

ANÁLISIS Y DISEÑO

Joan Casanovas | Rene Oviedo
Jerrarquixs 2024-25

ÍNDICE

ÍNDICE	2
INTRODUCCIÓN	3
TIPOS DE USUARIOS	4
REQUISITOS PARA EL FUNCIONAMIENTO DE LA APLICACIÓN	5
Requisitos funcionales	5
Requisitos no funcionales	5
INFORMACIÓN A GUARDAR EN LA BASE DE DATOS	6
MODELO ENTIDAD/RELACIÓN	7
MODELO RELACIONAL	8
CASOS DE USO	9

INTRODUCCIÓN

Este documento contiene el análisis y diseño del proyecto JoreFIT, una aplicación web destinada a ofrecer rutinas de entrenamiento personalizadas según los objetivos y el nivel físico de cada usuario.

El objetivo principal de esta aplicación es permitir a los usuarios registrarse, seleccionar un plan de entrenamiento (como ganancia muscular, definición, pérdida de peso o ejercicios cardiovasculares) y recibir una rutina adaptada a su nivel físico (aficionado, intermedio o avanzado). Además, podrán visualizar su progreso, modificar su plan y completar los ejercicios asignados.

Los apartados que componen este documento son:

- ★ **TIPOS DE USUARIOS:** Se describen los perfiles que utilizarán la aplicación (como administrador o cliente) y sus respectivas funcionalidades.
- ★ **MODELO DE DATOS:** Se presenta la estructura general de la base de datos, incluyendo las entidades clave y relaciones.
- ★ **FUNCIONALIDAD DE LA APLICACIÓN:** Se detallan las acciones que podrá realizar cada tipo de usuario, basadas en los objetivos del sistema.
- ★ **MAPAS DE NAVEGACIÓN:** Se describe cómo estará organizada la navegación por la aplicación y se muestran los primeros bocetos o esquemas de las pantallas principales.



TIPOS DE USUARIOS

En la plataforma JoreFIT existen dos tipos principales de usuarios, cada uno con permisos y funcionalidades específicas según su rol dentro del sistema.

Administrador

Es el usuario responsable de gestionar el correcto funcionamiento de la aplicación. Tiene acceso completo a todas las funcionalidades internas. Puede realizar las siguientes acciones:

- ★ Crear, modificar y eliminar ejercicios disponibles en el sistema.
- ★ Añadir, editar o eliminar planes de entrenamiento.
- ★ Visualizar y gestionar la lista completa de usuarios registrados.

Usuario

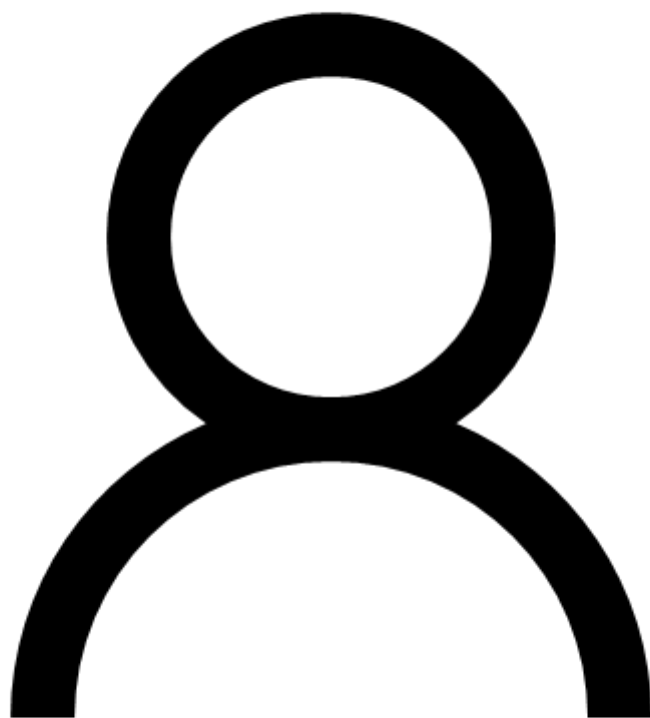
Este tipo de usuario representa a los clientes que utilizarán la plataforma para seguir sus rutinas de entrenamiento. Podrán:

- ★ Registrarse e iniciar sesión en el sistema.
- ★ Completar y modificar su perfil personal (peso, altura, objetivo, nivel).
- ★ Seleccionar un plan de entrenamiento entre las opciones disponibles.
- ★ Visualizar los ejercicios asignados según su nivel.
- ★ Marcar ejercicios como completados y consultar su progreso.
- ★ Cambiar su plan si así lo desea.

Seguridad de la aplicación

Para garantizar la privacidad y el acceso seguro a la plataforma, se implementarán las siguientes medidas de seguridad:

- ★ Encriptación de datos sensibles: Las contraseñas se almacenarán de forma segura en la base de datos utilizando algoritmos de hash como bcrypt. La información de tarjetas bancarias, si se guarda, será cifrada cumpliendo las normativas de seguridad.
- ★ Control de acceso por rol: El sistema limitará las funcionalidades según el tipo de usuario. Por ejemplo, un cliente no podrá acceder a funciones de administración ni ver datos de otros usuarios.
- ★ Autenticación y gestión de sesiones: Los usuarios deberán iniciar sesión con su correo electrónico y contraseña. Las sesiones estarán protegidas.
- ★ Recuperación de cuenta: En caso de olvidar la contraseña, se enviará un código temporal al correo registrado para permitir su recuperación de forma segura.



REQUISITOS PARA EL FUNCIONAMIENTO DE LA APLICACIÓN

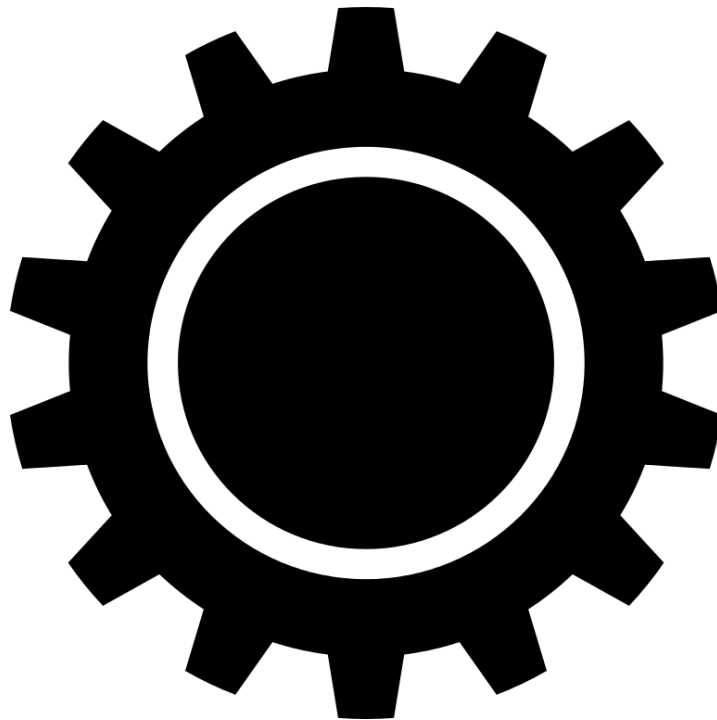
En este apartado, se describen detallada y ampliamente los distintos requisitos tanto funcionales como no funcionales que afectarán al correcto funcionamiento del software.

Requisitos funcionales

- ★ Permitir a los distintos usuarios registrarse en el sitio web.
- ★ Permitir el inicio de sesión de los distintos usuarios.
- ★ El usuario podrá personalizar la rutina generada.
- ★ El usuario debe poder modificar su perfil y actualizar información personal.
- ★ Cada ejercicio debe incluir un número de repeticiones, series.
- ★ Hay múltiples tipos de entrenamiento (Definición, Ganancia Muscular, etc.).

Requisitos no funcionales

- ★ Interfaz amigable y fácil de usar.
- ★ Aplicación accesible.

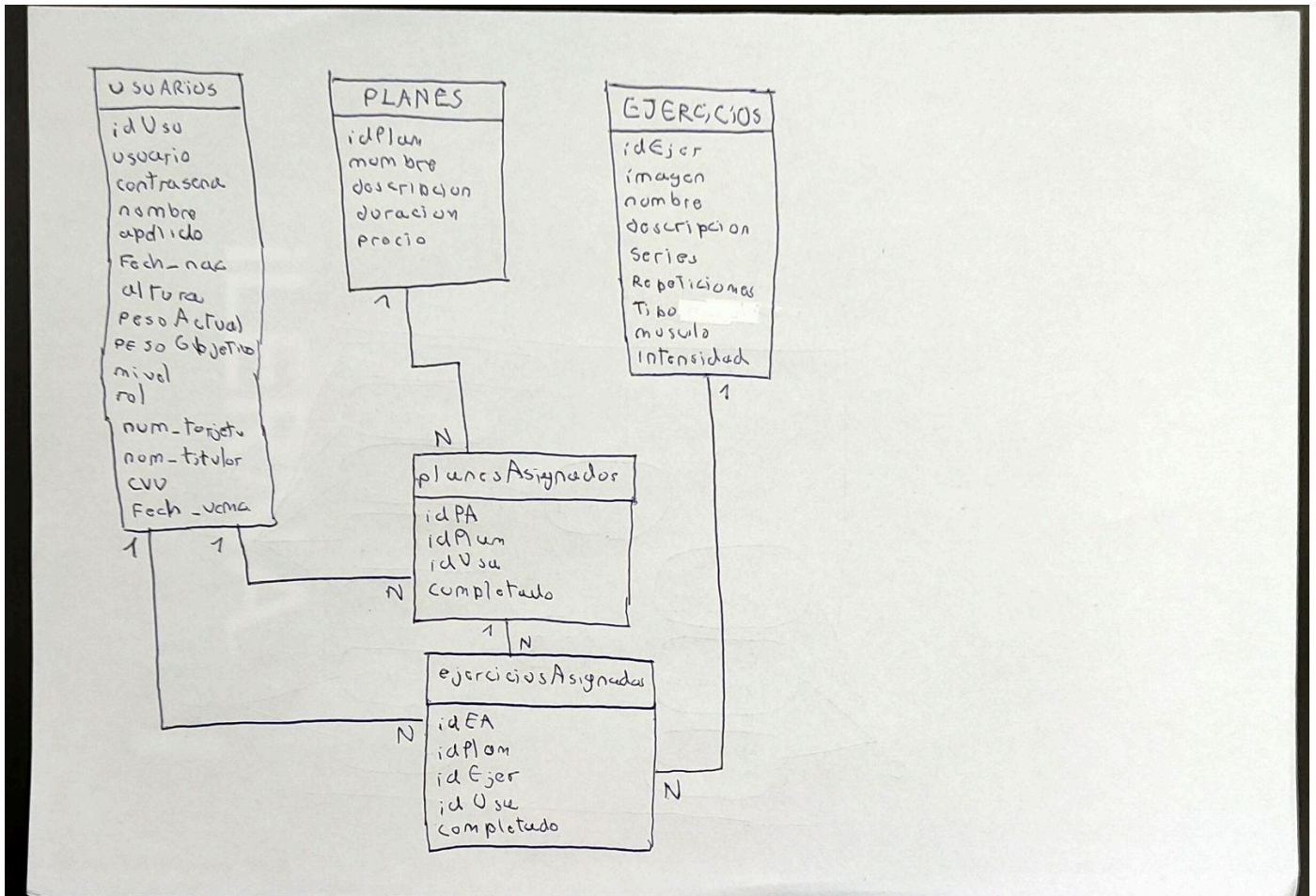


INFORMACIÓN A GUARDAR EN LA BASE DE DATOS

Tabla	Campos	Descripción
Usuarios	<ul style="list-style-type: none"> • id: BIGINT, auto incremental, PK, NN. • correo: STRING, único (máx. 255), UK, NN. Validación: formato email. • contraseña: STRING, NN. Validación: mínimo 8 caracteres. • nombre: STRING(50), NN. • apellido: STRING(50), NN. • fecha_nacimiento: DATE, NN. • altura: DOUBLE, NULLABLE. • pesoActual: DOUBLE, NN. • pesoObjetivo: DOUBLE, NN. • nivel: STRING, NN. Validación: debe ser uno de ['aficionado', 'intermedio', 'avanzado']. • rol: STRING, NN. Validación: debe ser 'cliente' o 'admin'. • numero_tarjeta: STRING(20), NULLABLE. Validación: numérica de 16 dígitos. • nombre_titular: STRING(100), NULLABLE. • cvv: STRING(3), NULLABLE. Validación: 3 dígitos numéricos. • fecha_vencimiento: STRING, NULLABLE. Validación: formato MM/AA. • recovery_code: STRING, NULLABLE. • code_expires_at: TIMESTAMP, NULLABLE. 	Esta tabla almacena información básica de los usuarios que interactúan con el sistema.
Planes	<ul style="list-style-type: none"> • id: BIGINT, auto incremental, PK, NN. • nombre: STRING(100), NN. • descripcion: STRING(255), NN. • duracion: INTEGER sin signo, 	Esta tabla define los diferentes planes de ejercicios disponibles.

	NN. Validación: mayor que 0. • precio: DOUBLE(8,2), NN. Validación: ≥ 0 . • imagen: TEXT, NN.	
Ejercicios	• id: BIGINT, auto incremental, PK, NN. • imagen: TEXT, NN. • nombres: STRING(100), NN. • descripcion: TEXT, NN. • series: INTEGER sin signo, NN. Validación: > 0 . • repeticiones: STRING, NN. Validación: formato "X reps" o rango tipo "10-12". • tipo: STRING(100), NN. Validación: debe coincidir con un plan. • musculo: STRING(50), NN. • intensidad: STRING, NN. Validación: debe ser uno de ['leve', 'media', 'alta'].	Esta tabla contiene la información sobre cada ejercicio disponible.
planesAsignados	• id: BIGINT, auto incremental, PK, NN. • idPlan: BIGINT, FK, NN → referencia a planes.id. • idUsu: BIGINT, FK, NN → referencia a usuarios.id. • completado: BOOLEAN, NN, valor por defecto false.	Esta tabla registra los planes asignados a cada usuario y su estado.
ejerciciosAsignados	• id: BIGINT, auto incremental, PK, NN. • idPlan: BIGINT, FK, NN → planes.id. • idEjer: BIGINT, FK, NN → ejercicios.id. • idUsu: BIGINT, FK, NN → usuarios.id. • completado: BOOLEAN, NN, valor por defecto false.	Esta tabla registra los ejercicios asignados a los usuarios en un plan específico, junto con su estado.

MODELO ENTIDAD/RELACIÓN



MODELO RELACIONAL

Tabla: usuarios

USUARIOS(idUsu (PK), usuario, contrasena, nombre, apellido, fecha_nacimiento, altura, pesoActual, pesoObjetivo, nivel, rol, numero_tarjeta, nombre_titular, cvv, fecha_venc)

Tabla: planes

PLANES(idPlan (PK), nombre, descripcion, duracion, precio)

Tabla: ejercicios

EJERCICIOS(idEjer (PK), imagen, nombres, descripcion, series, repeticiones, tipo, musculo, intensidad)

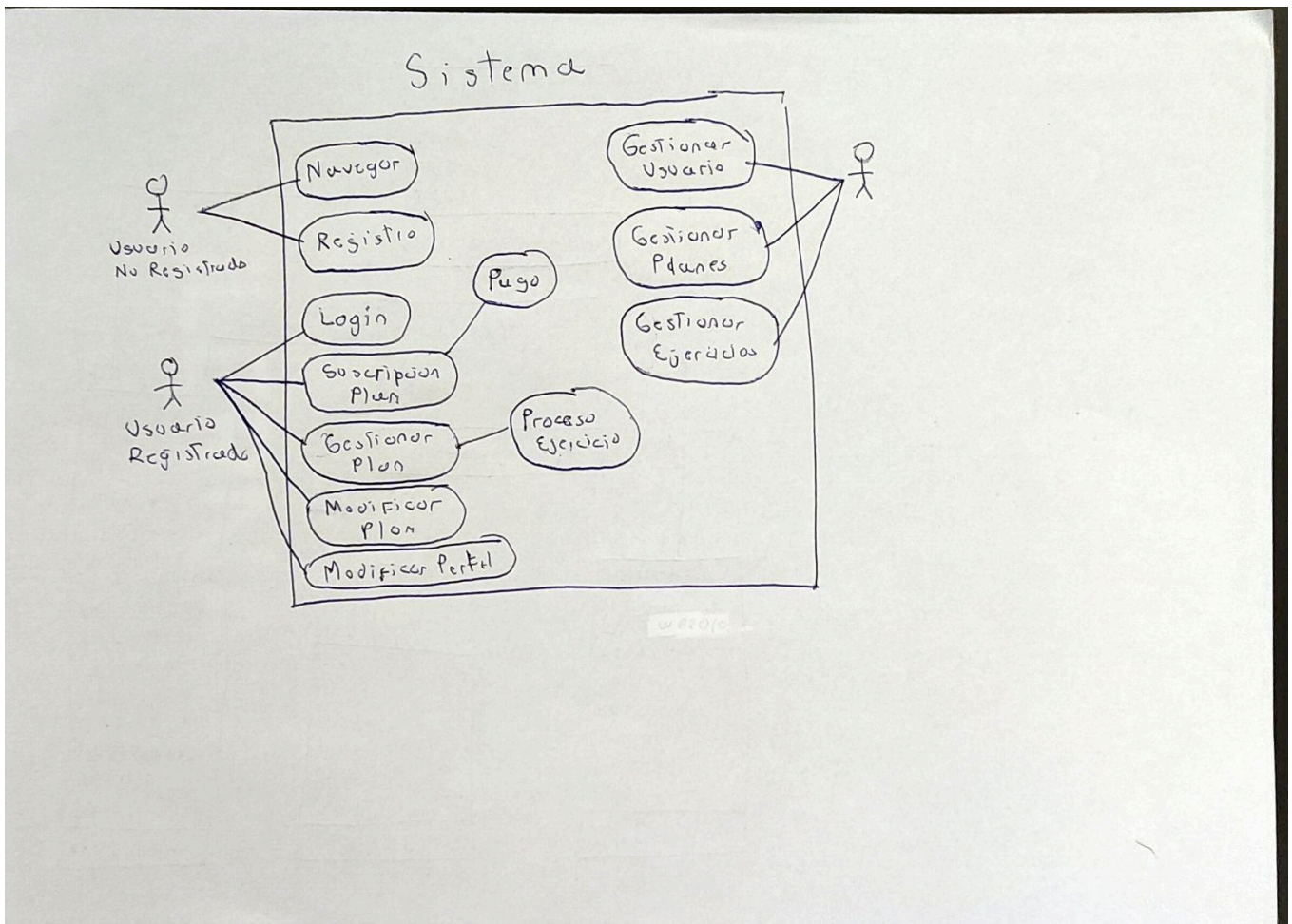
Tabla: planesAsignados

PLANES_ASIGNADOS(idPA (PK), idPlan (FK → PLANES.idPlan), idUsu (FK → USUARIOS.idUsu), completado)

Tabla: ejerciciosAsignados

EJERCICIOS_ASIGNADOS(idEA (PK), idPlan (FK → PLANES.idPlan), idEjer (FK → EJERCICIOS.idEjer), idUsu (FK → USUARIOS.idUsu), completado)

CASOS DE USO



DESCRIPCIÓN DE LOS CASOS DE USO

USUARIO NO REGISTRADO

- Caso de uso: NAVEGAR

Funcionalidad: Permitir al usuario no registrado explorar la página principal y acceder a información pública sin necesidad de crear cuenta.

Actor – Usuario no registrado

Entradas – Ninguna

Validaciones – Ninguna

Salidas – Listado de pantallas públicas: descripción del servicio, planes disponibles (solo vista), enlaces a registro/login.

Acceso a BD – Lectura: tablas, planes y ejercicios para mostrar resúmenes (nombre, imagen, precio).

Procesos internos – None

Almacenamiento – Ninguno

Errores controlados – Ninguno

Mensajes – No aplica

USUARIO REGISTRADO

- **Caso de uso: REGISTRO**

Funcionalidad: Dar de alta a un nuevo usuario en la plataforma para que pueda utilizar todas las funciones.

Actor – Usuario no registrado

Entradas

- correo (string)
- contraseña (string)
- nombre, apellido (string)
- fecha_nacimiento (date)
- peso, altura, nivel (string)

Validaciones

- Formato de correo válido y no duplicado en usuarios.
- Contraseña ≥ 8 caracteres.
- Fecha de nacimiento \leq fecha actual.
- Nivel {aficionado, intermedio, avanzado}.

Salidas

- Éxito: mensaje “Registro completado. Revisa tu email para verificar cuenta”.
- Error: “Correo ya registrado”, “Formato de email inválido”, “Contraseña demasiado corta”, etc.

Acceso a BD

- SELECT sobre usuarios para comprobar correo.
- INSERT en usuarios (almacena hash de contraseña y datos personales).

Procesos internos

- Hash de contraseña con bcrypt.
- Generación de código de verificación y fecha de expiración.

Almacenamiento

- Todos los campos de usuarios (incluyendo recovery_code y code_expires_at).

Errores controlados

- Duplicidad de correo.
- Validaciones de formato.

Mensajes

- “Registro completado...”, “Correo ya existe...”, “Contraseña inválida...”.

- **Caso de uso: LOGIN**

Funcionalidad: Autenticar a un usuario existente para permitirle acceso a su área privada.

Actor – Usuario registrado

Entradas

- correo (string)
- contraseña (string)

Validaciones

- Correo existe en usuarios.
- Comparación de contraseña con hash almacenado.

Salidas

- Éxito: token de sesión (JWT/Sanctum) y redirección a panel de usuario.
- Error: “Credenciales incorrectas”.

Acceso a BD

- SELECT en usuarios (correo, hash de contraseña).

Procesos internos

- Verificación de hash.
- Generación de token de autenticación.

Almacenamiento

- Registro de token en tabla de tokens (para Sanctum).

Errores controlados

- Usuario no encontrado.
- Contraseña errónea.

Mensajes

- “Login exitoso”, “Usuario o contraseña incorrectos”.

- Caso de uso: SOLICITAR PLAN

Funcionalidad: Permitir al usuario contratar un plan de entrenamiento según su nivel y objetivo.

Actor – Usuario registrado

Entradas

- id_plan (integer)
- id_usuario (del token de sesión)

Validaciones

- El plan existe en planes.
- El usuario no tenga ya ese plan activo (opcional).

Salidas

- Éxito: confirmación “Plan contratado”.
- Error: “Plan no válido”, “Ya tienes este plan activo”.

Acceso a BD

- SELECT en planes para validar existencia.
- INSERT en planes_asignados (id_plan, id_usu).

Procesos internos

- Filtrado de ejercicios según nivel de usuario.
- Asignación masiva en ejercicios_asignados con estado completado = false.

Almacenamiento

- Registro en planes_asignados y ejercicios_asignados.

Errores controlados

- Plan inexistente.
- Duplicado de asignación.

Mensajes

- “Plan X contratado con éxito”, “Error al contratar plan”.

- **Caso de uso: GESTIONAR PLAN**

Funcionalidad: Visualizar historial de planes del usuario y su estado.

Actor – Usuario registrado

Entradas

- id_usuario (token)

Validaciones

- Usuario autenticado.

Salidas

- Lista de planes asignados: id, nombre plan, fecha de contratación, estado (en curso/completado).

Acceso a BD

- SELECT en planes_asignados JOIN planes.

Procesos internos

- Cálculo de porcentaje de ejercicios completados por plan.

Almacenamiento – Ninguno

Errores controlados

- Token inválido.

Mensajes

- “No tienes planes asignados” (si lista vacía).

- **Caso de uso: MODIFICAR PLAN**

Funcionalidad: Permitir al usuario actualizar los ejercicios de un plan asignado y marcar completados.

Actor – Usuario registrado

Entradas

- id_plan_asignado, id_ejer_asignado, acción (reemplazar/ marcar completado)

Validaciones

- Pertenencia del plan al usuario.
- Existencia del ejercicio asignado.

Salidas

- Éxito: “Ejercicio marcado como completado” o “Ejercicio reemplazado”.
- Error: “Acceso denegado” o “Ejercicio no encontrado”.

Acceso a BD

- SELECT en ejercicios_asignados.
- UPDATE en ejercicios_asignados (campo completado o id_ejer).

Procesos internos

- Si se reemplaza, selección de nuevo ejercicio viable según intensidad.

Almacenamiento

- Actualización de ejercicios_asignados.completado o id_ejer.

Errores controlados

- Intento de modificar plan ajeno.
- Ejercicio no válido.

Mensajes

- “Modificado con éxito”, “No puedes modificar este plan”.

- Caso de uso: MODIFICAR PERFIL

Funcionalidad: Actualizar datos personales del usuario.

Actor – Usuario registrado

Entradas

- Campos a modificar: nombre, apellido, peso, altura, nivel, etc.

Validaciones

- Formatos correctos (nombres no vacíos, nivel válido).

Salidas

- Éxito: “Perfil actualizado”.
- Error: “Formato inválido en campo X”.

Acceso a BD

- UPDATE en usuarios.

Procesos internos – Ninguno

Almacenamiento

- Nuevos valores en usuarios.

Errores controlados

- Validaciones de formato.

Mensajes

- “Perfil modificado con éxito”.

ADMINISTRADOR

- **Caso de uso: GESTIONAR USUARIOS**

Funcionalidad: Administrar altas, bajas y modificaciones de usuarios.

Actor – Administrador

Entradas

- Para alta: mismos datos que en Registro.
- Para modificación: id_usuario + campos a cambiar.
- Para baja: id_usuario.

Validaciones

- Email único.
- Existencia del usuario.

Salidas

- Éxito: “Usuario X creado/ modificado/ eliminado”.
- Error: “Usuario no encontrado”, “Email duplicado”.

Acceso a BD

- SELECT, INSERT, UPDATE, DELETE en usuarios.

Procesos internos

- Mismo flujo de hash de contraseña si cambia.

Almacenamiento

- Según operación (nueva fila, cambios o eliminación física/lógica).

Errores controlados

- Duplicidad, inexistencia.

Mensajes

- “Operación completada”.

- **Caso de uso: GESTIONAR PLANES**

Funcionalidad: Administrar el catálogo de planes de entrenamiento.

Actor – Administrador

Entradas

- Datos de plan: nombre, descripción, duración, precio, imagen.
- id_plan para editar o eliminar.

Validaciones

- Duración > 0, precio ≥ 0.

Salidas

- Listado actualizado de planes.

Acceso a BD

- SELECT, INSERT, UPDATE, DELETE en planes.

Procesos internos – Ninguno

Almacenamiento

- Según operación.

Errores controlados

- Validación de campos.

Mensajes

- “Plan creado/ modificado/ eliminado”.

- **Caso de uso: GESTIONAR EJERCICIOS**

Funcionalidad: Administrar el catálogo de ejercicios disponibles.

Actor – Administrador

Entradas

- Datos de ejercicio: imagen, nombre, descripción, series, repeticiones, tipo, músculo, intensidad.
- id_ejercicio para editar o eliminar.

Validaciones

- Series > 0, repeticiones en formato correcto, intensidad ∈ {leve, media, alta}.

Salidas

- Listado actualizado de ejercicios.

Acceso a BD

- SELECT, INSERT, UPDATE, DELETE en ejercicios.

Procesos internos – Ninguno

Almacenamiento

- Según operación.

Errores controlados

- Validaciones de campo.

Mensajes

- “Ejercicio creado/ modificado/ eliminado”.

MAPA DE NAVEGACIÓN

