



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Desarrollo de una aplicación web para la creación y
seguimiento de rutinas de entrenamiento

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Martínez Vargas, Javier

Tutor/a: Valderas Aranda, Pedro José

CURSO ACADÉMICO: 2023/2024

Resumen

En un mundo donde la inmediatez y la personalización son la norma, el entrenamiento físico no escapa a esta tendencia. Este trabajo presenta el desarrollo de una aplicación web innovadora para la creación de rutinas de gimnasio personalizadas. El principal objetivo de esta aplicación es proporcionar a los usuarios una herramienta que les permita diseñar programas de entrenamiento adaptados a sus objetivos individuales.

Para lograr este objetivo, se ha seguido una Metodología de Diseño Centrado en el Usuario (DCU), comenzando con la creación de una persona modelo basada en una investigación cualitativa. A partir de esta base, se han desarrollado escenarios específicos que guían la interacción del usuario con la aplicación. La arquitectura de la aplicación se ha estructurado en tres capas: una capa de datos soportada por *PostgreSQL*, una capa de lógica de negocio implementada con *Java SpringBoot*, y una capa de presentación desarrollada utilizando *AstroJS*, *React* y *Tailwind CSS*.

La aplicación permite a los usuarios no solo definir sus rutinas de entrenamiento en función de sus metas, sino también ajustar estos planes según el grupo muscular, nivel de experiencia y tiempo disponible. Adicionalmente, la aplicación ofrece información detallada sobre la ejecución correcta de los ejercicios y sus beneficios, así como herramientas para registrar y monitorizar el progreso.

Los resultados obtenidos de este desarrollo muestran una herramienta robusta y versátil que facilita la creación de rutinas de entrenamiento personalizadas, mejorando la experiencia del usuario y optimizando su rendimiento en el gimnasio.

Palabras clave: Gimnasio, entrenamiento, SpringBoot, Java, AstroJS, PostgreSQL, TailwindCSS

Resum

En un món on la immediatesa i la personalització són la norma, l'entrenament físic no escapa a esta tendència. Este treball presenta el desenrotllament d'una aplicació web innovadora per a la creació de rutines de gimnàs personalitzades. El principal objectiu d'esta aplicació és proporcionar als usuaris una ferramenta que els permeta dissenyar programes d'entrenament adaptats als seus objectius individuals.

Per a aconseguir este objectiu, s'ha seguit una Metodologia de Disseny Centrat en l'Usuari (DCU), començant amb la creació d'una persona model basada en una investigació qualitativa. A partir d'esta base, s'han desenrotllat escenaris específics que guien la interacció de l'usuari amb l'aplicació. L'arquitectura de l'aplicació s'ha estructurat en tres capes: una capa de dades suportada per *PostgreSQL*, una capa de lògica de negoci implementada amb *Java SpringBoot*, i una capa de presentació desenrotllada utilitzant *AstroJS*, *React* i *Tailwind CSS*.

L'aplicació permet als usuaris no sols definir les seues rutines d'entrenament en funció de les seues metes, sinó també ajustar estos plans segons el grup muscular, nivell d'experiència i temps disponible. Adicionalment, l'aplicació ofereix informació detallada sobre l'execució correcta dels exercicis i els seus beneficis, així com ferramentes per a registrar i monitorar el progrés.

Els resultats obtinguts d'este desenrotllament mostren una ferramenta robusta i versàtil que facilita la creació de rutines d'entrenament personalitzades, millorant l'experiència de l'usuari i optimitzant el seu rendiment en el gimnàs.

Paraules clau: Gimnàs, entrenament, SpringBoot, Java, AstroJS, PostgreSQL, TailwindCSS

Abstract

In a world where immediacy and personalisation are the norm, fitness training does not escape this trend. This paper presents the development of an innovative web application for the creation of personalised gym routines. The main objective of this application is to provide users with a tool that allows them to design training programmes adapted to their individual goals.

To achieve this goal, a User-Centred Design Methodology (UCD) has been followed, starting with the creation of a model persona based on qualitative research. From this base, specific scenarios have been developed to guide the user's interaction with the application. The application architecture is structured in three layers: a data layer supported by *PostgreSQL*, a business logic layer implemented with *Java SpringBoot*, and a presentation layer developed using *AstroJS*, *React* and *Tailwind CSS*.

The application allows users to not only define their training routines based on their goals, but also to adjust these plans according to muscle group, experience level and time available. In addition, the application provides detailed information on the correct execution of the exercises and their benefits, as well as tools to record and monitor progress.

The results obtained from this development show a robust and versatile tool that facilitates the creation of personalised training routines, improving the user experience and optimising their performance in the gym.

Key words: Gym, training, SpringBoot, Java, AstroJS, PostgreSQL, TailwindCSS

Índice general

Índice general	V
Índice de figuras	VII

1	Introducción	1
1.1	Motivación	3
1.2	Objetivos	3
1.3	Estructura de la memoria	3
2	Estado del arte	5
2.1	Análisis del estado del arte	5
2.1.1	Entrenamientos.com	5
2.1.2	Cambiatufisico.com	6
2.1.3	Fitbod	8
2.1.4	Bajar de Peso hombre	10
2.2	Crítica al Estado del Arte	12
2.3	Propuesta	12
3	Metodología	13
3.1	DCU. Diseño Centrado en el Usuario	13
3.2	Kanban	14
4	Análisis de necesidades	17
4.1	Investigación cualitativa. Recogida de datos	17
4.1.1	Cuestionario y estudios de documentación	17
4.1.2	Datos obtenidos y análisis	19
4.2	Definición de Persona	20
4.2.1	Justificación de la persona con los datos obtenidos	21
4.3	Escenarios	23
4.3.1	Justificación de los escenarios con los datos de la persona	24
5	Análisis Conceptual y Diseño	27
5.1	Diagrama de clases	27
5.2	Modelo de la BD que use el servidor	28
5.3	Bocetos de las interfaces de la app	30
6	Desarrollo de la solución	43
6.1	Arquitectura	43
6.2	Contexto tecnológico	44
6.2.1	IntelliJ Idea	45
6.2.2	Visual Studio Code	45
6.2.3	Google Cloud Buckets	45
6.2.4	Java Spring, SpringBoot	46
6.2.5	Spring Security	47
6.2.6	Spring JPA y PostgreSQL	47
6.2.7	Swagger y SpringDoc	47
6.2.8	Spring Boot Test y SonarLint	47
6.2.9	React	48

6.2.10	AstroJS	48
6.2.11	TailwindCSS	48
6.3	Ejemplos de código	49
6.3.1	Implementación de JWT para la Autenticación	49
6.3.2	Seguridad de las APIs con Spring Security	49
6.3.3	Flujo de Trabajo: Controlador, Servicio, Mapper y Repositorio	50
6.3.4	Implementación de Tests Unitarios	53
6.3.5	Swagger	56
6.3.6	Componente de Astro	57
6.3.7	Ejemplo de uso de TailwindCSS	59
6.3.8	Ejemplo de un componente de React	59
7	Producto desarrollado	63
8	Validación	71
8.1	Evaluación Heurística	71
9	Conclusiones y trabajos futuros	77
	Bibliografía	79
	Glosario	81
	Anexos	83
A	Cuestionarios detallados	83
B	ODS	89

Índice de figuras

1.1	Los 3 pilares de la salud	1
1.2	Porcentaje de individuos que fue a un gimnasio en España de 1996 a 2023. Fuente: Statista	2
2.1	Pantalla inicial de la página web entrenamientos.com	5
2.2	Pantalla inicial de la página web cambiatufisico.com	6
2.3	Pantalla inicial de la página web cambiatufisico.com desde una interfaz móvil	7
2.4	Pantalla inicial de la aplicación Fitbod	8
2.5	Restricción de entrenamientos en la aplicación Fitbod	9
2.6	Planes de precio de la aplicación Fitbod	9
2.7	Pantalla inicial de la aplicación Bajar de Peso hombre	10
2.8	Anuncios pop-ups en la aplicación Bajar de Peso hombre	11
3.1	Etapas del diseño centrado en el usuario, Seba Gritti, Medium	14
3.2	Ilustración de un tablero Kanban	14
3.3	Tablero trello de este proyecto	15
4.1	Ficha de Esteban Duro	21
5.1	Diagrama de clases	28
5.2	Diagrama Entidad-Relación en 3FN	29
5.3	Pantalla inicial en PC	30
5.4	Pantalla inicial en Móvil	31
5.5	Pantalla general de ejercicios en PC	31
5.6	Pantalla lista de ejercicios y ejercicio concreto en Móvil	32
5.7	Pantalla de un ejercicio en PC	33
5.8	Pantalla general de rutinas en PC	34
5.9	Pantalla general de rutinas en Móvil	34
5.10	Pantalla de rutina concreta en PC	35
5.11	Pantalla de rutina concreta en Móvil	35
5.12	Pantalla de mis rutinas en PC	36
5.13	Pantalla de mis rutinas y registrar rutina en Móvil	37
5.14	Pantalla de registrar rutina en PC	38
5.15	Pantalla de crear rutina en PC	39
5.16	Pantalla de crear rutina en Móvil	39
5.17	Pantalla de rutina concreta con comentarios en PC	40
5.18	Pantalla de mis estadísticas en PC	41
5.19	Pantalla de mis estadísticas en Móvil	41
6.1	Arquitectura de SiuGym	43
6.2	Estructura de carpetas en el Bucket de Google	45
6.3	Detalle de una imagen en el Bucket de Google	46
6.4	Tabla exercise con las URLs a las imagenes guardadas en el Bucket	46

6.5	Herramientas utilizadas en el desarrollo del proyecto	48
6.6	Código más relevante de la clase JwtUtil	49
6.7	Configuración de seguridad de los endpoints	50
6.8	Función para crear usuario en el controlador	51
6.9	Función para crear usuario en el servicio	51
6.10	Función para transformar un UserDTO en un UserMO	52
6.11	Función para transformar un UserMO en un UserDTO	53
6.12	Clase repositorio para los usuarios	53
6.13	Test para el endpoint de crear usuarios	54
6.14	Test para el servicio de crear usuarios	55
6.15	Ejecución de todos los tests con Maven	56
6.16	Configuración de swagger en SiuGym	57
6.17	Swagger del API de SiuGym	57
6.18	Componente de Astro	58
6.19	Ejemplo de uso de Tailwind con Astro	59
6.20	Componente de React	60
7.1	Pantalla de registro de usuario en SiuGym	63
7.2	Pantalla con todos los ejercicios disponibles en SiuGym	64
7.3	Instrucciones, músculos y videotutorial de press de banca en SiuGym	64
7.4	Pantalla con las rutinas filtradas	65
7.5	Pantalla con los ejercicios de la rutina y el botón para añadirla al usuario	65
7.6	Vista móvil de la web en el apartado de las rutinas del usuario	66
7.7	Vista móvil de la web para ejecutar rutinas	66
7.8	Vista de la página de rutinas con el botón para crear rutinas	67
7.9	Página de creación de rutinas de SiuGym	67
7.10	Vista de la página de rutinas con las rutinas filtradas	68
7.11	Comentarios de una rutina junto con su valoración	68
7.12	Gráfico del progreso en el peso máximo del ejercicio de Press de Banca de Esteban	69
8.1	Alerta que aparece en la página al guardar una rutina	73
8.2	Aviso que se muestra al no rellenar un campo en la creación de rutinas	74
8.3	Ejemplos de ejercicios con sus respectivos botones en la creación de rutinas	75
8.4	Preguntas más frecuentes (FAQ) de SiuGym	75
A.1	Respuesta a la primera Pregunta del Cuestionario	83
A.2	Respuesta a la segunda Pregunta del Cuestionario	83
A.3	Respuesta a la tercera Pregunta del Cuestionario	84
A.4	Respuesta a la cuarta Pregunta del Cuestionario	84
A.5	Respuesta a la quinta Pregunta del Cuestionario	84
A.6	Respuesta a la sexta Pregunta del Cuestionario	85
A.7	Respuesta a la séptima Pregunta del Cuestionario	85
A.8	Respuesta a la octava Pregunta del Cuestionario	85
A.9	Respuesta a la novena Pregunta del Cuestionario	86
A.10	Respuesta a la décima Pregunta del Cuestionario	86
A.11	Respuesta a la undécima Pregunta del Cuestionario	86
A.12	Respuesta a la duodécima Pregunta del Cuestionario	87
A.13	Respuesta a la decimotercera Pregunta del Cuestionario	87
A.14	Respuesta a la decimocuarta Pregunta del Cuestionario	87
A.15	Respuesta a la decimoquinta Pregunta del Cuestionario	88
A.16	Respuesta a la decimosexta Pregunta del Cuestionario	88
A.17	Respuesta a la decimoséptima Pregunta del Cuestionario	88

CAPÍTULO 1

Introducción

Actualmente, con la creciente concienciación social, respaldada por numerosos estudios, seguir una vida saludable se ha convertido en el objetivo principal de gran parte de la sociedad. Esta preocupación por la salud no es solo una moda pasajera, sino una tendencia consolidada que refleja un cambio significativo en los hábitos y prioridades de las personas.

Para alcanzar esta meta, existen tres pilares básicos, como se menciona en el artículo del Colegio de Farmacéuticos de Sevilla [1]: la alimentación, el descanso y el ejercicio físico (véase la figura 1.1). Cada uno de estos pilares juega un papel crucial en el mantenimiento de un estado óptimo de salud y bienestar.

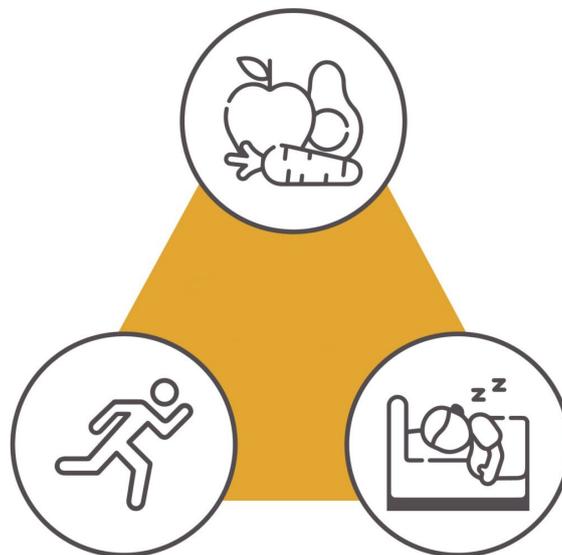


Figura 1.1: Los 3 pilares de la salud

La alimentación adecuada no solo proporciona los nutrientes necesarios para el funcionamiento del cuerpo, sino que también previene numerosas enfermedades crónicas y mejora la calidad de vida. Comer de manera equilibrada, incluyendo una variedad de frutas, verduras y proteínas, es esencial para mantener el cuerpo en plena forma.

El descanso, por otro lado, es igualmente vital. Dormir bien permite que el cuerpo y la mente se regeneren, mejorando la función cognitiva, el estado de ánimo y la salud general. La falta de sueño, en contraste, puede llevar a una serie de problemas de salud, incluyendo un mayor riesgo de enfermedades cardíacas, diabetes y depresión.

Sin embargo, este proyecto se centra en el tercer pilar: el ejercicio físico. El ejercicio regular no solo ayuda a mantener un peso saludable, sino que también fortalece los músculos y huesos, mejora la salud cardiovascular y aumenta la resistencia y la flexibilidad. Además, la actividad física tiene beneficios psicológicos significativos, como la reducción del estrés, la mejora del estado de ánimo y el aumento de la energía, como se comenta en el siguiente artículo del Ministerio de Sanidad [2].

Muchas son las formas de hacer deporte y llevar una vida saludable: salir a pasear, correr, jugar deportes individuales o colectivos, un sinfín de posibilidades, sin embargo, una que se encuentra en gran auge son los gimnasios.

Ir al gimnasio aporta una gran cantidad de ventajas, entre ellas:

- **Acceso a Equipamiento Especial.** En los gimnasios puedes encontrar con una gran cantidad de máquinas y utensilios para trabajar diferentes grupos musculares.
- **Asesoramiento.** Con la presencia de entrenadores o personas experimentadas, se puede pedir ayuda o consejo a la hora de tener una técnica correcta en la realización de los ejercicios, mejorando los resultados y evitando lesiones.
- **Motivación.** Al estar rodeado de personas con objetivos similares, se produce un ambiente motivador.
- **Temperatura.** La temperatura dentro de un gimnasio se mantiene constante independientemente del mes del año, evitando así problemas por exceso de temperaturas (insolación, hipotermia) o por inconveniencias naturales. (lluvias, tormentas, etc.)

Ajustándonos a los datos, como se puede ver en la figura 1.2, la popularidad de los gimnasios ha tenido un aumento constante (con excepción de los años del COVID, como se puede ver en estos artículos [3] [4]) alcanzando actualmente su máximo histórico en España con un 16,5 % de la población asistiendo a estos. (según Statista [5])

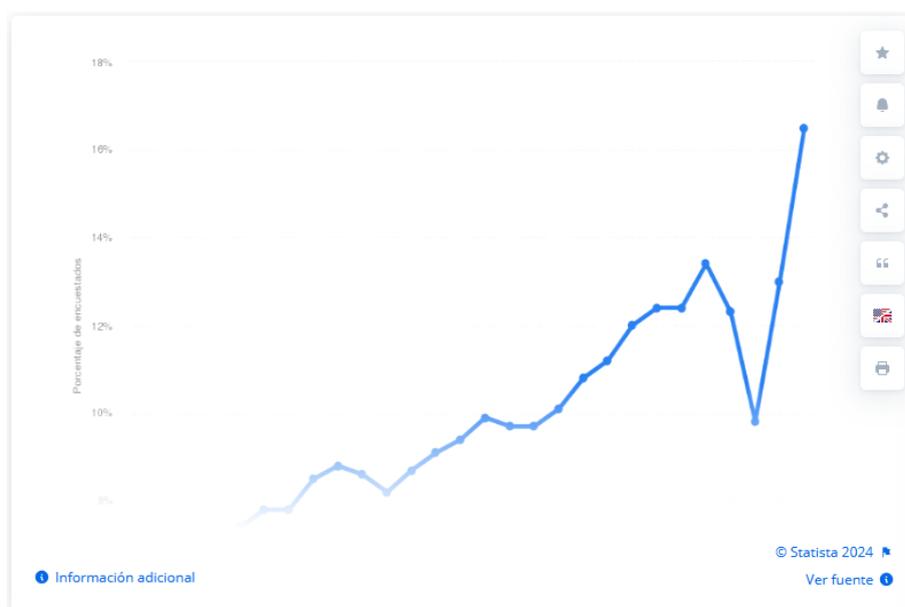


Figura 1.2: Porcentaje de individuos que fue a un gimnasio en España de 1996 a 2023. Fuente: Statista

1.1 Motivación

El auge de los gimnasios, combinado con el crecimiento exponencial de las redes sociales, ha facilitado el acceso a información sobre entrenamiento físico. Hoy en día, es más fácil que nunca aprender a entrenar correctamente gracias a vídeos didácticos de ejercicios, rutinas de entrenamiento disponibles en diversas plataformas y una amplia variedad de recursos en línea. Sin embargo, a pesar de esta abundancia de información, puede resultar desafiante para los usuarios (entre los que me incluyo) organizar y seguir sus rutinas de manera estructurada.

Este proyecto surgió de la necesidad de tener un espacio rápido, fiable, estético y personalizable en el cual organizar rutinas de gimnasio, con el fin de llevar un mejor control del entrenamiento físico.

1.2 Objetivos

Se pretende desarrollar una aplicación web que facilite la creación, organización y seguimiento de rutinas de gimnasio personalizadas, mejorando la experiencia de los usuarios en su entrenamiento físico. A continuación, se detallan los objetivos que se van a abordar:

1. **Personalizar.** Crear una web que permita diseñar rutinas de gimnasio adaptadas a aquellos objetivos que se marque cada uno.
2. **Informar.** Incluir descripciones y vídeos instructivos que ayuden a seguir una técnica correcta en los ejercicios.
3. **Monitorizar.** Implementar funcionalidades que permitan mantener registros de los distintos datos como por ejemplo el peso utilizado o las repeticiones, mostrando el progreso mediante estadísticas o gráficos.
4. **Gustar.** Diseñar una interfaz agradable y fácil de usar, accesible para usuarios con cualquier tipo de experiencia.
5. **Motivar.** Conseguir que la web sea adictiva, motivando a ir con más ganas a hacer ejercicio.

1.3 Estructura de la memoria

El trabajo esta organizado de la siguiente manera:

1. **Introducción.** En este capítulo se expone una breve presentación del tema, entrando en los objetivos que tiene la aplicación, la motivación para realizarla y la explicación de la estructura del documento.
2. **Estado del arte.** En este apartado se realiza un análisis de las distintas páginas web y aplicaciones existentes que permiten llevar un seguimiento de diferentes rutinas de gimnasio, donde se comparan sus características, ventajas y limitaciones.
3. **Metodología.** Este apartado explica las metodologías existentes para el desarrollo de aplicaciones y se detalla la metodología de Diseño Centrado en el Usuario (DCU) y las metodologías ágiles seleccionadas para llevar a cabo este proyecto (Kanban).

4. **Análisis de necesidades.** En este punto se recopilan los requisitos necesarios para el desarrollo de la web. Se incluyen los resultados de cuestionarios y encuestas realizados para identificar las necesidades y preferencias de los usuarios, además de la definición del perfil de usuario (persona modelo) y escenarios de uso.
5. **Análisis conceptual y diseño.** Esta sección se centra en el diseño y la creación de la estructura de la aplicación. Se presentan el diagrama de clases, el modelo de la base de datos utilizado por el servidor y los bocetos iniciales de las interfaces de usuario.
6. **Desarrollo de la solución.** En este apartado se explica el proceso de desarrollo de la aplicación web, describiendo la arquitectura del sistema y las herramientas utilizadas. Se detalla la programación tanto en la capa de negocio (Java SpringBoot) como en la capa de presentación (AstroJS + React + TailwindCSS), incluyendo ejemplos representativos de código.
7. **Producto desarrollado.** Aquí se presenta el producto final desde la perspectiva de la persona previamente definida. Se incluyen capturas de pantalla (web y móvil) de la página y se explica cómo están implementadas las diferentes funcionalidades basadas en los escenarios definidos.
8. **Validación.** Este capítulo se dedica a la validación de la aplicación. Se realiza una validación heurística para evaluar la usabilidad de la web, siguiendo los principios de Nielsen.
9. **Conclusiones y trabajos futuros.** En este apartado se resumen las conclusiones obtenidas a lo largo del proyecto, reflexionando sobre los objetivos alcanzados y mostrando las posibles mejoras y evoluciones futuras.
10. **Anexos.** Aquí se incluyen los anexos necesarios para complementar la información del proyecto, como un glosario con palabras específicas, los cuestionarios utilizados en el análisis de necesidades, y la relación del proyecto con las ODS.

CAPÍTULO 2

Estado del arte

En este capítulo se va a realizar una revisión de las aplicaciones y sitios web existentes que permiten llevar un seguimiento de rutinas de gimnasio. El objetivo es identificar las características comunes, evaluar sus ventajas y limitaciones, y proponer mejoras que serán tomadas en cuenta para la realización de la web.

2.1 Análisis del estado del arte

2.1.1. Entrenamientos.com

Entrenamientos.com es una plataforma en línea que ofrece rutinas de entrenamiento personalizadas, artículos sobre fitness y vídeos instructivos (figura 2.1). Ofrece una amplia variedad de artículos y consejos de fitness, así como vídeos de demostración que facilitan la correcta ejecución de los ejercicios. Estos recursos están diseñados para satisfacer las necesidades de todo tipo de usuarios, desde principiantes hasta avanzados, brindando una gran cantidad de información y apoyo para alcanzar sus objetivos físicos.

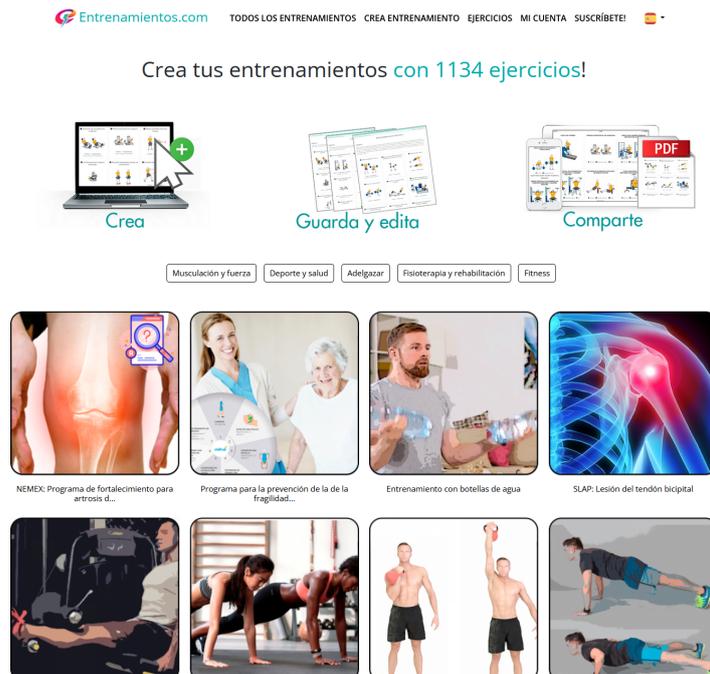


Figura 2.1: Pantalla inicial de la página web entrenamientos.com

Una de las principales ventajas de Entrenamientos.com es la amplia variedad de recursos disponibles. Los usuarios tienen acceso a una enorme cantidad de ejercicios y pueden beneficiarse de artículos informativos y consejos prácticos. Además, la plataforma permite una alta capacidad de personalización, ya que los usuarios pueden crear tablas de ejercicios personalizadas que se ajusten a sus necesidades específicas y objetivos de entrenamiento.

Sin embargo, a pesar de sus numerosas ventajas, Entrenamientos.com presenta algunas limitaciones. La interfaz de usuario puede resultar confusa para los nuevos usuarios, lo que puede dificultar la navegación y el uso eficiente de los recursos disponibles. Además, la plataforma carece de integración para el seguimiento del progreso personal, lo que significa que los usuarios deben buscar otras herramientas o métodos para monitorear su avance en sus rutinas de entrenamiento.

2.1.2. Cambiatufisico.com

Cambiatufisico.com (figura 2.2) es un sitio web dedicado al fitness que ofrece una sección detallada de rutinas de entrenamiento. Además de las rutinas, proporciona consejos nutricionales y planes de dieta, así como artículos informativos sobre diversos aspectos de la salud. Esta combinación de contenido hace que Cambiatufisico.com sea una fuente de información integral para quienes buscan mejorar su salud y condición física.

Una de las principales ventajas de Cambiatufisico.com es la amplitud y variedad de la información que ofrece. Los usuarios pueden acceder a rutinas de entrenamiento específicas y detalladas, recibir consejos nutricionales y planes de dieta personalizados, además de ofrecer una guía para saber que suplementación puede llegar a ser útil y recomendaciones de productos nutricionales.

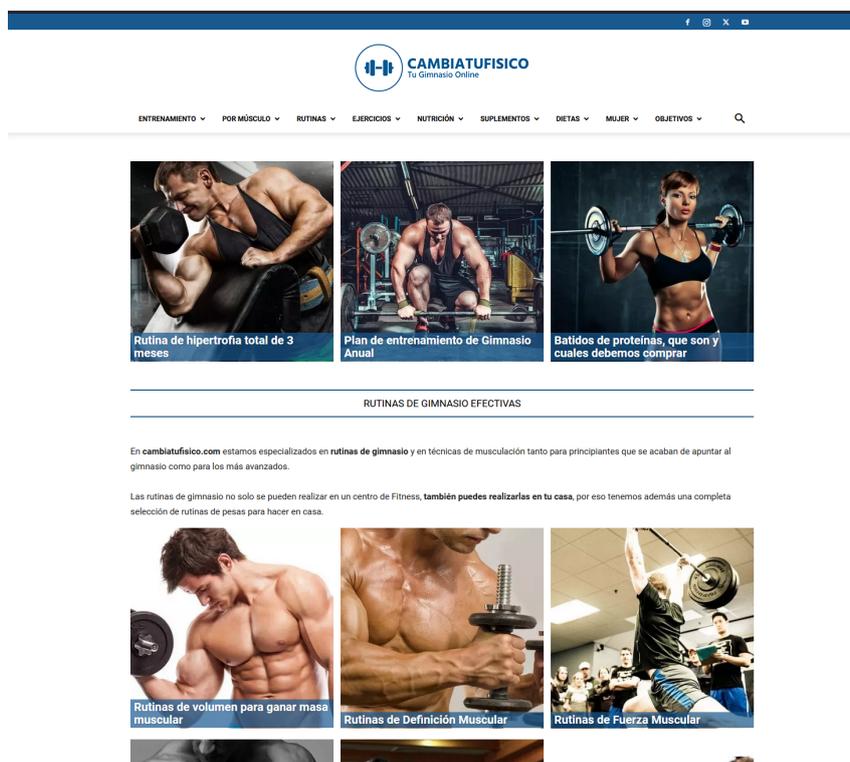


Figura 2.2: Pantalla inicial de la página web cambiatufisico.com

Sin embargo, cambiatufisico.com también presenta algunas limitaciones. La plataforma no ofrece herramientas interactivas para el seguimiento del progreso, lo que al igual que entrenamientos.com puede suponer sea necesario utilizar otros métodos para llevar un seguimiento de los progresos en el gimnasio. Además, el diseño de la web puede resultar anticuado, lo que podría afectar la experiencia del usuario, teniendo una interfaz móvil bastante poco atractiva y práctica. A lo mencionado anteriormente se le suma la aparición de anuncios continuos por la pantalla, entorpeciendo la experiencia del usuario al usar la página y pudiendo llegar a provocar frustraciones al equivocarse y clicar en los anuncios, llevando a páginas de terceros. (véase figura la 2.3)

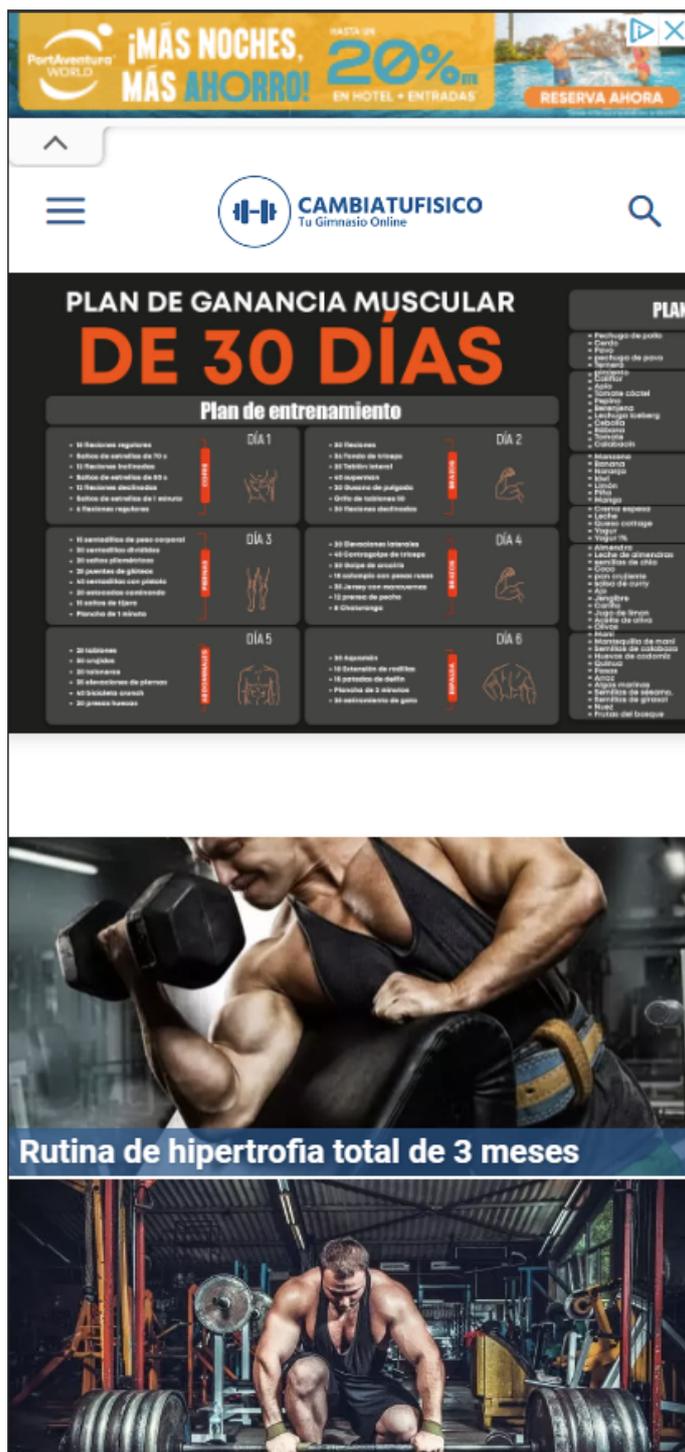


Figura 2.3: Pantalla inicial de la página web cambiatufisico.com desde una interfaz móvil

2.1.3. Fitbod

Fitbod (figura 2.4) es una aplicación móvil que se destaca por crear rutinas de entrenamiento personalizadas basadas en los objetivos del usuario y el equipo disponible. La aplicación utiliza un algoritmo avanzado que adapta las rutinas según el progreso del usuario, asegurando que los entrenamientos se mantengan efectivos y desafiantes. Además, Fitbod cuenta con una base de datos de ejercicios con instrucciones detalladas e integración con dispositivos de seguimiento de fitness, lo que permite a los usuarios llevar un control preciso de su actividad física.

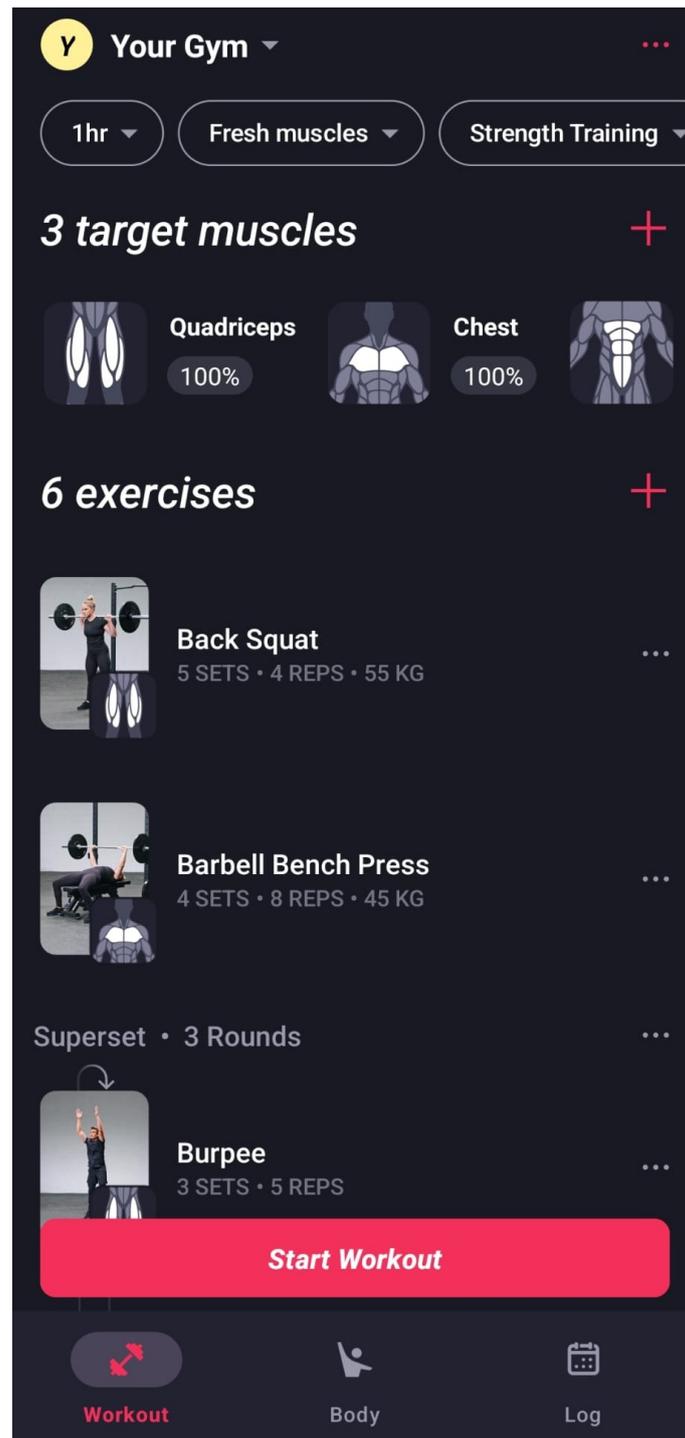


Figura 2.4: Pantalla inicial de la aplicación Fitbod

Una de las principales ventajas de Fitbod es el alto grado de personalización que ofrece en las rutinas de entrenamiento. Los usuarios pueden beneficiarse de una interfaz intuitiva y fácil de usar, que facilita la navegación y la planificación de los ejercicios. La capacidad de ajustar las rutinas en función del progreso personal y los objetivos específicos hace que Fitbod sea una herramienta valiosa para una amplia gama de usuarios.

Sin embargo, pese a que pueda considerarse como el estado del arte, Fitbod también presenta algunas desventajas. Algunas de sus características más avanzadas requieren una suscripción de pago (figuras 2.5 y 2.6) y solo te permite registrar 3 entrenamientos de prueba en su versión gratuita, lo que puede ser una barrera para algunos usuarios. Además, la aplicación puede no ser completamente adecuada para principiantes sin conocimientos previos, ya que la personalización avanzada y las instrucciones detalladas pueden resultar abrumadoras para aquellos que acaban de comenzar con su entrenamiento físico. Además de estas, la aplicación actualmente solo se encuentra disponible en inglés.

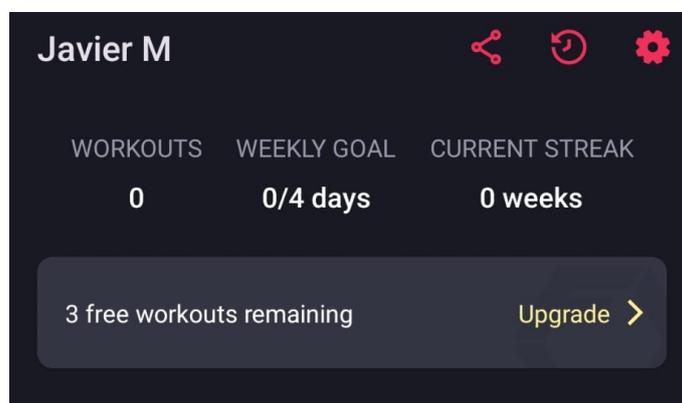


Figura 2.5: Restricción de entrenamientos en la aplicación Fitbod

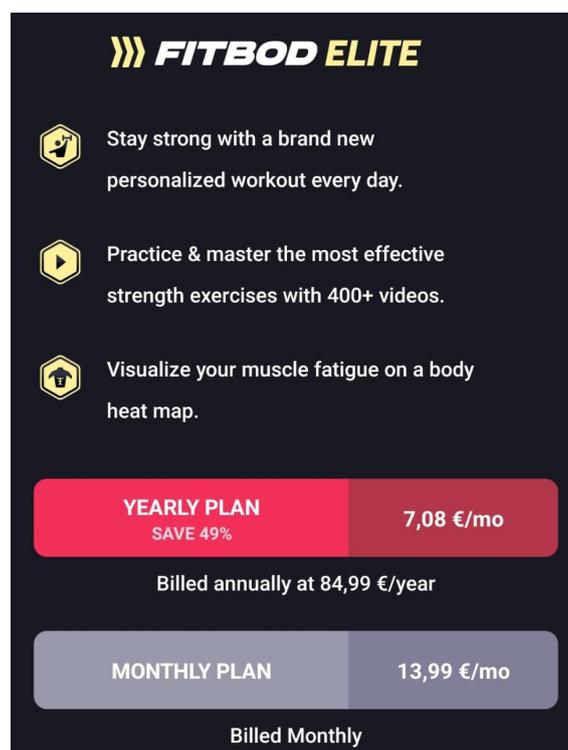


Figura 2.6: Planes de precio de la aplicación Fitbod

2.1.4. Bajar de Peso hombre

La aplicación Bajar de Peso hombre (tiene un clon llamado Bajar de Peso mujer) es una aplicación móvil diseñada específicamente para personas que desean adelgazar a través de rutinas de ejercicio y planes de dieta (figura 2.7). La aplicación ofrece entrenamientos personalizados adaptados a las necesidades individuales de cada usuario, además de proporcionar planes de dieta específicos y herramientas para el seguimiento del progreso y las estadísticas, lo que permite a los usuarios monitorizar su avance de manera efectiva.



Figura 2.7: Pantalla inicial de la aplicación Bajar de Peso hombre

Una de las principales ventajas de esta aplicación es su diseño claro con ejercicios sencillos para ayudar con la pérdida de peso. La integración de dieta y ejercicio en un solo lugar proporciona una solución completa, permitiendo a los usuarios gestionar todos los aspectos de su régimen de una manera coherente y estructurada. Las estadísticas y el seguimiento del progreso ayudan a los usuarios a mantenerse motivados y enfocados en sus objetivos.

Sin embargo, tiene unas cuantas desventajas. Aunque es excelente para quienes buscan adelgazar, su enfoque puede ser limitado en términos de ejercicios de fuerza y musculación, lo que puede no satisfacer las necesidades de todos los usuarios. Además, la interfaz de la aplicación puede no ser atractiva para todos, lo que podría afectar la experiencia de uso y la satisfacción general con la aplicación. Además, al igual que con la aplicación de *cambiatufisico*, tiene tanto anuncios estáticos como pop-ups (figura 2.8).



Toca de nuevo para salir

Figura 2.8: Anuncios pop-ups en la aplicación Bajar de Peso hombre

2.2 Crítica al Estado del Arte

El análisis de las aplicaciones y sitios web revisados revela varias áreas de oportunidad. Aunque muchos ofrecen una amplia gama de recursos y herramientas, presentan limitaciones en cuanto a la personalización, la usabilidad y la integración de seguimiento del progreso. Las aplicaciones móviles como Fitbod destacan por su personalización y facilidad de uso, pero a menudo requieren suscripciones para acceder a todas las funciones o disponen de molestos anuncios que perjudican al uso de la aplicación. Los sitios web como Entrenamientos.com y Cambiatufisico.com proporcionan información valiosa, pero carecen de herramientas interactivas para la gestión del progreso y la personalización de rutinas.

2.3 Propuesta

Tras el análisis de la situación actual, la propuesta a realizar trata del desarrollo de una página web, que llevará el nombre de SiuGym, que se propone llenar los vacíos identificados en las aplicaciones y sitios web analizados, ofreciendo una solución integral que combine sus mejores características y eliminando sus puntos débiles. La aplicación se enfocará en ofrecer una experiencia de usuario optimizada a través de varias funcionalidades clave:

- **Interfaz de Usuario Intuitiva:** Un diseño minimalista y fácil de usar que sea accesible para todos los niveles de experiencia.
- **Personalización y Flexibilidad:** Permitir a los usuarios crear y ajustar sus rutinas de manera flexible, adaptándose a sus objetivos específicos, nivel de experiencia y disponibilidad de tiempo.
- **Información Detallada:** Vídeos instructivos y descripciones detalladas para asegurar la correcta ejecución de los ejercicios y maximizar sus beneficios.
- **Seguimiento del Progreso:** Herramientas avanzadas de análisis y visualización para ayudar a los usuarios a monitorear su progreso de manera efectiva y ajustar sus rutinas según sea necesario.
- **Accesibilidad y Comunidad:** Diseñada para ser accesible y fomentar la creación de una comunidad activa, permitiendo a los usuarios compartir sus rutinas y progresos, motivándose mutuamente y compartiendo consejos.

Con estas mejoras, *SiuGym* pretende convertirse en una herramienta esencial para cualquier persona interesada en optimizar su entrenamiento en el gimnasio, ofreciendo una experiencia de usuario superior y resultados palpables, ayudando a todo aquel que se proponga superar sus límites.

CAPÍTULO 3

Metodología

En este trabajo de fin de grado (TFG) se ha utilizado una metodología basada en el Diseño Centrado en el Usuario (DCU), método visto en la asignatura de 4º del mismo nombre, y en la gestión ágil de proyectos mediante la metodología Kanban.

3.1 DCU. Diseño Centrado en el Usuario

Como se explica en la web nosolousabilidad [6], el Diseño Centrado en el Usuario (DCU) es una metodología de diseño que se centra en las necesidades, comportamientos y motivaciones de los usuarios para crear productos que sean fáciles de usar, útiles y satisfactorios. A diferencia de otros enfoques de diseño que se centran en las características o la tecnología, el DCU se basa en la comprensión profunda de las personas que utilizarán el producto.

El proceso de DCU se estructura en varias etapas fundamentales:

1. **Investigación cualitativa.** Esta fase implica la recolección de datos sobre los usuarios potenciales a través de entrevistas, encuestas, observaciones y análisis de estadísticas. El propósito es obtener una comprensión profunda de quiénes son los usuarios, cuáles son sus necesidades, y cómo interactúan con productos similares.

En el caso de este proyecto como se verá en el apartado 4.1, se utilizarán las técnicas de cuestionarios y análisis de estadísticas, con tal de obtener más información sobre los intereses y aspiraciones que tienen los posibles usuarios de la web.

2. **Definición de Requisitos.** Con la información recolectada, se definen los requisitos específicos del diseño. Estos requisitos guían todo el proceso de desarrollo y garantizan que el producto satisfaga las necesidades identificadas. Para ello se utilizará la técnica Persona, que como se explica en empredeaconciencia [7] consiste en una técnica para sintetizar los resultados de la investigación sobre los usuarios, ayudando a identificar patrones de comportamiento y crear arquetipos de usuarios.

En el caso de este proyecto como se verá como en el apartado 4.2 se define una persona modelo (Esteban) a partir del análisis de los datos obtenidos, persona sobre la cual se hará el diseño.

3. **Prototipado.** A partir de la persona definida, se crean prototipos de baja y alta fidelidad que permiten visualizar y probar las soluciones de diseño antes de la implementación final. Estos prototipos son iterados y refinados en función del feedback recibido durante las pruebas con usuarios.

En el caso de este proyecto, como se verá en el apartado 5.3, se han realizado una serie de prototipos de alta fidelidad tanto para las versiones web con vista de ordenador como en la versión web de móvil.

4. **Evaluación y validación de la Usabilidad.** Una vez están diseñados los prototipos son evaluados mediante pruebas de usabilidad que involucran a usuarios reales. Estas pruebas permiten identificar problemas de uso y realizar ajustes antes de la implementación final del producto.

En este proyecto se realiza el análisis de usabilidad en el apartado 8, utilizando una validación heurística.

Todas las etapas anteriores quedan resumidas en la figura 3.1 proporcionada por el sitio web medium [8].

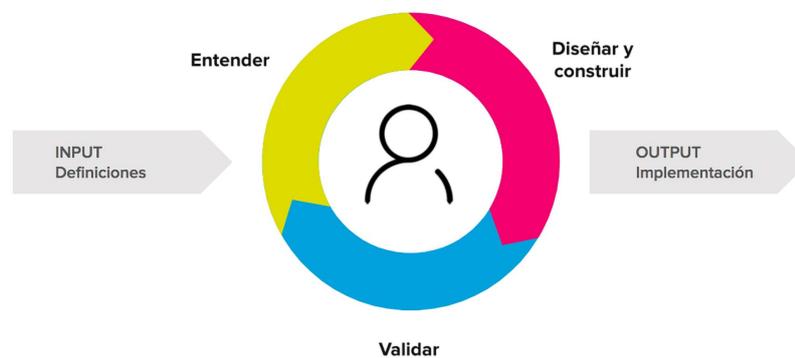


Figura 3.1: Etapas del diseño centrado en el usuario, Seba Gritti, Medium

3.2 Kanban

Kanban es una metodología ágil de gestión de proyectos que se centra en la visualización del flujo de trabajo y la limitación del trabajo en progreso (WIP) para mejorar la eficiencia y la flexibilidad. Originada en el sistema de producción de Toyota (véase [9]), Kanban utiliza un tablero visual dividido en columnas que representan las diferentes etapas de un proceso (ver figura 3.2). Cada tarea se representa con una tarjeta que avanza por el tablero conforme se completa cada etapa. Al limitar el trabajo en progreso (WIP), se evita sobrecargar y se garantiza que cada tarea reciba la atención necesaria, mejorando así la calidad y la productividad.



Figura 3.2: Ilustración de un tablero Kanban

Las ventajas que aporta seguir la metodología Kanban son las siguientes:

- **Visualización del Trabajo:** Permite al equipo ver el estado actual de las tareas, identificar cuellos de botella y gestionar el flujo de trabajo de manera efectiva. Concretamente en el desarrollo de este trabajo me ha permitido llevar una planificación de las tareas a realizar en cada momento.
- **Mejora Continua:** Promueve la revisión continua de las tareas y ajustes incrementales para optimizar la eficiencia y la calidad del trabajo.
- **Flexibilidad:** A diferencia de otras metodologías ágiles, como Scrum, Kanban no requiere sprints planificados, lo que permite un enfoque más adaptativo y menos rigidez en la planificación.
- **Reducción del Trabajo en Proceso:** Al limitar la cantidad de trabajo en proceso (WIP), se evitan sobrecargas en el equipo, lo que mejora la concentración y la calidad del trabajo.
- **Facilidad de Implementación:** Kanban es fácil de implementar y no requiere cambios estructurales drásticos, lo que lo hace accesible para equipos de cualquier tamaño.

Como se muestra en la Figura 3.3, para la realización de este proyecto se ha empleado la herramienta Trello. Esta plataforma digital, basada en los principios de Kanban, permite visualizar y organizar el flujo de trabajo a través de tableros y tarjetas. En este caso particular, se han definido cinco columnas: **Backlog** (para tareas pendientes), **To Do** (tareas próximas a iniciar), **Doing** (tareas en ejecución), **Review** (tareas en revisión) y **Done** (tareas finalizadas). Además se ha limitado la cantidad de tarjetas que puede haber en los procesos de To Do, Doing y Review con 3 o 4 tarjetas en cada momento como máximo. Cada tarjeta ha sido categorizada en según el tipo de trabajo del que trata siendo las categorías **Backend** (para tareas relacionadas con la lógica de negocio), **Frontend** (para tareas relacionadas con la capa de presentación de la web) y **TFG** (para las tareas relacionadas con el desarrollo y la redacción de este proyecto).

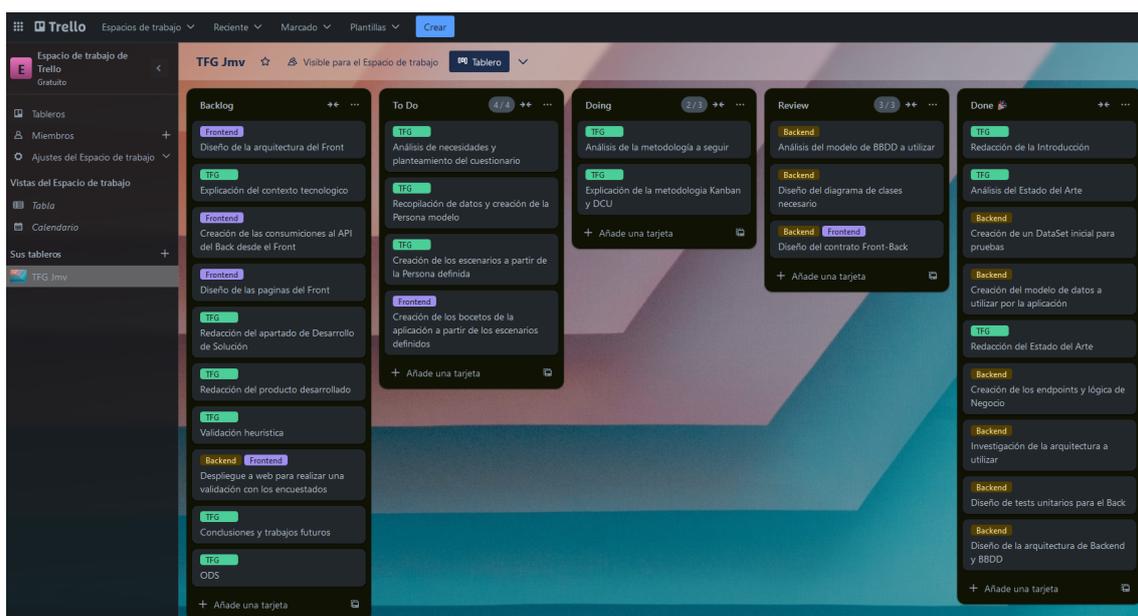


Figura 3.3: Tablero trello de este proyecto

CAPÍTULO 4

Análisis de necesidades

En esta fase del proyecto se llevará a cabo un análisis cualitativo y una definición de requisitos, primera y segunda etapa del Diseño Centrado en el Usuario (DCU). El objetivo principal es recopilar datos sobre los usuarios finales para comprender mejor sus necesidades y comportamientos. A través de esta información, se establecerá una base sólida de requisitos que orientarán el proceso de diseño.

Para alcanzar este objetivo, se han definido tres subtareas:

1. **Recogida de datos.** En esta subtarea se utilizarán cuestionarios para obtener información sobre los posibles usuarios de manera estructurada y eficiente. La elección de esta técnica se justifica por su capacidad para recopilar datos de un gran número de participantes de forma rápida y anónima. Además, se complementará con el análisis de diferentes estudios y artículos, con tal de obtener una visión más completa del contexto de uso.
2. **Creación de la Persona.** La información recopilada será analizada en profundidad para identificar patrones, tendencias y necesidades clave de los usuarios. Los resultados de este análisis serán utilizados y justificados para la creación de la persona modelo.
3. **Creación de los escenarios** Una vez definida la persona, se procederá a la creación de escenarios de uso. Estos escenarios simularán situaciones reales en las que la persona interactúa con el sistema, permitiendo identificar las funcionalidades y características necesarias para satisfacer sus necesidades.

4.1 Investigación cualitativa. Recogida de datos

4.1.1. Cuestionario y estudios de documentación

Para asegurar que el desarrollo de la aplicación esté alineado con las necesidades reales de los usuarios, se ha llevado a cabo un proceso de análisis y comprensión de los hábitos y motivaciones de aquellas personas que frecuentan el gimnasio, han tenido alguna experiencia en él o tienen la intención de comenzar a entrenar.

Para ello, como se ha comentado anteriormente se ha realizado una encuesta, en la cual se han identificado una serie de preguntas clave que ayudarán a obtener una visión clara de los aspectos más importantes para los usuarios. Estas preguntas clave son:

1. ¿Cuántas veces y cuánto tiempo dedican al entrenamiento?

2. ¿Como suelen planificar sus entrenamientos en el gimnasio?
3. ¿Cuáles son las principales motivaciones de los usuarios para ir al gimnasio?
4. ¿Que tipo de diseño web los atrae y los ayuda?
5. ¿Que edad y que nivel de experiencia tecnológica tienen?
6. ¿Qué tipo de contenido les puede ayudar a realizar sus entrenamientos?

Con tal de facilitar las respuestas de los usuarios, se han formulado una serie de preguntas concretas que permitirán responder a las cuestiones clave mencionadas anteriormente con mayor facilidad. Estas preguntas son:

1. Esta semana, ¿Cuantas veces has ido al gimnasio?
2. Este mes, ¿Cuánto tiempo has dedicado en cada entrenamiento?
3. ¿Tus rutinas de entrenamiento son fijas o las adaptas según tu disponibilidad?
4. ¿Qué te gustaría lograr a largo plazo con tu entrenamiento?
5. ¿Qué te motiva a seguir entrenando de forma regular?
6. ¿Sueles planificar tus entrenamientos con anticipación?
7. ¿Utilizas alguna aplicación o herramienta para planificar tus rutinas?
8. ¿Consideras que debes planificar tu entrenamiento?
9. ¿Qué dificultades encuentras al momento de diseñar tus propias rutinas de entrenamiento?
10. ¿Qué tipo de información te viene bien para planificar los entrenamientos?
11. ¿Cuál es tu edad?
12. ¿Con qué frecuencia utilizas dispositivos móviles (smartphone, tablet) para acceder a información?
13. ¿Tienes el modo oscuro activado en el móvil?
14. ¿Que tipo de diseño web prefieres?
15. En este ultimo mes, ¿Que plataformas audiovisuales has utilizado?
16. En un tutorial, ¿Prefieres que te expliquen las cosas paso a paso o te resuman lo más importante?
17. Cuando tienes que aprender algo sobre una materia concreta, ¿que utilizas?

Además de la información que se obtenga a partir de la encuesta, es destacable también la información que aporta el siguiente artículo [10]. Este artículo proporciona una guía valiosa sobre la duración óptima de un entrenamiento, lo que puede ser crucial para entender cómo los usuarios organizan su tiempo en el gimnasio.

El artículo menciona que el tiempo ideal de un entrenamiento puede variar según el objetivo de la persona, pero generalmente, un entrenamiento efectivo debería durar entre 45 minutos y una hora. Esto es relevante porque nos ayuda a contextualizar las respuestas de los usuarios en nuestra encuesta en cuanto al tiempo que dedican a entrenar.

Además, el artículo señala la importancia de la calidad sobre la cantidad, subrayando que entrenar más de dos horas podría ser contraproducente, ya que podría aumentar el riesgo de lesiones y la fatiga innecesaria, destacando la necesidad de utilizar una buena técnica a la hora de entrenar.

Además, este artículo también subraya la importancia de adaptar el tiempo de entrenamiento a los objetivos personales, algo que se alinea con otras preguntas de la encuesta, como las que indagan sobre las motivaciones y los objetivos a largo plazo de los usuarios. Entender si los usuarios están ajustando correctamente su tiempo de ejercicio en función de sus metas permitirá ofrecerles herramientas más personalizadas para optimizar sus rutinas.

4.1.2. Datos obtenidos y análisis

Para responder a las preguntas anteriores se ha realizado una encuesta en la plataforma Google Forms (véase el Anexo A), de la cual se han obtenido los siguientes resultados:

1. De la primera pregunta se obtiene que el grueso de nuestros posibles usuarios suele ir entre 1 y 5 veces en un 76 % de los casos, siendo más de la mitad que van entre 1 y 3 veces. (véase el resultado de la encuesta en la figura A.1)
2. En las respuestas de esta pregunta se puede ver que el grueso del tiempo que dedican al entrenamiento está concentrado en la franja de los 45-60 minutos, coincidiendo y reafirmando el estudio anteriormente mostrado. (véase el resultado de la encuesta en la figura A.2)
3. De esta pregunta se puede extraer que las rutinas son fijas o adaptadas según disponibilidad en la misma medida. (véase el resultado de la encuesta en la figura A.3)
4. Con esta pregunta se puede ver como el objetivo que más destaca entre los encuestados es el de ponerse en forma seguido de cerca de cuidar la salud. (véase el resultado de la encuesta en la figura A.4)
5. Aquí se reafirma la importancia que le dan los encuestados a la hora de motivarse para ir a entrenar a mejorar la salud, además de la pérdida de peso y la ganancia de músculo, quedando en menor medida la constancia. (véase el resultado de la encuesta en la figura A.5)
6. Con esta pregunta se puede ver que el porcentaje de personas que no se planifican los entrenamientos es muy bajo con un 10 % en comparación a los que se lo preparan. (véase el resultado de la encuesta en la figura A.6)
7. De esta séptima pregunta se ve que la mayoría no utiliza ninguna aplicación o herramienta de planificación de rutinas. (véase el resultado de la encuesta en la figura A.7)
8. Aquí se observa como la mayoría de encuestados opina que si que deberían planificarse sus entrenamientos. (véase el resultado de la encuesta en la figura A.8)
9. De aquí se obtiene que los dos principales motivos por los que se encuentran dificultades en la creación de rutinas son la falta de conocimiento y la falta de tiempo. (véase el resultado de la encuesta en la figura A.9)

10. Con esta pregunta se ve que la información que mas necesitan las personas encuestadas es información sobre ejercicios y sobre rutinas. (véase el resultado de la encuesta en la figura A.10)
11. A partir de esta pregunta se obtiene que la edad media de los encuestados se encuentra principalmente entre los 18 a 35 años con un 85 %. (véase el resultado de la encuesta en la figura A.11)
12. De esta pregunta se ve como prácticamente la totalidad de los encuestados utilizan diariamente dispositivos móviles para consultar información. (véase el resultado de la encuesta en la figura A.12)
13. Aquí podemos ver como el porcentaje de usuarios que utilizan el modo oscuro es equivalente a los que no lo utilizan. (véase el resultado de la encuesta en la figura A.13)
14. A partir de este resultado se puede ver como el diseño minimalista y sencillo junto con el diseño moderno y visual son los preferidos por los encuestados. (véase el resultado de la encuesta en la figura A.14)
15. De esta pregunta se puede ver como la totalidad de los encuestados hace uso de la plataforma YouTube así como de otro tipo de plataformas tanto de reproducción de vídeo como de imágenes. (véase el resultado de la encuesta en la figura A.15)
16. A partir de esta decimosexta pregunta se ve como lo que prefieren la mayoría de encuestados son explicaciones paso a paso con un 52.5 % frente a las explicaciones resumidas con un 13.6 %. (véase el resultado de la encuesta en la figura A.16)
17. En esta ultima pregunta se observa como para el aprendizaje los encuestados prefieren vídeos o búsquedas online sobre otros métodos como libros o artículos científicos. (véase el resultado de la encuesta en la figura A.17)

4.2 Definición de Persona

Con la información obtenida de los datos anteriores, podemos concluir en la descripción de una persona llamada Esteban.

Esteban es un profesor de matemáticas de 24 años que vive en Valencia, apasionado por los números y la lógica. En su día a día, se dedica a enseñar con entusiasmo a sus alumnos, siempre buscando la manera de hacer que las matemáticas sean accesibles y comprensibles para todos. Esteban es una persona extrovertida, a la que le encanta interactuar con sus colegas y estudiantes, disfrutando de las conversaciones que le permiten compartir conocimientos y aprender cosas nuevas. A pesar de su personalidad abierta, es también alguien muy calculador y meticuloso, características que se reflejan tanto en su trabajo como en su vida personal. Siempre busca la manera más eficiente de hacer las cosas, asegurándose de que cada decisión esté bien pensada y basada en hechos concretos.

Además de su dedicación a la enseñanza, Esteban es un gran aficionado al baloncesto. Aunque su apretada agenda como profesor le deja poco tiempo libre, trata de aprovechar cada oportunidad para disfrutar de este deporte que le apasiona. Recientemente, ha decidido que quiere mejorar su condición física y empezar a ir al gimnasio. Su objetivo es ponerse en forma, no solo para sentirse mejor consigo mismo, sino también para mejorar su rendimiento en la cancha de baloncesto. Esteban sabe que, dado su estilo de vida

ocupado, será un desafío encontrar tiempo para entrenar de manera regular, pero está decidido a hacerlo. Como persona calculadora, está buscando una manera de optimizar su tiempo en el gimnasio, asegurándose de que cada sesión de entrenamiento sea eficiente y efectiva.

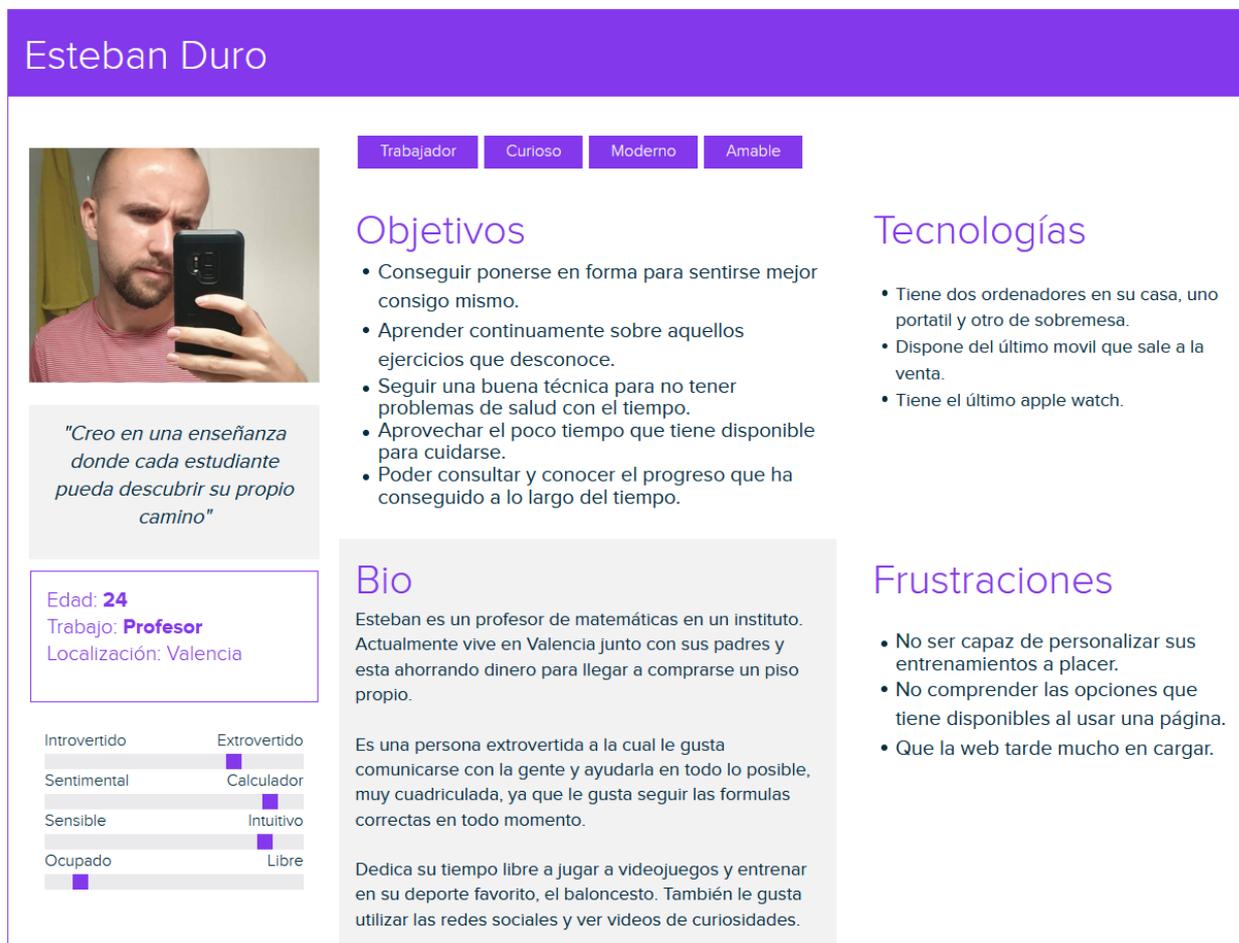


Figura 4.1: Ficha de Esteban Duro

4.2.1. Justificación de la persona con los datos obtenidos

La creación del perfil de Esteban como persona modelo se fundamenta en los resultados obtenidos de la encuesta y las recomendaciones del artículo.

Los datos de la encuesta muestran que un 76 % de los encuestados asisten al gimnasio entre 1 y 5 veces por semana, y la mayoría dedica entre 45 y 60 minutos a cada sesión. Este comportamiento es representativo de Esteban, quien también busca optimizar su tiempo en el gimnasio para adaptarlo a su ocupada agenda como profesor.

Además, la encuesta revela que las rutinas de entrenamiento suelen ser fijas o adaptadas según la disponibilidad, un aspecto relevante para Esteban, quien debido a su estilo de vida ocupado, necesita flexibilidad en su planificación. El 85 % de los encuestados tiene entre 18 y 35 años, al igual que Esteban, y comparten la motivación de ponerse en forma y mejorar su salud. La mayoría de los encuestados también reconocen la importancia de planificar sus entrenamientos, pero enfrentan dificultades como la falta de conocimiento y tiempo, desafíos que Esteban también enfrenta.

Los objetivos y frustraciones de Esteban (véase la figura 4.1) están directamente relacionados con los resultados obtenidos de la encuesta y el análisis de necesidades realizado previamente, por un lado los objetivos:

1. **Conseguir ponerse en forma para sentirse mejor consigo mismo.** Según la encuesta, el objetivo principal para muchos usuarios, incluido Esteban, es ponerse en forma y cuidar su salud. Estos objetivos son fundamentales para Esteban, quien desea sentirse mejor consigo mismo y mantener un estilo de vida saludable.
2. **Aprender continuamente sobre aquellos ejercicios que desconoce.** La encuesta muestra que los encuestados consideran crucial obtener información sobre ejercicios y rutinas para planificar mejor sus entrenamientos. Esteban comparte esta necesidad, ya que busca aprender constantemente sobre nuevos ejercicios que puedan mejorar su rendimiento y técnica.
3. **Seguir una buena técnica para no tener problemas de salud con el tiempo.** La importancia de la calidad sobre la cantidad de entrenamiento, mencionada en el artículo, resuena con Esteban, quien está enfocado en seguir una buena técnica para evitar problemas de salud a largo plazo. Esto se alinea con su deseo de entrenar de manera efectiva y segura.
4. **Aprovechar el poco tiempo que tiene disponible para cuidarse.** El hecho de que la mayoría de los usuarios entrenan entre 45 y 60 minutos y que Esteban es una persona ocupada refuerza su objetivo de maximizar el poco tiempo que tiene para el gimnasio. La encuesta revela que los usuarios valoran entrenamientos efectivos y bien planificados, algo que Esteban también busca.
5. **Poder consultar y conocer el progreso que ha conseguido a lo largo del tiempo.** El deseo de Esteban de seguir su progreso está justificado por la tendencia de los encuestados a valorar herramientas que les permitan planificar y revisar sus entrenamientos. Aunque muchos no usan herramientas de planificación actualmente, la mayoría reconoce su importancia, reflejando el interés de Esteban en monitorear su evolución.

Por otro lado las frustraciones:

1. **No ser capaz de personalizar sus entrenamientos a placer.** La encuesta muestra que la falta de conocimiento y tiempo son las principales dificultades para crear rutinas de entrenamiento personalizadas. Esto coincide con la frustración de Esteban, quien desea personalizar sus entrenamientos, pero enfrenta obstáculos debido a la complejidad de la planificación y la falta de opciones claras.
2. **No comprender las opciones que tiene disponibles al usar una página.** La encuesta indica que los usuarios prefieren diseños web que sean intuitivos y fáciles de usar. La frustración de Esteban con páginas complicadas refleja esta preferencia, ya que un diseño confuso o con opciones poco claras puede impedirle aprovechar al máximo las herramientas disponibles.
3. **Que la web tarde mucho en cargar.** La rapidez y eficiencia de una página web son cruciales para usuarios como Esteban, que tienen poco tiempo disponible. Esta frustración está en línea con la necesidad de maximizar el tiempo de entrenamiento y evitar pérdidas de tiempo debido a problemas técnicos, como páginas que tardan en cargar.

4.3 Escenarios

Escenario 1. Crear cuenta y aprender a hacer un ejercicio concreto

Esteban, que es nuevo en el gimnasio, decide que quiere aprender a hacer el ejercicio de press de banca. Se registra en una aplicación de entrenamiento que le recomendaron. Después de crear su cuenta, navega por la aplicación hasta encontrar el ejercicio específico que quiere aprender. La aplicación le muestra un video tutorial detallado y una guía paso a paso sobre cómo realizar el ejercicio correctamente. Esteban sigue las indicaciones, toma notas sobre la técnica correcta, y siente que ahora está listo para intentarlo en el gimnasio.

Escenario 2. Ver una rutina de pecho, aprender sus ejercicios y guardarla

Esteban ha decidido que quiere centrarse en desarrollar su pecho. Abre su aplicación de entrenamiento, busca una rutina específica para el pecho, y encuentra una que parece adecuada para su nivel. Explora cada ejercicio de la rutina, mirando videos instructivos y leyendo las descripciones para asegurarse de que entiende cómo hacerlos correctamente. Satisfecho con la rutina, la guarda en su perfil de la aplicación para poder acceder a ella fácilmente en su próxima visita al gimnasio.

Escenario 3. En el gimnasio hacer la rutina que ya tenía guardada

Una tarde, Esteban llega al gimnasio con poco tiempo disponible. Abre su aplicación de entrenamiento, donde ya tiene guardada la rutina de pecho que seleccionó anteriormente. Simplemente sigue la secuencia de ejercicios que la aplicación le indica, concentrándose en cada uno sin preocuparse por qué hacer a continuación. Esto le permite optimizar su tiempo en el gimnasio y asegurarse de que sigue un plan estructurado.

Escenario 4. Crear una rutina personalizada

Después de varias semanas en el gimnasio, Esteban ha identificado los ejercicios que más le gustan y que mejor funcionan para él. Decide que quiere crear su propia rutina personalizada. Abre la aplicación, elige los ejercicios favoritos para cada grupo muscular, y organiza el orden en que quiere realizarlos. También ajusta las repeticiones y el número de series según lo que ha aprendido sobre su propio cuerpo. Una vez creada, guarda la rutina y la nombra Rutina de Esteban para poder seguirla en sus próximos entrenamientos.

Escenario 5. Elegir una rutina a partir de la dificultad y las opiniones

Esteban quiere variar su entrenamiento y busca una rutina que sea desafiante y que se enfoque en la espalda y los bíceps. Usa la función de búsqueda de su aplicación para filtrar las rutinas según la dificultad y los grupos musculares específicos. Encuentra varias opciones, pero antes de decidirse, lee los comentarios y opiniones de otros usuarios que ya han probado esas rutinas. Se fija en las valoraciones y en las experiencias compartidas para asegurarse de que la rutina que elige es adecuada para sus objetivos y nivel de condición física.

Escenario 6. Analizar estadísticas

Después de varios meses de entrenamiento constante, Esteban quiere evaluar su progreso. Abre su aplicación de entrenamiento y revisa las estadísticas que ha ido registrando. Observa las gráficas de su rendimiento en diferentes ejercicios, las tendencias de peso levantado y la cantidad de repeticiones realizadas a lo largo del tiempo. Analiza su progreso y se da cuenta de que ha aumentado su fuerza y resistencia en varios ejercicios clave. Este análisis le ayuda a ajustar su plan de entrenamiento para seguir mejorando en áreas específicas.

4.3.1. Justificación de los escenarios con los datos de la persona

Escenario 1. Crear cuenta y aprender a hacer un ejercicio concreto

Este escenario se relaciona directamente con el objetivo de Esteban de *aprender continuamente sobre aquellos ejercicios que desconoce*. Al crear una cuenta, Esteban puede acceder a una base de datos de ejercicios, lo que le permite descubrir y aprender nuevas técnicas que mejorarán su rendimiento y evitarán posibles lesiones. Este proceso de aprendizaje continuo es esencial para alguien que busca mejorar su forma física de manera efectiva.

Escenario 2. Ver una rutina de pecho, aprender sus ejercicios y guardarla

Aquí, el escenario se conecta con el objetivo de *seguir una buena técnica para no tener problemas de salud con el tiempo*. Al ver y aprender los ejercicios de una rutina específica, Esteban se asegura de que está realizando los movimientos correctamente, lo que es crucial para evitar lesiones. Además, guardar la rutina le permite seguirla en sus entrenamientos futuros, garantizando que su técnica se mantenga consistente y efectiva.

Escenario 3. En el gimnasio hacer la rutina que ya tenía guardada

Este escenario se alinea con el objetivo de *aprovechar el poco tiempo que tiene disponible para cuidarse*. Esteban, siendo una persona ocupada, valora la eficiencia. Tener una rutina previamente seleccionada le permite maximizar su tiempo en el gimnasio, enfocándose directamente en el entrenamiento sin perder tiempo en decidir qué ejercicios hacer.

Escenario 4. Crear una rutina personalizada

La creación de una rutina personalizada está directamente relacionada con el objetivo de *poder consultar y conocer el progreso que ha conseguido a lo largo del tiempo* y con su frustración de *no ser capaz de personalizar sus entrenamientos a placer*. Al personalizar su rutina, Esteban puede adaptar su entrenamiento a sus necesidades y preferencias, lo que le permite hacer un seguimiento más preciso de su progreso en los ejercicios que más le importan.

Escenario 5. Elegir una rutina a partir de la dificultad y las opiniones

Este escenario se justifica en relación con el objetivo de *aprender continuamente sobre aquellos ejercicios que desconoce y seguir una buena técnica para no tener problemas de salud con el tiempo*. Esteban utiliza los comentarios y opiniones de otros usuarios para asegurarse de que la rutina que elige es adecuada para su nivel y objetivos, lo que le ayuda a mantener una técnica correcta y aprender de las experiencias de otros.

Escenario 6. Analizar estadísticas

Este escenario está fuertemente vinculado con el objetivo de *poder consultar y conocer el progreso que ha conseguido a lo largo del tiempo*. Analizar sus estadísticas permite a Esteban evaluar su rendimiento, lo que le motiva a seguir mejorando y ajustando su entrenamiento según sus metas a largo plazo. Este análisis también le proporciona una visión clara de su evolución, lo cual es fundamental para mantener la motivación y la efectividad de su entrenamiento.

CAPÍTULO 5

Análisis Conceptual y Diseño

En esta sección se aborda el diseño y la planificación de la aplicación. Se incluyen el diagrama de clases, que muestra las relaciones entre los diferentes componentes del sistema, y el modelo de la base de datos utilizado para gestionar y almacenar la información de manera eficiente. Además, se presentan los bocetos iniciales de las interfaces de usuario, diseñados para ofrecer una experiencia intuitiva y accesible.

5.1 Diagrama de clases

El diagrama de clases presentado en la figura 5.1 representa la estructura básica de un sistema de gestión de entrenamientos en un gimnasio, destacando las principales entidades involucradas: User, Routine, Exercise, WorkoutLog, MuscleGroup, Category, y Comment. Cada una de estas clases encapsula información clave relacionada con sus respectivas funciones. Por ejemplo, la clase User contiene atributos como username, email, y password, mientras que la clase Exercise detalla el name, description, instructions, entre otros. Estas clases se relacionan entre sí mediante asociaciones que reflejan la interacción de los usuarios con sus rutinas, ejercicios, y el seguimiento de su progreso.

Las relaciones entre estas clases están bien definidas para capturar la dinámica de uso del sistema. Por ejemplo, un User puede seguir múltiples Routines, y a su vez, cada Routine puede incluir varios Exercises. Asimismo, los WorkoutLogs permiten registrar las actividades de un usuario en un día específico, asociándose tanto con el User como con los Exercises que realiza. Esta estructura relacional facilita la personalización y el seguimiento detallado del progreso de los usuarios dentro del sistema.

Además, el sistema incluye elementos que permiten la interacción y retroalimentación, como la clase Comment, que permite a los usuarios dejar comentarios sobre las rutinas, aportando un componente social y de mejora continua. Las Categories y MuscleGroups proporcionan una organización adicional, ayudando a clasificar las rutinas y ejercicios según su enfoque y los músculos que trabajan, respectivamente. En conjunto, este diagrama de clases proporciona una visión integral del sistema, asegurando que cada aspecto del entrenamiento y la experiencia del usuario esté bien representado y estructurado.

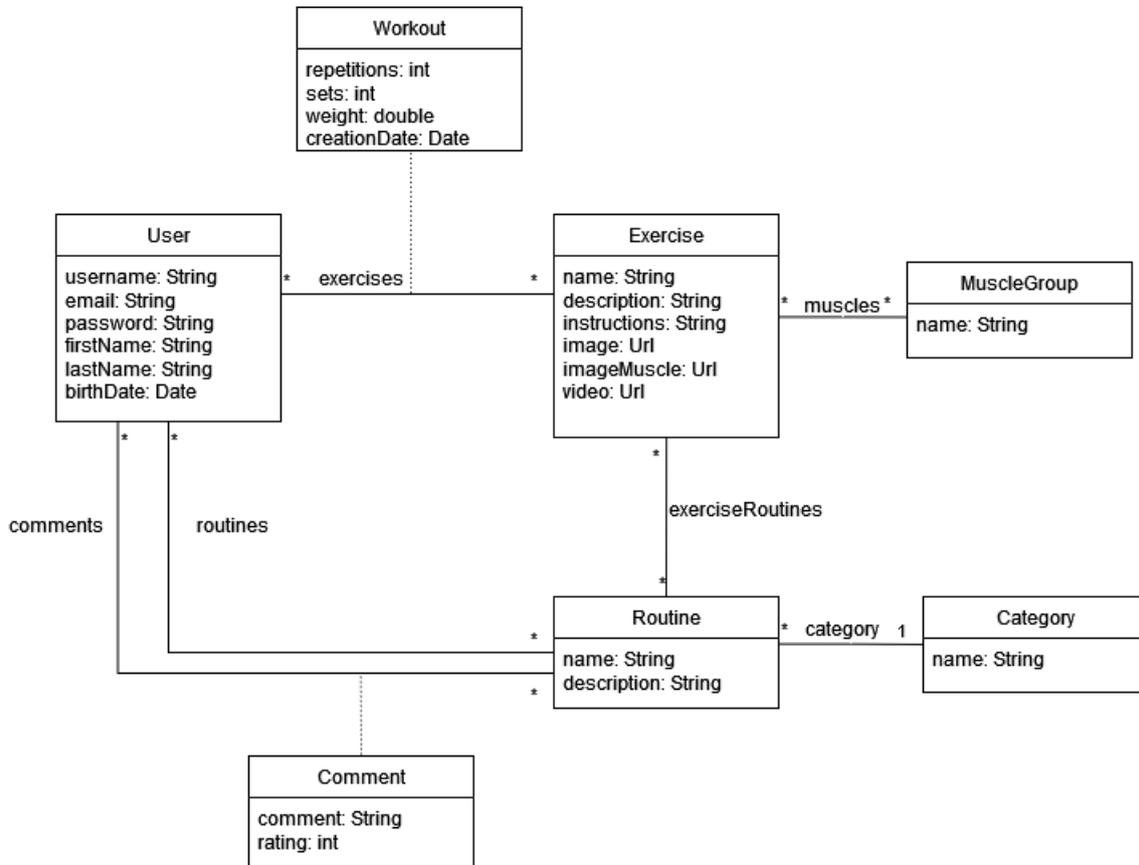


Figura 5.1: Diagrama de clases

5.2 Modelo de la BD que use el servidor

Para este proyecto se ha optado por utilizar una base de datos relacional, específicamente PostgreSQL. Postgres ofrece robustez, flexibilidad en las consultas y un manejo eficaz de transacciones, lo que asegura la integridad y consistencia de los datos. Su escalabilidad y soporte para operaciones simples y complejas lo hacen ideal para las necesidades actuales y futuras de la aplicación. (Véase la documentación [11])

Para garantizar la eficiencia y la integridad de los datos, se ha procedido a normalizar la base de datos hasta la Tercera Forma Normal (3FN). Este nivel de normalización es crucial para eliminar redundancias y evitar las anomalías de actualización, inserción y eliminación que podrían comprometer la consistencia de los datos. Para ello se ha pasado el diagrama de clases anterior a un diseño de base de datos relacional (figura 5.2) donde cada entidad relevante del modelo se ha convertido en una tabla. Estas tablas están relacionadas entre sí mediante claves primarias y claves foráneas, lo que asegura que los datos estén correctamente vinculados y que no haya duplicación innecesaria. Además, se han definido índices en campos clave para optimizar las consultas y mejorar el rendimiento general del sistema.

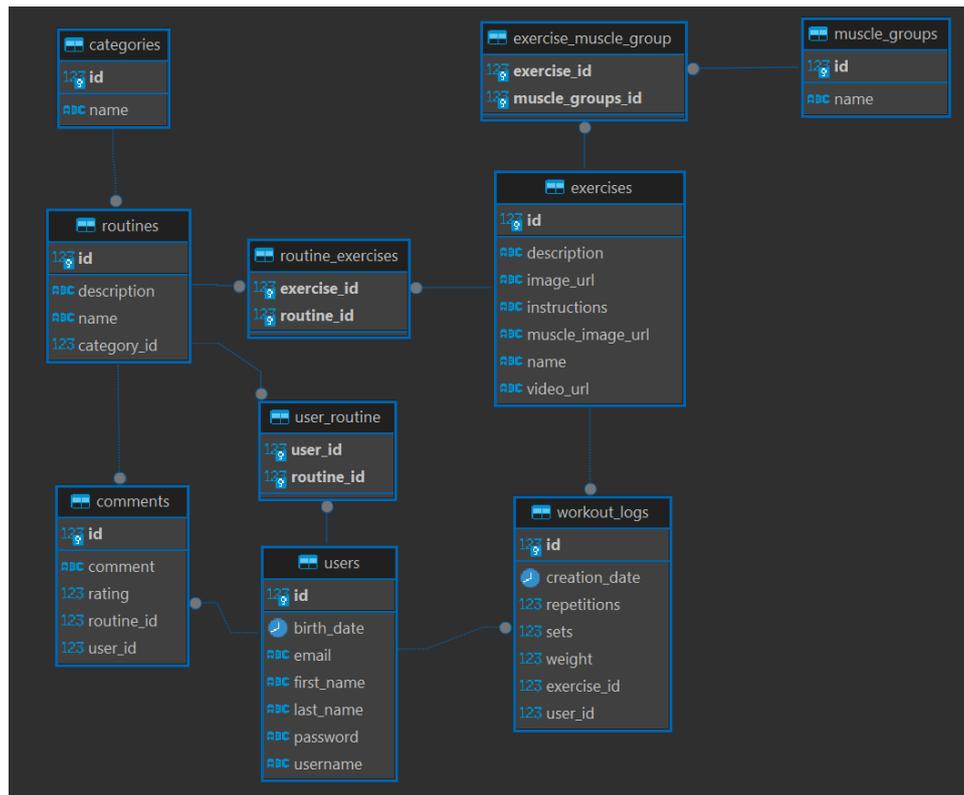


Figura 5.2: Diagrama Entidad-Relación en 3FN

A continuación se procede a dar una breve descripción de la utilidad de cada tabla:

- **users**: Contiene la información básica de los usuarios, como su nombre, email, fecha de nacimiento, y credenciales.
- **exercises**: Registra los ejercicios disponibles en la plataforma, incluyendo detalles como descripción, instrucciones y recursos multimedia.
- **routines**: Almacena las rutinas de ejercicios creadas, con información sobre su nombre, descripción y categoría.
- **categories**: Contiene las categorías disponibles para clasificar las rutinas de ejercicios.
- **comments**: Almacena los comentarios y calificaciones que los usuarios dejan sobre las rutinas.
- **muscle_groups**: Define los diferentes grupos musculares que pueden estar asociados a los ejercicios.
- **exercise_muscle_group**: Relaciona los ejercicios con los grupos musculares que trabajan.
- **routine_exercises**: Asocia los ejercicios específicos a las rutinas, permitiendo la personalización de las mismas.
- **user_routine**: Relaciona a los usuarios con las rutinas que han agregado o que están siguiendo.
- **workout_logs**: Registra los detalles de las sesiones de entrenamiento realizadas por los usuarios, incluyendo repeticiones, series, peso y fecha de creación.

El modelo de la base de datos está diseñado para gestionar de manera eficiente la información relacionada con usuarios, rutinas de ejercicios, y los componentes que las conforman. Este modelo permite a la aplicación ofrecer una experiencia personalizada, donde los usuarios pueden acceder a rutinas predefinidas, agregar rutinas a su perfil personal, y explorar ejercicios específicos. (tal y como se comenta en el apartado de la propuesta 2.3)

5.3 Bocetos de las interfaces de la app

A continuación, se presentan los bocetos de alta fidelidad de la web para cada uno de los escenarios descritos, tanto en su versión de escritorio como en la de móvil.

Estos bocetos han sido elaborados utilizando **Figma**, una potente herramienta que facilita la creación, edición y diseño de interfaces web, permitiendo una visualización detallada y precisa de cómo se verá la aplicación en diferentes dispositivos.

Escenario 1. Crear cuenta y aprender a hacer un ejercicio concreto

Lo primero que hace Esteban es crearse una cuenta, para ello pulsa en el botón de regístrate. (véase las figuras 5.3 y 5.4)

Esta primera pantalla incluye una breve presentación de las funcionalidades de la aplicación, ofreciendo una barra de navegación con las distintas pantallas disponibles para acceder.



Figura 5.3: Pantalla inicial en PC



Figura 5.4: Pantalla inicial en Móvil

Después de esto va al apartado de ejercicios (figuras 5.5 y 5.6) y pulsa en el ejercicio que quiere aprender (figuras 5.6 y 5.7).

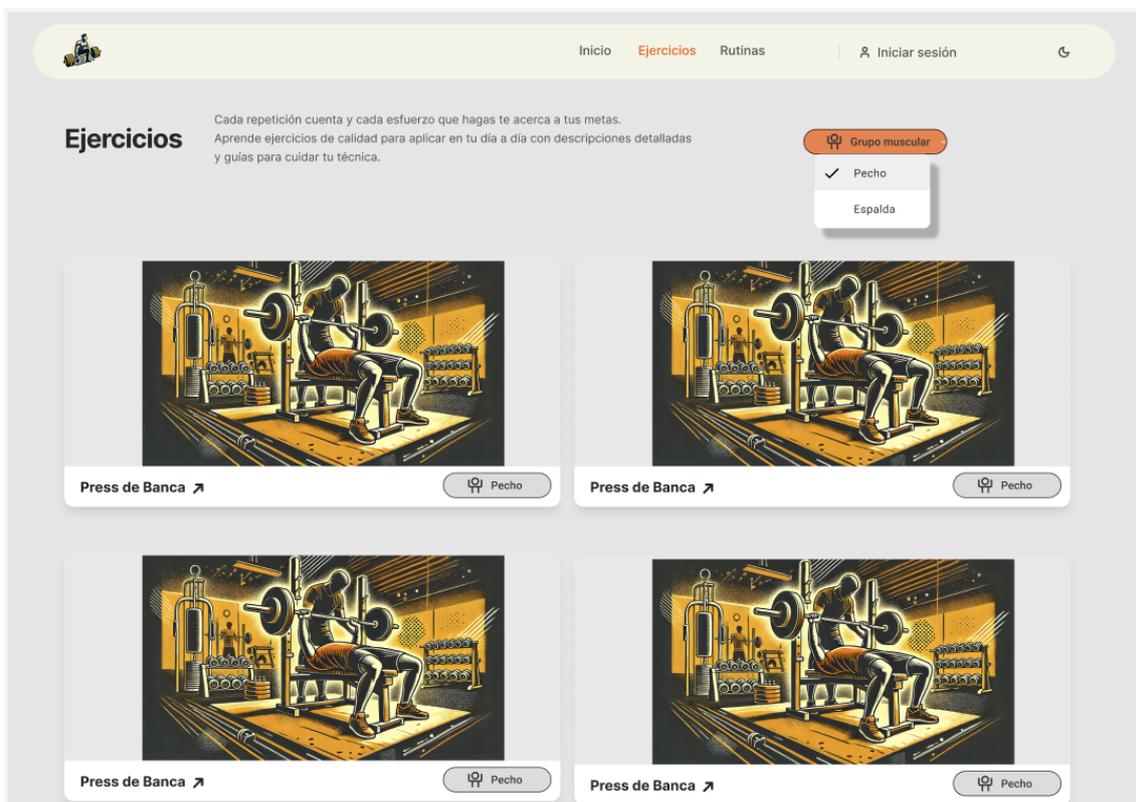


Figura 5.5: Pantalla general de ejercicios en PC

Se puede ver una pantalla con un filtro para enseñar los ejercicios en función de su grupo muscular. Se muestran los ejercicios indicando su nombre y músculos afectados, diseñados en un formato de carta con un *scroll* infinito.

Para las pantallas de los ejercicios concretos, se muestra como título el nombre del ejercicio junto con una breve descripción, las instrucciones para realizar el ejercicio paso a paso, una imagen con los músculos afectados y un corto vídeo en el cual se explica la técnica correcta.

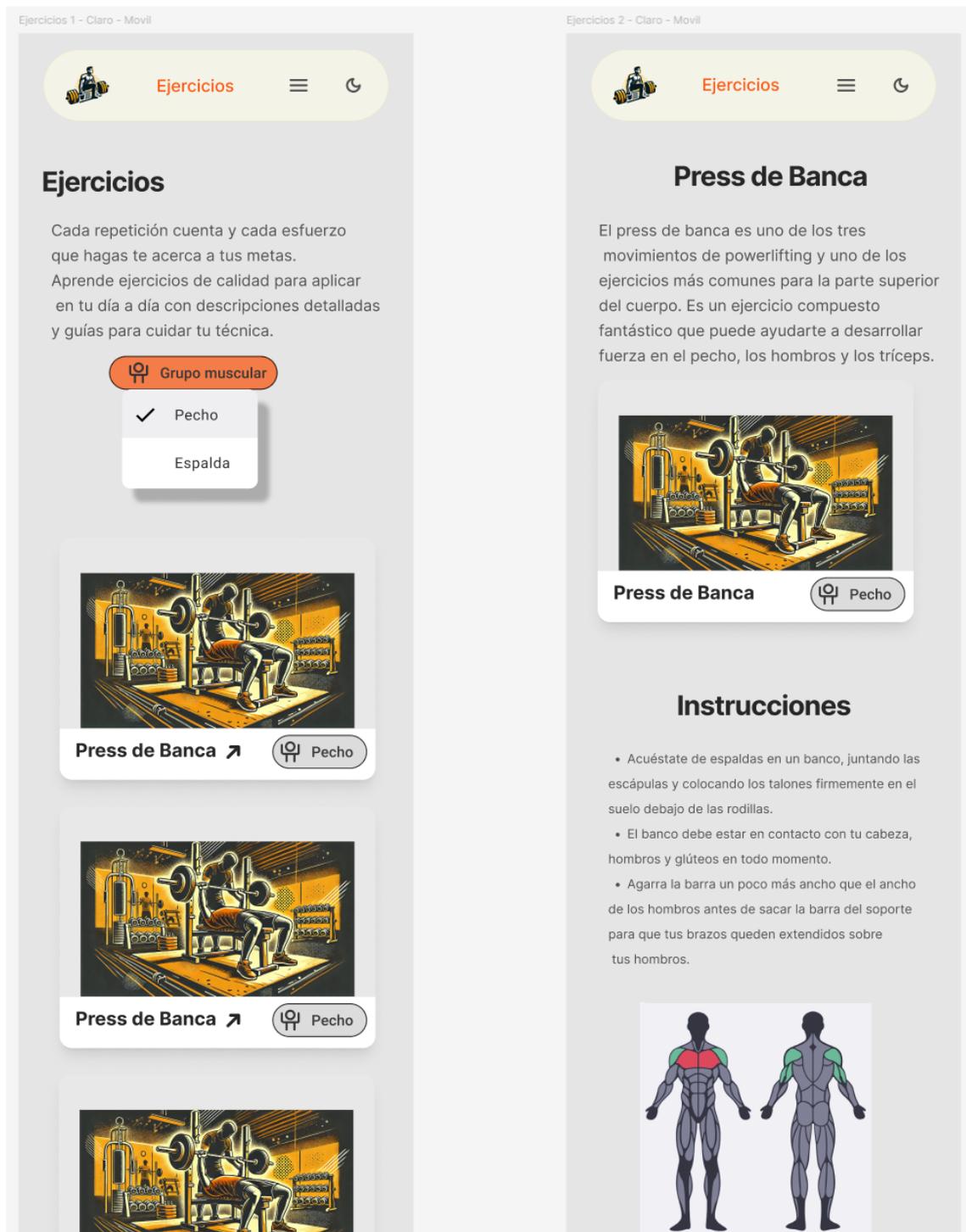


Figura 5.6: Pantalla lista de ejercicios y ejercicio concreto en Móvil

Inicio **Ejercicios** Rutinas 🔍 Iniciar sesión 🏠

Press de Banca

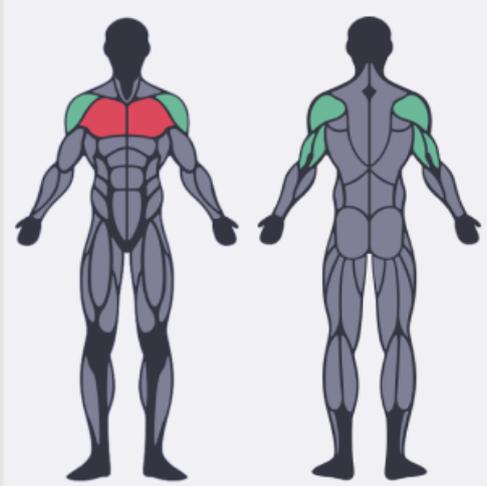
El press de banca es uno de los tres movimientos de powerlifting y uno de los ejercicios más comunes para la parte superior del cuerpo. Es un ejercicio compuesto fantástico que puede ayudarte a desarrollar fuerza en el pecho, los hombros y los tríceps.



Press de Banca 🏠 Pecho

Instrucciones

- Acuéstate de espaldas en un banco, juntando las escápulas y colocando los talones firmemente en el suelo debajo de las rodillas.
 - El banco debe estar en contacto con tu cabeza, hombros y glúteos en todo momento.
- Agarra la barra un poco más ancho que el ancho de los hombros antes de sacar la barra del soporte para que tus brazos queden extendidos sobre los hombros.
- Mantén el núcleo contraído respirando hacia el estómago y flexionando los músculos abdominales, baja la barra hasta tu pecho flexionando los codos y manteniéndolos a un ángulo de 45 grados con respecto a tu torso.
 - Toca suavemente el centro de tu pecho con la barra antes de exhalar y devolverla a la posición inicial.



Videotutorial



Figura 5.7: Pantalla de un ejercicio en PC

Escenario 2. Ver una rutina de pecho, aprender sus ejercicios y guardarla

Esteban accede a la sección de rutinas (figuras 5.8 y 5.9) y pincha en una de ellas. Tras ver los ejercicios que tiene, decide guardársela (véase las figuras 5.10 y 5.11).

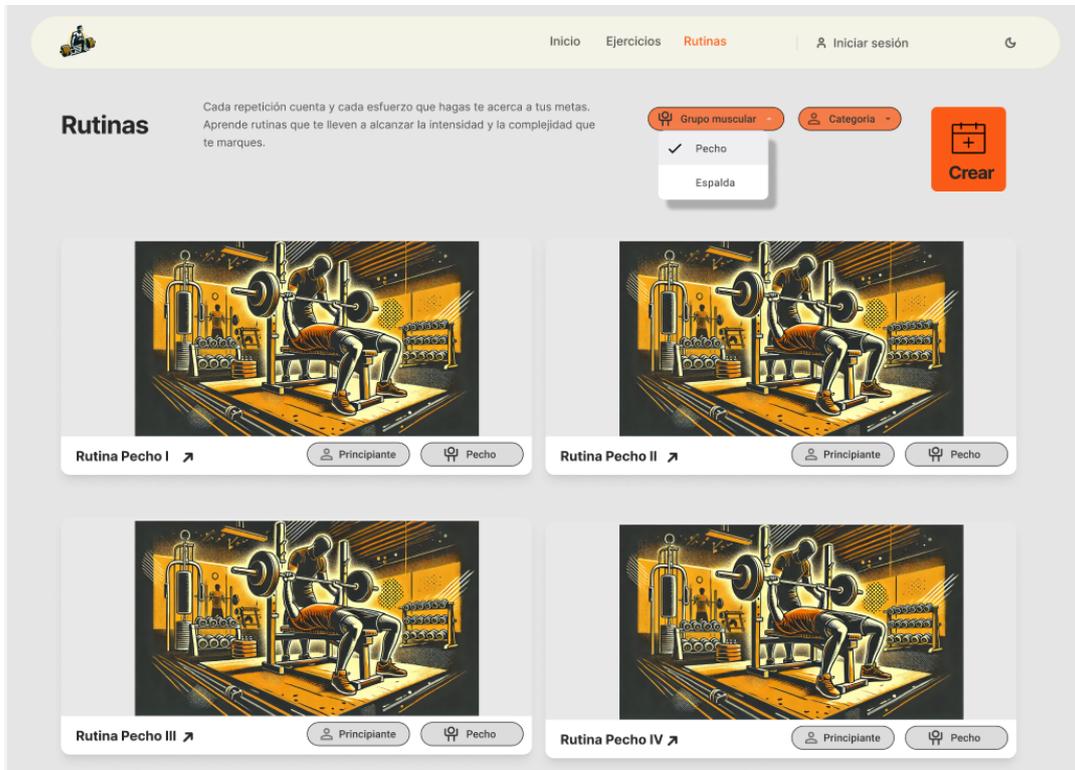


Figura 5.8: Pantalla general de rutinas en PC



Figura 5.9: Pantalla general de rutinas en Móvil

Al igual que la pantalla con la lista de ejercicios, en esta pantalla de rutinas se muestra una lista de rutinas permitiendo filtrar por Categoría y grupos musculares de los ejercicios que incluye. Una vez se selecciona una rutina se muestra los ejercicios que la conforman, permitiendo guardarla para usarla o pulsar en los distintos ejercicios para aprender su funcionamiento.

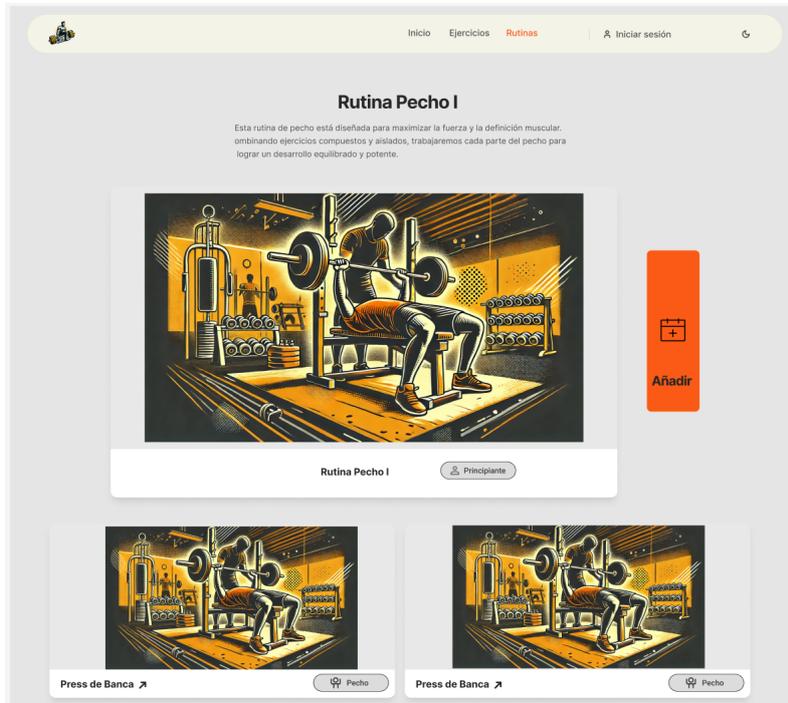


Figura 5.10: Pantalla de rutina concreta en PC



Figura 5.11: Pantalla de rutina concreta en Móvil

Escenario 3. En el gimnasio hacer la rutina que ya tenía guardada

Esteban accede al apartado de sus rutinas (figuras 5.12 y 5.13) y presiona el botón de empezar el entrenamiento (figuras 5.13 y 5.14).

Al presionar sobre el usuario, aparece una pantalla con datos generales del usuario y la lista de los ejercicios que tiene guardados. Al pulsar sobre el botón naranja se muestra la rutina seleccionada con los ejercicios que incluye, permitiendo configurar las series, repeticiones y peso que se realizan. Por defecto deberán aparecer las series y repeticiones recomendadas y el último peso realizado para cada ejercicio.

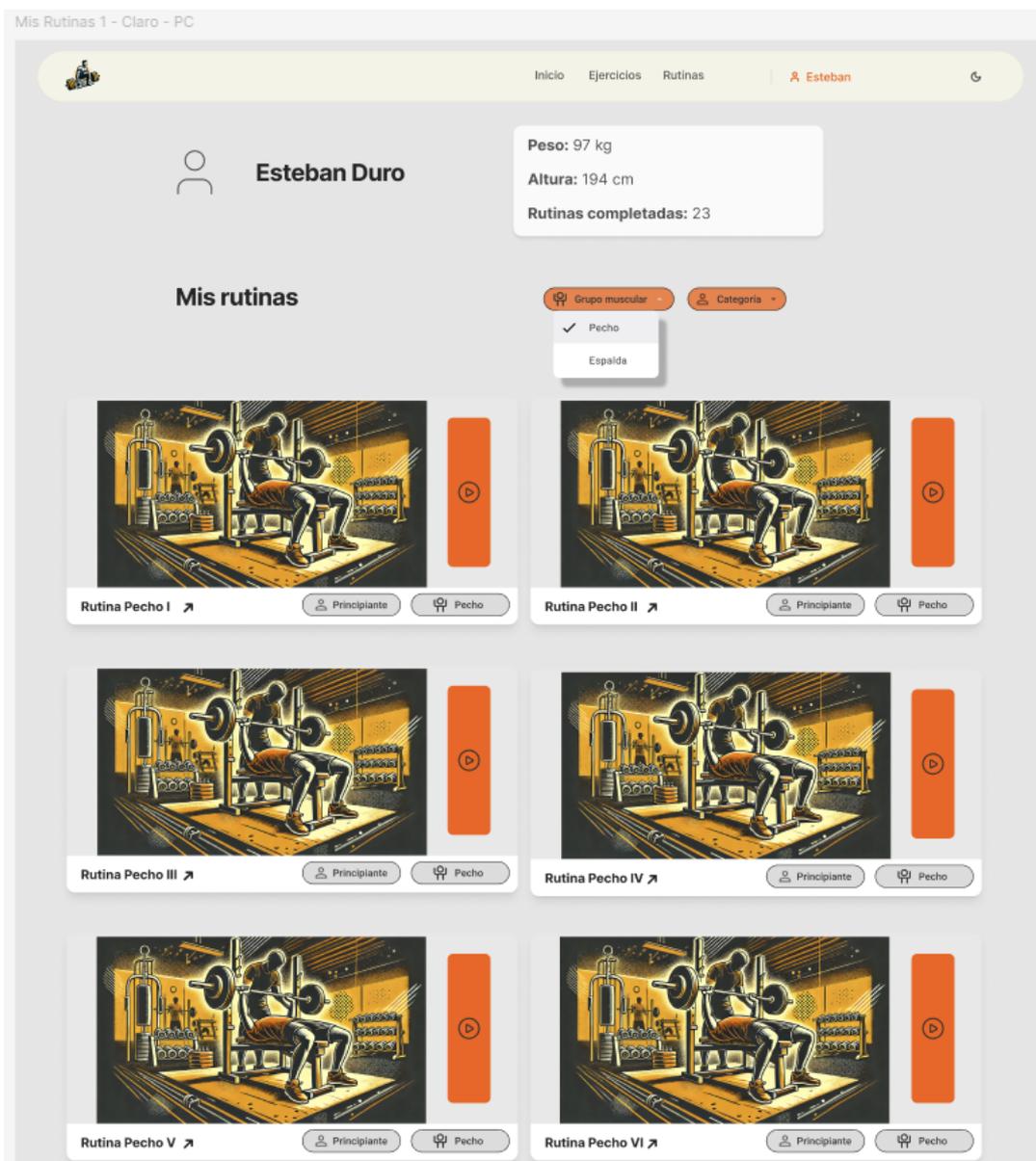


Figura 5.12: Pantalla de mis rutinas en PC

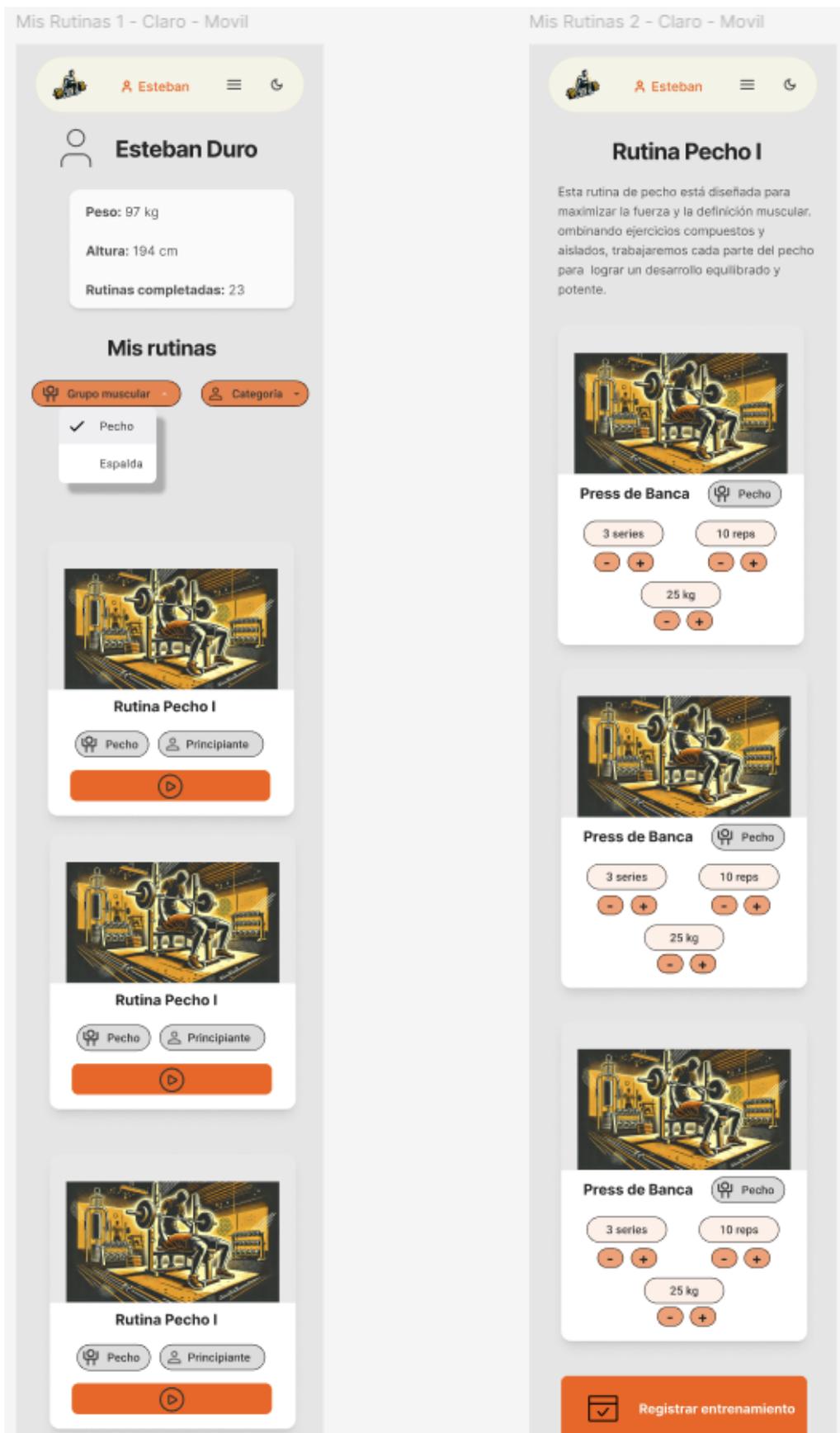


Figura 5.13: Pantalla de mis rutinas y registrar rutina en Móvil

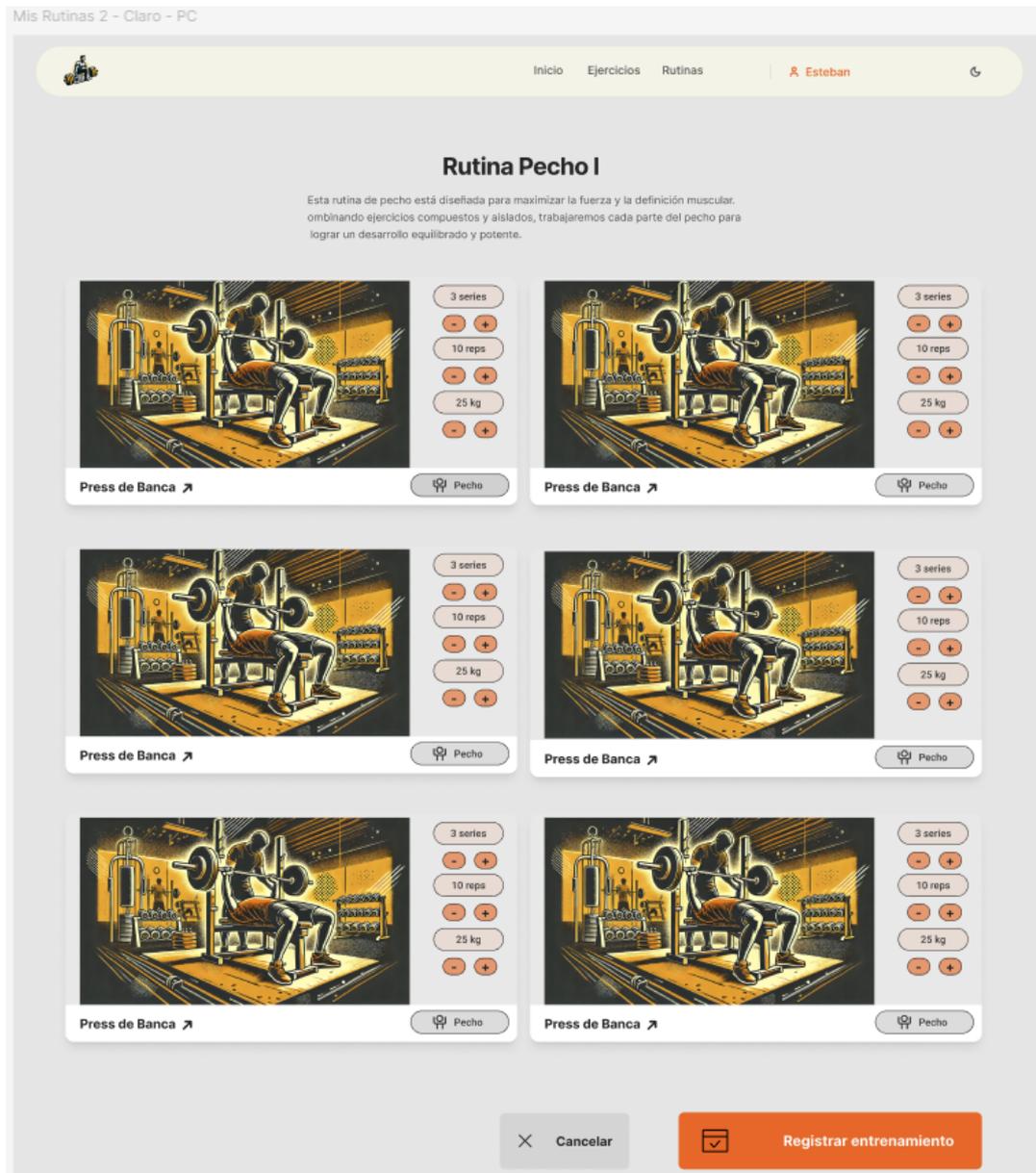


Figura 5.14: Pantalla de registrar rutina en PC

Escenario 4. Crear una rutina personalizada

Esteban decide crear una rutina para ello presiona el botón de crear rutina (véase figura 5.8 y 5.9) y una vez en la pantalla selecciona los ejercicios que le gustan (figuras 5.15 y 5.16).

En esta pantalla de creación de rutinas se pueden filtrar los ejercicios en función del grupo muscular y añadirlos a la rutina, indicando la descripción y el título que se le quiere dar a esta, mostrando aquellos ejercicios que ya se encuentran añadidos.

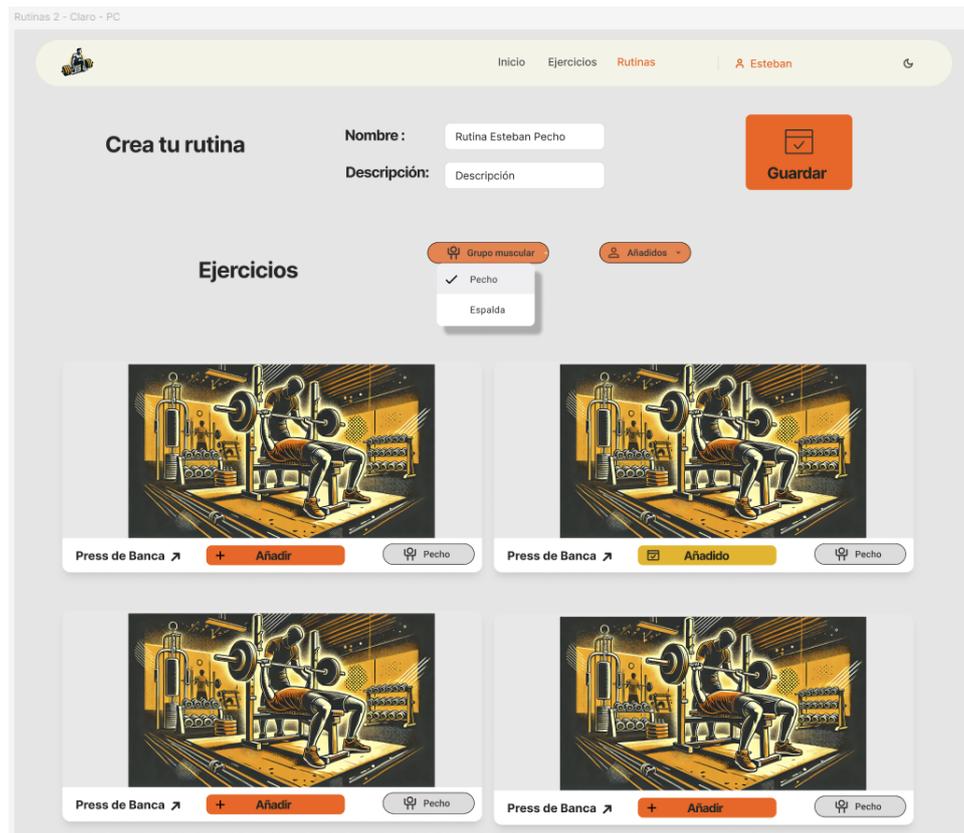


Figura 5.15: Pantalla de crear rutina en PC



Figura 5.16: Pantalla de crear rutina en Móvil

Escenario 5. Elegir una rutina a partir de la dificultad y las opiniones

Esteban está buscando una rutina para entrenar, ve una que le parece interesante y tras leer los comentarios, decide guardarla (figura 5.17).

Cuando tienes una rutina añadida es posible añadir un comentario a esta, fomentando la colaboración entre los usuarios de la aplicación.

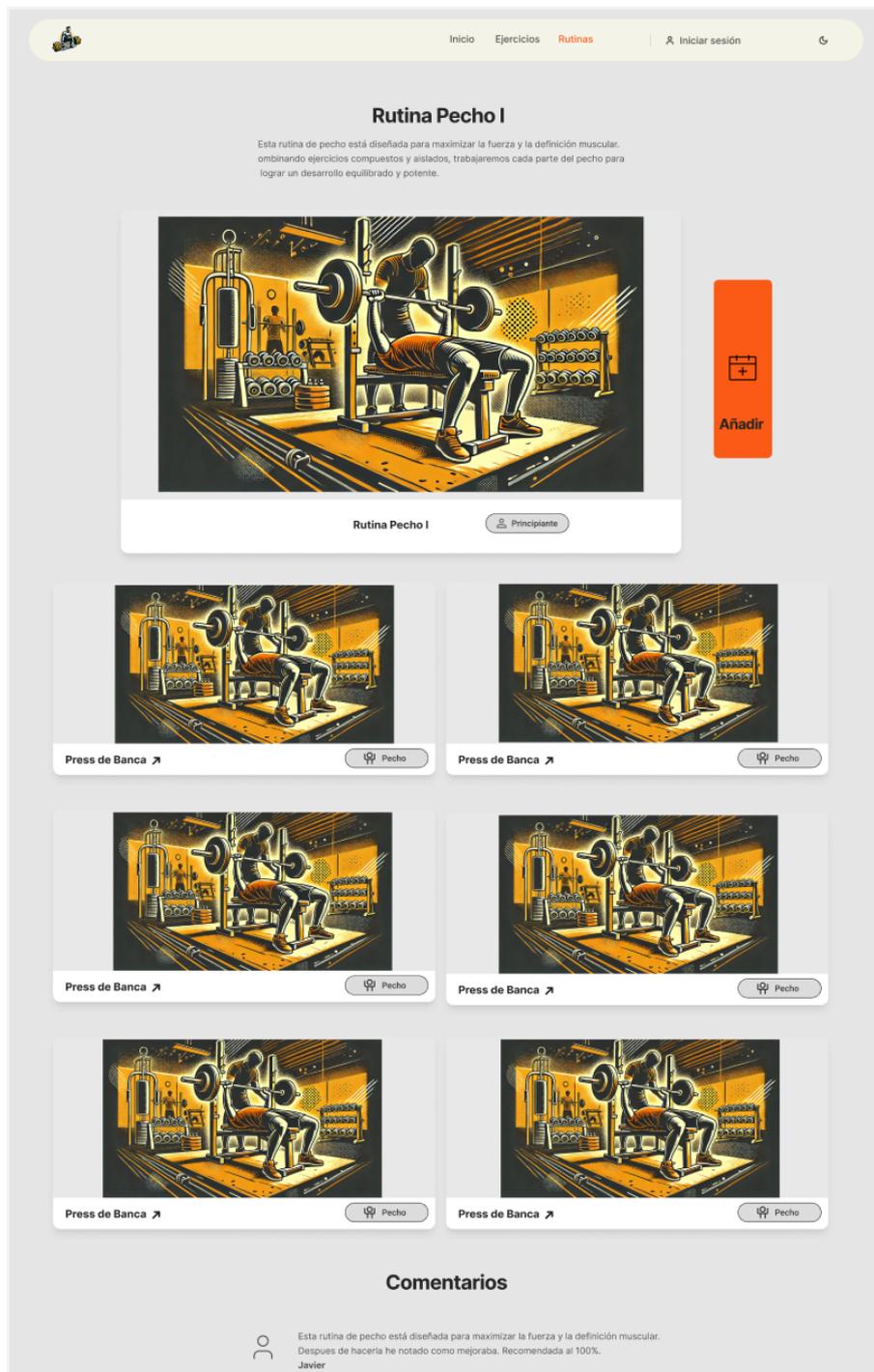


Figura 5.17: Pantalla de rutina concreta con comentarios en PC

Escenario 6. Analizar estadísticas

Después de un mes de entrenamiento Esteban decide analizar como ha ido progresando con el ejercicio de Press de Banca (figuras 5.18 y 5.19).

Al acceder al perfil del usuario hay un apartado donde se muestran las estadísticas para cada ejercicio del usuario, permitiendo seleccionarlo, además de mostrando la consistencia que ha tenido al ejecutar rutinas.



Figura 5.18: Pantalla de mis estadísticas en PC

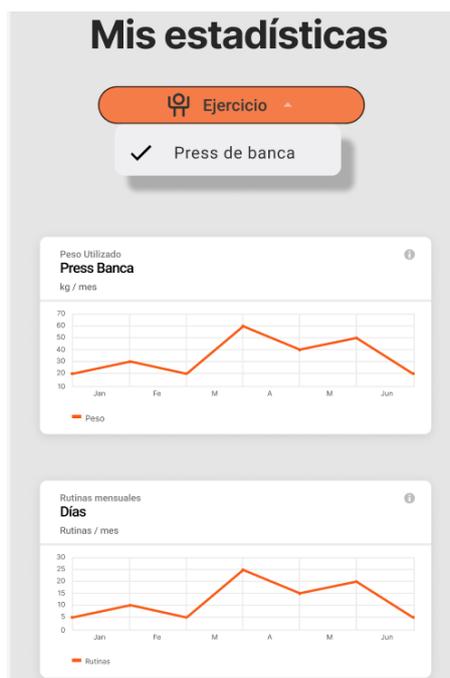


Figura 5.19: Pantalla de mis estadísticas en Móvil

Justificaciones de diseño

El uso de colores naranjas, amarillos y dorados en el diseño de la aplicación web puede ser justificado tanto desde un punto de vista psicológico como funcional, apoyándose en estudios sobre la percepción del color y en los resultados de la encuesta realizada entre los potenciales usuarios.

Según el artículo de La Vanguardia [12] sobre la psicología del color dorado, este color se asocia comúnmente con el éxito, la riqueza, la sabiduría y el prestigio. Estos atributos son ideales para una aplicación centrada en el fitness, ya que refuerzan el concepto de logro personal, motivación y superación, valores fundamentales para los usuarios que buscan mejorar su forma física y su salud. El color dorado, combinado con tonos naranjas y amarillos, que son conocidos por evocar sensaciones de energía, vitalidad, felicidad y optimismo, crea un ambiente positivo y motivador, que puede influir en la percepción y el comportamiento de los usuarios hacia su compromiso con la aplicación y sus metas de entrenamiento.

Además, la encuesta realizada muestra que una parte significativa de los usuarios valora un diseño visual moderno y atractivo a la par que minimalista y sencillo (véase el resultado de la encuesta en la figura A.14), lo cual se refuerza mediante el uso de estos colores cálidos y vibrantes. Los tonos naranjas y amarillos, en particular, son colores que destacan visualmente y pueden ayudar a captar la atención de los usuarios, guiándolos intuitivamente a través de las funcionalidades de la aplicación, lo que es crucial para mantener el interés y la satisfacción del usuario. Asimismo, estos colores son accesibles y legibles en distintas plataformas, lo que facilita el uso de la aplicación en diferentes dispositivos, como se sugiere por la alta frecuencia de uso de dispositivos móviles entre los encuestados (véase el resultado de la encuesta en la figura A.12).

CAPÍTULO 6

Desarrollo de la solución

6.1 Arquitectura

La arquitectura de la aplicación SiuGym ha sido diseñada para ser modular, escalable y mantener una clara separación de responsabilidades, garantizando tanto la eficiencia en el desarrollo como la mantenibilidad a largo plazo. La arquitectura se organiza en varias capas y componentes clave, que incluyen la aplicación web, el servidor backend (véase la definición en el glosario Backend), los servicios web externos, la base de datos, y un servicio de almacenamiento en la nube, todos los cuales se comunican entre sí mediante conexiones HTTP y servicios RESTful (véase la figura 6.1).

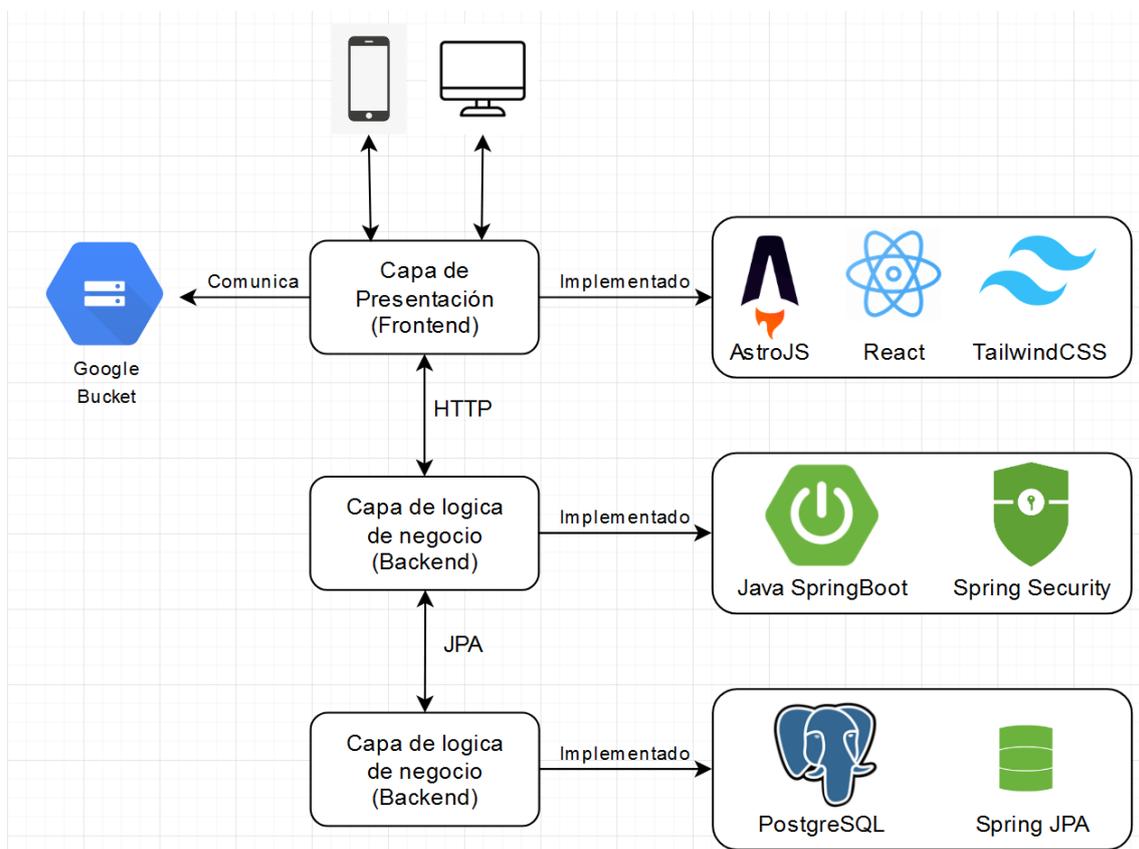


Figura 6.1: Arquitectura de SiuGym

En la parte superior de la arquitectura, se encuentra la capa de presentación, que incluye la interfaz de usuario (UI) tanto para dispositivos móviles como para navegadores web en ordenadores. Esta capa es responsable de la interacción directa con los usuarios, ofreciendo una experiencia fluida e intuitiva. La UI se desarrolla utilizando tecnologías modernas como HTML5, CSS3, y JavaScript, apoyadas por frameworks (véase la definición en el glosario Framework) como React.js para manejar la interfaz de manera dinámica. La comunicación con el servidor se realiza principalmente a través de solicitudes HTTP usando el protocolo REST (conceptos definidos en el glosario HTTP (Hypertext Transfer Protocol)), enviando y recibiendo datos en formato JSON. Este formato ligero y de fácil manejo es ideal para la transferencia eficiente de datos entre la UI y el backend.

Justo debajo de la capa de presentación se encuentra la capa de lógica de negocio, implementada en Java utilizando el framework Spring Boot. Esta capa es el núcleo de la aplicación, encargándose de procesar las reglas de negocio, gestionar las operaciones principales como la gestión de usuarios, rutinas de entrenamiento, y el seguimiento del progreso. Aquí, la arquitectura de servicios permite la integración de nuevas funcionalidades y la reutilización de componentes, haciendo que el sistema sea altamente adaptable a futuros requerimientos. Las conexiones entre esta capa y la capa de datos o con servicios web externos se realizan también a través de HTTP, utilizando REST.

Un aspecto clave de esta arquitectura es la integración con un bucket de Google Cloud Storage para el almacenamiento de imágenes y videos de los ejercicios. Esta solución en la nube permite almacenar de manera segura y eficiente los archivos multimedia que enriquecen la experiencia de los usuarios, como imágenes descriptivas y tutoriales en vídeo de cada ejercicio. El uso del bucket permite a la aplicación escalar en almacenamiento sin afectar el rendimiento del sistema y asegura la disponibilidad de los recursos multimedia con alta disponibilidad y redundancia. El acceso a estos recursos desde la UI se realiza mediante URLs generadas dinámicamente, permitiendo que las imágenes y vídeos se carguen rápidamente, mejorando la experiencia del usuario.

Finalmente, en la base de la arquitectura se encuentra la capa de acceso a datos, que interactúa con la base de datos. Esta capa utiliza JPA (Java Persistence API) junto con Hibernate para manejar las entidades y la persistencia de datos. El diseño de la base de datos sigue un modelo relacional, que como se ha comentado en el apartado 5.2 utiliza PostgreSQL, reflejando las relaciones entre usuarios, ejercicios, rutinas y registros de entrenamiento. Los datos se almacenan de manera estructurada y se garantiza la integridad y consistencia de la información.

Esta estructura en capas, la integración de almacenamiento en la nube y la capacidad de escalar fácilmente permiten que SiuGym sea una aplicación robusta, fácil de mantener y adaptable a los cambios, manteniendo al mismo tiempo una comunicación fluida y segura entre todos los elementos que componen el sistema.

6.2 Contexto tecnológico

A lo largo del desarrollo de este proyecto, he utilizado un conjunto diverso de herramientas y lenguajes de programación que me han permitido construir una aplicación robusta y eficiente. A continuación, presento brevemente cada una de estas tecnologías, destacando su relevancia en el contexto de este trabajo (véase la figura 6.5).

Cada una de estas herramientas y tecnologías ha sido clave en el desarrollo del proyecto, permitiendo la creación de una aplicación sólida y bien estructurada. Han contribuido de manera significativa a asegurar que la aplicación no solo funcione de manera eficiente, sino que también esté alineada con las mejores prácticas en el desarrollo de soft-

ware. Cada decisión tecnológica ha sido tomada con el objetivo de garantizar la estabilidad, seguridad y facilidad de mantenimiento del sistema, asegurando que el resultado final no solo cumpla con los requisitos planteados, sino que también esté preparado para futuras expansiones y mejoras.

6.2.1. IntelliJIdea

IntelliJ IDEA ha sido el entorno de desarrollo integrado (IDE) elegido para el desarrollo del backend de la aplicación. Es ampliamente reconocido por su robustez y su capacidad para manejar proyectos complejos, especialmente aquellos basados en Java. IntelliJ ofrece una integración profunda con el ecosistema de Spring, lo que facilita la creación, gestión y depuración de aplicaciones Spring Boot. Además, su soporte para herramientas como Maven, junto con características como el autocompletado inteligente, análisis de código en tiempo real, y herramientas avanzadas de refactorización, han acelerado significativamente el proceso de desarrollo, permitiéndome escribir código de manera más rápida y con menos errores.

6.2.2. Visual Studio Code

Visual Studio Code ha sido la elección escogida para el desarrollo del frontend (véase la definición en el glosario Frontend). Este editor de código, que es ligero pero increíblemente potente, ofrece una extensa colección de extensiones que han sido clave para trabajar con tecnologías como React, AstroJS, y Tailwind CSS. Su flexibilidad y capacidad para adaptarse a diferentes flujos de trabajo, junto con su integración con control de versiones como Git, han permitido un desarrollo ágil y organizado, que sumado a la gran capacidad de personalización que tiene han hecho de esta herramienta una gran opción.

6.2.3. Google Cloud Buckets

Google Cloud Buckets ha sido una solución esencial para el almacenamiento y gestión de recursos multimedia en la aplicación, como imágenes y vídeos relacionados con los ejercicios. Esta plataforma de almacenamiento en la nube ofrece gran escalabilidad, asegurando que los archivos se almacenen de forma segura y estén disponibles de manera eficiente para los usuarios finales.



The screenshot shows a Google Cloud Storage bucket named 'slugym-tfg'. The left sidebar displays a folder structure with 'imagenes-Ejercicio/' and 'imagenes-Musculo/' expanded to show sub-folders 'Videos/'. The main area shows a list of files with columns for Name, Size, Type, Creation Date, Storage Class, Last Modified, and Public Access. All files are of type 'image/webp' and have 'Publico para Internet' access.

Nombre	Tamaño	Tipo	Fecha de creación	Clase de almacenamiento	Última modificación	Acceso público
Curl-Biceps-Con-Barra.webp	17.9 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:38:05	▲ Público para Internet
Dominada.webp	17.2 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:38:23	▲ Público para Internet
Elevacion-Lateral.webp	17.1 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:40:45	▲ Público para Internet
Fondo-Paralela.webp	17.5 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:41:12	▲ Público para Internet
Musculos-Peso-Muerto.webp	17.7 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:41:29	▲ Público para Internet
Musculos-Press-Banca.webp	17.9 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:41:57	▲ Público para Internet
Musculos-Sentadilla-Con-Barra.w...	18.4 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:42:13	▲ Público para Internet
Press-Militar.webp	17.2 KB	image/webp	3 ago 2024 20:33:32	Standard	3 ago 2024 20:42:31	▲ Público para Internet
Talones-levantados.webp	18.1 KB	image/webp	3 ago 2024 20:33:33	Standard	3 ago 2024 20:42:46	▲ Público para Internet

Figura 6.2: Estructura de carpetas en el Bucket de Google

Como se puede ver en las figuras 6.2 y 6.3, las URLs correspondientes a las imágenes y vídeos de los ejercicios se han almacenado directamente en el bucket de Google, lo que elimina la necesidad de manejar archivos multimedia pesados dentro del sistema principal (véase figura 6.4). Esto no solo reduce la carga en el servidor de la aplicación, sino que también mejora el rendimiento general de la aplicación al permitir que los recursos

se sirvan de manera rápida y eficiente a través de la infraestructura de Google Cloud. Destacar que las URLs se han puesto públicas debido a que no es necesario tener cuenta de usuario en la aplicación para poder ver las imágenes y vídeos de los ejercicios.

Depósitos > siugym-tfg > Imagenes-Musculo > Fondo-Paralela.webp

OBJETO PUBLICADO HISTORIAL DE VERSIONES

DESCARGAR EDITAR METADATOS EDITAR ACCESO BORRAR

Descripción general

Tipo	image/webp
Tamaño	17.5 KB
Fecha y hora de creación	3 ago 2024 20:33:32
Última modificación	3 ago 2024 20:41:12
Clase de almacenamiento	Standard
Tiempo personalizado	—
URL pública	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Fondo-Paralela.webp
URL autenticada	https://storage.cloud.google.com/siugym-tfg/Imagenes-Musculo/Fondo-Paralela.webp?authuser=3
URI de gsutil	gs://siugym-tfg/Imagenes-Musculo/Fondo-Paralela.webp

Permisos

Acceso público Público para Internet

Protección

Historial de versiones	—
Fecha y hora de vencimiento de la retención	Ninguno
Período de retención del objeto	Ninguno
Período de retención del bucket	Ninguno
Estado de conservación	Ninguno
Tipo de encriptación	Administrada por Google

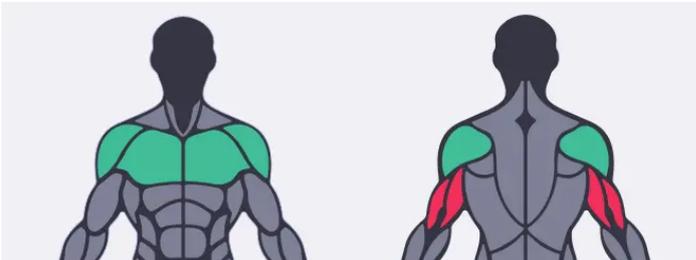
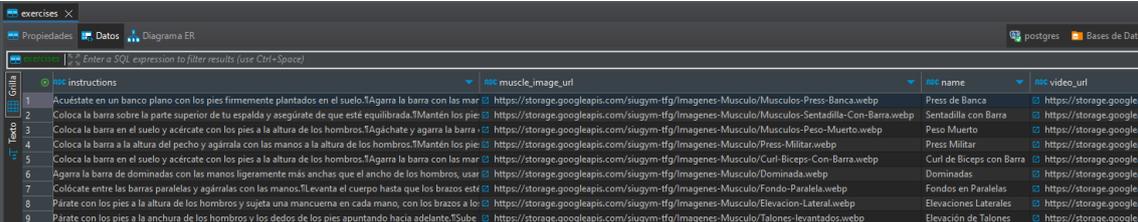


Figura 6.3: Detalle de una imagen en el Bucket de Google



id	instructions	muscle_image_url	video_name	video_url
1	Acuéstate en un banco plano con los pies firmemente plantados en el suelo. Agarra la barra con las manos...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Musculos-Press-Banca.webp	Press de Banca	https://storage.googleapis.com/siugym-tfg/Videos-Press-Banca.mp4
2	Coloca la barra sobre la parte superior de tu espalda y asegúrate de que esté equilibrada. Mantén los pies...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Musculos-Sentadilla-Con-Barra.webp	Sentadilla con Barra	https://storage.googleapis.com/siugym-tfg/Videos-Sentadilla-Con-Barra.mp4
3	Coloca la barra en el suelo y recástrate con los pies a la altura de los hombros. Apóyate y agarra la barra...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Musculos-Peso-Muerto.webp	Peso Muerto	https://storage.googleapis.com/siugym-tfg/Videos-Peso-Muerto.mp4
4	Coloca la barra a la altura del pecho y apóyala con las manos a la altura de los hombros. Mantén los pies...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Press-Militar.webp	Press Militar	https://storage.googleapis.com/siugym-tfg/Videos-Press-Militar.mp4
5	Coloca la barra en el suelo y acércate con los pies a la altura de los hombros. Agarra la barra con las manos...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Curl-Biceps-Con-Barra.webp	Curl de Biceps con Barra	https://storage.googleapis.com/siugym-tfg/Videos-Curl-Biceps-Con-Barra.mp4
6	Agarra la barra de dominadas con las manos ligeramente más anchas que el ancho de los hombros, usa...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Dominada.webp	Dominadas	https://storage.googleapis.com/siugym-tfg/Videos-Dominadas.mp4
7	Colócate entre las barras paralelas y agárralas con las manos. Levanta el cuerpo hasta que los brazos est...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Fondo-Paralela.webp	Fondos en Paralelas	https://storage.googleapis.com/siugym-tfg/Videos-Fondo-Paralela.mp4
8	Parate con los pies a la altura de los hombros y sujeta una mancuerna en cada mano, con los brazos a lo...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Elevacion-Lateral.webp	Elevaciones Laterales	https://storage.googleapis.com/siugym-tfg/Videos-Elevacion-Lateral.mp4
9	Parate con los pies a la anchura de los hombros y los dedos de los pies apuntando hacia adelante. Sube...	https://storage.googleapis.com/siugym-tfg/Imagenes-Musculo/Talones-levantados.webp	Elevación de Talones	https://storage.googleapis.com/siugym-tfg/Videos-Talones-levantados.mp4

Figura 6.4: Tabla exercise con las URLs a las imágenes guardadas en el Bucket

6.2.4. Java Spring, SpringBoot

El framework Spring ha sido la base sobre la que se ha construido toda la lógica de negocio de la aplicación. Spring Boot ha simplificado la configuración y el despliegue de la aplicación, permitiendo crear aplicaciones independientes listas para producción con muy poca configuración inicial. Utilizando Spring Boot, he podido desarrollar servicios RESTful de manera eficiente, gestionando la inyección de dependencias, la configuración

del servidor y la conexión a la base de datos de forma fluida. El ecosistema de Spring, con su modularidad y extensibilidad, ha sido fundamental para mantener un código limpio, escalable y fácil de mantener.

6.2.5. Spring Security

Spring Security ha jugado un papel crucial en la protección de la aplicación, proporcionando una sólida infraestructura de autenticación y autorización. Con esta herramienta, he implementado controles de acceso detallados, asegurando que solo los usuarios autorizados puedan acceder a ciertas funcionalidades de la aplicación. Además, Spring Security ofrece una amplia gama de características adicionales, como la protección contra ataques de tipo CSRF (Cross-Site Request Forgery) y la encriptación de contraseñas, lo que ha permitido construir una aplicación que cumple con altos estándares de seguridad. En el siguiente apartado se verá la implementación concreta Spring Security en el proyecto.

6.2.6. Spring JPA y PostgreSQL

Para la persistencia de datos, he utilizado Spring JPA junto con PostgreSQL. Spring JPA proporciona una abstracción sobre la base de datos que permite trabajar con objetos en lugar de tablas y registros, facilitando el desarrollo y mantenimiento del código. PostgreSQL, una base de datos relacional avanzada, ha sido elegida por su robustez, rendimiento y características avanzadas, como el soporte para datos JSON y transacciones complejas. Juntos, Spring JPA y PostgreSQL han permitido implementar un acceso a datos eficiente y escalable, asegurando la integridad y consistencia de los datos a lo largo de la aplicación.

6.2.7. Swagger y SpringDoc

Swagger y Springdoc han sido herramientas clave para la documentación de la API. Swagger facilita la creación de una documentación interactiva y visual de la API, lo que permite a los desarrolladores explorar y probar los endpoints (véase la definición en el glosario Endpoint) de manera sencilla. Springdoc, en particular, ha sido utilizado para generar automáticamente la documentación de la API REST basada en Spring Boot, asegurando que la documentación esté siempre actualizada con el código fuente. Esto no solo mejora la experiencia de desarrollo, sino que también facilita la integración con otros sistemas y servicios externos que podrían interactuar con la API.

6.2.8. Spring Boot Test y SonarLint

Spring Boot Test y SonarLint son dos herramientas fundamentales que he integrado en el desarrollo del proyecto para asegurar la calidad y fiabilidad del código.

Spring Boot Test es una poderosa herramienta que permite realizar pruebas unitarias y de integración de manera sencilla y eficiente dentro de aplicaciones Spring Boot. Gracias a su integración con el ecosistema de Spring, facilita la simulación de diferentes contextos de la aplicación, permitiendo verificar que cada componente funcione correctamente tanto de forma aislada como en conjunto con otros.

Por otro lado, SonarLint es una herramienta de análisis estático de código que me ha permitido mantener un alto estándar de calidad en el desarrollo del proyecto. Integrado

directamente en el entorno de desarrollo, SonarLint analiza el código en tiempo real, detectando posibles vulnerabilidades, malas prácticas y errores de estilo. Al usar SonarLint, he podido identificar y corregir problemas en el código antes de que lleguen a afectar al funcionamiento de la aplicación, asegurando un código limpio, eficiente y seguro.

6.2.9. React

En el frontend, React ha sido la biblioteca central para construir interfaces de usuario interactivas y dinámicas. Su enfoque basado en componentes ha permitido construir una UI modular y reutilizable, facilitando el mantenimiento y la extensión de la aplicación a medida que crece. React ofrece una experiencia de usuario fluida y rápida, permitiendo la actualización en tiempo real de los datos en la interfaz sin necesidad de recargar la página. Esta tecnología ha sido clave para ofrecer una experiencia de usuario moderna y reactiva, acorde con las expectativas de los usuarios actuales.

6.2.10. AstroJS

AstroJS ha sido utilizado para construir páginas web estáticas que son rápidas y optimizadas para el rendimiento. Esta herramienta permite combinar múltiples frameworks de frontend en un mismo proyecto, aprovechando lo mejor de cada uno. En este proyecto, AstroJS ha permitido la integración de React junto con otros componentes estáticos, resultando en un sitio web que no solo es rápido, sino que también es fácil de gestionar y escalar. AstroJS ha sido crucial para mejorar los tiempos de carga de la aplicación y ofrecer una experiencia de usuario más ágil.

6.2.11. TailwindCSS

Para el diseño y la personalización del estilo, he utilizado Tailwind CSS, un framework CSS altamente personalizable que permite crear interfaces de usuario modernas y atractivas de manera rápida y eficiente. Tailwind CSS se basa en clases utilitarias, lo que facilita el diseño directo en el HTML sin necesidad de escribir CSS personalizado. Esta metodología ha acelerado el desarrollo del frontend, permitiendo aplicar estilos coherentes en toda la aplicación, y ha asegurado que el diseño sea responsivo y adaptable a diferentes dispositivos y tamaños de pantalla.

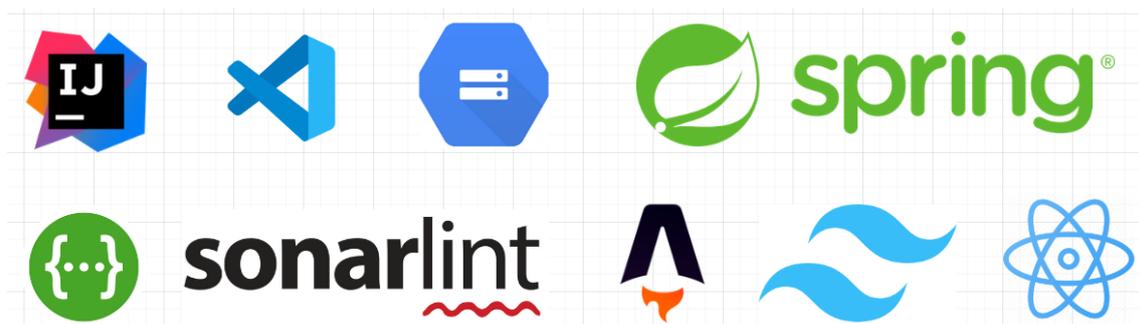


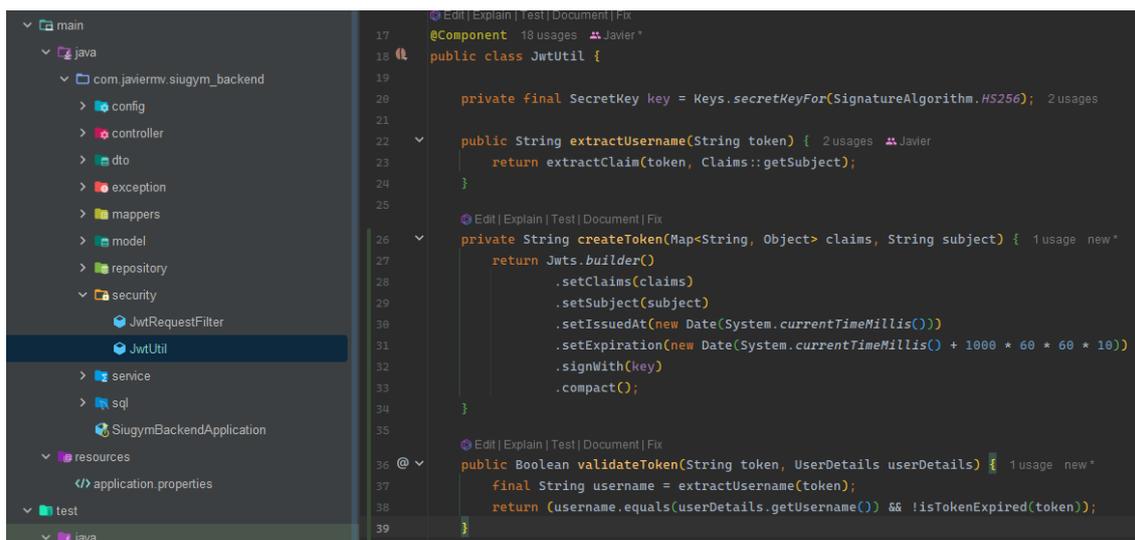
Figura 6.5: Herramientas utilizadas en el desarrollo del proyecto

6.3 Ejemplos de código

Para ilustrar de manera clara cómo se han implementado los diferentes componentes de la aplicación, presentaré una serie de ejemplos que reflejan las principales funcionalidades del sistema. Estos ejemplos se centran en la gestión de la seguridad mediante JWT (véase la definición del concepto en el glosario JWT (JSON Web Token)), la protección de las APIs con Spring Security, el flujo típico de un controlador, servicio, mapper y repositorio, los tests que aseguran la robustez del backend, y finalmente, la documentación de la API utilizando Swagger, los componentes astro utilizados, el uso de tailwindCss y de los componentes de react.

6.3.1. Implementación de JWT para la Autenticación

En el sistema SiuGym, la autenticación se gestiona mediante JSON Web Tokens (JWT). Este enfoque garantiza que solo los usuarios autenticados puedan acceder a las funcionalidades protegidas de la aplicación. La clase `JwtUtil` es fundamental en este proceso, encargándose de la generación y validación de los tokens. A continuación, se muestra un fragmento de código (figura 6.6) clave en esta implementación:



```
17 @Component 10 usages Javier*
18 public class JwtUtil {
19
20     private final SecretKey key = Keys.secretKeyFor(SignatureAlgorithm.HS256); 2 usages
21
22     public String extractUsername(String token) { 2 usages Javier
23         return extractClaim(token, Claims::getSubject);
24     }
25
26     private String createToken(Map<String, Object> claims, String subject) { 1 usage new*
27         return Jwts.builder()
28             .setClaims(claims)
29             .setSubject(subject)
30             .setIssuedAt(new Date(System.currentTimeMillis()))
31             .setExpiration(new Date(System.currentTimeMillis() + 1000 * 60 * 60 * 10))
32             .signWith(key)
33             .compact();
34     }
35
36     public Boolean validateToken(String token, UserDetails userDetails) { 1 usage new*
37         final String username = extractUsername(token);
38         return (username.equals(userDetails.getUsername()) && !isTokenExpired(token));
39     }
```

Figura 6.6: Código más relevante de la clase `JwtUtil`

Este ejemplo muestra cómo se manejan los tokens JWT, desde su creación hasta su validación. La seguridad del sistema se refuerza asegurando que solo los usuarios con un token válido puedan interactuar con las APIs protegidas.

6.3.2. Seguridad de las APIs con Spring Security

El acceso a las diferentes rutas del sistema está controlado mediante Spring Security. Esta configuración asegura que solo los usuarios autenticados puedan realizar determinadas acciones dentro de la aplicación. La siguiente configuración de seguridad demuestra cómo se gestionan estas restricciones (figura 6.7)

```
19 @Configuration
20 @EnableWebSecurity
21 public class SecurityConfig {
22     private final UserDetailsService userDetailsService;
23     private final JwtRequestFilter jwtRequestFilter;
24
25     public SecurityConfig(UserDetailsService userDetailsService, JwtRequestFilter jwtRequestFilter) {
26         this.userDetailsService = userDetailsService;
27         this.jwtRequestFilter = jwtRequestFilter;
28     }
29
30     @Bean
31     public SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {
32         http
33             .csrf(AbstractHttpConfigurer::disable)
34             .authorizeHttpRequests(auth -> auth
35                 .requestMatchers("/api/auth/**").permitAll()
36                 .requestMatchers("/swagger-ui/**", "/v3/api-docs/**").permitAll()
37                 .requestMatchers("/api/exerciseg/**").permitAll()
38                 .requestMatchers(HttpMethod.GET, "/api/routines", "/api/routines/{id}", "/api/routines/{routineId}/comments").permitAll()
39                 .requestMatchers("/api/users").permitAll()
40                 .requestMatchers("/api/muscles").permitAll()
41                 .requestMatchers("/api/categories").permitAll()
42                 .anyRequest().authenticated()
43             )
44             .sessionManagement(session -> session.sessionCreationPolicy(SessionCreationPolicy.STATELESS))
45             .addFilterBefore(jwtRequestFilter, UsernamePasswordAuthenticationFilter.class);
46
47         return http.build();
48     }
49 }
```

Figura 6.7: Configuración de seguridad de los endpoints

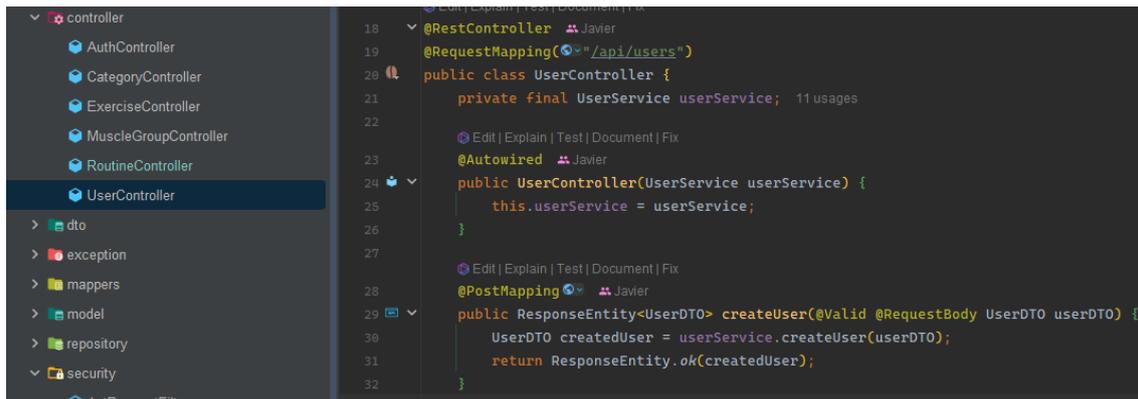
Este fragmento de código garantiza que todas las rutas, excepto las relacionadas con la autenticación y las públicas, están protegidas y requieren un token JWT válido para ser accedidas. De este modo, se protege la integridad y la seguridad de la información manejada por la aplicación.

6.3.3. Flujo de Trabajo: Controlador, Servicio, Mapper y Repositorio

El flujo de datos y la interacción entre las capas de la aplicación son fundamentales para entender cómo se maneja la lógica de negocio y cómo se accede a los datos en el proyecto. A continuación, describiré de manera detallada cómo funcionan y se interrelacionan el controlador, el servicio, el mapper y el repositorio, siguiendo el flujo típico en una aplicación Spring Boot, como la desarrollada para este proyecto.

El controlador es el punto de entrada de la aplicación para las solicitudes HTTP. Es el encargado de recibir las peticiones que realizan los usuarios a través de la interfaz web o de otras aplicaciones que interactúan con la API. En el contexto de Spring Boot, los controladores están anotados con `@RestController` y manejan las rutas específicas que se definen mediante anotaciones como `@GetMapping`, `@PostMapping`, `@PutMapping`, y `@DeleteMapping`.

Por ejemplo, hay un controlador que maneja las operaciones relacionadas con los usuarios. Este controlador tiene un método para crear un nuevo usuario (véase la figura 6.8), que recibe una solicitud POST con los datos del usuario. Este método delega la lógica de negocio al servicio correspondiente. Aquí es donde el controlador actúa como intermediario, gestionando la entrada del usuario y entregando la respuesta adecuada.



```

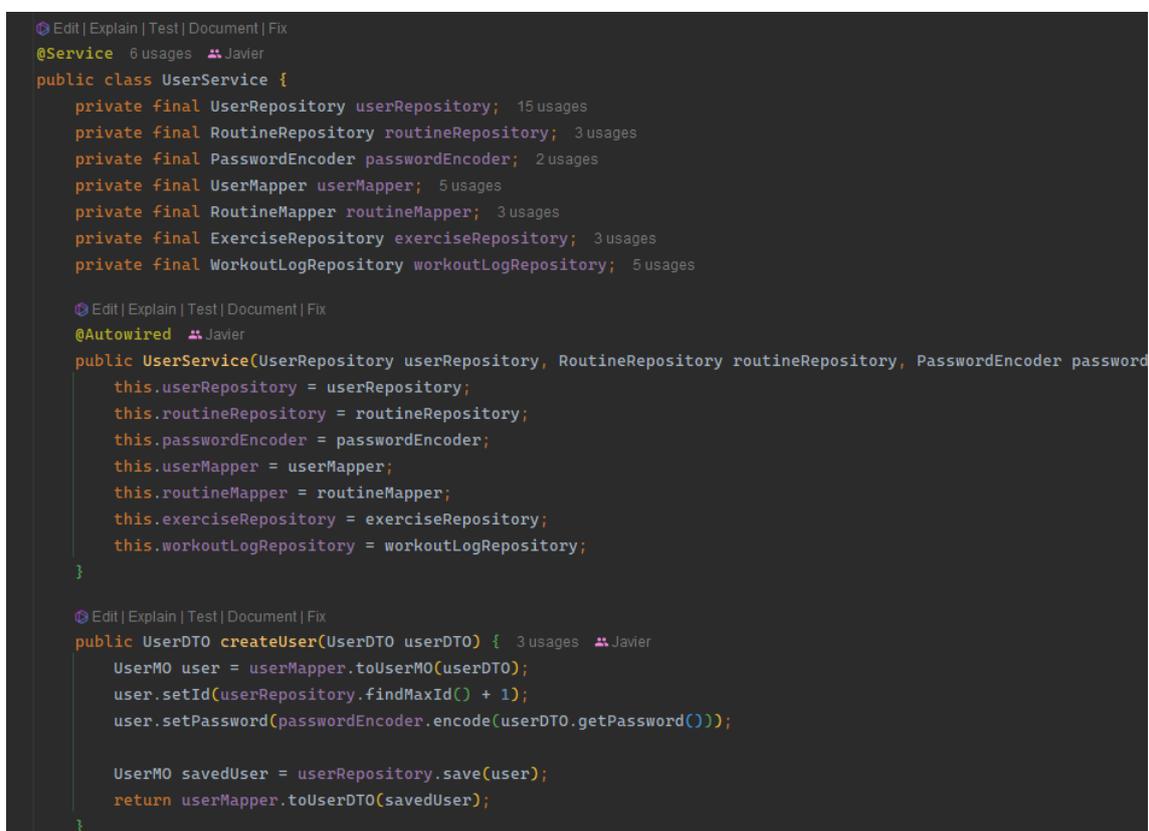
18  @RestController
19  @RequestMapping("/api/users")
20  public class UserController {
21      private final UserService userService;
22
23      @Autowired
24      public UserController(UserService userService) {
25          this.userService = userService;
26      }
27
28      @PostMapping
29      public ResponseEntity<UserDTO> createUser(@Valid @RequestBody UserDTO userDTO) {
30          UserDTO createdUser = userService.createUser(userDTO);
31          return ResponseEntity.ok(createdUser);
32      }

```

Figura 6.8: Función para crear usuario en el controlador

La capa de servicio contiene la lógica de negocio de la aplicación. Es aquí donde se implementan las reglas y procesos que gobiernan las operaciones del sistema. El servicio interactúa con el repositorio para acceder a los datos y con el mapper para transformar los datos entre las diferentes capas de la aplicación.

Siguiendo el ejemplo anterior, el servicio de usuario es la responsable de manejar la creación de un nuevo usuario. El método createUser (véase figura 6.9) es responsable de manejar la lógica para la creación de un nuevo usuario en el sistema. Este método recibe un objeto UserDTO desde el controlador, que contiene los datos que el usuario ha proporcionado. La primera tarea del servicio es convertir este DTO en una entidad de dominio, UserMO, utilizando el userMapper. Esta conversión es crucial para asegurar que los datos estén en el formato adecuado para ser procesados y almacenados en la base de datos.



```

@Service
public class UserService {
    private final UserRepository userRepository;
    private final RoutineRepository routineRepository;
    private final PasswordEncoder passwordEncoder;
    private final UserMapper userMapper;
    private final RoutineMapper routineMapper;
    private final ExerciseRepository exerciseRepository;
    private final WorkoutLogRepository workoutLogRepository;

    @Autowired
    public UserService(UserRepository userRepository, RoutineRepository routineRepository, PasswordEncoder passwordEncoder, UserMapper userMapper, RoutineMapper routineMapper, ExerciseRepository exerciseRepository, WorkoutLogRepository workoutLogRepository) {
        this.userRepository = userRepository;
        this.routineRepository = routineRepository;
        this.passwordEncoder = passwordEncoder;
        this.userMapper = userMapper;
        this.routineMapper = routineMapper;
        this.exerciseRepository = exerciseRepository;
        this.workoutLogRepository = workoutLogRepository;
    }

    public UserDTO createUser(UserDTO userDTO) {
        UserMO user = userMapper.toUserMO(userDTO);
        user.setId(userRepository.findMaxId() + 1);
        user.setPassword(passwordEncoder.encode(userDTO.getPassword()));

        UserMO savedUser = userRepository.save(user);
        return userMapper.toUserDTO(savedUser);
    }
}

```

Figura 6.9: Función para crear usuario en el servicio

Una vez que se tiene la entidad UserMO, el servicio asigna un nuevo ID al usuario, asegurando que cada usuario en el sistema tenga un identificador único. Luego, el servicio se encarga de encriptar la contraseña proporcionada por el usuario utilizando passwordEncoder, garantizando así que las credenciales se almacenen de manera segura.

Finalmente, el servicio delega la tarea de persistir la entidad en la base de datos al repositorio userRepository. Después de que el usuario ha sido guardado en la base de datos, el servicio convierte la entidad de nuevo a un UserDTO utilizando el userMapper, y devuelve este DTO al controlador, que a su vez, enviará la respuesta al cliente. Este proceso asegura que la lógica de negocio y la manipulación de datos estén claramente separadas, lo que facilita la mantenibilidad y escalabilidad del sistema.

El mapper es una pieza que facilita la transformación de los datos entre diferentes capas de la aplicación, especialmente entre las entidades de dominio y los objetos de transferencia de datos (DTOs). En proyectos basados en Spring Boot, es común utilizar un mapper para convertir entre entidades y DTOs, asegurando que los datos se manipulen y presenten en el formato correcto en cada capa.

Por ejemplo, cuando el servicio crea un nuevo usuario, el mapper convierte el DTO recibido desde el controlador en una entidad de dominio que puede ser manejada por el repositorio (véase la figura 6.10). Del mismo modo, cuando el servicio necesita devolver información al controlador, el mapper convierte la entidad en un DTO (véase la figura 6.11) que se puede enviar como respuesta al cliente. Este proceso de mapeo es crucial para mantener la integridad de los datos y para asegurar que las distintas capas de la aplicación se comuniquen de manera eficiente.

```
public UserMO toUserMO(UserDTO userDTO) { 2 usages  🧑 Javier
    return new UserMO(
        id: null, // ID se genera automáticamente en la base de datos
        userDTO.getUsername(),
        userDTO.getEmail(),
        userDTO.getPassword(),
        userDTO.getFirstName(),
        userDTO.getLastName(),
        userDTO.getBirthDate(),
        routines: null, // Routines se manejan por otro proceso
        null, // WorkoutLogs se manejan por otro proceso
        null // Comments se manejan por otro proceso
    );
}
```

Figura 6.10: Función para transformar un UserDTO en un UserMO

```
@Component 5 usages  👤 Javier
public class UserMapper {

    Edit | Explain | Test | Document | Fix
    public UserDTO toUserDTO(UserMO user) { 6 usages  👤 Javier
        return new UserDTO(
            user.getUsername(),
            user.getEmail(),
            password: null, // No devolvemos la contraseña en el DTO
            user.getFirstName(),
            user.getLastName(),
            user.getBirthDate()
        );
    }
}
```

Figura 6.11: Función para transformar un UserMO en un UserDTO

El repositorio es la capa que maneja el acceso a la base de datos. En Spring Boot, los repositorios son interfaces que extienden `JpaRepository` o `CrudRepository`, lo que permite interactuar con la base de datos de manera abstracta y declarativa. El repositorio se encarga de las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre las entidades, utilizando consultas SQL generadas automáticamente o personalizadas según sea necesario.

Siguiendo con el ejemplo del usuario, el repositorio de usuarios sería el encargado de persistir el nuevo usuario en la base de datos, buscar usuarios existentes, o eliminar usuarios cuando sea necesario (véase la figura 6.12). Esta capa asegura que la lógica de acceso a los datos esté separada de la lógica de negocio, facilitando así la mantenibilidad y escalabilidad del sistema.

```
@Repository 11 usages  👤 Javier +1
public interface UserRepository extends JpaRepository<UserMO, Long> {
    Optional<UserMO> findByUsername(String username); 21 usages  👤 Javier

    @Query(value = "SELECT MAX(u.id) FROM UserMO u") 1 usage  👤 Javier
    Long findMaxId();
}
```

Figura 6.12: Clase repositorio para los usuarios

6.3.4. Implementación de Tests Unitarios

En el proyecto SiuGym, la implementación de tests unitarios ha sido una parte fundamental para asegurar que cada componente del sistema funcione como se espera. A través de los tests, se valida la correcta interacción entre las diferentes capas, como el controlador, servicio, y los mappers, asegurando que se mantenga la integridad de los datos y la lógica de negocio en todas las operaciones.

El archivo `UserControllerTest` es un ejemplo representativo de cómo se han estructurado los tests para los controladores en el proyecto. En este caso, se utilizan las anotacio-

nes `@WebMvcTest` y `@WithMockUser` para configurar un entorno de prueba adecuado para el controlador `UserController`.

Por ejemplo, en el test `testCreateUser`, se simula la creación de un nuevo usuario mediante una llamada POST al endpoint `/api/users` (figura 6.13). Aquí, se utiliza `MockMvc` para enviar una solicitud HTTP con un cuerpo JSON que contiene los datos del usuario. Antes de ejecutar la solicitud, se mockea el comportamiento del `UserService` para devolver un `UserDTO` preconfigurado cuando se llama al método `createUser`. Esto permite validar que la respuesta HTTP es la esperada, en este caso, se espera un error debido a una configuración específica de validación.

```
31  Edit | Explain | Test | Document | Fix
32  @WebMvcTest(UserController.class)  Javier
33  class UserControllerTest {
34
35      @Autowired
36      private MockMvc mockMvc;
37
38      @MockBean
39      private UserService userService;
40
41      @Mock no usages
42      private Authentication authentication;
43
44      @MockBean
45      private JwtUtil jwtUtil;
46
47      Edit | Explain | Test | Document | Fix
48      @BeforeEach  Javier
49      void setUp() {
50          Authentication auth = Mockito.mock(Authentication.class);
51          Mockito.when(auth.getName()).thenReturn("testUser");
52          SecurityContext securityContext = Mockito.mock(SecurityContext.class);
53          Mockito.when(securityContext.getAuthentication()).thenReturn(auth);
54          SecurityContextHolder.setContext(securityContext);
55      }
56
57      Edit | Explain | Test | Document | Fix
58      @Test  Javier
59      @WithMockUser
60      void testCreateUser() throws Exception {
61          UserDTO userDTO = new UserDTO();
62          userDTO.setUsername("testUser");
63
64          when(userService.createUser(any(UserDTO.class))).thenReturn(userDTO);
65
66          mockMvc.perform(post( urlTemplate: "/api/users")
67                      .contentType(MediaType.APPLICATION_JSON)
68                      .content("{\"username\":\"testUser\"}"))
69                      .andExpect(status().is4xxClientError());
70      }
71  }
```

Figura 6.13: Test para el endpoint de crear usuarios

En `UserServiceTest`, se validan las funcionalidades críticas del servicio que maneja la lógica de negocio de los usuarios. Por ejemplo, el método `testCreateUser` (figura 6.14) se encarga de verificar que el proceso de creación de un nuevo usuario se ejecute correctamente. Aquí, se mockea el comportamiento de varias dependencias como `UserRepository`, `PasswordEncoder`, y `UserMapper`. Se asegura que, una vez que el usuario ha sido creado, se almacene correctamente en la base de datos y que los datos se conviertan de vuelta a un `UserDTO` para ser enviados al controlador.

```
65  Edit | Explain | Test | Document | Fix
66  @Test  Javier
67  void testCreateUser() {
68      UserMO userMO = new UserMO();
69      userMO.setUsername("testUser");
70
71      UserDTO userDTO = new UserDTO();
72      userDTO.setUsername("testUser");
73
74      when(userMapper.toUserMO(any(UserDTO.class))).thenReturn(userMO);
75      when(passwordEncoder.encode(anyString())).thenReturn("encodedPassword");
76      when(userRepository.save(any(UserMO.class))).thenReturn(userMO);
77      when(userMapper.toUserDTO(any(UserMO.class))).thenReturn(userDTO);
78
79      UserDTO createdUser = userService.createUser(userDTO);
80
81      assertNotNull(createdUser);
82      assertEquals("testUser", createdUser.getUsername());
83      verify(userRepository, times(1)).save(any(UserMO.class));
84  }
```

Figura 6.14: Test para el servicio de crear usuarios

En el contexto de este proyecto, la ejecución de todos los tests se ha integrado de manera fluida en el ciclo de vida de desarrollo utilizando Maven. Esta herramienta de automatización no solo facilita la gestión de dependencias, sino que también permite ejecutar de manera automática todos los tests definidos en el proyecto, garantizando que cualquier cambio realizado en el código sea validado de inmediato. A través del comando `mvn test`, se inicia la ejecución de todos los tests unitarios y de integración, lo que asegura que cada componente del sistema funcione según lo esperado.

Al ejecutar los tests con Maven, se aprovechan las capacidades de esta herramienta para generar reportes detallados de los resultados, lo que permite identificar rápidamente cualquier fallo o regresión en el sistema (véase la figura 6.15). Estos reportes no solo muestran los tests que han fallado, sino que también proporcionan información útil sobre el tiempo de ejecución de cada uno, lo que contribuye a mejorar la eficiencia del proceso de desarrollo.

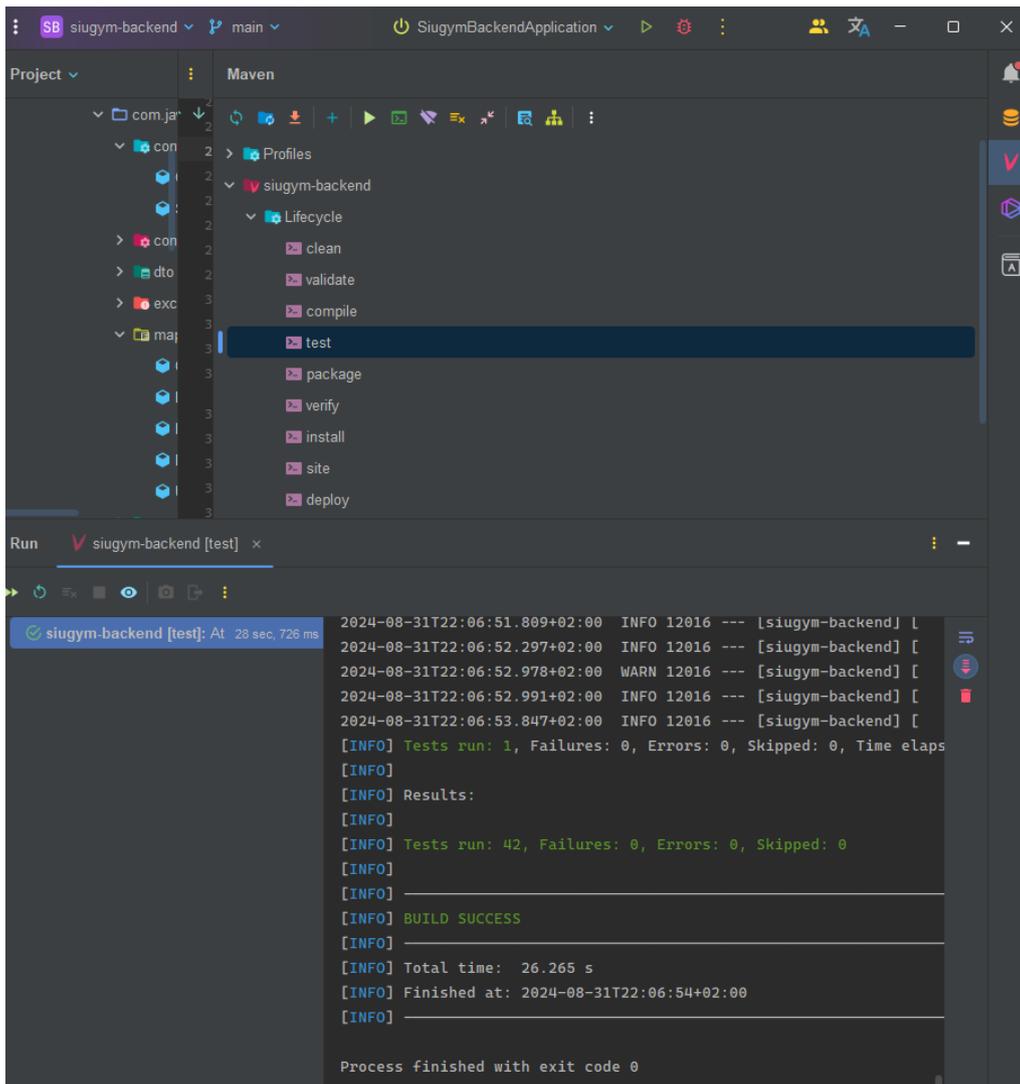


Figura 6.15: Ejecución de todos los tests con Maven

6.3.5. Swagger

Para asegurar que los desarrolladores y otros interesados puedan interactuar fácilmente con la API de SiuGym, se ha implementado Swagger. Esta herramienta genera automáticamente la documentación de la API y permite una navegación intuitiva a través de los endpoints disponibles (figura 6.16).

```
import io.swagger.v3.oas.models.Components;
import io.swagger.v3.oas.models.OpenAPI;
import io.swagger.v3.oas.models.info.Info;
import io.swagger.v3.oas.models.security.SecurityRequirement;
import io.swagger.v3.oas.models.security.SecurityScheme;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;

@Configuration
public class OpenApiConfig {
    @Bean
    public OpenAPI customOpenAPI() {
        return new OpenAPI()
            .info(new Info().title("SIUGym API").version("1.0").description("Documentación de la API de SiuGym"))
            .addSecurityItem(new SecurityRequirement().addList("bearerAuth"))
            .components(new Components().addSecuritySchemes("bearerAuth",
                new SecurityScheme().type(SecurityScheme.Type.HTTP).scheme("bearer").bearerFormat("JWT")));
    }
}
```

Figura 6.16: Configuración de swagger en SiuGym

Al acceder a la URL <http://localhost:8080/swagger-ui.html> (figura 6.17), los usuarios pueden visualizar y probar los endpoints de la API de manera interactiva. Esto no solo facilita el desarrollo y la integración, sino que también mejora la transparencia y la usabilidad del sistema.

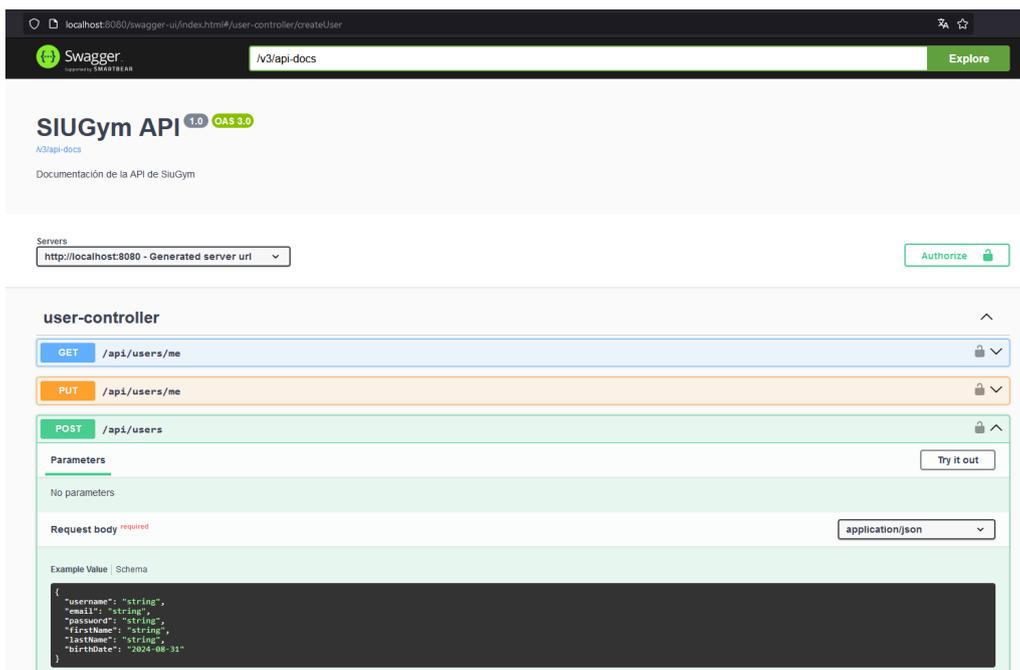


Figura 6.17: Swagger del API de SiuGym

6.3.6. Componente de Astro

Astro me permite estructurar de manera clara y modular el código, facilitando la reutilización de componentes y garantizando un mantenimiento más eficiente. Un ejemplo de ello es el siguiente componente (figura 6.18), en el cual he definido la estructura básica de una página, combinando la lógica de presentación con la personalización dinámica de su contenido. Este enfoque asegura que cada página de la aplicación mantenga una coherencia visual y funcional, mientras que su contenido específico puede ser fácilmente gestionado y modificado según sea necesario.

```
1 ---
2 // Importing necessary components
3 import Meta from "../components/Meta.astro";
4 import Navbar from "../components/sections/Navbar.astro";
5 import FooterSection from "../components/sections/FooterSection.astro";
6
7 // Setting expected props
8 const { title = "SiuGym", meta, structuredData } = Astro.props;
9
10 // Interface to type-check the properties
11 interface Props {
12   title?: string;
13   meta?: string;
14   structuredData?: object;
15 }
16 ---
17
18 <html lang="es" class="scrollbar-hide scroll-pt-16">
19   <head>
20     <!-- Adding metadata to the HTML document -->
21     <Meta meta={meta} structuredData={structuredData} />
22     <!-- Define the title of the page -->
23     <title>{title}</title>
24     <script is:inline>
25       // Script to handle dark mode. It will check if the theme is stored in lo
26       if (
27         localStorage.getItem("hs_theme") === "dark" ||
28         (!("hs_theme" in localStorage) &&
29           window.matchMedia("(prefers-color-scheme: dark)").matches)
30       ) {
31         document.documentElement.classList.add("dark");
32       } else {
33         document.documentElement.classList.remove("dark");
34       }
35     </script>
36     <script is:inline src="/scripts/vendor/lenis/lenis.js"></script>
37     <script is:inline>
38       // Script to handle Lenis library settings for smooth scrolling
39       const lenis = new Lenis({ smooth: true, smoothTouch: false });
40
41       function raf(time) {
42         lenis.raf(time);
43         requestAnimationFrame(raf);
44       }
45
46       requestAnimationFrame(raf);
47     </script>
48   </head>
```

Figura 6.18: Componente de Astro

En el fragmento de código de la figura, la sección superior se encarga de gestionar las importaciones necesarias, tales como Meta, Navbar y FooterSection, que forman parte esencial de la estructura de la página. Además, he definido las propiedades esperadas (title, meta, structuredData), lo cual permite una personalización del contenido basado en los datos proporcionados al componente. Este mecanismo de paso de propiedades no solo facilita la flexibilidad del componente, sino que también asegura que el contenido se adapte perfectamente al contexto en el que es utilizado, manteniendo la integridad y consistencia de la interfaz.

6.3.7. Ejemplo de uso de TailwindCSS

Tailwind CSS es una herramienta que permite diseñar directamente en el HTML mediante clases utilitarias, lo que facilita la creación de interfaces de usuario personalizadas y coherentes con la identidad visual de la aplicación. Este enfoque permite escribir CSS de manera más eficiente, reutilizando estilos y adaptando fácilmente el diseño a diferentes estados y temas, como se observa en el siguiente fragmento de código (véase figura 6.19).

El código presentado muestra la implementación de un botón que permite alternar entre los temas claro y oscuro, lo cual es fundamental para mejorar la experiencia del usuario, brindándole la opción de personalizar la apariencia según sus preferencias. Gracias a Tailwind CSS, se pueden aplicar una serie de clases que gestionan el diseño del botón, como `rounded-full` para bordes redondeados, `transition duration-300` para transiciones suaves, y `hover:bg-neutral-200` para cambiar el fondo al pasar el cursor. Estas clases hacen que el código sea más limpio y fácil de mantener, permitiendo ajustes rápidos y eficientes sin necesidad de escribir CSS adicional.

```
Themelcon.astro X
1 <!-- Dark Theme Toggle Button --><!-- This button is shown when the light theme is active, and when clicked, it switches the the
2 <button
3   type="button"
4   aria-label="Dark Theme Toggle"
5   class="hs-dark-mode group flex h-8 w-8 items-center justify-center rounded-full font-medium text-neutral-600 outline-none
6   data-hs-theme-click-value="dark"
7 >
8 <!-- The SVG displayed shows an abstract icon that represents the moon (dark theme) -->
9 <svg
10  class="h-4 w-4 flex-shrink-0"
11  width="24"
12  height="24"
13  viewBox="0 0 24 24"
14  fill="none"
15  stroke="currentColor"
16  stroke-width="2"
17  stroke-linecap="round"
18  stroke-linejoin="round"
19  ><path d="M12 3a6 6 0 0 0 9 9 9 9 0 1 1-9-9Z"/></path></svg
20 >
21 <!-- Light Theme Toggle Button -->
22 <!-- This button is hidden by default and only appears when the dark theme is active, when clicked, it switches to the light t
23 </button>
24 <button
25   type="button"
26   aria-label="Light Theme Toggle"
27   class="hs-dark-mode group hidden h-8 w-8 items-center justify-center rounded-full font-medium text-neutral-600 outline-none
28   data-hs-theme-click-value="light"
29 >
30 <!-- The SVG displayed shows a standard sun icon that stands for the light theme -->
31 <svg
32  class="h-4 w-4 flex-shrink-0"
33  width="24"
34  height="24"
35  viewBox="0 0 24 24"
36  fill="none"
37  stroke="currentColor"
38  stroke-width="2"
39  stroke-linecap="round"
40  stroke-linejoin="round"
41  ><circle cx="12" cy="12" r="4"/></circle><path d="M12 8a2 2 0 1 0 4 4"
42  ></path><path d="M12 2v2"/></path><path d="M12 20v2"/></path><path
43  d="m4.93 4.93 1.41 1.41"/></path><path d="m17.66 17.66 1.41 1.41"
44  ></path><path d="M2 12h2"/></path><path d="M20 12h2"/></path><path
45  d="m6.34 17.66-1.41 1.41"/></path><path d="m19.07 4.93-1.41 1.41"
46  ></path></svg
47 >
```

Figura 6.19: Ejemplo de uso de Tailwind con Astro

6.3.8. Ejemplo de un componente de React

En el desarrollo del proyecto SiuGym, uno de los elementos clave es la integración del frontend, desarrollado en React, con el backend. Esto se ejemplifica en el componente `RoutineDetail` (véase figura 6.20), diseñado para mostrar los detalles de una rutina específica, gestionar los comentarios asociados, y recuperar información adicional como los

músculos involucrados en la rutina. React permite construir interfaces de usuario dinámicas y reactivas mediante el uso de hooks como `useState` y `useEffect`, los cuales facilitan la gestión del estado y la ejecución de efectos secundarios, respectivamente.

El componente `RoutineDetail` destaca por su capacidad para interactuar con el backend, haciendo uso de `fetch` para realizar llamadas HTTP y obtener datos en formato JSON desde la API. En este caso, se realizan múltiples solicitudes para recuperar los detalles de la rutina, los comentarios asociados y la lista de músculos. La primera llamada se dirige a `http://localhost:8080/api/routines/$id` para obtener los detalles de la rutina, que se almacenan en el estado `routineDetail`. La capacidad de React para manejar estados permite que estos datos se actualicen automáticamente en la interfaz de usuario, ofreciendo una experiencia dinámica y fluida para el usuario.

```
RoutineDetail.jsx x
1  import { useState, useEffect } from "react";
2  import ExerciseFilter from "../../components/ui/filters/ExerciseFilter";
3
4  function RoutineDetail({ id }) {
5    const [routineDetail, setRoutineDetail] = useState(null);
6    const [comments, setComments] = useState([]);
7    const [newComment, setNewComment] = useState("");
8    const [newRating, setNewRating] = useState(0);
9    const [isLoggedIn, setIsLoggedIn] = useState(false);
10   const [isAdded, setIsAdded] = useState(false);
11   const [muscles, setMuscles] = useState([]); // Estado para almacenar los músculos
12
13   useEffect(() => {
14     async function fetchRoutineDetail() {
15       try {
16         const response = await fetch(`http://localhost:8080/api/routines/${id}`);
17         const data = await response.json();
18         setRoutineDetail(data);
19       } catch (error) {
20         console.error("Error fetching routine details:", error);
21       }
22     }
23
24     async function fetchComments() {
25       try {
26         const response = await fetch(`http://localhost:8080/api/routines/${id}/comments`);
27         const data = await response.json();
28         setComments(data);
29       } catch (error) {
30         console.error("Error fetching comments:", error);
31       }
32     }
33
34     async function fetchMuscles() {
35       try {
36         const muscleResponse = await fetch("http://localhost:8080/api/muscles");
37         const musclesData = await muscleResponse.json();
38         setMuscles(musclesData); // Guardar los músculos en el estado
39       } catch (error) {
40         console.error("Error fetching muscles:", error);
41       }
42     }
43
44     fetchRoutineDetail();
45     fetchComments();
46     fetchMuscles();
47
48     const token = localStorage.getItem("token");
49     setIsLoggedIn(!!token);
50   }, [id]);
51 }
```

Figura 6.20: Componente de React

Además, el componente incluye una lógica para manejar la autenticación del usuario, utilizando un token almacenado en `localStorage`. Este token se verifica en cada renderizado del componente, actualizando el estado `isLoggedIn` para reflejar si el usuario está autenticado. Este enfoque asegura que ciertas funcionalidades, como la adición de comentarios o la calificación de la rutina, estén disponibles únicamente para los usuarios registrados. Esta integración entre frontend y backend no solo mejora la seguridad, sino que también personaliza la experiencia del usuario.

Finalmente, el uso de `fetchMuscles` muestra cómo se recupera información adicional desde el backend, en este caso, los grupos musculares disponibles, a través de una llamada a `http://localhost:8080/api/muscles`. Los datos obtenidos se almacenan en el estado `muscles`, lo que permite que otros componentes de la interfaz puedan acceder y utilizar esta información de manera eficiente. Este enfoque modular y la capacidad de React para manejar estados complejos refuerzan la robustez y escalabilidad del proyecto, asegurando que los usuarios de `SiuGym` puedan interactuar con una aplicación web altamente dinámica y adaptativa.

CAPÍTULO 7

Producto desarrollado

El producto desarrollado se ha centrado en ofrecer una experiencia personalizada y eficiente para los usuarios como Esteban, un profesor de matemáticas que, además de su pasión por el baloncesto, ha decidido mejorar su condición física a través del gimnasio. La aplicación SiuGym ha sido diseñada para cubrir las necesidades específicas de Esteban a través de una serie de escenarios que ilustran cómo interactúa con la plataforma en su viaje de entrenamiento.

En el primer escenario, Esteban, siendo nuevo en el gimnasio, decide aprender a hacer un ejercicio específico, el press de banca. La aplicación le proporciona una interfaz intuitiva para registrarse rápidamente (figura 7.1).

The screenshot displays the SiuGym registration interface. At the top, there is a navigation bar with the SiuGym logo and links for 'Inicio', 'Ejercicios', 'Rutinas', and 'Inicia sesión'. The main content area is a dark-themed landing page with the headline 'Transforma tu entrenamiento con SiuGym' and a sub-headline 'Convierte cada entrenamiento en un paso hacia tu mejor versión'. A central white modal form titled 'Regístrate' is overlaid on the page. The form contains the following fields: '¿Ya tienes una cuenta? [Inicia sesión aquí](#)', 'Usuario' (Esteban00), 'Nombre' (Esteban), 'Apellidos' (Duro), 'Correo Electronico' (estebanduro@gmail.com), a date picker for '13/04/2000', 'Contraseña' (masked with dots), and 'Confirm Password' (masked with dots). A prominent yellow 'Regístrate' button is located at the bottom of the form. The background of the landing page features an illustration of a gym with people exercising and text describing 'Entrenamientos a medida' and 'Guiado'.

Figura 7.1: Pantalla de registro de usuario en SiuGym

Una vez dentro, navega fácilmente hasta el ejercicio que quiere aprender, utilizando el link del navegador que se encuentra en la parte superior de la pantalla y presionando en el ejercicio (véase 7.2). SiuGym le ofrece un videotutorial detallado junto con una guía paso a paso que asegura que Esteban comprenda la técnica adecuada (véase la figura 7.3). Esto le da la confianza necesaria para practicar el ejercicio en el gimnasio, sabiendo que está realizando los movimientos de manera segura y efectiva.

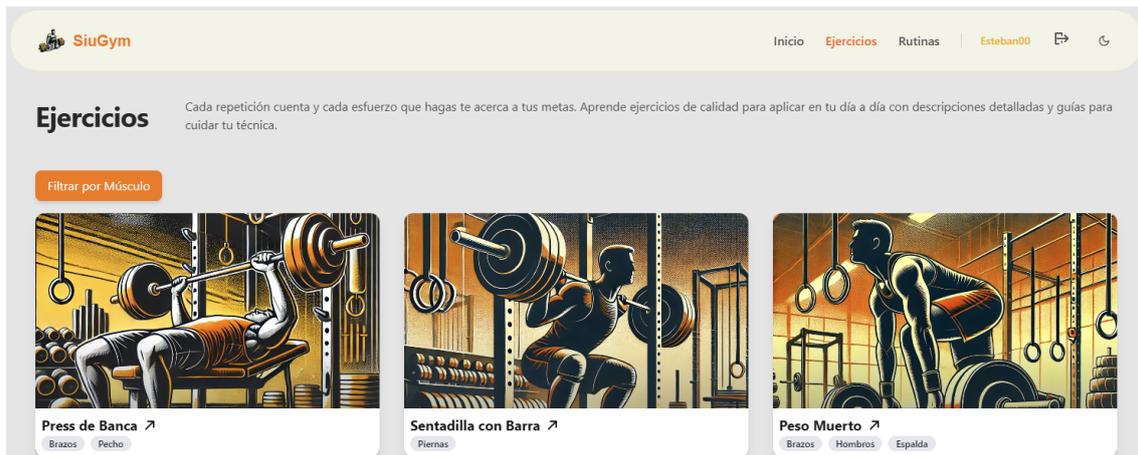


Figura 7.2: Pantalla con todos los ejercicios disponibles en SiuGym

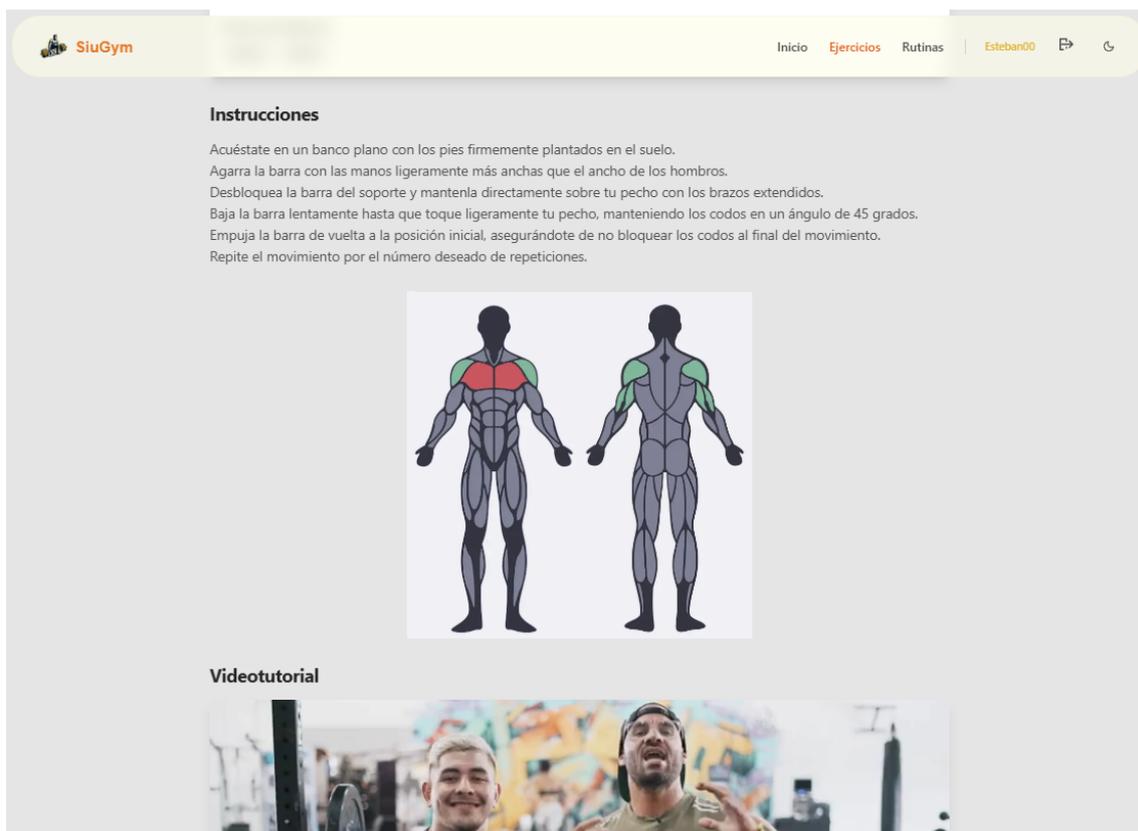


Figura 7.3: Instrucciones, músculos y videotutorial de press de banca en SiuGym

A medida que Esteban se siente más cómodo con la aplicación, en el segundo escenario, decide enfocarse en desarrollar su pecho. Usa SiuGym para buscar una rutina de pecho adecuada a su nivel, para ello busca en la página de rutinas utilizando los filtros de la página, indicando que quiere entrenar el músculo del pecho y de dificultad de principiante (véase la figura 7.4).

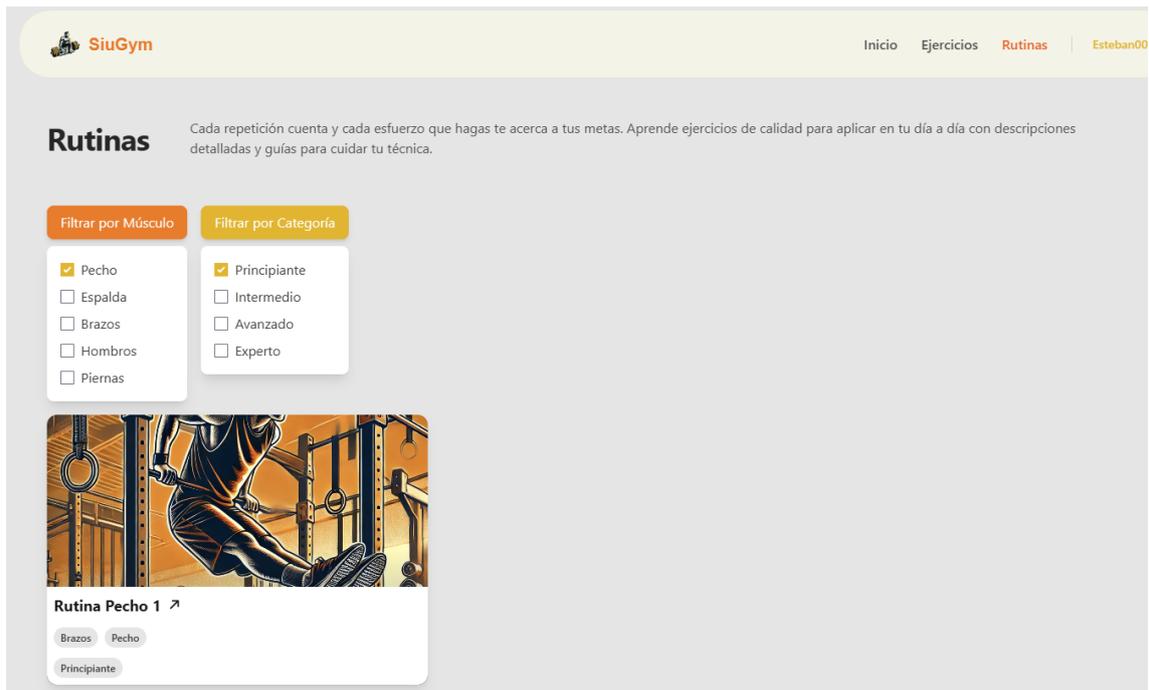


Figura 7.4: Pantalla con las rutinas filtradas

La aplicación no solo le proporciona la rutina, sino que también incluye vídeos instructivos y descripciones detalladas de cada ejercicio. Esteban examina cuidadosamente cada ejercicio, se asegura de entenderlo bien, y luego guarda la rutina en su perfil (véase la figura 7.5) para su próxima sesión en el gimnasio. Esta funcionalidad le permite acceder fácilmente a su plan de entrenamiento cuando lo necesita, optimizando su tiempo y esfuerzo.

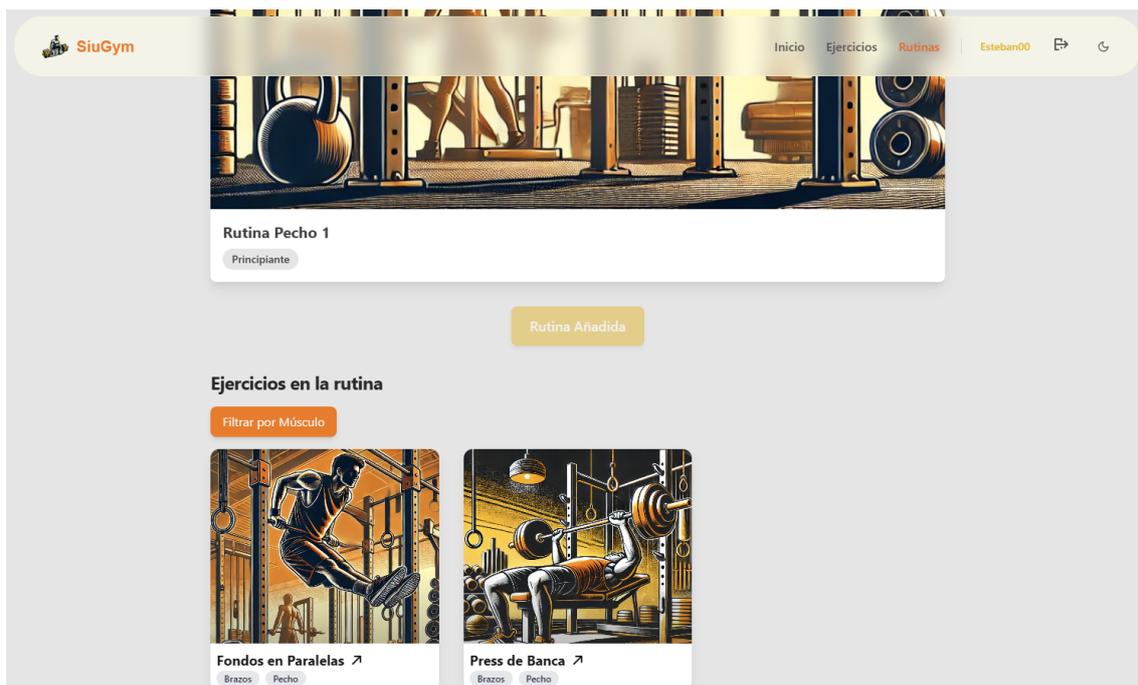


Figura 7.5: Pantalla con los ejercicios de la rutina y el botón para añadirla al usuario

Después de acabar las clases, Esteban va al gimnasio y ejecuta la rutina previamente guardada. Para ello inicia sesión en la página en su móvil, va a su perfil y presiona el botón de ejecución de su rutina (véase figura 7.6). Una vez ejecutada, le aparecen los distintos ejercicios y la posibilidad de ir marcando en cada uno de estos el número de series, repeticiones y peso que utiliza en cada uno para posteriormente guardar la ejecución de la rutina (véase la figura 7.7)



Figura 7.6: Vista móvil de la web en el apartado de las rutinas del usuario

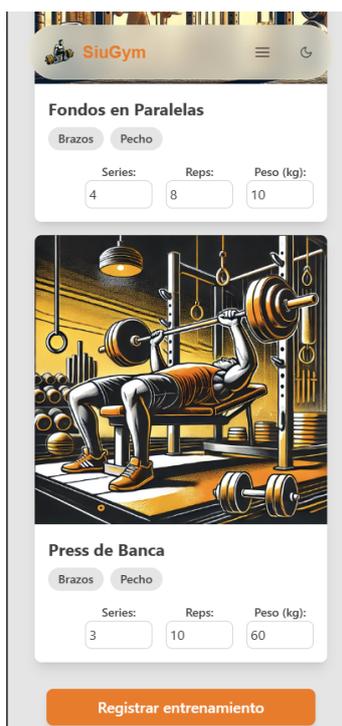


Figura 7.7: Vista móvil de la web para ejecutar rutinas

Después de llevar un tiempo entrenando, Esteban decide crear su propia rutina personalizada, para ello accede al menú de rutinas y presiona el botón de crear rutina (véase la figura 7.8). Una vez en la pantalla de creación, le da un nombre a su rutina, escribe una descripción con información sobre esta rutina, selecciona la categoría de dificultad que tiene y elige aquellos ejercicios que mas le gustan, procediendo a guardar la rutina (véase la figura 7.9).

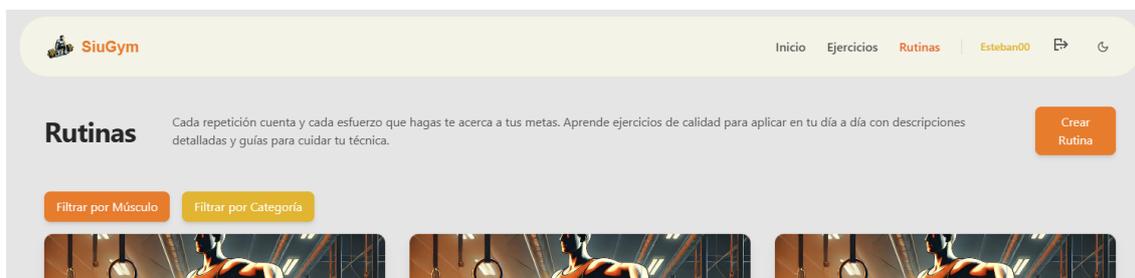


Figura 7.8: Vista de la pagina de rutinas con el botón para crear rutinas

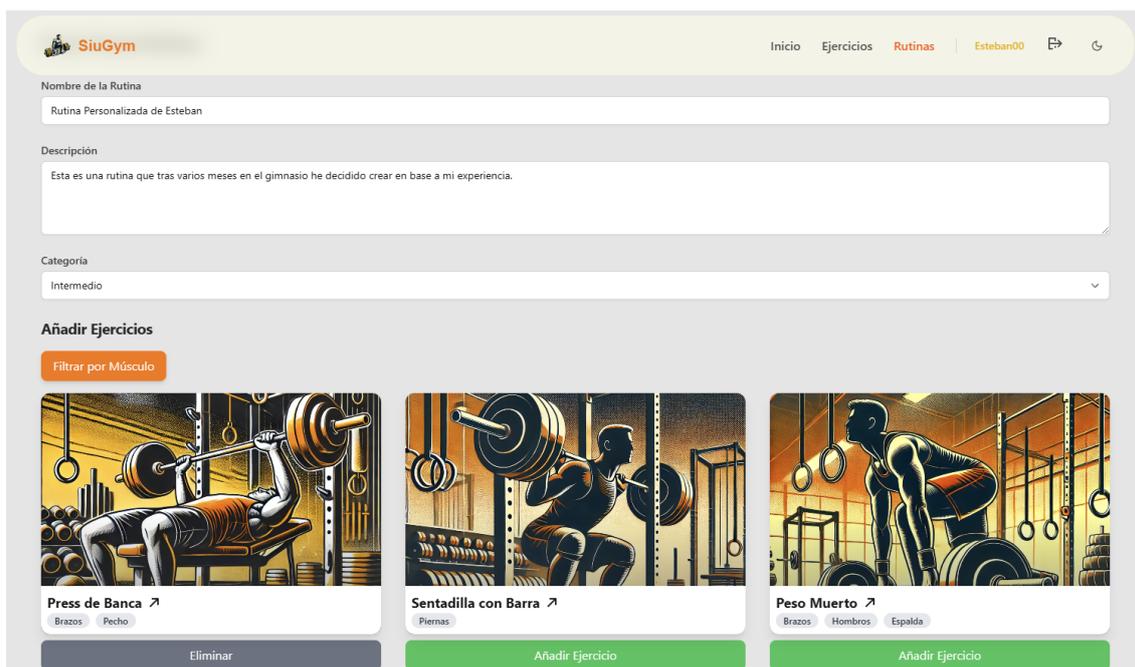


Figura 7.9: Pagina de creación de rutinas de SiuGym

Esteban, en su búsqueda por diversificar su entrenamiento, decide explorar nuevas rutinas dentro de la aplicación. Para ello, accede al menú de búsqueda de rutinas y utiliza las funciones de filtrado para seleccionar aquellas que se ajusten a su nivel de dificultad y los grupos musculares que desea trabajar (véase la figura 7.10). Una vez filtradas las opciones, revisa las opiniones y valoraciones de otros usuarios, lo que le permite evaluar cuál de las rutinas se alinea mejor con sus objetivos personales (véase la figura 7.11). Finalmente, escoge una rutina que le parece adecuada y la guarda en su perfil para probarla en su próximo entrenamiento.

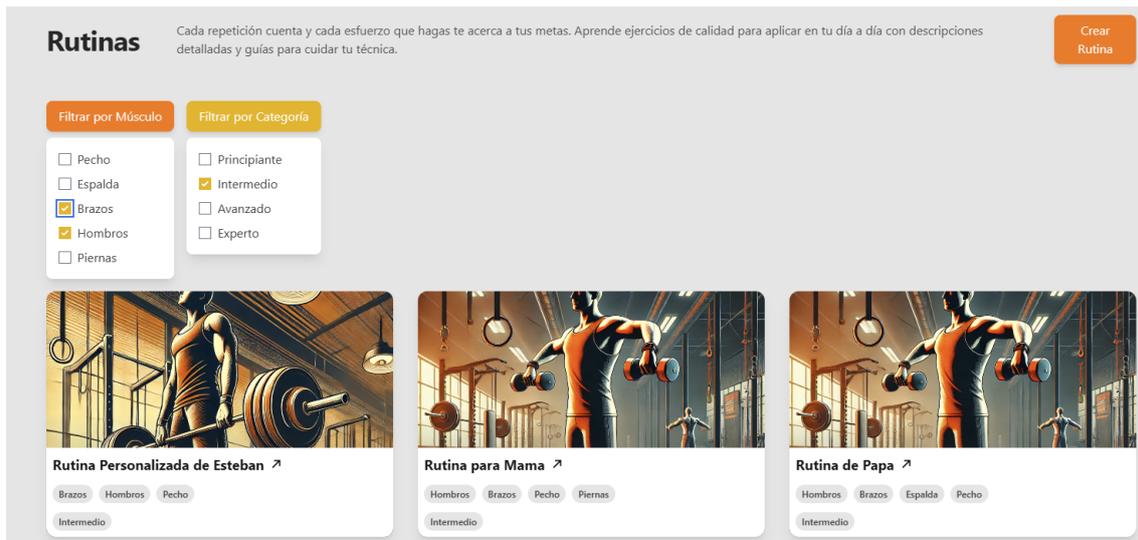


Figura 7.10: Vista de la pagina de rutinas con las rutinas filtradas

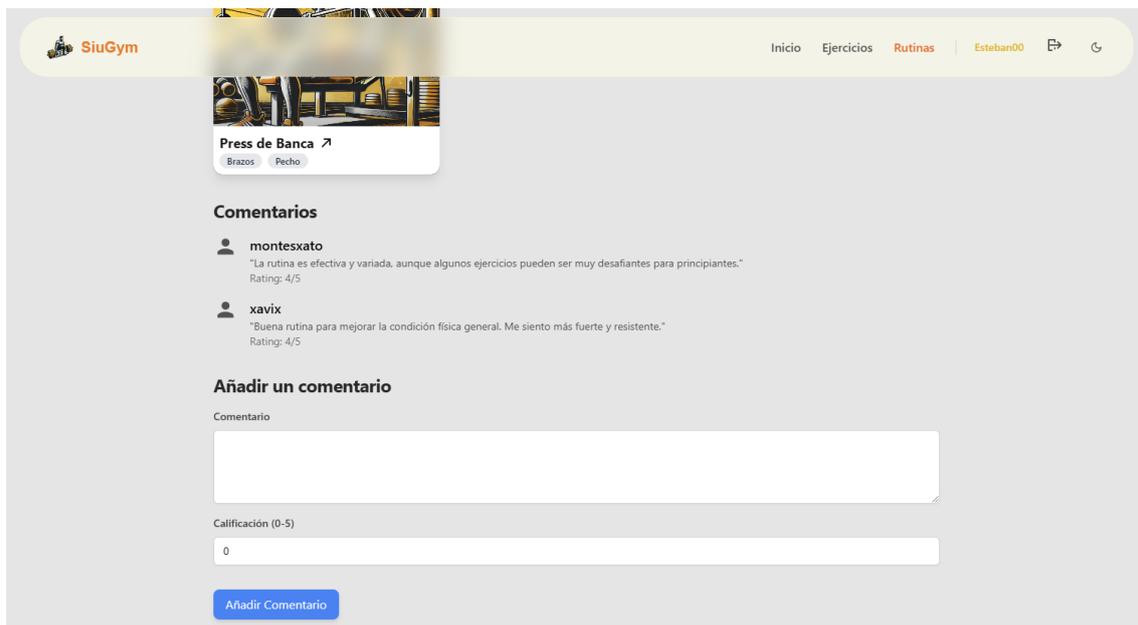


Figura 7.11: Comentarios de una rutina junto con su valoración

Finalmente, Esteban desea analizar el progreso que ha logrado tras varios meses de entrenamiento. Para ello, accede a la sección de estadísticas dentro de la aplicación, en su perfil, donde puede ver los gráficos detallados que muestran su evolución en diferentes ejercicios y su aumento de fuerza a lo largo del tiempo (véase la figura 7.12). Esta funcionalidad le permite identificar músculos en los que ha mejorado y otros en las que puede enfocarse más, ayudándole a ajustar su plan de entrenamiento de manera estratégica.



Figura 7.12: Gráfico del progreso en el peso máximo del ejercicio de Press de Banca de Esteban

CAPÍTULO 8

Validación

La evaluación del prototipo de una página web es esencial para garantizar el éxito del proyecto. Identificar problemas en una etapa temprana permite correcciones eficientes y económicas, evitando retrabajos costosos más adelante. Además, la evaluación asegura una experiencia del usuario óptima al verificar la usabilidad, la coherencia visual y la alineación con los objetivos del proyecto.

La evaluación del prototipo también desempeña un papel crucial en la gestión del tiempo y los recursos. Detectar y abordar problemas antes de la implementación completa ahorra tiempo y evita costos innecesarios asociados con correcciones tardías. Además, al centrarse en la coherencia visual y la alineación con los objetivos del proyecto, la evaluación garantiza que la página web cumpla con los estándares de calidad establecidos. Este enfoque no solo beneficia al equipo de desarrollo, sino que también contribuye a una entrega más rápida y eficiente del producto final.

8.1 Evaluación Heurística

La evaluación heurística es una técnica de usabilidad que se emplea para identificar problemas de diseño en la interfaz de un sistema. Dentro de este contexto, los principios heurísticos de Jakob Nielsen son fundamentales, ya que proporcionan un marco de referencia que permite a los diseñadores y evaluadores analizar la efectividad de una interfaz desde la perspectiva del usuario. Como se comenta en la página mtp [13], los principios son los siguientes:

1. **Visibilidad del estado del sistema.** Este principio establece que el sistema debe mantener siempre informados a los usuarios sobre lo que está ocurriendo, proporcionando retroalimentación adecuada en un tiempo razonable. Esto ayuda a los usuarios a entender las consecuencias de sus acciones y a mantener el control sobre el sistema.
2. **Correspondencia entre el sistema y el mundo real.** La interfaz debe utilizar un lenguaje claro y familiar para los usuarios, con conceptos y términos que reflejen su mundo real. Esto reduce la curva de aprendizaje y evita confusiones, permitiendo que el usuario se sienta más cómodo e intuitivamente sepa cómo interactuar con el sistema.
3. **Control y libertad del usuario.** Los usuarios a menudo seleccionan funciones por error y necesitan una salida de emergencia claramente marcada para salir del estado no deseado sin tener que pasar por un proceso largo. Este principio sugiere que las

interfaces deben proporcionar deshacer y rehacer, de modo que el usuario se sienta en control y libre de explorar el sistema sin temor a cometer errores irreversibles.

4. **Consistencia y estándares.** Los usuarios no deberían tener que preguntarse si diferentes palabras, situaciones o acciones significan lo mismo. La consistencia en el diseño reduce la carga cognitiva, facilitando que los usuarios aprendan y utilicen la interfaz de manera más eficaz y eficiente.
5. **Prevención de errores.** En lugar de diseñar mensajes de error buenos, el sistema debería estar diseñado para evitar que ocurran errores en primer lugar. Esta prevención puede lograrse mediante la limitación de las opciones que pueden llevar a errores o proporcionando advertencias antes de que se cometan errores.
6. **Reconocimiento antes que memorización.** Este principio enfatiza la importancia de reducir la carga de memoria del usuario al hacer que los objetos, acciones y opciones sean visibles. El usuario no debería tener que recordar la información de una parte del diálogo a otra. Las instrucciones para el uso del sistema deben ser visibles o fácilmente recuperables siempre que sea necesario.
7. **Flexibilidad y eficiencia de uso.** Los sistemas deben ser lo suficientemente flexibles para atender tanto a usuarios novatos como a expertos, proporcionando vías rápidas para los usuarios avanzados que permiten realizar acciones frecuentes de manera más eficiente.
8. **Estética y diseño minimalista.** Las interfaces deben estar diseñadas de manera que no contengan información irrelevante o que no se utilice con frecuencia, ya que cada unidad de información adicional compite con la información relevante y disminuye su visibilidad.
9. **Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores.** Los mensajes de error deben expresarse en un lenguaje simple, indicar el problema claramente y sugerir una solución. Esto ayuda a los usuarios a comprender qué ha salido mal y cómo pueden solucionarlo, minimizando la frustración.
10. **Ayuda y documentación.** Aunque lo ideal es que el sistema se utilice sin documentación, es necesario proporcionar ayuda y documentación accesibles para los usuarios cuando lo necesiten. Esta información debe estar enfocada en la tarea del usuario, listar los pasos concretos a seguir y no ser demasiado extensa.

A continuación, se desglosan cómo cada uno de los principios de usabilidad de Nielsen se refleja en la aplicación desarrollada. Estos principios guían el diseño y la estructura de SiuGym, asegurando que la experiencia del usuario sea intuitiva, fluida y satisfactoria en todo momento.

Visibilidad del Estado del Sistema

En este proyecto, la visibilidad del estado del sistema es crucial para garantizar una experiencia de usuario fluida y sin fricciones. Cada vez que Esteban interactúa con la aplicación, ya sea al iniciar sesión, seleccionar una rutina, o registrar un ejercicio, el sistema proporciona una retroalimentación inmediata. Por ejemplo, al guardar una rutina personalizada, la aplicación muestra un mensaje de confirmación que asegura a Esteban que su acción ha sido exitosa (véase figura 8.1). Este tipo de visibilidad le permite a los usuarios mantener un control total sobre sus actividades dentro de la plataforma, reduciendo la incertidumbre y mejorando la confianza en el uso de la aplicación.

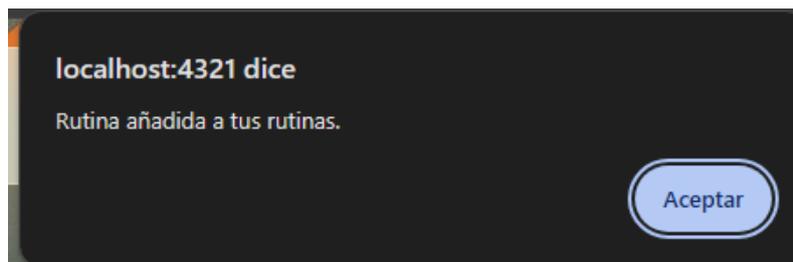


Figura 8.1: Alerta que aparece en la página al guardar una rutina

Coincidencia entre el Sistema y el Mundo Real

La interfaz de la aplicación se ha diseñado teniendo en cuenta la familiaridad de los usuarios con el entorno del gimnasio. Los términos utilizados son claros y están directamente relacionados con el mundo del fitness, como ejercicios, rutinas o progreso. Esto facilita que los usuarios, como Esteban, comprendan de inmediato las funciones disponibles y cómo interactuar con ellas. Además, la representación visual de los ejercicios, con imágenes y vídeos instructivos, refleja lo que Esteban esperaría ver en un entorno de entrenamiento real, lo que ayuda a conectar el mundo digital con su experiencia en el gimnasio.

Control y Libertad del Usuario

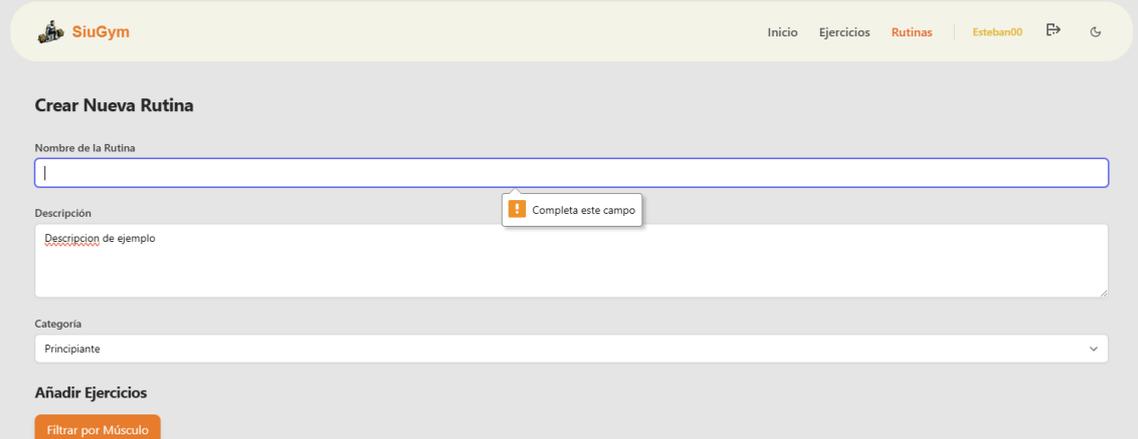
La aplicación proporciona a los usuarios un alto grado de control sobre su experiencia, permitiéndoles moverse libremente entre las diferentes secciones sin perder su progreso. Esteban, por ejemplo, puede cambiar de rutina en cualquier momento o ajustar su plan de entrenamiento sin temor a perder los datos guardados. La barra de navegación superior, presente en todas las pantallas, permite regresar fácilmente a la página principal o explorar otras secciones de la aplicación sin perderse.

Consistencia y Estándares

Este proyecto mantiene una consistencia visual y funcional que facilita la navegación y el aprendizaje del usuario. Cada sección de la aplicación sigue un diseño coherente, con un título claro, contenido relevante y un pie de página informativo. La navegación a través de las diferentes funcionalidades es intuitiva, y Esteban siempre puede regresar a la página principal haciendo clic en el logo de la aplicación. Este enfoque consistente no solo mejora la usabilidad, sino que también reduce la carga cognitiva para los usuarios, permitiéndoles centrarse en sus objetivos de entrenamiento sin distracciones innecesarias.

Prevención de Errores

La prevención de errores es un aspecto fundamental en el diseño de esta aplicación. Desde la selección de ejercicios hasta la configuración de rutinas, la interfaz está diseñada para minimizar las posibilidades de que los usuarios cometan errores. Por ejemplo, cuando se crea una rutina y no se rellena un campo sale un aviso por pantalla para recordarle al usuario que tiene que rellenar un campo (véase la figura 8.2).



The screenshot shows the 'Crear Nueva Rutina' (Create New Routine) page in the SiuGym application. At the top, there is a navigation bar with the SiuGym logo and links for 'Inicio', 'Ejercicios', 'Rutinas', and 'Esteban00'. The main form is titled 'Crear Nueva Rutina' and contains three input fields: 'Nombre de la Rutina' (empty), 'Descripción' (containing 'Descripción de ejemplo'), and 'Categoría' (set to 'Principiante'). A red warning icon and the text 'Completa este campo' are displayed above the description field. Below the form, there is a section titled 'Añadir Ejercicios' with a button labeled 'Filtrar por Músculo'.

Figura 8.2: Aviso que se muestra al no rellenar un campo en la creación de rutinas

Reconocimiento en lugar de Recuerdo

El diseño de la aplicación se enfoca en reducir la carga de memoria de los usuarios. Las opciones y acciones más importantes son visibles y accesibles en todo momento, lo que permite a los usuarios reconocer fácilmente lo que necesitan sin tener que recordar detalles específicos de una sesión anterior. Por ejemplo, al explorar diferentes rutinas, los botones y opciones son claramente visibles y están etiquetados de manera intuitiva, lo que facilita la navegación y la ejecución de tareas sin necesidad de memorizar rutas complejas o detalles de la interfaz.

Flexibilidad y Eficiencia de Uso

La aplicación ha sido diseñada para ser flexible y eficiente, adaptándose tanto a usuarios novatos como avanzados. Esteban puede utilizar funciones básicas con facilidad, como seleccionar una rutina predefinida, pero también tiene la opción de acceder a funcionalidades más avanzadas, como la creación de rutinas personalizadas. Además, para los usuarios más experimentados, la aplicación ofrece atajos y funciones que permiten realizar acciones frecuentes de manera más rápida, optimizando el tiempo que Esteban dedica a su planificación y seguimiento de entrenamientos.

Diseño Estético y Minimalista

El diseño de la interfaz se ha mantenido limpio y minimalista, enfocándose en la funcionalidad sin añadir elementos innecesarios que puedan distraer al usuario. Cada pantalla presenta solo la información esencial, permitiendo a Esteban concentrarse en sus tareas sin sentirse abrumado por detalles irrelevantes. La jerarquía visual está bien definida, con títulos claros y opciones organizadas de manera lógica, lo que facilita la navegación y asegura que cada tarea pueda realizarse de manera eficiente y sin complicaciones.

Ayudar a salir de los errores

La aplicación ha sido diseñada para facilitar la recuperación de errores, permitiendo a los usuarios corregir sus acciones sin frustraciones. Por ejemplo, si Esteban selecciona un ejercicio incorrecto para su rutina, puede deshacer esa acción fácilmente antes de guardar los cambios, volviendo a pulsar el botón de añadido (véase la figura 8.3). Además, la interfaz proporciona mensajes claros y concisos cuando ocurre un error, guiando a Esteban sobre cómo resolver el problema y continuar con su sesión de entrenamiento sin interrupciones significativas.

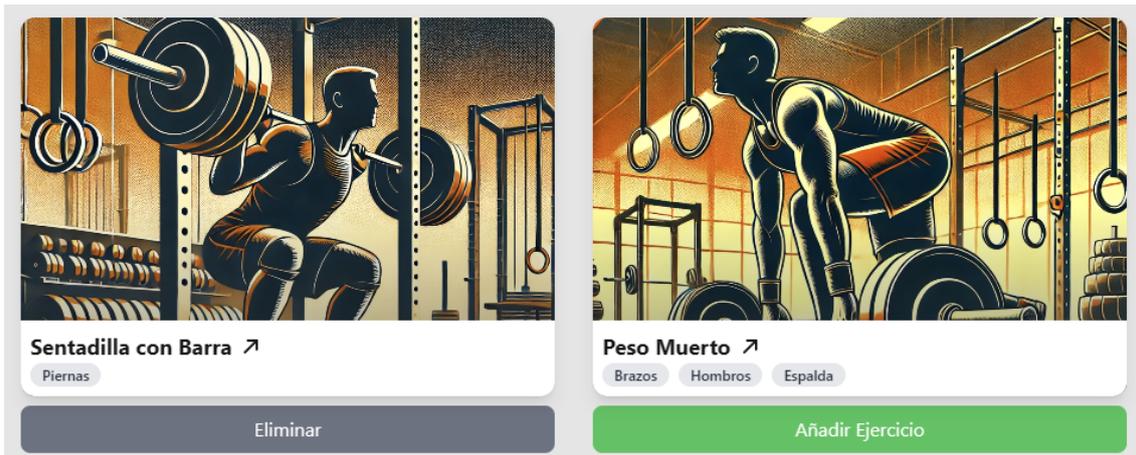


Figura 8.3: Ejemplos de ejercicios con sus respectivos botones en la creación de rutinas

Ayuda y documentación

Si bien la aplicación ha sido diseñada para ser intuitiva y fácil de usar, se proporciona poca ayuda a los usuarios. Esteban puede acceder a una sección de preguntas frecuentes (FAQ) donde se abordan las dudas más comunes (véase la figura 8.4), sin embargo no hay ningún tipo de documentación adicional para el uso de la página. La adición de más documentación sobre la página es un tema a mejorar.



Figura 8.4: Preguntas más frecuentes (FAQ) de SiuGym

CAPÍTULO 9

Conclusiones y trabajos futuros

Al concluir este proyecto, puedo afirmar que he logrado cumplir con los objetivos planteados inicialmente. El desarrollo de la aplicación SiuGym ha permitido no solo crear una herramienta accesible y útil para los usuarios interesados en mejorar su rendimiento físico, sino también me ha permitido profundizar en aspectos clave del diseño de software, experiencia de usuario, y manejo de tecnologías modernas como Spring Boot, JWT para autenticación, Google Cloud Storage para la gestión de medios y tecnologías tan modernas como AstroJS y React.

Durante el proceso, he enfrentado varios desafíos, especialmente en la integración de los distintos servicios y la creación de una interfaz intuitiva, pero cada obstáculo ha contribuido significativamente a mi crecimiento tanto personal como profesional.

El proyecto me ha permitido aplicar y reforzar mis conocimientos sobre el desarrollo web aprendidos tanto en la asignatura de DEW (Desarrollo Web) como de DCU (Diseño Centrado en el Usuario), el diseño de bases de datos, aprendido en gran medida en las asignaturas de tercero de BBDD (Bases de Datos) y TBD (Tecnologías de Bases de Datos) y el desarrollo de la arquitectura de software, aprendido en la asignatura de ISW (Ingeniería del Software). Además, a través del análisis de necesidades de los usuarios y la creación de prototipos, he podido comprender mejor la importancia de adaptar las soluciones tecnológicas a las expectativas y habilidades de los usuarios finales. Estoy particularmente satisfecho con cómo la aplicación aborda las necesidades de personalización y seguimiento del progreso en el entrenamiento físico.

A pesar del trabajo hecho, considero que aún existen áreas de mejora y expansión para el proyecto. En el futuro, me gustaría explorar la posibilidad de integrar inteligencia artificial para ofrecer recomendaciones personalizadas de rutinas de ejercicio basadas en los datos de progreso del usuario.

Otra dirección que sería interesante es la ampliación de la aplicación para soportar una mayor variedad de dispositivos y plataformas, incluyendo wearables que puedan proporcionar datos en tiempo real sobre el rendimiento físico del usuario o la creación de una aplicación nativa para Android e IOS. Finalmente, también me parece crucial continuar mejorando la accesibilidad de la aplicación, asegurando que pueda ser utilizada por personas con diferentes niveles de habilidades tecnológicas y físicas, tal vez añadiendo funciones concretas para el manejo de la aplicación por voz.

Este proyecto ha sido una experiencia enriquecedora, y espero seguir desarrollándolo en el futuro para maximizar su impacto y utilidad, a la par que permitiéndome aumentar y mejorar mis conocimientos.

Bibliografía

- [1] Colegio de Farmacéuticos de Sevilla. Los 3 pilares de la salud, 2017. Disponible en <https://www.farmaceticosdesevilla.es/consejossaludables/vida-saludable/los-3-pilares-de-la-salud/>, consultada el 17/07/2024].
- [2] Ministerio de Sanidad. Beneficios de la actividad física, 2020. Disponible en <https://www.sanidad.gob.es/areas/promocionPrevencion/actividadFisica/guiaPadresMadres/beneficiosActividadFisica.htm>, consultada el 17/07/2024].
- [3] Palco23. El fitness llora por la pandemia y apunta a una nueva concentración, 2020. Disponible en <https://www.palco23.com/fitness/el-fitness-llora-por-la-pandemia-y-apunta-a-una-nueva-concentracion>, consultada el 19/07/2024].
- [4] CmdSport. El periodo más aciago de la historia del fitness español, 2021. Disponible en <https://www.cmdsport.com/fitness/actualidad-fitness/periodo-mas-aciago-la-historia-del-fitness-espanol/>, consultada el 19/07/2024].
- [5] Statista. Porcentaje de individuos que fue a un gimnasio en España de 1996 a 2023, 2024. Disponible en <https://es.statista.com/estadisticas/569644/evolucion-del-porcentaje-de-individuos-que-fueron-a-un-gimnasio/>, consultada el 19/07/2024].
- [6] Nosolousabilidad. Diseño centrado en el usuario (dcu), 2020. Disponible en <https://www.nosolousabilidad.com/manual/3.htm>, consultada el 19/08/2024].
- [7] Emprende a conciencia. Ficha de persona, 2021. Disponible en <https://www.emprendeaconciencia.com/ficha-persona>, consultada el 19/08/2024].
- [8] Medium Seba Gritti. Diseño centrado en el usuario (dcu), 2021. Disponible en <https://grittiseba.medium.com/dise%C3%B1o-centrado-en-el-usuario-dcu-7f1d5893f14b>, consultada el 19/08/2024].
- [9] Episer Marketing Digital. Historia del kanban, 2021. Disponible en <https://tormetal.com/blog/historia-del-kanban/>, consultada el 19/08/2024].
- [10] Libertad Digital Belen Lazaro. ¿cuánto tiempo debe durar un entrenamiento ideal?, 2023. Disponible en <https://www.libertaddigital.com/deportes/mas-deporte/2023-04-26/cuanto-tiempo-debe-durar-un-entrenamiento-ideal-0e-7007888/>, consultada el 20/08/2024].
- [11] The PostgreSQL Global Development Group. PostgreSQL 16.4 documentation, 2024. Disponible en <https://www.postgresql.org/docs/current/>, consultada el 12/08/2024].
- [12] La Vanguardia Beatriz Benítez. ¿qué significa el color dorado en psicología?, 2022. Disponible en <https://www.lavanguardia.com/vivo/psicologia/20220514/>

8266458/que-significa-color-dorado-psicologia-nbs.html, consultada el 20/08/2024].

- [13] MTP UX Héctor Bernardo López. Evaluación heurística, la metodología más utilizada en ux para medir la usabilidad de una interfaz, 2018. Disponible en <https://www.mtp.es/blog/experiencia-de-usuario-blog/evaluacion-heuristica-la-usabilidad-una-interfaz/>, consultada el 31/08/2024].

Glosario

Backend	Es el motor que impulsa la aplicación desde detrás de las cámaras. Es la parte responsable de la lógica del negocio, la gestión de bases de datos, y el manejo de solicitudes y respuestas que no son visibles para el usuario final. En este proyecto, el backend se ha construido utilizando tecnologías robustas que garantizan la seguridad, la eficiencia y la escalabilidad, permitiendo que la aplicación maneje grandes volúmenes de datos y operaciones complejas sin comprometer el rendimiento..
Endpoint	Un Endpoint es un punto de acceso específico en una API que define una ruta para recibir y procesar solicitudes HTTP. Cada endpoint corresponde a una URL concreta que, cuando es llamada, desencadena una acción en el servidor, como la recuperación o el envío de datos. En el contexto de este proyecto, los endpoints se han diseñado cuidadosamente para que sean coherentes, claros y seguros, facilitando la interacción entre el frontend y el backend y asegurando que cada operación se realice de manera eficiente y sin errores..
Framework	Es una estructura predefinida que facilita y acelera el desarrollo de software, proporcionando un conjunto de herramientas, bibliotecas y convenciones que guían al desarrollador en la construcción de aplicaciones. En este proyecto, se han utilizado frameworks modernos y bien establecidos que no solo simplifican el proceso de desarrollo, sino que también aseguran que la aplicación esté alineada con las mejores prácticas de la industria, permitiendo un código más limpio, mantenible y escalable..

Frontend	Es la capa de la aplicación con la que interactúa directamente el usuario. Es la parte visible y tangible de un sitio web o aplicación, encargada de la presentación visual y la experiencia de usuario. En este proyecto, se ha prestado especial atención a la usabilidad y el diseño intuitivo del frontend, asegurando que cada elemento visual no solo sea estéticamente agradable, sino que también facilite una navegación fluida y accesible para todos los usuarios..
HTTP (Hypertext Transfer Protocol)	Es el protocolo fundamental que permite la comunicación entre el cliente y el servidor en la web. Es el canal a través del cual se envían y reciben las solicitudes y respuestas que hacen posible la carga de páginas, la recuperación de datos y la interacción con aplicaciones web. En el contexto de este proyecto, HTTP es esencial para las operaciones diarias, asegurando que cada interacción del usuario con la aplicación se traduzca en acciones precisas y respuestas rápidas del servidor..
JWT (JSON Web Token)	Es un estándar para la creación de tokens que permiten la transmisión de información de manera segura entre el cliente y el servidor en forma de un objeto JSON. Este proyecto utiliza JWT para manejar la autenticación y la autorización de usuarios, asegurando que solo las personas autorizadas puedan acceder a recursos específicos de la aplicación. La implementación de JWT refuerza la seguridad del sistema, protegiendo los datos sensibles y garantizando que cada sesión sea confiable y segura..

Anexos

A Cuestionarios detallados

Aquí se encuentran recogidas todas las respuestas al cuestionario creado en el apartado 4.1. Para ver la encuesta acceder al siguiente enlace: Encuesta en Google Forms

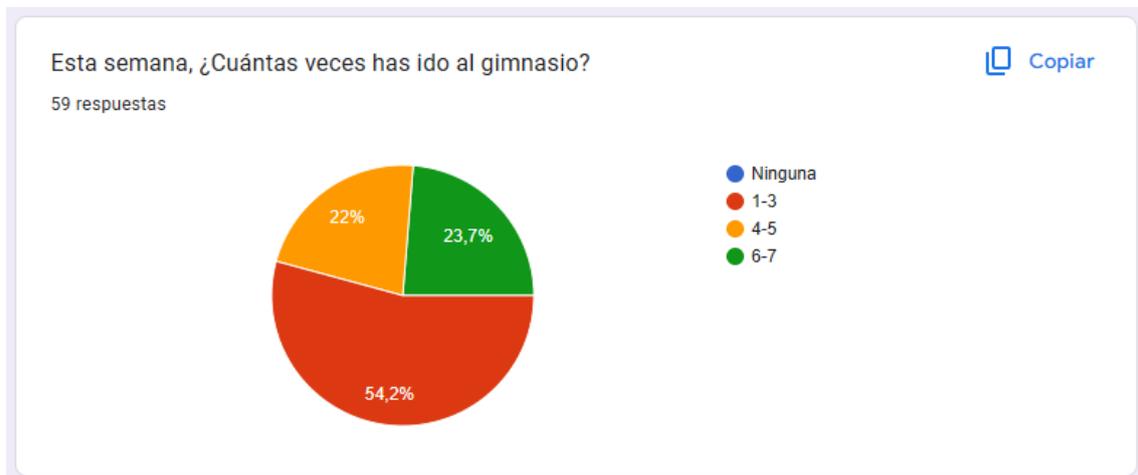


Figura A.1: Respuesta a la primera Pregunta del Cuestionario

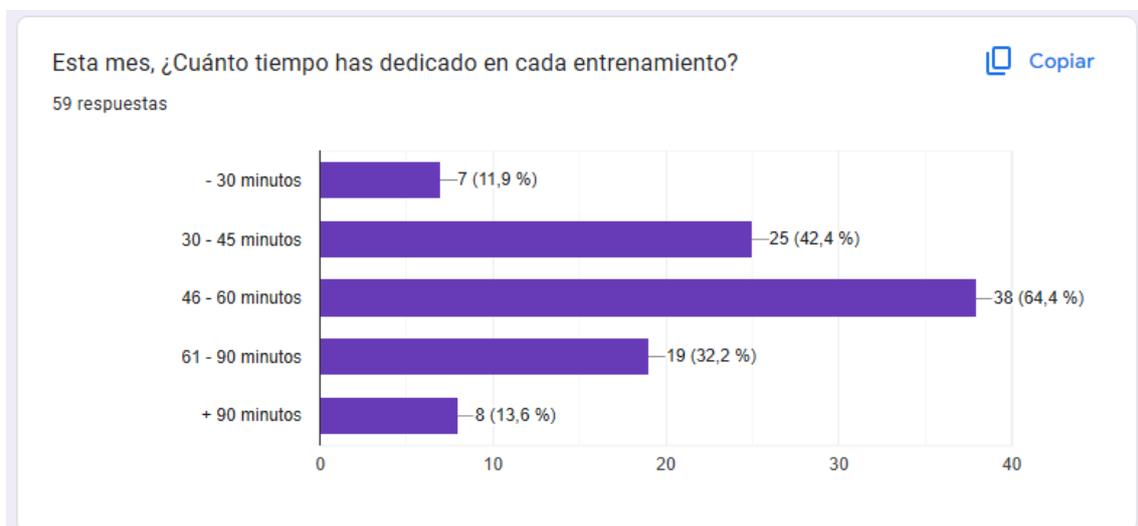


Figura A.2: Respuesta a la segunda Pregunta del Cuestionario

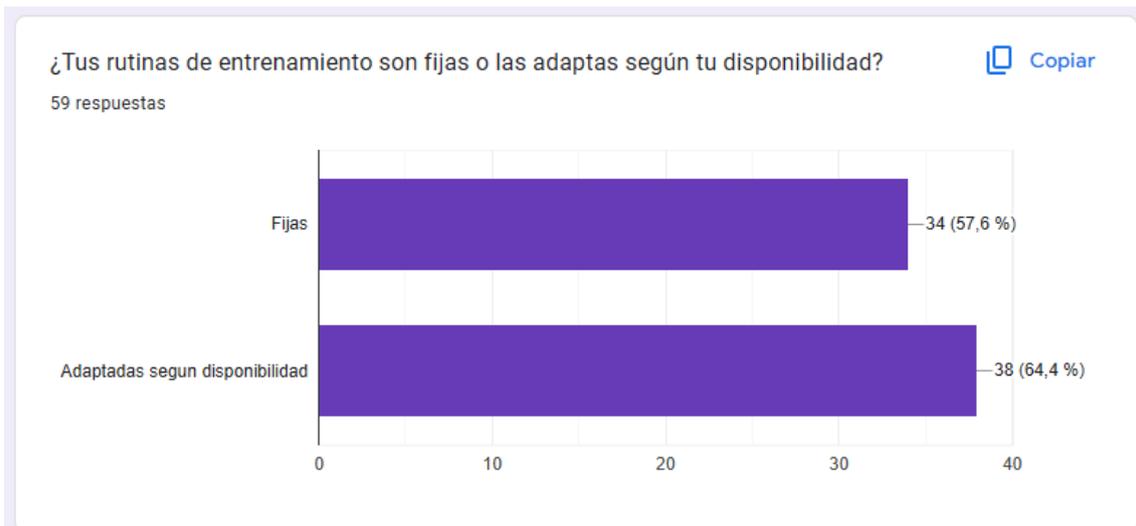


Figura A.3: Respuesta a la tercera Pregunta del Cuestionario



Figura A.4: Respuesta a la cuarta Pregunta del Cuestionario

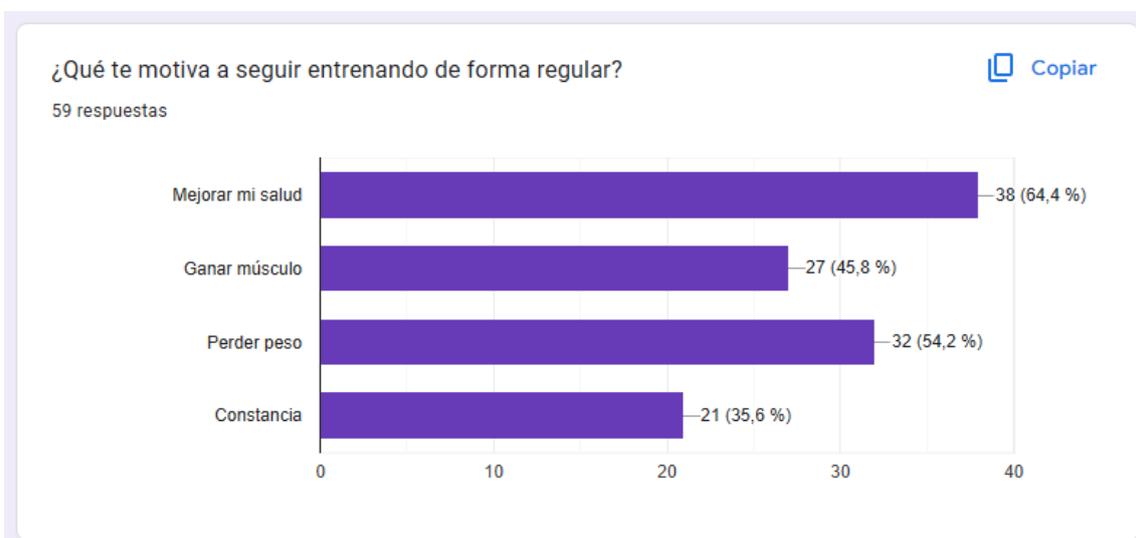


Figura A.5: Respuesta a la quinta Pregunta del Cuestionario



Figura A.6: Respuesta a la sexta Pregunta del Cuestionario



Figura A.7: Respuesta a la séptima Pregunta del Cuestionario

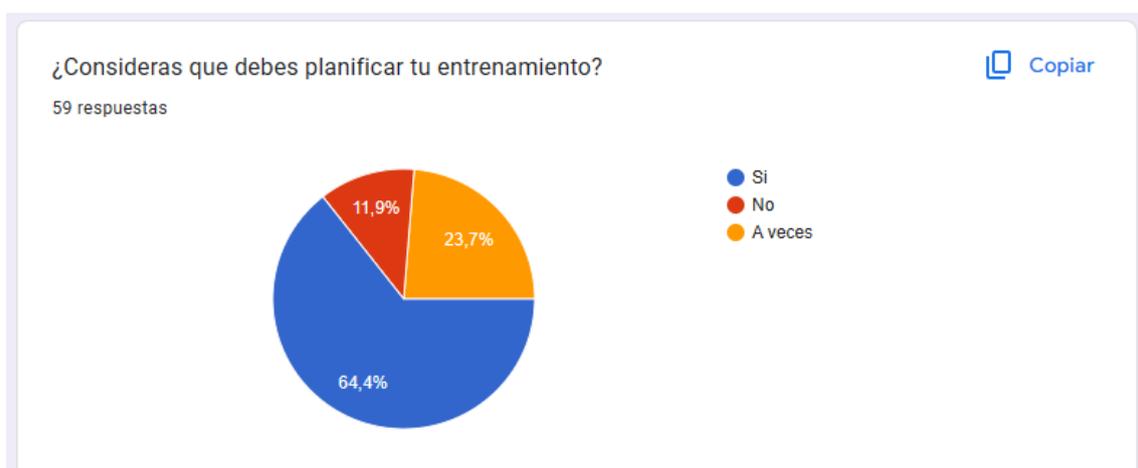


Figura A.8: Respuesta a la octava Pregunta del Cuestionario

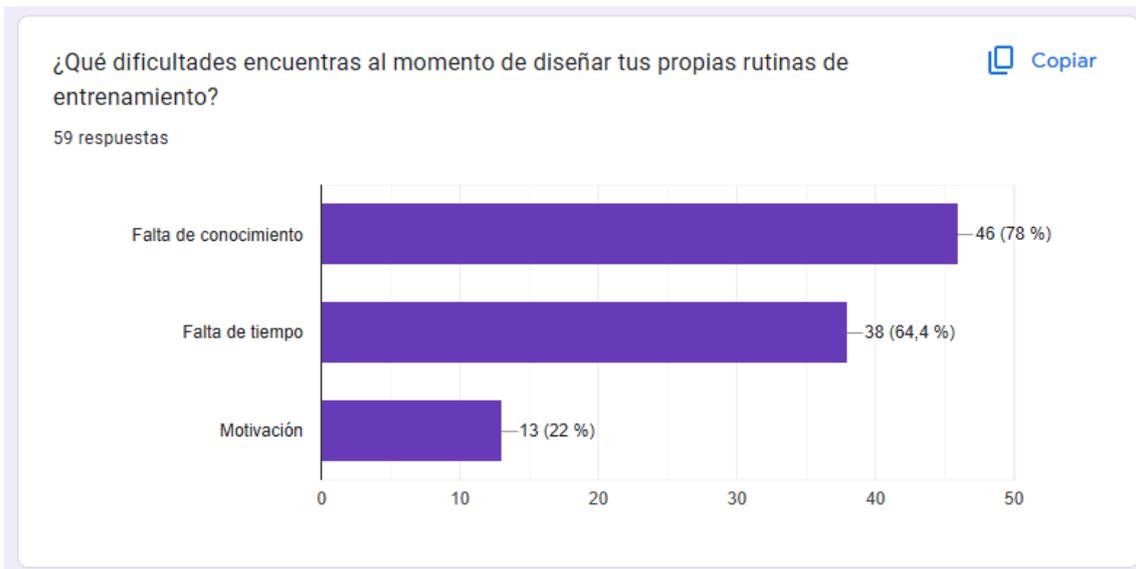


Figura A.9: Respuesta a la novena Pregunta del Cuestionario

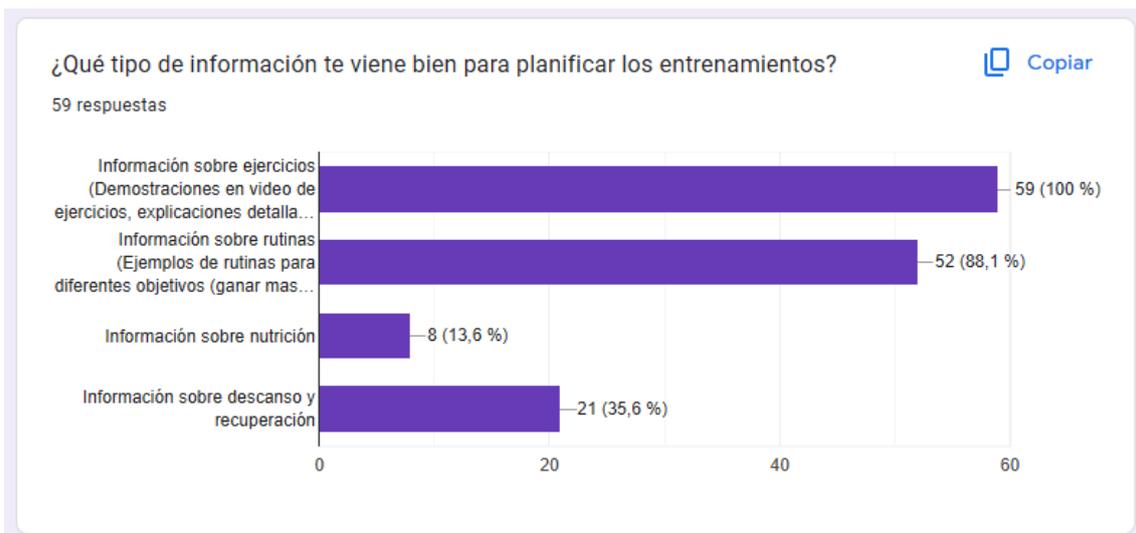


Figura A.10: Respuesta a la décima Pregunta del Cuestionario

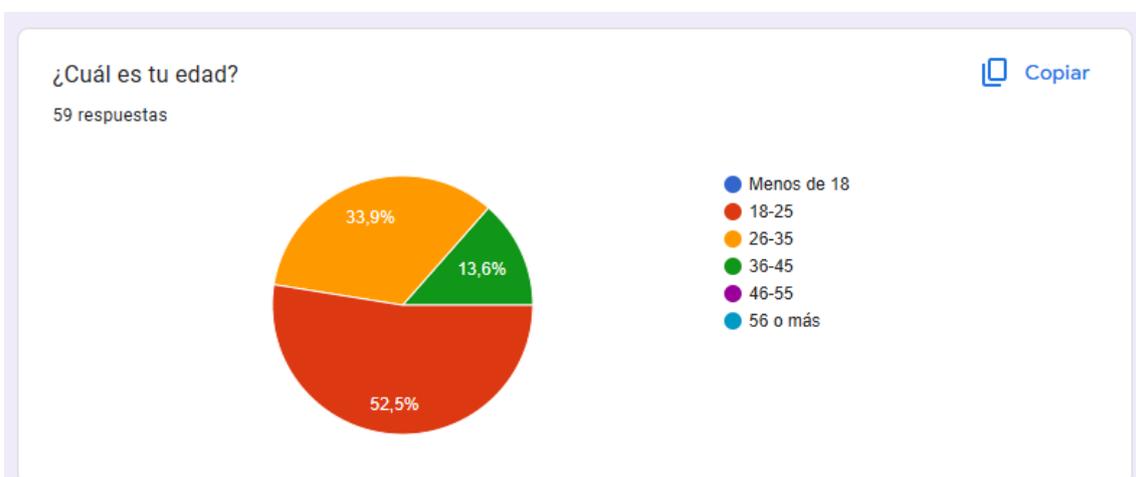


Figura A.11: Respuesta a la undécima Pregunta del Cuestionario

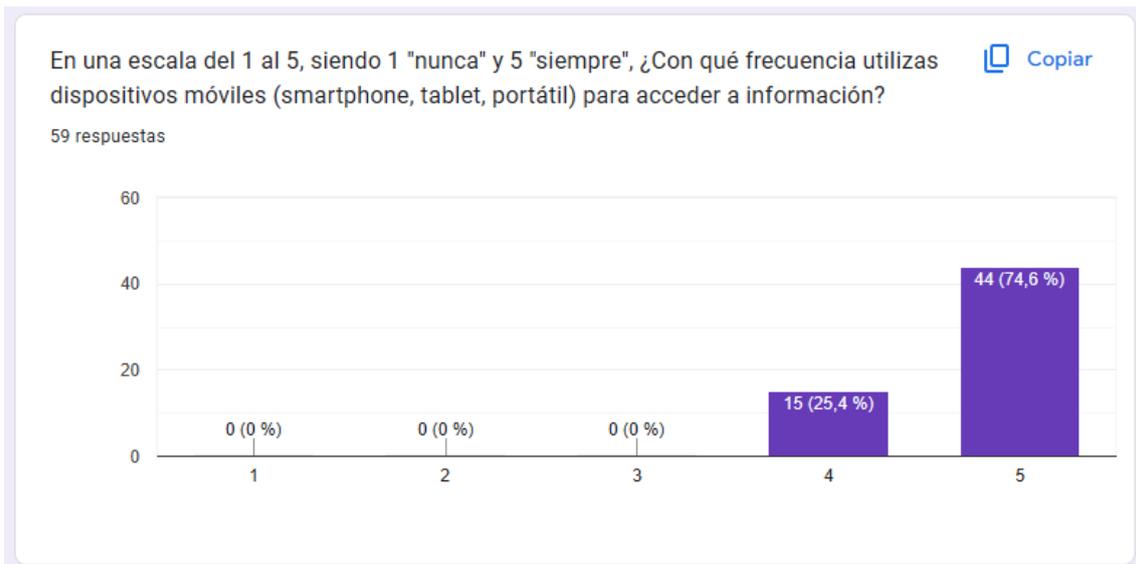


Figura A.12: Respuesta a la duodécima Pregunta del Cuestionario



Figura A.13: Respuesta a la decimotercera Pregunta del Cuestionario

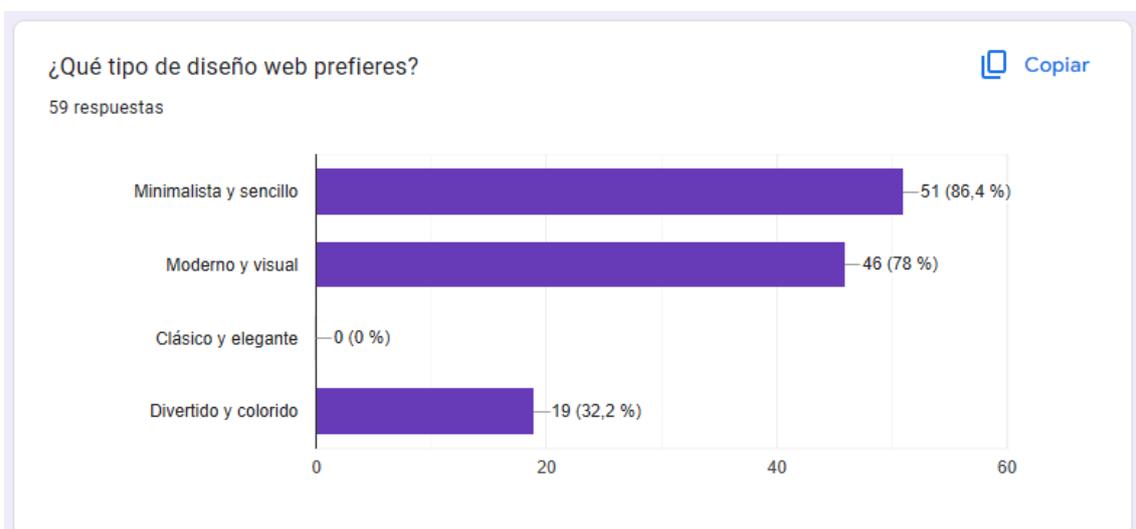


Figura A.14: Respuesta a la decimocuarta Pregunta del Cuestionario

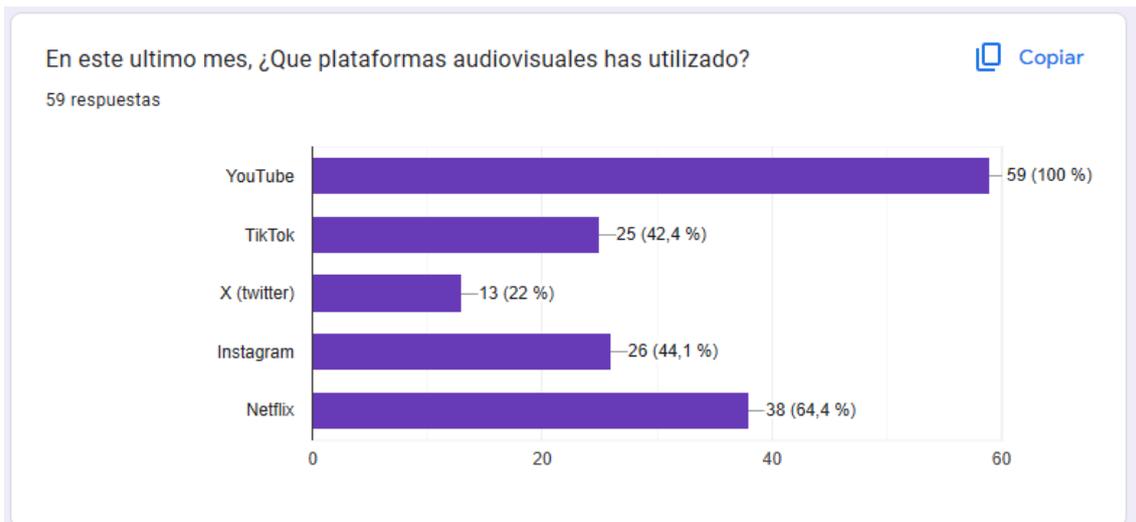


Figura A.15: Respuesta a la decimoquinta Pregunta del Cuestionario

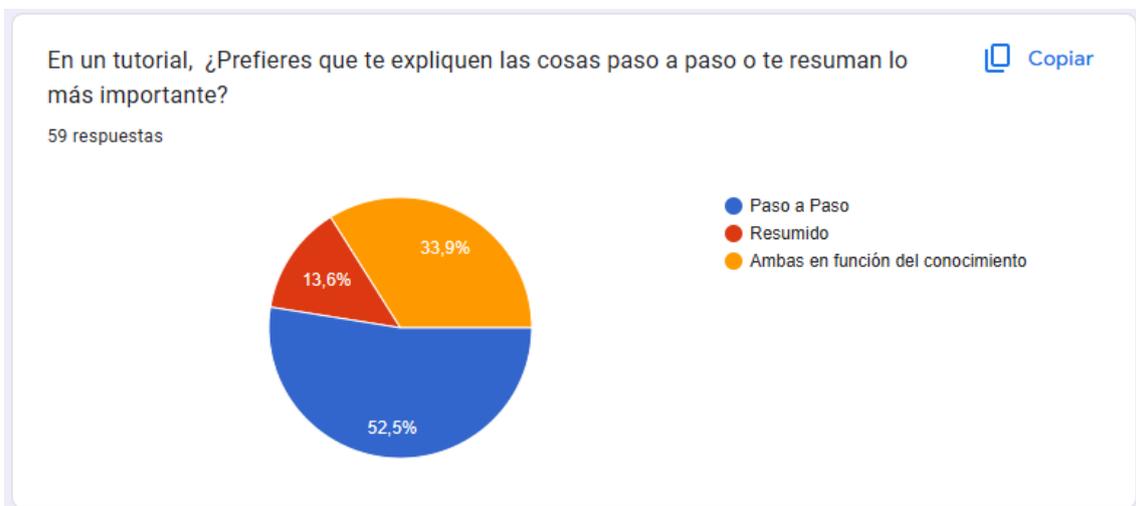


Figura A.16: Respuesta a la decimosexta Pregunta del Cuestionario

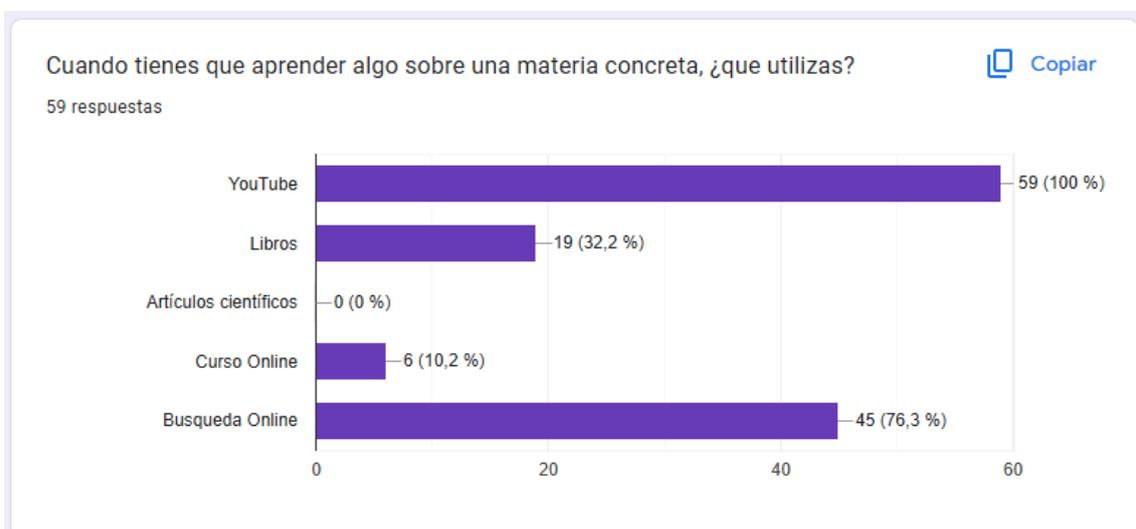


Figura A.17: Respuesta a la decimoséptima Pregunta del Cuestionario

B ODS

OBJETIVOS DE DESARROLLO SOSTENIBLE

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

Objetivos de Desarrollo Sostenible	Alto	Medio	Bajo	No procede
ODS 1. Fin de la pobreza.				X
ODS 2. Hambre cero.				X
ODS 3. Salud y bienestar.	X			
ODS 4. Educación de calidad.		X		
ODS 5. Igualdad de género.				X
ODS 6. Agua limpia y saneamiento.				X
ODS 7. Energía asequible y no contaminante.				X
ODS 8. Trabajo decente y crecimiento económico.				X
ODS 9. Industria, innovación e infraestructuras.				X
ODS 10. Reducción de las desigualdades.				X
ODS 11. Ciudades y comunidades sostenibles.				X
ODS 12. Producción y consumo responsables.				X
ODS 13. Acción por el clima.				X
ODS 14. Vida submarina.				X
ODS 15. Vida de ecosistemas terrestres.				X
ODS 16. Paz, justicia e instituciones sólidas.				X
ODS 17. Alianzas para lograr objetivos.				X

Reflexión sobre la relación del TFG/TFM con los ODS y con el/los ODS más relacionados.

De los anteriores Objetivos de Desarrollo Sostenible (ODS) mencionados, el proyecto desarrollado en este Trabajo de Fin de Grado está estrechamente relacionado con dos de ellos: Salud y Bienestar (ODS 3) y Educación de Calidad (ODS 4).

Salud y Bienestar (ODS 3).

El ODS 3 se centra en garantizar una vida saludable y promover el bienestar para todos en todas las edades. Este proyecto contribuye directamente a este objetivo al proporcionar una herramienta que ayuda a los usuarios a mantenerse físicamente activos y a gestionar sus rutinas de ejercicio de manera efectiva. La aplicación permite a los usuarios crear y personalizar rutinas de entrenamiento, acceder a guías y tutoriales sobre ejercicios específicos, y realizar un seguimiento de su progreso a lo largo del tiempo. Al facilitar el acceso a estos recursos, la aplicación fomenta la adopción de hábitos saludables, lo que puede prevenir enfermedades crónicas como la obesidad, la diabetes y la hipertensión.

Además, el proyecto incorpora elementos educativos que permiten a los usuarios comprender mejor los beneficios de la actividad física y la nutrición, lo que les capacita para tomar decisiones informadas sobre su salud. Este enfoque preventivo está alineado con la meta de reducir la carga de enfermedades no transmisibles mediante la promoción de estilos de vida saludables. La capacidad de la aplicación para adaptar las rutinas de ejercicio a las necesidades individuales de los usuarios también es crucial, ya que reconoce que cada persona tiene diferentes niveles de habilidad y objetivos, lo que refuerza el enfoque inclusivo del ODS 3.

Educación de Calidad (ODS 4)

El ODS 4 tiene como objetivo garantizar una educación inclusiva y equitativa de calidad, promoviendo oportunidades de aprendizaje a lo largo de la vida. Aunque la aplicación desarrollada se centra en el ámbito de la salud y el bienestar, su contribución a la educación es significativa. La aplicación no solo proporciona información básica sobre ejercicios y técnicas, sino que también ofrece contenido educativo más profundo, como consejos sobre nutrición y programas de entrenamiento estructurados. Este conocimiento es esencial para que los usuarios puedan maximizar los beneficios de sus entrenamientos y evitar lesiones, promoviendo así una educación práctica y de calidad.

La accesibilidad del contenido es otro aspecto clave que relaciona este proyecto con el ODS 4. La aplicación está diseñada para ser fácil de usar y está disponible para cualquier persona con acceso a internet, lo que democratiza el acceso al conocimiento sobre salud y bienestar. Esto es particularmente importante en un mundo donde el acceso a información confiable puede estar limitado por barreras económicas o geográficas. Al eliminar estas barreras, la aplicación contribuye a la equidad en la educación y asegura que más personas puedan beneficiarse de una educación de calidad en temas relacionados con su bienestar físico y mental.