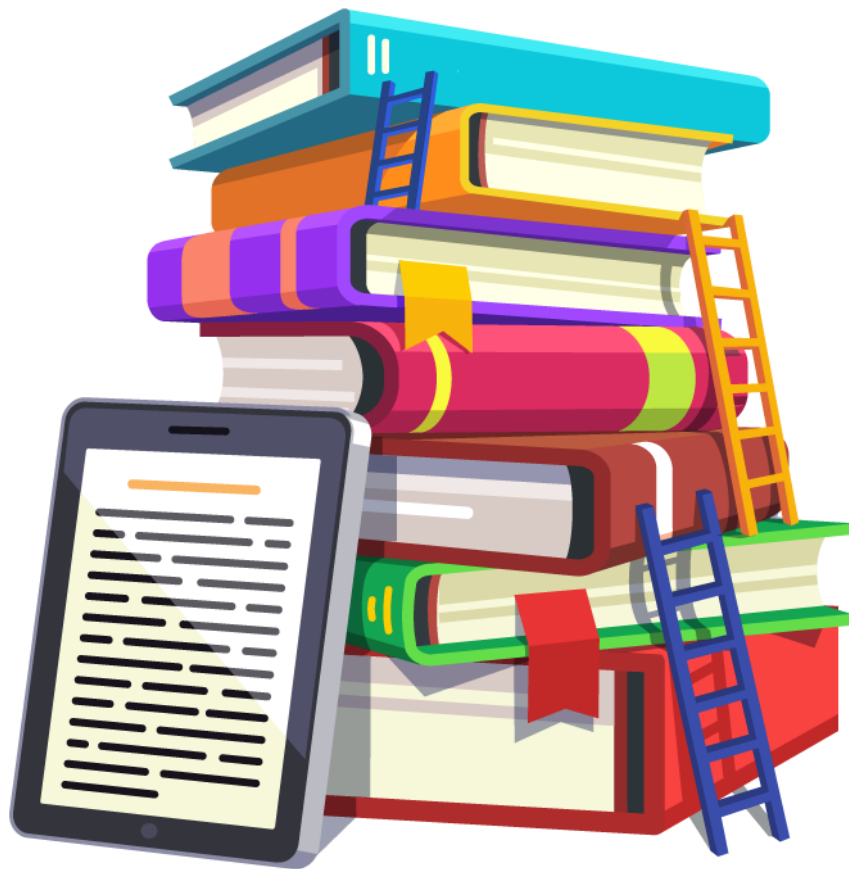


# TRABAJO DE ACCESO A DATOS

*Práctica 1*



**Leire Gorricho**

30/10/2022

## INTRODUCCIÓN

He creado una aplicación para una biblioteca. A ella van a poder acceder tanto empleados como clientes. Los empleados podrán ver, añadir, eliminar y editar los libros, los clientes y los préstamos, así como exportar los ficheros binarios (.dat) a ficheros XML. Los clientes solo podrán ver los libros que hay en la biblioteca. La aplicación está enfocada en ayudar a los empleados de la biblioteca a gestionar los datos.

Cuando un cliente quiera pedir prestado un libro tendrá que ir a donde el empleado y este último creará el préstamo. Al principio mi idea era que el cliente pudiera crear el préstamo, pero tendría que controlar que como nombre del cliente que pide el préstamo sólo pudiera poner su propio nombre y por otro lado pensé que en la realidad, cuando vas a una biblioteca y quieres pedir prestado un libro te lo hace el bibliotecario.

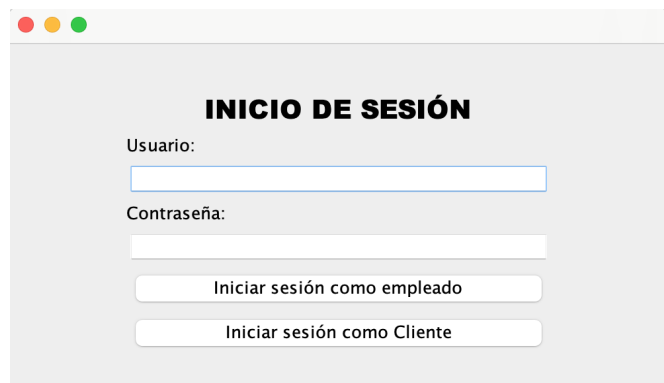
Por la aplicación se navega a través de ventanas, me parecía más cómodo hacerlo por interfaz gráfica sobre todo para editar datos. La aplicación no es muy bonita que digamos pero me he centrado en que todo funcione bien y se controlen los errores y si me daba tiempo podía investigar sobre cómo hacerla más agradable a la vista.

Para ingresar a la aplicación como empleado he creado un usuario admin con contraseña admin. Y para ingresar como cliente hay un usuario cliente con contraseña cliente. Para que los datos se carguen y la aplicación pueda funcionar bien, hay que runear las clases de crear los ficheros(CrearFicheroLibros, CrearFicheroClientes, CrearFicheroPrestamos, CrearFicheroEmpleados), sino en el login da error de archivo no encontrado.

## MANUAL DE USUARIO

### VENTANA PRINCIPAL

En la ventana principal nos piden nuestro usuario y contraseña para entrar a la aplicación, podemos entrar como empleados o como clientes. Para hacer el login tenía varias opciones: que los clientes pudieran entrar sin credenciales y que solo los empleados tuvieran que registrarse, tener un único



The screenshot shows a login window titled "INICIO DE SESIÓN". It contains two input fields: "Usuario:" and "Contraseña:". Below the password field are two buttons: "Iniciar sesión como empleado" and "Iniciar sesión como Cliente". The window has a standard macOS-style title bar with red, yellow, and green buttons.

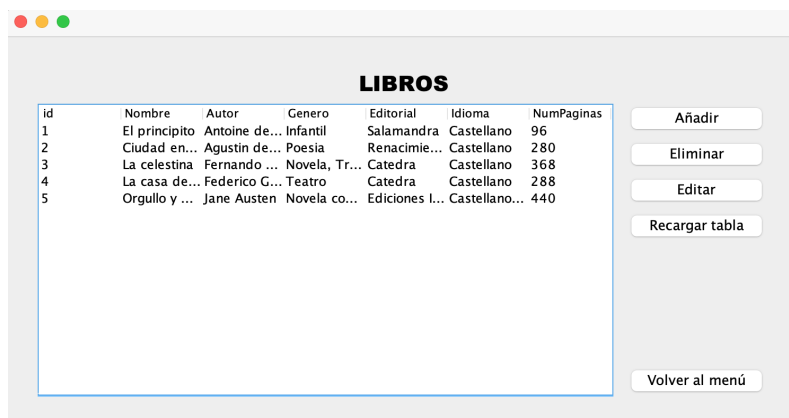
botón de inicio de sesión y por último, hacer un botón para el inicio de sesión de los empleados y otro para los clientes. Esta opción no sería el método más seguro pero para que la comprobación de usuario y la redirección a los menús me parecía más sencillo.

## MENÚ DE LOS EMPLEADOS

Los empleados pueden gestionar los libros, los clientes y los préstamos, y además, pueden exportar a un fichero xml los préstamos mediante DOM, o exportar todos los ficheros binarios a xml con XStream. Al cerrar sesión saldrán del menú de empleado y volverán a la ventana principal. Tanto los libros como los clientes se pueden ver, añadir, eliminar y editar, los préstamos no se pueden editar.

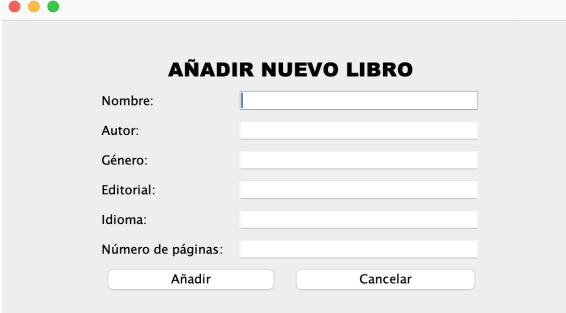


## LIBROS



En esta ventana podemos ver los libros que hay en la biblioteca. Podemos añadir, eliminar y editar libros. El botón de recargar tabla recarga los datos. Para eliminar un libro, lo seleccionamos y le damos a eliminar.

Para añadir un libro nuevo tenemos que insertar los datos que nos piden y darle al botón de añadir.



AÑADIR NUEVO LIBRO

Nombre:

Autor:

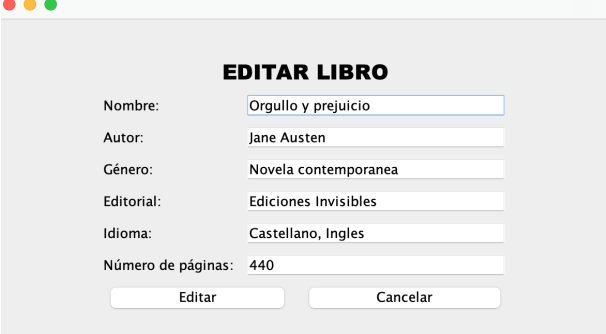
Género:

Editorial:

Idioma:

Número de páginas:

Para editar un libro tenemos que seleccionarlo en la tabla y pulsar el botón de editar. Nos aparecerá esta ventana con los datos rellenos del libro. Editamos lo que queramos modificar y la damos al botón editar.



EDITAR LIBRO

Nombre:

Autor:

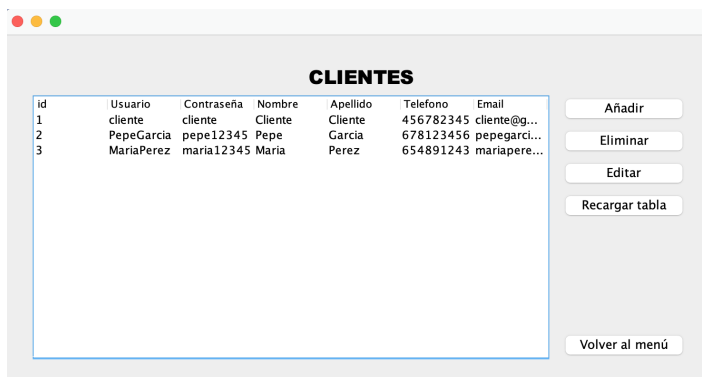
Género:

Editorial:

Idioma:

Número de páginas:

## CLIENTES

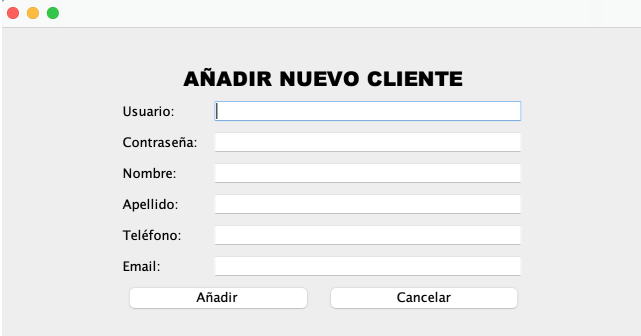


CLIENTES

id	Usuario	Contraseña	Nombre	Apellido	Telefono	Email
1	cliente	cliente	Cliente	Cliente	456782345	cliente@g...
2	PepeGarcia	pepe12345	Pepe	Garcia	678123456	pepegarci...
3	MariaPerez	maria12345	Maria	Perez	654891243	mariapere...

Con los clientes es igual que con los libros. En la ventana vemos los clientes y podemos añadir uno nuevo, editar y eliminar otro. Para eliminar un cliente, lo seleccionamos y le damos a eliminar.

Para añadir un cliente nuevo tenemos que insertar los datos que nos piden y darle al botón de añadir.



AÑADIR NUEVO CLIENTE

Usuario:

Contraseña:


Nombre:

Apellido:

Teléfono:

Email:

Para editar un cliente tenemos que seleccionarlo en la tabla y pulsar el botón de editar. Nos aparecerá esta ventana con los datos rellenos del cliente. Editamos lo que queramos modificar y la damos al botón editar.



**EDITAR CLIENTE**

Usuario:

Contraseña:

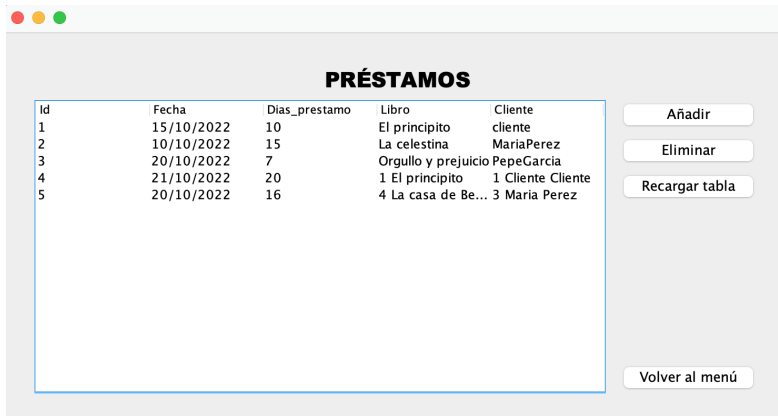
Nombre:

Apellido:

Teléfono:

Email:

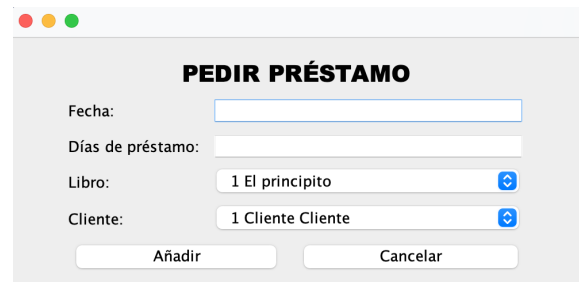
## PRÉSTAMOS



Id	Fecha	Días_prestamo	Libro	Cliente
1	15/10/2022	10	El principito	cliente
2	10/10/2022	15	La celestina	MariaPerez
3	20/10/2022	7	Orgullo y prejuicio	PepeGarcia
4	21/10/2022	20	1 El principito	1 Cliente Cliente
5	20/10/2022	16	4 La casa de Be...	3 Maria Perez

Podemos ver los préstamos, añadirlos y eliminarlos. Para eliminar un préstamo, lo seleccionamos y le damos a eliminar.

Para añadir un préstamo nuevo tenemos que insertar los datos que nos piden y darle al botón de añadir.



**PEDIR PRÉSTAMO**

Fecha:

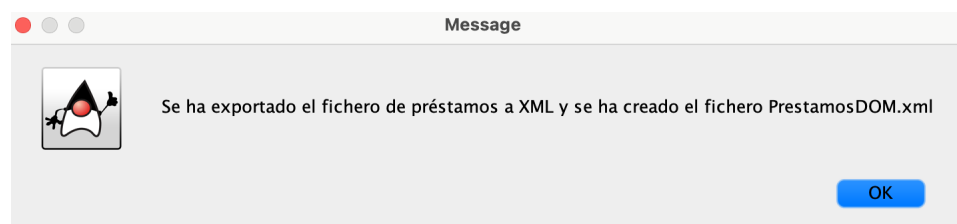
Días de préstamo:

Libro:

Cliente:

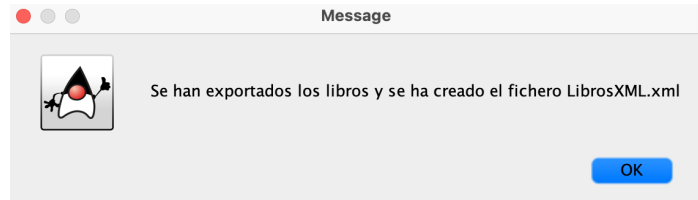
## EXPORTAR PRÉSTAMOS CON DOM

Para implementar la exportación a xml con DOM creo que es interesante exportar los préstamos, de esta manera se podrían manejar mejor los datos con un fichero xml. Al pulsar este botón se exportarán los préstamos a xml usando DOM y nos lo comunicará si no ha habido ningún fallo.



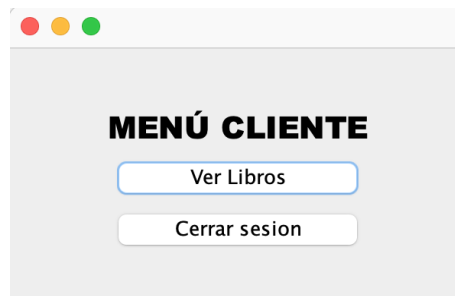
## EXPORTAR TODOS LOS FICHEROS CON XSTREAM

He pensado que el poder exportar todos los ficheros binarios a xml va a ser de utilidad para el próximo proyecto, y con XStream me resultaba más fácil. Al pulsar este botón se exportarán todos los ficheros.dat a xml usando XStream y nos irá comunicando, si no ha habido ningún fallo, mientras se van generando.



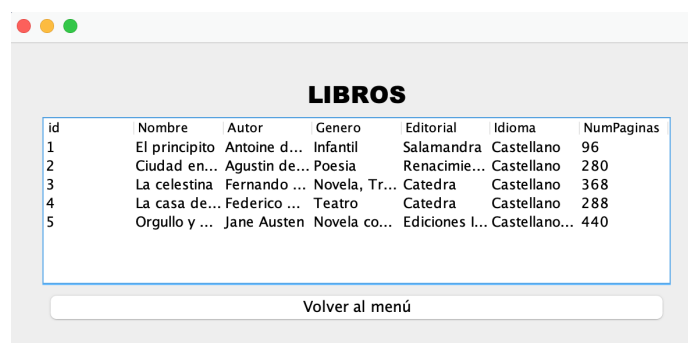
## MENÚ DE LOS CLIENTES

Los clientes solo pueden ver los libros. Como ya he dicho antes, la aplicación se centra en la gestión de datos por lo que quienes realmente van a usar la aplicación son los empleados de la biblioteca. Al principio había pensado en hacer una selección semanal con que recogiera algunos títulos de libros para luego exportarlo, pero pensé que tenía más sentido que el empleado pueda exportar datos, como por ejemplo, los préstamos. Al cerrar sesión saldrán del menú de empleado y volverán a la ventana principal.



## VER LIBROS

Como clientes podemos ver los libros que están disponibles en la biblioteca. Si queremos pedir un préstamo tendremos que pedirle a un empleado que lo cree rellenando los datos con el libro que queremos pedir y con nuestro usuario.



## DOCUMENTACIÓN DE CÓDIGO

### ADMINISTRACIÓN DE DATOS

La administración de los datos de libros, clientes y préstamos es igual, así que voy a coger de ejemplo los libros para explicar como he hecho para ver, añadir, eliminar y editar. He controlado todos los errores posibles (o al menos eso creo y espero) y que el programa ponga un mensajito cuando se añadan, eliminen o editen cosas. A la hora de insertar datos he controlado que estén no haya ningún campo sin rellenar, que si pide un número tiene que ser un int, y en teléfono tiene que tener 9 números.

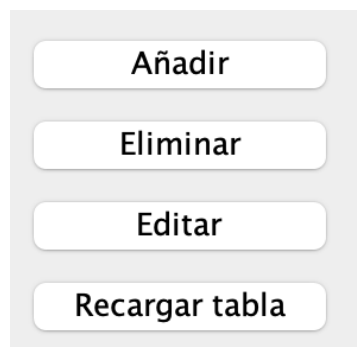
### ACCIONES DE LOS BOTONES

Al presionar el botón de añadir se abre la ventana de añadir libro, que recogerá los datos que insertemos, se asegurará de que son correctos y creará un nuevo libro. Para añadirlo, una función carga todos los libros, añade el nuevo y sobrescribe el archivo.

Para eliminar un libro se recogerá el id del libro que esté seleccionado en la tabla y una función lo eliminará. Para ello tiene que haber un libro seleccionado, sino nos avisará de que tenemos que seleccionarlo.

Para editar, la ventana de editar libro carga los datos del libro seleccionado en la tabla. Se comprueba que todos los datos sean correctos. Una función carga todos los libros, busca el libro que queremos editar, lo elimina e inserta el libro modificado.

Hay un botón de recargar tabla que está por si acaso, porque cuando realizamos alguna de estas acciones, las funciones que se encargan de eso también recargan los datos de nuevo en la tabla. Lee los datos y recarga la tabla.



## CLASES

Tengo las clases clientes, libros, préstamos y empleados, para crear objetos, con sus diferentes variables, constructor y getters y setters. Los empleados solo los uso para el login. En el caso de querer añadir un empleado nuevo lo tendrá que crear la persona responsable de la aplicación.

Para facilitar la exportación a xml y que no den errores, he creado unas clases simplemente tienen una lista del objeto, un constructor y un método add().

Otras clases son para crear los ficheros .dat. Y la clase Main que inicia el programa abriendo la ventana del login.

