,5 sobre 100, valor que se sitúa ampliamente sobre el umbral de aceptabilidad definido en la metodología. Este resultado indica una percepción de usabilidad muy alta por parte del usuario evaluado.

En términos generales, la ejecución de la evaluación confirmó que los flujos principales de la aplicación son comprensibles y utilizables en un contexto real de entrenamiento presencial. El principal hallazgo se concentra en el proceso de registro detallado de ejercicios, que presenta una mayor carga cognitiva y requiere ajustes para mejorar la claridad durante su uso. El resto de las funcionalidades evaluadas mostró un desempeño consistente y acorde a los objetivos planteados.

5.4. Pruebas manuales durante el desarrollo

Durante el desarrollo del sistema se realizaron pruebas manuales de forma iterativa, integradas directamente al proceso de implementación de cada funcionalidad. Estas pruebas se ejecutaron a medida que se incorporaban nuevas acciones en la aplicación, permitiendo validar de forma temprana el correcto funcionamiento del sistema y evitar la propagación de errores hacia etapas posteriores del desarrollo.

Las pruebas se centraron en escenarios de uso reales, verificando principalmente que las operaciones realizadas desde la aplicación móvil se reflejaran correctamente en la base de datos de Supabase. Para cada funcionalidad implementada (como el registro de clientes, la creación y cierre de sesiones o la consulta del historial) se comprobó que los datos fueran insertados, recuperados y actualizados de forma coherente. La validación se apoyó en la inspección directa de las tablas en Supabase y en la revisión visual de la información presentada en la aplicación móvil y el dashboard web, asegurando consistencia entre los datos almacenados y su visualización. Este enfoque permitió confirmar la estabilidad funcional del prototipo dentro del alcance definido para el proyecto.

6. Resultados y Análisis

6.1. Resultados obtenidos

Los resultados de la evaluación de usabilidad se obtuvieron a partir de la aplicación de la métrica SEQ por tarea y de la escala SUS como evaluación global de la aplicación móvil. La evaluación se realizó con un usuario real del sistema, en un contexto operativo controlado y coherente con el uso habitual de la aplicación.

Tarea	Puntaje SEQ	Completo sin ayuda	Observación
Registro de sesión	7	Sí	Flujo claro
Asignación de rutina por etapas	7	Sí	—
Registro de ejecución del ejercicio	4,5	No	Duda en asignar ejercicio y unidad de medida
Revisión del historial del cliente	7	Sí	Historial claro

Tabla 5. Resultados SEQ por tarea

El puntaje promedio SEQ obtenido considerando todas las tareas evaluadas es de 6,4. Este valor se sitúa sobre el umbral definido como aceptable para el contexto de uso de la aplicación.

En relación con la experiencia global, la aplicación obtuvo un puntaje SUS de 92,5 sobre 100. Este valor se encuentra ampliamente por sobre el umbral de usabilidad aceptable y se clasifica dentro del rango de usabilidad excelente.

Los resultados cuantitativos se complementan con observaciones cualitativas registradas durante la evaluación. Las tareas asociadas a la planificación de sesiones y consulta de historial se ejecutaron sin ayuda y sin errores. La principal dificultad observada se concentró en el registro detallado de la ejecución del ejercicio, específicamente en la selección del ejercicio desde el catálogo y en la asignación del valor según la unidad de medida.

6.2. Análisis de resultados

Los resultados obtenidos permiten analizar el cumplimiento de los objetivos específicos definidos para el proyecto, desde la perspectiva de la usabilidad y la operación del sistema.

Respecto al objetivo específico relacionado con la propuesta de un modelo de datos que soporte la asignación de rutinas y el seguimiento del progreso, los resultados muestran que la información estructurada es accesible y comprensible para el usuario. La facilidad observada en la revisión del

historial y en la planificación de sesiones respalda que el modelo permite consultar y reutilizar información sin fricción.

En relación con el objetivo específico asociado a la interfaz gráfica de usuario para dispositivos móviles, los puntajes SEQ de 7 obtenidos en tres de las cuatro tareas evaluadas indican que los flujos principales se alinean con el uso esperado en un entorno presencial. La navegación, la creación de sesiones y la visualización del historial se realizan de forma directa, sin requerir asistencia externa.

El objetivo específico vinculado a la visualización del progreso del entrenamiento también se ve respaldado por los resultados. La tarea de revisión del historial obtuvo un puntaje máximo en la métrica SEQ, lo que indica que la información presentada permite al entrenador consultar sesiones anteriores y comprender el estado del cliente de manera rápida.

El hallazgo asociado a la tarea de registro de ejecución del ejercicio, con un puntaje SEQ de 4,5, indica la presencia de una fricción localizada. Esta dificultad no impide la finalización de la tarea, pero evidencia una mayor carga cognitiva en comparación con el resto de los flujos. Este resultado no invalida la solución, sino que entrega una señal clara sobre un punto susceptible de mejora, particularmente en la asistencia visual durante la selección de ejercicios y la definición de unidades de medida.

El puntaje SUS de 92,5 confirma que, a nivel global, la aplicación ofrece una experiencia de uso altamente satisfactoria para el usuario evaluado. Este resultado respalda que el sistema cumple con los criterios de efectividad, eficiencia y satisfacción definidos en la norma ISO 9241-11, en el contexto específico para el cual fue diseñado.

Además, durante la evaluación del sistema, el entrenador y dueño del centro de entrenamiento destacó positivamente la facilidad de uso y el carácter amigable de la aplicación móvil, señalando que su interacción resulta intuitiva incluso en contextos de alta carga operativa. Asimismo, valoró especialmente la capacidad de respuesta del proceso de desarrollo, en términos de adaptación del sistema a las necesidades reales del entrenamiento presencial. Entre las funcionalidades más relevantes, el participante destacó el historial de sesiones en modo calendario, identificándolo como una herramienta clave para la planificación del entrenamiento y la visualización de la continuidad del trabajo con cada alumno. Estos comentarios refuerzan la pertinencia del enfoque centrado en el usuario y validan la utilidad práctica de la solución en el contexto real del centro.

En conjunto, los resultados muestran que la aplicación cumple los objetivos planteados y que las dificultades detectadas son acotadas y abordables en futuras iteraciones, sin comprometer la viabilidad ni el aporte principal del proyecto.

7. Discusión y tendencias

7.1. Contrastación con el estado del arte

La contrastación con el estado del arte permite posicionar la solución propuesta frente a plataformas consolidadas de gestión del entrenamiento como Trainerize [7], TrueCoach [8], TrainingPeaks [9] y StrengthLog [10]. Estas aplicaciones representan enfoques predominantes en el mercado actual y se orientan principalmente al entrenamiento remoto o al uso autónomo por parte del alumno, donde el rol del entrenador se limita a la planificación previa y a la revisión posterior de resultados. En este contexto, la gestión del entrenamiento ocurre fuera del espacio físico del gimnasio y no considera la supervisión simultánea de varios alumnos durante una sesión presencial, en línea con las tendencias dominantes del fitness digital [11].

En contraste, la solución desarrollada aborda un problema distinto al situar el foco en la gestión operativa del entrenamiento presencial y en la reducción de la carga cognitiva del entrenador durante la sesión. La aplicación móvil se utiliza como herramienta activa en sala, permitiendo planificar, ajustar y cerrar sesiones mientras el entrenamiento se desarrolla. Un elemento diferenciador es el registro de la ejecución real de la sesión, que en este caso es realizado directamente por el entrenador al finalizar el entrenamiento, asegurando coherencia entre lo planificado y lo ejecutado dentro del contexto presencial.

La usabilidad constituye otro factor distintivo. Mientras las aplicaciones del estado del arte están diseñadas para interacciones asincrónicas y uso individual, la solución propuesta prioriza flujos rápidos y vistas resumidas, pensadas para contextos de alta demanda operativa en los que el entrenador supervisa a varios alumnos en paralelo, alineándose con principios de usabilidad móvil orientados a tareas breves y repetitivas [3]. El dashboard web actúa como un complemento a la aplicación móvil, apoyando la visualización y análisis de la información histórica sin reemplazar la operación en sala, reforzando así el enfoque presencial de la solución y delimitando claramente su aporte frente a las alternativas existentes.

7.2. Alineación con tendencias emergentes

La solución desarrollada se inserta en el contexto actual de la ingeniería informática aplicada al deporte, alineándose con tendencias orientadas a la digitalización, el uso sistemático de datos y la mejora de la toma de decisiones en entornos presenciales de entrenamiento[12], [13].

7.2.1. Salud digital y entrenamiento basado en datos

El sistema se enmarca dentro del enfoque de Salud Digital al transformar procesos manuales en información digital estructurada. La gestión de clientes, sesiones y registros de entrenamiento pasa desde soportes físicos y memoria informal a una base de datos relacional centralizada. Este cambio habilita un enfoque de entrenamiento basado en datos, donde el entrenador puede apoyar sus decisiones en el historial de sesiones, el cumplimiento registrado y la evolución del trabajo realizado por cada alumno, en línea con enfoques actuales de digitalización del entrenamiento [11], [13].

Este enfoque reduce la dependencia exclusiva de la intuición y favorece la trazabilidad del proceso de entrenamiento. La disponibilidad inmediata de la información permite evaluar continuidad, adherencia y carga de trabajo de forma objetiva. De esta manera, prácticas asociadas al entrenamiento basado en datos se integran en un contexto de gimnasio funcional, sin requerir infraestructura avanzada ni procesos complejos.

7.2.2. Modelo de fitness híbrido

La solución refuerza un modelo de fitness híbrido en el que la interacción presencial sigue siendo el eje central del entrenamiento. La tecnología actúa como apoyo operativo durante la sesión, sin desplazar el rol del entrenador ni la supervisión directa del alumno. La aplicación móvil acompaña la planificación, el registro y el cierre de la sesión mientras el entrenamiento se desarrolla en sala.

Este enfoque responde a una tendencia actual que combina presencia física y soporte digital. El sistema mejora la eficiencia operativa del entrenador y ordena la información sin interferir en la dinámica del entrenamiento. La interacción humana se mantiene como componente principal, mientras la tecnología opera de forma transparente como herramienta de apoyo.

7.2.3. Preparación para ecosistemas conectados

Desde la perspectiva de la arquitectura de software, el proyecto adopta un diseño en la nube y una estructura de datos modular. Esta decisión prepara al sistema para integrarse, en el futuro, con ecosistemas

tecnológicos más amplios. Aunque la versión actual se basa en el ingreso manual de información para mantener el control del entrenador, la separación entre frontend y backend y el uso de una API estructurada permiten incorporar nuevas fuentes de datos si el contexto lo requiere.

En este sentido, el sistema se encuentra alineado con la evolución hacia entornos conectados, como los asociados al Internet of Medical Things. La arquitectura definida facilita una posible integración futura con sensores o dispositivos, sin comprometer el funcionamiento actual ni el enfoque presencial del proyecto[14].

7.3. Impacto del proyecto

7.3.1. Impacto económico y productividad

La implementación del sistema introduce un cambio estructural en la gestión operativa del centro, transitando desde un registro manual no estructurado hacia una gestión digital centralizada. El impacto económico principal se relaciona con el costo de oportunidad asociado al tiempo administrativo del entrenador, quien anteriormente debía dedicar parte relevante de su jornada a tareas de planificación y reconstrucción de información.

Antes de la incorporación del sistema, la planificación de sesiones y la revisión de entrenamientos previos dependían de la memoria del entrenador y del uso de cuadernos físicos, lo que dificultaba la recuperación del historial y aumentaba el tiempo invertido en tareas administrativas. Con la solución implementada, la información se encuentra disponible de forma inmediata y las sesiones pueden reutilizarse y ajustarse sin repetir procesos manuales, optimizando el flujo de trabajo diario.

De manera conservadora, la digitalización permite reducir el tiempo destinado a planificación y revisión administrativa en un rango aproximado de cinco a diez minutos por sesión. Este tiempo recuperado se traduce en una disminución del esfuerzo operativo y libera horas de trabajo que pueden ser redirigidas a la supervisión del entrenamiento, fortaleciendo la continuidad del seguimiento individual y habilitando una mayor capacidad operativa del centro sin incrementar proporcionalmente la carga laboral ni requerir apoyo administrativo adicional.

7.3.2. Impacto ambiental y eficiencia tecnológica

El proyecto se alinea con un enfoque de transformación digital sostenible a través de la desmaterialización de procesos y el uso eficiente de la infraestructura tecnológica.

En primer lugar, la sustitución del registro en papel por un sistema digital elimina el uso de cuadernos y soportes físicos. Este cambio reduce el consumo de insumos y facilita la conservación de la información en el tiempo, evitando pérdidas y duplicaciones propias del registro manual.

En segundo lugar, la arquitectura adoptada favorece la eficiencia energética. El sistema utiliza servicios en la nube gestionados en lugar de infraestructura local dedicada. Este enfoque permite que los recursos computacionales se utilicen bajo demanda, sin mantener servidores físicos activos de forma permanente. De esta manera, el consumo energético se ajusta al uso real del sistema y se evita el gasto asociado a equipos encendidos sin carga operativa.

Estas decisiones de diseño responden a criterios actuales de eficiencia tecnológica y uso responsable de recursos, coherentes con prácticas modernas de ingeniería de software orientadas a reducir el impacto ambiental de las soluciones digitales.

7.3.3. Implicancias éticas y protección de datos

El diseño del sistema considera la responsabilidad ética asociada al manejo de información personal en un contexto de acondicionamiento físico. El tratamiento de los datos se rige por principios de finalidad y seguridad, orientados a proteger la información de los alumnos y a limitar su uso al ámbito estrictamente necesario para la gestión del entrenamiento.

El sistema se enmarca en lo dispuesto por la Ley N° 19.628 sobre Protección de la Vida Privada. Se aplica el principio de minimización de datos, almacenando únicamente información básica y registros de entrenamiento necesarios para la prescripción y el seguimiento técnico del ejercicio. No se recolectan ni almacenan datos sensibles de salud que no sean pertinentes para la operación del sistema.

Desde una perspectiva profesional, la solución se concibe como una herramienta de apoyo a la labor del entrenador. El sistema no reemplaza el criterio experto, sino que respalda la toma de decisiones mediante un historial estructurado y consultable. La privacidad de la información se protege mediante el control físico del dispositivo utilizado y mecanismos de seguridad a nivel de base de datos. Este enfoque asegura que la digitalización respete la confidencialidad del alumno y mantenga la confianza propia de la relación entre entrenador y usuario.

7.4. Limitaciones del estudio

El estudio presenta limitaciones asociadas al alcance definido y al contexto específico en el que se desarrolló la solución. La evaluación de usabilidad se realizó con un único participante, correspondiente

al dueño y entrenador principal del centro, decisión coherente con el carácter no masivo del sistema y su uso en un entorno operativo particular. Si bien el participante es un usuario real y habitual, los resultados obtenidos no pueden generalizarse a otros perfiles de usuario ni a centros con dinámicas distintas, delimitando el alcance de las conclusiones.

Asimismo, la evaluación se concentra en sesiones acotadas y no considera un análisis longitudinal del uso del sistema. Esto impide observar efectos de aprendizaje, cambios en la percepción de usabilidad a largo plazo o impactos sostenidos en la operación del centro. El sistema evaluado corresponde a un prototipo funcional enfocado en los procesos centrales del entrenamiento presencial, excluyendo funcionalidades como operación sin conexión, integración con dispositivos externos o análisis automatizado avanzado. Finalmente, el análisis se centra exclusivamente en la aplicación móvil como principal artefacto de interacción, considerando el dashboard web solo como apoyo funcional y no como parte de la evaluación de usabilidad. Estas limitaciones establecen un marco claro para interpretar los resultados y fundamentan las proyecciones de mejora y trabajo futuro.

8. Conclusiones y trabajo futuro

8.1. Conclusiones generales

El proyecto cumplió el objetivo general propuesto al optimizar la gestión del entrenamiento en un centro deportivo mediante una aplicación móvil orientada al trabajo presencial del entrenador. La solución integra una interfaz clara, visualización del progreso y un modelo de datos estructurado que respalda la planificación y el registro de las rutinas.

El modelo relacional definido permitió separar planificación y ejecución, mantener un historial consistente por cliente y facilitar la reutilización de sesiones. La evaluación de usabilidad mostró que los flujos principales se ejecutan de forma directa en un contexto real, sin interferir con la dinámica del entrenamiento presencial, reduciendo la dependencia de registros manuales y de la memoria del entrenador.

El sistema se consolida como un prototipo funcional validado en terreno, con viabilidad técnica y utilidad práctica dentro de su alcance. Desde la ingeniería informática, el trabajo demuestra la integración efectiva de desarrollo móvil, modelado de datos y servicios en la nube para resolver un problema concreto del ámbito del entrenamiento presencial.

8.2. Aprendizaje autónomo y desafíos técnicos

El desarrollo del proyecto implicó un proceso sostenido de aprendizaje autónomo orientado a la resolución de un problema real mediante herramientas de ingeniería informática. Este aprendizaje se centró en tecnologías y enfoques que no formaban parte del trabajo previo del autor y que debieron ser abordados mediante estudio independiente y experimentación práctica, reforzando competencias asociadas al aprendizaje continuo y a la adaptación técnica frente a contextos reales.

Uno de los principales aprendizajes se relaciona con el desarrollo de aplicaciones móviles utilizando Flutter, adquiriendo competencias en la construcción de interfaces, manejo de navegación, control de estado y validación de flujos orientados a un uso intensivo en contexto presencial. De manera complementaria, la integración frontend-backend mediante Supabase permitió comprender el uso de bases de datos PostgreSQL gestionadas, el consumo de APIs y la implementación de operaciones CRUD, fortaleciendo la comprensión de arquitecturas cliente-servidor desacopladas del dispositivo. El modelado de datos relacional aplicado representó un desafío técnico relevante, especialmente en la separación entre planificación y ejecución del entrenamiento, validando en la práctica conceptos de normalización y consistencia de la información.

Como complemento, se desarrolló un dashboard básico utilizando Streamlit, lo que permitió reforzar la comprensión del ciclo completo de los datos, desde su generación en la aplicación móvil hasta su visualización y análisis en un entorno web. Un aspecto clave del aprendizaje fue la vivencia directa del problema abordado, mediante la participación activa del autor en el centro de entrenamiento, lo que facilitó el levantamiento de requerimientos y orientó las decisiones técnicas hacia necesidades reales. En conjunto, este proceso evidencia la capacidad de integrar nuevas tecnologías y aplicar conocimientos de ingeniería informática para resolver un problema práctico de manera contextualizada.

8.3. Trabajo futuro

El desarrollo del proyecto abre diversas posibilidades de evolución orientadas principalmente a la mejora funcional del sistema y a su adaptación a escenarios operativos más amplios. Estas proyecciones se plantean de forma coherente con el diseño actual y sin modificar el problema central abordado.

Una línea prioritaria de trabajo futuro corresponde a la incorporación de un modo de operación sin conexión a internet, que permita registrar sesiones y actividades en ausencia de conectividad y sincronizar los datos cuando ésta se restablezca. Esta mejora resultaría especialmente relevante para mantener la continuidad operativa del sistema ante fallas de red.

Otra extensión funcional relevante es la incorporación de un sistema administrativo más personalizado, que permita definir configuraciones específicas por cliente y generar notificaciones asociadas a eventos relevantes del entrenamiento. Este tipo de funcionalidades podría apoyar la planificación, el seguimiento y la comunicación interna del entrenador, sin aumentar la carga operativa durante la sesión presencial.

Asimismo, el sistema puede evolucionar para soportar la gestión de múltiples entrenadores dentro de un mismo centro, manteniendo la centralización de los datos y el historial de los clientes. Esta extensión permitiría escalar la solución a centros con mayor dotación de profesionales, sin alterar la estructura principal del modelo de datos.

Desde una perspectiva de análisis, el historial acumulado de sesiones habilita futuras líneas de desarrollo orientadas a la generación de indicadores más detallados de progreso y adherencia. De manera conceptual, estos datos podrían utilizarse como base para explorar mecanismos de apoyo a la toma de decisiones, incluyendo el uso de técnicas de análisis automatizado, sin que ello constituya un compromiso de implementación dentro del alcance actual.

En conjunto, estas proyecciones refuerzan que la solución desarrollada no es un sistema cerrado, sino una base técnica sólida sobre la cual es posible construir mejoras funcionales y explorar desarrollos aplicados, manteniendo un enfoque realista y alineado con las necesidades del entrenamiento presencial supervisado.

9. Bibliografía

- [1] S. L. Halson, "Monitoring training load to understand fatigue in athletes," *Sports Medicine*, vol. 44, no. Suppl 2, pp. 139–147, 2014. doi: 10.1007/s40279-014-0253-z.
- [2] International Organization for Standardization, *Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts*, ISO 9241-11:2018, 2018.
- [3] J. Nielsen and R. Budiu, *Mobile Usability*. Berkeley, CA, USA: New Riders, 2012.
- [4] J. Sauro, "Measuring Perceived Usability: The SUS, UMUX-LITE, and Net Promoter Score," Nielsen Norman Group, Feb. 2011. [Online]. Available: <https://www.nngroup.com/articles/measuring-perceived-usability/>. [Accessed: Jan. 2026].
- [5] S. Few, *Information Dashboard Design: The Effective Visual Communication of Data*, 2nd ed. Burlingame, CA, USA: Analytics Press, 2013.

- [6] J. Goudsmit et al., “Co-operative design of a coach dashboard for training monitoring and feedback,” *Sensors*, vol. 22, no. 23, Art. no. 9073, 2022. [Online]. Available: <https://doi.org/10.3390/s22239073>
- [7] Trainerize, “Trainerize Features and Coach API,” 2024. [Online]. Available: <https://www.trainerize.com/features>. [Accessed: Jan. 2026].
- [8] TrueCoach, “TrueCoach for Personal Trainers,” 2024. [Online]. Available: <https://truecoach.co>. [Accessed: Jan. 2026].
- [9] TrainingPeaks, “The Complete Solution for Coaches,” 2024. [Online]. Available: <https://www.trainingpeaks.com/>. [Accessed: Jan. 2026].
- [10] StrengthLog, “StrengthLog: The Workout Tracker,” 2024. [Online]. Available: <https://www.strengthlog.com/>. [Accessed: Jan. 2026].
- [11] S. Angosto, J. García-Fernández, M. Chavarrias, and M. Grimaldi-Puyana, “Segmentation of fitness users based on health consciousness: Implications for digital sport management,” *Quality & Quantity*, pp. 1–20, 2025. [Online]. Available: <https://doi.org/10.1007/s11135-025-02128-4>
- [12] A. M. Newsome et al., “2024 ACSM worldwide fitness trends: Future directions of the health and fitness industry,” *ACSM’s Health & Fitness Journal*, vol. 28, no. 1, pp. 14–26, Jan./Feb. 2024. [Online]. Available: <https://doi.org/10.1249/FIT.0000000000000933>
- [13] D. C. Mănescu, “Big Data Analytics Framework for Decision-Making in Sports Performance Optimization,” *Data*, vol. 10, no. 7, Art. no. 116, 2025. [Online]. Available: <https://doi.org/10.3390/data10070116>
- [14] D. Nishad and D. Tripathi, “Internet of medical things (IoMT): Applications and challenges,” *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 11, no. 3, pp. 2885–2889, 2020. [Online]. Available: <https://doi.org/10.61841/turcomat.v11i3.14654>

10. Anexos

El presente capítulo recopila el material complementario que sirve de soporte documental y técnico al desarrollo de la memoria. Su propósito es profundizar en detalles específicos que, debido a su extensión o nivel de granularidad, fueron abstraídos de los capítulos principales para facilitar la fluidez de la lectura.

[1] Dashboard web: <https://dashboard-tesis-434gwgptkndxjrvvzeqs2o.streamlit.app/>

[2] Detalle de Resultados Escala de Usabilidad del Sistema (SUS):
<https://docs.google.com/document/d/1bBXUveVB2mUfeactfXmYuocv4dVFkBzaws-7BedsdRE/edit?usp=sharing>