

Documentación: Aplicación de cestas

Descripción del proyecto	1
Cómo instalar e iniciar el proyecto	2
Requisitos	6
Cobertura de pruebas	10

Descripción del proyecto

Para dar solución a los requisitos propuestos, he estimado la necesidad de crear una aplicación que empleara JAVA, como requerido, y SQL.

El proyecto JAVA tiene la tarea de ejecutar la lógica necesaria para la aplicación, y además servir de interfaz de **comunicación con el usuario mediante la consola del IDE**. Esta parte es la que se encarga del control de los flujos que hacen posible el cumplimiento de cada requisito, el control de opciones introducidas por el usuario para evitar errores, la comunicación con la base de datos y la realización de las consultas, y la comunicación con el usuario en el idioma elegido entre las opciones de inglés o español.

El SQL, tiene la misión de controlar la persistencia a los datos con los que va a trabajar la aplicación. Creando la base de datos y las tablas que contendrán la información, evitando la duplicidad de ids, y eliminando los registros que a criterio de la aplicación ya no sean necesario mantener.

La base de datos se compone de tres tablas: CARTS, que almacena las diferentes cestas creadas y su fecha de modificación; PRODUCTS, que almacena nombres, descripciones y precios de los diferentes productos disponibles; y CART_PRODUCTS, que almacena los productos y su cantidad, asignando un id de producto y cantidad a un id de cesta. También incluye una secuencia, CART_ID_SEQ, para ir asignando ids a los nuevos carros creados, y restricción en cascada para eliminar los registros de CART_PRODUCTS asignados a un id de cesta, cuando esta cesta sea borrada.

Para su realización me he valido de las herramientas de eclipse y HeidiSQL.

Cómo instalar e iniciar el proyecto

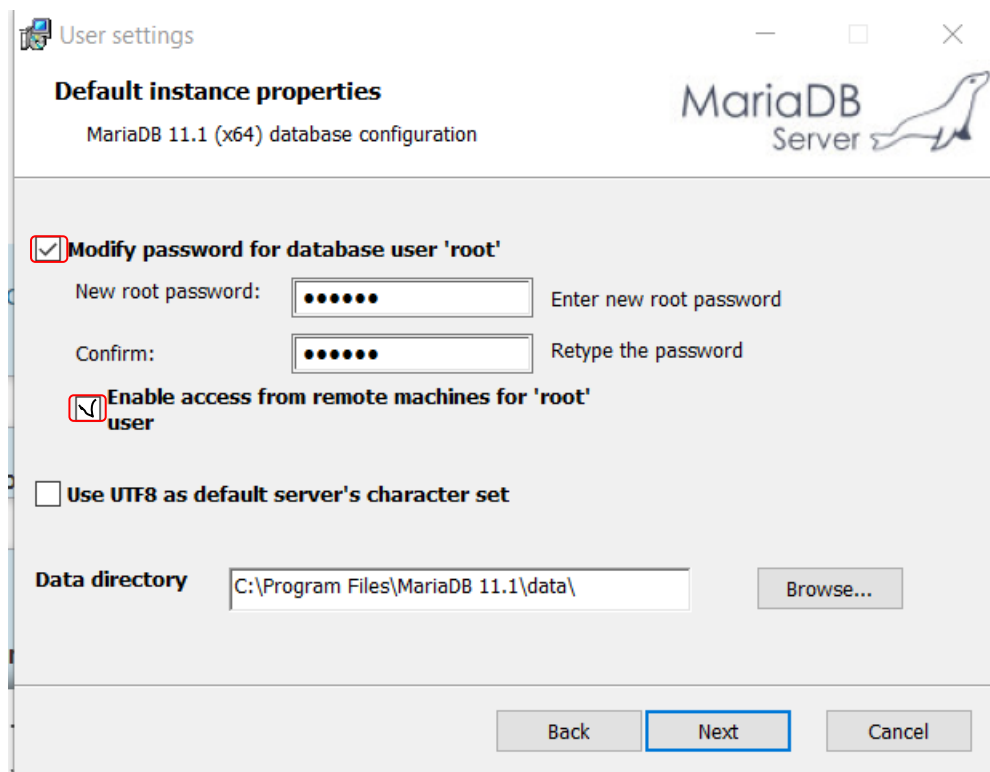
Para hacer funcionar la aplicación hay que preparar la base de datos y el proyecto de java en el IDE.

Base de datos:

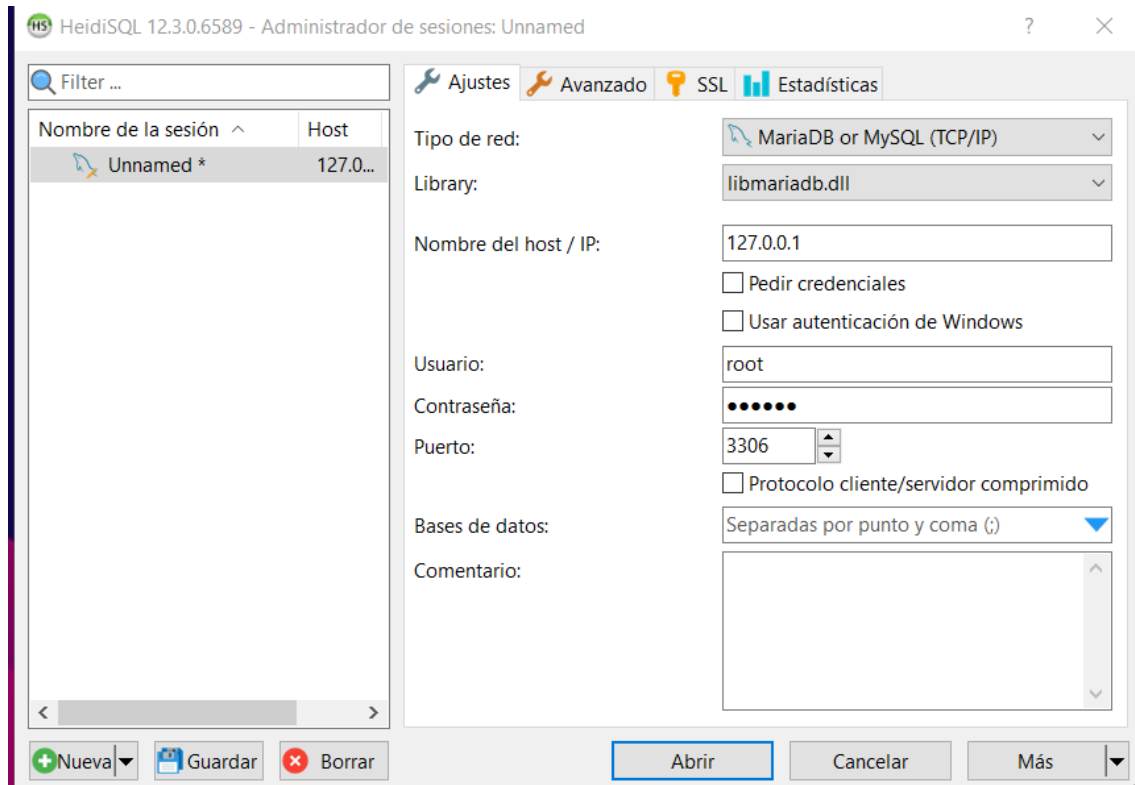
Para la base de datos he usado MariaDB server (versión 11.1.2) y HeidiSQL, configuración que recomiendo usar para hacer funcionar la aplicación de forma más rápida y sencilla, evitando tener que tocar las dependencias y configuración del proyecto. Puede descargarse de este enlace:

https://mariadb.org/download/?t=mariadb&p=mariadb&r=11.1.2&os=windows&cpu=x86_64&pkg=msi&m=fe_up_pt

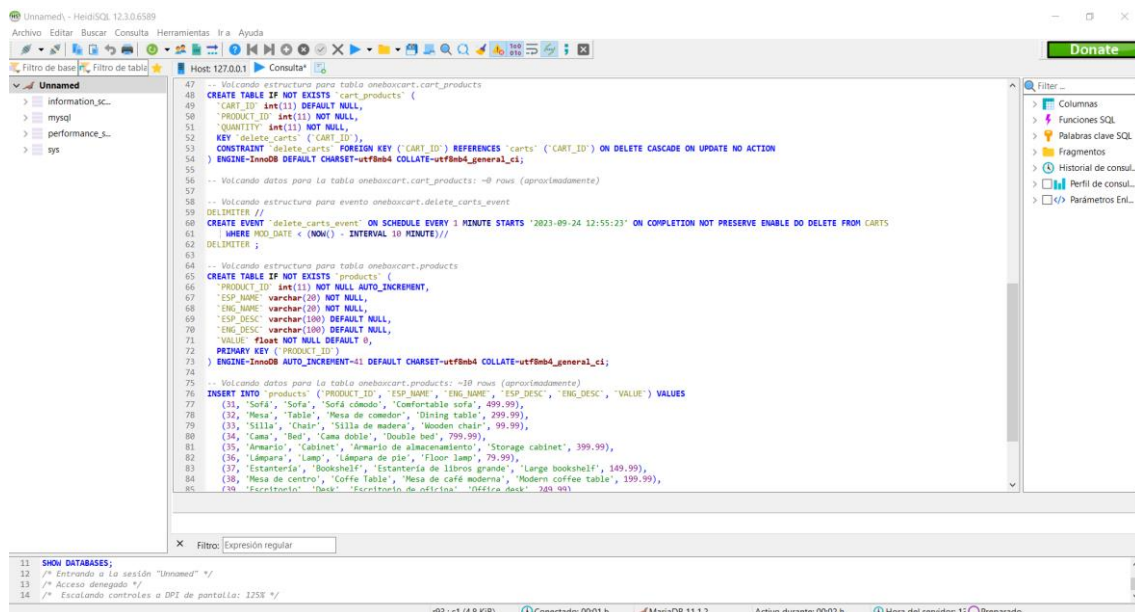
Ejecutamos el instalador y simplemente continuamos hasta la pantalla de 'Default instance properties', habilitamos el acceso en remoto, y configuramos el usuario por defecto 'root' con la contraseña 'ff5544' para que la aplicación funcione con la configuración que ya tiene.



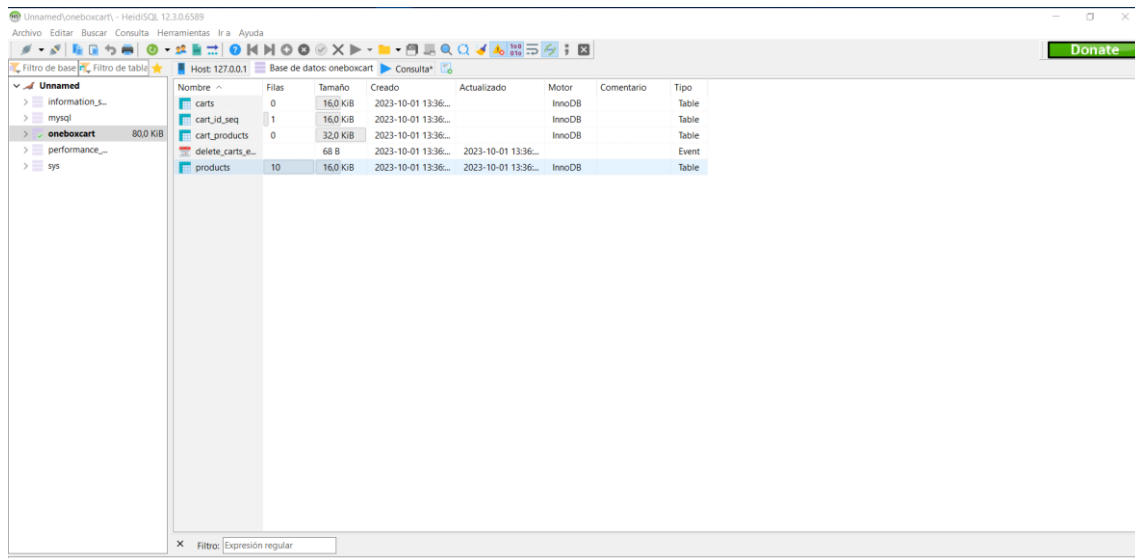
Una vez terminado el proceso de instalación. Abrimos la aplicación HeidiSQL, y creamos una nueva sesión, botón de abajo a la izquierda “Nueva”, introducimos la contraseña de usuario ‘ff5544’, y abrimos la sesión en el botón central de abajo.



Para crear la base de datos, simplemente hay que ejecutar el SQL del archivo *oneboxcart.sql* adjunto, se puede arrastrar el archivo a la pantalla de consulta para mayor comodidad. Nos colocamos sobre la pestaña de consulta, pegamos la consulta, y la ejecutamos pulsando F9 o en el icono azul de play.

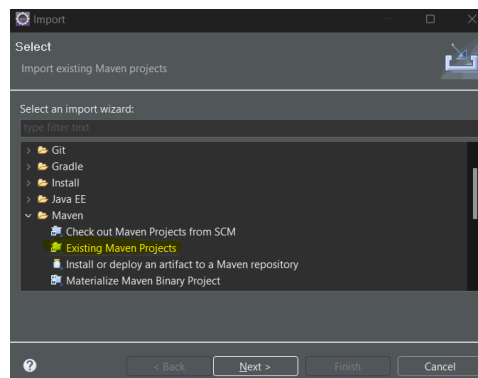


Por último, actualizamos pulsando F5 o en el icono de un círculo verde con una flecha en su interior, para que aparezca la nueva base de datos y su contenido.

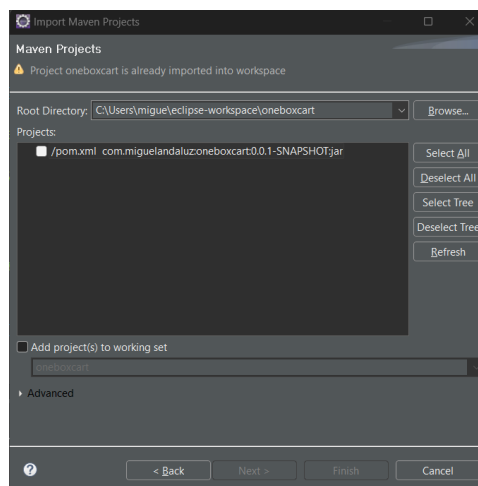


Proyecto JAVA:

Para el desarrollo en JAVA he empleado Eclipse IDE. Para hacer funcionar el proyecto, hay que descomprimir el ZIP en la carpeta de workspace que estemos utilizando, y luego importarlo como proyecto Maven existente.

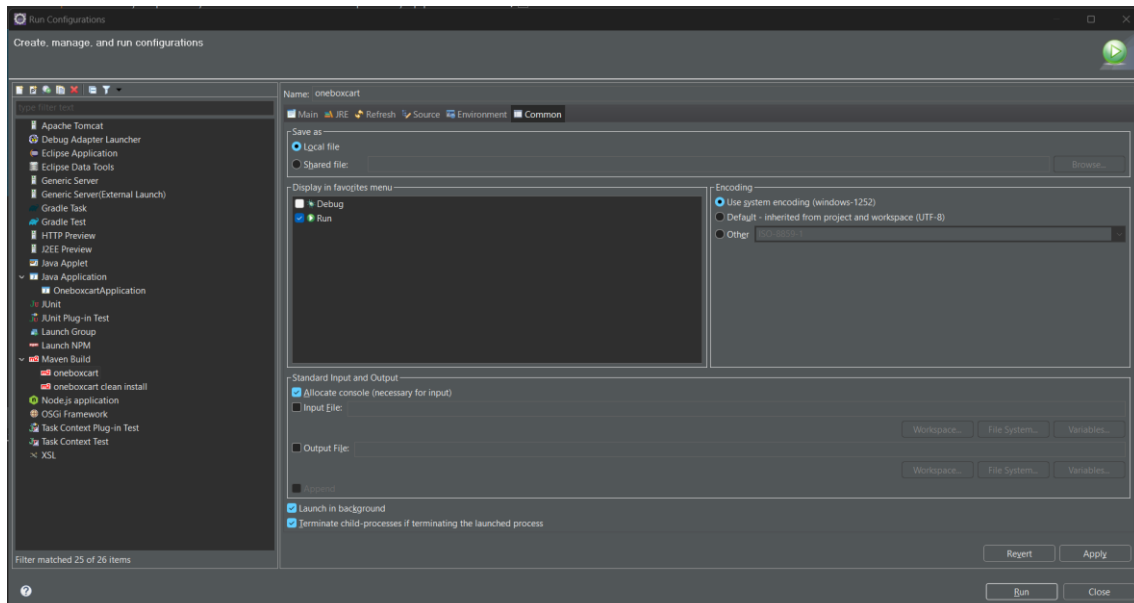
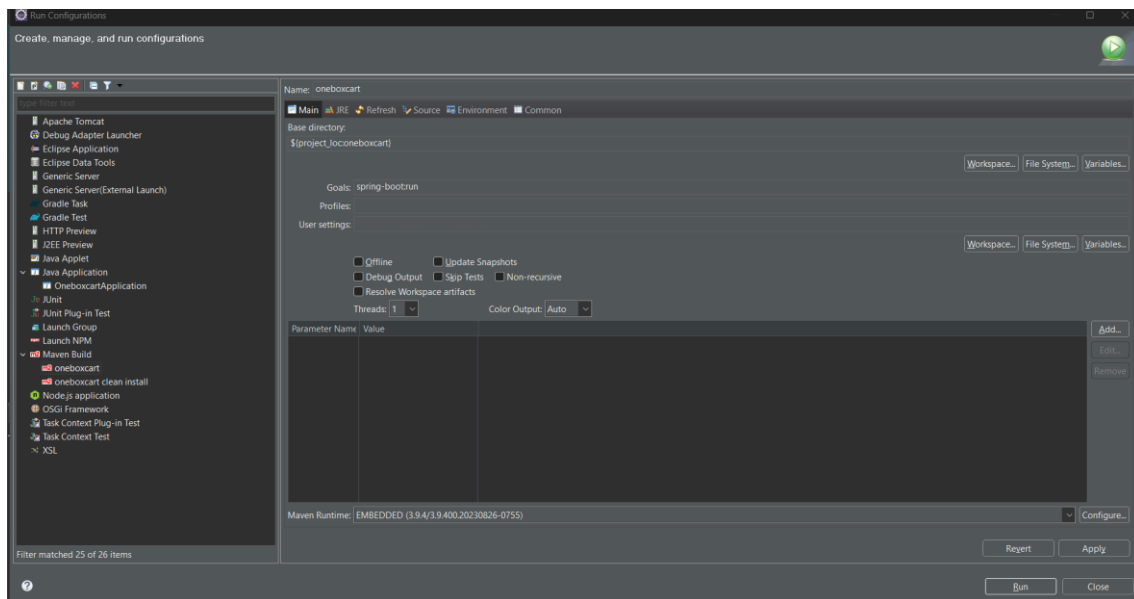


Buscamos el proyecto en la carpeta y lo seleccionamos.



Para iniciar la aplicación hay dos opciones:

- Ejecutarlo como aplicación java: buscando la clase *OneboxcartApplication*, y seleccionando *Run as -> Java application* en Eclipse la primera vez, el resto de veces se podrá seleccionar directamente la opción *Oneboxcartapplication*.
- La segunda opción es construyéndolo como proyecto Maven, seleccionando el proyecto, *Run as -> Maven build...* y en la pantalla emergente de Run Configurations, estableciendo el *goal: spring-boot:run* y en la pestaña *Common*, eligiendo el *encoding: use system encoding...* para evitar problemas de visualización con los caracteres especiales del español, y se pulsa *Run*. Después de la primera configuración, quedará la configuración guardada con el nombre de la configuración.



Una vez finalizada la inicialización, debemos elegir idioma, introduciendo por consola 'es' para español, o 'en' para inglés.

```
Choose language / Elige idioma
Type 'en' for English / Escribe 'es' para español
es
```

Requisitos

ID de cesta generado por la aplicación:

Para generar un id para la cesta, he creado una secuencia en SQL, *'cart_id_seq'*, que cada vez que se consulta nos da un valor. Este valor se obtiene sumándole uno al valor previo cacheado, y que se insertará en la tabla como id de la cesta, CART_ID de la tabla CARTS.

```
Nueva cesta
2023-10-01T20:52:54.827+02:00  WARN 28084 --- [ restartedMa
2023-10-01T20:52:54.872+02:00  WARN 28084 --- [ restartedMa
Cesta creada con id: 1005
¿Qué te gustaría hacer?
Introduce el valor numérico indicado para las opciones:
Añadir productos: 1
Volver al menú principal: 2
```

Borrado de cestas a los 10 minutos de inactividad:

Para el borrado, he creado un evento en SQL, *'delete_carts_event'*, que se ejecuta cada minuto, y comprueba que no hayan pasado más de 10 minutos desde la hora de modificación de la cesta, MOD_DATE de la tabla CARTS, y la hora actual.

Con una restricción al borrado en cascada por el id de la cesta, se eliminan todos los productos asociados a ese id de la tabla CART_PRODUCTS.

Se considera inactividad, que no se haya consultado esa cesta por id, o que no se hayan añadido productos.

Para su correcto funcionamiento, es necesario que la base de datos tenga activados los eventos, lo que se consigue ejecutando la siguiente *query*, que ya viene aportada en el archivo *oneboxcarty.sql*, pero quizá en caso de no funcionamiento sea necesaria volver a ejecutarla.

```
SET GLOBAL event_scheduler = ON;
```

Creación de cesta con capacidad para productos

Para crear una cesta debemos introducir 1 en la consola, y pulsar intro.

```
Bienvenid@
¿Qué te gustaría hacer?
Introduce el valor numérico indicado para las opciones:
Crear nueva cesta de compra: 1
Continuar con cesta ya existente: 2
Eliminar cesta: 3
1
```

La aplicación consultará la secuencia 'cart_id_seq' para obtener el id de la cesta, y realizará un insert a la tabla CARTS creando la cesta. Una vez creado el carro se nos permite añadir productos o volver al menú principal.

```
Nueva cesta
2023-10-01T20:52:54.827+02:00 WARN 28084 --- [ restartedMa
2023-10-01T20:52:54.872+02:00 WARN 28084 --- [ restartedMa
Cesta creada con id: 1005
¿Qué te gustaría hacer?
Introduce el valor numérico indicado para las opciones:
Añadir productos: 1
Volver al menú principal: 2
```

Obtención de información de cesta por ID

Para obtener información de una cesta debemos introducir 2 en la consola, y pulsar intro.

```
Bienvenid@
¿Qué te gustaría hacer?
Introduce el valor numérico indicado para las opciones:
Crear nueva cesta de compra: 1
Continuar con cesta ya existente: 2
Eliminar cesta: 3
2
```

La aplicación realizará una consulta en la base de datos para mostrar que cestas hay actualmente disponibles. Para consultar la información introducimos el id de la cesta a consultar en la consola y se pulsa intro. Con esto la aplicación nos muestra un listado de los productos en el carro, con su nombre, descripción, cantidad, y el coste total de esa línea de producto.

```
Gestor de cestas
Cestas activas:
Id: 1005 - fecha de modificación: 2023-10-01 20:52:54.0
Inserte el id de la cesta:
1005
```

```
Lámpara - Lámpara de pie - 2 - 159.98 €
Estantería - Estantería de libros grande - 2 - 299.98 €
2023-10-01T21:00:19.788+02:00 WARN 25732 --- [ restartedMai
Introduce el valor numérico indicado para las opciones:
Añadir productos: 1
Volver al menú principal: 2
```

Una vez consultado, la aplicación nos permitirá añadir productos a ese carro o volver al menú principal.

Adición de productos a la cesta

Para llegar a la adición de productos hemos debido seguir el curso de los requisitos anteriores, opciones 1 o 2, y haber elegido la opción de añadir productos.

```
Añadir prodcutos: 1
Volver al menú principal: 2
1
```

La aplicación nos mostrará un listado de los productos disponibles mediante a una consulta a base de datos, con su nombre, descripción (en función del idioma elegido), y valor.

```
Sofá - Sofá cómodo - 499.99 €
Sofá opción: 1
Mesa - Mesa de comedor - 299.99 €
Mesa opción: 2
Silla - Silla de madera - 99.99 €
Silla opción: 3
Cama - Cama doble - 799.99 €
Cama opción: 4
Armario - Armario de almacenamiento - 399.99 €
Armario opción: 5
Lámpara - Lámpara de pie - 79.99 €
Lámpara opción: 6
Estantería - Estantería de libros grande - 149.99 €
Estantería opción: 7
Mesa de centro - Mesa de café moderna - 199.99 €
Mesa de centro opción: 8
Escritorio - Escritorio de oficina - 249.99 €
Escritorio opción: 9
Silla de escritorio - Silla de escritorio ergonómica - 149.99 €
Silla de escritorio opción: 10
Inserta el número de opción deseada:
```

Debemos introducir el número de la opción deseada y pulsar intro, y por último elegir la cantidad deseada, y pulsar intro.

```
Inserta el número de opción deseada:
7
Inserta la cantidad deseada:
1
```

Una vez hecho esto, la aplicación comprobará si en nuestra cesta ya se encuentra este producto o no. De encontrarlo, se ejecutará un UPDATE a la tabla CART_PRODUCTS, haciendo un sumatorio entre la cantidad previa y la nueva cantidad. De no encontrarlo se hará un INSERT a dicha tabla asignando el producto y la cantidad al carrito.

Una vez finalizado este proceso podremos seguir añadiendo productos, o podremos volver al menú principal.

```
¿Añadir más productos?  
Introduce el valor numérico indicado para las opciones:  
Añadir: 1 Finalizar: 0  
0
```

Eliminación de cestas

Para la eliminación de una cesta debemos introducir 3 en la consola, y pulsar intro.

```
¿Qué te gustaría hacer?  
Introduce el valor numérico indicado para las opciones:  
Crear nueva cesta de compra: 1  
Continuar con cesta ya existente: 2  
Eliminar cesta: 3  
3
```

Llegados a este punto debemos conocer el id de la cesta que queremos eliminar. La aplicación comprobará que la cesta existe para poder ser eliminada. Una vez insertado el id de la cesta a eliminar, se hará una pregunta de seguridad que nos permitirá elegir entre seleccionar otro id a borrar, opción 1, eliminar la cesta para la que hemos introducido el id, opción 2, o salir al menú principal, opción 3.

```
Borrado de cestas  
Inserte el id de la cesta a borrar:  
1005  
2023-10-01T21:11:07.433+02:00 WARN 3  
Vas a eliminar las cesta con id: 1005  
¿Estás seguro?  
Elegir otra cesta: 1  
Borrar cesta elegida: 2  
Salir al menu: 3  
2  
2023-10-01T21:11:23.527+02:00 WARN 3  
Cesta borrada
```

La opción 1 nos vuelve a dejar introducir un id a borrar. La opción 2 borra la cesta cuyo id hemos introducido, y todos los productos asociados a él gracias a la restricción en cascada.

Cobertura de pruebas

Los flujos y distintas opciones han sido testados varias veces, en ambas opciones de idioma disponibles, comprobando el correcto funcionamiento de todas las funcionalidades y requisitos, y buscando y dando solución a los distintos errores. Para ello no me he limitado a las opciones disponibles a la aplicación, si no que realizado pruebas de lo que podría ser un mal uso por parte de un usuario, para probar la robustez de la aplicación y la solidez de los flujos.

Para demostrar su correcto funcionamiento, apporto imágenes para cada requisito como evidencia del funcionamiento correcto y de las pruebas llevadas a cabo. Exceptuando el borrado automático para el que no puedo aportar prueba gráfica.

Además, estas pruebas las he realizado en dos PC distintos, para comprobar que la configuración para hacer funcionar este proyecto, que he presentado en este documento, es válida.