



UNIVERSITAT DE
BARCELONA

Proyecto final de carrera

GRADO EN INGENIERÍA
INFORMÁTICA

Facultad de Matemáticas e Informática
Universidad de Barcelona

FITPROX
Desarrollo de una App de Gestión
de Entrenamientos y Planes
Nutricionales

Autor: Pedro Sendon

Tutor: Xavier Baró

Realitzado en: Departament de
Matemàtiques i Informàtica

Barcelona, Enero 2025

Abstract

This Final Degree Project aims to create a web application designed to deliver personalized training programs and macronutrient recommendations tailored to the unique goals and characteristics of each user. By utilizing a detailed form, the application gathers essential information to generate comprehensive training schedules and accurate nutritional guidance, ensuring recommendations are customized to individual needs. The overarching objective is to provide an all-in-one solution that empowers users to meet their fitness and wellness targets through a system of tailored and profile-specific advice.

Resum

El propòsit d'aquest Treball de Fi de Grau és desenvolupar una aplicació web que ofereixi recomanacions personalitzades d'entrenament i macronutrients, basades en les característiques i objectius individuals de cada usuari. A través d'un formulari, l'aplicació recopilarà informació rellevant per a generar plans d'entrenament detallats i recomanacions precises de macronutrients adaptades a les necessitats específiques de cada persona. L'objectiu principal és proporcionar una solució integral que faciliti als usuaris aconseguir les seves metes de salut i benestar mitjançant recomanacions ajustades al seu perfil.

Resumen

El propósito de este Trabajo de Fin de Grado es desarrollar una aplicación web que ofrezca recomendaciones personalizadas de entrenamiento y macronutrientes, basadas en las características y objetivos individuales de cada usuario. A través de un formulario, la aplicación recopilará información relevante para generar planes de entrenamiento detallados y recomendaciones precisas de macronutrientes adaptadas a las necesidades específicas de cada persona. El objetivo principal es proporcionar una solución integral que facilite a los usuarios alcanzar sus metas de salud y bienestar mediante recomendaciones ajustadas a su perfil.

Expresiones de gratitud

Quiero expresar mi más sincero agradecimiento a mi tutor, Xavier Baró, por su apoyo constante, paciencia, dedicación y valiosa guía a lo largo de la realización de este Trabajo de Fin de Grado. Su confianza en mí, así como sus conocimientos y consejos, han sido fundamentales para la culminación exitosa de este proyecto. Gracias por todo su tiempo y esfuerzo.

Índice

1. INTRODUCCIÓN	2
1.1. Objetivos	2
1.2. Motivación	2
1.3. Estructura de la memoria	3
1.4. Planificación	4
2. ANÁLISIS Y DISEÑO	6
2.1. Metodología de Trabajo	6
2.2. Requisitos	8
2.3. Flujo de Gestión de Usuarios y Asignación de Planes	19
2.4. Diseño de la Arquitectura del Sistema	22
2.5. Evolución de los Diseños de las Vistas	25
3. TECNOLOGÍAS	26
3.1. Tecnologías de Desarrollo Web	26
3.2. Despliegue de la Aplicación	30
4. IMPLEMENTACIÓN	32
4.1. Gestión de Funcionalidades	32
5. PRUEBAS	42
5.1. Pruebas de usuario	42
5.2. Pruebas automatizadas	42
5.3. Errores encontrados	44
5.4. Sugerencias de mejora	45
6. RESULTADOS	47
6.1. Resultados obtenidos	47
6.2. Cambios implementados tras las pruebas	52
7. CONCLUSIONES	54
8. LÍNEAS DE DESARROLLO FUTURO	57
8.1. Biblioteca de alimentos	57
8.2. Sistema de chat con entrenadores y nutricionistas	57
8.3. Mejoras en ciberseguridad	57
8.4. Optimización de la experiencia del usuario (UX)	58
8.5. Escalabilidad y rendimiento	58
8.6. Recuperación de contraseña	58
8.7. Mantener sesión iniciada	58
8.8. Selección personalizada de días de entrenamiento	58
8.9. Inicio de sesión con Google, Microsoft y Facebook	59
8.10. Soporte multilingüe en datos generados por el usuario	59
9. ANEXO	60

Índice de figuras

1.	Diagrama de Gannt del proyecto FitProx	5
2.	Diagrama del flujo de vistas según el estado del cliente.	21
3.	Diagrama de Navegación de la Aplicación FitProx	23
4.	Diagrama Entidad-Relación (MER) de la aplicación FitProx	25
5.	Diagrama de la arquitectura del proyecto	31
6.	Interfaz del cambio de idioma con banderas representativas.	39
7.	Informe de cobertura de las pruebas unitarias del backend.	49
8.	Vista del plan de entrenamiento semanal.	52
9.	Vista del plan nutricional semanal.	52
10.	Vista del progreso del usuario.	52
11.	Vista de la lista de usuarios desde Administrador.	52
12.	Diseño inicial de la pantalla para Cambiar Contraseña.	81
13.	Vista de Edición de Perfil.	81
14.	Vista de Ejercicio.	81
15.	Pantalla de Entrenamiento.	81
16.	Vista de Entreno por Día.	82
17.	Formulario inicial de registro.	82
18.	Vista de Información del Ejercicio.	82
19.	Pantalla de Inicio de Sesión.	82
20.	Pantalla de Macronutrientes.	83
21.	Vista principal de la aplicación.	83
22.	Pantalla de Perfil.	83
23.	Pantalla de Registro.	83

1. INTRODUCCIÓN

En el ámbito del *fitness* y la *salud*, el acceso a información personalizada y ajustada a las necesidades individuales ha cobrado una relevancia creciente. Las aplicaciones tecnológicas han jugado un papel clave en este proceso, facilitando a los usuarios el seguimiento de sus rutinas de entrenamiento y nutrición. Sin embargo, la mayoría de las herramientas disponibles ofrecen recomendaciones genéricas, que no siempre reflejan las metas, el nivel de condición física ni las particularidades de cada individuo.

El propósito de este *Trabajo de Fin de Grado* (TFG) es desarrollar una aplicación web centrada en el mundo del *fitness*, que proporcione recomendaciones personalizadas de entrenamiento y nutrición. Esta aplicación busca mejorar la experiencia del usuario mediante la creación de planes específicos adaptados a las *características personales y objetivos particulares* de cada individuo.

1.1. Objetivos

El objetivo principal de este *TFG* es desarrollar una aplicación web capaz de ofrecer recomendaciones personalizadas en el ámbito del *fitness*. Esta aplicación permitirá a los usuarios acceder a planes de entrenamiento y nutrición adaptados a sus metas, que se definirán mediante un proceso de registro y la cumplimentación de un formulario detallado sobre sus objetivos físicos y su estado actual. Los principales objetivos del proyecto son:

- *Proporcionar planes de entrenamiento semanales detallados*, con listas de ejercicios y explicaciones claras.
- *Ofrecer recomendaciones nutricionales basadas en los macronutrientes* que el usuario debe consumir, representados visualmente mediante gráficos.
- *Crear una interfaz de usuario sencilla e intuitiva*, que permita navegar fácilmente entre las distintas secciones de la aplicación (entrenamiento, macronutrientes, perfil).
- *Facilitar el seguimiento y actualización de los objetivos físicos del usuario*, promoviendo su progreso hacia un estado de bienestar integral.

1.2. Motivación

La motivación detrás de este proyecto surge de mi experiencia personal en el ámbito del *fitness* y la *nutrición*. Durante los últimos años, he trabajado intensamente con preparadores físicos y nutricionistas para optimizar mis entrenamientos y dietas con el fin de alcanzar mis metas de salud. En este proceso, he experimentado de primera mano la importancia de contar con planes personalizados, ajustados no solo a mis características físicas, sino también a mis objetivos específicos.

He observado que muchas personas, al igual que yo, enfrentan dificultades para encontrar soluciones adecuadas y efectivas que se adapten a sus necesidades individuales. Las aplicaciones actuales suelen ser demasiado genéricas, ofreciendo

recomendaciones que no siempre reflejan las particularidades del usuario. Mi intención es crear una herramienta que solvete este problema, proporcionando planes personalizados y fácilmente accesibles que puedan ser seguidos de manera intuitiva. Con esta aplicación, espero ayudar a otros a optimizar su camino hacia sus metas de salud y bienestar, facilitando un enfoque integral y personalizado.

1.3. Estructura de la memoria

La memoria de este proyecto se organiza en ocho capítulos, que abarcan desde los antecedentes y planificación del desarrollo hasta la implementación, pruebas, resultados, conclusiones y líneas de desarrollo futuro. A continuación, se presenta un resumen de cada capítulo:

- **Introducción:** Describe los *objetivos* del proyecto y la *motivación* detrás de la creación de *FitProx*. También incluye una descripción de la *estructura de la memoria* y un apartado sobre la *planificación*, destacando las fases clave del desarrollo mediante un diagrama de Gantt.
- **Análisis y Diseño:** Se detalla la *metodología de trabajo*, incluyendo herramientas como *casos de uso* y *historias de usuario*. También se describen los *requisitos funcionales y no funcionales*, el *flujo de gestión de usuarios* y la *asignación de planes*, además del *diseño de la arquitectura del sistema* con diagramas y el *Modelo Entidad-Relación (MER)*. Este capítulo concluye con la *evolución de los diseños de las vistas*.
- **Tecnologías:** Se explican las *tecnologías utilizadas* para el desarrollo web (*React, FastAPI, Django*) y para el despliegue de la aplicación, destacando las herramientas adicionales empleadas.
- **Implementación:** Este capítulo detalla la *gestión de funcionalidades*, mostrando cómo se desarrollaron las principales características de *FitProx*.
- **Pruebas:** Se presentan las *pruebas de usuario*, las *pruebas automatizadas*, y los *errores encontrados* durante el desarrollo. Además, se incluyen sugerencias de mejora basadas en las pruebas realizadas.
- **Resultados:** En este capítulo se muestran los *resultados obtenidos* tras las pruebas, junto con los *cambios implementados* para optimizar el funcionamiento de la aplicación.
- **Conclusiones:** Ofrece una *reflexión final* sobre el cumplimiento de los objetivos planteados y el aprendizaje adquirido durante el proyecto.
- **Líneas de Desarrollo Futuro:** Describe posibles ampliaciones y mejoras, como la inclusión de una *biblioteca de alimentos*, un *sistema de chat*, mejoras en *ciberseguridad*, y soporte para *inicio de sesión con redes sociales*, entre otras.

1.4. Planificación

El *diagrama de Gantt* presenta la planificación del proyecto, dividida en fases a lo largo de 15 semanas [1]. Durante las primeras tres semanas (1-3), me dediqué al *estudio de tecnologías* y a la *identificación de usuarios*, realizando una investigación sobre las herramientas a utilizar y un análisis de las necesidades de los futuros usuarios.

En las semanas 4 y 5, se definieron los *requisitos* y se crearon *historias de usuario*, proporcionando una base sólida para el desarrollo. A partir de la semana 6, se inició el *desarrollo de la aplicación* junto con el *diseño de pantallas e interfaz de usuario*, un proceso que se extendió hasta la semana 10.

La semana 11 estuvo dedicada a las *pruebas iniciales de calidad*, con ajustes implementados según los resultados obtenidos. Las semanas 12 a 15 se centraron en la *evaluación de los resultados* y la *redacción de la memoria*.

Gracias a esta planificación secuencial, el proyecto avanzó de manera eficiente, permitiendo completar cada fase antes de pasar a la siguiente.

2. ANÁLISIS Y DISEÑO

2.1. Metodología de Trabajo

El desarrollo de la aplicación *FitProx* se ha llevado a cabo bajo un enfoque ágil y bien organizado, utilizando diferentes herramientas y métodos para asegurar un avance continuo y eficiente en el proyecto. Al tratarse de un trabajo en solitario, se ha optado por un enfoque flexible y adaptable que ha permitido gestionar el tiempo y los recursos de manera eficiente, mientras que las tutorías periódicas han brindado la orientación necesaria para mantener el proyecto alineado con los objetivos principales [2].

Tutorías periódicas

El contacto continuo con mi tutor, con reuniones cada 2-3 semanas, ha sido clave para validar avances, resolver problemas y ajustar el rumbo del proyecto. La **retroalimentación constante** permitió realizar cambios oportunos, evitando desviaciones respecto a los objetivos originales [3].

Diagrama de Gantt

Desde el inicio del proyecto, se empleó un *diagrama de Gantt* para organizar y gestionar las tareas. Esta herramienta permitió dividir el trabajo en **fases claras y alcanzables**, visualizar el flujo de tareas y marcar hitos clave a lo largo del tiempo [4]. Su uso facilitó una planificación estructurada, identificando tiempos de entrega y controlando el progreso de cada etapa [5].

Casos de uso e historias de usuario

Para guiar el desarrollo de las funcionalidades, se elaboraron *casos de uso* e *historias de usuario*, herramientas clave para definir requisitos y expectativas del sistema [6]. Las *historias de usuario* garantizaron un enfoque centrado en el usuario, asegurando que cada funcionalidad desarrollada respondiera a sus necesidades [7].

Como ejemplo, a continuación, se presenta una historia de usuario sobre el proceso de registro en la aplicación.

FitProx:

Historia de Usuario: Registro en la Aplicación

- **Como:** Visitante de la aplicación.
- **Quiero:** Registrarme.
- **Para:** Utilizar los servicios de la aplicación.

Criterios de Aceptación:

1. El usuario presiona *Registrarse* en la pantalla de bienvenida.

2. Se solicita un *correo electrónico* válido y no registrado previamente.
3. Se ingresan *nombre y apellidos* sin números ni símbolos especiales.
4. La *contraseña* debe tener al menos 8 caracteres, incluyendo una letra mayúscula, una minúscula, un número y un símbolo.
5. La *fecha de nacimiento* debe estar en formato dd/mm/yyyy y validar que el usuario sea mayor de 18 años y menor de 120.
6. Se selecciona el *sexo* (M: Masculino, F: Femenino, O: Otro).
7. Es obligatorio aceptar los *términos y condiciones*.
8. Se verifican los campos, mostrando errores si hay datos inválidos.
9. Una vez validados todos los campos, se habilita el botón *Registrarse* y se procesa el registro.
10. Tras el registro exitoso, el usuario es redirigido a un formulario sobre *objetivos de salud o entrenamiento*.

El resto de las historias de usuario elaboradas para la aplicación *FitProx* se encuentran detalladas en el Anexo 9, donde se especifican las funcionalidades principales y los requisitos de cada una.

Esquemas de Endpoints

Además de los casos de uso e historias de usuario, se crearon esquemas preliminares para los endpoints del sistema, que, aunque no definitivos, sirvieron como base sólida para estructurar las funcionalidades del backend de forma clara y organizada. Definir los endpoints antes de programar permitió implementar las funcionalidades de manera más eficiente y ordenada, asegurando que todas las rutas necesarias estuvieran bien planteadas desde el inicio.

Los esquemas se encuentran en el Anexo 9, donde se detallan los métodos y parámetros considerados en la planificación inicial. Aunque algunos endpoints se ajustaron durante el desarrollo, su diseño previo evitó confusiones y garantizó una coherencia estructural a lo largo del proyecto.

Gestión del código y control de versiones

Para la gestión del código fuente y control de versiones, se utilizó *GitHub*, que permitió registrar de manera estructurada los cambios realizados, facilitando el retorno a versiones anteriores en caso de errores [8]. Además, el uso de ramas separadas organizó el desarrollo y las pruebas de nuevas funcionalidades sin comprometer la estabilidad de la aplicación [9].

Flexibilidad y adaptación

La *flexibilidad* fue clave en la metodología. Aunque se definió un plan detallado desde el inicio, la naturaleza del desarrollo implicó ajustes e imprevistos [10]. La combinación de *tutorías regulares* con una planificación estructurada, pero adaptable, permitió responder a los cambios sin perder de vista los objetivos finales [2].

2.2. Requisitos

Requisitos funcionales

En esta sección se abordarán los *requisitos funcionales* de la aplicación *FitProx*. Estos requisitos representan las funcionalidades clave que la aplicación debe cumplir para satisfacer las necesidades de los usuarios y asegurar un rendimiento eficiente y efectivo. Se han definido a partir de *historias de usuario* que reflejan los objetivos y expectativas del usuario final, garantizando así un enfoque centrado en sus necesidades. Cada historia de usuario especifica una funcionalidad particular de la aplicación y, a través de estas, se han concretado los aspectos que guían el desarrollo del proyecto. Los requisitos funcionales detallados a continuación describen cómo la aplicación debe comportarse y qué características debe ofrecer para proporcionar una experiencia completa y personalizada a los usuarios en el ámbito del *fitness* y la *nutrición*.

Registro de Usuarios

La funcionalidad de *Registro de Usuarios* debe cumplir con los siguientes requisitos para garantizar un acceso seguro y personalizado a la plataforma:

- El sistema debe permitir a los usuarios crear una cuenta proporcionando los siguientes datos básicos:
 - Correo electrónico válido.
 - Nombre completo.
 - Contraseña que cumpla con los requisitos de seguridad.
 - Fecha de nacimiento.
 - Sexo.
- Los usuarios deben aceptar los términos y condiciones antes de completar el registro.
- Todos los campos deben ser obligatorios y validados antes de permitir que el usuario continúe.
- Tras completar el registro básico, los usuarios deben rellenar un formulario adicional con información sobre:
 - Objetivos físicos (e.g., ganar masa muscular, perder peso).

- Nivel de actividad física actual.
- Métricas personales, como peso y altura.
- Los datos deben ser suficientes para personalizar los planes de entrenamiento y nutrición de manera eficiente.

Inicio de Sesión

La funcionalidad de *inicio de sesión* debe cumplir con los siguientes requisitos para garantizar un acceso seguro y personalizado a la plataforma:

- El sistema debe permitir a los usuarios acceder a su cuenta proporcionando:
 - *Correo electrónico* válido.
 - *Contraseña* previamente registrada.
- El sistema debe validar las credenciales ingresadas con las almacenadas en la base de datos.
- Si las credenciales son incorrectas, debe mostrar un mensaje de error específico para orientar al usuario.
- Después de una autenticación exitosa, el usuario debe ser redirigido al panel principal con acceso a sus funcionalidades personalizadas.

Visualización de Entrenamientos Semanales

La funcionalidad de *Visualización de Entrenamientos Semanales* debe cumplir con los siguientes requisitos para garantizar que los usuarios puedan gestionar y seguir sus rutinas personalizadas de manera eficiente:

- El sistema debe permitir a los usuarios visualizar un *plan de entrenamiento semanal* asignado manualmente por un entrenador basado en los objetivos y datos iniciales proporcionados durante el registro.
- Cada día de la semana debe incluir la siguiente información sobre los entrenamientos asignados:
 - Lista de ejercicios a realizar.
 - Número de *series* y *repeticiones* para cada ejercicio.
 - Parámetros adicionales, como el tiempo de descanso entre series.
- Los días de la semana deben ser interactivos, permitiendo al usuario seleccionar un día específico para ver los detalles del entrenamiento asignado.
- El sistema debe proporcionar información clara y relevante para ayudar al usuario a completar su rutina diaria de manera eficiente.

Lista de Ejercicios por Entrenamiento

La funcionalidad de *Lista de Ejercicios por Entrenamiento* debe cumplir con los siguientes requisitos para garantizar que los usuarios puedan seguir y gestionar sus entrenamientos diarios de manera eficiente:

- El sistema debe mostrar una lista detallada de los ejercicios asignados para cada día específico del plan semanal.
- Para cada ejercicio, debe incluirse la siguiente información:
 - *Nombre del ejercicio*: Indica el tipo de ejercicio a realizar.
 - *Número de series y repeticiones*: Especifica la cantidad de series y repeticiones requeridas.
- Los usuarios deben tener la opción de *marcar un ejercicio como completado*, actualizando automáticamente el progreso del día.
- El progreso del entrenamiento debe reflejarse como un *porcentaje* basado en la cantidad de ejercicios completados.
- Una vez completados todos los ejercicios del día, el usuario debe poder *marcar el entrenamiento como finalizado*.

Detalle de Ejercicios

La funcionalidad de *Detalle de Ejercicios* debe cumplir con los siguientes requisitos para garantizar que los usuarios puedan comprender y realizar los ejercicios de manera efectiva:

- El sistema debe proporcionar acceso a una vista detallada de cada ejercicio incluido en el plan de entrenamiento.
- Para cada ejercicio, la vista debe incluir la siguiente información:
 - *Descripción técnica* que explique cómo realizar el ejercicio.
 - Visualización de *imágenes o gráficos* que muestren la técnica correcta.
 - Información sobre el *grupo muscular* o grupos musculares trabajados en el ejercicio.
 - *Parámetros del ejercicio*, incluyendo:
 - Número de series y repeticiones.
 - Tiempo de descanso entre series.
- La información debe mostrarse de forma clara y estructurada para facilitar su seguimiento durante las sesiones.

Macronutrientes y Nutrición

La funcionalidad de *Macronutrientes y Nutrición* debe cumplir con los siguientes requisitos para proporcionar a los usuarios información clara y personalizada sobre sus necesidades nutricionales:

- La aplicación debe mostrar el **tipo de dieta recomendado**, como alta en proteínas, baja en carbohidratos, o balanceada, basado en los objetivos del usuario.
- Debe proporcionar **recomendaciones nutricionales** que incluyan:
 - Cantidad diaria recomendada de carbohidratos, proteínas y grasas, expresada en gramos y calorías.
 - Calorías totales diarias necesarias para alcanzar los objetivos del usuario.
- La aplicación debe sugerir una **distribución de comidas**, indicando cómo repartir los macronutrientes entre desayuno, almuerzo, cena y otras comidas, optimizando la ingesta diaria.

Visualización del Perfil

La funcionalidad de *Visualización del Perfil* debe cumplir con los siguientes requisitos para garantizar que los usuarios puedan gestionar y revisar su información personal y objetivos de manera eficiente:

- La sección debe permitir a los usuarios visualizar:
 - *Datos personales*: Nombre, correo electrónico, edad, altura, peso inicial y peso actual.
 - *Objetivos de entrenamiento y nutrición*: Objetivo principal (e.g., perder peso, ganar masa muscular), nivel de actividad física y frecuencia de entrenamiento semanal.
- El sistema debe proporcionar un botón para *editar perfil*, que permita a los usuarios modificar su información personal y objetivos.
- Debe incluir una funcionalidad de *cierre de sesión* para que los usuarios puedan desconectarse de la aplicación de manera sencilla y segura.

Edición de Información del Perfil

La funcionalidad de *Edición de Información del Perfil* debe cumplir con los siguientes requisitos para permitir a los usuarios mantener su información actualizada y adaptada a sus necesidades:

- Los usuarios deben poder modificar los siguientes datos personales:
 - Nombre.
 - Apellido.

- Peso actual.
- El sistema debe permitir la actualización de objetivos de entrenamiento y nutrición, incluyendo:
 - Objetivo de peso.
 - Nivel de actividad física.
 - Frecuencia de entrenamientos.
- Debe incluir una opción para cambiar la *contraseña*, garantizando un flujo seguro de validación.
- El sistema debe permitir la gestión de la *foto de perfil*, con la opción de subir una nueva imagen.
- Todos los cambios realizados deben ser validados antes de ser guardados en la base de datos.
- El usuario debe ser redirigido al perfil actualizado una vez completados los cambios.

Cambio de Contraseña: Requisitos

La funcionalidad de *Cambio de Contraseña* debe cumplir con los siguientes requisitos para garantizar un proceso seguro y claro para los usuarios:

- El sistema debe permitir a los usuarios cambiar su *contraseña* en cualquier momento desde la sección de perfil.
- Los usuarios deben proporcionar su **contraseña actual** para confirmar su identidad antes de proceder con el cambio.
- Los usuarios deben ingresar una **nueva contraseña** que cumpla con los siguientes requisitos de seguridad:
 - Mínimo 8 caracteres.
 - Al menos una letra mayúscula.
 - Al menos una letra minúscula.
 - Al menos un número.
 - Al menos un símbolo especial.
- Se debe confirmar la **nueva contraseña** ingresándola nuevamente para asegurar que coincide con la primera.
- Si cualquiera de las validaciones falla (contraseña actual incorrecta, nueva contraseña no válida, o confirmación incorrecta), el sistema debe mostrar mensajes de error claros para guiar al usuario.
- Una vez aceptados los cambios, el sistema debe actualizar la contraseña en la base de datos de manera segura.

Historial de Progreso

La funcionalidad de *Historial de Progreso* debe cumplir con los siguientes requisitos para permitir a los usuarios realizar un seguimiento detallado y motivador de su evolución física y actividad de entrenamiento:

■ Calendario y próximo entrenamiento:

- El sistema debe proporcionar un *calendario* que muestre los días en los que se han realizado entrenamientos y los programados para el futuro.
- Debe incluir información detallada del *próximo entrenamiento*, mostrando:
 - Nombre de la rutina.
 - Progreso realizado hasta el momento.
 - Tiempo restante hasta la próxima sesión.

■ Historial de entrenamientos completados y progreso del plan de entrenamiento:

- La aplicación debe registrar y mostrar un *historial* de los entrenamientos completados por el usuario.
- Debe incluir un indicador de *progreso del plan de entrenamiento* en porcentaje, basado en la cantidad de entrenamientos realizados en comparación con el total programado.

■ Seguimiento del peso corporal:

- La aplicación debe permitir al usuario registrar su *peso actual* periódicamente.
- Debe incluir una gráfica que muestre la evolución del peso a lo largo del tiempo.

■ Mensajes de motivación personalizados:

- El sistema debe generar *mensajes de motivación* basados en la cantidad de entrenamientos completados, para reforzar la adherencia del usuario a sus rutinas.

Gestión de Planes de Entrenamiento, Entrenamientos y Ejercicios

La funcionalidad de *Gestión de Planes de Entrenamiento, Entrenamientos y Ejercicios* debe cumplir con los siguientes requisitos para garantizar que el administrador pueda gestionar eficazmente los datos relacionados con los planes y entrenamientos personalizados:

■ Planes de entrenamiento:

- El sistema debe permitir al administrador *añadir nuevos planes de entrenamiento*, definiendo múltiples entrenamientos adaptados a los objetivos específicos de los usuarios.

- Debe incluir la funcionalidad de *modificar planes de entrenamiento*, permitiendo cambiar entrenamientos, objetivos y estructuras de los planes existentes.
 - El administrador debe poder *eliminar planes de entrenamiento* que ya no sean relevantes o útiles.
- **Entrenamientos:**
 - El sistema debe permitir al administrador *crear nuevos entrenamientos*, especificando ejercicios, series, repeticiones y parámetros adicionales.
 - Debe incluir la funcionalidad para *modificar entrenamientos* existentes, ajustando ejercicios, series y otros elementos según las necesidades.
 - El administrador debe poder *eliminar entrenamientos* específicos dentro de un plan.
 - **Ejercicios:**
 - El sistema debe proporcionar la capacidad de *añadir nuevos ejercicios* a la base de datos general, incluyendo detalles como el nombre, la descripción y los parámetros necesarios.
 - Debe incluir la funcionalidad para *modificar ejercicios* existentes, ajustando información como descripción o parámetros.
 - El administrador debe poder *eliminar ejercicios* de la base de datos general si ya no son necesarios.

Gestión de Planes de Macronutrientes

La funcionalidad de *Gestión de Planes de Macronutrientes* debe cumplir con los siguientes requisitos para garantizar que las recomendaciones nutricionales sean precisas y estén alineadas con los objetivos de los usuarios:

- **Añadir planes de macronutrientes:**
 - El sistema debe permitir al administrador *crear nuevos planes de macronutrientes*.
 - Cada plan debe incluir:
 - Cantidad total de *calorías*.
 - Valores específicos de *proteínas, carbohidratos y grasas*.
 - Asociación con un *tipo de dieta* (e.g., alta en proteínas, baja en carbohidratos).
- **Modificar planes de macronutrientes:**
 - El sistema debe permitir ajustar los valores existentes de:
 - Calorías y macronutrientes (proteínas, carbohidratos, grasas).
 - Descripción del plan.

- Asociación con tipos de dieta.

■ **Eliminar planes de macronutrientes:**

- El administrador debe poder *eliminar planes de macronutrientes* que ya no sean relevantes o efectivos.
- El sistema debe requerir una confirmación previa antes de eliminar un plan para evitar errores.

Gestión de Usuarios

La funcionalidad de *Gestión de Usuarios* debe cumplir con los siguientes requisitos para garantizar que el administrador pueda administrar eficientemente los perfiles de los usuarios de la aplicación:

■ **Añadir usuarios:**

- El sistema debe permitir al administrador *registrar nuevos usuarios* en el sistema, ingresando información básica como:
 - Nombre.
 - Correo electrónico.
 - Objetivos de entrenamiento o nutrición.
 - Nivel de actividad.

■ **Modificar usuarios:**

- El sistema debe permitir al administrador *actualizar información de usuarios existentes*, incluyendo:
 - Datos personales como nombre y correo.
 - Objetivos de entrenamiento y nutrición.
 - Nivel de actividad física.

■ **Eliminar usuarios:**

- El sistema debe permitir al administrador *eliminar usuarios* inactivos o irrelevantes para mantener la base de datos actualizada.
- Antes de la eliminación, el sistema debe requerir una confirmación para evitar errores.

■ **Asignar planes nutricionales o entrenamientos:**

- El sistema debe permitir al administrador *asignar planes nutricionales y de entrenamiento personalizados* a los usuarios.
- Los planes deben ser seleccionables desde una lista predefinida, alineados con los objetivos y necesidades del usuario.

Cambio de Idioma

La funcionalidad de *Cambio de Idioma* debe cumplir con los siguientes requisitos para garantizar que los usuarios puedan interactuar con la aplicación en su idioma preferido:

■ Selección de idioma:

- El sistema debe permitir a los usuarios seleccionar entre tres idiomas: *inglés*, *español* y *atalán*.
- La selección debe realizarse mediante un *menú desplegable* accesible desde un icono en la esquina superior derecha de la interfaz.
- Cada idioma debe estar representado con su respectiva bandera para facilitar la identificación visual.

■ Contexto de idioma:

- El sistema debe utilizar un *LanguageContext* o un mecanismo similar para gestionar las traducciones dinámicamente en toda la aplicación.
- Los textos de la interfaz deben cambiar de manera instantánea al seleccionar un nuevo idioma.

■ Almacenamiento de idiomas:

- Las traducciones deben estar almacenadas en archivos *JSON*, organizados por idioma.
- El sistema debe cargar únicamente el archivo correspondiente al idioma seleccionado, optimizando el rendimiento.

Asignación de Planes Pendientes a Usuarios

La funcionalidad de *Asignación de Planes Pendientes a Usuarios* debe cumplir con los siguientes requisitos para que administradores, entrenadores y nutricionistas gestionen eficientemente a los usuarios sin planes asignados:

■ Visualización de usuarios pendientes:

- El sistema debe mostrar una lista de usuarios pendientes de recibir planes, adaptada al rol del usuario:
 - *Entrenadores*: Usuarios sin plan de entrenamiento.
 - *Nutricionistas*: Usuarios sin plan de nutrición.
 - *Administradores*: Usuarios sin plan de entrenamiento, nutrición, o ambos.
- La lista debe incluir información básica del usuario, como nombre, correo electrónico y estado del plan.

■ Acceso a detalles del usuario y asignación de planes:

- El sistema debe redirigir al detalle del usuario seleccionado, mostrando su información personal y necesidades específicas.
 - Debe incluir un *selector de plan* para elegir un plan existente.
 - Debe proporcionar una opción para *crear un nuevo plan* personalizado, ajustado a las necesidades del usuario.
- **Confirmación de asignación:**
 - El sistema debe permitir confirmar la asignación del plan, actualizando automáticamente el estado del usuario.
 - Una vez asignado el plan, el usuario debe ser eliminado de la lista de pendientes.

Visualización Detallada de Entidades

La funcionalidad de *Visualización Detallada de Entidades* debe cumplir con los siguientes requisitos para permitir que los administradores, entrenadores y nutricionistas accedan a información completa y estructurada de las diversas entidades gestionadas en la aplicación:

- **Acceso directo desde las listas de entidades:**
 - El sistema debe permitir a los usuarios con roles administrativos seleccionar cualquier entidad desde las listas principales (ejercicios, entrenamientos, planes de nutrición, planes de entrenamiento y usuarios) para acceder a su vista detallada.
- **Visualización de detalles completos:**
 - Cada entidad debe mostrar una vista detallada con información específica:
 - *Ejercicio*: Nombre, descripción técnica, grupo muscular trabajado, imágenes o gráficos, número de series y repeticiones.
 - *Entrenamiento*: Lista de ejercicios, series, repeticiones y tiempo de descanso.
 - *Plan de nutrición*: Tipo de dieta, calorías, distribución de macronutrientes y recomendaciones diarias.
 - *Plan de entrenamiento*: Objetivos del plan, entrenamientos programados, frecuencia recomendada.
 - *Usuario*: Perfil completo, incluyendo datos personales y objetivos específicos.

Tipos de Usuarios y Gestión de Roles

La funcionalidad de *Tipos de Usuarios y Gestión de Roles* debe cumplir con los siguientes requisitos para garantizar que cada usuario acceda únicamente a las funcionalidades relevantes según su rol dentro de la plataforma:

- **Tipos de usuarios:**

- El sistema debe soportar cuatro tipos principales de usuarios: *cliente*, *administrador*, *entrenador* y *nutricionista*.

- **Gestión de permisos:**

- Cada rol debe tener permisos específicos para acceder y gestionar diferentes funcionalidades:
 - *Cliente*: Acceso limitado a entrenamientos, planes nutricionales, seguimiento de progreso y personalización de su perfil.
 - *Administrador*: Permiso completo para gestionar usuarios, entrenamientos, planes nutricionales y visualizar estadísticas.
 - *Entrenador*: Permiso para gestionar entrenamientos y ejercicios, y asignar planes de entrenamiento a los clientes.
 - *Nutricionista*: Permiso para gestionar planes nutricionales y asignarlos a los clientes.

- **Seguridad y restricciones:**

- Los entrenadores y nutricionistas no deben tener acceso a la gestión de usuarios.
- El cliente solo debe poder acceder a su propia información y funcionalidades relacionadas.

- **Interfaz basada en roles:**

- La interfaz debe adaptarse dinámicamente según el rol del usuario, mostrando solo las funcionalidades relevantes.

Requisitos no funcionales

Usabilidad

La aplicación debe cumplir con los siguientes requisitos de usabilidad:

- **Interfaz intuitiva:** La aplicación debe ofrecer una interfaz *intuitiva* y fácil de usar, basada en principios de diseño de experiencia de usuario (*UX*).

Seguridad

Para garantizar la seguridad del sistema, se deben cumplir los siguientes requisitos:

- **Autenticación segura:** El sistema debe utilizar *cifrado (TLS)* para proteger los datos de inicio de sesión.
- **Política de contraseñas:** Las contraseñas de los usuarios deben tener al menos *8 caracteres* e incluir una combinación de letras mayúsculas, minúsculas, números y símbolos.

Escalabilidad y Mantenibilidad

Con el objetivo de facilitar la escalabilidad y el mantenimiento del sistema, se deben considerar los siguientes aspectos:

- **Código modular:** El código debe estar estructurado de manera *modular* para permitir futuras ampliaciones y facilitar su mantenimiento.

2.3. Flujo de Gestión de Usuarios y Asignación de Planes

Una de las funcionalidades más importantes y complejas de la aplicación *Fit-Prox* es el flujo de gestión de usuarios y la asignación de planes personalizados de entrenamiento y nutrición. Este flujo asegura que cada cliente reciba planes ajustados a sus objetivos, mientras que entrenadores y nutricionistas gestionan de manera eficiente las asignaciones. A continuación, se describe el funcionamiento en detalle:

Estado inicial del cliente tras el registro

Cuando un usuario completa el proceso de registro, su acceso a la aplicación queda limitado a dos pestañas:

- **Perfil:** Donde puede visualizar la información básica proporcionada durante el registro.
- **Pantalla de carga:** Una vista informativa que le notifica que sus datos están siendo revisados por entrenadores y nutricionistas para asignarle planes personalizados lo antes posible.

Mientras no tenga asignados ambos planes (entrenamiento y nutrición), el cliente solo puede acceder a estas dos pestañas. Esto garantiza que no haya acceso a funcionalidades incompletas mientras se gestionan las asignaciones.

Condiciones para el acceso completo

Una vez que el cliente tiene asignados ambos planes, se le desbloquean las siguientes vistas:

- **Entrenamientos:** Plan semanal de entrenamientos personalizados.
- **Macronutrientes:** Recomendaciones nutricionales basadas en sus objetivos.
- **Progreso:** Seguimiento detallado de su evolución física.
- **Perfil:** Vista de sus datos personales y metas.

Esto asegura una transición fluida desde el estado inicial a una experiencia completa una vez que ambos planes han sido configurados.

Gestión de usuarios pendientes

Entrenadores y nutricionistas gestionan a los clientes pendientes desde una sección específica de la aplicación:

- **Entrenadores:** Los entrenadores acceden a una lista de usuarios pendientes de asignación de un plan de entrenamiento. Al seleccionar un usuario, pueden visualizar su información, crear un nuevo plan o asignar uno existente. Una vez que se asigna el plan, el cliente desaparece de la lista de pendientes del entrenador.
- **Nutricionistas:** Los nutricionistas también disponen de una lista de usuarios pendientes de asignación de un plan nutricional. Aunque el cliente haya recibido su plan de entrenamiento, seguirá apareciendo en esta lista hasta que se le asigne un plan nutricional.

Este diseño permite una gestión clara y organizada de los usuarios, asegurando que cada profesional complete su parte del proceso.

Cambio dinámico de vistas

La transición entre vistas para el cliente ocurre de manera automática según el estado de sus planes. Inicialmente, solo ve las pestañas de "Perfil", "Pantalla de carga". Tras la asignación de ambos planes, la aplicación desbloquea las vistas de "Entrenamientos", "Macronutrientes" y "Progreso".

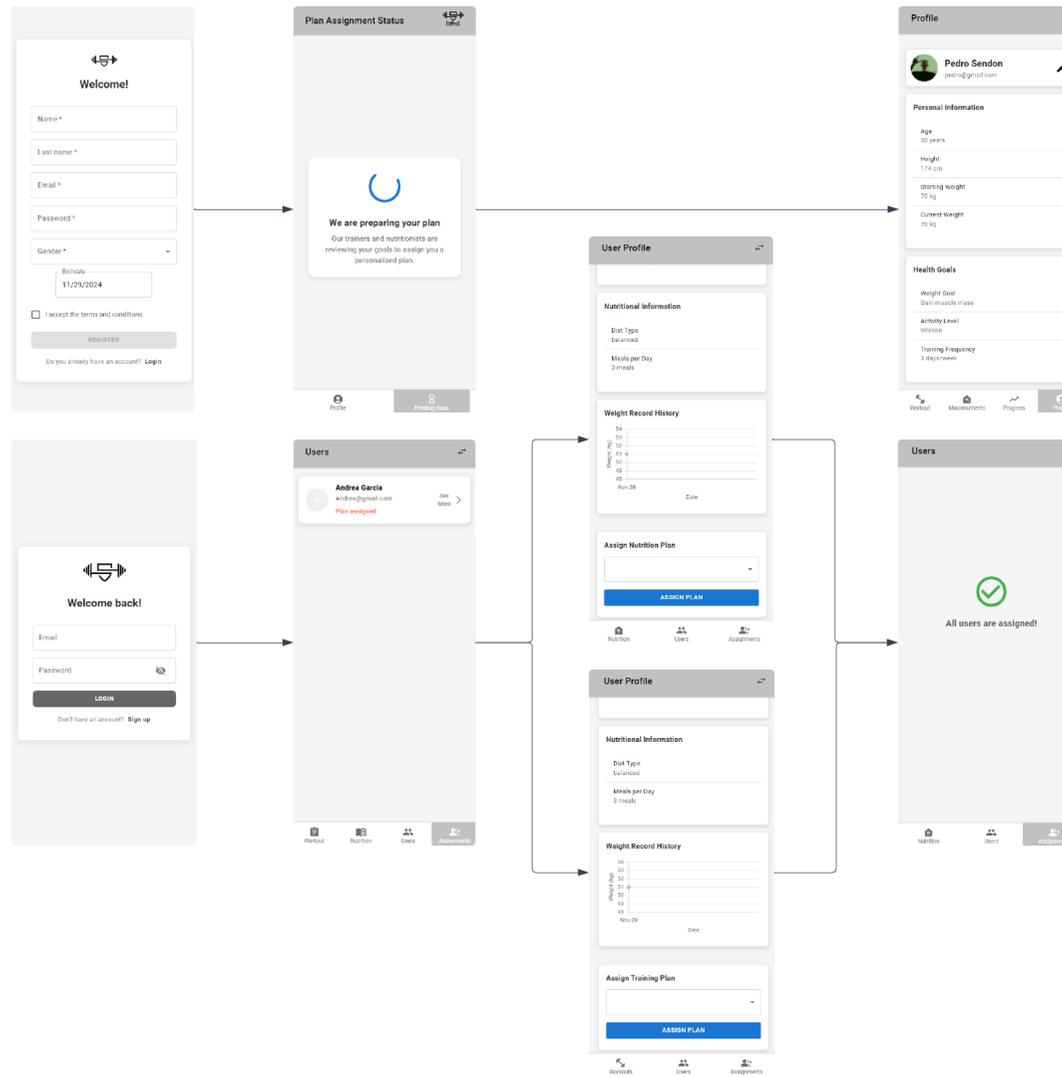


Figura 2: Diagrama del flujo de vistas según el estado del cliente.

El diagrama en la Figura 2 muestra las diferentes vistas y cómo cambian en función del punto en el que se encuentre el usuario dentro del flujo de asignación.

2.4. Diseño de la Arquitectura del Sistema

Diagrama de Navegación

El diagrama de navegación describe la estructura principal de *FitProx* y cómo los diferentes usuarios (cliente y administrador) interactúan con las pantallas y funcionalidades de la plataforma [11, 12].

- **Inicio y Registro:** Los usuarios comienzan en la pantalla de inicio, donde pueden *iniciar sesión* o registrarse completando un formulario con sus datos personales [13].
- **Clientes:** Tras iniciar sesión, acceden a su **pantalla de entrenamientos**, con opciones para navegar a:
 - **Detalle de entrenamiento:** Visualización de ejercicios diarios.
 - **Detalle de ejercicio:** Instrucciones detalladas para cada ejercicio, accesibles también de forma independiente [14].
 - **Pantalla de macronutrientes:** Consulta del plan nutricional.
 - **Pantalla de progreso:** Seguimiento del desempeño.
 - **Perfil de usuario:** Revisión y edición de datos personales.
- **Administradores:** Tienen acceso a funciones avanzadas como:
 - **Gestión de entrenamientos:** Añadir y modificar entrenamientos.
 - **Gestión de nutrición:** Crear o ajustar planes nutricionales [15].
 - **Gestión de usuarios:** Añadir y modificar usuarios.
 - **Estadísticas:** Visualización del uso de la plataforma y progreso de usuarios.
- **Nutricionistas:** Pueden gestionar planes de nutrición, asignándolos o ajustándolos para usuarios, pero no pueden añadir, eliminar ni modificar usuarios [14].
- **Entrenadores:** Acceden a la gestión de entrenamientos para crear, asignar y modificar planes, con las mismas limitaciones que los nutricionistas en la gestión de usuarios.

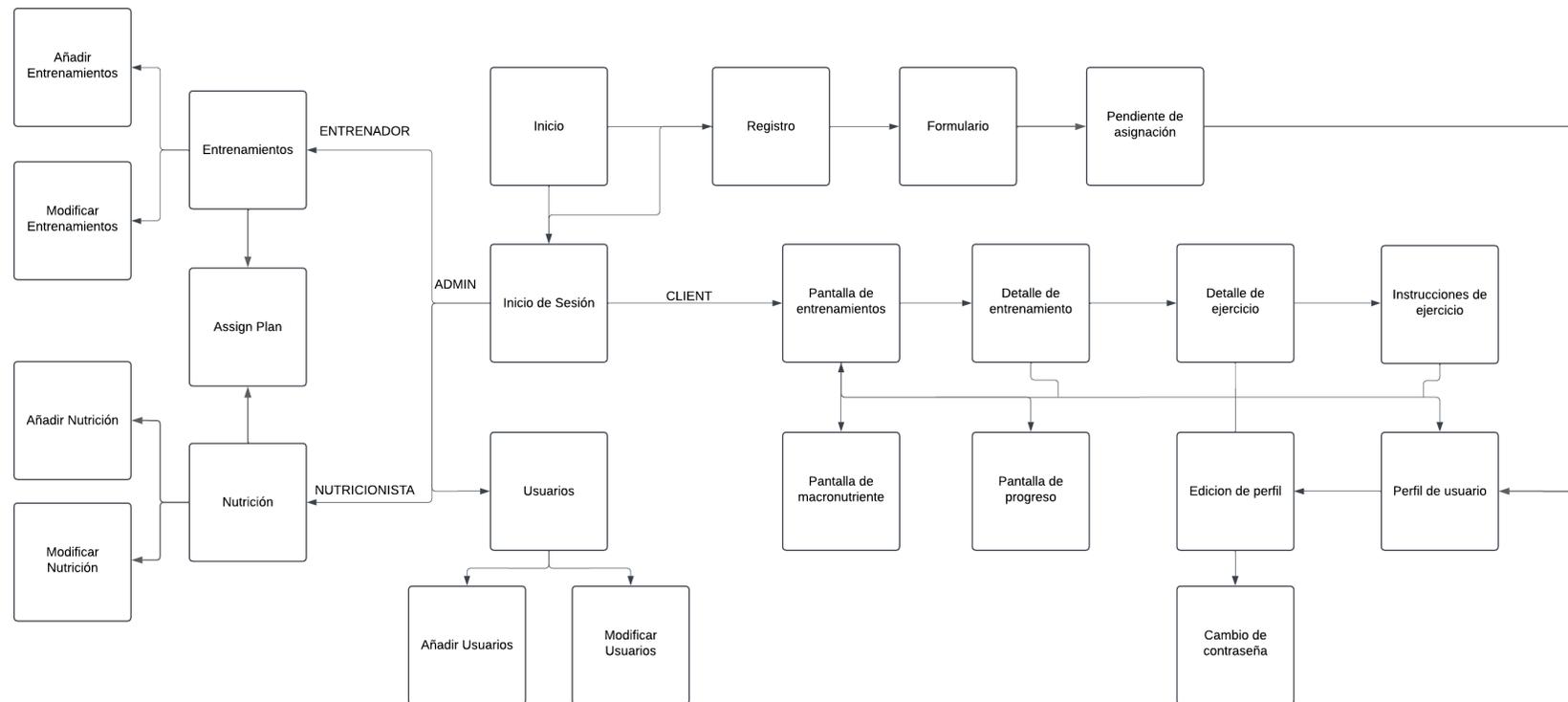


Figura 3: Diagrama de Navegación de la Aplicación FitProx

Modelo Entidad-Relación (MER)

El *Modelo Entidad-Relación* (MER) es esencial para estructurar correctamente la base de datos de *FitProx*. Este modelo identifica las principales entidades del sistema y sus relaciones, asegurando un almacenamiento eficiente y coherente de los datos [16]. Permite gestionar las interacciones entre usuarios, entrenamientos, planes nutricionales y el seguimiento del progreso, proporcionando una base sólida para las funcionalidades clave de la aplicación.

El diagrama MER describe cómo se relacionan entidades como usuarios, entrenamientos, ejercicios, grupos musculares y recomendaciones nutricionales. También muestra las relaciones clave entre usuarios, sus planes de entrenamiento, recomendaciones de macronutrientes y seguimiento de progreso.

El MER garantiza la integridad referencial y facilita la **escalabilidad** futura, permitiendo añadir nuevas funcionalidades, como la gestión de dietas o entrenamientos personalizados, sin afectar la estructura base [17].

Relaciones en el MER El modelo incluye:

- **Relación uno a uno (1:1):** Entre entidades como *User* y *UserDetails*, *Diet-Preferences* o *MedicalConditions*, donde un usuario tiene detalles únicos relacionados con su perfil [16].
- **Relación uno a muchos (1:N):** Un usuario puede tener varios registros de progreso (*ProgressTracking*) y un plan de comidas (*MealPlan*) puede asignarse a múltiples usuarios.
- **Relación muchos a muchos (N:M):** Entre *Workout* y *Exercise*, gestionada por *WorkoutExercise*, lo que permite asignar múltiples ejercicios a entrenamientos y viceversa [17]. De manera similar, *UserWorkout* relaciona usuarios con múltiples entrenamientos.

Claves foráneas y estructura de datos Cada entidad incluye **claves foráneas** para garantizar integridad referencial y conectar datos entre tablas:

- En *ExerciseLog*, las claves *user_id*, *exercise_id* y *workout_id* vinculan usuarios con ejercicios y entrenamientos realizados.
- En *UserNutritionPlan*, las claves *user_id* y *plan_id* conectan usuarios con sus planes nutricionales asignados.

Escalabilidad del modelo El MER está diseñado para ser **escalable**, permitiendo integrar nuevas funcionalidades o entidades, como entrenamientos especializados o dietas avanzadas, sin alterar la estructura básica del sistema [17].

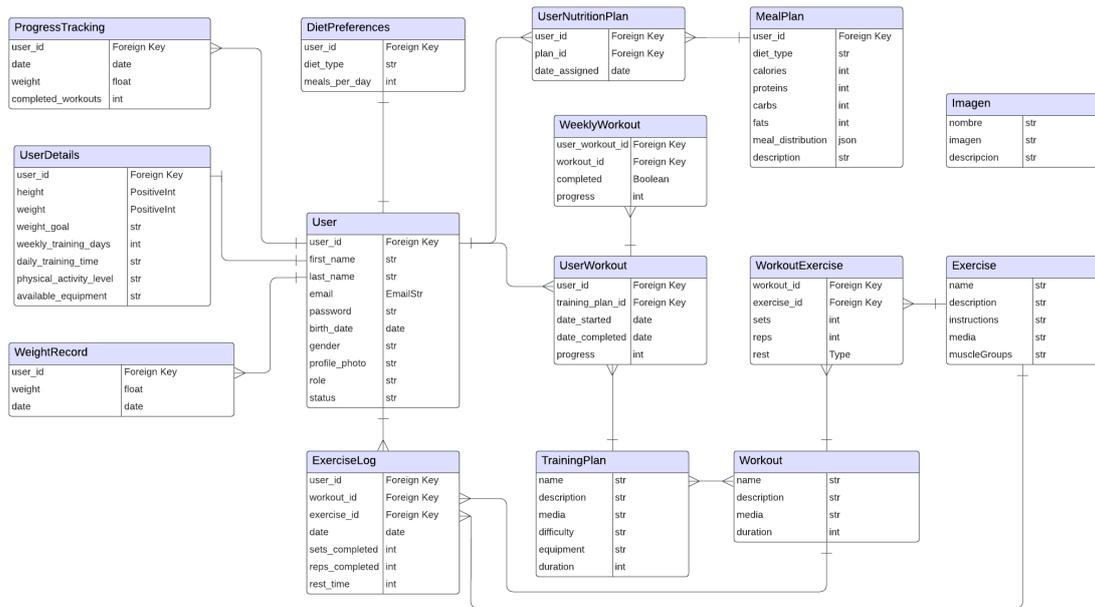


Figura 4: Diagrama Entidad-Relación (MER) de la aplicación FitProx

2.5. Evolución de los Diseños de las Vistas

Durante las etapas iniciales del proyecto, se crearon diseños preliminares de las vistas principales para orientar el desarrollo y estructurar las funcionalidades clave. Estos diseños sirvieron como guía para implementar las características esenciales y establecer la base visual de la aplicación.

A medida que avanzaba el proyecto, surgieron nuevas necesidades y funcionalidades adicionales que no se contemplaron inicialmente. Algunas vistas experimentaron cambios significativos, mientras que otras se desarrollaron sobre la marcha para mejorar la experiencia del usuario y la adaptabilidad del sistema. Este proceso refleja la naturaleza iterativa del desarrollo de software, donde los diseños evolucionan en paralelo con el refinamiento del proyecto.

En el Anexo 9, se incluyen los diseños originales, mostrando cómo la planificación inicial se ajustó para satisfacer las demandas emergentes y alcanzar los objetivos finales.

3. TECNOLOGÍAS

3.1. Tecnologías de Desarrollo Web

En esta parte, se ofrece una explicación de las tecnologías que son fundamentales para el *desarrollo web*, incluyendo *frameworks* y lenguajes de programación. Dentro del diseño web, un *framework* es un conjunto estructurado de herramientas y recursos que proporciona una base sólida para la creación de aplicaciones web. Actúa como una estructura base que facilita y optimiza el proceso de desarrollo al proporcionar a los desarrolladores un punto de partida organizado y con funciones pre construidas.

El uso de *frameworks* aporta una serie de beneficios que mejoran el proceso de creación, dando lugar a aplicaciones y sitios web más robustos, adaptables y eficientes. Entre las ventajas de utilizar *frameworks* se incluyen:

- *Reducción del tiempo y esfuerzo*: Los *frameworks* ofrecen componentes y herramientas reutilizables, lo que elimina la necesidad de crear código desde cero para funciones comunes como la autenticación de usuarios, el manejo de formularios o la conexión a bases de datos.
- *Aumento de la eficiencia y productividad*: Al proporcionar elementos reutilizables y marcos de trabajo probados, se promueve un desarrollo más ágil y productivo.
- *Escalabilidad y facilidad de mantenimiento*: Están diseñados para desarrollar aplicaciones web que puedan crecer y ser fácilmente mantenidas. Conforme el proyecto se expande y las demandas aumentan, el *framework* puede ajustarse sin complicaciones, facilitando la incorporación de nuevas características.
- *Código claro y estandarizado*: Promueven un estilo de programación consistente y legible, lo cual facilita la colaboración entre desarrolladores y simplifica el entendimiento y mantenimiento del código a largo plazo.
- *Seguridad y confiabilidad*: Muchos *frameworks* incorporan medidas de seguridad predeterminadas para proteger las aplicaciones web de amenazas comunes como la inyección de código.

Desarrollo del Frontend

Para el desarrollo del *frontend* de la aplicación se han utilizado tecnologías ampliamente reconocidas, como *HTML*, *CSS* y *JavaScript*, junto con herramientas específicas como *React* e *Ionic*. A continuación, se destacan brevemente las características de las tecnologías base y se profundiza en las herramientas seleccionadas.

HTML, CSS y JavaScript Estas tecnologías forman la base del desarrollo web:

- *HTML*: Estructura el contenido de la aplicación, asegurando compatibilidad y accesibilidad [18].

- *CSS*: Proporciona un diseño visual coherente y adaptable a diferentes dispositivos [19].
- *JavaScript*: Permite añadir interactividad y dinamismo a la interfaz [20].

Si bien son fundamentales, su uso es estándar en el desarrollo de aplicaciones web y no constituye una decisión específica del proyecto.

React *React* es una biblioteca de *JavaScript* elegida para construir interfaces de usuario debido a sus características avanzadas [21]:

- *Componentes reutilizables*: Facilitan el desarrollo modular y escalable [21].
- *Virtual DOM*: Mejora el rendimiento al actualizar únicamente las partes necesarias [22].
- *Unidirectional Data Flow*: Simplifica la gestión del estado de la aplicación [23].
- *Ecosistema robusto*: Proporciona numerosas herramientas y recursos [21].

Ionic *Ionic* fue clave para desarrollar una experiencia consistente en diferentes plataformas, destacando por:

- *Desarrollo multiplataforma*: Permite crear aplicaciones para iOS, Android y web desde una base de código única [24].
- *Componentes optimizados para móviles*: Garantizan una experiencia de usuario nativa [25].
- *Integración con React*: Mejora la eficiencia en la construcción de vistas dinámicas [26].
- *Acceso a funcionalidades nativas*: Con *Capacitor*, ofrece acceso a cámaras, GPS y notificaciones push [27].
- *Soporte para PWA*: Facilita el desarrollo de aplicaciones web progresivas con experiencia de *app* nativa [28].

Conclusión La combinación de tecnologías estándar como *HTML*, *CSS* y *JavaScript* con herramientas avanzadas como *React* e *Ionic* permitió crear un *frontend* eficiente, escalable y adaptable a múltiples plataformas, cumpliendo con los requisitos de la aplicación *FitProx*.

Desarrollo del Back-End

El desarrollo del *back-end* de *FitProx* ha sido llevado a cabo utilizando *Python 3* y el *framework Django*, acompañado de *Django REST Framework (DRF)* para la creación de una *API RESTful* robusta. Estas herramientas fueron seleccionadas por su capacidad para facilitar un desarrollo ágil, escalable y seguro.

Python 3 *Python* es un lenguaje de programación de alto nivel reconocido por su simplicidad, legibilidad y versatilidad. Python 3 fue elegido por sus características avanzadas y mejoras respecto a versiones anteriores, siendo ideal para aplicaciones web modernas [29].

- *Sintaxis clara y legible*: Facilita el desarrollo y la depuración, permitiendo centrarse en la lógica del negocio [30].
- *Amplia biblioteca estándar*: Incluye módulos para tareas comunes como manejo de fechas, encriptación y conexión a bases de datos [31].
- *Comunidad activa*: La extensa comunidad de Python ofrece herramientas, bibliotecas y documentación que enriquecen su ecosistema [32].

Django *Django* es un *framework* web basado en Python, conocido por su rapidez, seguridad y escalabilidad, características clave para el desarrollo de *FitProx* [33]. Fue elegido como núcleo del *back-end* por sus ventajas:

- *Desarrollo rápido*: Herramientas preconfiguradas como autenticación, *ORM* y validación de formularios reducen configuraciones adicionales [34].
- *Arquitectura "baterías incluidas"*: Permite implementar funciones críticas, como manejo de usuarios y permisos, con rapidez [35].
- *Autenticación robusta*: Sistema integrado con gestión de usuarios, manejo de sesiones y contraseñas encriptadas [36].
- *Seguridad avanzada*: Protección contra amenazas como inyección *SQL*, *XSS* y *CSRF* [37].

Django REST Framework (DRF) Para la *API RESTful* de *FitProx*, se utilizó *Django REST Framework* (DRF), que extiende Django para crear interfaces web modernas [38]. DRF fue esencial para construir endpoints de usuarios, planes de entrenamiento y estadísticas, ofreciendo:

- *Serialización de datos*: Conversión sencilla entre formatos (JSON) y modelos, mejorando la comunicación *frontend-backend*.
- *Validación y permisos*: Sistema robusto para validar datos y gestionar accesos según permisos [39].
- *Manejo de solicitudes HTTP*: Simplifica la gestión de métodos *GET*, *POST*, *PUT* y *DELETE*.

Conclusión El uso de *Python 3*, *Django* y *Django REST Framework*, junto con la integración de *PostgreSQL* como sistema de gestión de bases de datos, permitió desarrollar un *backend* seguro, eficiente y escalable para *FitProx*. Estas herramientas no solo cubrieron las necesidades técnicas del proyecto, sino que también aseguraron una base sólida para futuras expansiones y un manejo confiable de los datos de los usuarios.

Base de Datos: PostgreSQL

Para el almacenamiento de datos de la aplicación, inicialmente se consideró *Firebase* por sus servicios de sincronización en tiempo real y arquitectura sin servidor [40]. Sin embargo, se optó por *PostgreSQL*, un RDBMS robusto y apto para consultas complejas [41].

Evaluación de Firebase *Firebase* presentaba ventajas como sincronización en tiempo real y desarrollo ágil [42], pero también limitaciones:

- Dificultad para consultas relacionales complejas, esenciales en *FitProx* [43].
- Falta de soporte para transacciones avanzadas, cruciales para garantizar integridad de datos [44].
- Dependencia de Google, limitada en portabilidad y flexibilidad a largo plazo [45].

PostgreSQL: Elección Final *PostgreSQL*, un sistema relacional de código abierto, fue elegido por su equilibrio entre rendimiento y características avanzadas:

- **Modelo relacional:** Gestión de datos estructurados con integridad referencial [46].
- **Consultas SQL avanzadas:** Soporte para uniones, subconsultas y análisis complejos [47].
- **Transacciones ACID:** Garantizan fiabilidad mediante restricciones como llaves foráneas y valores únicos [48].
- **Extensibilidad:** Permite tipos personalizados y funciones adaptadas a *FitProx* [49].
- **Escalabilidad y rendimiento:** Maneja grandes volúmenes de datos de forma eficiente [50].
- **Soporte JSON:** Flexibilidad para datos no estructurados junto a datos relacionales [51].
- **Compatibilidad externa:** Fácil integración con herramientas de análisis y visualización [52].

Gestión de Imágenes con Google Cloud Storage Además de la gestión de datos estructurados con *PostgreSQL*, *FitProx* utiliza *Google Cloud Storage* para almacenar y gestionar archivos multimedia como imágenes de perfil. Este servicio en la nube permite un almacenamiento seguro y escalable de archivos no estructurados [53].

Ventajas de Google Cloud Storage

- **Escalabilidad:** Manejo de grandes volúmenes de datos con alta disponibilidad [54].
- **Seguridad:** URLs firmadas para acceso controlado a los archivos [55].
- **Durabilidad:** Infraestructura redundante con 99.99 % de disponibilidad [56].

Uso e Implementación *Google Cloud Storage* gestiona imágenes de usuarios y otros recursos multimedia, optimizando la base de datos relacional al almacenar solo enlaces en *PostgreSQL* [57]. La integración incluyó:

1. **Subida de archivos:** Almacenados con identificadores únicos.
2. **URLs firmadas:** Generadas para acceso temporal.
3. **Eliminación automática:** Archivos removidos al ser actualizados o eliminados.

Conclusión La combinación de *PostgreSQL* y *Google Cloud Storage* asegura una gestión eficiente de datos estructurados y multimedia, adaptándose a las necesidades actuales y futuras de *FitProx*.

3.2. Despliegue de la Aplicación

El despliegue de la aplicación ha sido un componente crucial para garantizar su correcto funcionamiento en entornos de desarrollo y producción. Se han empleado dos estrategias complementarias: *Docker Compose* para el desarrollo local y *Render* para el despliegue en producción.

Docker Compose en Desarrollo

Durante la etapa de desarrollo, se utilizó *Docker Compose* para levantar los servicios principales de la aplicación, como el *backend* desarrollado en *Django* y la base de datos *PostgreSQL*. Esto permitió configurar un entorno reproducible y aislado, asegurando la correcta integración de los componentes y facilitando la ejecución de pruebas locales.

Render en Producción

El despliegue en producción se llevó a cabo utilizando *Render*, un servicio de *cloud hosting* que facilita la implementación de aplicaciones web y servicios backend. Los tres componentes principales del proyecto —*frontend*, *backend* y base de datos— se alojaron de manera separada, asegurando una arquitectura modular y escalable. Este enfoque permitió gestionar las configuraciones específicas de producción y simplificó la puesta en marcha del sistema en un entorno controlado.

Ventajas del Enfoque Híbrido

El uso de *Docker Compose* en desarrollo y *Render* en producción proporcionó un flujo de trabajo eficiente:

- **Reproducibilidad:** El uso de contenedores en desarrollo aseguró consistencia entre entornos.
- **Escalabilidad:** Render ofrece una infraestructura optimizada para gestionar el crecimiento de la aplicación en producción.
- **Simplicidad:** La configuración simplificada de servicios en ambos entornos permitió un despliegue ágil y sin contratiempos.

Este enfoque ha permitido mantener un desarrollo ágil y una transición fluida a producción, garantizando un despliegue robusto y eficiente.

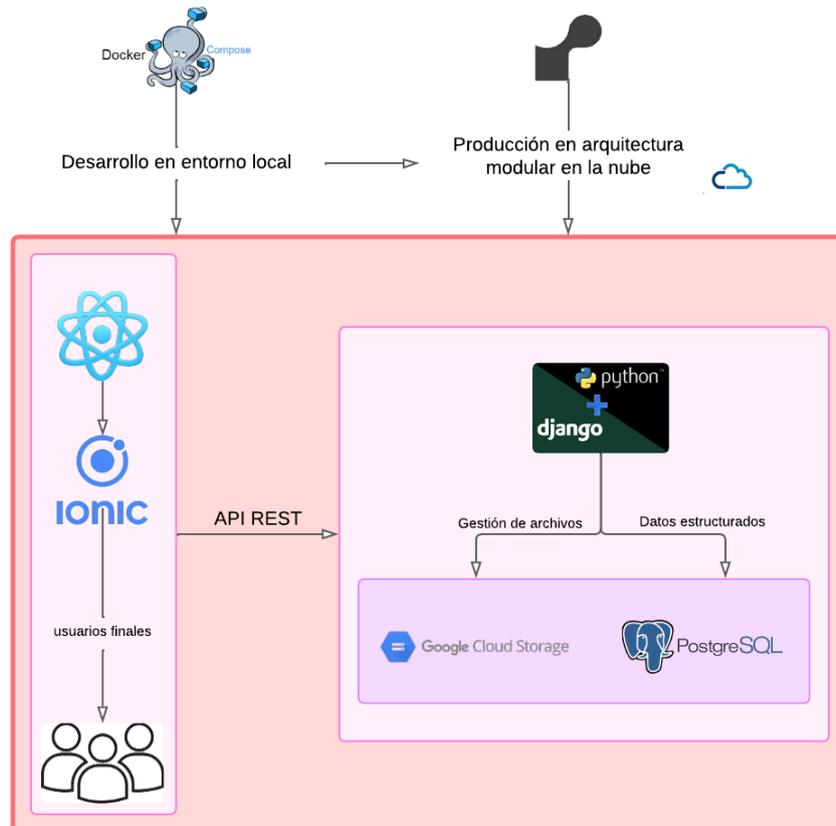


Figura 5: Diagrama de la arquitectura del proyecto

4. IMPLEMENTACIÓN

4.1. Gestión de Funcionalidades

Registro de Usuarios

La funcionalidad de *Registro de Usuarios* se implementa en dos etapas principales para garantizar que los requisitos se cumplan adecuadamente:

- **Creación de cuenta básica:** El usuario ingresa información básica (correo, nombre, contraseña, fecha de nacimiento y sexo) en un formulario interactivo. Los campos son validados individualmente:
 - El correo debe ser válido y único.
 - La contraseña debe tener al menos 8 caracteres, una mayúscula, una minúscula, un número y un símbolo.

Si algún campo es inválido, se muestra un mensaje de error. Tras la validación, el usuario puede completar el registro.

- **Formulario adicional:** Después del registro, el usuario completa un formulario con información detallada sobre sus objetivos y hábitos de entrenamiento:
 - Objetivo físico principal, nivel de actividad y frecuencia de entrenamiento.
 - Métricas personales como peso, altura y otras características.

Este formulario también utiliza validaciones para asegurar la integridad de los datos.

Los datos se almacenan de forma segura y se utilizan para generar planes personalizados.

Inicio de Sesión

La funcionalidad de *inicio de sesión* asegura un proceso seguro mediante los siguientes pasos:

- **Validación de credenciales:** El usuario ingresa su correo electrónico y contraseña, verificando su formato y comparando la contraseña con la almacenada de forma segura usando un algoritmo de hash. Si no coinciden, se muestra un mensaje de error.
- **Autenticación:** Tras validar las credenciales, se genera un token de sesión único con una duración limitada para autenticar las solicitudes.
- **Redirección al panel principal:** Después de la autenticación, el usuario es redirigido al panel principal donde puede acceder a sus funcionalidades personalizadas.

Visualización de Entrenamientos Semanales

La funcionalidad de *Visualización de Entrenamientos Semanales* garantiza que los planes personalizados asignados por los entrenadores sean accesibles y fáciles de seguir:

- **Asignación manual:** Los entrenadores crean planes semanales adaptados a cada usuario, que se almacenan en la base de datos y quedan disponibles para su consulta.
- **Visualización del plan:** Cada día se representa como una tarjeta interactiva que muestra un resumen del entrenamiento. Al seleccionar un día, el usuario accede a detalles como lista de ejercicios, series, repeticiones y tiempos de descanso.
- **Acceso en tiempo real:** Los usuarios pueden acceder a su plan desde cualquier dispositivo, y las actualizaciones realizadas por los entrenadores se reflejan inmediatamente.

Lista de Ejercicios por Entrenamiento

La funcionalidad de *Lista de Ejercicios por Entrenamiento* proporciona a los usuarios una experiencia clara y dinámica, destacando las siguientes características:

- **Visualización de ejercicios:** La lista se presenta como tarjetas o filas interactivas con el nombre del ejercicio y detalles de series y repeticiones, diseñada para dispositivos móviles y escritorio.
- **Interactividad:** Cada ejercicio incluye un *checkbox* para marcarlo como completado, actualizando automáticamente el porcentaje de progreso del día.
- **Finalización del entrenamiento:** Tras completar todos los ejercicios, un botón permite marcar el entrenamiento como finalizado, reflejándose en el historial de progreso.
- **Seguimiento del progreso:** El sistema calcula dinámicamente el porcentaje de progreso del día, representado con una barra o indicador gráfico.

Detalle de Ejercicios

La funcionalidad de *Detalle de Ejercicios* ofrece a los usuarios una comprensión clara y accesible de cada ejercicio mediante las siguientes características:

- **Vista detallada:** Al seleccionar un ejercicio, el usuario accede a una vista con:
 - *Título:* Nombre destacado en la parte superior.
 - *Descripción técnica:* Breve explicación del ejercicio.
 - Imágenes o gráficos accesibles mediante un carrusel interactivo.

- **Información técnica:** Detalles estructurados como grupos musculares trabajados, series, repeticiones y tiempos de descanso.
- **Explicaciones adicionales:** Sección con beneficios del ejercicio, errores comunes y consejos para mejorar la técnica, presentada en un formato visual claro.
- **Interacción y usabilidad:** Interfaz responsiva optimizada para móviles y escritorio, organizada para facilitar una consulta rápida y sin distracciones.

Macronutrientes y Nutrición

La funcionalidad de *Macronutrientes y Nutrición* proporciona a los usuarios una experiencia interactiva y personalizada mediante las siguientes características:

- **Tipo de dieta:** La interfaz muestra el tipo de dieta recomendado, destacado con una breve descripción basada en la información proporcionada por el nutricionista.
- **Macronutrientes:** Las cantidades diarias de carbohidratos, proteínas y grasas se presentan en:
 - Tablas que detallan gramos y calorías por macronutriente.
 - Un gráfico circular con la distribución porcentual de macronutrientes.
- **Distribución de comidas:** Tabla que sugiere cómo repartir los macronutrientes entre desayuno, almuerzo, cena y snacks, con porcentajes recomendados para cada comida.
- **Interactividad y usabilidad:** Información accesible desde la pestaña *Macronutrientes*, con una interfaz responsiva optimizada para móviles y escritorio.

Visualización del Perfil

La funcionalidad de *Visualización del Perfil* ofrece a los usuarios una experiencia intuitiva y organizada para consultar y gestionar su información:

- **Datos personales:** Vista organizada con información destacada como nombre, correo electrónico, edad, altura, peso inicial y peso actual. Las categorías están claramente diferenciadas con etiquetas legibles.
- **Objetivos:** Los objetivos de entrenamiento y nutrición se presentan en bloques separados, resaltando el objetivo principal, nivel de actividad física y frecuencia de entrenamientos.
- **Edición de perfil:** Un botón de edición permite actualizar información personal y objetivos, redirigiendo a un formulario con validaciones para garantizar la consistencia.

- **Cierre de sesión:** Botón ubicado en la parte inferior que elimina el token de autenticación y redirige a la pantalla de inicio de sesión.
- **Usabilidad y diseño:** Interfaz responsiva optimizada para móviles y escritorio, con datos organizados visualmente para facilitar su consulta rápida.

Edición de Información del Perfil

La funcionalidad de *Edición de Información del Perfil* permite a los usuarios actualizar su información de manera fluida y segura mediante las siguientes características:

- **Formulario de edición:** Formulario interactivo que muestra los datos actuales como valores predeterminados, permitiendo actualizar:
 - Nombre y apellido: Campos de texto.
 - Peso actual: Campo numérico.
 - Objetivo de peso, nivel de actividad física y frecuencia de entrenamientos: Menús desplegables.
- **Cambio de contraseña:** Botón dedicado que dirige a una vista específica para cambiar la contraseña, validando requisitos de seguridad como longitud y caracteres especiales.
- **Gestión de la foto de perfil:** Permite subir una nueva foto desde el dispositivo, reemplazando la imagen anterior tras su procesamiento.
- **Validación y guardado:** Validaciones aseguran que los datos sean válidos antes de guardarse en la base de datos, redirigiendo al usuario al perfil actualizado tras el éxito.
- **Usabilidad y diseño:** Interfaz responsiva optimizada para móviles y escritorio, con campos organizados intuitivamente y mensajes claros en caso de error.

Cambio de Contraseña: Implementación

La funcionalidad de *Cambio de Contraseña* garantiza un proceso seguro y fluido para los usuarios mediante las siguientes características:

- **Validación de la contraseña actual:** El usuario debe ingresar su contraseña actual, que es verificada contra la base de datos utilizando un algoritmo de hash. Si no coincide, se muestra un mensaje de error.
- **Ingreso de la nueva contraseña:** El formulario incluye validaciones para asegurar que la nueva contraseña cumpla con los requisitos de seguridad (mínimo 8 caracteres, al menos una mayúscula, una minúscula, un número y un símbolo). Si no cumple, se indica al usuario.

- **Confirmación de la nueva contraseña:** El sistema compara la nueva contraseña con su confirmación para garantizar que coincidan. Si no lo hacen, se muestra un mensaje de error.
- **Actualización segura:** Una vez validada, la contraseña se actualiza en la base de datos utilizando un hash seguro (e.g., bcrypt). El usuario es redirigido a una vista de confirmación tras el éxito.
- **Usabilidad y diseño:** Los campos están protegidos para ocultar los caracteres ingresados, y la interfaz ofrece mensajes claros de error y confirmación.

Historial de Progreso

La funcionalidad de *Historial de Progreso* permite a los usuarios realizar un seguimiento detallado de su evolución física y actividad de entrenamiento, destacando las siguientes características:

- **Calendario y próximo entrenamiento:**
 - Calendario interactivo con colores para diferenciar días con entrenamientos completados y programados.
 - Widget destacado que muestra nombre de la rutina, progreso del plan y cuenta regresiva hasta la próxima sesión.
- **Historial y progreso:**
 - Registro cronológico de entrenamientos completados.
 - Barra de progreso que muestra el porcentaje del plan completado.
 - Opciones de filtro por fechas o tipo de rutina.
- **Seguimiento del peso:**
 - Formulario para añadir peso manualmente.
 - Gráfica interactiva que muestra la evolución del peso y resalta hitos como peso inicial y objetivo.
- **Mensajes de motivación:**
 - Mensajes basados en entrenamientos completados y progresión del peso hacia el objetivo, visibles al final de cada sesión y en el historial.
- **Usabilidad y diseño:**
 - Interfaz responsiva optimizada para móviles y escritorio, con gráficos y barras de progreso fáciles de interpretar.

Gestión de Planes de Entrenamiento, Entrenamientos y Ejercicios

La funcionalidad de *Gestión de Planes de Entrenamiento, Entrenamientos y Ejercicios* permite al administrador gestionar eficazmente los recursos de entrenamiento, asegurando una experiencia personalizada para los usuarios. Sus principales características son:

- **Planes de entrenamiento:**
 - Formulario interactivo para crear planes definiendo nombre, objetivos y entrenamientos asociados.
 - Función de modificación para reorganizar entrenamientos y ajustar objetivos.
 - Interfaz para eliminar planes con confirmación previa.
- **Entrenamientos:**
 - Interfaz para crear entrenamientos añadiendo ejercicios, configurando series, repeticiones y parámetros.
 - Función de modificación para ajustar ejercicios y parámetros.
 - Opción para eliminar entrenamientos con confirmación previa.
- **Ejercicios:**
 - Formulario para crear ejercicios con detalles como nombre, descripción, parámetros y medios visuales.
 - Función de modificación para actualizar descripciones y parámetros.
 - Opción para eliminar ejercicios, manteniendo la base de datos actualizada.
- **Usabilidad y diseño:**
 - Interfaz intuitiva para navegar entre planes, entrenamientos y ejercicios.
 - Validaciones y confirmaciones para minimizar errores en todas las acciones.

Gestión de Planes de Macronutrientes

La funcionalidad de *Gestión de Planes de Macronutrientes* permite al administrador gestionar eficazmente las recomendaciones nutricionales de la aplicación mediante las siguientes características:

- **Añadir planes:** Formulario interactivo para especificar calorías, porcentajes o valores exactos de proteínas, carbohidratos y grasas, asociándolos a un tipo de dieta. Los datos son validados antes de guardarse.
- **Modificar planes:** Interfaz que permite ajustar calorías, macronutrientes, descripción y asociación con tipos de dieta. Las modificaciones se reflejan automáticamente tras superar validaciones.

- **Eliminar planes:** Lista de planes con opción para eliminarlos, requiriendo confirmación previa. Los planes eliminados se marcan como inactivos para mantener trazabilidad.
- **Usabilidad y diseño:** Interfaz intuitiva y organizada con mensajes claros de error y confirmación para guiar al administrador.

Gestión de Usuarios

La funcionalidad de *Gestión de Usuarios* permite a los administradores manejar de manera eficiente y segura los perfiles de los usuarios mediante las siguientes características:

- **Añadir usuarios:** Formulario exclusivo para administradores, donde se registran datos como nombre, correo, objetivos iniciales y nivel de actividad física. Los datos son validados antes de guardarse.
- **Modificar usuarios:** Permite actualizar datos personales, objetivos de entrenamiento y nutrición, y otros detalles relevantes, reflejándose de inmediato tras la validación.
- **Eliminar usuarios:** Incluye una lista con opciones para eliminar usuarios. Un mensaje de confirmación previene errores, y los usuarios eliminados se marcan como inactivos para mantener trazabilidad.
- **Asignar planes:** Facilita la asignación de planes de entrenamiento o nutrición mediante listas desplegadas predefinidas o personalizadas, alineadas con los objetivos del usuario.
- **Usabilidad y diseño:** Una interfaz organizada e intuitiva, con mensajes de confirmación y retroalimentación visual para guiar al administrador durante las acciones.

Cambio de Idioma

La funcionalidad de *Cambio de Idioma* permite a los usuarios seleccionar su idioma preferido, ofreciendo una experiencia personalizada y accesible. Sus características principales son:

- **Selección de idioma:** Un *menú desplegable* accesible desde un icono en la esquina superior derecha permite elegir entre *inglés*, *español* y *atalán*, representados con banderas. Al seleccionar un idioma, la interfaz se actualiza dinámicamente.
- **Contexto de idioma:** Un *LanguageContext* centralizado administra las traducciones y asegura que todas las vistas accedan a los textos actualizados, utilizando bibliotecas como `react-i18next`.

- **Almacenamiento de idiomas:** Las traducciones están organizadas en archivos *JSON* (*en.json*, *es.json*, *ca.json*), que se cargan dinámicamente según la selección del usuario, optimizando el rendimiento.
- **Usabilidad y diseño:** El cambio de idioma se aplica instantáneamente en toda la interfaz, garantizando una experiencia fluida y funcional en diferentes dispositivos.

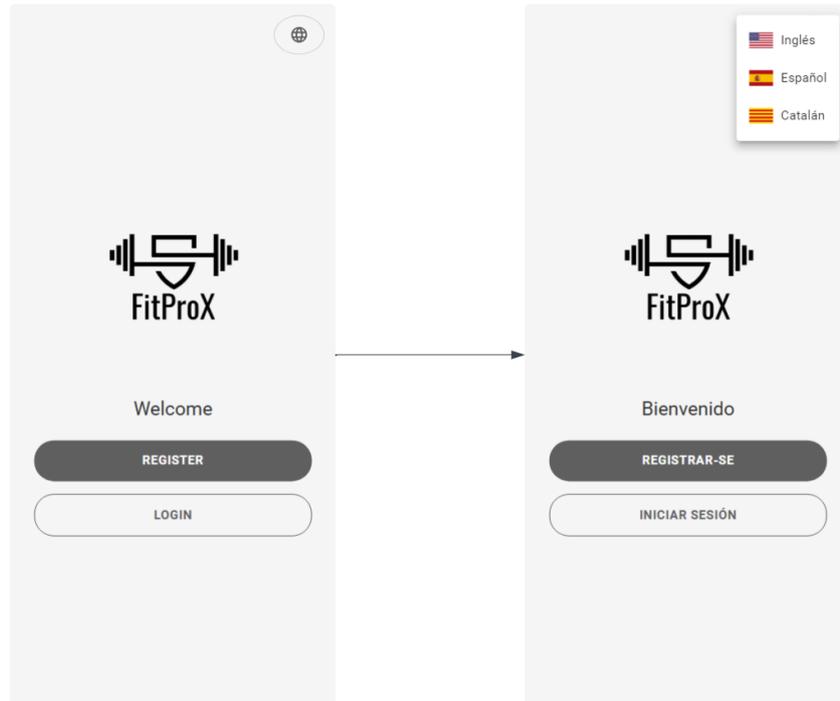


Figura 6: Interfaz del cambio de idioma con banderas representativas.

Asignación de Planes Pendientes a Usuarios

La funcionalidad de *Asignación de Planes Pendientes a Usuarios* optimiza la gestión de usuarios sin planes asignados, permitiendo a administradores, entrenadores y nutricionistas gestionar esta tarea de manera eficiente:

- **Visualización de usuarios pendientes:** Una pantalla dinámica muestra una lista de usuarios sin planes, filtrada según el rol:
 - *Entrenadores:* Usuarios sin plan de entrenamiento.
 - *Nutricionistas:* Usuarios sin plan de nutrición.
 - *Administradores:* Usuarios sin planes categorizados según tipo de plan faltante.

La lista incluye datos básicos como nombre, correo y estado de asignación.

- **Acceso y asignación de planes:** Al seleccionar un usuario, se accede a una vista con su información detallada y un *selector de planes* para asignar uno existente o crear uno nuevo si es necesario.
- **Confirmación de asignación:** El sistema solicita confirmación antes de asignar el plan, actualiza la base de datos y elimina al usuario de la lista de pendientes.
- **Usabilidad y diseño:** La interfaz responsiva facilita la navegación y proporciona mensajes claros durante todo el proceso.

Visualización Detallada de Entidades

La funcionalidad de *Visualización Detallada de Entidades* permite acceder de manera organizada a la información de ejercicios, entrenamientos, planes y usuarios, destacando las siguientes características:

- **Acceso directo desde listas:** Cada lista de entidades incluye botones o enlaces para acceder rápidamente a vistas detalladas, organizadas con filtros y opciones de búsqueda.
- **Detalles completos:** Vistas dedicadas para cada entidad muestran información estructurada:
 - *Ejercicio:* Nombre, descripción, grupo muscular, imágenes, series y repeticiones.
 - *Entrenamiento:* Lista de ejercicios, series, repeticiones y descansos.
 - *Plan de nutrición:* Tipo de dieta, calorías, macronutrientes y recomendaciones.
 - *Plan de entrenamiento:* Objetivos, frecuencia y entrenamientos asociados.
 - *Usuario:* Datos personales, objetivos y estado de planes asignados.
- **Navegación optimizada:** Barra de navegación para volver a listas o acceder a vistas relacionadas. La interfaz es responsiva, optimizada para dispositivos móviles y resalta información clave.

Tipos de Usuarios y Gestión de Roles

La funcionalidad de *Tipos de Usuarios y Gestión de Roles* se implementó para garantizar un control claro y seguro de los permisos de cada tipo de usuario, cumpliendo con los requisitos establecidos. Las principales características son:

- **Definición de roles:**
 - Se implementaron cuatro roles principales: *cliente*, *administrador*, *entrenador* y *nutricionista*.

- Cada rol se definió mediante un sistema de permisos asignados en la base de datos y gestionados a través de un middleware.
- **Gestión de permisos:**
 - El acceso a las funcionalidades está controlado por permisos que se validan en cada solicitud.
 - Los permisos se organizan en niveles:
 - *Cliente*: Solo puede acceder a entrenamientos, planes nutricionales, historial de progreso y personalización de su perfil.
 - *Administrador*: Tiene acceso completo a todas las funcionalidades, incluyendo la gestión de usuarios, estadísticas y planes.
 - *Entrenador*: Puede gestionar entrenamientos y asignar planes de entrenamiento a los clientes.
 - *Nutricionista*: Puede gestionar planes nutricionales y asignarlos a los clientes.
- **Seguridad:**
 - Los permisos se gestionan mediante un sistema centralizado que asegura que los usuarios no puedan acceder a funcionalidades no autorizadas.
 - Cada acción está protegida con validaciones de rol en el backend, garantizando que los datos solo sean accesibles para los usuarios con permisos correspondientes.
- **Interfaz basada en roles:**
 - La interfaz de usuario se adapta dinámicamente según el rol, utilizando un sistema de control que filtra las vistas y componentes disponibles.
 - Por ejemplo:
 - Los clientes solo ven entrenamientos, planes nutricionales y su progreso.
 - Los entrenadores y nutricionistas acceden a vistas específicas para gestionar entrenamientos o planes nutricionales, respectivamente.
 - Los administradores tienen acceso completo a todos los paneles.

5. PRUEBAS

5.1. Pruebas de usuario

En esta sección se describen las pruebas de usuario realizadas con el objetivo de evaluar la funcionalidad, usabilidad y efectividad de la aplicación. Para llevar a cabo estas pruebas, se han seguido dos tipos de enfoques: la exploración libre y las entrevistas guiadas.

Las pruebas exploratorias permitieron a los usuarios interactuar con la aplicación de manera autónoma, sin instrucciones detalladas. Este enfoque permitió observar el flujo natural de uso y detectar posibles dificultades o errores que pudieran surgir durante la interacción espontánea con la aplicación. Por otro lado, en las entrevistas guiadas, los usuarios fueron acompañados a través de una serie de tareas predefinidas. Esto permitió obtener información detallada sobre las percepciones y opiniones de los usuarios respecto a características específicas de la aplicación.

Para garantizar una perspectiva amplia y diversa en los resultados, se contó con la participación de varios perfiles de usuarios. En las pruebas participaron entrenadores, nutricionistas, usuarios habituales de gimnasio, y usuarios sin experiencia previa en entrenamiento o en el uso de aplicaciones de fitness. Esta variedad de perfiles fue fundamental para obtener una retroalimentación rica y comprensiva, ya que cada grupo aportó observaciones y sugerencias desde su experiencia particular.

En el *Anexo 9* se incluyen los guiones de las entrevistas guiadas, que contienen las preguntas y tareas planteadas a los participantes. Estos guiones proporcionan una estructura detallada que asegura la consistencia en la recolección de información durante las entrevistas, permitiendo así comparar las respuestas entre diferentes perfiles de usuario de manera más eficaz.

A continuación, se detallan los errores encontrados, las sugerencias de mejora y los cambios implementados en respuesta a esta retroalimentación.

5.2. Pruebas automatizadas

En esta sección se detalla el proceso de implementación y los resultados de las pruebas automatizadas realizadas para el backend de la aplicación. El objetivo de estas pruebas es asegurar que las funcionalidades clave del sistema operen correctamente, validando tanto la lógica de negocio como la interacción con la base de datos.

Las pruebas automatizadas se llevaron a cabo utilizando el framework *Django REST Framework* en combinación con *APITestCase* para realizar pruebas de los endpoints de la API, evaluando la correcta respuesta de cada operación en el sistema.

Estructura de las pruebas

El directorio de pruebas está organizado en archivos separados, cada uno correspondiente a una entidad o módulo del sistema, tales como `test_exercise.py`, `test_mealplan.py`, `test_trainingplan.py`, `test_user.py` y `test_workout.py`. Es-

ta estructura facilita la organización y permite el mantenimiento independiente de las pruebas para cada funcionalidad.

Cada archivo de pruebas contiene un conjunto de casos de prueba que verifican el correcto funcionamiento de los endpoints y métodos críticos. Los principales tipos de pruebas desarrolladas incluyen:

- **Pruebas de creación:** Se verificó que los endpoints pudieran crear correctamente nuevos registros en el sistema, como entrenamientos, planes de comidas, planes de entrenamiento y usuarios.
- **Pruebas de actualización:** Las pruebas evalúan si los métodos de actualización funcionan como se espera, modificando datos existentes en la base de datos sin generar errores.
- **Pruebas de eliminación:** Estas pruebas aseguran que los registros puedan eliminarse correctamente, sin afectar otras partes del sistema.
- **Pruebas de obtención:** Se realizaron pruebas para confirmar que los datos se obtienen correctamente desde la base de datos, tanto en operaciones que devuelven un solo registro (por ID) como en aquellas que devuelven listas de registros (por categoría o usuario).
- **Pruebas de permisos:** Los casos de prueba validaron que solo los usuarios con los roles adecuados pudieran acceder o modificar ciertos datos, como planes de entrenamiento, de nutrición, y el acceso a información de usuarios y entrenamientos.

Cobertura de las pruebas

Los endpoints de las principales funcionalidades del backend han sido cubiertos por pruebas automatizadas, incluyendo la creación, actualización y eliminación de ejercicios, planes de comidas, planes de entrenamiento, usuarios y entrenamientos. Además, se implementaron pruebas para endpoints específicos de la lógica de negocio, como la asignación de planes a usuarios y el seguimiento de su progreso.

Las pruebas de cada endpoint simulan peticiones HTTP, y se verifica que las respuestas contengan los datos esperados y el código de estado correcto (por ejemplo, 200 OK para operaciones exitosas o 404 Not Found para casos de registros inexistentes).

Ejemplos de pruebas implementadas

A continuación, se presentan ejemplos de pruebas implementadas para ilustrar el tipo de validaciones realizadas:

- **Creación de ejercicio:** Se verificó que el endpoint de creación de ejercicios respondiera con un código 201 (Creado) cuando se proporcionan datos válidos, y que los datos almacenados coincidan con los datos de entrada.

- **Marcado de entrenamiento como completado:** Se desarrolló una prueba para verificar que un usuario pueda marcar un entrenamiento como completado, y que al hacerlo, se actualice el progreso del plan de entrenamiento del usuario. También se probó que, al completar todos los entrenamientos, el sistema restablezca el progreso a cero.
- **Obtención de plan de comidas por ID:** Se evaluó que el sistema devuelva la información completa de un plan de comidas específico, incluyendo macronutrientes, distribución de comidas y calorías.
- **Acceso restringido a roles específicos:** Las pruebas de permisos aseguraron que solo los usuarios con roles de *entrenador* o *administrador* puedan modificar o eliminar planes de entrenamiento, mientras que el acceso a los planes de nutrición se restringió a los roles de *nutricionista* y *administrador*.

5.3. Errores encontrados

Durante las pruebas de usuario se identificaron varios errores en la aplicación, los cuales afectaban a diferentes funcionalidades y aspectos de la interfaz. A continuación, se detallan los problemas encontrados:

- **Error al añadir plan de nutrición (Error 405 Not Allowed):** La funcionalidad de añadir un nuevo plan de nutrición presenta un error de método no permitido (405 Not Allowed), lo que impide que los nutricionistas y administradores puedan crear planes de alimentación directamente desde la aplicación.
- **Fallo en la funcionalidad de modificación de usuarios:** Al intentar modificar la información de un usuario, la aplicación no guarda correctamente los cambios, lo que causa que los datos actualizados no se reflejen en el perfil del usuario.
- **Texto en formato de variable en la ventana de asignación de planes:** En la ventana de asignación de planes de entrenamiento, algunos textos se muestran en formato de variable (por ejemplo, `training_plan`) en lugar del nombre legible de la sección, lo que genera confusión en el usuario.
- **Desmarcado de checkbox de ejercicios al navegar de vuelta a la lista:** Los checkbox de selección de ejercicios en la lista de entrenamientos se desmarcan automáticamente cuando el usuario accede a un ejercicio en particular y luego regresa a la lista. Esto interrumpe el flujo de selección de ejercicios, obligando al usuario a realizar la selección de nuevo.
- **Acceso permitido a ejercicios bloqueados:** Aunque ciertos ejercicios deberían estar bloqueados para algunos usuarios, la aplicación permite que se acceda a estos ejercicios al seleccionar la opción correspondiente. Este problema afecta a la seguridad y control del acceso a los contenidos.

- **Actualización incorrecta de los datos del perfil en la edición de usuario:** Al modificar la información del perfil del usuario, los datos no se actualizan correctamente, lo que obliga al usuario a realizar múltiples intentos para que los cambios sean visibles.
- **Error al actualizar la gráfica de peso tras modificar el valor:** La funcionalidad de registro de peso permite que el usuario introduzca nuevos valores, pero la gráfica de peso no se actualiza automáticamente tras la modificación. Esto genera confusión, ya que el usuario no puede visualizar su progreso de forma inmediata.
- **Desplazamiento incorrecto en la pantalla de detalles de ejercicio:** Al acceder a la pantalla de detalles de un ejercicio, el desplazamiento de la página no se ajusta adecuadamente, dejando partes importantes del contenido fuera de la vista inicial y obligando al usuario a desplazarse manualmente.
- **Mensaje de error genérico al asignar un plan de entrenamiento a un usuario existente:** Al intentar asignar un plan de entrenamiento a un usuario que ya tiene uno asignado, la aplicación muestra un mensaje de error genérico sin detallar el problema. Esto dificulta la identificación de la causa del error para los entrenadores y administradores.
- **Problema con la barra de navegación que muestra múltiples botones seleccionados:** En la barra de navegación, algunos botones permanecen marcados como seleccionados en color incluso después de cambiar de sección, lo que genera confusión en el usuario al no saber en qué pantalla se encuentra realmente.
- **Permisos incorrectos para nutricionistas y entrenadores en la gestión de usuarios:** La aplicación permite a los nutricionistas y entrenadores editar, crear y eliminar usuarios, cuando estas acciones deberían estar restringidas únicamente a los administradores. Esto representa un problema de seguridad y control de acceso en la aplicación.

Estos errores se identificaron a lo largo de las pruebas de usuario y serán abordados en futuras iteraciones de desarrollo para mejorar la experiencia general de los usuarios y asegurar el correcto funcionamiento de la aplicación.

5.4. Sugerencias de mejora

Durante las pruebas de usuario, se recibieron varias sugerencias de mejora enfocadas en optimizar la usabilidad, accesibilidad y experiencia general de la aplicación. A continuación, se enumeran las principales recomendaciones realizadas por los usuarios:

- **Permitir marcar un ejercicio como hecho desde la vista de detalles del ejercicio:** Los usuarios sugieren la posibilidad de marcar un ejercicio como completado directamente desde su pantalla de detalles, en lugar de tener que

regresar a la lista de ejercicios para hacerlo. Esto facilitaría el seguimiento del progreso en cada ejercicio sin interrumpir el flujo.

- **Incluir un cronómetro en la vista del ejercicio para calcular el descanso automáticamente:** Los usuarios propusieron añadir un cronómetro integrado en la pantalla de detalles de los ejercicios, para poder medir el tiempo de descanso entre series sin tener que salir de la aplicación o utilizar un dispositivo externo.
- **Reubicar la barra de progreso del entrenamiento en la parte superior de la pantalla:** Actualmente, la barra de progreso se encuentra en la parte inferior, lo que algunos usuarios consideran poco intuitivo. Colocarla en la parte superior de la pantalla permitiría un seguimiento más visible del avance durante el entrenamiento.
- **Fijar la barra de navegación superior en la sección de estadísticas:** Los usuarios recomendaron mantener fija la barra de navegación superior en la vista de estadísticas para facilitar el acceso a las opciones sin necesidad de desplazarse hacia arriba en la pantalla.
- **Mantener fija la barra de navegación en las secciones de nutrición y entrenamientos:** Similar a la sugerencia anterior, los usuarios expresaron que una barra de navegación fija en las secciones de nutrición y entrenamientos mejoraría la accesibilidad y usabilidad al facilitar el cambio entre diferentes opciones sin tener que desplazarse.
- **Agregar una vista detallada para observar la información de usuarios, entrenamientos, ejercicios, etc.:** Los usuarios sugirieron la incorporación de vistas detalladas para obtener información específica de cada usuario, entrenamiento, ejercicio, entre otros, lo que permitiría una navegación más completa y profunda en cada sección.
- **Revisar y mejorar la paleta de colores de la aplicación:** Anteriormente, la paleta incluía un verde lima que ahora ha sido reemplazado por tonos más grises. Los usuarios sugieren considerar una combinación de colores que sea atractiva y funcional, manteniendo la consistencia con los objetivos de la aplicación.
- **Añadir texto a los iconos de la barra de navegación inferior:** La barra de navegación inferior actualmente contiene solo iconos, lo que algunos usuarios consideran insuficiente para entender rápidamente la función de cada sección. Se recomienda incluir texto junto a los iconos para hacer la navegación más intuitiva, especialmente para usuarios nuevos.

Estas sugerencias de mejora proporcionan una base valiosa para realizar ajustes en la aplicación, incrementando la facilidad de uso y mejorando la experiencia general del usuario.

6. RESULTADOS

6.1. Resultados obtenidos

La sección de resultados presenta de manera clara y detallada los logros alcanzados durante el desarrollo de *FITPROX*, tanto desde una perspectiva técnica como funcional. Se muestran las funcionalidades implementadas, los objetivos cumplidos y las pruebas realizadas para garantizar la calidad y confiabilidad del sistema.

Incluye capturas de pantalla de las principales vistas de la aplicación, ilustrando su diseño e interfaz, y evidenciando cómo se cumplen los objetivos iniciales planteados. Además, se presentan métricas clave, como el porcentaje de cobertura de las pruebas automatizadas en el backend, que alcanzó un **90 %** en funcionalidades críticas. Estas pruebas permitieron detectar errores de manera temprana y mejorar continuamente el sistema.

En resumen, esta sección proporciona una visión integral de los resultados obtenidos, mostrando el estado final del proyecto y el camino recorrido para lograr un sistema funcional, estable y orientado a las necesidades de los usuarios.

Cumplimiento de objetivos planteados

A continuación, se detalla cómo los objetivos iniciales del proyecto fueron alcanzados, respaldados por las funcionalidades específicas de la aplicación:

Planes de entrenamiento semanales Se cumplió el objetivo de proporcionar planes de entrenamiento organizados semanalmente. Cada plan incluye:

- Lista de ejercicios diarios con detalles como series, repeticiones y tiempo de descanso.
- Seguimiento del progreso semanal de los entrenamientos completados.

La Figura 8 muestra cómo los usuarios pueden consultar su plan semanal, organizado por días y con acceso rápido a los detalles de cada ejercicio.

Recomendaciones nutricionales Se implementaron recomendaciones personalizadas basadas en las necesidades del usuario, incluyendo:

- Cantidad diaria recomendada de macronutrientes (proteínas, carbohidratos y grasas).
- Representación visual en gráficas que facilita la comprensión de la distribución calórica.

La Figura 9 ilustra cómo los usuarios visualizan sus requerimientos nutricionales diarios de forma clara y fácil de interpretar.

Interfaz de usuario intuitiva Se diseñó una interfaz sencilla e intuitiva, lograda mediante:

- Un diseño minimalista con navegación clara entre secciones como entrenamiento, nutrición y perfil.
- Componentes visuales consistentes en todas las vistas.

Seguimiento de objetivos físicos Los usuarios pueden realizar un seguimiento de su progreso físico, destacándose:

- Historial de cambios en datos como peso, medidas y nivel de actividad.
- Actualización de objetivos físicos directamente desde el perfil.

La Figura 10 muestra cómo consultar el historial de cambios y ajustar objetivos desde la vista de seguimiento.

Conclusión Se ha demostrado cómo los resultados obtenidos cumplen con los objetivos iniciales. Cada funcionalidad implementada está alineada con las metas propuestas, garantizando una aplicación funcional y una experiencia de usuario satisfactoria.

Resultados técnicos

En el desarrollo del proyecto se lograron importantes avances en pruebas y aseguramiento de calidad, tanto en el backend como en el frontend.

Pruebas unitarias del backend Se implementaron **220 pruebas unitarias** para verificar funcionalidades del backend, como la gestión de usuarios, la creación y modificación de planes de entrenamiento y nutrición, y las operaciones CRUD de entidades clave.

El sistema alcanzó un **90 % de cobertura de código**, reflejando un alto grado de confiabilidad en las funcionalidades críticas. La Figura 7 muestra los resultados obtenidos en el informe de cobertura generado tras ejecutar las pruebas.

Las pruebas ayudaron a identificar y solucionar problemas en etapas tempranas del desarrollo, como:

- Errores en permisos de acceso para ciertas operaciones.
- Validaciones faltantes en los modelos y vistas.
- Problemas en la lógica de negocio para la asignación de planes de entrenamiento y nutrición.
- Comportamientos inesperados en operaciones de actualización y eliminación.

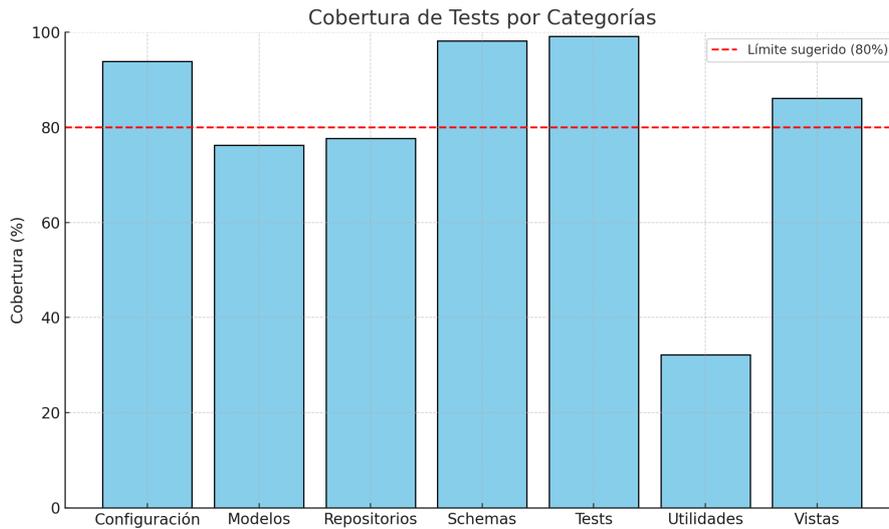


Figura 7: Informe de cobertura de las pruebas unitarias del backend.

Pruebas manuales Además de las pruebas automatizadas, se realizaron **pruebas manuales** en el frontend para verificar la correcta integración con el backend y la funcionalidad general de la aplicación. Estas pruebas incluyeron:

- Validación de la navegación entre vistas (entrenamiento, nutrición, perfil).
- Verificación de la visualización y actualización de datos en tiempo real.
- Pruebas de usabilidad para asegurar que la interfaz fuera intuitiva para los usuarios.
- Pruebas de integración para confirmar que las respuestas del backend eran procesadas y mostradas correctamente en el frontend.

Estas pruebas manuales permitieron ajustar aspectos visuales y funcionales de la aplicación, mejorando la experiencia del usuario y asegurando la coherencia entre el diseño y las funcionalidades implementadas.

Vista de administrador

Una de las funcionalidades clave de **FITPROX** es la vista de administrador, diseñada para facilitar la gestión de usuarios y el seguimiento general de la actividad en la aplicación. Esta vista proporciona herramientas para administrar usuarios, asignar planes pendientes y consultar estadísticas globales, asegurando un control eficiente de la plataforma.

Gestión de usuarios La Figura 11 muestra cómo el administrador puede visualizar una lista de usuarios registrados, junto con opciones para ver, editar o eliminar su información. Esto permite mantener un control centralizado sobre los datos de los usuarios, garantizando una experiencia segura y organizada.

Gestión de planes de nutrición La vista de administrador incluye herramientas para crear, editar o eliminar planes de nutrición disponibles en la plataforma. Esto asegura que los profesionales tengan acceso a un catálogo actualizado y adaptado a las necesidades de los usuarios.

Gestión de planes de entrenamiento De manera similar, el administrador puede gestionar los planes de entrenamiento, añadiendo, modificando o eliminando planes según sea necesario. Esto garantiza que los entrenadores dispongan de opciones organizadas y actualizadas para asignar a los usuarios.

Asignación de planes pendientes Como se describió en la Sección 2.3, el flujo de gestión de usuarios y asignación de planes es esencial para la personalización dentro de *FitProx*. La vista de administrador integra herramientas para gestionar usuarios pendientes y asignar planes desde una interfaz clara y accesible, optimizando el proceso.

Conclusión La vista de administrador cumple un rol esencial en la operatividad de **FITPROX**, proporcionando funcionalidades robustas para gestionar la plataforma de manera eficiente. Estas herramientas aseguran un sistema bien administrado, adaptado tanto a usuarios como a profesionales.

Impacto y utilidad

La aplicación **FITPROX** ha generado un impacto significativo en los usuarios finales y profesionales del fitness gracias a funcionalidades clave que responden a sus necesidades específicas. A continuación, se detalla cómo los resultados obtenidos mejoran la accesibilidad, experiencia y gestión en el ámbito del fitness.

Impacto en los usuarios finales Acceso a planes personalizados de fitness: Los usuarios disponen de planes de entrenamiento y nutrición personalizados, lo que facilita la planificación y mejora la consistencia en su progreso hacia objetivos específicos.

Seguimiento y motivación: La funcionalidad de seguimiento del progreso físico permite visualizar avances a lo largo del tiempo, reforzando la motivación y promoviendo un compromiso continuo, esencial para mantener el enfoque a largo plazo.

Interfaz amigable e intuitiva: Un diseño limpio y organizado permite a los usuarios navegar fácilmente entre secciones, mejorando la experiencia general y reduciendo la curva de aprendizaje para nuevos usuarios.

Impacto en entrenadores y nutricionistas Gestión eficiente de usuarios: Los profesionales pueden gestionar clientes de forma eficiente a través de la vista de administrador, asignando o modificando planes y siguiendo el progreso individual. Por ejemplo, la funcionalidad de asignación de planes pendientes reduce el tiempo necesario para organizar rutinas y dietas personalizadas.

Mejora en la comunicación: Aunque no implementado aún, se planea un sistema de chat en tiempo real que permitirá a los profesionales comunicarse directamente con los usuarios, agilizando modificaciones y resolviendo dudas de forma inmediata.

Centralización de datos: La aplicación centraliza toda la información de los clientes (entrenamientos, progreso, objetivos, planes), eliminando la necesidad de herramientas adicionales.

Conclusión El principal impacto de **FITPROX** radica en su capacidad para empoderar a usuarios y profesionales del fitness. Los usuarios finales disponen de herramientas completas para mejorar su salud y rendimiento, mientras que entrenadores y nutricionistas encuentran una plataforma eficiente para optimizar su trabajo. Esta combinación de utilidad y accesibilidad posiciona a la aplicación como una solución integral en el fitness personalizado.

Conclusión general de los resultados

En conclusión, los resultados obtenidos reflejan el éxito del proyecto en diversos aspectos clave. Las pruebas exhaustivas aseguraron un alto nivel de calidad técnica, con un 90% de cobertura en el backend, mientras que la implementación de mejoras derivadas de las pruebas demostró el compromiso con la optimización continua de la experiencia de usuario. Además, el impacto positivo en usuarios finales y profesionales del fitness, combinado con herramientas robustas de gestión para administradores, posiciona a **FITPROX** como una solución efectiva y completa en el ámbito del fitness personalizado. Estos logros resaltan la importancia de integrar retroalimentación constante y pruebas rigurosas para cumplir y superar los objetivos planteados.

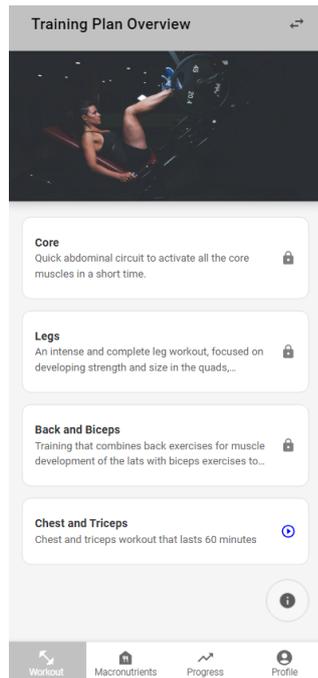


Figura 8: Vista del plan de entrenamiento semanal.

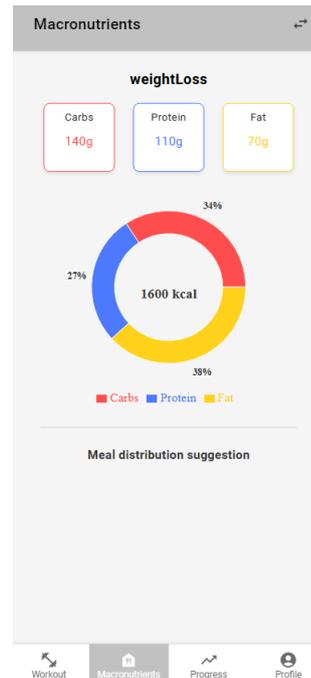


Figura 9: Vista del plan nutricional semanal.

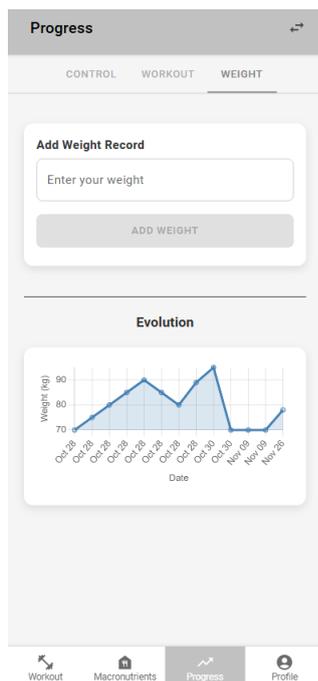


Figura 10: Vista del progreso del usuario.

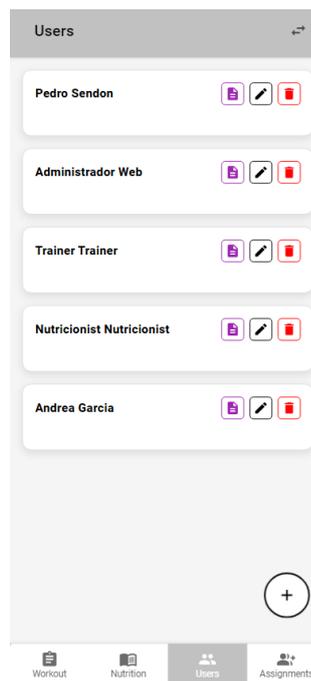


Figura 11: Vista de la lista de usuarios desde Administrador.

6.2. Cambios implementados tras las pruebas

Tras analizar las sugerencias de mejora y los errores encontrados durante las pruebas de usuario, se implementaron diversas correcciones y mejoras para optimi-

zar la funcionalidad y la experiencia del usuario. A continuación, se describen los principales cambios realizados:

Corrección de errores técnicos

- **Planes de nutrición:** Solucionado un error que mostraba "método no permitido." al añadir o modificar planes.
- **Edición de usuarios:** Corregido un problema que impedía actualizar los datos de los usuarios.
- **Acceso a ejercicios bloqueados:** Ahora solo se permite acceder a ejercicios disponibles, garantizando la consistencia.
- **Gráfica de peso:** Refleja correctamente los cambios en el progreso de los usuarios.
- **Restricción de accesos:** Se limitó a administradores el acceso a funciones de gestión de usuarios.
- **Barra de navegación:** Resueltos problemas de selección simultánea de botones, mejorando la claridad visual.

Mejoras de usabilidad

- **Nuevas funcionalidades:**
 - Cronómetro para calcular automáticamente el tiempo de descanso en la vista de ejercicio.
 - Marcar ejercicios como completados desde la vista de detalles.
- **Mejoras en la experiencia visual:**
 - Reubicación de la barra de progreso en la parte superior para mayor visibilidad.
 - Actualización de la paleta de colores a tonos más equilibrados.
 - Fijación de barras de navegación en secciones clave (estadísticas, nutrición y entrenamientos).
 - Inclusión de texto junto a iconos en la barra inferior para facilitar su comprensión.
- **Vistas detalladas:** Añadidas vistas para mostrar información específica de usuarios, entrenamientos y ejercicios.

Estos cambios reflejan un compromiso constante con la mejora continua, basándose en la retroalimentación de los usuarios y el análisis de sus experiencias. Al implementar estas correcciones y mejoras, se ofrece una plataforma más intuitiva, funcional y alineada con las necesidades de todos los usuarios.

7. CONCLUSIONES

Cumplimiento de los objetivos iniciales

El desarrollo de *FitProx* ha demostrado ser un proyecto ambicioso y enriquecedor, logrando cumplir con los objetivos planteados al inicio del trabajo. La aplicación ofrece planes de entrenamiento y nutrición personalizados, integrando una interfaz intuitiva que facilita la navegación y el acceso a las diferentes funcionalidades. Además, se implementaron herramientas robustas para el seguimiento de los objetivos físicos de los usuarios, promoviendo su progreso y bienestar integral.

Desde un punto de vista técnico, se alcanzaron logros significativos, como un alto grado de cobertura de pruebas automatizadas y la implementación de un backend y frontend integrados que garantizan la estabilidad y funcionalidad del sistema.

Aprendizajes adquiridos

Este proyecto ha sido una experiencia de aprendizaje profunda en múltiples aspectos del desarrollo de software. Algunos de los aprendizajes más destacados incluyen:

- *La importancia de una buena planificación previa:* Una base sólida en la organización inicial facilita el desarrollo y previene problemas a futuro.
- *Gestión de un proyecto a gran escala:* Afrontar un sistema con una cantidad significativa de código ha reforzado la importancia de dividir el proyecto en módulos y subsecciones claras para mantener el orden.
- *Mejora en la calidad del código:* Se han adoptado buenas prácticas como la creación de funciones reutilizables que mejoran la legibilidad y mantenibilidad del código.
- *Primeros pasos en despliegue:* Este trabajo me permitió realizar mi primer despliegue utilizando *Docker Compose*, adquiriendo una habilidad fundamental para futuros proyectos.
- *Adaptación a nuevas tecnologías:* Durante el desarrollo, aprendí a trabajar con herramientas que no había utilizado previamente, como *Google Cloud Storage*, fortaleciendo mi capacidad de aprendizaje autodidacta.
- *Pruebas y calidad del software:* La implementación de más de 200 pruebas unitarias y funcionales no solo aseguró la calidad del sistema, sino que también me ayudó a entender mejor cómo prevenir y solucionar errores de manera eficiente.
- *Trabajo estructurado y metodológico:* Aprendí a estructurar mi tiempo y tareas para cumplir con plazos definidos, asegurando un balance adecuado entre desarrollo, pruebas y documentación.

- *Comunicación técnica:* La necesidad de documentar cada parte del proyecto y explicar las decisiones técnicas tomadas me permitió mejorar mis habilidades de comunicación en contextos profesionales.
- *Enfoque en la experiencia de usuario:* La retroalimentación obtenida durante las pruebas de usabilidad me ayudó a valorar la importancia de priorizar una interfaz amigable y accesible para los usuarios.
- *Resolución de problemas bajo presión:* Enfrentar errores críticos en momentos avanzados del desarrollo fortaleció mi capacidad para manejar situaciones de estrés y encontrar soluciones prácticas.
- *Colaboración potencial en equipo:* Aunque este proyecto fue desarrollado de forma individual, entendí la importancia de mantener un código organizado y claro que facilite la colaboración futura.
- *Integración de sistemas:* La conexión entre el backend, frontend y servicios externos como Google Cloud Storage me permitió profundizar en la integración de componentes para crear un sistema cohesivo.
- *Aprendizaje continuo:* Este proyecto me recordó que el aprendizaje no se detiene; cada error, desafío técnico o retroalimentación es una oportunidad para crecer como desarrollador.

Impacto del proyecto

El impacto de *FitProx* es significativo tanto a nivel académico como profesional. Académicamente, este proyecto representa la culminación de mi formación, reflejando los conocimientos adquiridos durante la carrera. Profesionalmente, este sistema está diseñado para ser una herramienta clave en mi futuro como preparador físico, facilitando la creación y seguimiento de planes personalizados para mis futuros clientes. Esto permitirá ofrecer un servicio más eficiente, accesible y adaptado a las necesidades individuales de cada usuario.

Reflexión personal

Desarrollar *FitProx* ha sido una experiencia transformadora. Este proyecto no solo ha puesto a prueba mis habilidades técnicas, sino que también ha resaltado la importancia de la organización y la gestión del tiempo. Gracias a una planificación cuidadosa, pude dedicar el tiempo necesario a cada etapa del proyecto, asegurando su finalización dentro de los plazos establecidos. Completar esta aplicación por mí mismo ha sido una fuente de gran satisfacción, demostrando los resultados de años de estudio y esfuerzo.

Conclusión general

En conclusión, *FitProx* no solo cumple con los objetivos planteados, sino que los supera al ofrecer una solución robusta, funcional y centrada en las necesidades

de los usuarios. Este proyecto subraya la importancia de integrar retroalimentación constante y pruebas rigurosas en el desarrollo de software, lo que permitió garantizar un sistema confiable y bien diseñado. Además, establece una base sólida para futuras mejoras y expansiones, posicionando *FitProx* como una herramienta clave en el ámbito del fitness personalizado.

8. LÍNEAS DE DESARROLLO FUTURO

En este apartado se detallan posibles mejoras y cambios que podrían implementarse en futuras versiones de la aplicación, orientados a la expansión de funcionalidades y al fortalecimiento de la seguridad y escalabilidad.

8.1. Biblioteca de alimentos

Una funcionalidad a añadir sería una *biblioteca de alimentos*, que permitiría a los usuarios acceder a un catálogo de alimentos clasificados por *categorías* (frutas, vegetales, carnes, etc.), incluyendo datos como *calorías*, *macronutrientes* (proteínas, carbohidratos y grasas), y otros aspectos como *fibra*, *vitaminas* y *minerales*. Esto mejoraría la experiencia de los usuarios, facilitando la creación de dietas personalizadas y vinculándose con las actuales *recomendaciones nutricionales* para sugerencias más precisas.

8.2. Sistema de chat con entrenadores y nutricionistas

Se podría implementar un *sistema de chat* en tiempo real para que los usuarios se comuniquen con *entrenadores* y *nutricionistas*, facilitando modificaciones de planes y resolución de dudas. Este sistema incluiría:

- *Mensajería en tiempo real* con *websockets*.
- *Historial de conversaciones* almacenado en la base de datos.
- *Asignación personalizada* con el profesional asignado.
- *Notificaciones* de nuevos mensajes.
- *Cifrado de extremo a extremo* para proteger datos sensibles.

Esto optimizaría la comunicación y mejoraría la experiencia del usuario al proporcionar asesoramiento rápido y personalizado.

8.3. Mejoras en ciberseguridad

Es fundamental reforzar la *ciberseguridad* de la aplicación con medidas adicionales:

- *Autenticación de dos factores (2FA)*: Añadir una segunda capa de protección para el *inicio de sesión*.
- *Cifrado avanzado de datos*: Migrar a *AES-256* para proteger datos sensibles.
- *Monitorización y auditoría*: Detectar accesos no autorizados en tiempo real.
- *Políticas de contraseñas más estrictas*: Exigir actualizaciones periódicas y bloqueo temporal tras varios intentos fallidos.

Estas medidas fortalecerían la seguridad y aumentarían la confianza de los usuarios en la protección de sus datos.

8.4. Optimización de la experiencia del usuario (UX)

Aunque la aplicación cuenta con un diseño intuitivo, se podrían perfeccionar aspectos como:

- *Interfaz adaptativa*: Garantizar que sea completamente *responsive*.
- *Personalización de la interfaz*: Permitir modificar colores, fuentes y disposición del panel de control.
- *Gamificación*: Añadir insignias o recompensas para fomentar el compromiso y la motivación.

8.5. Escalabilidad y rendimiento

Para soportar un mayor número de usuarios, sería importante:

- Optimizar consultas a la base de datos.
- Implementar una arquitectura de *microservicios*.
- Usar *caché* para mejorar tiempos de respuesta.

8.6. Recuperación de contraseña

Un sistema de *recuperación de contraseña* mediante correo electrónico mejoraría la accesibilidad y seguridad, incluyendo:

- *Formulario de recuperación* para introducir el correo asociado.
- *Envío de enlace* con validez limitada.
- *Restablecimiento de contraseña* en una página segura.
- *Notificación de confirmación* tras el cambio.

8.7. Mantener sesión iniciada

La posibilidad de *mantener la sesión iniciada* mediante *tokens persistentes* mejoraría la experiencia de usuario, permitiendo acceder a la aplicación sin reingresar credenciales. Se incluiría una opción para activar o desactivar esta función en la pantalla de inicio de sesión.

8.8. Selección personalizada de días de entrenamiento

Además de elegir el número de días de entrenamiento, permitir a los usuarios seleccionar días específicos de la semana mediante un calendario semanal mejoraría la personalización de sus planes, adaptándolos mejor a sus horarios.

8.9. Inicio de sesión con Google, Microsoft y Facebook

Integrar opciones de *inicio de sesión* con *Google*, *Microsoft* y *Facebook* simplificaría el registro y acceso, mejorando la experiencia de usuario y aumentando la adopción al reducir fricciones.

8.10. Soporte multilingüe en datos generados por el usuario

Aunque la aplicación permite cambiar el idioma de la interfaz, los datos creados por los usuarios (nombres de ejercicios, planes de entrenamiento, etc.) permanecen en el idioma original. Para solucionar esto, se podrían implementar:

- **Soporte de traducciones en la base de datos:** Campos adicionales para almacenar traducciones.
- **Integración con servicios de traducción automática:** Uso de *Google Translate API* o *Microsoft Translator API*.
- **Selección de idioma al crear contenido:** Opción para añadir traducciones manuales.
- **Visualización adaptada al idioma del usuario:** Mostrar automáticamente las traducciones disponibles.

Esto fomentaría una comunidad más inclusiva y conectada.

Impacto esperado Estas mejoras posicionarían a **FITPROX** como una aplicación global, aumentando su alcance y usabilidad en contextos multilingües, potenciando su valor para una audiencia diversa.

9. ANEXO

Historias de Usuario

Este anexo incluye todas las historias de usuario elaboradas para el desarrollo de la aplicación *FitProx*. Cada historia de usuario describe una funcionalidad clave de la aplicación, definiendo los requisitos desde la perspectiva de los usuarios.

US-03 Iniciar sesión

- **Como:** Visitante de la app (app visitor).
- **Quiero:** Iniciar sesión.
- **Para:** Utilizar los servicios de la aplicación.

Criterios de Aceptación:

1. El usuario accede a la aplicación y presiona el botón de "Iniciar sesión", ubicado en el centro de la pantalla de bienvenida.
2. Aparece un campo de entrada (input) para que el usuario ingrese su correo electrónico. El correo debe estar en formato válido.
3. Se muestra un campo de entrada para la contraseña, que incluye un botón para alternar entre mostrar u ocultar la contraseña (función de "ver u ocultar").
4. Debajo de los campos de entrada, se presenta una opción con un checkbox que permite al usuario seleccionar "Recordar sesión".
5. También se muestra un enlace con el texto "¿Has olvidado tu contraseña?" para permitir la recuperación de la misma en caso de olvido.
6. El botón de "Iniciar sesión" estará visible.
7. Si las credenciales son correctas, el usuario será autenticado y redirigido al dashboard o pantalla principal de la aplicación.
8. Debajo del botón de iniciar sesión, se muestra un enlace con el texto "¿No tienes cuenta? Regístrate", que redirige a la pantalla de registro de la aplicación.

US-04 Completar Formulario

- **Como:** Usuario registrado.
- **Quiero:** Completar el formulario.
- **Para:** Acceder a los resultados personalizados y ajustar la experiencia de la aplicación a mis necesidades.

Criterios de Aceptación:

1. Tras completar el registro, el usuario es redirigido automáticamente a un formulario dividido en 6 secciones: Datos personales, Objetivos, Actividad física, Dieta y nutrición, Información de salud, Preferencias.
2. Datos personales: Altura (cm), Peso (kg).
3. Objetivos:
 - ¿Cuántos días estás dispuesto a dedicar al entrenamiento semanalmente? (Opciones: 1, 2, 3, 4, 5 o 6).
 - ¿Cuánto tiempo estás dispuesto a dedicar al entrenamiento diariamente? (Opciones: Menos de 1 hora, 1-2 horas, Más de 2 horas).
 - ¿Cuál es tu objetivo de pérdida de peso o aumento de masa muscular? (Número en kg o lbs).
4. Actividad física:
 - ¿Cuál es tu nivel de actividad física actual? (Opciones: Sedentario, Actividad ligera, Actividad moderada, Actividad intensa).
 - ¿Qué tipo de ejercicios realizas habitualmente? (Opciones: Cardio, Fuerza, Flexibilidad, Entrenamiento en circuito, Otros).
 - ¿Qué equipos o instalaciones tienes disponibles? (Opciones: Gimnasio completo, Pesas libres, Máquinas de cardio, Sin equipamiento, Otros).
5. Dieta y nutrición:
 - ¿Cuál es tu tipo de dieta actual? (Opciones: Omnívora, Vegetariana, Vegana, Baja en carbohidratos, Alta en proteínas, Otros).
 - ¿Tienes alguna restricción alimentaria? (Opciones: Alergias alimentarias, Intolerancias, Preferencias dietéticas, Ninguna).
 - ¿Cuántas comidas haces al día? (Número).
 - ¿Cómo evalúas tu ingesta actual de macronutrientes? (Opciones: Bajo en proteínas, Bajo en carbohidratos, Bajo en grasas, Balanceado, No lo sé).
6. Información de salud:
 - ¿Tienes alguna condición médica relevante que deba considerarse? (Opciones: Diabetes, Hipertensión, Problemas articulares, Otros, Ninguno).
 - ¿Estás tomando algún suplemento? (Opciones: Proteína, Multivitamínico, Omega-3, Otros, Ninguno).
7. Preferencias:

- ¿Qué tipo de entrenamiento prefieres? (Opciones: Entrenamiento en casa, Entrenamiento en gimnasio, Ejercicios al aire libre, Clases grupales, Otros).
 - ¿Tienes alguna preferencia en cuanto a tipos de ejercicios o actividades? (Texto libre).
8. El usuario deberá completar todos los campos obligatorios antes de poder continuar. En caso de que algún campo esté vacío o no sea válido, se mostrará un mensaje de error indicando qué se debe corregir.
 9. Una vez completado el formulario, el usuario será redirigido a la sección de resultados personalizados.

US-05 Página de entrenamientos

- **Como:** Usuario registrado.
- **Quiero:** Ver la página principal de la aplicación
- **Para:** Acceder a todas las funciones de la app y gestionar mis entrenamientos.

Criterios de aceptación:

1. La página principal incluirá una navbar en la parte inferior con cuatro pestañas: Entrenamiento (la que se mostrará inicialmente), Macronutrientes, Progreso, Perfil.
2. Después de iniciar sesión o completar el formulario inicial, el usuario será redirigido automáticamente a la página principal de la app.
3. Al cargar la página principal, la pestaña Entrenamiento estará seleccionada por defecto.
4. En la página de entrenamiento, el usuario verá su planificación semanal de entrenamientos, separada por cada día de la semana.
5. Cada día de entrenamiento se mostrará como un botón o tarjeta. Al presionar uno de estos días, el usuario será redirigido a la página de detalle del entrenamiento de ese día.
6. Al hacer clic en alguna de las pestañas de la navbar (Macronutrientes, Progreso, Perfil), el usuario será redirigido a la página correspondiente.
7. Debe estar presente una barra de navegación inferior con los siguientes iconos que permitan cambiar entre las distintas secciones de la app: “Workout” (resaltado porque es la pestaña activa), “Macros”, “Progress”, “Perfil”.

US-06 Acceder a entrenamiento diario

- **Como:** Usuario registrado.
- **Quiero:** Ver la página de ejercicios diarios.
- **Para:** Poder acceder a cada ejercicio específico y gestionar mi progreso.

Criterios de aceptación:

1. Navegación desde la sección de entrenamientos: Al seleccionar un día específico desde la página de planificación semanal de entrenamientos, el usuario será redirigido a la página de ejercicios del día correspondiente.
2. La página mostrará una lista de ejercicios planificados para ese día, con la siguiente información visible para cada uno: Nombre del ejercicio, Número de series, Número de repeticiones.
3. Cada ejercicio se mostrará como un botón o tarjeta.
4. Al hacer clic en uno de estos botones, el usuario será redirigido a la página detallada del ejercicio, donde podrá ver información específica y guías para realizarlo.
5. A la derecha de cada botón o tarjeta de ejercicio habrá un checkbox que permitirá al usuario marcar ese ejercicio como completado.
6. Los ejercicios completados actualizarán el progreso general del entrenamiento.
7. En la parte inferior de la página se mostrará un resumen con: Tiempo estimado total del entrenamiento, Número total de series, Número total de repeticiones.
8. Debajo del resumen, se mostrará una barra de progreso que reflejará el porcentaje de entrenamiento completado basado en los ejercicios marcados como completados.
9. Al final de la página habrá un botón que permitirá al usuario marcar el entrenamiento completo una vez que haya completado el entrenamiento.
10. Debe estar presente una barra de navegación inferior con los siguientes iconos que permitan cambiar entre las distintas secciones de la app: "Workout" (resaltado porque es la pestaña activa), "Macros", "Progress", "Perfil".

US-07 Acceder a ejercicio

- **Como:** usuario que está realizando un entrenamiento.
- **Quiero:** ver la descripción detallada de cada ejercicio.
- **Para:** seguir correctamente las instrucciones y completar el entrenamiento con la técnica adecuada.

Criterios de Aceptación

1. Al seleccionar un ejercicio específico dentro del entrenamiento, se debe abrir una pantalla que muestre el nombre del ejercicio en la parte superior de la página.
2. Debe haber un visualizador de imágenes en la parte central de la pantalla, que permita deslizar entre diferentes imágenes o instrucciones visuales del ejercicio, usando flechas a la izquierda y derecha.
3. Debajo de la imagen se deben mostrar etiquetas con las zonas musculares que se trabajan, como se ve en el mock-up: bíceps, dorsal, trapecio, entre otros.
4. Las etiquetas deben ser ajustables de acuerdo a los músculos trabajados en el ejercicio en particular.
5. Al lado del nombre del ejercicio, debe haber un botón o enlace que abra un modal con la descripción completa del ejercicio (técnica adecuada, errores comunes, y beneficios de realizarlo).
6. Deben aparecer tres círculos grandes en la parte inferior con los parámetros del ejercicio: Series, Repeticiones, Tiempo de descanso.
7. Debe estar presente una barra de navegación inferior con los siguientes iconos que permitan cambiar entre las distintas secciones de la app: **Workout** (resaltado porque es la pestaña activa), **Macros**, **Progress**, **Perfil**.

US-07 Acceder a ejercicio

- **Como:** Usuario que está realizando un entrenamiento
- **Quiero:** Ver la descripción detallada de cada ejercicio
- **Para:** Seguir correctamente las instrucciones y completar el entrenamiento con la técnica adecuada

Criterios de aceptación:

1. Al seleccionar un ejercicio específico dentro del entrenamiento, se debe abrir una pantalla que muestre el nombre del ejercicio en la parte superior de la página.
2. Debe haber un visualizador de imágenes en la parte central de la pantalla, que permita deslizar entre diferentes imágenes o instrucciones visuales del ejercicio, usando flechas a la izquierda y derecha.
3. Debajo de la imagen se deben mostrar etiquetas con las zonas musculares que se trabajan, como se ve en el mock-up: bíceps, dorsal, trapecio, entre otros.
4. Las etiquetas deben ser ajustables de acuerdo a los músculos trabajados en el ejercicio en particular.

5. Al lado del nombre del ejercicio, debe haber un botón o enlace que abra un modal con la descripción completa del ejercicio (técnica adecuada, errores comunes, y beneficios de realizarlo).
6. Deben aparecer tres círculos grandes en la parte inferior con los parámetros del ejercicio: Series, Repeticiones, Tiempo de descanso.
7. Debe estar presente una barra de navegación inferior con los siguientes iconos que permitan cambiar entre las distintas secciones de la app: "Workout" (resaltado porque es la pestaña activa), "Macros", "Progress", "Perfil".

US-08 Página de Macronutrientes

- **Como:** Usuario registrado
- **Quiero:** Ver la página de macronutrientes
- **Para:** Poder acceder a la información de mi dieta y seguimiento diario

Criterios de Aceptación:

1. Al pulsar el botón de "Macronutrientes" en la barra de navegación (navbar), se debe abrir la pestaña de macronutrientes.
2. En esta pestaña se mostrará la siguiente información:
 - Tipo de dieta: Aparece como título en la parte superior de la página, por ejemplo, "Dieta alta en proteínas".
 - Calorías totales (Kcal): Se mostrarán justo debajo del título.
 - Macronutrientes (Carbohidratos, Proteínas, Grasas): Se deben mostrar en tarjetas con la siguiente información: Gramos a consumir de cada macronutriente, Calorías que aporta cada macronutriente.
 - Gráfico circular: Representa el porcentaje de cada macronutriente con diferentes colores.
3. Debajo de esta información se debe incluir una tabla con sugerencias de distribución de comidas. Esta tabla tendrá la siguiente estructura: Las filas indican los nombres de las comidas: Desayuno, Almuerzo, Cena y Snacks. Las celdas mostrarán los porcentajes recomendados de cada comida, por ejemplo: "Desayuno 20 %, Almuerzo 40 %".
4. Debe estar presente una barra de navegación inferior con los siguientes iconos que permitan cambiar entre las distintas secciones de la app: "Workout" (resaltado porque es la pestaña activa), "Macros", "Progress", "Perfil".

US-09 Página del perfil

- **Como:** Usuario registrado
- **Quiero:** Ver la página de mi perfil
- **Para:** Poder consultar y modificar mi información personal y mis objetivos.

Criterios de aceptación:

1. En la parte superior de la página habrá un avatar circular con la foto de perfil.
2. Si el usuario no ha cargado una foto, se mostrará un icono predeterminado.
3. Se mostrará la siguiente información personal: Nombre de usuario, Correo electrónico, Edad, Altura, Peso inicial, Peso actual.
4. La página mostrará los objetivos físicos del usuario, tales como: Objetivo físico (perder peso, ganar masa muscular, mantener peso, etc.), Nivel de actividad física, Frecuencia de entrenamiento (días por semana).
5. Habrá un botón `.Editar.en` la parte superior derecha que permitirá al usuario modificar su información personal.
6. Al hacer clic en el botón, el usuario será redirigido a una página donde podrá editar su perfil.
7. El usuario podrá cerrar sesión a través de un botón ubicado en la parte inferior de la página. Al hacer clic, se cerrará la sesión y se redirigirá al usuario a la página de inicio de sesión.
8. La página mostrará una gráfica sencilla que reflejará la evolución del peso del usuario a lo largo del tiempo.
9. Debe estar presente una barra de navegación inferior con los siguientes iconos que permitan cambiar entre las distintas secciones de la app: "Workout" (resaltado porque es la pestaña activa), "Macros", "Progress", "Perfil".

US-10 Editar perfil

- **Como:** Usuario registrado
- **Quiero:** Editar la página de mi perfil
- **Para:** Poder modificar mi información personal y mis objetivos.

Criterios de aceptación:

1. Al acceder a la página de edición de perfil, debe mostrarse un encabezado que diga `.Editar Perfil.en` color verde.

2. Debe haber una sección con la foto de perfil con un borde verde lima y un botón para cambiar la foto de perfil que también esté en verde.
3. Debe aparecer un título en negrita que diga "Información Personal".
4. Deben mostrarse los siguientes campos editables con los valores actuales del usuario: Nombre, Apellido, Peso Actual (kg).
5. Debajo de estos campos, debe haber un botón verde con el texto "CAMBIAR CONTRASEÑA".
6. Debe haber un título que diga "Información Deportiva" en negrita.
7. Se deben mostrar los siguientes campos desplegados editables: Objetivo de Peso (con opciones como "Perder peso", "Ganar masa muscular", etc.), Nivel de Actividad (con opciones como "Sedentario", "Ligera", "Moderada", etc.), Frecuencia de Entrenos (días/semana).
8. Debe existir un botón verde grande que diga "GUARDAR CAMBIOS".
9. Al presionar el botón "GUARDAR CAMBIOS", los datos modificados deben ser enviados y guardados correctamente, redirigiendo al usuario a su perfil actualizado.
10. Debe estar presente una barra de navegación inferior con los siguientes iconos que permitan cambiar entre las distintas secciones de la app: "Workout" (resaltado porque es la pestaña activa), "Macros", "Progress", "Perfil".

US-12 Cambiar Contraseña

- **Como:** Usuario registrado
- **Quiero:** Tener la opción de cambiar mi contraseña desde la página de perfil
- **Para:** Poder actualizar mis credenciales de forma segura y mantener la integridad de mi cuenta

Criterios de Aceptación:

1. Al hacer clic en el botón "Cambiar Contraseña", debe aparecer un modal con el título "Cambiar contraseña".
2. El modal debe contener tres campos: Contraseña actual, Nueva contraseña, Confirmación de la nueva contraseña.
3. Cada campo debe tener el ícono de "mostrar/ocultar contraseña a la derecha".
4. Si las contraseñas no coinciden, debe aparecer un mensaje de error: "Las contraseñas no coinciden".
5. La nueva contraseña debe cumplir los requisitos de seguridad (mínimo 8 caracteres, una letra mayúscula, una minúscula, un número y un símbolo).

6. Si la contraseña actual no es correcta, debe aparecer un mensaje de error: "La contraseña actual es incorrecta".
7. El modal debe tener dos botones: "Cancelar" (blanco con borde verde) y "Confirmar" (verde).

US-13 Información del Ejercicio

- **Como:** Usuario que está realizando un entrenamiento
- **Quiero:** Ver la información detallada del ejercicio que estoy realizando
- **Para:** Seguir correctamente los pasos y asegurar que lo realizo con la técnica adecuada

Criterios de Aceptación:

1. Debe haber un header con el título "Exercise informationz" debe seguir el diseño estándar de la aplicación.
2. Debe mostrarse el nombre del ejercicio en negrita y con un tamaño más grande que el resto del texto.
3. Debajo del título del ejercicio, debe haber una breve descripción del ejercicio, explicando qué músculos trabaja.
4. Instrucciones paso a paso:
 - Cada paso debe estar numerado y el número debe mostrarse dentro de una etiqueta verde con fondo blanco.
 - Las instrucciones deben mostrarse en tarjetas que ocupen la mayor parte del ancho de la pantalla, con bordes redondeados y un pequeño margen entre cada paso.
5. Debe haber un botón "Close" que ocupe todo el ancho de la pantalla, con fondo verde lima y texto en blanco. Al presionar el botón, el modal debe cerrarse.

US-14 Ver entrenamiento (Admin / Entrenador)

- **Como:** Administrador o entrenador
- **Quiero:** Ver la lista de entrenamientos disponibles en el sistema
- **Para:** Gestionar los entrenamientos existentes, verificar su contenido y tomar acciones como modificar o eliminar entrenamientos que ya no sean adecuados

Criterios de Aceptación:

1. El usuario debe poder acceder a la pantalla de "Entrenamientos" desde la pestaña correspondiente en la barra de navegación superior, donde se muestran las opciones de "Workoutsz Exercises".

2. En la pantalla se debe mostrar una lista de los entrenamientos disponibles, con el nombre y una breve descripción de cada entrenamiento.
3. Cada entrenamiento debe mostrar dos botones:
 - **Modificar:** Abre la vista de modificación para editar los detalles del entrenamiento (que se describirá en la US-15).
 - **Eliminar:** Muestra un pop-up de confirmación antes de eliminar el entrenamiento.
4. El botón de Eliminar debe estar destacado en color rojo y al hacer clic, debe aparecer un pop-up solicitando confirmación de la eliminación del entrenamiento. El pop-up debe tener dos opciones:
 - **Confirmar:** Eliminará el entrenamiento de la base de datos.
 - **Cancelar:** Cerrará el pop-up sin realizar ninguna acción.
5. Al final de la pantalla debe haber un botón flotante verde -" que permita crear un nuevo entrenamiento, redirigiendo a la vista de creación de entrenamientos (que se describirá en la US-17).
6. La lista de entrenamientos debe ser visualmente clara, con tarjetas que incluyan el nombre del entrenamiento, la descripción y los botones de acción.

US-15 Modificar entrenamiento (Admin / Entrenador)

- **Como:** Administrador o Entrenador
- **Quiero:** Modificar los entrenamientos existentes en la aplicación, añadiendo o eliminando ejercicios, cambiando el nombre del entrenamiento, la descripción o la imagen/video asociado
- **Para:** Mantener actualizados y personalizados los entrenamientos según las necesidades de los usuarios

Criterios de Aceptación:

1. Se mostrará un formulario con los campos actuales del entrenamiento, incluyendo nombre, descripción, imagen/video y lista de ejercicios.
2. Los campos del nombre y la descripción estarán rellenos con la información actual del entrenamiento.
3. El usuario puede cambiar la imagen o video del entrenamiento mediante un botón "Cambiar imagen/video", que permite subir una nueva imagen desde el dispositivo o eliminar la existente.
4. El usuario podrá añadir nuevos ejercicios a la rutina presionando el botón "Add Exercise". Esto generará nuevas filas en el formulario con los siguientes campos:

- Nombre del ejercicio: Dropdown para seleccionar un ejercicio existente.
 - Sets: Número de series a realizar.
 - Reps: Número de repeticiones por serie.
 - Descanso: Tiempo de descanso en segundos entre sets.
5. Para los ejercicios ya añadidos, el usuario puede modificar el número de series (sets), repeticiones (reps) y descanso, así como cambiar el ejercicio seleccionado en cada caso.
 6. Un botón "Guardar cambios." estará disponible en la parte inferior de la pantalla.
 7. El usuario podrá cancelar la modificación del entrenamiento y regresar a la lista de entrenamientos sin guardar cambios, presionando un botón "Cancelar".

US-16 Crear entrenamientos (Admin / Entrenador)

- **Como:** Administrador o entrenador
- **Quiero:** Poder crear entrenamientos personalizados, añadiendo ejercicios, con la opción de definir el número de series, repeticiones y tiempo de descanso para cada ejercicio
- **Para:** Mantener la base de datos de entrenamientos actualizada y ofrecer rutinas personalizadas a los usuarios

Criterios de Aceptación:

1. Debe existir un campo obligatorio para el nombre del entrenamiento.
2. El campo descripción del entrenamiento debe ser opcional.
3. Debe haber un botón "Add Exercise" que permita al administrador/entrenador añadir nuevos ejercicios al entrenamiento.
4. Al pulsar "Add Exercise", debe mostrarse una card donde se pueda seleccionar un ejercicio existente, y definir:
 - Número de series.
 - Número de repeticiones.
 - Tiempo de descanso en segundos.
5. Debe haber un botón para "Upload Image/Video" que permita añadir un medio opcional para el entrenamiento (imagen o video).
6. Debe haber un botón "Cancel" en rojo para descartar los cambios y regresar a la lista de entrenamientos sin guardar.

7. Debe haber un botón "Save.^{en} verde para guardar los cambios, lo que creará un nuevo entrenamiento con los ejercicios seleccionados.
8. Al pulsar "Save", el entrenamiento creado debe ser guardado en la base de datos y aparecer en la lista de entrenamientos.

US-17 Ver ejercicios (Admin / Entrenador)

- **Como:** Administrador o entrenador
- **Quiero:** Poder visualizar la lista de ejercicios disponibles, con la opción de modificar o eliminar cada ejercicio
- **Para:** Gestionar los ejercicios disponibles en la aplicación y mantener la base de datos actualizada según las necesidades de los usuarios

Criterios de Aceptación:

1. En la vista principal debe mostrarse una lista de los ejercicios registrados, cada uno con el nombre visible.
2. Debe existir un botón "MODIFY.^{en} verde para cada ejercicio, que permita acceder a la vista de edición del ejercicio seleccionado.
3. Debe existir un botón "DELETE.^{en} rojo para cada ejercicio, que permita eliminar el ejercicio de la base de datos, tras una confirmación del usuario.
4. La lista de ejercicios debe ser dinámica, mostrando los ejercicios almacenados en la base de datos.
5. En la esquina inferior derecha debe haber un botón flotante ".^{en} verde, que redirija a la vista para crear un nuevo ejercicio.
6. Debe haber una barra superior que permita al usuario cambiar entre la vista de "Workoutsz la de .^{Ex}ercises".

US-18 Modificar ejercicios (Admin / Entrenador)

- **Como:** Administrador o entrenador
- **Quiero:** Poder modificar la información de los ejercicios existentes, incluyendo su nombre, grupos musculares, instrucciones y multimedia asociada
- **Para:** Asegurar que la información de los ejercicios en la aplicación está actualizada y adaptada a las necesidades de los usuarios

Criterios de Aceptación:

1. En la vista de modificación de ejercicios, debe mostrarse un formulario prellenado con los datos actuales del ejercicio seleccionado.

2. Debe ser posible cambiar:
 - El nombre del ejercicio.
 - La descripción.
 - Los grupos musculares trabajados (seleccionables en un desplegable).
 - Las instrucciones para realizar el ejercicio.
 - La imagen o video del ejercicio, con la opción de "Cambiar imagen/video".
3. Deben existir dos botones en la parte inferior:
 - **CANCEL** en rojo, que permite cancelar los cambios y volver a la lista de ejercicios.
 - **SAVE** en verde, que guarda los cambios realizados y actualiza el ejercicio en la base de datos.
4. Al intentar guardar sin haber realizado cambios, el botón "SAVE" debe permanecer habilitado para permitir actualizar el contenido.
5. Al guardar los cambios, debe mostrarse un mensaje de confirmación indicando que el ejercicio ha sido actualizado correctamente.
6. Al pulsar el botón "CANCEL", los cambios realizados deben descartarse y redirigir al usuario a la lista de ejercicios.
7. El campo para cambiar la imagen o video debe permitir la carga desde archivos locales del dispositivo.

US-19 Crear ejercicios (Admin / Entrenador)

- **Como:** Administrador o entrenador
- **Quiero:** Poder crear un nuevo ejercicio especificando su nombre, descripción, grupos musculares trabajados, instrucciones y cargar una imagen o video
- **Para:** Añadir nuevos ejercicios a la base de datos, asegurando que los usuarios dispongan de una variedad de ejercicios actualizados y completos

Criterios de Aceptación:

1. Debe mostrarse un formulario para ingresar los siguientes campos:
 - **Exercise Name:** Campo obligatorio para el nombre del ejercicio.
 - **Description:** Campo opcional para una breve descripción del ejercicio.
 - **Muscle Groups:** Un desplegable que permita seleccionar los grupos musculares implicados en el ejercicio.
 - **Instructions:** Campo para ingresar las instrucciones sobre cómo realizar el ejercicio.

- **Upload Image/Video:** Un botón para cargar un archivo multimedia (imagen o video) relacionado con el ejercicio.
2. Deben existir dos botones en la parte inferior:
 - **CANCEL** en rojo, que permite cancelar el proceso y volver a la lista de ejercicios sin guardar.
 - **SAVE** en verde, que guarda el nuevo ejercicio en la base de datos.
 3. Si la validación falla, deben mostrarse mensajes de error debajo de los campos correspondientes.

US-20 Ver usuarios (Admin / Entrenador / Nutricionista)

- **Como:** Administrador / Entrenador / Nutricionista
- **Quiero:** Poder visualizar una lista de todos los usuarios registrados en la aplicación, con opciones para asignar planes, modificar o eliminar a los usuarios
- **Para:** Gestionar eficientemente a los usuarios, asegurando que puedan recibir planes personalizados, realizar modificaciones en su perfil o eliminarlos si es necesario

Criterios de Aceptación:

1. Debe mostrarse una lista con el nombre completo y el correo electrónico de cada usuario registrado.
2. La lista debe incluir botones con las siguientes opciones para cada usuario:
 - **Assign:** Permite asignar un plan de entrenamiento o un plan nutricional al usuario.
 - **Modify:** Permite modificar la información del usuario.
 - **Delete:** Permite eliminar al usuario del sistema tras una confirmación.
3. Los botones de "Assign", "Modify" y "Delete" deben estar claramente visibles y diferenciados por colores.
4. Los roles de Administrador, Entrenador y Nutricionista deben poder acceder a esta vista y gestionar a los usuarios según sus permisos.
5. Los usuarios deben ser ordenados alfabéticamente por nombre o apellidos.
6. Debe haber un botón flotante en la esquina inferior derecha para agregar un nuevo usuario.

US-21 Modificar usuarios (Admin)

- **Como:** Administrador
- **Quiero:** Poder modificar la información de los usuarios existentes, como nombre, apellidos, peso, meta de peso, nivel de actividad, frecuencia de entrenamiento y rol
- **Para:** Actualizar la información de los usuarios en la base de datos de manera eficiente y mantener los perfiles personalizados según sus necesidades actuales

Criterios de Aceptación:

1. Debe mostrarse un formulario con los siguientes campos editables:
 - **Nombre:** Campo de texto para modificar el nombre del usuario.
 - **Apellidos:** Campo de texto para modificar los apellidos del usuario.
 - **Peso Actual (kg):** Campo numérico para introducir el peso actual del usuario.
 - **Meta de Peso:** Desplegable con opciones para seleccionar la meta de peso (por ejemplo, perder peso, ganar masa muscular, mantener peso).
 - **Nivel de Actividad:** Desplegable para seleccionar el nivel de actividad (sedentario, ligera, moderada, intensa).
 - **Frecuencia de Entrenamiento (días/semana):** Campo numérico para introducir la frecuencia de entrenamiento.
 - **Rol:** Desplegable con las opciones de roles (Cliente, Administrador, Entrenador, Nutricionista).
2. El administrador debe poder cambiar la foto de perfil del usuario al hacer clic en el botón "Cambiar foto".
3. Los campos obligatorios deben estar marcados con un asterisco (*) y deben validarse antes de permitir guardar los cambios.
4. Debe haber dos botones:
 - **Cancel:** Permite descartar los cambios y regresar a la vista anterior.
 - **Add/Save:** Permite guardar los cambios realizados en la información del usuario.
5. Al pulsar el botón "Save", los cambios deben reflejarse en la base de datos y mostrarse una confirmación de éxito.

US-22 Crear usuarios (Admin)

- **Como:** Administrador
- **Quiero:** Poder registrar nuevos usuarios en la aplicación proporcionando su información básica, como nombre, apellidos, correo electrónico, contraseña, género, fecha de nacimiento y rol
- **Para:** Agregar nuevos usuarios al sistema de manera eficiente y asegurando que cuenten con la información adecuada y los permisos correspondientes desde su creación

Criterios de Aceptación:

1. Debe mostrarse un formulario con los siguientes campos obligatorios:
 - **Nombre:** Campo de texto para introducir el nombre del usuario.
 - **Apellidos:** Campo de texto para introducir los apellidos del usuario.
 - **Correo electrónico:** Campo de texto para introducir la dirección de correo electrónico del usuario.
 - **Contraseña:** Campo de texto para definir la contraseña del usuario.
 - **Género:** Desplegable con opciones (Masculino, Femenino, Otro) para seleccionar el género del usuario.
 - **Rol:** Desplegable con opciones (Cliente, Administrador, Entrenador, Nutricionista) para asignar el rol al usuario.
 - **Fecha de nacimiento:** Selector de fecha para introducir la fecha de nacimiento del usuario.
2. Los campos obligatorios deben estar marcados con un asterisco (*) y deben validarse antes de permitir la creación del usuario.
3. Al pulsar el botón "Add", los datos deben guardarse en la base de datos y el administrador debe recibir una confirmación de que el usuario ha sido creado correctamente.
4. Debe haber un botón "Cancelar" que permita descartar el proceso de creación y regresar a la pantalla anterior sin guardar ningún cambio.

US-23 Cambiar entreno o plan nutricional a usuarios (Admin / entrenador / nutricionista)

- **Como:** Administrador, entrenador o nutricionista
- **Quiero:** Poder asignar o modificar el plan de entrenamiento y/o el plan nutricional de un usuario específico
- **Para:** Personalizar y adaptar los planes de entrenamiento y nutrición de cada usuario según sus objetivos y necesidades

Criterios de Aceptación:

1. Deben mostrarse dos menús desplegables:
 - **Seleccionar entrenamiento:** Desplegable que contiene la lista de entrenamientos disponibles para asignar.
 - **Seleccionar plan nutricional:** Desplegable que contiene la lista de planes nutricionales disponibles para asignar.
2. Es posible seleccionar solo uno o ambos planes (entrenamiento y/o plan nutricional).
3. El botón "Save" debe estar habilitado una vez se haya realizado alguna selección.
4. Al hacer clic en "Save", los planes seleccionados deben asignarse al usuario y debe mostrarse un mensaje de confirmación indicando que la operación fue exitosa.
5. El botón "Cancel" debe permitir descartar cualquier cambio y regresar a la vista anterior sin realizar modificaciones.

US-24 Ver estadísticas (Admin)

- **Como:** Administrador
- **Quiero:** Visualizar las estadísticas de uso de la aplicación para monitorear la popularidad de los ejercicios, entrenamientos y el crecimiento de la base de usuarios
- **Para:** Tomar decisiones informadas sobre cómo mejorar la experiencia de los usuarios y ajustar el contenido y los entrenamientos según las preferencias de los usuarios

Criterios de Aceptación:

1. Debe mostrarse un apartado con las estadísticas de popularidad de ejercicios, donde se visualizarán los nombres de los ejercicios y la cantidad de veces que han sido realizados, como se muestra en la vista "Exercise Popularity".
2. El administrador podrá ver el crecimiento de la plataforma en la sección "Platform Growth", que mostrará una gráfica de barras con el número de nuevos usuarios por mes.
3. El año de la gráfica de crecimiento de usuarios debe poder seleccionarse mediante un desplegable.
4. Los datos mostrados deben actualizarse dinámicamente según el año seleccionado en el desplegable.

5. Los gráficos deben ser claros y con colores distintivos (ej. verde para nuevos usuarios, púrpura para la popularidad de ejercicios).
6. El administrador debe poder acceder a esta sección desde el menú de navegación inferior, seleccionando la opción "Statistics".

US-25 Ver macronutrientes (Admin / Nutricionista)

- **Como:** Administrador o nutricionista
- **Quiero:** Visualizar los planes de macronutrientes disponibles en la aplicación, clasificados por tipo de dieta (pérdida de peso, ganancia muscular, mantenimiento)
- **Para:** Poder gestionar, modificar o eliminar planes de macronutrientes de manera efectiva y asegurar que las recomendaciones nutricionales estén alineadas con los objetivos de los usuarios

Criterios de Aceptación:

1. La vista debe mostrar los diferentes planes de macronutrientes clasificados por pestañas: **Weight Loss**, **Muscle Gain**, y **Maintenance**.
2. Cada plan de macronutrientes debe mostrar un resumen de calorías, proteínas, carbohidratos y grasas, tal como se visualiza en las tarjetas de la imagen.
3. El administrador/nutricionista debe poder modificar o eliminar un plan de macronutrientes utilizando los botones correspondientes (**Modificar** y **Eliminar**).
4. Al seleccionar el botón de **Modificar**, se debe redirigir al formulario de edición del plan seleccionado.
5. Al seleccionar el botón de **Eliminar**, debe mostrarse una confirmación antes de proceder a la eliminación del plan.
6. Un botón de **Añadir (+)** debe estar disponible en la parte inferior derecha para permitir la creación de nuevos planes de macronutrientes.
7. Los planes de macronutrientes deben estar organizados y mostrarse de manera clara para facilitar su gestión.

US-26 Crear plan nutricional (Admin / Nutricionista)

- **Como:** Administrador o nutricionista
- **Quiero:** Crear un nuevo plan nutricional especificando los valores de calorías, proteínas, carbohidratos, grasas, tipo de dieta y una descripción opcional
- **Para:** Agregar nuevas opciones de planes nutricionales a la base de datos para adaptarse a las necesidades y objetivos de los usuarios

Criterios de Aceptación:

1. La vista debe mostrar los siguientes campos para el nuevo plan nutricional:
 - **Kcal:** Campo obligatorio para definir las calorías del plan.
 - **Proteínas (g):** Campo obligatorio para definir la cantidad de proteínas.
 - **Carbohidratos (g):** Campo obligatorio para definir la cantidad de carbohidratos.
 - **Grasas (g):** Campo obligatorio para definir la cantidad de grasas.
 - **Tipo de dieta:** Seleccionable entre las opciones como pérdida de peso, ganancia muscular, mantenimiento.
 - **Descripción opcional:** Permite agregar una descripción adicional sobre el plan.
2. El botón **Cancelar** debe estar presente y debe permitir al usuario abandonar la creación del plan sin guardar datos, redirigiendo a la vista anterior.
3. El botón **Añadir** debe estar disponible y permitir que el nuevo plan se guarde en la base de datos.
4. Al hacer clic en **Añadir**, el plan nutricional debe crearse con los valores proporcionados y una confirmación debe mostrarse al usuario.
5. En caso de que haya errores (como campos obligatorios vacíos o con datos incorrectos), deben mostrarse mensajes de error debajo de los campos correspondientes.

US-27 Modificar plan nutricional (Admin / Nutricionista)

- **Como:** Administrador o nutricionista
- **Quiero:** Modificar un plan nutricional existente para ajustar las calorías, proteínas, carbohidratos, grasas, el tipo de dieta y la descripción opcional
- **Para:** Actualizar las recomendaciones nutricionales y asegurarse de que estén alineadas con los objetivos de los usuarios

Criterios de Aceptación:

1. La vista debe mostrar los campos editables para las siguientes propiedades de un plan nutricional:
 - **Kcal:** Calorías totales.
 - **Proteínas (g):** Cantidad de proteínas.
 - **Carbohidratos (g):** Cantidad de carbohidratos.
 - **Grasas (g):** Cantidad de grasas.

- **Tipo de dieta:** Seleccionable entre opciones como pérdida de peso, ganancia muscular, mantenimiento.
 - **Descripción opcional:** Permite agregar detalles adicionales sobre el plan.
2. El administrador o nutricionista debe poder editar estos campos.
 3. El botón **Cancelar** debe estar presente y debe redirigir a la vista anterior sin guardar cambios.
 4. El botón **Guardar** debe estar disponible y permitir que los cambios se apliquen y se guarden en la base de datos.
 5. Al hacer clic en **Guardar**, el usuario debe recibir una confirmación de que los cambios se han guardado correctamente.
 6. En caso de un error en el formulario (campos vacíos o con valores incorrectos), se deben mostrar mensajes de error debajo de los campos correspondientes.

Diseños de las Vistas

Este anexo presenta los diseños iniciales de las principales vistas de la aplicación *FitProx*. Estas primeras versiones sirvieron como guía para la implementación de las funcionalidades clave. A continuación, se incluyen las capturas de los diseños originales junto con una breve descripción de su propósito.

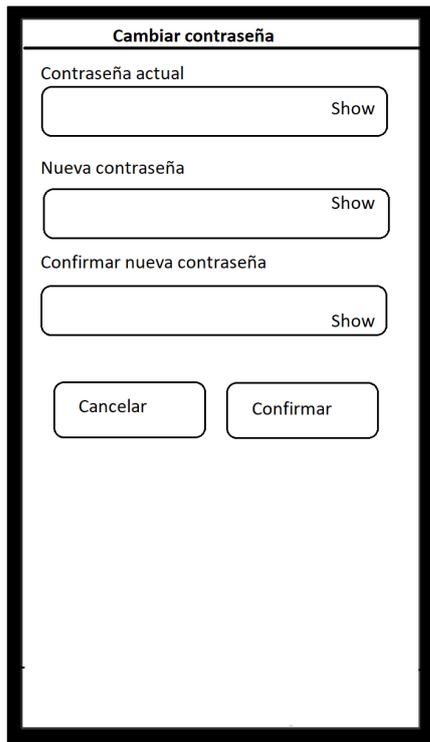


Figura 12: Diseño inicial de la pantalla para Cambiar Contraseña.

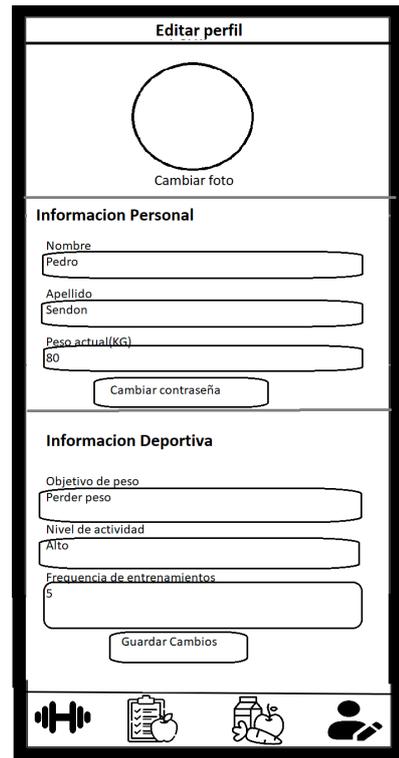


Figura 13: Vista de Edición de Perfil.



Figura 14: Vista de Ejercicio.

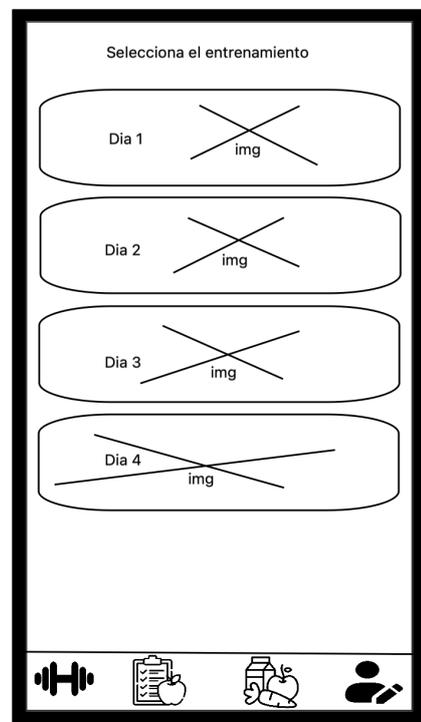


Figura 15: Pantalla de Entrenamiento.

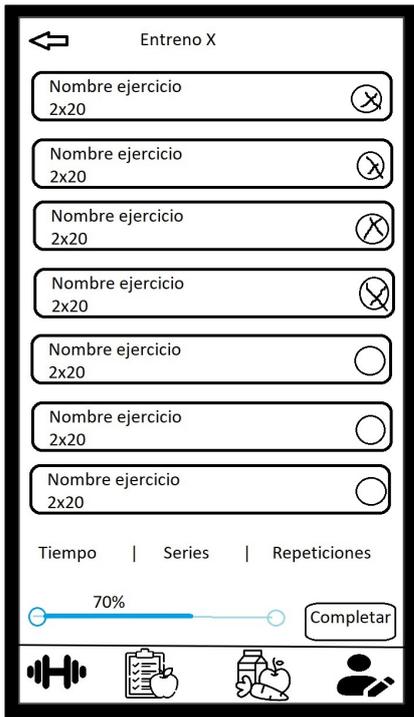


Figura 16: Vista de Entreno por Día.

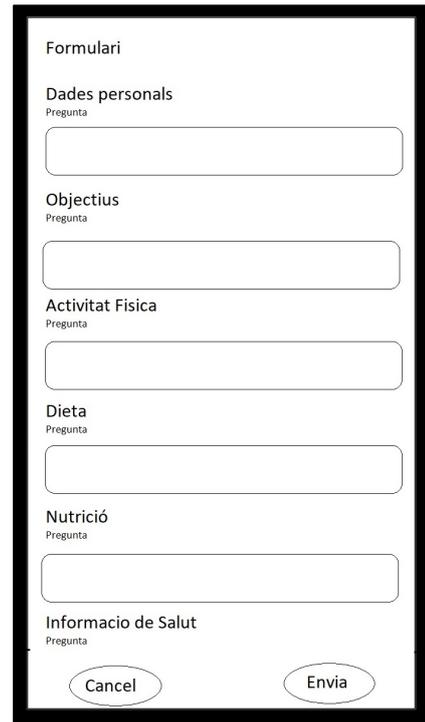


Figura 17: Formulario inicial de registro.

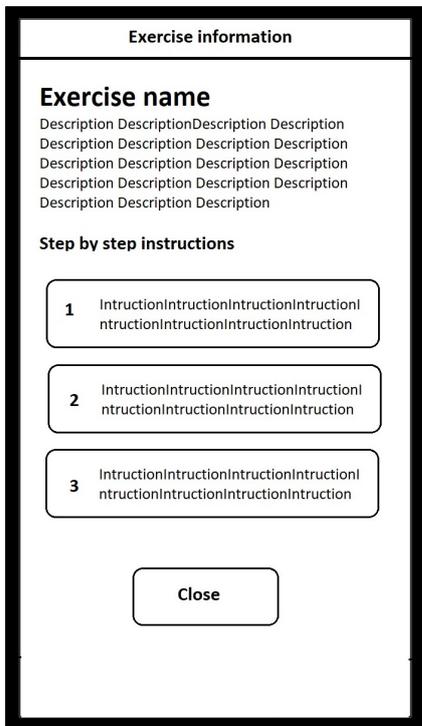


Figura 18: Vista de Información del Ejercicio.

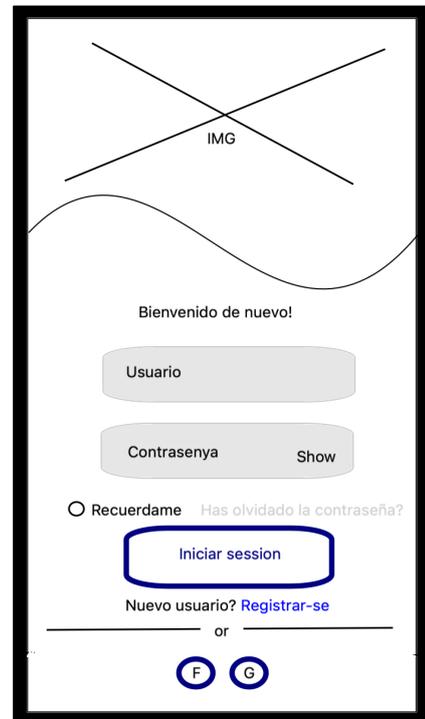


Figura 19: Pantalla de Inicio de Sesión.

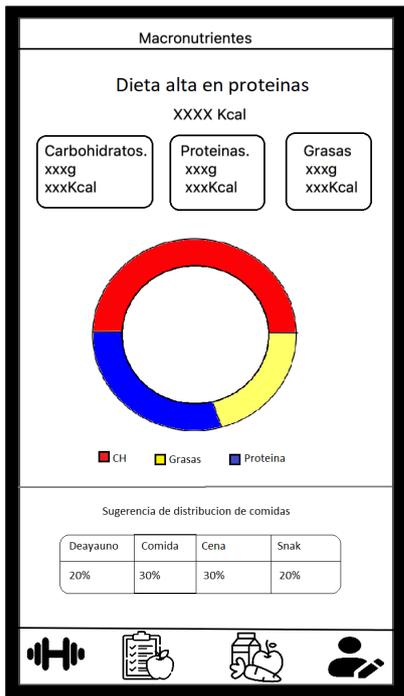


Figura 20: Pantalla de Macro-nutrientes.

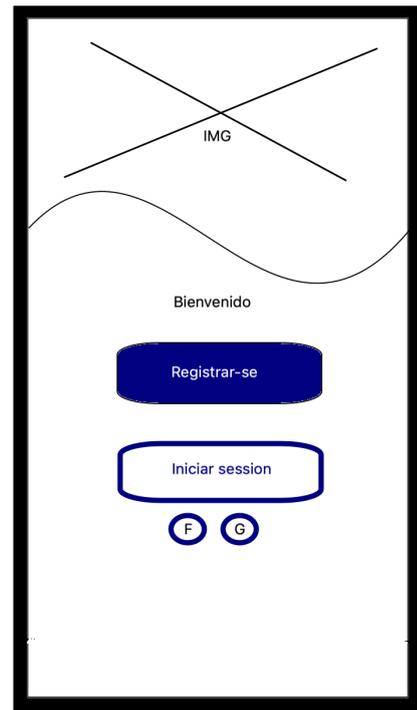


Figura 21: Vista principal de la aplicación.

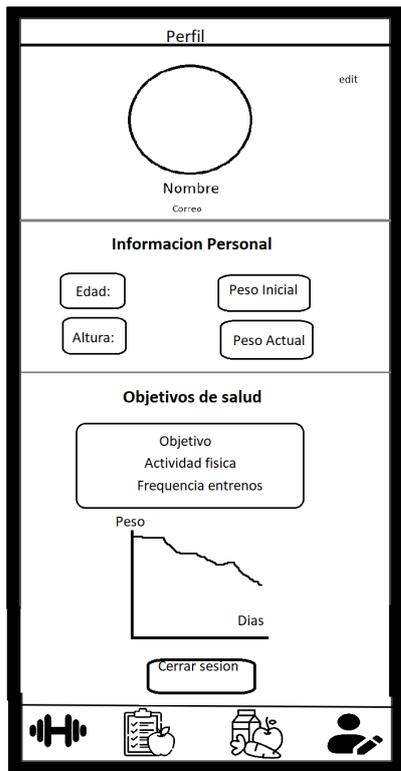


Figura 22: Pantalla de Perfil.

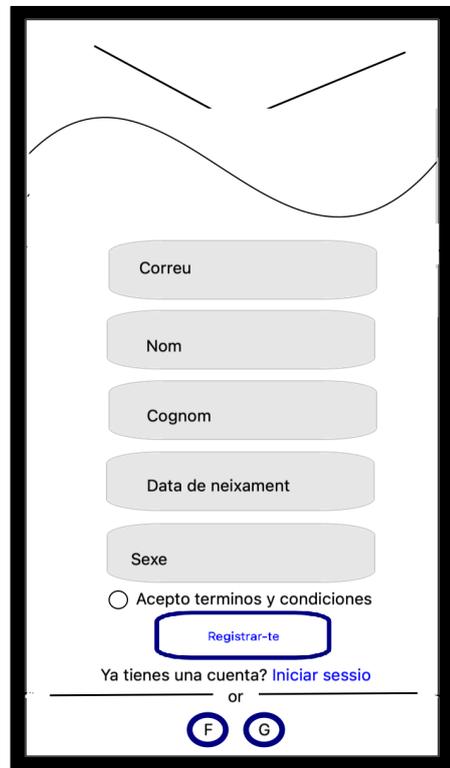


Figura 23: Pantalla de Registro.

Esquemas de Endpoints

En esta sección del anexo se presentan los esquemas de los endpoints que fueron diseñados antes de comenzar con la implementación del sistema. Estos esquemas no representan los endpoints finales del proyecto, sino que sirvieron como una base inicial para guiar el proceso de programación. La creación de estos endpoints preliminares me permitió estructurar de manera más clara las funcionalidades que el sistema debía implementar y facilitó la organización del código a medida que avanzaba en el desarrollo. Aunque algunos de estos esquemas pudieron sufrir cambios durante la implementación, son una representación útil del pensamiento estructural previo a la codificación.

POST Endpoints

- **POST /register**

Descripción: Registrar un nuevo usuario.

Parámetros (body):

- **firstName:** string
- **lastName:** string
- **email:** string
- **password:** string
- **birthDate:** date
- **gender:** string ("M", "F", "Otro")
- **termsAccepted:** boolean

- **POST /login**

Descripción: Autenticación de usuario.

Parámetros (body):

- **email:** string
- **password:** string
- **rememberMe:** boolean

- **POST /forgot-password**

Descripción: Solicitar recuperación de contraseña.

Parámetros (body):

- **email:** string

- **POST /refresh-token**

Descripción: Refrescar el token de autenticación (si rememberMe está activado).

Parámetros (headers o body):

- **refreshToken:** string

- **POST /user-details**
Descripción: Guardar los detalles del usuario.
Parámetros (body):
 - **height:** string
 - **weight:** string
 - **weightChangeAmount:** string (opcional)
 - **weeklyTrainingDays:** number
 - **dailyTrainingTime:** string
 - **weightGoal:** string
 - **physicalActivityLevel:** string
 - **currentTrainingDays:** number
 - **dietType:** string
 - **mealsPerDay:** number
 - **macronutrientIntake:** string
 - **availableEquipment:** string
 - **trainingPreference:** string
 - **foodRestrictions:** string
 - **customMedicalCondition:** string (opcional)
 - **medicalCondition:** string
 - **customFoodRestrictions:** string (opcional)

- **POST /profile/logout**
Descripción: Cerrar sesión del usuario y limpiar cualquier token de autenticación.
Parámetros (headers o body):
 - **token:** string

- **POST /workout/day/{dayId}/complete**
Descripción: Marcar un día de entrenamiento como completado.
Parámetros (body):
 - **completedExercises:** array (con el nombre y estado de completado de cada ejercicio)

- **POST /users**
Descripción: Este endpoint permite añadir un nuevo usuario al sistema.
Parámetros (body):
 - **firstName:** string (Nombre del usuario)
 - **lastName:** string (Apellido del usuario)
 - **email:** string (Correo electrónico del usuario)

- **password:** string (Contraseña del usuario)
 - **birthDate:** string (Fecha de nacimiento en formato ISO)
 - **gender:** string (Género del usuario: 'M', 'F', 'Otro')
- **POST /macros**
Descripción: Este endpoint permite añadir una nueva recomendación de macronutrientes al sistema.
Parámetros (body):
 - **kcal:** number (Cantidad de calorías recomendadas)
 - **proteins:** number (Cantidad de proteínas recomendadas en gramos)
 - **carbs:** number (Cantidad de carbohidratos recomendados en gramos)
 - **fats:** number (Cantidad de grasas recomendadas en gramos)
 - **dietType:** string (Tipo de dieta: 'weightLoss', 'muscleGain', 'maintenance')
 - **description:** string, opcional (Descripción adicional sobre la dieta, por ejemplo, "Dieta alta en proteínas, baja en carbohidratos")
- **POST /workouts**
Descripción: Este endpoint permite crear un nuevo entrenamiento en el sistema.
Parámetros (body):
 - **name:** string (Nombre del entrenamiento)
 - **description:** string (Descripción del entrenamiento)
 - **exercises:** array (Lista de ejercicios asociados al entrenamiento, incluyendo el nombre, series, repeticiones y descanso)
 - **media:** string, opcional (URL o base64 de la imagen o video asociado al entrenamiento)
- **POST /exercises**
Descripción: Este endpoint permite crear un nuevo ejercicio en el sistema, incluyendo los grupos musculares trabajados, instrucciones y medios asociados (imagen o video).
Parámetros (body):
 - **name:** string (Nombre del ejercicio)
 - **description:** string (Descripción del ejercicio)
 - **muscleGroups:** array (Lista de grupos musculares que trabaja el ejercicio, por ejemplo, ["Biceps", "Legs"])
 - **instructions:** string (Instrucciones para realizar el ejercicio)
 - **media:** string, opcional (URL o base64 de la imagen o video del ejercicio)

GET Endpoints

- **GET /email-exists**
Descripción: Verificar si un correo ya está registrado.
Parámetros (query):
 - **email:** string
- **GET /workouts**
Descripción: Obtener la lista de entrenamientos disponibles.
Parámetros (query o headers, opcional):
 - **userId:** string (para entrenamientos personalizados)
- **GET /workout/{day}**
Descripción: Obtener los detalles específicos del entrenamiento para un día en particular.
Parámetros (path):
 - **day:** string (número del día seleccionado)
- **GET /workout/day/{dayId}**
Descripción: Obtener los detalles de los ejercicios para un día de entrenamiento específico.
Parámetros (path):
 - **dayId:** string
- **GET /exercise/{exerciseId}**
Descripción: Obtener la información detallada de un ejercicio específico.
Parámetros (path):
 - **exerciseId:** string
- **GET /exercise/{exerciseId}/steps**
Descripción: Obtener las instrucciones paso a paso para realizar el ejercicio.
Parámetros (path):
 - **exerciseId:** string
- **GET /progress/control-panel**
Descripción: Obtener los datos del panel de control, incluyendo el calendario y el próximo entrenamiento.
Parámetros (query o headers, opcional):
 - **userId:** string
- **GET /workout/history**
Descripción: Obtener el historial de entrenamientos del usuario, con el estado de cada uno (completado o bloqueado).
Parámetros (query o headers, opcional):

- **userId:** string
- **GET /workout/{workoutName}**
Descripción: Obtener los detalles de un entrenamiento específico cuando el usuario selecciona un entrenamiento completado.
Parámetros (path):
 - **workoutName:** string
- **GET /workout/completed-count**
Descripción: Obtener el número total de entrenamientos completados por el usuario.
Parámetros (query o headers, opcional):
 - **userId:** string
- **GET /progress/graphics**
Descripción: Obtener las gráficas de progreso, como peso y otras métricas.
Parámetros (query o headers, opcional):
 - **userId:** string
- **GET /progress/weight**
Descripción: Obtener el progreso de peso actual y la meta del usuario.
Parámetros (query o headers, opcional):
 - **userId:** string
- **GET /workout/next**
Descripción: Obtener la información del próximo entrenamiento (nombre, progreso, y días restantes).
Parámetros (query o headers, opcional):
 - **userId:** string
- **GET /workout/schedule**
Descripción: Obtener el calendario de días de entrenamiento del usuario.
Parámetros (query o headers, opcional):
 - **userId:** string
- **GET /profile**
Descripción: Obtener los datos del perfil del usuario, como nombre, correo, edad, peso actual, y metas de salud.
Parámetros (query o headers, opcional):
 - **userId:** string
- **GET /profile/weight-history**
Descripción: Obtener el historial de peso del usuario a lo largo del tiempo.
Parámetros (query o headers, opcional):

- **userId**: string
- **GET /macronutrients**
Descripción: Obtener los datos de macronutrientes del usuario, incluidos carbohidratos, proteínas, grasas y el total de calorías.
Parámetros (query o headers, opcional):
 - **userId**: string
- **GET /meal-distribution**
Descripción: Obtener una sugerencia para la distribución de las comidas del día en porcentajes.
Parámetros (query o headers, opcional):
 - **userId**: string
- **GET /users**
Descripción: Obtener la lista de usuarios con su información básica como nombre y correo.
- **GET /macros/{id}**
Descripción: Obtener los detalles de una recomendación específica de macronutrientes.
Parámetros (path):
 - **id**: number (El ID de la recomendación de macronutrientes que se desea modificar)
- **GET /statistics/exercises**
Descripción: Este endpoint devuelve la popularidad de los ejercicios más realizados en la plataforma.
Parámetros (query):
 - **year**: string, opcional (Año para filtrar los datos, por defecto el año actual)
- **GET /workouts**
Descripción: Este endpoint devuelve una lista de todos los entrenamientos disponibles en el sistema.
Parámetros (query):
 - **limit**: number, opcional (Limitar el número de entrenamientos devueltos)
 - **offset**: number, opcional (Desplazamiento para paginación)
- **GET /exercises**
Descripción: Este endpoint devuelve una lista de todos los ejercicios disponibles en el sistema.
Parámetros (query):

- **limit**: number, opcional (Limitar el número de ejercicios devueltos)
- **offset**: number, opcional (Desplazamiento para paginación)

PUT Endpoints

- **PUT /workout/exercise/{exerciseId}/complete**
Descripción: Marcar un ejercicio como completado o incompleto.
Parámetros (body):
 - **completed**: boolean**Parámetros (path):**
 - **exerciseId**: string
- **PUT /profile**
Descripción: Actualizar la información del perfil del usuario, como nombre, apellido, peso actual, meta de peso, nivel de actividad, y frecuencia de entrenamiento.
Parámetros (body):
 - **firstName**: string
 - **lastName**: string
 - **currentWeight**: number
 - **weightGoal**: string
 - **activityLevel**: string
 - **trainingFrequency**: number
- **PUT /profile/password**
Descripción: Cambiar la contraseña del usuario.
Parámetros (body):
 - **currentPassword**: string
 - **newPassword**: string
 - **confirmPassword**: string
- **PUT /profile/photo**
Descripción: Subir o actualizar la foto de perfil del usuario.
Parámetros (body):
 - **photo**: file
- **PUT /users/**
Descripción: Este endpoint permite modificar la información de un usuario existente en el sistema.
Parámetros (URL):

- **id**: number (El ID del usuario que se está modificando)

Parámetros (body):

- **firstName**: string (Nombre del usuario)
- **lastName**: string (Apellido del usuario)
- **currentWeight**: number (Peso actual del usuario en kilogramos)
- **weightGoal**: string (Objetivo de peso del usuario, ej. 'Perder peso', 'Ganar masa muscular')
- **activityLevel**: string (Nivel de actividad física, ej. 'Sedentario', 'Moderada')
- **trainingFrequency**: number (Días de entrenamiento por semana)

■ **PUT /macros/**

Descripción: Este endpoint permite modificar una recomendación de macronutrientes existente.

Parámetros (path):

- **id**: number (El ID de la recomendación que se desea modificar)

Cuerpo de la solicitud (body):

- **kcal**: 2500
- **proteins**: 160
- **carbs**: 320
- **fats**: 75
- **dietType**: "muscleGain"
- **description**: "Dieta ajustada para mayor ganancia muscular"

■ **PUT /workouts/**

Descripción: Este endpoint permite modificar los detalles de un entrenamiento existente, incluyendo el nombre, la descripción, los ejercicios y los medios (imagen/video).

Parámetros (body):

- **name**: string (Nuevo nombre del entrenamiento)
- **description**: string (Nueva descripción del entrenamiento)
- **exercises**: array (Lista de ejercicios asociados al entrenamiento, incluyendo el nombre, series, repeticiones y descanso)
- **media**: string, opcional (URL o base64 del nuevo archivo multimedia)

■ **PUT /exercises/**

Descripción: Este endpoint permite modificar los detalles de un ejercicio existente, incluyendo el nombre, la descripción, los grupos musculares, las instrucciones y los medios (imagen/video).

Parámetros (body):

- **name:** string (Nuevo nombre del ejercicio)
- **description:** string (Nueva descripción del ejercicio)
- **muscleGroups:** array (Lista de grupos musculares asociados al ejercicio)
- **instructions:** string (Nuevas instrucciones del ejercicio)
- **media:** string (URL o base64 del nuevo archivo multimedia)

Guiones de entrevistas de usuario

En este anexo se presentan los guiones utilizados en las entrevistas guiadas con los diferentes tipos de usuarios. Estos guiones proporcionan una estructura detallada para las entrevistas, incluyendo preguntas y flujos de tareas que ayudan a evaluar la usabilidad y funcionalidad de la aplicación desde la perspectiva de cada perfil de usuario.

Se utilizaron dos enfoques en las entrevistas guiadas:

- **Entrevistas Guiadas con Flujos Libres:** Dirigidas a captar impresiones generales y la facilidad de navegación sin imponer un camino específico. Cada entrevista se adaptó a los roles de entrenador, nutricionista y cliente.
- **Entrevistas Guiadas con Flujos Específicos:** Centradas en guiar al usuario a través de tareas clave dentro de la aplicación, evaluando su experiencia en procesos como asignación de planes, gestión de entrenamientos y personalización de perfil.

A continuación, se presentan los guiones de entrevista para cada perfil de usuario:

Entrevista Guiada con Flujos Libres para Entrenador

Primera impresión

- ¿Qué opinas de la interfaz al abrir la aplicación por primera vez?
- ¿Es fácil encontrar las secciones que podrías necesitar como entrenador?

Navegación General

- Si quisieras gestionar los entrenamientos, ¿dónde crees que podrías acceder a esa sección?
- ¿Cómo accederías a la lista de clientes o usuarios para asignarles un plan de entrenamiento?

Usabilidad de Entrenamientos

- ¿Qué tan fácil te resulta encontrar opciones para añadir o modificar entrenamientos?
- Si quisieras ver el detalle de un ejercicio dentro de un entrenamiento, ¿cómo procederías?

Perfil y Configuración

- ¿Dónde ajustarías tu perfil o cambiarías tu contraseña?
- ¿Percibes alguna dificultad al encontrar opciones de configuración personal?

Estadísticas y Seguimiento

- Si quisieras revisar estadísticas o progresos de los clientes, ¿dónde esperas encontrar esa información?

Comentarios Generales

- ¿Hay algo en la navegación que te haya resultado confuso?
- ¿Tienes alguna sugerencia sobre cómo mejorar la experiencia de navegación para entrenadores?

Entrevista Guiada con Flujos Libres para Nutricionista

Primera impresión

- ¿Qué te parece la organización de la aplicación? ¿Encuentras intuitiva la disposición de las secciones?

Acceso a Nutrición

- ¿Cómo accederías a la sección donde puedes gestionar o modificar los planes de nutrición?
- Si quisieras añadir un nuevo plan nutricional, ¿cómo procederías?

Gestión de Usuarios

- ¿Dónde buscarías la lista de usuarios para asignarles planes de nutrición?
- ¿Es claro para ti cómo podrías modificar un plan nutricional existente?

Perfil y Configuración

- ¿Cómo accederías a tu perfil o a las opciones para cambiar la configuración de la cuenta?
- ¿Crees que es fácil acceder a la información de configuración o perfil?

Navegación General

- ¿Te sientes cómodo navegando entre las diferentes pantallas de la aplicación?
- ¿Encontraste alguna sección donde te hayas sentido desorientado/a?

Comentarios Generales

- ¿Qué mejorarías de la aplicación para hacerla más intuitiva para nutricionistas?
- ¿Hay alguna funcionalidad que sientas que debería estar más accesible?

Entrevista Guiada con Flujos Libres para Cliente

Primera impresión

- ¿Qué opinas de la apariencia de la aplicación? ¿Te resulta amigable y fácil de entender?

Acceso a Entrenamientos

- Si quisieras ver los entrenamientos que tienes asignados, ¿dónde crees que deberías ir?
- ¿Te resulta fácil identificar dónde están los detalles de cada entrenamiento?

Acceso a Plan de Nutrición

- ¿Dónde buscarías tu plan de macronutrientes o cualquier plan de nutrición que te hayan asignado?
- ¿Es fácil acceder a la información de nutrición o te parece que está algo oculta?

Perfil y Configuración

- ¿Cómo editarías tu información personal, como nombre o preferencias?
- ¿Dónde cambiarías tu contraseña si fuera necesario?

Navegación General

- ¿Te sientes cómodo/a moviéndote por la aplicación? ¿Es claro cómo acceder a las diferentes secciones?
- ¿Hubo algún momento en que te hayas sentido confundido/a sobre qué hacer o a dónde ir?

Comentarios Generales

- ¿Qué te gustaría mejorar de la aplicación en cuanto a la navegación?
- ¿Sientes que la aplicación cubre tus necesidades como usuario?

Entrevista Guiada con Flujos Específicos para Entrenador

Flujo 1: Acceso y gestión de entrenamientos

1. Inicia sesión en la aplicación con tus credenciales de entrenador.
2. Dirígete a la sección de *Entrenamientos*.
3. ¿Cómo añadirías un nuevo entrenamiento para tus clientes? Describe el proceso.
4. ¿Qué pasos sigues para modificar un entrenamiento existente?

Flujo 2: Asignación de planes a clientes

1. Accede a la sección de *Usuarios*.
2. Selecciona un cliente al que desees asignarle un plan de entrenamiento.
3. Describe cómo realizarías la asignación de un plan. ¿Te resulta intuitivo el proceso?
4. Si el cliente ya tiene un plan asignado, ¿cómo procederías para modificarlo o actualizarlo?

Flujo 3: Acceso y análisis de estadísticas

1. Ve a la sección de *Estadísticas*.
2. ¿Cómo utilizarías esta sección para revisar el progreso de tus clientes?
3. Describe cualquier funcionalidad que te haya resultado útil o que consideres que debería mejorarse.

Comentarios Generales

- ¿Qué mejorarías en el flujo de navegación para entrenadores?
- ¿Sientes que la aplicación cubre todas las necesidades de un entrenador?

Entrevista Guiada con Flujos Específicos para Nutricionista

Flujo 1: Gestión de planes de nutrición

1. Inicia sesión en la aplicación con tu cuenta de nutricionista.
2. Ve a la sección de *Nutrición*.
3. ¿Cómo añadirías un nuevo plan de nutrición? Explica los pasos que sigues.
4. Describe cómo modificarías un plan de nutrición ya existente.

Flujo 2: Asignación de planes nutricionales a usuarios

1. Accede a la sección de *Usuarios*.
2. Selecciona un cliente al que quieras asignarle un plan de nutrición.
3. Describe el proceso para asignar un plan de nutrición. ¿Es fácil de seguir?
4. ¿Consideras que el flujo de asignación es claro y funcional?

Flujo 3: Verificación de la información de nutrición

1. Dirígete a un perfil de cliente que tenga un plan de nutrición asignado.
2. ¿Cómo visualizarías los detalles del plan de nutrición del cliente?
3. ¿Es fácil acceder y comprender la información nutricional?

Comentarios Generales

- ¿Qué mejorarías en los flujos de navegación para nutricionistas?
- ¿La aplicación facilita la gestión de la nutrición para los usuarios?

Entrevista Guiada con Flujos Específicos para Cliente

Flujo 1: Acceso a entrenamientos asignados

1. Inicia sesión en la aplicación como cliente.
2. Ve a la *Pantalla de entrenamientos*.
3. ¿Cómo visualizarías el detalle de un entrenamiento específico?
4. Describe cualquier aspecto que te haya resultado confuso o poco claro en este flujo.

Flujo 2: Acceso a plan de nutrición

1. Accede a la sección de *Macronutrientes* o *Nutrición*.
2. ¿Cómo visualizarías los detalles de tu plan de nutrición?
3. Describe si el proceso para acceder a esta información es intuitivo.

Flujo 3: Edición de perfil y configuración de cuenta

1. Ve a tu *Perfil de usuario*.
2. Intenta cambiar algún dato personal, como el nombre o la contraseña.
3. Describe cualquier dificultad que hayas encontrado al hacer estas modificaciones.

Comentarios Generales

- ¿Sientes que la aplicación cubre tus necesidades como cliente?
- ¿Qué mejorarías en los flujos de navegación para hacerlos más intuitivos?

Referencias

- [1] R. S. Tate, *Software Project Management: A Concise Study*. Pearson Education, 2019.
- [2] K. Schwaber and J. Sutherland, *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org, 2020.
- [3] Scrum.org, “Retrospective meeting guide.” <https://www.scrum.org/resources/what-is-a-sprint-retrospective>, 2023.
- [4] MindTools, “The history of gantt charts.” https://www.mindtools.com/pages/article/newPPM_03.htm, 2023.
- [5] ProjectManager, “Advantages and disadvantages of gantt charts.” <https://www.projectmanager.com/blog/gantt-chart-advantages-disadvantages>, 2023.
- [6] A. Cockburn, *Writing Effective Use Cases*. Addison-Wesley, 2001.
- [7] M. Cohn, *User Stories Applied: For Agile Software Development*. Addison-Wesley, 2004.
- [8] S. Chacon and B. Straub, *Pro Git*. Apress, 2nd ed., 2014.
- [9] G. Documentation, “Understanding git branching and merging.” <https://docs.github.com/en/get-started/using-git/about-branches>, 2023.
- [10] Scrum.org, “Embracing change: The importance of flexibility in agile.” <https://www.scrum.org/resources/blog/embracing-change-importance-flexibility-agile>, 2023.
- [11] S. Krug, *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*. New Riders, 2014.
- [12] P. Morville and L. Rosenfeld, *Information Architecture for the World Wide Web: For the Web and Beyond*. O'Reilly Media, 2015.
- [13] J. Nielsen, “Usability 101: Introduction to usability.” <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>, 2023.
- [14] U. D. Patterns, “Best practices for user navigation in mobile apps.” <https://uxdesignn.cc/designing-effective-navigation-for-mobile-apps-8f8d11de58>, 2023.
- [15] P. Library, “Navigation patterns for web applications.” <https://patternlibrary.com/navigation-patterns>, 2023.
- [16] T. Connolly and C. Begg, *Database Systems: A Practical Approach to Design, Implementation, and Management*. Pearson Education, 2015.

- [17] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*. Addison-Wesley, 2011.
- [18] WHATWG, “Html living standard,” 2024.
- [19] W3C, “Css: Cascading style sheets,” 2024.
- [20] M. W. Docs, “Javascript guide,” 2024.
- [21] M. (Facebook), “React documentation,” 2024.
- [22] R. Team, “Optimizing performance in react,” 2024.
- [23] R. Team, “State management and data flow in react,” 2024.
- [24] I. Framework, “Build multiplatform apps with ionic,” 2024.
- [25] I. Framework, “Ui components for ionic,” 2024.
- [26] I. Framework, “Using react with ionic,” 2024.
- [27] I. Framework, “Capacitor: Access native features with ionic,” 2024.
- [28] I. Framework, “Progressive web apps with ionic,” 2024.
- [29] P. S. Foundation, “Python documentation.” <https://docs.python.org/3/>, 2023.
- [30] T. Peters, “The zen of python.” <https://peps.python.org/pep-0020/>, 2023.
- [31] P. S. Foundation, “Python standard library.” <https://docs.python.org/3/library/>, 2023.
- [32] P. Community, “Python community support.” <https://www.python.org/community/>, 2023.
- [33] D. S. Foundation, “Django documentation.” <https://docs.djangoproject.com/en/stable/>, 2023.
- [34] D. S. Foundation, “Quick start with django.” <https://docs.djangoproject.com/en/stable/intro/>, 2023.
- [35] D. S. Foundation, “Django features - batteries included.” <https://docs.djangoproject.com/en/stable/misc/design-philosophies/#batteries-included>, 2023.
- [36] D. S. Foundation, “Django authentication system.” <https://docs.djangoproject.com/en/stable/topics/auth/>, 2023.
- [37] D. S. Foundation, “Django security features.” <https://docs.djangoproject.com/en/stable/topics/security/>, 2023.

- [38] D. R. Framework, “Drf - django rest framework documentation.” <https://www.django-rest-framework.org/>, 2023.
- [39] E. OSS, “Handling http requests with django rest framework.” <https://www.django-rest-framework.org/api-guide/views/>, 2024.
- [40] F. Documentation, “Firebase platform overview.” <https://firebase.google.com/docs>, 2023.
- [41] P. G. D. Group, “About postgresql: The world’s most advanced open source relational database.” <https://www.postgresql.org/about/>, 2024.
- [42] F. Documentation, “Firebase features and overview.” <https://firebase.google.com/>, 2023.
- [43] G. Firebase, “Differences between nosql and sql databases.” <https://firebase.google.com/docs/firestore/query-data/structure-data>, 2024.
- [44] G. Firebase, “Limitations of firestore with sql queries.” <https://firebase.google.com/docs/firestore/query-data/limitations>, 2024.
- [45] F. Developers, “Considerations for firebase scalability and lock-in.” <https://firebase.google.com/docs/resources/scaling>, 2024.
- [46] P. G. D. Group, “Relational model in postgresql.” <https://www.postgresql.org/docs/current/intro-what-is.html>, 2024.
- [47] P. G. D. Group, “Advanced sql features in postgresql.” <https://www.postgresql.org/docs/current/tutorial-advanced.html>, 2024.
- [48] P. G. D. Group, “Acid compliance in postgresql.” <https://www.postgresql.org/docs/current/transaction-iso.html>, 2024.
- [49] P. G. D. Group, “Extensions in postgresql.” <https://www.postgresql.org/docs/current/external-extensions.html>, 2024.
- [50] P. G. D. Group, “Scalability in postgresql.” <https://www.postgresql.org/docs/current/high-availability.html>, 2024.
- [51] P. G. D. Group, “Json support in postgresql.” <https://www.postgresql.org/docs/current/functions-json.html>, 2024.
- [52] P. G. D. Group, “Tools compatible with postgresql.” <https://www.postgresql.org/download/products/3rd-party-tools/>, 2024.
- [53] G. Cloud, “Google cloud storage overview.” <https://cloud.google.com/storage/docs/overview>, 2024.
- [54] G. Cloud, “Google cloud storage scalability.” <https://cloud.google.com/storage>, 2024.

- [55] G. Cloud, “Signed urls for google cloud storage.” <https://cloud.google.com/storage/docs/access-control/signed-urls>, 2024.
- [56] G. Cloud, “Durability and availability in google cloud storage.” <https://cloud.google.com/storage/docs/reliability>, 2024.
- [57] G. Cloud, “Integrating google cloud storage with databases.” <https://cloud.google.com/solutions/architecture>, 2024.