* DAMA Sports * Proyecto

Hugo Garrido Rojo, Rebeca Cerón Camps, José Andrés Cava Torres



Índice

1.Modelo de clases	Diagrama simple de Socio
2.Servicio ClubDeportivo	Métodos principales y reglas
3.Reglas Negocio	Disponibilidad, bajas, tiempos válidos
4.Persistencia	Formato ficheros, lectura/escritura, lecciones aprendidas
5.Flujo de Git	Ramas, revisiones
6.Caso práctico	1 caso exitoso + 1 caso con validación.



Modelo de clases (UML)

Socio

- ID SOCIO: string

- dni : string

- nombre : string

- apellido : string

- telefono : string

- email: string

+ getIdSocio(): string

+ getDni(): string

+ getNombre(): string

+ getApellidos(): string

+ getTelefono(): string

+ getEmail(): string

+ setDni(in dni: string)

+ setNombre(in nombre: string)

+ setApellidos(in apellidos: string)

+ setTelefono(in telefono: string)

+ setEmail(in email: string)

Pista

- ID PISTA: string

- deporte : string

- descripcion : string

- disponible : string

+ getIdPista(); string

+ is Disponible(): boolean

+ setDisponible(in disponible: string)

+ getDescripcion(); string

+ setDescripcion(in descripcion: string)

+ getDeporte(): string

+ setDeporte(in deporte: string)

ClubDeportivo

+ pistas : string [*]

+ socios : string [*]

+ reservas : string [*]

+ agregarSocio(in socio: Socio)

+ agregarPista(in pista: Pista)

+ agregarReserva(in reserva: Reserva)

+ getSocios(): Socio [*]

+ getPistas(): Pista [*]

+ getReservas(); Reserva [*]

+ cargarDatosSocio()

+ quardarDatos Socio(in socio: Socio)

+ cargarDatosPista()

+ quardarDatosPista(in pista: Pista)

+ cargarDatosReserva()

+ quardarDatosReserva(in reserva: Reserva)

+ quardarDatos Globales ()

+ altaSocio(in socio: Socio)

+ bajaSocio(in idSocio: string)

+ altaPista(in pista: Pista)

+ cambiarDisponibilidadPista(in idPista; string, in disponible; boolean)

+ crearReserva(in reserva: Reserva): boolean

+ cancelarReserva(in idReserva: string)

Reserva

- ID RESERVA: string

- idSocio : string

- idPista : string

- fecha : date

- horalnicio : date

duracionMin : integer

- precio : double

+ getIdReserva(): string

+ getIdSocio(): string

+ setIdSocio(in idSocio: string)

+ getIdPista(): string

+ setIdPista(in idPista: string)

+ getFecha(); date

+ setFecha(in fecha: date)

+ getHoralnicio(); date

+ setHoralnicio(in horalnicio: string)

+ getDuracionMin(): integer

+ setDuracionMin(in duracionMin: integer)

+ getPrecio(): double

+ setPrecio(in precio: double)



Clases: Socio, Pista y Reserva

```
public class Socio { &rebecaccamps +1
  private final String idSocio;
  private String dni;
  private String nombre;
  private String apellidos;
  private String telefono;
  private String email;
```

- Identificador único
- Cada cliente del club
- No mismo socio con ID

```
public class Reserva { & rebecaccamps +1

private final String idReserva;
private String idSocio;
private String idPista;
private LocalDate fecha;
private LocalTime horaInicio;
private int duracionMin;
private double precio;
```

- Cada pista tiene un estado de disponibilidad
- Se puede marcar como no disponible

```
public class Pista { & rebecaccamps

private final String idPista;

private String deporte;

private String descripcion;

private boolean disponible;
```

- Conecta Socios y Pistas
- Valida que estén disponibles y no se solapen horas



Clase ClubDeportivo

- Núcleo del sistema
- Contiene las colecciones de Socios, Pistas y Reservas
- Contiene toda la lógica

```
• •
```

```
//Dar de alta a | un Socio
public boolean altaSocio(Socio socio) { &rebecaccamps
  for (Socio s : socios) {
      if (s.getIdSocio().equals(socio.getIdSocio())) {
          return false;
      }
  }
  socios.add(socio);
  return true;
}
```

```
Clase: CubDeportivo
```

ArrayList<Socio> socios

ArrayList<Pista> pistas

ArrayList<Reserva> reservas

Métodos principales

altaSocio()

bajaSocio()

cambiarDisponibilidadPista()

crearReserva()

cancelarReserva()

guardarDatos() (De todos los objetos)



Reglas de Negocio

- No se puede crear un Socio con un id ya existente
- Se deben rellenar los campos obligatorios *

- La fecha de reserva no puede ser anterior a la actual
- La pista tiene que estar disponible para poder reservar



Persistencia de datas

Contenido:

- Lectura y escritura al presionar el botón de guardado.
- Formato de ficheros: .txt

Archivos:

 socios.txt, pistas.txt, reservas.txt

Formato:

SOCIOS:

id, dni, nombre, apellidos, telefono, email

RESERVA:

id Reserva, id Socio, id Pista, fecha, hora Inicio, duracion Min, precio

PISTA:

idPista, deporte, descripcion, disponible

Método Cargar

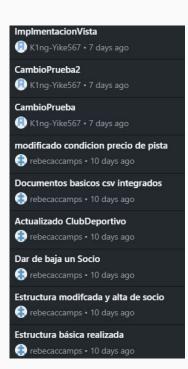
Método Guardar

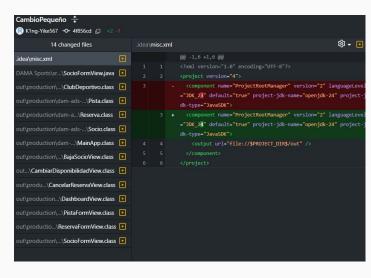


Flujo de Git

Título: Flujo de Trabajo con Git **Contenido:**

- Repositorio: dam-ads-u1-equipoN
- Ramas: Master, Rebeca, Hugo, José.
- Pull Requests y revisiones de código
- Commits con mensajes claros







Caso práctico

- Reservar con una fecha anterior a la de hoy (da error)
- Intentar Reservar una pista no disponible
- Intentar dar de alta Socio con mismo id
- Guardado de datos de un Socio con solo campo ID.

→ ¡Muchas gracias! →

¿ALGUNA PREGUNTA?