



**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 .....	4
3. Tecnologías utilizadas.....	6
4. Desarrollo e implementación.....	7
5. Metodología .....	18
6. Resultados y conclusiones.....	20
7. Trabajos futuros .....	20
Anexos .....	21
Anexo I – Listado de requisitos de la aplicación .....	21
Anexo II – Guía de uso de la aplicación .....	21
Anexo III – Historia del proyecto.....	29

## Resumen

Este proyecto ha consistido en el desarrollo de una aplicación de escritorio en Java que permite la gestión de actividades deportivas dentro de un centro. A través de una interfaz gráfica intuitiva, los usuarios pueden registrarse en actividades, consultar información personal y modificar su contraseña, mientras que los monitores tienen acceso a funcionalidades avanzadas como la creación, edición y eliminación de actividades. El sistema distingue entre usuarios normales y monitores, siendo estos últimos quienes gestionan las actividades que luego son ofrecidas a los usuarios. La conexión con una base de datos MySQL permite almacenar de forma estructurada todos los datos relacionados con usuarios, actividades, salas e inscripciones, garantizando la persistencia de la información. La base de datos ha sido diseñada para reflejar relaciones claras entre las entidades, permitiendo consultas complejas y operaciones eficientes. Se han implementado mecanismos de validación para evitar duplicidades y asegurar la integridad de los datos, como en el caso del control del aforo máximo de las actividades. Además, el sistema incorpora una gestión de accesos basada en credenciales, mostrando distintos menús según el tipo de usuario que acceda. A lo largo del desarrollo se han aplicado principios de la programación orientada a objetos, dividiendo el código en paquetes por funcionalidad (modelo, vista, controlador, acceso a datos), lo que facilita su mantenimiento y escalabilidad. En conjunto, el proyecto logra cumplir los objetivos funcionales establecidos, ofreciendo una solución completa y estructurada para la gestión de actividades deportivas en un entorno educativo o institucional.

## 1. Introducción

El presente proyecto tiene como objetivo el desarrollo de una aplicación destinada a gestionar de forma integral las actividades deportivas del Polideportivo de la Universidad Europea de Madrid. Esta herramienta ha sido solicitada con el fin de optimizar la organización de los espacios, la planificación de actividades y el control de usuarios, especialmente en lo que respecta al alumnado del Ciclo Formativo de Actividad Física y Deportiva (TAFD), quienes actúan como monitores de las diferentes actividades.

El contexto de esta petición que ha realizado El Polideportivo se enmarca en el entorno académico de la Universidad, donde los estudiantes del ciclo TAFD deben impartir actividades prácticas como parte de su formación. Para ello, se requiere una plataforma que permita gestionar qué actividades se ofrecen, quiénes participan, qué salas están disponibles y en qué horarios se desarrollan las sesiones.

La aplicación ha sido realizada en el lenguaje de programación Java empleando elementos del paquete Java Swing, siguiendo el patrón modelo-vista-controlador. La base de datos que contiene los datos necesarios para la ejecución del sistema ha sido creada y administrada en MySQL.

Ofrece funcionalidades básicas de gestión, permitiendo dar de alta, modificar, eliminar y consultar tanto usuarios como actividades y salas. El sistema maneja información esencial como datos personales de los alumnos (usuarios del polideportivo), características de las salas (tipo, capacidad, código), y detalles completos de las actividades programadas (nombre, monitor, horario, sala asignada, etc.).

La aplicación gestiona tres tipos principales de información: usuarios, salas y actividades. En cuanto a los usuarios, podrán ser registrados todos los alumnos de la Universidad Europea de Madrid que participen en las actividades del polideportivo. De cada usuario se almacena un identificador único, su número de matrícula, nombre y apellidos, así como el ciclo formativo en el que están matriculados. Los ciclos disponibles se limitan a: DAW (Desarrollo de Aplicaciones Web), DAM (Desarrollo de Aplicaciones Multiplataforma), ASIR (Administración de Sistemas Informáticos en Red), A3D (Animaciones 3D), TAFD (Actividad Física y Deportiva), EI (Educación Infantil) y CI (Comercio Internacional). En particular, los estudiantes del ciclo TAFD desempeñan el rol de monitores, por lo que solo ellos podrán ser asignados como responsables de las actividades.

De esta manera, se busca no solo facilitar la administración diaria del polideportivo, sino también brindar a los estudiantes del ciclo TAFD una herramienta práctica que apoye su formación y desarrollo profesional dentro de un entorno real de trabajo.

## 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 Programación 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 Entornos de Desarrollo 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

#### Desarrollo:

-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.

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

#### Planificación:

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

-Discord y WhatsApp: para la comunicación y distribución de documentos durante el desarrollo del proyecto se han empleado ambas aplicaciones.

#### Diseño:

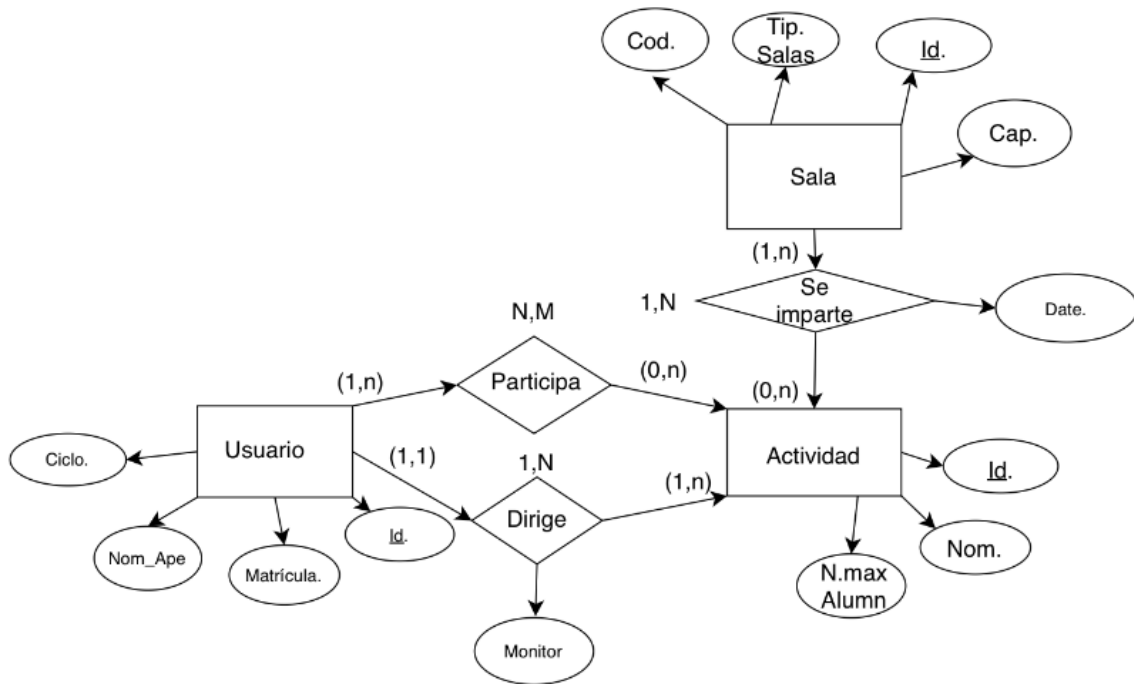
-Draw.io: se ha empleado Draw.io para la creación de los diagramas de flujo, entidad-relación y diagrama de casos de uso.

-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 se realiza una actividad de pilates el 14 de abril a las 12:00, esa actividad no es la misma que la actividad de 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)

INSCRIPCION (ID\_Usuario\*, ID\_Actividad\*, Fecha\_inscripción)

PK (ID\_Usuario, ID\_Actividad)

FK – ID\_Usuario -> USUARIO (ID)

FK – ID\_Actividad -> ACTIVIDAD (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.

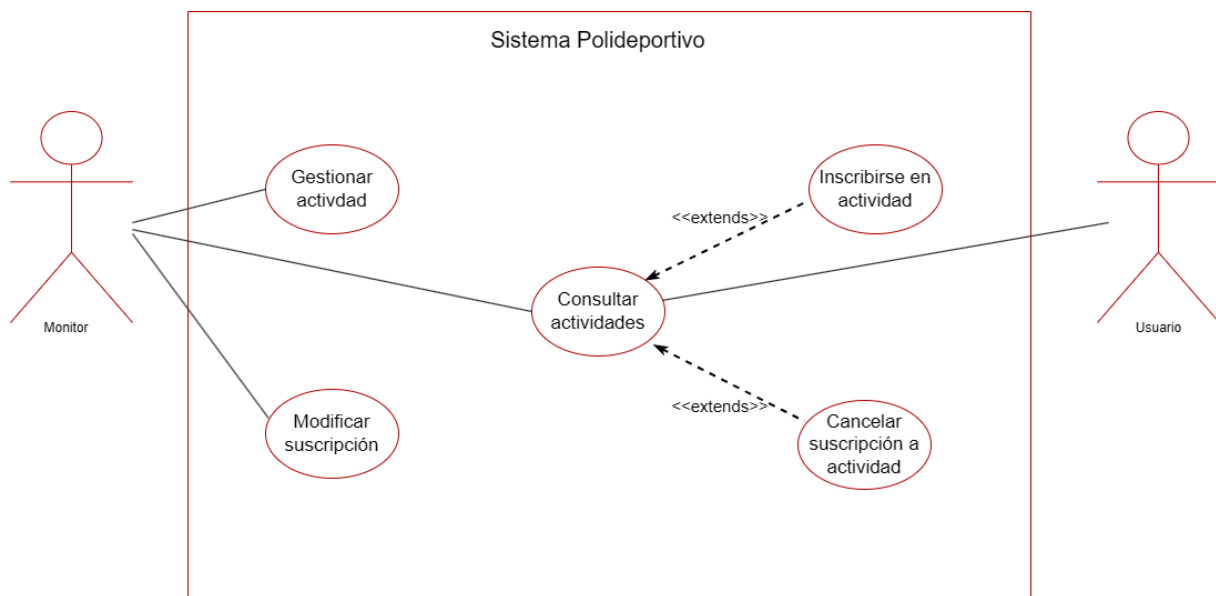
A su vez se encuentra en segunda forma normal debido a que no se encuentran dependencias parciales, es decir, no existe ningún atributo que dependa solo de una parte de una clave primaria compuesta.

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 poder inscribirse en la actividad o cancelar la suscripción a la actividad tendría primero que realizar dicha consulta, de ahí que la relación establecida sea de tipo EXTENDS.

## Logo

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



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 realizada en el ejercicio deportivo con la silueta de una persona corriendo. Esta misma premisa se ha aplicado al nombre de la aplicación, EuroSportsClub, al dotarla de esa curvatura y el dinamismo presente en el mundo del deporte.

En cuanto a los colores, se ha elegido el rojo como color del rótulo ya que se trata del color más relacionado con la universidad.

Para la figura que simula un corredor se ha optado por el negro para aportar contraste a la hora de introducirlo en la aplicación.

## 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:



Logo of EuroSports Club (a stylized figure running) and the text "EuroSports Club" in a red, curved font.

Usuario:

Contraseña:

**Iniciar sesión**

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



Logo of EuroSports Club (a stylized figure running) and the text "EuroSports Club" in a red, curved font.

¿Entras como usuario o como monitor?

**Usuario** **Monitor**

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



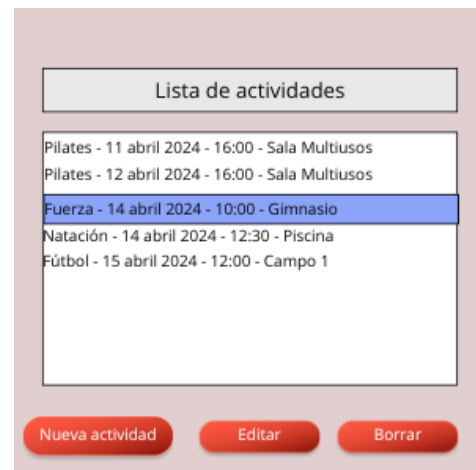
Actividades Salas Datos personales

Ver todas las actividades

Nueva actividad

Mis actividades

Logo of EuroSports Club (a stylized figure running) and the text "EuroSports Club" in a red, curved font.



Lista de actividades

Pilates - 11 abril 2024 - 16:00 - Sala Multiusos
Pilates - 12 abril 2024 - 16:00 - Sala Multiusos
<b>Fuerza - 14 abril 2024 - 10:00 - Gimnasio</b>
Natación - 14 abril 2024 - 12:30 - Piscina
Fútbol - 15 abril 2024 - 12:00 - Campo 1

**Nueva actividad** **Editar** **Borrar**

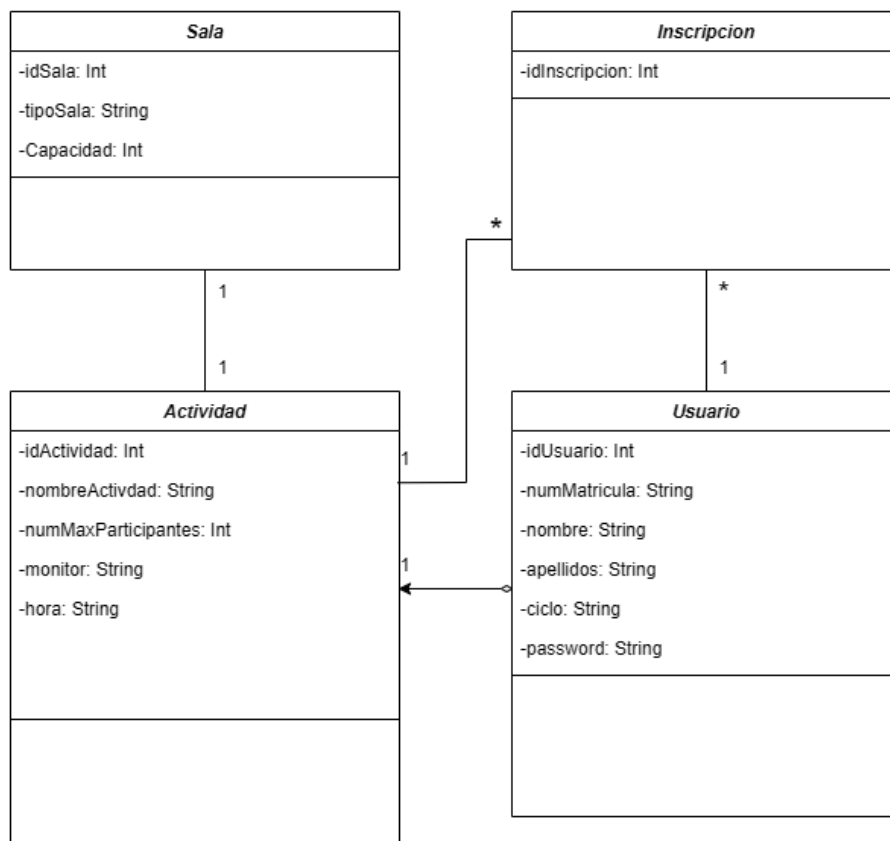
Las vistas del usuario respecto al menú inicial y a las actividades disponibles son las siguientes:



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

### Diagrama de clases

Se ha realizado mediante Draw.io el diagrama de clases de las clases correspondientes al paquete modelo.



No se han añadido métodos, ya que se han obviado los getters y setters y el resto de funcionalidades de las clases que se han realizado mediante distintos eventos y controladores del programa.

### **Estructura de la aplicación**

Se ha seguido el patrón modelo-vista-controlador para el desarrollo de la aplicación. En Eclipse IDE se han creado los diferentes paquetes pertenecientes a este patrón, además de un paquete para almacenar la clase de acceso a la base de datos y un último que almacena el main, es decir, el programa desde donde se ejecuta la aplicación.

En el paquete correspondiente al modelo se han creado las clases necesarias para construir los objetos presentes en la aplicación; actividad, usuario, sala e inscripción.

El paquete vista almacena las clases encargadas de la interfaz gráfica del programa, es decir, las distintas ventanas en las que van apareciendo los paneles requeridos por el usuario.

Los controladores encargados de lanzar eventos se han almacenado en el paquete control. Estas clases son necesarias para que la aplicación sea funcional y dinámica.

### **Funcionamiento de la aplicación**

La funcionalidad de la aplicación está basada en paneles (elementos JPanel), estos se intercambian en una ventana principal (elementos JFrame) mediante un menú y distintos botones.

La ventana principal que contiene el menú varía según el inicio de sesión haya sido realizado por un usuario o por un monitor.

En la ventana principal del monitor pueden aparecer los siguientes paneles: un panel con la lista de todas las actividades (informativo), un panel para crear una nueva actividad, un panel con una lista de actividades propias del monitor (las cuales puede editar o borrar).

En la ventana principal del usuario pueden aparecer los siguientes paneles: un panel con una lista de todas las actividades en las que puede inscribirse y un panel con una lista de las actividades en las que se ha inscrito, pudiendo borrar la inscripción de cada una de estas.

Ambas ventanas tienen dos paneles que son comunes: uno con que muestra una lista con la información de las salas disponibles y otro panel que muestra los datos personales del usuario y le permite cambiar su contraseña.

En ambas ventanas se accede a los paneles mediante un menú desplegable con las distintas opciones previamente descritas en este apartado.

Este acceso es realizado mediante los distintos controladores asociados a las opciones del menú.

La creación de nuevas actividades, el borrado y la edición de estas, la inscripción a las actividades, la eliminación de dicha inscripción y el cambio de contraseña es realizado mediante distintos controladores asociados a los botones de los paneles.

### **Acceso a la base de datos**

El acceso a la base de datos se ha realizado mediante la clase AccesoBBDDLogin, perteneciente al paquete bbdd (bases de datos). En ella se han programado los distintos métodos necesarios con consultas SQL para acceder a la información almacenada y ponerlos a disposición de las vistas, controladores y métodos presentes en la aplicación.

### **Pruebas realizadas en la aplicación**

Para comprobar el correcto funcionamiento de la aplicación, se han realizado las siguientes pruebas manuales:

#### ***Log in:***

Se ha intentado iniciar sesión con el ID Usuario y la contraseña correctos. El sistema lo ha permitido y ha mostrado una alerta de “inicio de sesión correcto”. Posteriormente, cuando el usuario ha sido un alumno de TAFD, se ha mostrado la vista de elección para que este acceda como monitor o usuario. Cuando el usuario ha sido un alumno de cualquier otro ciclo, se ha mostrado el menú principal de usuario.

Se ha intentado iniciar sesión con una contraseña errónea a la asociada al usuario. La aplicación no ha iniciado sesión y ha lanzado una alerta indicando “Contraseña incorrecta. Inténtalo de nuevo”.

Se ha intentado iniciar sesión con los campos ID Usuario y contraseña vacíos, en un caso introduciendo datos en solo uno de los dos campos y en otro dejando vacíos los dos campos. La aplicación no ha iniciado sesión y ha lanzado una alerta indicando “Contraseña incorrecta. Inténtalo de nuevo”.

***Nueva actividad:***

Se ha intentado añadir una actividad con fecha anterior a la actual. La aplicación no lo ha permitido y ha lanzado una alerta indicando “La fecha no puede ser anterior a hoy”.

Se ha intentado crear un mismo tipo de actividad en la misma fecha y hora que una creada anteriormente. El sistema no lo ha permitido y ha lanzado una alerta indicando “Ya existe una actividad con ese nombre en esa fecha y hora”.

Se ha intentado crear una actividad sin seleccionar ninguna actividad de las disponibles en la lista. La aplicación no lo ha permitido y ha lanzado una alerta indicando “Debe seleccionar una actividad válida”.

Se ha intentado crear una actividad dirigida por el mismo monitor en la misma fecha y hora en la que dirige otra actividad. El sistema no lo ha permitido y ha lanzado una alerta indicando “Tienes otra actividad con esta fecha y hora”.

Se ha intentado añadir una actividad sin conflictos de fecha y hora y de solapamiento de actividades del monitor. La aplicación lo ha permitido y ha lanzado una alerta indicando “Actividad añadida correctamente”.

***Borrar actividad:***

Se ha intentado borrar una actividad sin seleccionar ninguna de la lista de actividades. El sistema no lo permite y muestra una alerta indicando “Selecciona una actividad para borrar”.

Se ha intentado borrar una actividad seleccionada de la lista. Al pulsar el botón “Borrar”, el sistema ha lanzado una alerta de confirmación con la pregunta “¿Estás seguro de que quieres borrar esta actividad?” y dos botones; uno con la opción “Sí” y otro con la opción “No”. Al pulsar “No” la actividad no se borra y al pulsar “Sí” la actividad se borra tanto de la lista como de la base de datos y el sistema lanza una alerta indicando “Actividad borrada correctamente”.

***Editar actividad:***

Se ha intentado editar una actividad sin seleccionar ninguna de la lista de actividades. El sistema no lo permite y muestra una alerta indicando “Selecciona una actividad para editar”.



Se ha pulsado el botón “Editar” habiendo seleccionado una actividad de la lista. En la ventana se ha cambiado al panel de editar actividad. Una vez en este panel se han intentado guardar los cambios sin editar ningún registro. La aplicación no lo ha permitido y ha lanzado una alerta indicando “La fecha no puede ser anterior a hoy”.

Posteriormente se ha intentado editar los datos de la actividad haciendo coincidir la fecha con la de otra actividad creada anteriormente y pulsado el botón “Guardar cambios”. El sistema no lo ha permitido y ha lanzado una alerta indicando el conflicto “Ya existe otra actividad del mismo tipo en ese horario”.

Posteriormente se ha intentado editar los datos de la actividad evitando posibles conflictos pulsado el botón “Guardar cambios”. El sistema lo ha permitido ha lanzado un mensaje indicando “Actividad actualizada correctamente”.

#### ***Inscribirse en actividad:***

Se ha pulsado el botón “Inscribirse” para inscribirse en una actividad sin seleccionar ninguna de la lista disponible. La aplicación no lo ha permitido y ha lanzado la alerta “Por favor, selecciona una actividad”.

Se ha intentado inscribir a un usuario en una actividad en la que ya estaba inscrito. El sistema no lo ha permitido y ha lanzado el mensaje de error “No se pudo realizar la inscripción. Es posible que ya estés inscrito o no haya plazas disponibles”.

Se ha intentado inscribir a un usuario en una actividad en la que ya se ha alcanzado el cupo de inscripciones. El sistema no lo ha permitido y ha lanzado el mensaje de error “No se pudo realizar la inscripción. Es posible que ya estés inscrito o no haya plazas disponibles”.

Se ha intentado inscribir a un usuario en una actividad evitando conflictos. El sistema lo ha permitido y ha mostrado el mensaje “Inscripción realizada correctamente”.

#### ***Borrar la inscripción a una actividad:***

Se ha intentado borrar una inscripción pulsando el botón “Borrar inscripción” sin seleccionar ninguna actividad en la que está inscrito el usuario. La aplicación no lo ha permitido ha lanzado el mensaje “Selecciona una actividad para borrar tu inscripción”.

Se ha intentado borrar la inscripción de una actividad en la que el usuario estaba inscrito. Al pulsar el botón “Borrar inscripción”, el sistema ha lanzado una alerta de confirmación con la pregunta “¿Estás seguro de que quieres borrar tu inscripción en: \*nombre de la actividad\*?” y dos botones; uno con la opción “Sí” y otro con la opción “No”. Al pulsar “No” la inscripción no se borra y al pulsar “Sí” la inscripción se borra de la base de datos correctamente y el sistema lanza una alerta indicando “La inscripción fue cancelada con éxito”.

### ***Cambiar contraseña:***

Se ha pulsado el botón “Cambiar contraseña” sin rellenar ninguno de los campos requeridos. El sistema no lo permite y muestra un mensaje que indica “La contraseña actual no es correcta”.

Se ha pulsado el botón “Cambiar contraseña” rellenando el campo de contraseña actual con una contraseña errónea. El sistema no lo permite y muestra un mensaje que indica “La contraseña actual no es correcta”.

Se ha pulsado el botón “Cambiar contraseña” rellenando el campo de la contraseña actual correctamente, pero dejando vacíos los campos de la nueva contraseña y de confirmar la nueva contraseña. El sistema no lo permite y muestra un mensaje que indica “La nueva contraseña no puede estar vacía”.

Se ha pulsado el botón “Cambiar contraseña” rellenando el campo de la contraseña actual correctamente y rellenando el campo de la nueva contraseña, pero dejando vacío el campo para confirmar la nueva contraseña. El sistema no lo permite y muestra un mensaje que indica “Las nuevas contraseñas no coinciden”.

Se ha pulsado el botón “Cambiar contraseña” rellenando con la contraseña correcta el campo contraseña actual e introduciendo diferentes contraseñas en los campos de nueva contraseña y confirmar nueva contraseña. La aplicación no lo permite y muestra un mensaje que indica “Las nuevas contraseñas no coinciden”.

Finalmente se han rellenado todos los campos de manera correcta, es decir, el campo contraseña actual con la contraseña correspondiente al usuario y los campos de nueva contraseña con la misma contraseña. El sistema registra el cambio y muestra un mensaje que indica “Contraseña actualizada correctamente”.

## Pruebas unitarias

Se han realizado pruebas unitarias con JUnit en los métodos que contiene la clase AccesoBBDDLogin.

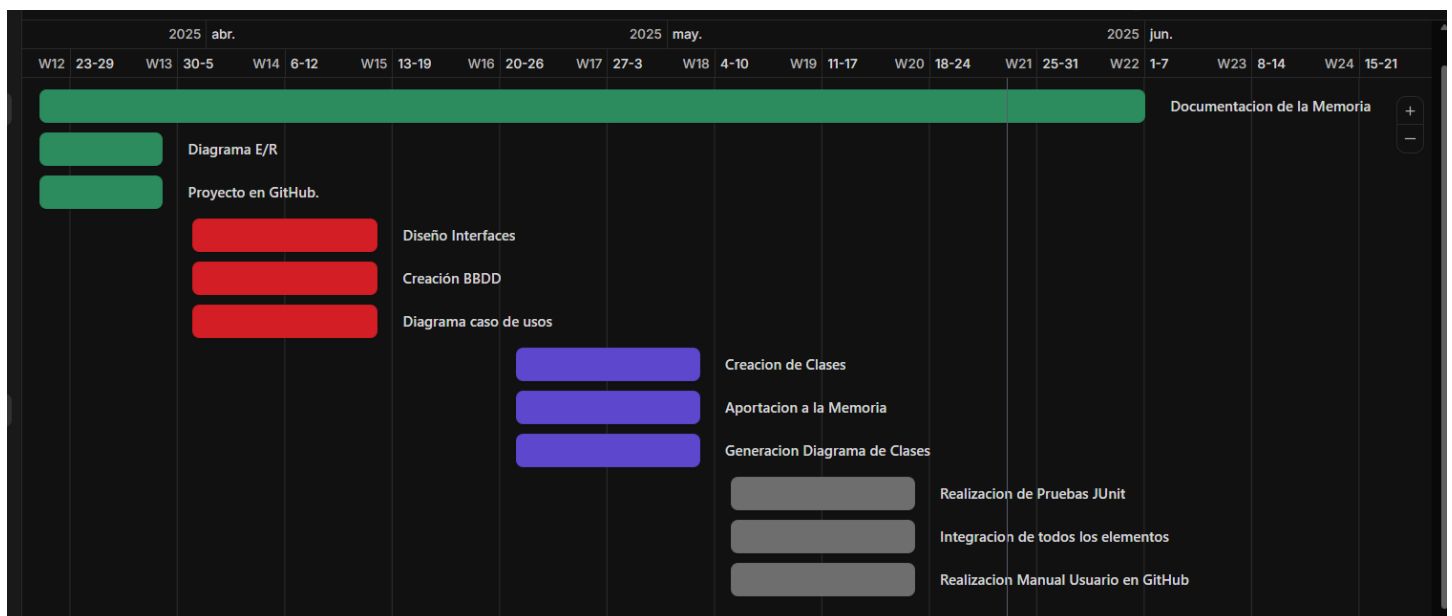
## 5. Metodología

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.

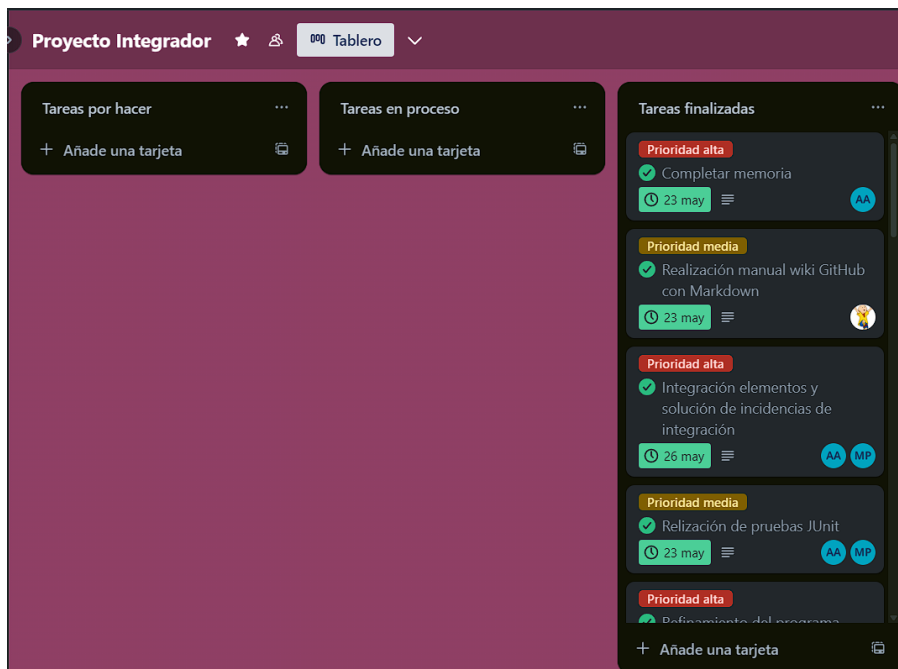


Además de la metodología SCRUM, también se ha aplicado la metodología Kanban a través de la plataforma Trello. Esta consiste en un tablero dividido en columnas donde se representa el

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 ha empleado un tablero Trello destinado a la coordinación y distribución de tareas.

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



A su vez, se ha utilizado 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 ha empleado un servidor de Discord con distintos canales temáticos sobre el proyecto.

También se ha empleado WhatsApp en ocasiones para establecer una comunicación más directa.

## 6. Resultados y conclusiones

Durante el desarrollo de este proyecto se logró construir una aplicación completa orientada a usuarios y monitores del centro deportivo de la universidad. La aplicación permite el registro, visualización y gestión de actividades en una base de datos MySQL, con acceso diferenciado según el tipo de usuario. Los monitores pueden crear, editar y eliminar actividades, así como consultar las salas disponibles, mientras que los usuarios pueden visualizar todas las actividades disponibles y gestionar sus inscripciones.

Uno de los resultados más relevantes es la implementación de controles de integridad lógica en operaciones como la inscripción en actividades o la creación de nuevas actividades y edición de estas evitando duplicidades y sobreocupación. Se han incorporado de manera efectiva validaciones para evitar que un usuario se inscriba dos veces en una misma actividad o que se creen actividades duplicadas. Sin duda este ha sido uno de los mayores retos a la hora de afrontar el proyecto.

Se ha desarrollado una interfaz gráfica intuitiva y funcional utilizando Swing, que permite a los usuarios navegar fácilmente por menús, consultar datos personales y realizar acciones específicas dependiendo de su rol. Todos los formularios están conectados a la base de datos mediante una única clase de acceso, lo que facilita el mantenimiento y la reutilización del código.

El grupo se encuentra satisfecho con el resultado del proyecto, ya que se han cumplido los objetivos de funcionalidad de la aplicación y se ha logrado construir esta de forma operativa. Sin embargo, el grupo ha considerado también que el desarrollo del proyecto habría sido más fluido y eficiente si se hubiera contado desde el inicio con unos conocimientos más sólidos y asentados sobre aspectos como el trabajo grupal con git, la estructura de vistas y controladores en Java o la realización de pruebas con JUnit.

Esta experiencia ha sido, no obstante, muy enriquecedora y ha permitido afianzar y ampliar significativamente las competencias desarrolladas durante el curso académico.

## 7. Trabajos futuros

A pesar de haber alcanzado los objetivos iniciales, el sistema presenta margen de mejora tanto en funcionalidad como en diseño y escalabilidad. Una de las mejoras más relevantes sería

implementar un sistema de roles más robusto. Actualmente, la distinción entre monitores y usuarios se basa en atributos concretos como el “ciclo”, lo que puede resultar poco flexible. En un futuro, se podría introducir un sistema de autenticación con roles diferenciados a nivel de base de datos y lógica de negocio, lo que permitiría un control más amplio de los permisos de cada tipo de usuario.

Otra mejora considerable sería la introducción de validaciones más complejas y gestión de errores más detallada. Aunque el sistema actual controla algunos errores como fechas inválidas o duplicidad de actividades, sería útil añadir comprobaciones más avanzadas como limitar acciones según el estado de una actividad (finalizada, en curso, etc.) o controlar intentos repetidos de inscripción de manera más informativa.

Desde el punto de vista de la interfaz gráfica, también se pueden realizar mejoras. Actualmente, la experiencia del usuario es funcional pero bastante básica. Las ventanas y los paneles constan de un diseño simple y visualmente presentan una gran posibilidad de mejora.

## **Anexos**

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

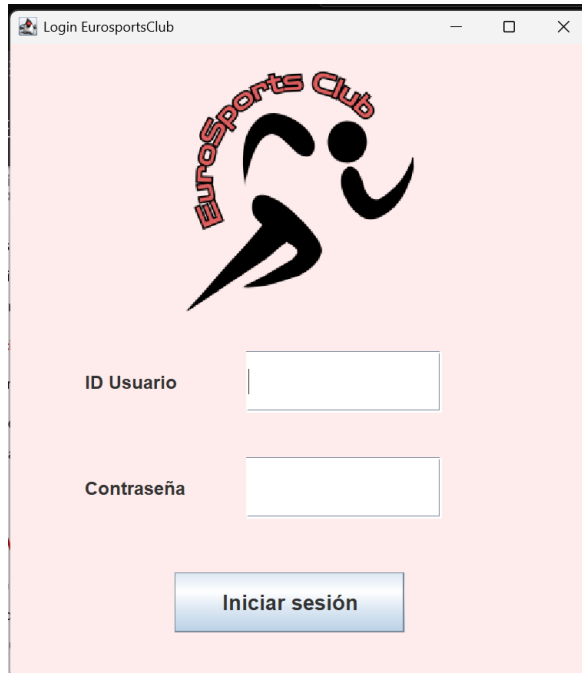
Para garantizar el correcto funcionamiento de la aplicación se ha de disponer de los siguientes requisitos:

En cuanto al hardware, los requisitos no son especialmente exigentes. Los requisitos mínimos son un procesador de 2 núcleos, 4GB de memoria RAM, un disco duro con más de 1GB de espacio libre para almacenar Eclipse (más proyectos y dependencias), un sistema operativo Windows10/ macOS 10.14+ /Linux y JDK11 o superior.

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

El funcionamiento de la aplicación es el siguiente:

Al ejecutarla, la primera ventana en aparecer es la ventana correspondiente al Login de la aplicación. En ella el usuario debe introducir su ID de usuario y la contraseña correspondiente.

A screenshot of a web application window titled "Login EurosportsClub". The window has a light pink background. At the top center is the Eurosports Club logo, which consists of a stylized black figure in a dynamic pose with the text "EuroSports Club" in a red, curved font above it. Below the logo, there are two input fields: the first is labeled "ID Usuario" and the second is labeled "Contraseña". Below these fields is a blue button with the text "Iniciar sesión".

Si el usuario ha iniciado sesión correctamente y este es un alumno del ciclo TAFD, aparecerá una ventana preguntándole si quiere acceder a la aplicación como monitor o como usuario.

A screenshot of a web application window with a light pink background. At the top center is the Eurosports Club logo. Below the logo, the text "¿Accedes como usuario o monitor?" is displayed. At the bottom of the window, there are two red buttons: the left one is labeled "Monitor" and the right one is labeled "Usuario".

Si pulsa en el botón monitor, accede al sistema como monitor y se abre el panel con el menú principal del monitor.

Si pulsa en el botón usuario, accede al sistema como usuario y se abre el panel con el menú principal del usuario.

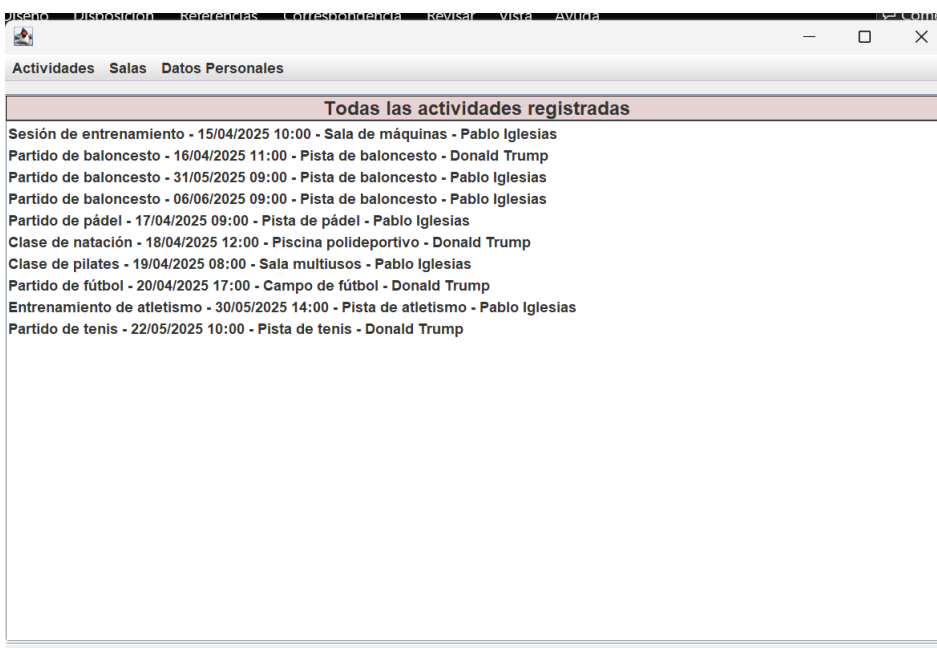
Si el usuario que ha realizado el inicio de sesión no cursa TAFD, se abre el panel con el menú principal del usuario directamente.

### Menú principal del monitor:



### Funcionalidades del monitor:

Al pulsar en la opción de “Ver todas las actividades”, aparece un panel informativo con la lista de actividades en el sistema:

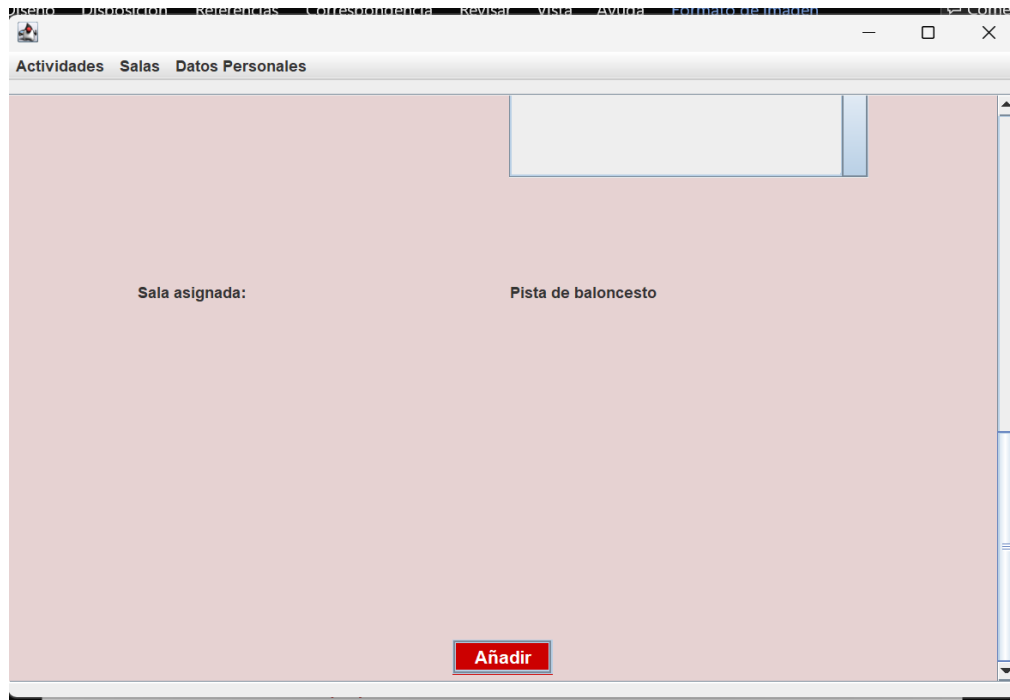




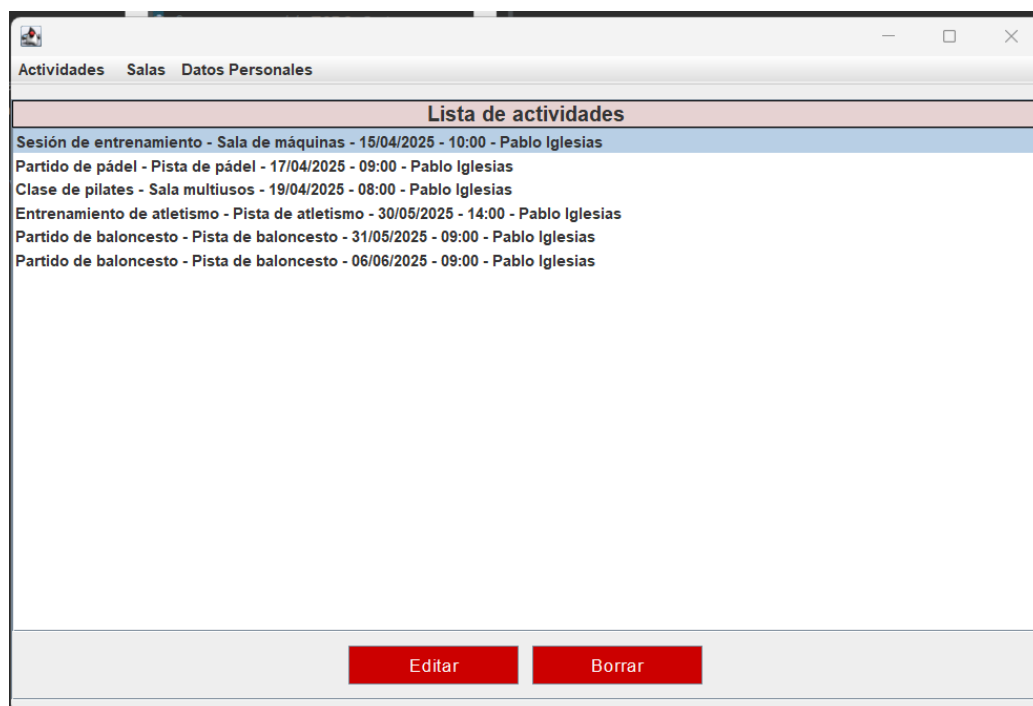
Al pulsar en la opción “Nueva actividad”, el monitor ve un panel donde puede crear una actividad nueva eligiendo su tipo y su fecha y hora.

The screenshot shows a web application window titled 'Añadir una nueva actividad'. The interface includes a top navigation bar with tabs: 'Actividades', 'Salas', and 'Datos Personales'. The main content area has a light pink background. On the left, there are labels for 'Monitor:' and 'Actividad:'. The 'Monitor:' field is populated with 'Pablo Iglesias'. The 'Actividad:' field has a dropdown menu open, showing a list of activity types: 'Sesión de entrenamiento', 'Partido de baloncesto' (highlighted), 'Partido de pádel', 'Clase de natación', 'Clase de pilates', 'Partido de fútbol', 'Entrenamiento de atletismo', and 'Partido de tenis'. Below the dropdown, there are three input fields for 'Fecha:' with labels 'Día', 'Mes', and 'Año'.

This screenshot shows the same form as the previous one, but with the date and time fields filled out. The 'Fecha:' section now displays three dropdown menus for 'Día', 'Mes', and 'Año', with values '24', '05', and '2025' respectively. Below this, the 'Hora:' field has a dropdown menu showing '09:00'. The 'Sala asignada:' field at the bottom is now populated with 'Pista de baloncesto'.



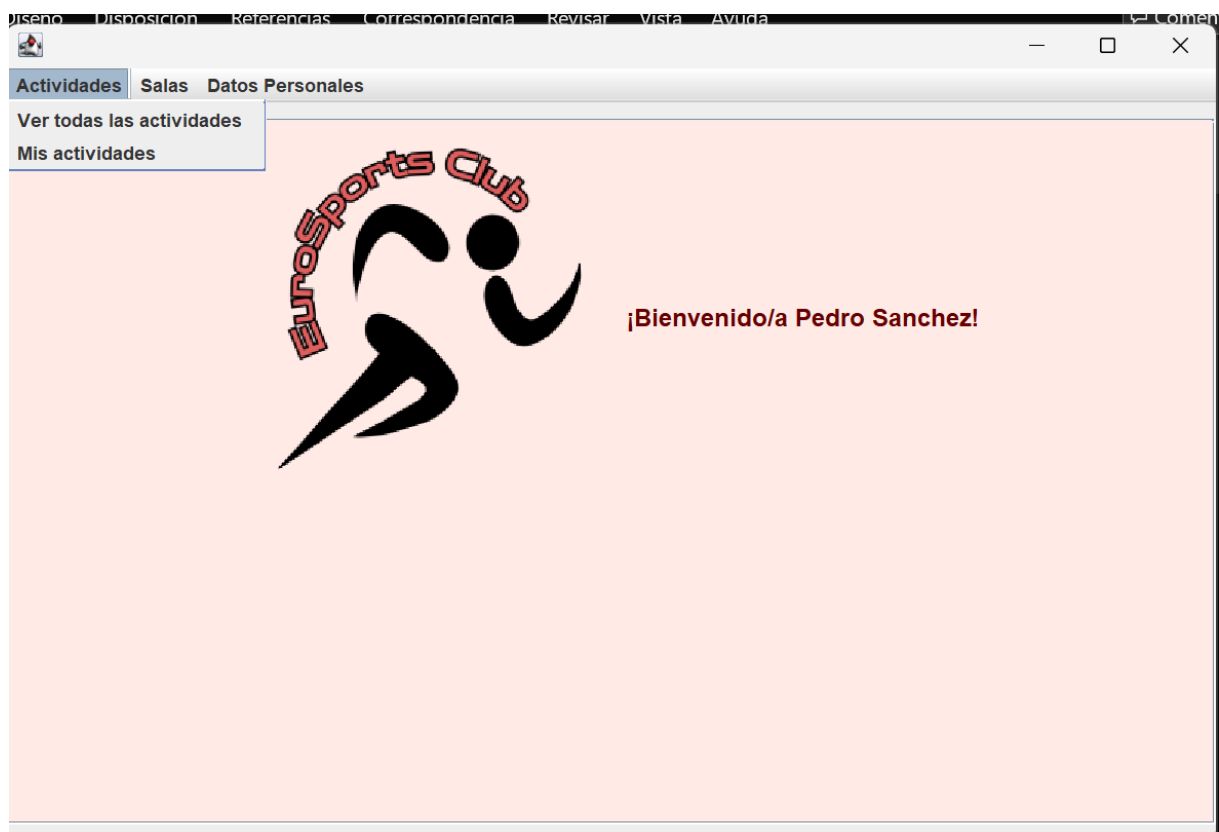
Al pulsar en “Mis actividades” el monitor ve un panel con la lista de actividades que él imparte, pudiendo editar y borrar estas mediante los botones correspondientes:



Si pulsa en “Editar” se abrirá el siguiente panel donde puede editar la fecha y hora de la actividad seleccionada en la lista y guardar los cambios.

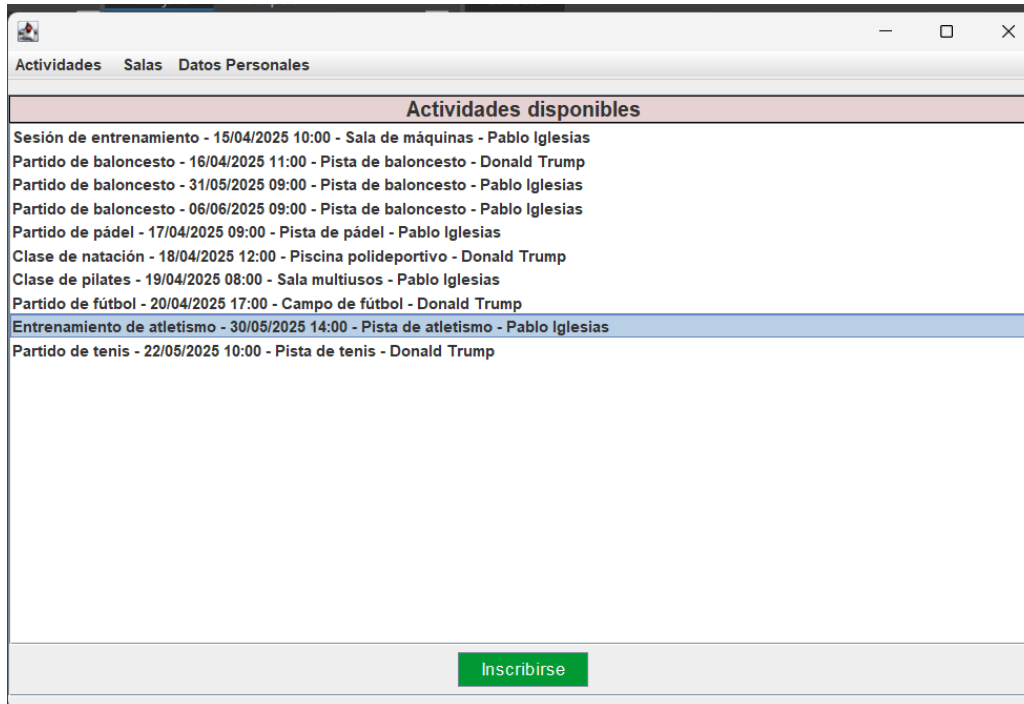
A screenshot of a web application window titled 'Editar actividad: Sesión de entrenamiento'. The window has a menu bar with 'Actividades', 'Salas', and 'Datos Personales'. The main content area has a light pink background. In the center, there are four dropdown menus for date and time selection: 'Día:' with '15', 'Mes:' with '04', 'Año:' with '2025', and 'Hora:' with '10:00'. Below these is a red button labeled 'Guardar cambios'.

### Menú principal del usuario:

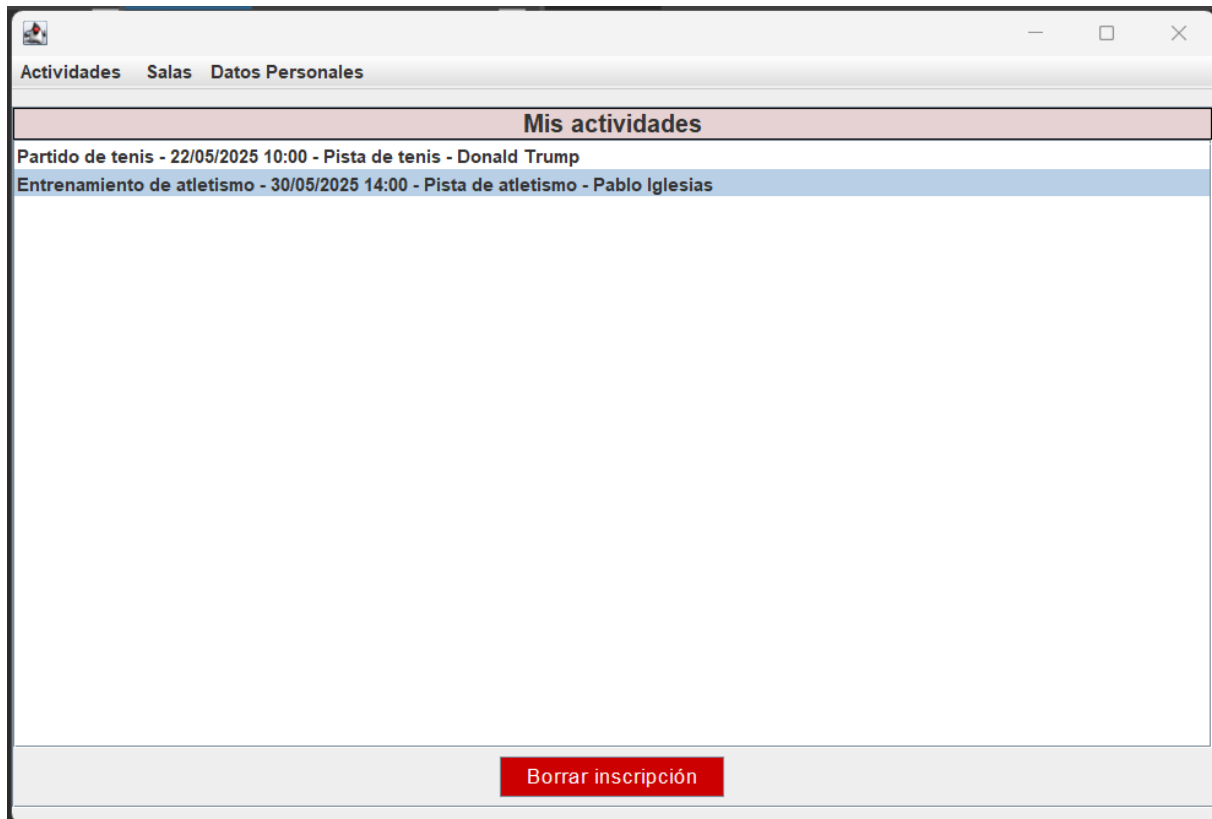


### Funcionalidades del usuario:

Al pulsar el botón “Ver todas las actividades” el usuario ve un panel con la lista de actividades que puede seleccionar para inscribirse pulsando en el botón “Inscribirse”.

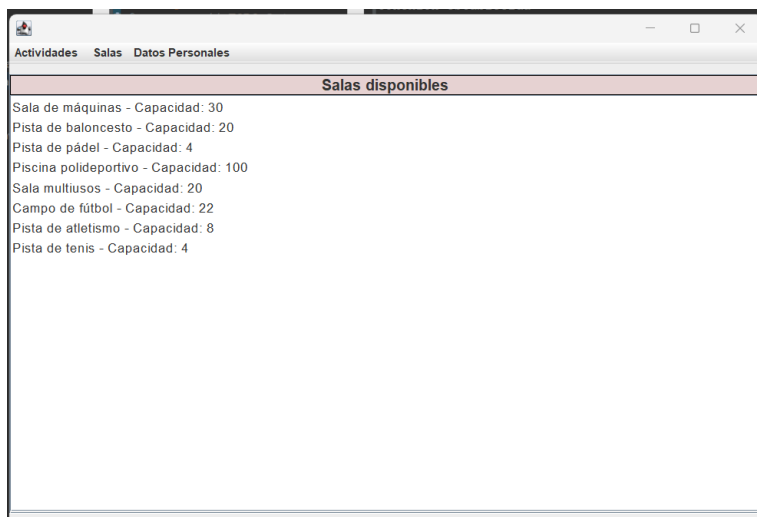


Al pulsar en la opción “Mis actividades” el usuario ve un panel con la lista de actividades en la que se ha inscrito, pudiendo borrar la inscripción de la actividad que seleccione con el botón “Borrar inscripción”:



### Funcionalidades comunes:

Al pulsar en “Ver salas” se obtiene un panel con una lista de las salas y su capacidad:



Finalmente, al pulsar en “Ver Datos” en el apartado de Datos Personales, aparece un panel con la información del usuario que inició sesión en el sistema y se le ofrece la posibilidad de cambiar su contraseña:

The screenshot shows a web application window with a title bar containing a small icon and standard window controls (minimize, maximize, close). Below the title bar is a navigation menu with three items: 'Actividades', 'Salas', and 'Datos Personales'. The main content area is titled 'Datos personales' and contains a form with the following fields and labels:

- Nombre:
- Apellidos:
- Ciclo:
- Matrícula:
- Contraseña actual:
- Nueva contraseña:
- Confirmar nueva:

Below the password fields is a red button with the text 'Cambiar contraseña'.

### Anexo III – Historia del proyecto

Durante los 2 primeros sprints, la mayoría del trabajo se ha realizado en clase con todos los integrantes del grupo presentes. Sin embargo, en el tercer sprint se ha enfrentado la dificultad de no poder contar con un miembro del equipo debido al inicio de prácticas en empresa.

Para la comunicación y buen desarrollo del trabajo se ha mantenido contacto con esta persona principalmente vía WhatsApp ya que se ha considerado un contacto más directo que a través de Discord.

Una vez distribuida la carga de trabajo entre los miembros del equipo, cada uno ha realizado su parte añadiendo las modificaciones al repositorio de GitHub y mediante el envío de partes de código por discord.