



SocialDumbbell

Guillermo Llosa Alonso
2º DAM
2024 - 2025

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN

- 1.1. Contexto
- 1.2. Presentación
- 1.3. Objetivos

2. ANÁLISIS Y DISEÑO DE LA APLICACIÓN

- 2.1. Análisis de requisitos funcionales
 - 2.1.1. Introducción
 - 2.1.2. Tabla numerada de requisitos
- 2.2. Diagrama de casos de uso
 - 2.2.1. Introducción
 - 2.2.2. Diagramas y descripción
- 2.3. Diagrama de interfaces
 - 2.3.1. Introducción
 - 2.3.2. Diagramas y descripción
- 2.4. Análisis de las estructuras de datos utilizadas (bases de datos, ficheros...)
 - 2.4.1. Introducción
 - 2.4.2. Diagramas y descripción
- 2.5. Fase de pruebas
 - 2.5.1. Pruebas de validación de código
 - 2.5.2. Pruebas de implantación en distintos soportes

3. IMPLEMENTACIÓN

- 3.1. Tecnologías utilizadas en el desarrollo del proyecto

4. PUESTA EN MARCHA. DESPLIEGUE

5. APLICACIÓN EN EL ÁMBITO EMPRESARIAL

- 5.1. Idea de negocio
- 5.2. Planificación de necesidades de RRHH
- 5.3. Planificación de necesidades de producción
- 5.4. Planificación de la Prevención de Riesgos Laborales
- 5.5. Previsiones de inversión y financiación

6. CONCLUSIÓN

- 6.1. Valoración personal del trabajo realizado
- 6.2. Posibles ampliaciones

7. BIBLIOGRAFÍA Y WEBGRAFÍA

8. ANEXOS

- I – Manual de usuario
- II – Código completo del proyecto

1. INTRODUCCIÓN

1.1. Contexto

El auge de las aplicaciones móviles y plataformas sociales ha llegado también al ámbito deportivo. Actualmente existen aplicaciones destinadas a compartir rutinas y consejos de entrenamiento, sin embargo, muy pocas fomentan la competencia directa entre usuarios ni la validación de marcas de fuerza como ocurre en las competiciones oficiales de powerlifting. En este contexto, surge SocialDumbbell, una plataforma que combina las funcionalidades de una app de rutinas con la interacción social y la competición.

1.2. Presentación

SocialDumbbell es una aplicación multiplataforma desarrollada con Spring Boot, Hibernate, Ionic y MySQL, que permite a los usuarios compartir rutinas personalizadas, registrar sus marcas personales en ejercicios de fuerza, competir mediante rankings, y dar likes a perfiles y rutinas de otros usuarios. Además, cuenta con un sistema de administración que verifica los levantamientos subidos para asegurar la fiabilidad de las clasificaciones.

1.3. Objetivos

Los objetivos específicos de este proyecto son:

- Permitir a los usuarios compartir rutinas de entrenamiento personalizadas.
- Registrar y validar marcas personales en los tres ejercicios de fuerza más comunes.
- Fomentar la competencia mediante rankings basados en marcas validadas.
- Posibilitar la interacción social mediante likes en rutinas y perfiles.
- Ofrecer una plataforma que sirva de comunidad fitness.

2. ANÁLISIS Y DISEÑO DE LA APLICACIÓN

2.1. Análisis de requisitos funcionales

2.1.1. Introducción

A continuación se presentan los requisitos funcionales de la aplicación, los cuales definen las funcionalidades esenciales que debe ofrecer el sistema tanto para usuarios como para administradores. Estos requisitos establecen las acciones que los usuarios pueden realizar dentro de la plataforma, como registrarse, iniciar sesión, gestionar su perfil, subir contenido y visualizar información de otros usuarios. Además, incluyen las funciones administrativas necesarias para la gestión de ejercicios y validación de levantamientos.

2.1.2. Tabla numerada de requisitos

Número de requisito	RF1
Nombre de requisito	Registro de usuarios
Descripción del requisito	Permitir a los usuarios registrarse en la aplicación mediante un formulario con sus datos personales.
Casos de uso asociados	Registrarse

Número de requisito	RF2
Nombre de requisito	Inicio de sesión
Descripción del requisito	Al abrir la aplicación, el usuario podrá iniciar sesión introduciendo su correo y contraseña.
Casos de uso asociados	Iniciar sesión

Número de requisito	RF3
Nombre de requisito	Recuperación de contraseña
Descripción del requisito	El usuario podrá recuperar su contraseña mediante la generación y envío de una nueva contraseña por correo electrónico.
Casos de uso asociados	Recuperar contraseña

Número de requisito	RF4
Nombre de requisito	Visualización de perfiles de otros usuarios, con filtros por nombre de usuario, altura y peso, tambien ordenar por likes.
Descripción del requisito	Los usuarios podrán ver los perfiles de otros usuarios registrados en la aplicación.
Casos de uso asociados	Ver usuarios

Número de requisito	RF5
Nombre de requisito	Dar 'me gusta' a otros usuarios
Descripción del requisito	Los usuarios podrán dar "me gusta" a los perfiles de otros usuarios.
Casos de uso asociados	Dar like a perfiles

Número de requisito	RF6
Nombre de requisito	Visualización de rutinas, con búsqueda por nombre, grupo muscular y ejercicios, tambien ordenar por likes.
Descripción del requisito	Los usuarios podrán visualizar rutinas públicas creadas por otros usuarios.
Casos de uso asociados	Ver rutinas

Número de requisito	RF7
Nombre de requisito	Dar 'me gusta' a rutinas
Descripción del requisito	Los usuarios podrán dar "me gusta" a las rutinas que les gusten.
Casos de uso asociados	Dar like a rutinas

Número de requisito	RF8
Nombre de requisito	Compartir por las redes las rutinas
Descripción del requisito	Los usuarios podrán dar compartir por redes a las rutinas que les gusten.
Casos de uso asociados	Compartir rutinas

Número de requisito	RF9
Nombre de requisito	Visualización de la clasificación de levantamientos por tipo de ejercicio
Descripción del requisito	Los usuarios podrán ver la clasificación de levantamientos dividida por tipo de ejercicio (press banca, peso muerto, sentadilla).
Casos de uso asociados	Ver clasificación levantamientos

Número de requisito	RF10
Nombre de requisito	Subida de nuevas rutinas personales
Descripción del requisito	Los usuarios podrán subir sus propias rutinas personales.
Casos de uso asociados	Subir rutinas

Número de requisito	RF11
Nombre de requisito	Subida de levantamientos personales con video
Descripción del requisito	Los usuarios podrán subir vídeos de sus levantamientos personales.
Casos de uso asociados	Subir levantamientos

Número de requisito	RF12
Nombre de requisito	Visualización y edición de los datos del perfil personal
Descripción del requisito	Los usuarios podrán ver y editar sus datos personales en su perfil.
Casos de uso asociados	Editar perfil

Número de requisito	RF13
Nombre de requisito	Visualización de las rutinas y levantamientos propios subidos
Descripción del requisito	Los usuarios podrán ver las rutinas y levantamientos que han subido.
Casos de uso asociados	Ver mi perfil

Número de requisito	RF14
Nombre de requisito	Edición de datos de usuario y cambio de contraseña
Descripción del requisito	Los usuarios podrán editar sus datos y cambiar su contraseña.
Casos de uso asociados	Editar usuario y cambiar contraseña

Número de requisito	RF15
Nombre de requisito	Cierre de sesión
Descripción del requisito	Los usuarios podrán cerrar su sesión de la aplicación.
Casos de uso asociados	Cerrar sesión

Número de requisito	RF16
Nombre de requisito	Subida de nuevos ejercicios a la base de datos
Descripción del requisito	Los administradores podrán subir nuevos ejercicios para que los usuarios los seleccionen en las rutinas.
Casos de uso asociados	Subir ejercicios

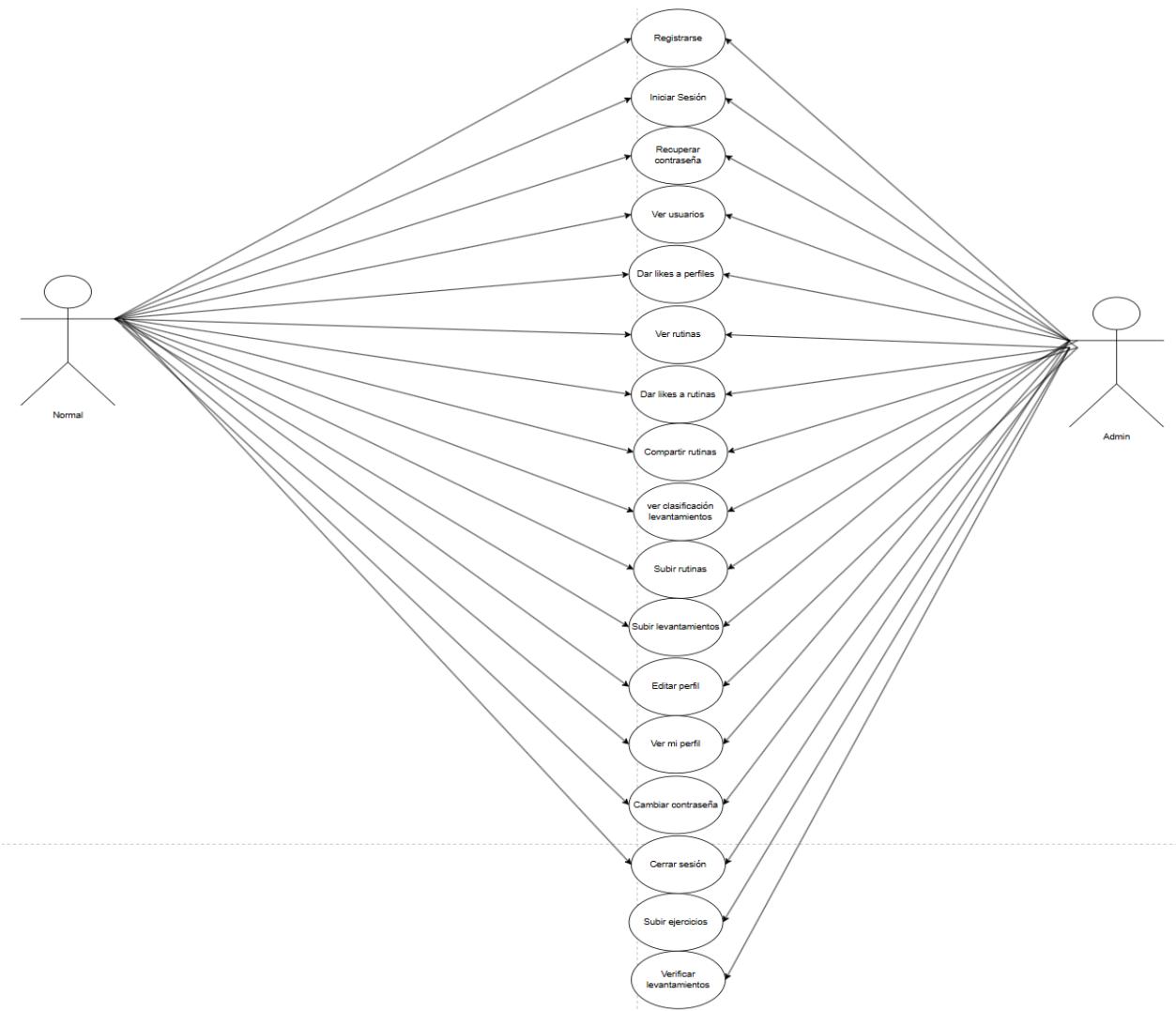
Número de requisito	RF17
Nombre de requisito	Aceptación o rechazo de levantamientos pendientes
Descripción del requisito	Los administradores podrán aceptar o rechazar los videos de levantamientos subidos por usuarios.
Casos de uso asociados	Verificar levantamientos

2.2. Diagrama de casos de uso

2.2.1. Introducción

A través de los casos de uso se especifican las acciones que los usuarios pueden realizar dentro de la aplicación que estarán protegidas mediante token JWT, así como las tareas de gestión que corresponden al administrador.

2.2.2. Diagramas y descripción



2.3. Diagramas de interfaces

2.3.1. Introducción

Explicación del Diagrama de Interfaces del Frontend

La aplicación cuenta con dos áreas principales: la sección pública y la sección protegida, que se accede tras autenticación.

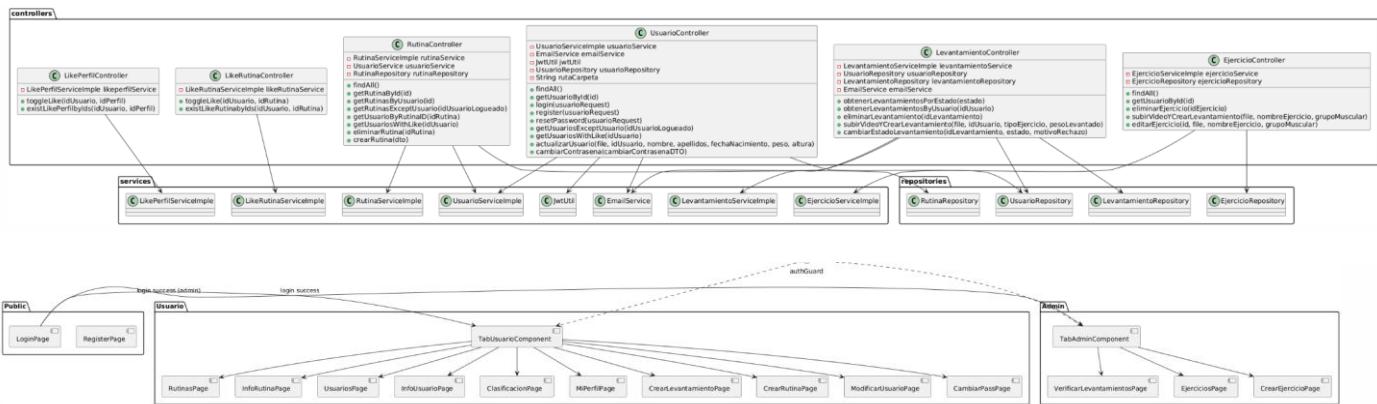
- Rutas públicas: Incluyen las páginas de `login` y `register`, accesibles sin autenticación previa.
- Sección protegida de usuario: Tras iniciar sesión, el usuario accede a la ruta `/usuario`, que carga un componente tipo `tab` (`TabUsuarioComponent`). Este contenedor maneja varias páginas hijas como `rutinas`, `usuarios`, `mi-perfil`, `crear-levantamiento`, entre otras, que permiten gestionar rutinas, ver otros usuarios, editar perfil y registrar levantamientos.
- Sección protegida de administrador: Bajo la ruta `/admin`, accesible solo para usuarios con permisos especiales, se presenta otro componente tipo `tab` (`TabAdminController`). Desde aquí, el administrador puede verificar levantamientos pendientes, gestionar ejercicios y crear o modificar ejercicios.
- Guardias de ruta (`authGuard`): Se utilizan para proteger las rutas internas de usuario y administrador, garantizando que solo usuarios autenticados puedan acceder a esas secciones.
-

Explicación del Diagrama de Controladores y Servicios del Backend

- Controladores: Cada controlador se encarga de manejar las solicitudes HTTP relacionadas con una entidad o funcionalidad específica, como `UsuarioController`, `RutinaController`, `EjercicioController`, entre otros. Estos controladores reciben las peticiones, validan datos y delegan la lógica al servicio correspondiente.
- Servicios: Los servicios (`ServiceImple`) implementan la lógica de negocio, interactuando con los repositorios para acceder a la base de datos y procesar la información según las reglas de la aplicación.
- Repositorios: Son las interfaces que permiten la comunicación directa con la base de datos mediante consultas y operaciones CRUD para las distintas entidades.

- Dependencias: El diagrama muestra claramente cómo cada controlador depende de uno o varios servicios y repositorios para cumplir su función, además de otros servicios auxiliares como el de correo electrónico (EmailService) y seguridad (JwtUtil).

2.3.2. Diagramas y descripción



2.4. Análisis de las estructuras de datos utilizadas (bases de datos, ficheros...)

2.4.1. Introducción

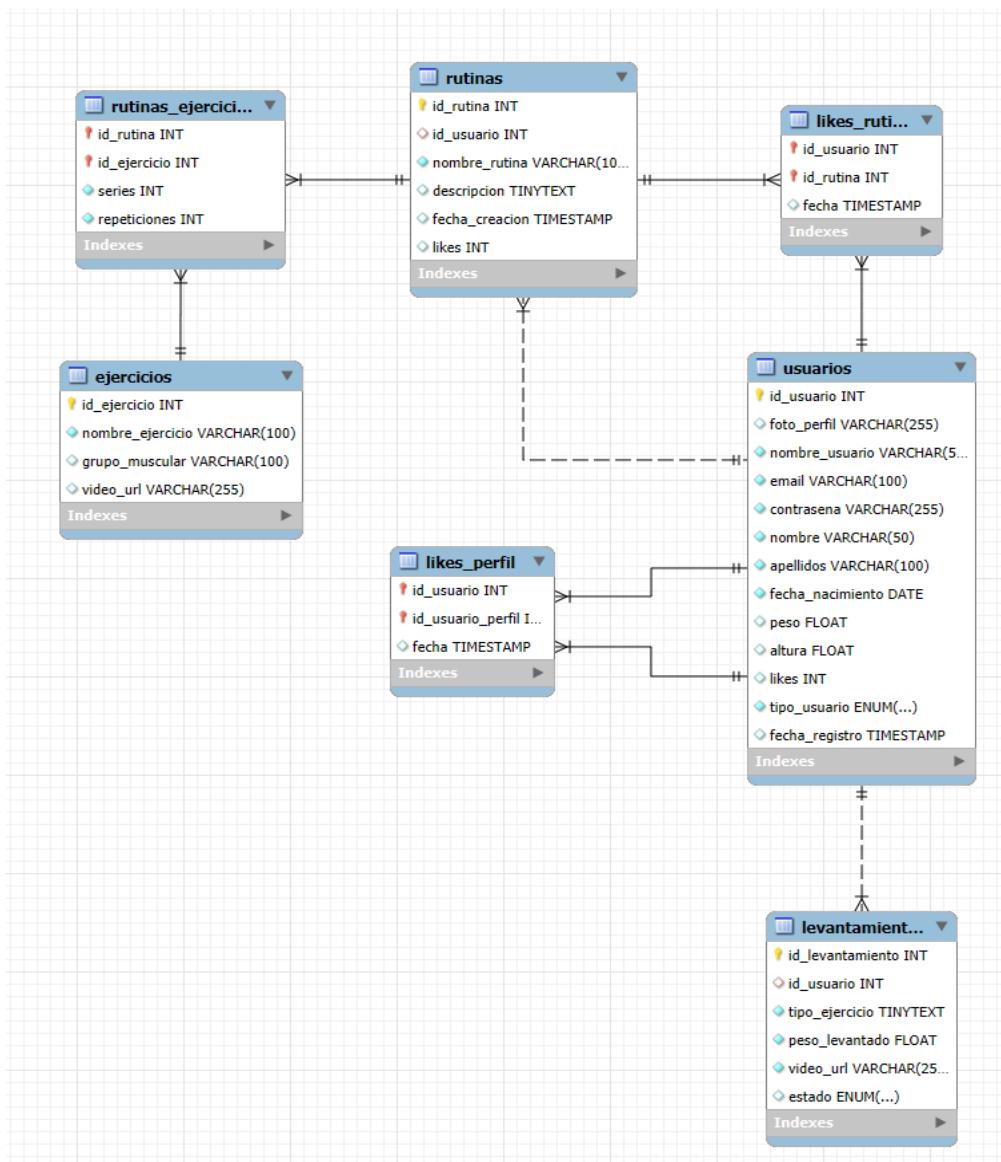
La base de datos socialdumbbell está diseñada para soportar una aplicación social orientada al fitness, donde los usuarios pueden registrarse, crear y compartir rutinas de ejercicio, registrar levantamientos personales y participar en una comunidad interactiva mediante "me gusta" y visualización de perfiles.

Estructura principal:

- usuarios: Esta tabla almacena la información básica de cada usuario registrado, incluyendo sus datos personales (nombre, apellidos, fecha de nacimiento, peso, altura), credenciales (email, contraseña), foto de perfil(el nombre el cual utilizaremos para recoger la imagen almacenada en el contenedor) y tipo de usuario (normal o administrador). También registra la fecha de registro y lleva un conteo de "likes" recibidos en su perfil.
- ejercicios: Contiene los ejercicios disponibles en la aplicación, junto con su grupo muscular asociado y un enlace a un video demostrativo.
- rutinas: Guarda las rutinas creadas por los usuarios, incluyendo el nombre, descripción, fecha de creación, y la cantidad de "likes" recibidos ademas de los ejercicios asociados cuando un usuario crea la rutina. Cada rutina está vinculada a un usuario mediante una clave foránea.
- rutinas_ejercicios: Esta tabla intermedia relaciona las rutinas con los ejercicios que las componen, especificando el número de series y repeticiones para cada ejercicio en una rutina determinada.

- levantamientos: Registra los levantamientos personales que los usuarios suben, detallando el tipo de ejercicio (press de banca, sentadilla, peso muerto), el peso levantado y un video asociado. Cada levantamiento puede estar en estado pendiente, verificado o rechazado, y está vinculado al usuario que lo subió.
- likes_rutina: Tabla que relaciona usuarios con las rutinas a las que han dado "me gusta", guardando la fecha en que se realizó la acción.
- likes_perfil: Similar a la anterior, esta tabla relaciona usuarios con otros perfiles de usuarios a los que han dado "me gusta", almacenando también la fecha.

2.4.2. Diagramas y descripción



2.5. Fase de probe's

2.5.1. Pruebas de validación de código

Las pruebas se realizaron manualmente, simulando distintas situaciones de uso en diferentes dispositivos y navegadores para garantizar la estabilidad de la aplicación. Se comprobaron validaciones de formularios, gestión de errores en el login y registro, correcto funcionamiento de los filtros, ordenaciones, subida de rutinas y levantamientos, así como la interacción mediante likes.

2.5.2. Pruebas de implantación en distintos soportes

Se realizó la ejecución de la aplicación en varios navegadores (Google Chrome, Safari) y en dispositivos móviles Android mediante emuladores y dispositivos físicos, asegurando la correcta visualización y funcionalidad en todos los entornos. Además, se verificó la funcionalidad del modo administrador mediante la subida de ejercicios y la validación de levantamientos.

3. IMPLEMENTACIÓN

Repositorios-

<https://github.com/DAM2A-PDAM-24-25/SocialDumbbell-GuillermoLlosa-Backend>

<https://github.com/DAM2A-PDAM-24-25/SocialDumbbell-GuillermoLlosa-FrontEnd>

3.1. Tecnologías utilizadas en el desarrollo del proyecto

- Spring Boot: framework Java para crear la API REST y gestionar la lógica de negocio.
- Hibernate: ORM para la persistencia de datos con MySQL.
- Ionic: framework basado en Angular para el desarrollo de la interfaz móvil y web.
- MySQL: sistema gestor de bases de datos relacional.
- Visual Studio Code, IntelliJ IDEA y MySQL Workbench como entornos de desarrollo.

4. PUESTA EN MARCHA. DESPLIEGUE

Para el despliegue de la aplicación se utilizó un entorno de contenedores Docker, en el cual se levantaron tanto la API desarrollada con Spring Boot como la base de datos MySQL mediante contenedores independientes. Esto permitió centralizar la configuración y asegurar la portabilidad del entorno de desarrollo y despliegue.

El frontend desarrollado con Ionic se desplegó también en un contenedor Docker, también se generó una APK para dispositivos Android . De este modo, se consiguió un entorno de ejecución unificado, facilitando la puesta en marcha, las pruebas y la portabilidad de la aplicación en diferentes entornos.

5. APLICACIÓN EN EL ÁMBITO EMPRESARIAL

5.1. *Idea de negocio*

SocialDumbbell podría plantearse como una aplicación social deportiva de nicho, destinada a entusiastas del powerlifting y la musculación, donde además de compartir rutinas se pueda competir de forma virtual. Su modelo de negocio podría basarse en una versión gratuita y una versión premium que permita acceso a estadísticas avanzadas o contenido exclusivo.

5.2. *Planificación de necesidades de RRHH*

Para una versión inicial, sería suficiente con:

- 1 desarrollador backend.
- 1 desarrollador frontend.
- 1 persona para soporte y gestión de contenido.

5.3. Planificación de necesidades de producción

Se requeriría un servidor cloud de gama media para alojar la API y la base de datos, además de un entorno de despliegue para la aplicación web y móvil. No se contemplan oficinas físicas, ya que podría gestionarse de manera remota.

5.4. Planificación de la Prevención de Riesgos Laborales

Al tratarse de un proyecto de carácter digital, las medidas de prevención se limitarían al entorno de trabajo virtual: uso de licencias de software seguras, copias de seguridad periódicas y un plan de recuperación de datos.

5.5. Previsiones de inversión y financiación

Se estima una inversión inicial modesta, destinada al alquiler del servidor, dominios y servicios cloud:

- Servidor cloud: 20-30 €/mes.
- Dominio web: 10 €/año.
- Herramientas de desarrollo: 0-50 € según las licencias necesarias.

La financiación podría lograrse mediante recursos propios o ayudas a jóvenes emprendedores.

6. CONCLUSIÓN

6.1. Valoración personal del trabajo realizado

Este proyecto ha supuesto un reto muy completo, combinando el desarrollo backend, la creación de una API REST, la integración con una base de datos relacional y el desarrollo de una interfaz multiplataforma. El mayor aprendizaje ha sido la gestión de los distintos roles y permisos, así como la estructuración de una base de datos relacional eficiente y su integración con Hibernate y Spring Boot. Como dificultad destacable, ha resultado especialmente desafiante la realización de los estilos de la aplicación.

6.2. Posibles ampliaciones

Como mejora futura, se podría incorporar:

- Un sistema de comentarios en rutinas y levantamientos.
- Rankings por países, genero o zonas.
- Integración con dispositivos wearables.
- Notificaciones push.
- Compatibilidad con iOS.
- Reportar rutinas
- Baneo de usuarios
- Internacionalización

7. BIBLIOGRAFÍA Y WEBGRAFÍA

- <https://spring.io/projects/spring-boot>
- <https://hibernate.org>
- <https://ionicframework.com>
- <https://www.mysql.com>
- Documentación oficial de Java, Angular e Ionic.
- Imagen de fondo login y registro - <https://pixabay.com/es/photos/hombre-fuerte-aptitud-física-7847245/>

8. ANEXOS

MANUAL DE USUARIO

Url back: guillermola25.iesmontenaranco.com/8000

Url front: guillermola25.iesmontenaranco.com

La apk de Socialdumbbell requiere de un móvil con sistema android, solo habrá que hacer click en la apk y instalarla.

Usuario administrador

Correo usuario: guimo99@hotmail.com

Contraseña: Contraseña123

Usuario prueba

Correo usuario: prueba@hotmail.com

Contraseña: Contraseña123

Login



Si el campo email esta completo te envía una contraseña nueva al correo

Podrás iniciar sesión completando los datos o ir al registro dándole al link de Regístrate

Rutinas

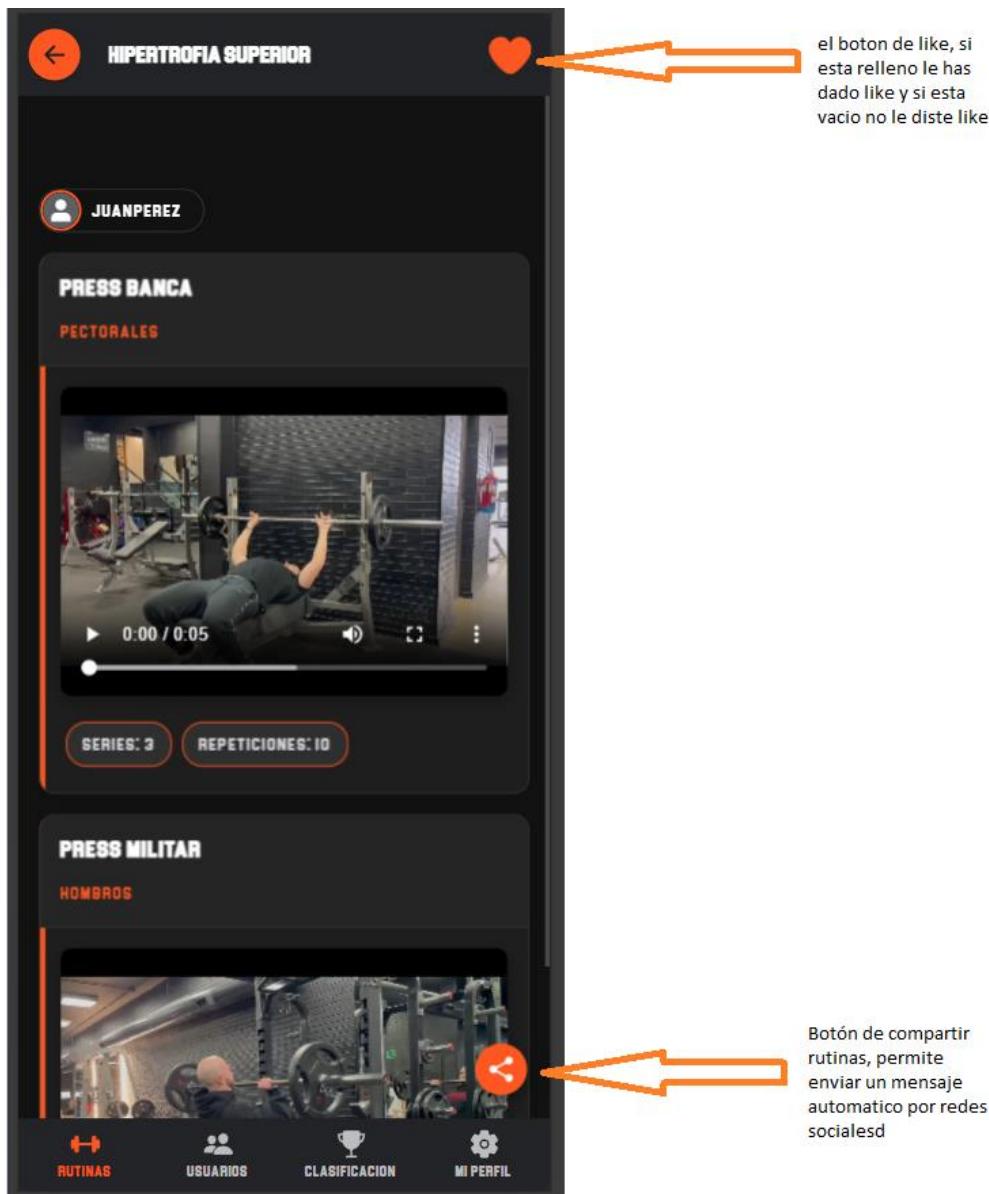


El buscador filtra por nombre de la rutina, nombre de los ejercicios que contiene la rutina y grupos musculares de los ejercicios que contiene la rutina.

El botón Ordenar Por likes ordena de mayor a menor cantidad de likes o al revés según le das.

Si haces click en una rutina navegaras a ella para verla en detalle .

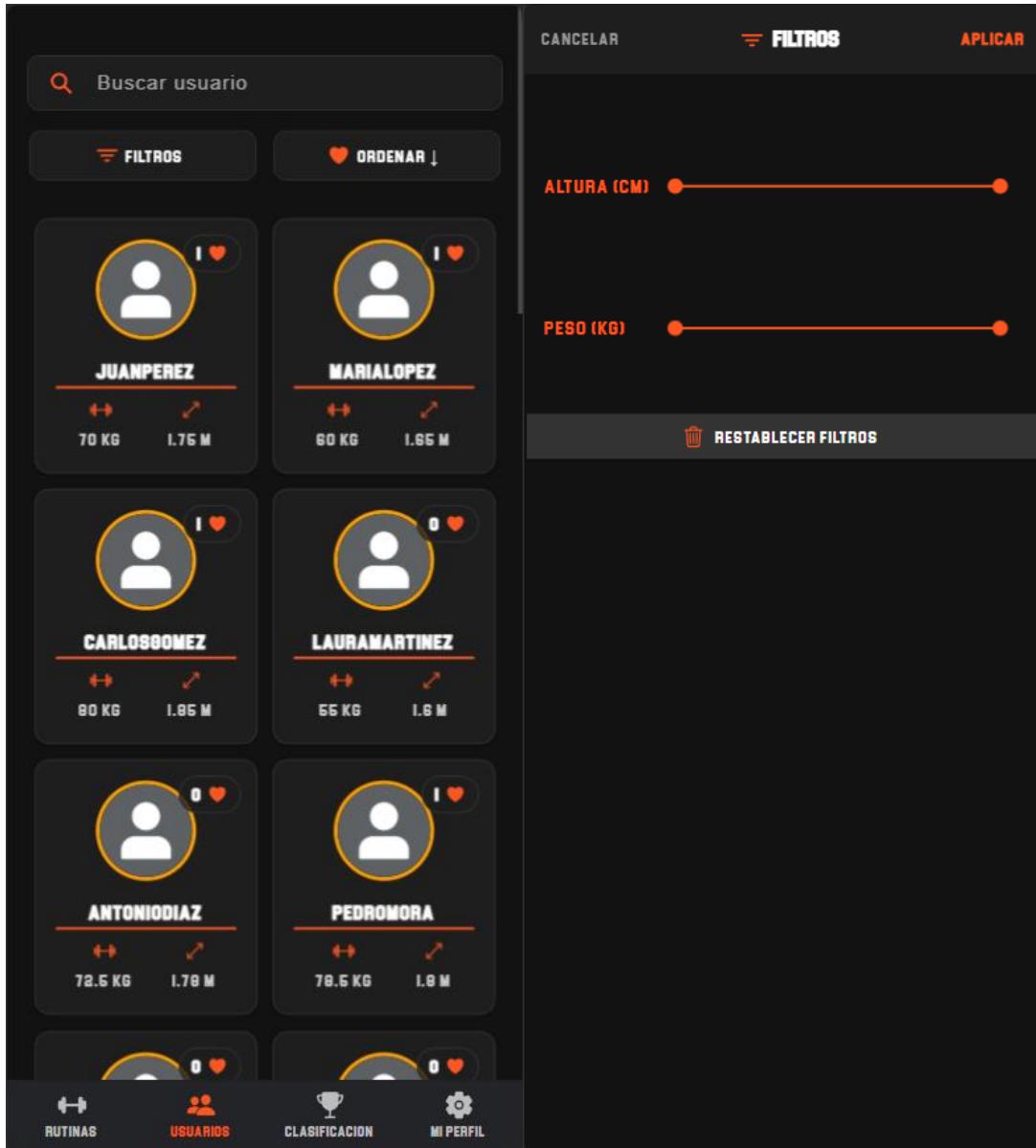
Info Rutina



La rutina contiene los ejercicios que le asigno el usuario que la creó con el video correspondiente al ejercicio de demostración y las series y repeticiones que hay que hacer en cada uno.

Si le das al usuario que aparece su foto y nombre al inicio de la página navegaras al usuario que la creó.

Usuarios



El buscador filtra por nombre de usuario.

El botón de filtros abre el modal de la derecha en el que puedes seleccionar altura y peso para filtrar los usuarios, el botón restablecer filtros elimina los filtros aplicados.

El botón ordenar ordena por likes me mayor a menor o al revés según le des.

Al darle a un usuario navegas a él para ver toda su información.

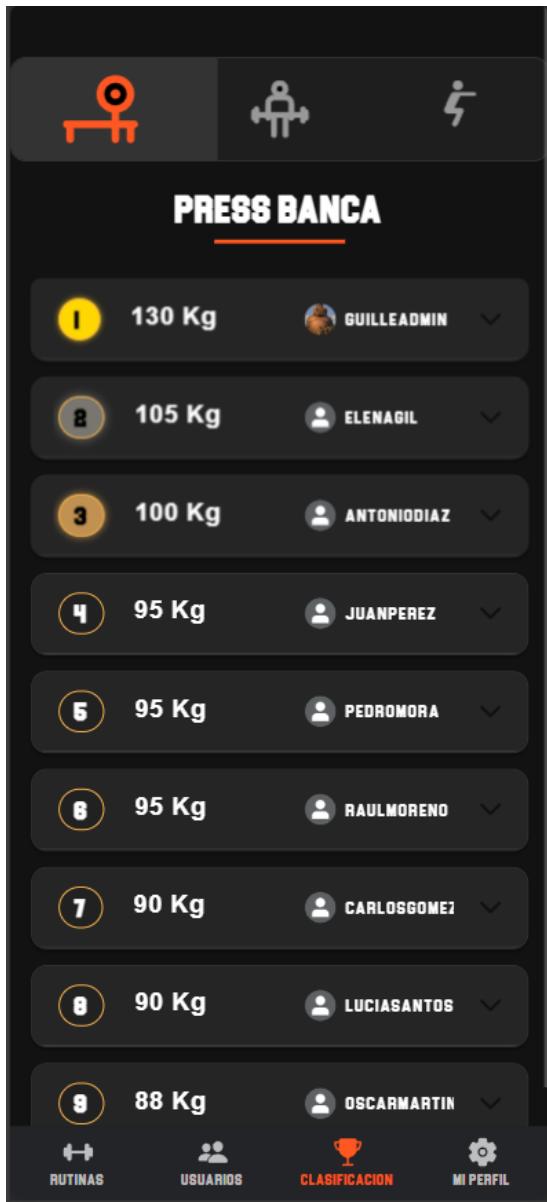
Info Usuario



Puedes visualizar las rutinas y levantamientos del usuario dándole al ícono de la mancuerna o al trofeo además de darle like en el botón de corazón.

Puedes darle al levantamiento para visualizar el video de este o darle a la rutina para ir a la info de ella.

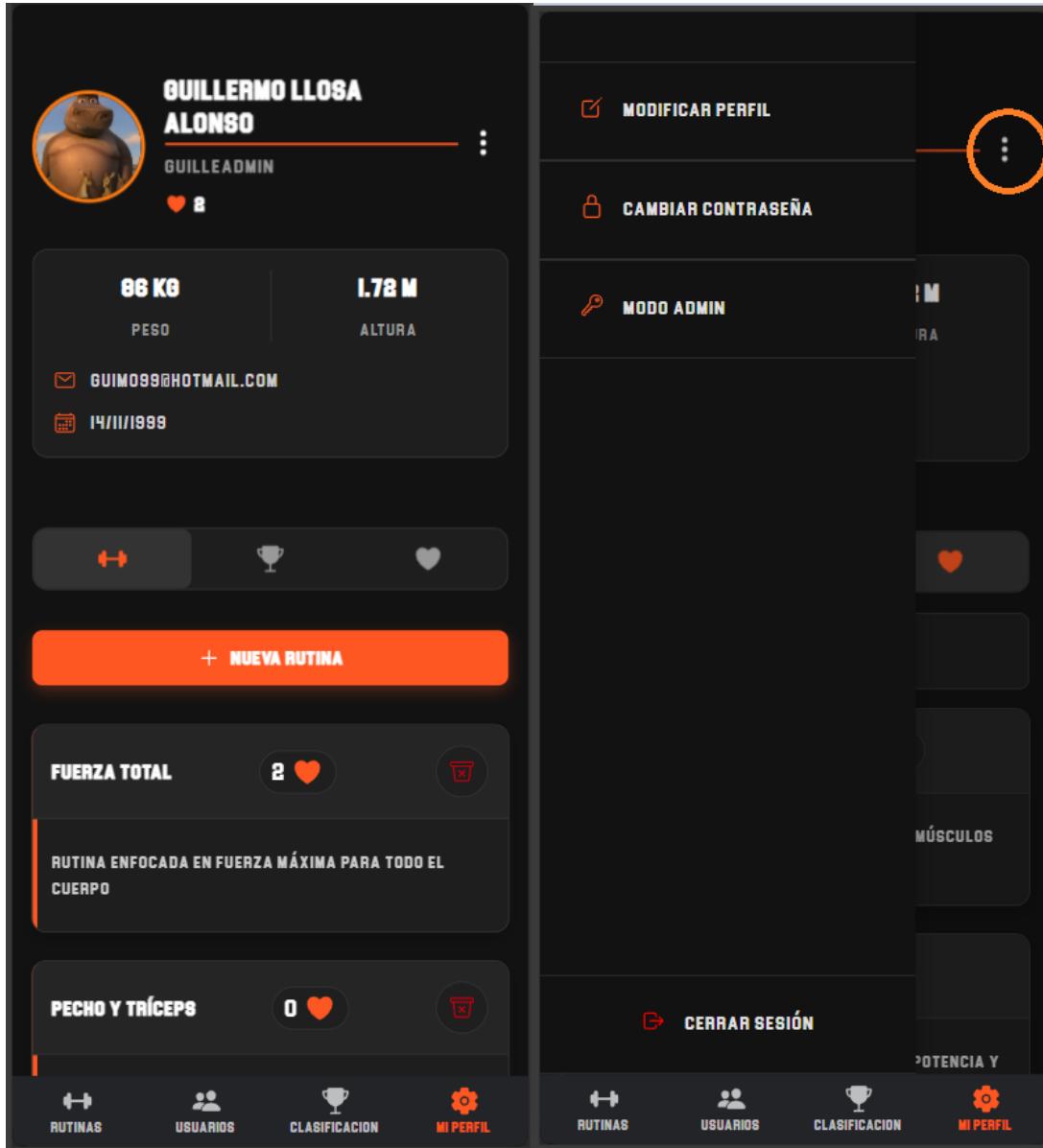
Clasificación



Podrás seleccionar uno de los tres ejercicios básicos dándole a los iconos de arriba y visualizar los levantamientos que haya subidos de cada uno.

Si le das al levantamiento se despliega el video de este.

Mi perfil



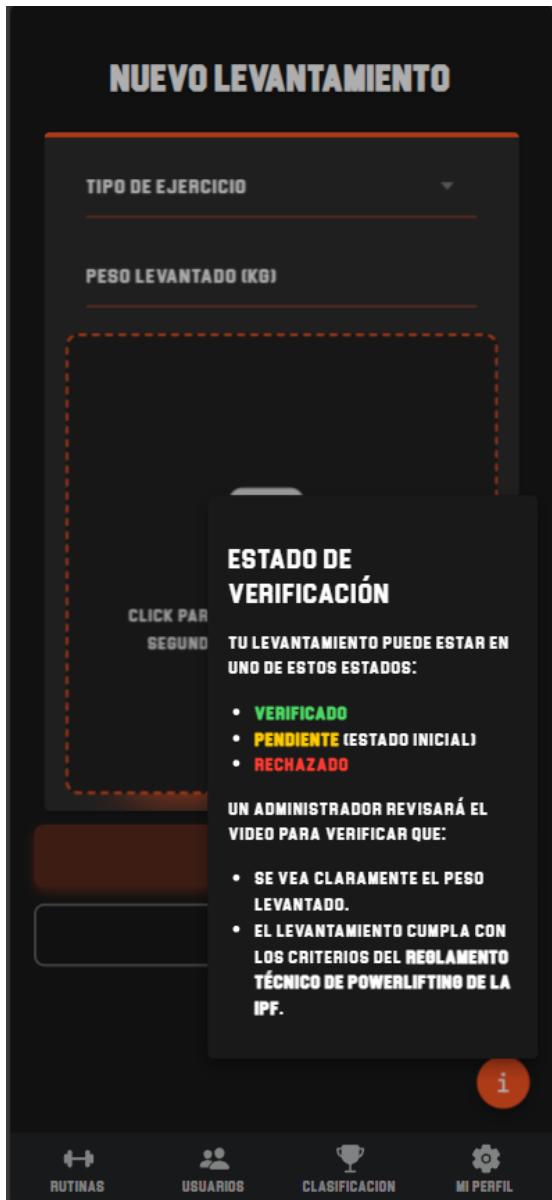
Podrás ver tus rutinas creadas, además de poder crear otra o borrar, lo mismo con los levantamientos donde también podrás ver el estado del levantamiento (pendiente, rechazado, verificado).

Tambien puedes visualizar los likes que has dado a usuarios y rutinas.

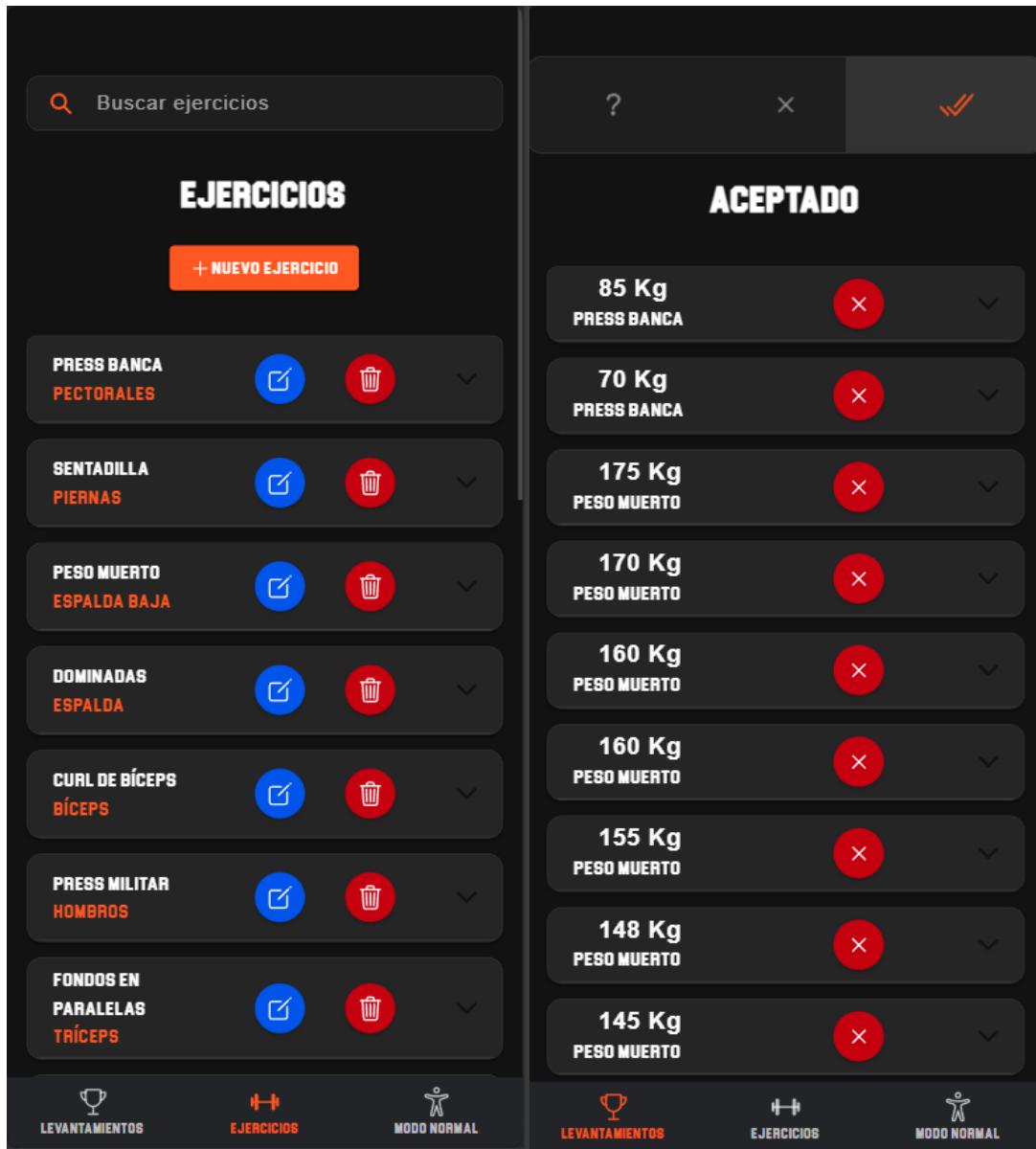
En el boton que esta marcado con un circulo naranja despliegas un menu.

La opción de modo admin solo aparece si eres administrador.

Nuevo levantamiento

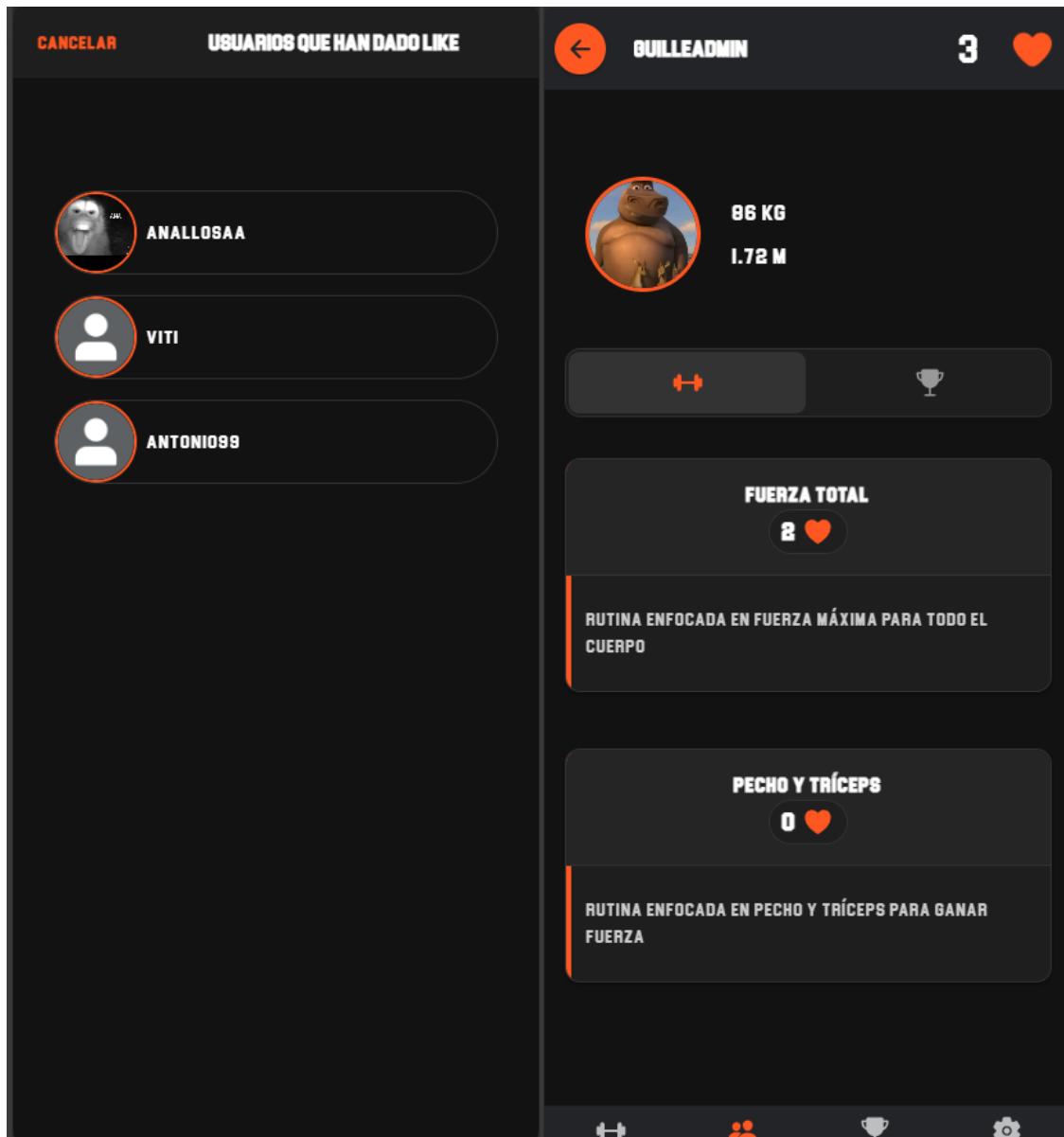


En el botón con la i de información muestra las condiciones para verificar un levantamiento.



En la parte de administrador hay dos pantallas, la de verificación de levantamientos en la que podrás visualizarlos y determinas si aceptas o rechazas los levantamientos.

En la de ejercicios puedes añadir nuevos ejercicios, filtrar por nombre de ejercicio o editar los ejercicios o eliminarlos.



Tanto en la información de la rutina como del usuario, si haces click en el número de likes, podrás ver quien ha dado like a ese perfil.

Documentación Back

<https://guillermola25.iesmontenaranco.com:8000/swagger-ui/index.html#/>