



**Universidad**  
**Europea** MADRID

# PROYECTO INTEGRADOR

## 1DAW

Dos Hombres y Medio



Antonio Alonso Alonso

Juan José González Lanza

Miguel De Pablo García

## Índice

Resumen.....	2
1. Introducción .....	3
2. Objetivos .....	3
3. Tecnologías utilizadas.....	5
4. Desarrollo e implementación.....	6
5. Metodología .....	12
6. Resultados y conclusiones.....	14
7. Trabajos futuros .....	14
Anexos .....	14
Anexo I – Listado de requisitos de la aplicación .....	14
Anexo II – Guía de uso de la aplicación .....	14
Anexo III.....	14

## Resumen

(Al finalizar el proyecto)

## 1. Introducción

## 2. Objetivos

En cuanto a la organización y el desarrollo del trabajo grupal los objetivos son los siguientes:

- Gestionar las cargas de trabajo adaptándose a la planificación del tiempo.
- Respetar la autonomía de los compañeros y realizar consultas a la persona adecuada al encontrar un conflicto o dificultad.
- Buscar habilidades para el liderazgo y la sinergia del grupo de trabajo
- Gestionar eficazmente los conflictos que puedan llegar a producirse estableciendo un ambiente de trabajo agradable, respetuoso y tolerante.
- Tener iniciativa individual para resolver problemas y tomar decisiones.

En cuanto a las capacidades relacionadas con el módulo de Bases de Datos los objetivos son:

- Crear bases de datos definiendo su estructura y las características de sus elementos según el modelo relacional.
- Diseñar modelos normalizados interpretando diagramas entidad/relación.
- Realizar el diseño físico de bases de datos utilizando asistentes, herramientas gráficas y el lenguaje de definición de datos.
- Consultar y modificar la información almacenada utilizando asistentes, herramientas gráficas y el lenguaje de manipulación de datos.

En cuanto a las capacidades relacionadas con el módulo de Bases de Datos los objetivos son:

- Realizar el diseño de clases necesarias para seguir el patrón MVC.
- Implementar cada una de las clases para lograr su objetivo: clases de interfaz gráfica pertenecientes a la vista, clases que representen el modelo y clases encargadas de la lógica de la aplicación pertenecientes al control.
- Desarrollar una aplicación que gestione información almacenada en bases de datos relacionales identificando y utilizando mecanismos de conexión.

En cuanto a las capacidades relacionadas con el módulo de Bases de Datos los objetivos son:

- Realizar el análisis y el diseño de cualquier aplicación empleando técnicas UML.
- Documentar aplicaciones.
- Gestionar las diferentes versiones de un software y el trabajo colaborativo.
- Realizar pruebas de testeo sobre los programas.
- Seguir metodologías ágiles para el desarrollo y planificación de programas.

### 3. Tecnologías utilizadas

#### Hardware:

-Ordenador con componentes de gama media-baja actuales.

#### Software:

-Eclipse IDE: es el entorno de desarrollo en el que se emplea la aplicación.

-Java Development Kit (JDK): necesario para la ejecución de la aplicación en Java.

-Sistema gestor de bases de datos: MySQL para crear la base de datos y almacenar y gestionar los datos de la aplicación.

-Trello y GitHub: para desarrollar la planificación del proyecto y el control de versiones se han empleado estas herramientas organizativas.

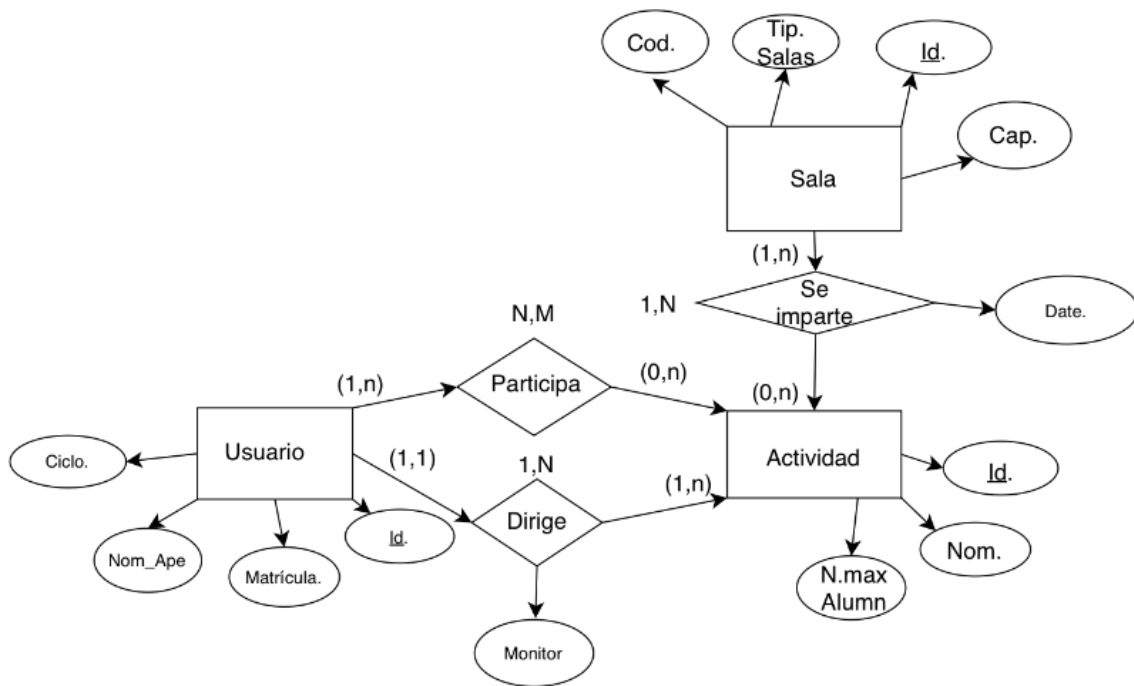
-Software de modelado de diagramas: se ha empleado Draw.io para la creación de los diagramas de flujo, entidad-relación y diagrama de casos de uso.

-Canva: para la creación del logo del proyecto se ha empleado Canva, una plataforma de diseño gráfico que proporciona herramientas para la realización de diseños de diversa índole de forma intuitiva.

## 4. Desarrollo e implementación

### Análisis de las especificaciones del proyecto. Diagrama E/R

Una vez analizadas y entendidas las especificaciones del proyecto se ha procedido a la realización del diagrama entidad-relación. Este representa visualmente los requisitos de la base de datos que tendrá la aplicación:



Se ha considerado que cada actividad es única y solo se realiza una vez, es decir, si por ejemplo realiza una actividad de pilates el 14 de abril a las 12:00 esa actividad no es la misma que el pilates del 15 abril a las 12:00 en lo que a la base de datos se refiere. Aunque sea la misma actividad y se realice en la misma sala, son actividades diferentes.

Se ha considerado que cada actividad es dirigida por exactamente un usuario (el monitor).

Se ha considerado también que un usuario puede participar en muchas actividades y que en una actividad pueden participar varios usuarios (hasta llegar al límite disponible en dicha actividad).

Por último, se ha considerado que una actividad se imparte en una sala y que en una sala se pueden impartir varias actividades.

### Modelo relacional

Se ha realizado el siguiente modelo relacional:

USUARIO (ID, Matricula, Nombre y Apellidos, Ciclo)

PK (ID)

ACTIVIDAD (ID, Nombre, N.Max alumnos, monitor\*, sala\*, fecha, hora).

PK (ID)

FK- monitor -> USUARIO (ID)

FK – sala -> SALA (ID)

SALA (Código, Tipo Salas, ID, Capacidad)

PK (ID)

Este modelo se encuentra en primera forma normal debido a que solo hay un tipo de dato por columna.

A su vez se encuentra en segunda forma normal debido a que no se encuentran dependencias parciales, es decir, no existe ningún atributo que dependa solo de una parte de una clave primaria compuesta.

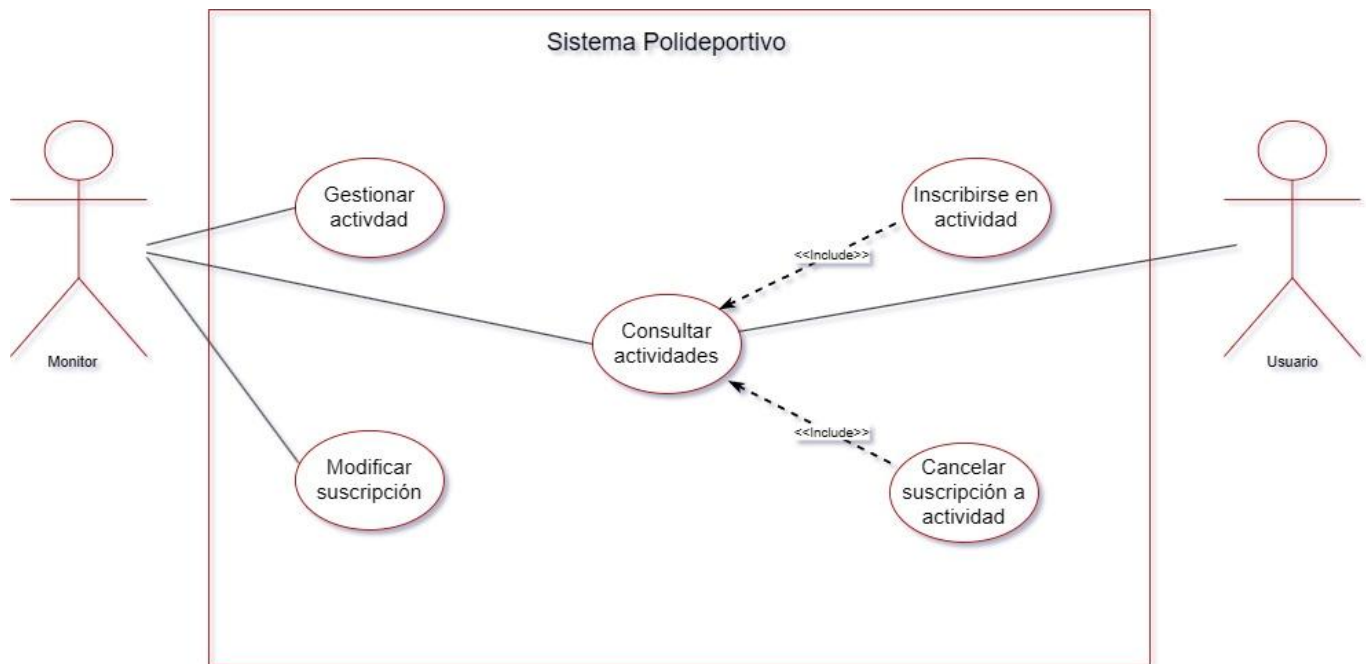
Al no encontrar dependencias transitivas, es decir, ningún atributo no clave depende de otro atributo no clave que dependa a su vez de una clave primaria, el modelo se encuentra también en tercera forma normal.

Se ha elaborado y rellenado una base de datos en MySQL que ha sido añadida al GitHub del proyecto.



## Diagrama de caso de uso

Se ha realizado el siguiente diagrama de caso de uso:



Dentro del caso “gestionar actividad” se encuentran englobadas las acciones de crear nueva actividad, editar actividad y borrar actividad por parte del monitor.

El caso “modificar suscripción” hace referencia a la capacidad del monitor para modificar qué usuarios están dados de alta en la actividad.

Ambos actores pueden consultar las actividades disponibles, en el caso del usuario para inscribirse en la actividad o cancelar la suscripción a la actividad tiene primero que realizar dicha consulta, de ahí que la relación establecida sea de tipo include.

## Logo

Para el logo de la aplicación se ha optado por el siguiente diseño:



Se trata de un diseño sencillo y minimalista adaptado para una correcta implementación en entornos digitales.

Se ha querido representar el movimiento y la actividad que se realiza en el ejercicio deportivo con la silueta de una persona corriendo. Esta misma premisa se ha aplicado al nombre de la aplicación, EuroSports Club, al dotarla de esa curvatura y dinamismo presente en el mundo del deporte.

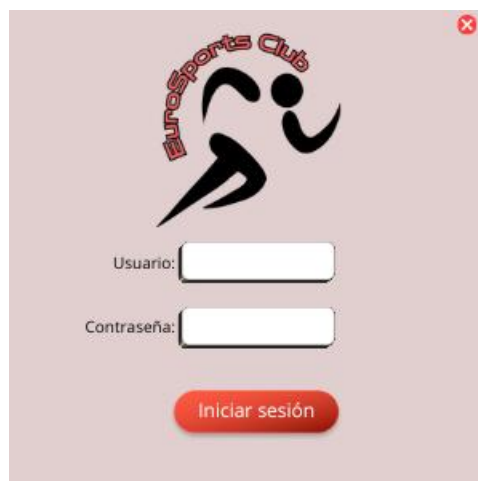
En cuanto a los colores se han elegido los que se han considerado más relacionados con la universidad como son el rojo, el blanco y el negro.

### **Diseño de Wireframes**

Como paso previo a la creación de las clases del proyecto pertenecientes a la vista, se han realizado en Canva una serie de diseños de wireframes con el diseño que se pretende que muestren dichas vistas.

Se ha optado por definir 2 vistas diferentes para cada ventana dependiendo del rol del usuario, es decir, dependiendo de si accede a la aplicación como usuario normal o como monitor.

Pantalla de Log in:



Tras iniciar sesión se plantea al usuario monitor la forma en la que desea acceder, si como usuario normal o como monitor:



La vista del monitor respecto al menú inicial y las actividades disponibles son las siguiente:



Las vistas del usuario respecto al menú inicial y a las actividades disponibles es la siguiente:



Se han realizado más wireframes que se encuentran disponibles en el documento correspondiente subido a GitHub.

## 5. Metodología

Durante la realización del proyecto se ha utilizado la metodología ágil SCRUM, una metodología de desarrollo ágil de software para gestionar el desarrollo de proyectos y productos.

Está basada en equipos autoorganizados y multifuncionales que trabajan en ciclos de tiempo cortos llamados sprints. Cada uno de estos tiene como objetivo entregar un incremento de producto funcional y potencialmente entregable.

En el siguiente diagrama de Gantt se muestra de una forma visual la organización del proyecto en dichos sprints con 3 tareas por cada uno de ellos que se van desarrollando en el tiempo.

Cada color representa un sprint, yendo desde el 1 hasta el 4.



Además de la metodología SCRUM, también se ha aplicado la metodología Kaban a través de la plataforma Trello. Esta consiste en un tablero dividido en columnas donde se representa el flujo de trabajo. Las columnas se dividen en tareas por hacer, tareas en proceso y tareas finalizadas, en ellas se van posicionando las tareas según avanza el trabajo en el proyecto.

Para la planificación del proyecto se empleará un tablero Trello destinado a la coordinación y distribución de tareas.

<https://trello.com/b/oVlq77T4/proyecto-integrador>

A su vez, se utilizará el repositorio de GitHub cuyo enlace aparece a continuación para realizar el control de versiones del proyecto.

<https://github.com/miguelon03/ProyectoIntegrador>

Para la coordinación del trabajo del grupo se empleará, en un principio, un servidor de Discord con distintos canales temáticos sobre el proyecto.

## **6. Resultados y conclusiones**

## **7. Trabajos futuros**

### **Anexos**

#### **Anexo I – Listado de requisitos de la aplicación**

Para garantizar el correcto funcionamiento de la aplicación se ha de disponer de los siguientes requisitos:

En cuanto a el diseño de los distintos diagramas se han de tener conocimientos necesarios sobre diagramas de uso y como crearlos, así como de creación de diagramas entidad relación y modelos relacionales. Son recomendable unas mínimas habilidades de diseño gráfico para la realización de los wireframes de las vistas.

En cuanto a la creación y llenado de bases de datos se han de tener conocimientos de programación en lenguaje SQL para poder desarrollar este trabajo correctamente.

En cuanto a la programación de las interfaces en el entorno de desarrollo Eclipse IDE se necesitan conocimientos de programación en Java que se han ido impartiendo durante el módulo.

#### **Anexo II – Guía de uso de la aplicación**

#### **Anexo III**