

# Explicacion de los Repositorios - Persistencia con Archivos Planos

## 1. Introducción

Este documento explica la implementación de tres repositorios que utilizan archivos de texto plano para la persistencia de datos. Cada repositorio corresponde a una entidad del sistema: Persona, Evento Deportivo y Reserva. La persistencia se realiza en archivos CSV y se mantiene un mecanismo simple para la generación de IDs únicos, garantizando que no se reutilicen incluso si se eliminan registros.

## 2. Conceptos Comunes a los Tres Repositorios

Los tres repositorios comparten características comunes:

- Manejo de IDs únicos mediante un archivo separado que guarda el último ID utilizado.
- Uso de directorios específicos para almacenar los archivos de datos (ej. 'Personas', 'Eventos', 'Reservas').
- Persistencia en archivos CSV donde cada línea representa un registro con campos separados por comas.
- Implementación básica de las operaciones CRUD (Crear, Leer, Actualizar, Eliminar).
- Para modificar o eliminar registros, el archivo completo se sobrescribe con la nueva lista de registros.

## 3. RepositorioPersona

El RepositorioPersona gestiona la entidad Persona con los siguientes atributos: Id, Nombre, Apellido, DNI y Email.

Métodos implementados:

- Agregar: Añade una nueva persona asignándole un ID único.
- Modificar: Actualiza los datos de una persona existente buscando por ID.
- Eliminar: Borra una persona del archivo según su ID.
- Listar: Devuelve todas las personas.
- ObtenerPorId, ObtenerPorDni, ObtenerPorEmail: Devuelven la persona que coincide con el criterio.
- TieneReservas y EsResponsable: Se dejan sin implementación real y retornan falso, dado que corresponden a otras lógicas de negocio.

La persistencia se realiza en 'Personas/personas.csv' y el último ID en 'Personas/ultimoid.txt'. Se asegura la

# Explicacion de los Repositorios - Persistencia con Archivos Planos

creación de los directorios y archivos si no existen.

## 4. RepositorioEventoDeportivo

El RepositorioEventoDeportivo maneja la entidad EventoDeportivo con atributos: Id, Nombre, Fecha y Lugar.

Métodos implementados:

- Agregar: Inserta un nuevo evento con ID único.
- Modificar: Actualiza un evento existente.
- Eliminar: Borra un evento por ID.
- ListarTodos: Devuelve todos los eventos.
- ObtenerPorId: Devuelve un evento específico por ID.
- Guardar: Inserta o actualiza un evento dependiendo si el ID existe.

La fecha se guarda en formato ISO (yyyy-MM-dd). Los archivos están en 'Eventos/eventos.csv' y el último ID en 'Eventos/ultimold.txt'. Se sobrescribe el archivo completo para actualizar datos.

## 5. RepositorioReserva

El RepositorioReserva gestiona la entidad Reserva con atributos: Id, Eventold, Personald y FechaReserva.

Métodos implementados:

- Agregar: Añade una nueva reserva con ID único.
- Modificar: Actualiza una reserva existente.
- Eliminar: Borra una reserva según su ID.
- ExisteReserva: Verifica si una persona tiene una reserva para un evento dado.
- ObtenerPorId: Devuelve una reserva específica.
- Listar: Devuelve todas las reservas.
- ListarPorEvento: Devuelve todas las reservas para un evento específico.

La persistencia es en 'Reservas/reservas.csv' y 'Reservas/ultimold.txt'. FechaReserva se almacena en formato ISO. Se sobrescribe el archivo para actualizar registros.

# Explicacion de los Repositorios - Persistencia con Archivos Planos

## 6. Conclusión

Esta implementación simple cumple con los contratos de las interfaces y ofrece persistencia básica en archivos planos. Si bien es funcional, para sistemas más grandes o con alta concurrencia es recomendable migrar a bases de datos relacionales o NoSQL para mejorar eficiencia, integridad y escalabilidad.

Además, los métodos no implementados en `RepositorioPersona` (`TieneReservas`, `EsResponsable`) deben conectarse con otros repositorios o servicios para respetar el principio de responsabilidad única.