

# Proyecto final

## Exploración



# **Índice:**

**Introducción**

**Herramientas - Asignaturas**

**Diagramas - Diseño inicial**

**Base de datos - Diseño inicial**

## **Introducción**

En esta fase vamos a elegir las herramientas y fuentes de información que vamos a usar en cada asignatura y en cada proceso creativo de nuestro software pasando por las aplicaciones para desarrollar y diseñar cada cosa y finalmente mostraremos los resultados del desarrollo de estos diagramas y elementos iniciales de la aplicación, base de datos y su funcionamiento.

## **Herramientas - Asignaturas**

### **Programación**

Para el diseño del CRUD que permite acceder a la base de datos, utilizaremos el IDE Apache NetBeans en su versión 23, en el lenguaje de programación Java con el entorno de desarrollo de interfaces JavaFX de Ant en su versión 8. En caso de que tengamos algún problema en el desarrollo de la aplicación, usaremos como soporte inteligencias artificiales como Gemini o Chat GPT, foros de Reddit y Stack Overflow.



### **Sistemas Informáticos**

Para sistemas informáticos vamos a implementar un servidor, que alojará un servicio web, y un servicio de bases de datos de MySQL. Este servidor se basará en Debian 12 Bookworm (utilizando una máquina virtual en VMware Workstation Pro). Para la instalación del servidor web vamos a usar Apache Webserver dentro de la herramienta Webmin, que facilitará la creación y administración del servidor web, y para la base de datos usaremos MySQL Workbench para Linux.



### **Bases de Datos**

Para la base de datos, iniciaremos con los diagramas de entidad-relación y el modelo relacional en la aplicación draw.io, que permite su diseño de una manera efectiva y simple. Para la creación física de la base de datos vamos

a utilizar MySQL Workbench 8.0 para programarla, soportada por MySQL 9.1. Como ya he comentado anteriormente, alojaremos el servidor en nuestro servidor, para mantener la base de datos en un entorno seguro y actualizado.



draw.io



### **Entornos de desarrollo**

Para entornos de desarrollo vamos a mantener el proyecto actualizado mediante la subida diaria del proyecto a la plataforma en línea de repositorios remotos Github, donde cada integrante del equipo va a subir sus cambios al repositorio remoto para poder trabajar cómodamente y de manera organizada. También vamos a realizar los diagramas de desarrollo a mano, y posteriormente los convertiremos en digital con la herramienta draw.io que hemos mostrado anteriormente.



### **Lenguaje de marcas**

Para lenguajes de marcas vamos a desarrollar la web con HTML5, CSS3 y Bootstrap en el avanzado editor de texto Visual Studio Code

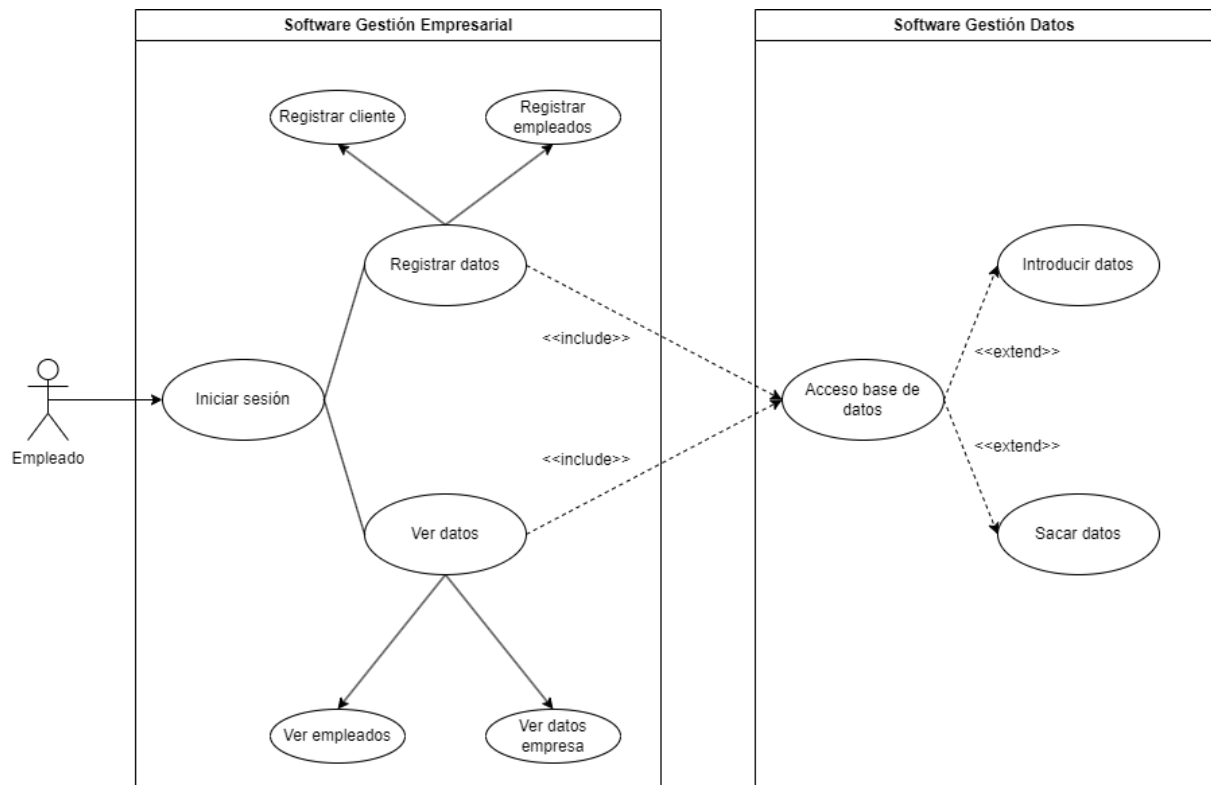


Visual Studio Code



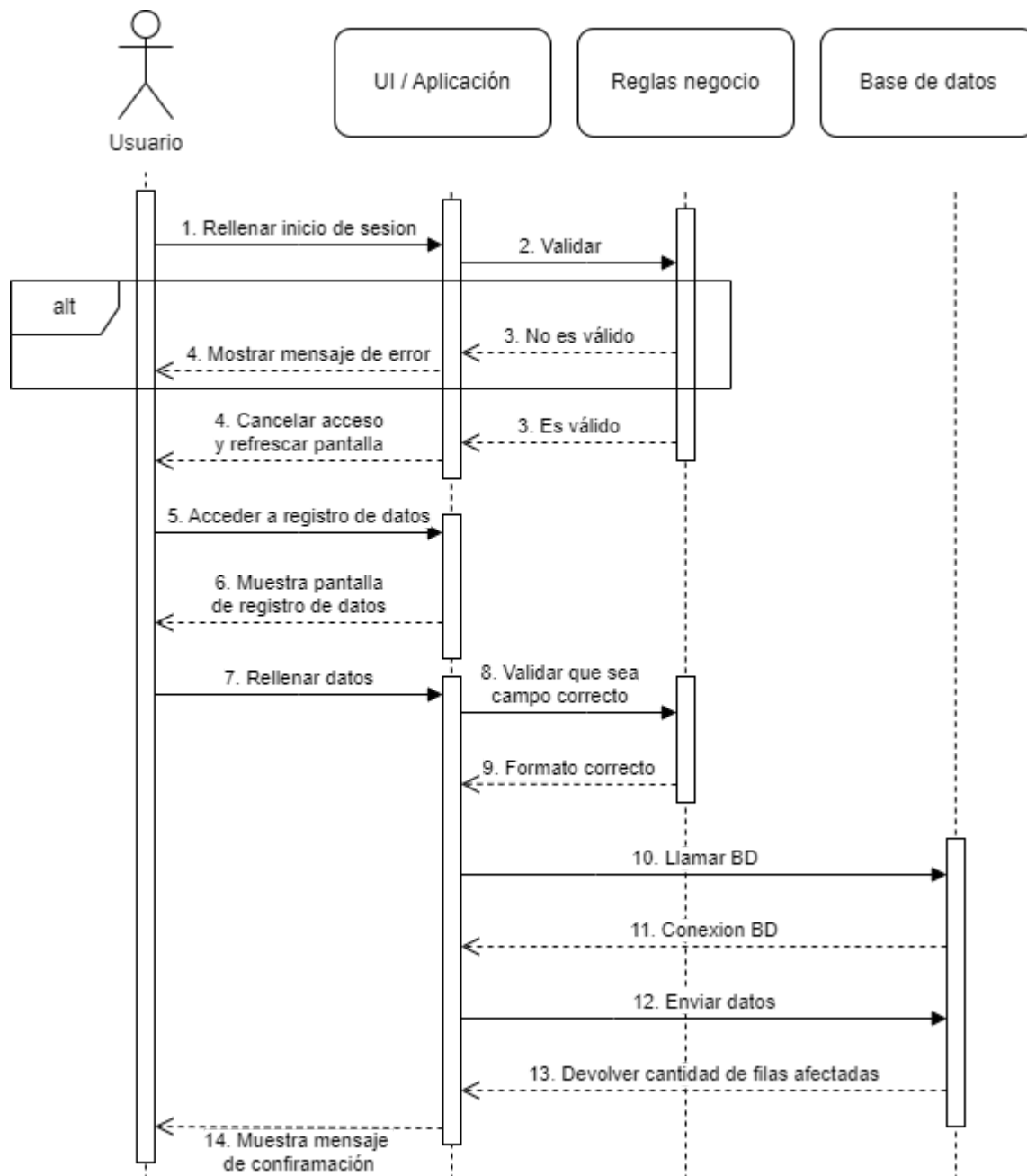
## **Diagramas - Diseño inicial**

Hemos empezado pensando en los casos de uso que se pueden realizar en nuestra aplicación y en base a ello hemos realizado un diagrama del cual hemos sacado los dos diagramas de secuencia que vemos posibles al usar la aplicación.



El usuario (trabajador de jGym) debe iniciar sesión al abrir la aplicación de gestión empresarial y después puede elegir si quiere introducir nuevos datos o consultarlos en la base de datos, hay un par de ejemplos de herencia en cada opción (“Registrar cliente”, “Ver empleados”, etc.) y al elegir uno, la aplicación procede a la conexión a la base de datos la cual según la opción elegida introduce o saca datos.

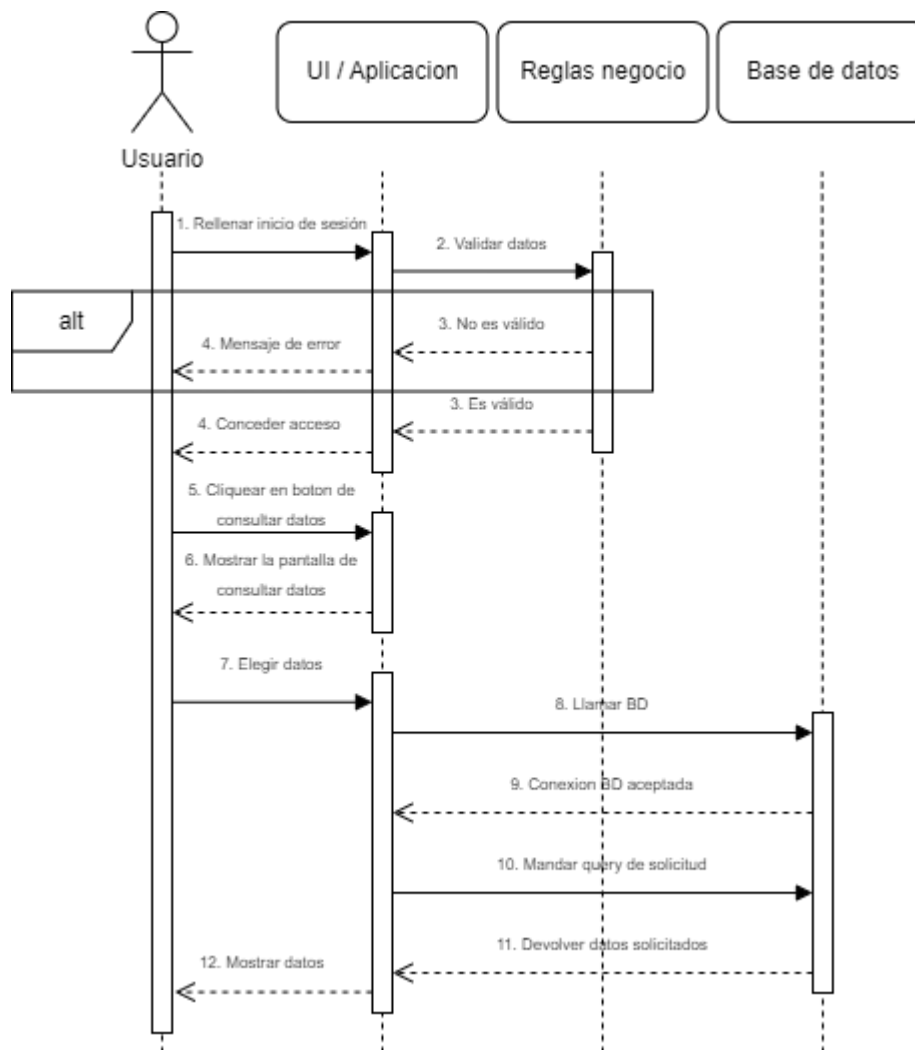
De aquí hemos sacado dos diagramas de secuencia según si el trabajador elige meter o sacar datos:



Este primer diagrama es el de registro de datos, el usuario inicia la aplicación, que valida los datos rellenos en el inicio de sesión y después la aplicación muestra la pantalla inicial de la aplicación, ahí el usuario elige el botón de acceder a la pantalla para registrar datos, después rellena los datos que quiera introducir, la aplicación valida que tengan un formato correcto para que la base de datos lo pueda registrar correctamente mediante las reglas del negocio, que son las credenciales que tengamos guardadas para poder iniciar sesión en el código de la aplicación o base de datos. Después la aplicación realiza la conexión con la base de datos y le envía los datos. La base de datos devuelve la cantidad de filas afectadas por el registro (esto es así por la forma de programarlo que hemos pensado) y



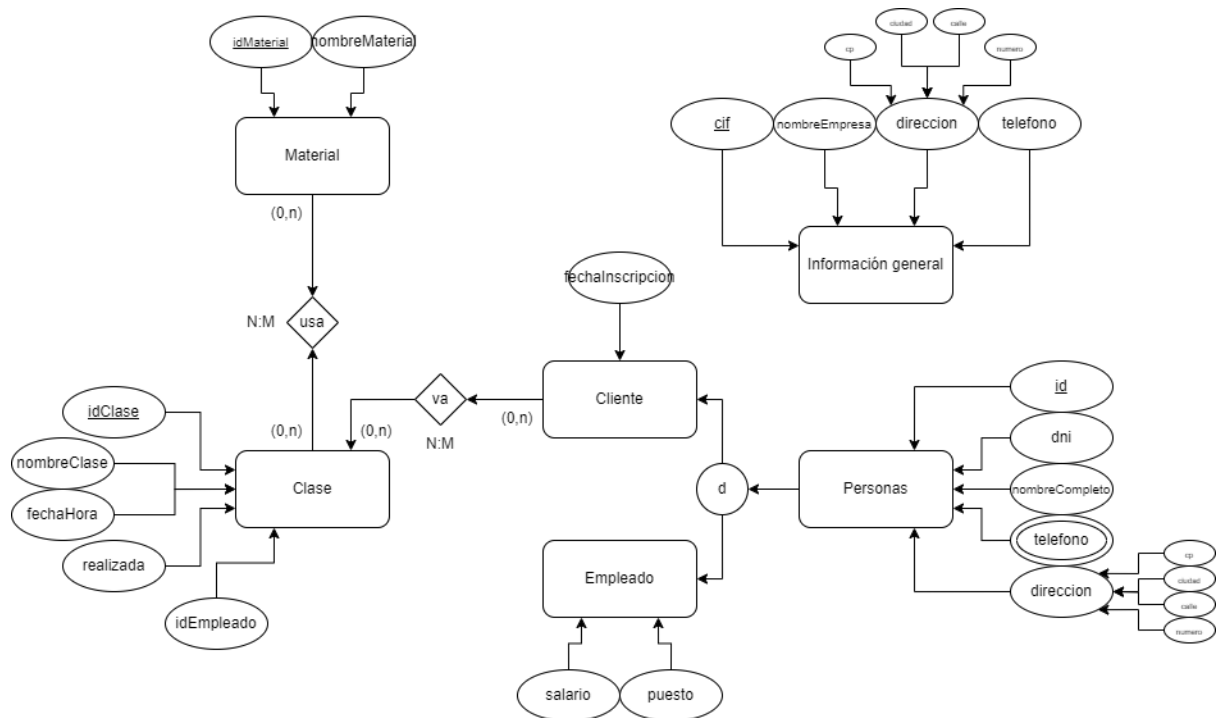
finalmente la aplicación muestra que los datos se han registrado correctamente.



Este segundo diagrama representa la consulta de información de la base de datos. Empieza igual: El usuario inicia sesión, la aplicación revisa que sea correcta con las reglas del negocio... El trabajador elige la pantalla de consultar datos y la recibe del programa, después elige los datos que quiere ver y la aplicación se conecta con la base de datos y manda la solicitud de los datos que quiere obtener. La base de datos devuelve la información solicitada y finalmente la aplicación le muestra al usuario lo que pidió.

## **Base de datos - Diseño inicial**

Para el diseño de la base de datos, primero hemos pensado en diseñar un diagrama de entidad-relación para usar de base para el desarrollo:



Hemos pensado en que nuestro programa debería tener elementos como información general para que los trabajadores puedan ver elementos básicos del gimnasio (CIF, dirección, teléfono, etc.). También queremos añadir una tabla llamada “personas”, de la que hereden las tablas para almacenar “cliente” y “empleado”. Ambas dos tendrán id, DNI, nombre completo, teléfono(s) y dirección y cada una tendrá otros elementos, los empleados tendrán guardado también su salario y los clientes la fecha en la que se inscribieron a “jGym”. Hemos relacionado la clase cliente con una que guarda las clases que daremos en el gimnasio para registrar quienes asistieron, la hora que se hizo la clase, su nombre y el empleado que la impartió. Por último, habrá una tabla que contenga el material del que dispone el gimnasio como mancuernas, bancos y otro tipo de máquinas y elementos. Ésta última está relacionada con la tabla “clase” para evitar problemas en caso de que desaparezca material del gimnasio para saber su paradero y cuándo, cómo y dónde se ha usado cada cosa.

Cabe destacar que éste es un esquema inicial y está abierto a cambios que pueda necesitar durante el proceso de creación

En base al diagrama entidad-relación hemos desarrollado uno de modelo relacional en el que se dispone de mejor manera los elementos que serán claves primarias y foráneas en la base de datos.

