

Diseño y Desarrollo de Sistemas de Información

Grado en Ingeniería Informática

Proyecto de Prácticas

[Curso 2022-23]



Objetivo

- Diseñar e implementar un Sistema de Información partiendo de un análisis de requisitos y un esquema de base de datos relacional

Tecnologías utilizadas

- Entorno de desarrollo de integrado – IDE (se recomienda *NetBeans*)
- Lenguaje de programación Java
- Librería gráfica *Swing*
- Sistema Gestor de Bases de Datos: Oracle y MariaDB
- Herramienta de Mapeo Objeto-Relacional (*Hibernate*)

Enunciado

El proyecto consistirá en el diseño y desarrollo (implementación) de un Sistema de Información cuyos requisitos se describen en el siguiente texto:

El gimnasio "Ponte en Forma" necesita actualizar su sistema de información. Para ello ha diseñado una base de datos para almacenar la información de sus actividades, monitores y socios. Cada actividad tiene un monitor responsable y un precio base mensual. Los socios pueden ser de distinta categoría (desde la 'A' hasta la 'E') y la cuota de sus actividades se calcula en función de dicha categoría:

- 'A': sin descuento
- 'B': 10% de descuento
- 'C': 20% de descuento
- 'D': 30% de descuento
- 'E': 40% de descuento

Ver diagrama Entidad-Relación en: <https://aulasvirtuales.uhu.es/mod/resource/view.php?id=119872>



¿Qué hay que hacer?

El Sistema de Información será una aplicación cliente-servidor de escritorio implementada en lenguaje Java haciendo uso de la librería gráfica *Swing*. La aplicación debe tener la capacidad de conectarse a dos servidores (Oracle y MariaDB) y funcionar para ambos SGBD. La conexión y comunicación con las bases de datos se realizará mediante la herramienta ORM *Hibernate*.

El proyecto tendrá el siguiente diseño y funcionalidad:

- Una pantalla inicial para seleccionar el SGBD con el que se va a conectar
- Una pantalla principal en la que se situarán los distintos menús de opciones
- Un panel para la gestión de los monitores. En esta interfaz se deberán realizar las operaciones de consulta, inserción, actualización y borrado de monitores, tal como se describe en el enunciado de la práctica 3
- Un panel para la gestión de los socios. En esta interfaz se deberán realizar las operaciones de consulta, inserción, actualización y borrado de socios, tal como se describe en el enunciado de la práctica 3
- Una ventana que muestre el nombre y el correo electrónico de los socios inscritos en una determinada actividad, tal como se describe en el enunciado de la práctica 4
- Una ventana en la que se gestionen las inscripciones de los socios en las actividades, tal como se describe en el enunciado de la práctica 5
- **¡¡ NUEVO !!** Un elemento de búsqueda o filtrado de socios en el panel de gestión de socios. Su diseño y funcionalidad será de libre elección y se podrá utilizar cualquiera de los campos del socio (ejemplos: búsqueda por DNI, nombre, categoría, etc.). El objetivo es mostrar en el *JTable* la información de los socios que cumplen con el criterio de búsqueda

NOTAS MUY IMPORTANTES

- El proyecto final consiste en la realización de la **aplicación completa** que se ha ido desarrollando durante las prácticas propuestas, una vez "migrada" a *Hibernate* y añadiendo la opción de búsqueda de socios
- La aplicación debe seguir, obligatoriamente, el patrón MVC (Modelo-Vista-Controlador). **Las clases con las que se diseñan las vistas solo pueden contener el código necesario para su diseño y visualización**
- El estudiante debe asegurarse de que su proyecto se puede desplegar en un IDE *NetBeans* sin necesidad de realizar ninguna operación (inclusión de librerías, etc.), que compila sin errores, que se conecta correctamente con la base de datos, etc. **Si el proyecto no se puede ejecutar, no será evaluado**
- Una vez entregado el proyecto, el estudiante debe asegurarse que, en sus cuentas de Oracle y MariaDB, están los esquemas y los datos originales de la base de datos
- Las operaciones deben cumplir los requisitos del enunciado y controlar **todas las posibles excepciones**, tanto a nivel de base de datos como a nivel de implementación (en el propio programa)
- Para una mejor lectura y escritura de las fechas, se recomienda utilizar el complemento calendario *jCalendar* o alguno similar