



# Diseño y Desarrollo de Sistemas de Información

---

Grado en Ingeniería Informática

---

## Práctica 4

---

**Llamadas a procedimientos y funciones almacenadas  
Transacciones**

---

Curso 2021/22

Duración estimada: 2 sesiones (4 horas)

---

## OBJETIVOS

---

- Aprender a gestionar distintos *JPanel* dentro de un *JFrame*
- Aprender a realizar llamadas a procedimientos y funciones almacenadas mediante JDBC
- Aprender a manejar transacciones

### 1. Gestión de la interfaz principal

Crear las clases **Socio** y **SocioDAO** y diseñar un panel para la gestión de socios con la misma funcionalidad que el de los monitores. Añadir un ítem al menú "Socios" para mostrarlo

### 2. Procedimientos y funciones almacenadas

2.1. Crear un procedimiento con 2 parámetros:

- Un parámetro de entrada para recibir un identificador de actividad
- Un parámetro de salida para devolver un cursor

El procedimiento recibe un identificador de actividad y devuelve, en el parámetro de salida, un cursor con el nombre y el correo electrónico de los socios que están inscritos en dicha actividad.

2.2. Añadir un nuevo elemento (Actividades) al menú principal con un ítem (Socios por Actividad)

2.3. Crear las clases **Actividad** y **ActividadDAO** para la gestión de actividades. *La clase ActividadDAO solo debe tener, de momento, un método que llame al procedimiento almacenado*

2.4. Crear una nueva ventana (*JFrame*) para mostrar los socios inscritos en una actividad. En este caso será una ventana independiente (no un panel) con los siguientes componentes:

- Un cuadro de edición para leer un identificador de actividad (*si tenéis tiempo, podéis sustituir el cuadro de edición por un ComboBox cargado con el nombre de todas las actividades de la base de datos*)
- Un *Jtable* para mostrar el resultado del procedimiento
- Un botón para lanzar el procedimiento y otro para cerrar la ventana

### 3. Manejo de transacciones

El esquema de la base de datos está diseñado para no permitir el borrado de monitores que sean responsables de alguna actividad ya que no tiene sentido hacer un borrado en "cascada", es decir, que la eliminación de un monitor responsable produzca la eliminación de las actividades de las que es responsable.

Para mantener la "regla de negocio" del modelo ER que indica que una actividad debe tener, obligatoriamente, un monitor responsable, se debe implementar la siguiente secuencia de operaciones para eliminar un monitor:

- Comprobar si el monitor es responsable de alguna actividad. Si no lo es, eliminarlo directamente de la base de datos. En caso afirmativo:
  - Insertar un monitor ficticio con código "M999", de nombre "Responsable Genérico" y con DNI "00000000A" (son los tres únicos campos obligatorios de la tabla MONITOR). Esto solo debe hacerse cuando no exista este monitor ficticio
  - Sustituir la clave ajena por el valor "M999" en todas las filas de la tabla ACTIVIDAD en las que el monitor es responsable de alguna de ellas
  - Eliminar el monitor de la tabla MONITOR

Modificar el código del evento "Eliminar Monitor" por una transacción que realice esta secuencia de operaciones en un único bloque.