



Copyright 2022. Todos los derechos reservados.

Salvatierra Bernal, Lucia – Cucalón Gutiérrez, Ángel

Contenido

Introducción	3
Diagrama UML	4
Diagrama Relacional	6
Descripción del proyecto	7
Funcionalidad de la aplicación	8
Pruebas de funcionalidad	12
Coordinación de grupo de trabajo	14
Bibliografía	15

Introducción

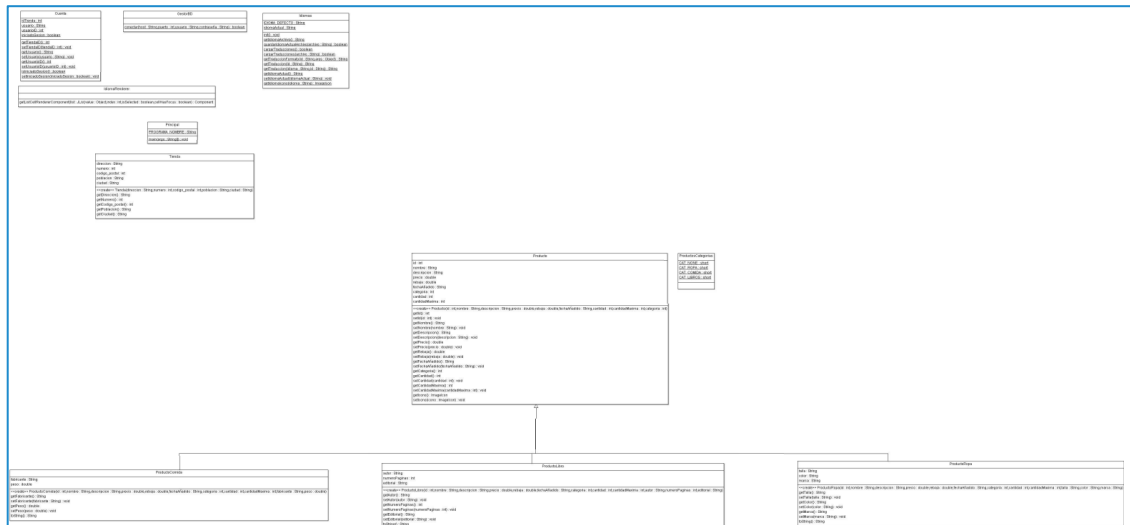
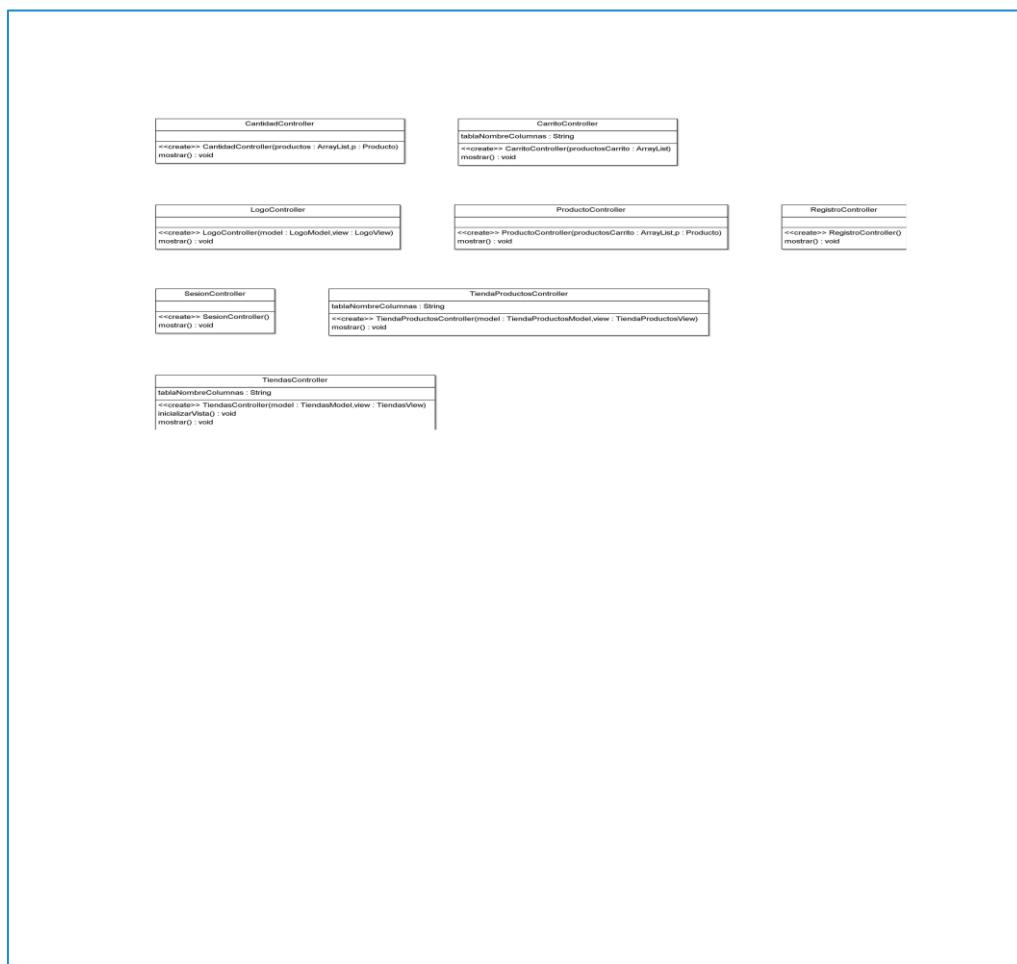
El presente proyecto se redacta con carácter de Trabajo Profesional de Fin de Grado, para la obtención de productos mediante una selección de tiendas, y según el deseo de un cliente. Es iniciativa del cliente aprovechar la cantidad de productos demanda por su abastecimiento. Para tal fin dispone de una selección de tiendas, en la ventana principal.

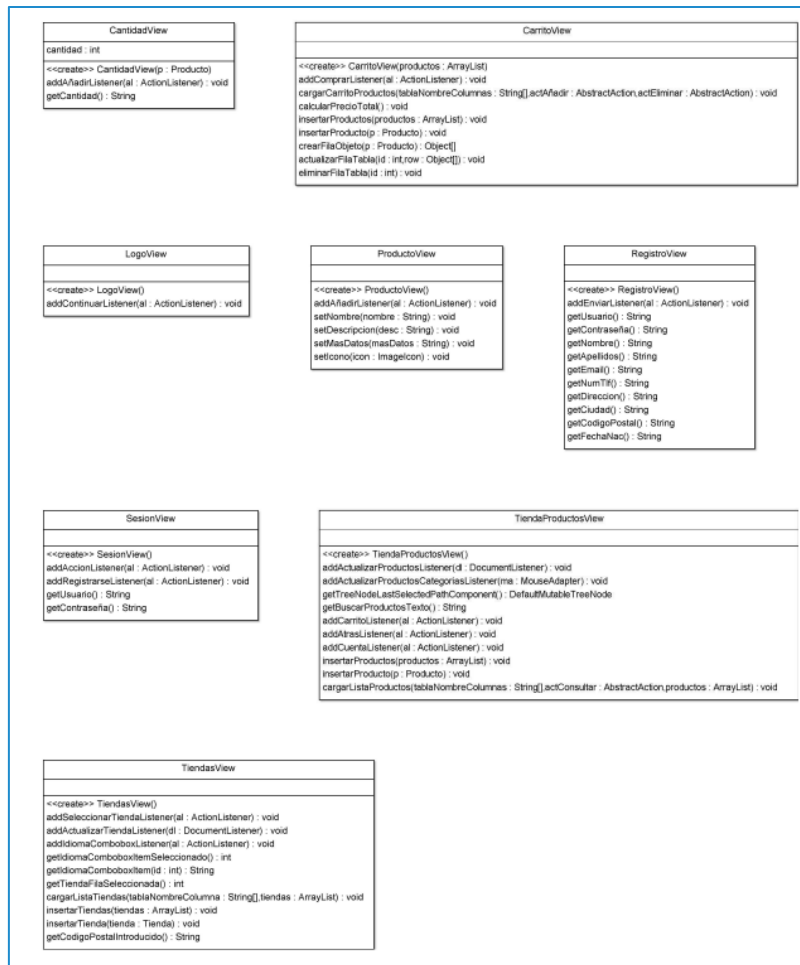
En los documentos que se presentan a continuación, se recogen todas las categorías y características de los productos que han sido obtenidos como resultado de la selección de una tienda. El proyecto consta de los documentos siguientes:

- Diseño UML
- Diseño Relacional

Diagrama UML

APP

CONTROLLERVISTA



MODELO

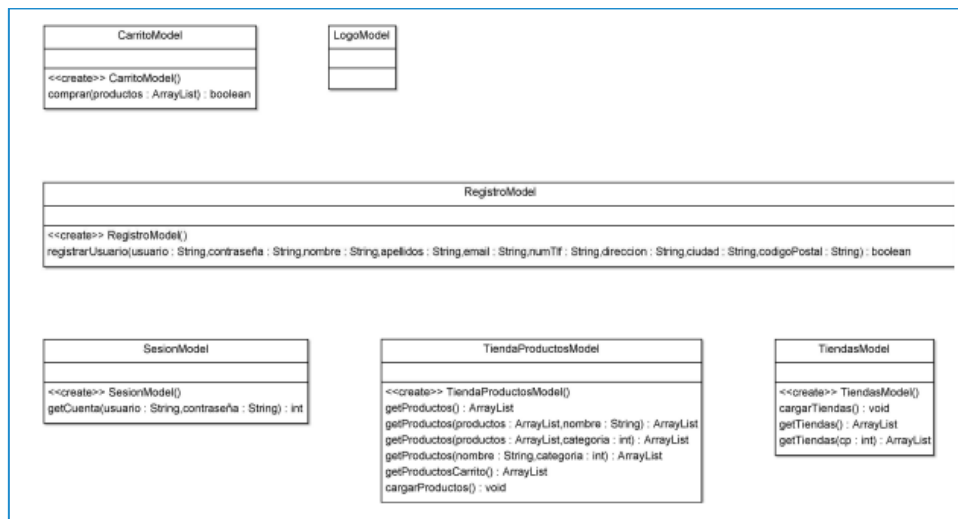
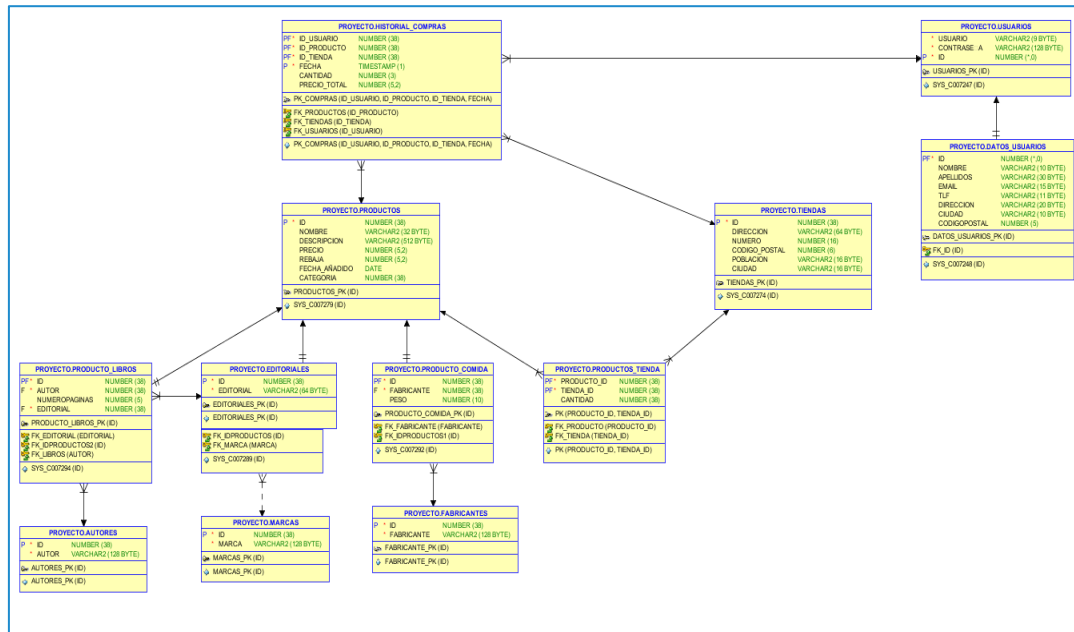


Diagrama Relacional



Descripción del proyecto

En los diagramas se recoge una vista de productos mediante la selección de una categoría (libros, ropa, comida), que desplegará una perspectiva de los productos que estos inducen una descripción y la cantidad existente en la tienda seleccionada.

- OBJETIVO DEL PROYECTO

El presente proyecto tiene como objetivo poder tener varias tiendas en alcance del cliente. Asimismo, el presente proyecto ha de servir como un servicio de compra rápida de productos ante los organismos competentes, al objeto de recabar de los mismos las ayudas financieras necesarias para su ejecución, en lo que respecta a subvenciones a fondo perdido. También servirá como base para realizar estudios a posteriori sobre los productos más necesitados por los clientes.

- MOTIVACIÓN Y DIRECTRICES DEL PROYECTO

- Motivación del proyecto

La realización del presente proyecto ha sido debida a varias razones:

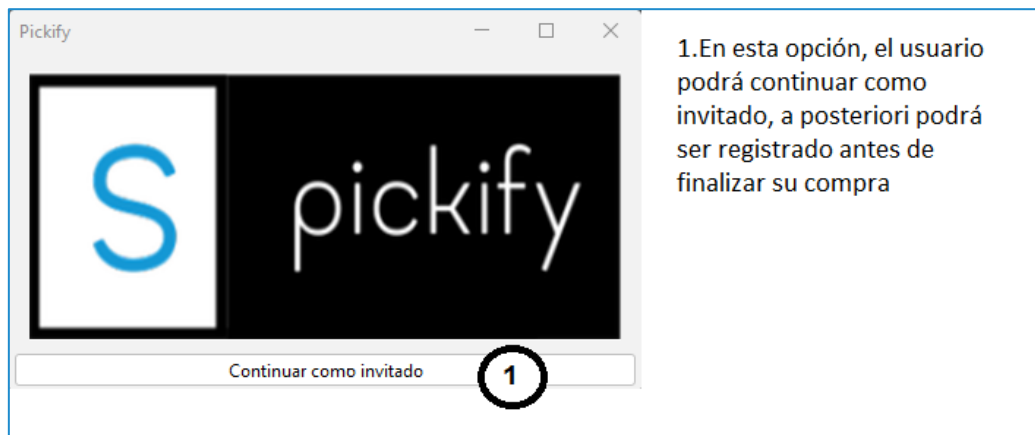
- Comercialización de los productos de forma más rápida.

- Expansión reforzada de la ubicación de las tiendas a partir de la selección de estas. De esta forma se podrían presentar productos existentes de varias tiendas en una sola aplicación. Además, se ha demostrado que los consumidores tienden a buscar aplicaciones de diferentes tiendas de productos, por lo tanto, esto se podría reducir a una sola aplicación que muestre diferentes tiendas con sus respectivos productos, para así evitar la descarga de varias aplicaciones, y tener al alcance todo tipo de productos, de forma más sencilla.

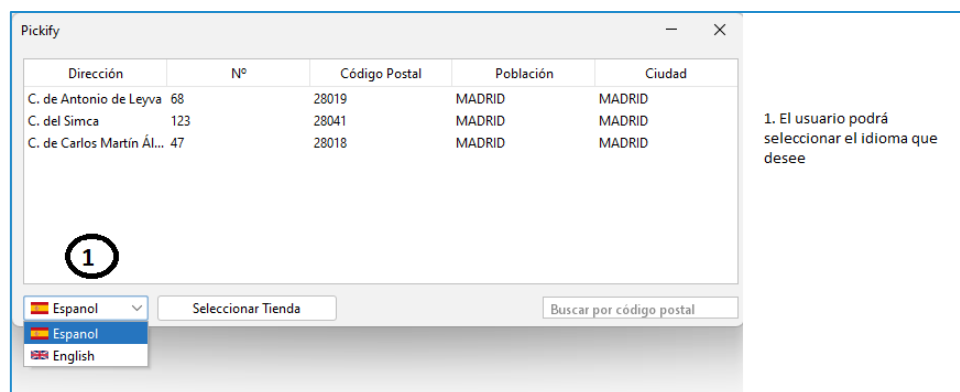
- Directrices del proyecto. La realización del proyecto obedece a la necesidad de conseguir modernizar el sistema de compra de productos, adaptándolo a las exigencias de los consumidores y obteniendo una calidad elevada de selección de estos. Consistiría, por lo tanto, en transformar la situación actual o situación problema, en la que existe un alto grado de atomización en la compra de productos, una exigencia de descargar aplicaciones, y que no se adapta a las exigencias y gustos de los consumidores.

Funcionalidad de la aplicación

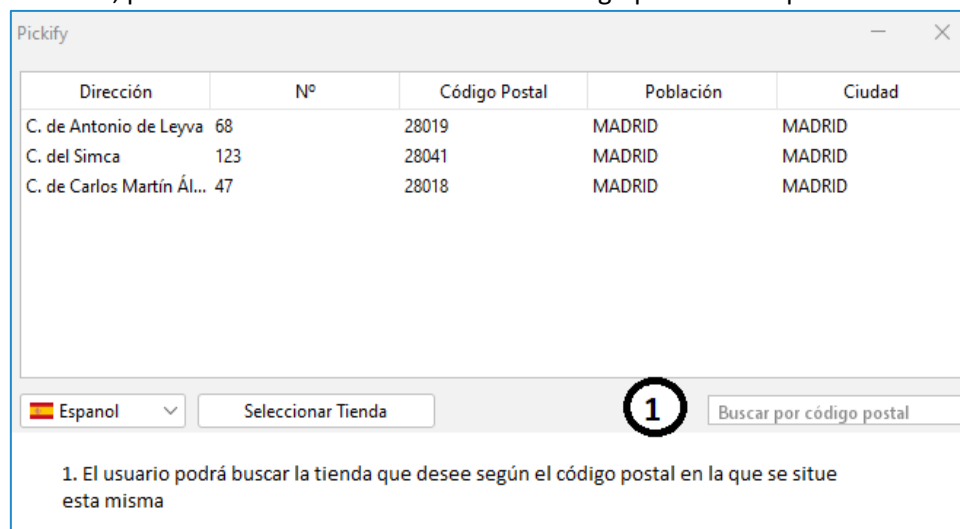
Para comenzar, el usuario debe iniciar como invitado. Previamente se podrá registrar.



El usuario podrá traducir la aplicación en el idioma que este desee.



A continuación, podrá buscar la tienda mediante el código postal en la que se encuentre esta.




Cuando el usuario haya encontrado la tienda que desee, podrá seleccionar esta misma para dirigirse a la interfaz principal de esta.

1

2

Dirección	Nº	Código Postal	Población	Ciudad
C. de Antonio de Leyva	68	28019	MADRID	MADRID
C. del Simca	123	28041	MADRID	MADRID
C. de Carlos Martín Ál...	47	28018	MADRID	MADRID

 Español

Seleccionar Tienda

Buscar por código postal

1.El usuario podrá seleccionar la tienda que más se ajuste a sus necesidades

2.A posteriori, el usuario podrá dirigirse a ella

Una vez en la interfaz de la tienda, el usuario podrá adquirir una cuenta previamente, o iniciar sesión con esta, para guardar el progreso de su compra y poder realizar esta.

1

Categorías


Ropa

Comida




Libros

Buscar productos ...

Cuenta



Últimos productos

	Nombre	Precio	
	Mandarinas	1.0€	Ver
	Leggins	7.5€	Ver
	El Quijote	12.0€	Ver

Atrás

1.El usuario podrá guardar su progreso e avanzar dicha compra, mediante la otorgación de una cuenta

El usuario podrá registrarse en la aplicación, o iniciar previamente en ella.

The image shows two overlapping windows from the 'Pickify' application. The background window is the login form, featuring fields for 'Usuario' (Username) and 'Contraseña' (Password), along with 'Registrarse' (Register) and 'Iniciar sesión' (Login) buttons. The foreground window is the registration form, which includes fields for 'Nombre' (Name), 'Apellidos' (Surnames), 'Fecha Nacimie...' (Date of Birth), 'E-mail', 'Tlf' (Phone), 'Dirección' (Address), 'Ciudad' (City), and 'Código Postal' (Postal Code). It also has an 'Enviar' (Send) button at the bottom.


Los productos están organizados por categorías, por lo que el usuario podrá dirigirse a la sección que más le convenga en cierto momento.

The image displays the 'Pickify' product catalog interface. On the left, a sidebar lists categories: 'Ropa' (Clothing), 'Comida' (Food), and 'Libros' (Books), with a circled '1' next to the 'Libros' category. The main area features a search bar, a 'Cuenta' button, and a shopping cart icon. Below these is a table titled 'Últimos productos' (Latest products) with columns for 'Nombre' (Name), 'Precio' (Price), and 'Ver' (View). The table lists three items: 'Mandarinas' (1.0€), 'Leggins' (7.5€), and 'El Quijote' (12.0€). A circled '1' is placed over the 'Ver' button for 'Mandarinas'. To the right of the interface, a text box states: '1.El usuario podrá seleccionar la categoría que requiere para visualizar sus productos'.

El usuario podrá ver información del producto y añadir el producto al carro de compra.

The image shows two overlapping windows from the 'Pickify' application. The background window is the product catalog, with the 'Mandarinas' product highlighted and its 'Ver' button circled with a '1'. The foreground window is the product detail view for 'Mandarinas', showing an image, a description field, and manufacturer/weight information. It includes an 'Añadir al carrito' (Add to cart) button, which is circled with a '2'. Below the windows, two numbered notes are provided: '1.El usuario podrá ver información del producto' and '2.El usuario podrá añadir el producto al carrito'.

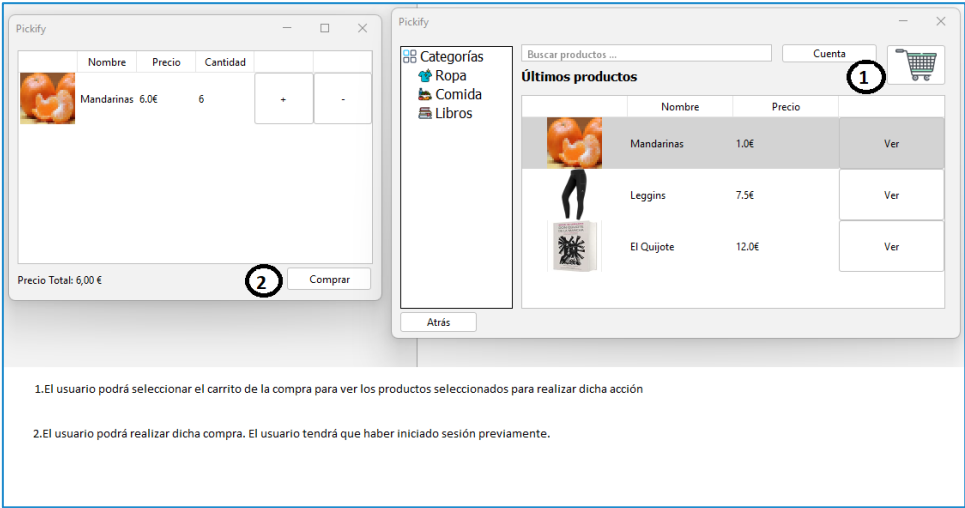
El usuario podrá seleccionar la cantidad que requiere del producto. Antes de finalizar su compra, el usuario posteriormente podrá cambiar la cantidad que requiere por cada producto.



A screenshot of a 'Pickify' window for selecting a quantity. It features a title bar with the name 'Pickify' and standard window controls. The main content area has the label 'Cantidad' followed by a text input field containing '1', a separator '/ 100', and two buttons labeled '+' and '-'. Below this is a large button labeled 'Añadir'.

El usuario posteriormente podrá dirigirse a la cesta de la compra y ver los productos que este ha seleccionado para realizar dicha compra. Este mismo podrá seleccionar la cantidad necesaria que requiere por cada producto, y la cantidad total a pagar.

El usuario deberá iniciar sesión primero, para ejecutar dicha acción.



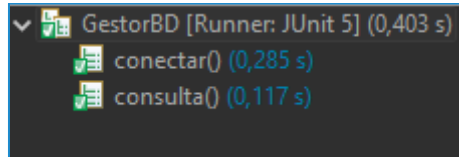
A screenshot of the Pickify application showing two overlapping windows. The background window displays a list of categories (Ropa, Comida, Libros) and a table of 'Últimos productos' (Latest products) with columns for Nombre, Precio, and a 'Ver' button. The foreground window shows a shopping cart with a table containing one item: Mandarinas at 6.00€ with a quantity of 6. It includes a 'Comprar' button and a total price of 6,00 €. Numbered callouts 1 and 2 are present. Below the windows, two numbered instructions are provided.

1.El usuario podrá seleccionar el carrito de la compra para ver los productos seleccionados para realizar dicha acción

2.El usuario podrá realizar dicha compra. El usuario tendrá que haber iniciado sesión previamente.

Pruebas de funcionalidad

La interfaz requiere de una conexión a una base de datos, por lo que se realizan pruebas de JUnit en un entorno de desarrollo, para comprobar que esta conecta correctamente, e y que en ella se pueden realizar dichas consultas a la base de datos.



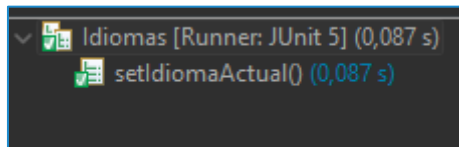
Mediante el método conectar, se comprueba si la interfaz conecta de forma correcta a la base de datos. El método conectar devuelve true, si esta conecta de forma correcta.

```
void conectar() {  
    if (!app.GestorBD.conectar("localhost", 1521, "proyecto", "proyecto"))  
        fail("Error al conectar la base de datos");  
}
```

Mediante una consulta se seleccionan todos los nombres de las tablas, almacenadas en un array, para comprobar que hace una consulta a la base de datos de forma correcta.

```
@Test  
void consulta() throws SQLException {  
    ResultSet rs = app.GestorBD.consulta("SELECT table_name FROM user_tables");  
  
    String[] tablas = {"HISTORIAL_COMPRAS",  
        "USUARIOS",  
        "DATOS_USUARIOS",  
        "TIENDAS",  
        "PRODUCTOS",  
        "PRODUCTOS_TIENDA",  
        "PRODUCTO_ROPA",  
        "PRODUCTO_COMIDA",  
        "PRODUCTO_LIBROS",  
        "EDITORIALES",  
        "MARCAS",  
        "AUTORES",  
        "FABRICANTES"};  
    List<String> listaTablas = Arrays.asList(tablas);  
  
    while(rs.next()) {  
        if (!listaTablas.contains(rs.getString(1))) {  
            fail("Faltan tablas en la base de datos");  
        }  
    }  
}
```

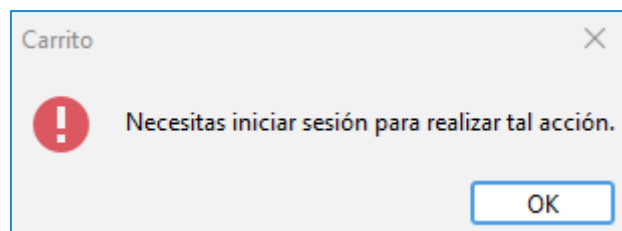
La interfaz puede traducirse en varios idiomas, por lo que se comprueba que el método cambiar idioma funciona correctamente.



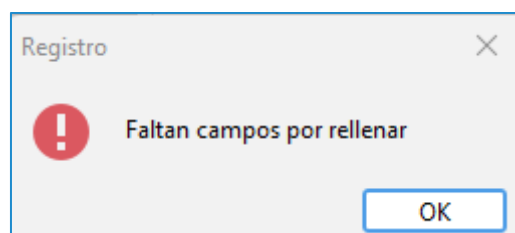
Para ello, se crea un método que carga un archivo con las traducciones equivalentes en ciertos idiomas. Si el archivo no carga o no se ejecuta correctamente, nos devolverá un booleano con valor negativo 'false'.

```
@Test
void setIdiomaActual() {
    app.Idiomas.setIdiomaActual(app.Idiomas.IDIOMA_DEFECTO);
    app.Idiomas.init();
    System.out.println(app.Idiomas.getIdiomaActual());
    if (!app.Idiomas.getIdiomaActual().equals(app.Idiomas.IDIOMA_DEFECTO)) {
        fail("Error en la carga/guardado de idiomas");
    }
}
```

Si el usuario no ha iniciado sesión, se muestra una advertencia de error y este no puede realizar dichas compras.



Si el usuario no ha rellenado los campos de registro correctamente, se muestra una advertencia de error y este no puede finalizar su registro.



Coordinación de grupo de trabajo

Para la coordinación del grupo, se ha creado una rama principal, que contendrá el proyecto en sí. El proyecto ha sido elaborado en la rama principal. Salvo dos ramas, que han sido elaboradas por partes:

La rama 'p-carritocompra' en la que se ha trabajado el carrito de la compra de dicha interfaz.

La rama 'p-horasesion' en la que se ha trabajado en la hora de sesión en la que el usuario inicia sesión en la interfaz.

Default branch					
main	Updated 9 days ago by Lyuu			Default	
Active branches					
p-carritocompra	Updated 2 months ago by Lyuu	24	0	#2	Merged
p-horasesion	Updated 2 months ago by Lyuu	27	0	#1	Merged

A posteriori se han sumergido a la rama principal:

0 Open ✓ 2 Closed		Author ▼	Label ▼	Projects ▼	Milestones ▼	Reviews ▼	Assignee ▼	Sort ▼
implementar carrito de compra		#2 by Lyuu17 was merged on 24 Mar						
implementar hora al iniciar sesion		#1 by Lyuu17 was merged on 24 Mar						

Bibliografía

- Combo Box Icons: <https://www.logicbig.com/tutorials/java-swing/combo-box-icons.html>
- Button Column: <https://tips4java.wordpress.com/2009/07/12/table-button-column/>
- Imagen Icon Resize: <https://stackoverflow.com/a/18335435>
- Place Holder Formatted: <https://stackoverflow.com/a/16229082>