

# ■ Entrega I – Proyecto DAM

**Proyecto:** FitLife – App de Hábitos y Rutinas Personales

**Fecha de entrega:** 24/10

**Repositorio GitHub:** (cuando lo cree pondré la URL)

## 1. Descripción y justificación del proyecto

El proyecto FitLife será una aplicación para ayudar a los usuarios a llevar un control de sus hábitos y rutinas personales (ejercicio, estudio, lectura, descanso, agua, sueño, etc.). El usuario podrá crear sus propias rutinas y hábitos, y la app le enviará recordatorios para realizar las actividades. Además, la app mostrará estadísticas de cumplimiento para motivar al usuario a seguir sus hábitos.

## Alternativas en el mercado

- MyFitnessPal: muy buena en nutrición, pero no permite crear rutinas personalizadas con recordatorios.
- Google Fit: registra actividad, pero no permite hábitos de estudio o lectura ni recordatorios personalizados.
- Habitica: gamifica hábitos, pero no tiene estadísticas visuales completas ni flexibilidad de rutinas.

FitLife se diferencia porque permite crear hábitos y rutinas personales, enviar recordatorios y mostrar estadísticas de progreso de manera sencilla.

## 2. Alcance

El proyecto tendrá estas partes:

- Autenticación de usuarios con roles (usuario normal y administrador para configuración básica).
- Creación de rutinas personalizadas por el propio usuario.
- Recordatorios automáticos según la planificación.
- Registro de hábitos diarios (agua, sueño, estudio, ejercicio, lectura, etc.).
- Reportes y estadísticas sobre el cumplimiento de los hábitos y rutinas.

## 3. Objetivos

- Crear una aplicación útil para gestionar hábitos y rutinas personales.
- Que el usuario reciba recordatorios automáticos para seguir sus hábitos.
- Mostrar estadísticas de progreso y cumplimiento.
- Usar PostgreSQL para guardar toda la información.
- Implementar roles de usuario y administrador básicos para gestión de la app.

## 4. Requisitos del sistema

**Requisitos funcionales:**

- RF1: Registro e inicio de sesión de usuarios.
- RF2: Creación y gestión de rutinas y hábitos.
- RF3: Envío de recordatorios automáticos según la planificación.

- RF4: Marcado de cumplimiento de hábitos por el usuario.
- RF5: Generación de estadísticas y reportes.

#### **Requisitos no funcionales:**

- RNF1: Seguridad en la autenticación.
- RNF2: Guardar los datos en PostgreSQL.
- RNF3: Escalabilidad para varios usuarios.
- RNF4: Código en un repositorio GitHub público.

#### **Requisitos de interfaz:**

- IU1: Pantalla de login/registro.
- IU2: Panel de usuario con resumen de hábitos y rutinas.
- IU3: Panel de configuración de rutinas y recordatorios.
- IU4: Pantalla de estadísticas y reportes.

## **5. Casos de uso importantes**

- CU1: Crear rutina/hábito → el usuario define actividades y horarios.
- CU2: Recordatorio → la app envía alertas al usuario según la planificación.
- CU3: Marcar actividad realizada → el usuario indica si completó la actividad.
- CU4: Generar reporte → el usuario solicita un reporte de progreso en PDF/Excel.

## **6. Modelo de base de datos (simplificado)**

Usuarios (id\_usuario, nombre, email, rol)

Rutinas (id\_rutina, id\_usuario, nombre, descripcion, horario)

Habitos (id\_habito, id\_usuario, tipo, valor, fecha, completado)

Recordatorios (id\_recordatorio, id\_usuario, id\_rutina, fecha\_hora, mensaje)

Reportes (id\_reporte, id\_usuario, fecha, tipo, contenido)

## **7. Stack tecnológico**

- Lenguaje: Java (Spring Boot) o C# (ASP.NET Core).
- Base de datos: PostgreSQL.
- Frontend (opcional): Angular/React o .NET MAUI.
- Repositorio: GitHub público.