

**«Álvaro Bajo Tabero»**

**Trabajo Obligatorio Asignatura Java**

**Junio 2023**

**FUNDACIÓN SAN VALERO**

**SEAS, Centro de Formación Abierta**

**ZARAGOZA**

Contenido

[Introducción 2](#_Toc138781591)

[Base de datos 2](#_Toc138781592)

[Observaciones de programación referente a BBDD 3](#_Toc138781593)

[APLICACIÓN 3](#_Toc138781594)

[Loguin 3](#_Toc138781595)

[Panel Principal 4](#_Toc138781596)

[Empleados 4](#_Toc138781597)

[Productos 5](#_Toc138781598)

[Clientes 7](#_Toc138781599)

[TPV 9](#_Toc138781600)

[Sin cliente registrado 10](#_Toc138781601)

[*Con cliente registrado* 13](#_Toc138781602)

[*Notas de TPV* 14](#_Toc138781603)

[Reportes 14](#_Toc138781604)

[Ventas 15](#_Toc138781605)

[Ventas por cliente 16](#_Toc138781606)

[Ventas por fecha 17](#_Toc138781607)

[Cerrar sesión 18](#_Toc138781608)

# Descarga del proyecto

El trabajo completo podrán encontrarlo en el siguiente enlace de GitHub

## [Trabajo en gitHub](https://github.com/alvarowau/Tienda_Bajo_tabero_Alvaro.git)

## Introducción

Una vez visto el planteamiento que se me ofrece para crear una aplicación tipo tpv como por ejemplo en un restaurante de comida rápida, le doy otro enfoque al facilitado por la fundación san Valero.

Con el enfoque que le doy busco que la app sea una app sencilla y amigable con el usuario a la par de que la curva de aprendizaje sea de lo más sencilla posible.

La app que propongo cubre todos los campos que veo importante en un restaurante como este, que la generación en la caja sea lo más rápido posible.

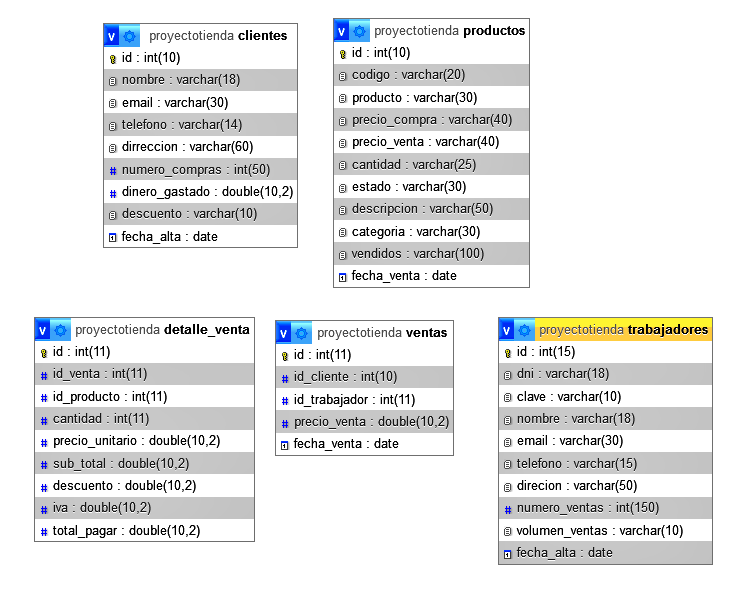
Desde la aplicación también se puede ver el inventario de productos, así como de clientes, trabajadores y las ventas realizadas en rangos de fecha.

Como veremos a lo largo de este documento, la aplicación tiene un ambiete sencillo y funciona mediante logueo, la persona logueada es la que guardara todas las ventas y modificaciones a su nombre.

La aplicación esta creada en su totalidad mediante código Java.

La base de datos que utilizo es MySQL.

## Base de datos



Creo la base de datos con estas tablas y estos atributos, en la gran mayoría utilizo atributos tipo varchar para que no haya problemas de corrupción en la subida o bajada en los mismos, por estos inconvenientes decido crear la gran mayoría tipo varchar.

## Observaciones de programación referente a BBDD

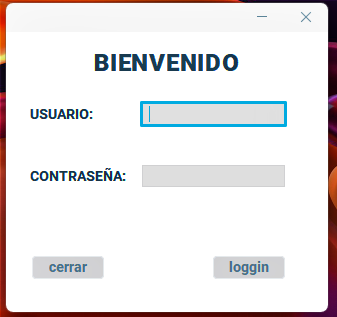
Una vez creada la base de datos hay que fijarse muy bien en que los atributos no inicien en “null”. Ya que durante la programación de los mismos me dio demasiados problemas.

En los que tendrían que ser tipo numérico en la programación he creado métodos, para que el usuario no pueda añadir caracteres en los mismos.

Los problemas relacionados con “null” suelen ser en la lectura de las tablas ya que los omite, además de cuando quieres operar con valores numéricos, en la base de datos he personalizado la entrada de los mismos para que no haya ningún tipo de error en ellos, ya que los inicialice en ‘0’.

# APLICACIÓN

## Loguin

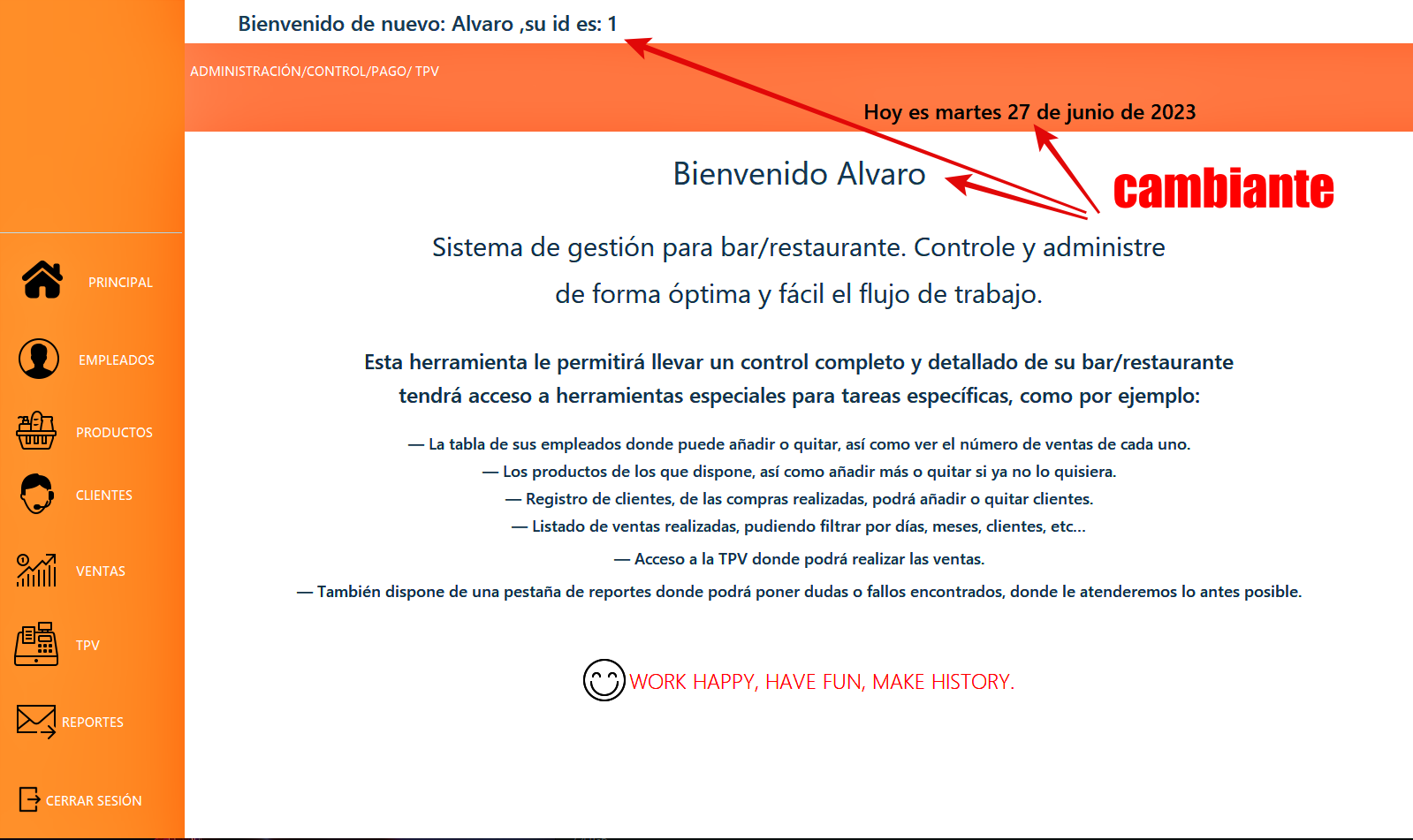


Esta es la primera imagen que nos encontraremos al ejecutar la app, su manejo es bastante simple, cabe destacar que en todo el proyecto he usado un LOOK AND FEEL llamado flat, para que la visualización de la aplicación sea mas adaptada al actual.

En esta ventana lo único que podremos hacer es loguear con nuestro nombre y contraseña, o salir del programa.

En caso de que el logueo sea erróneo nos lanzara un mensaje de error, en caso e que sea exitoso nos dirigirá al panel central de la aplicación.

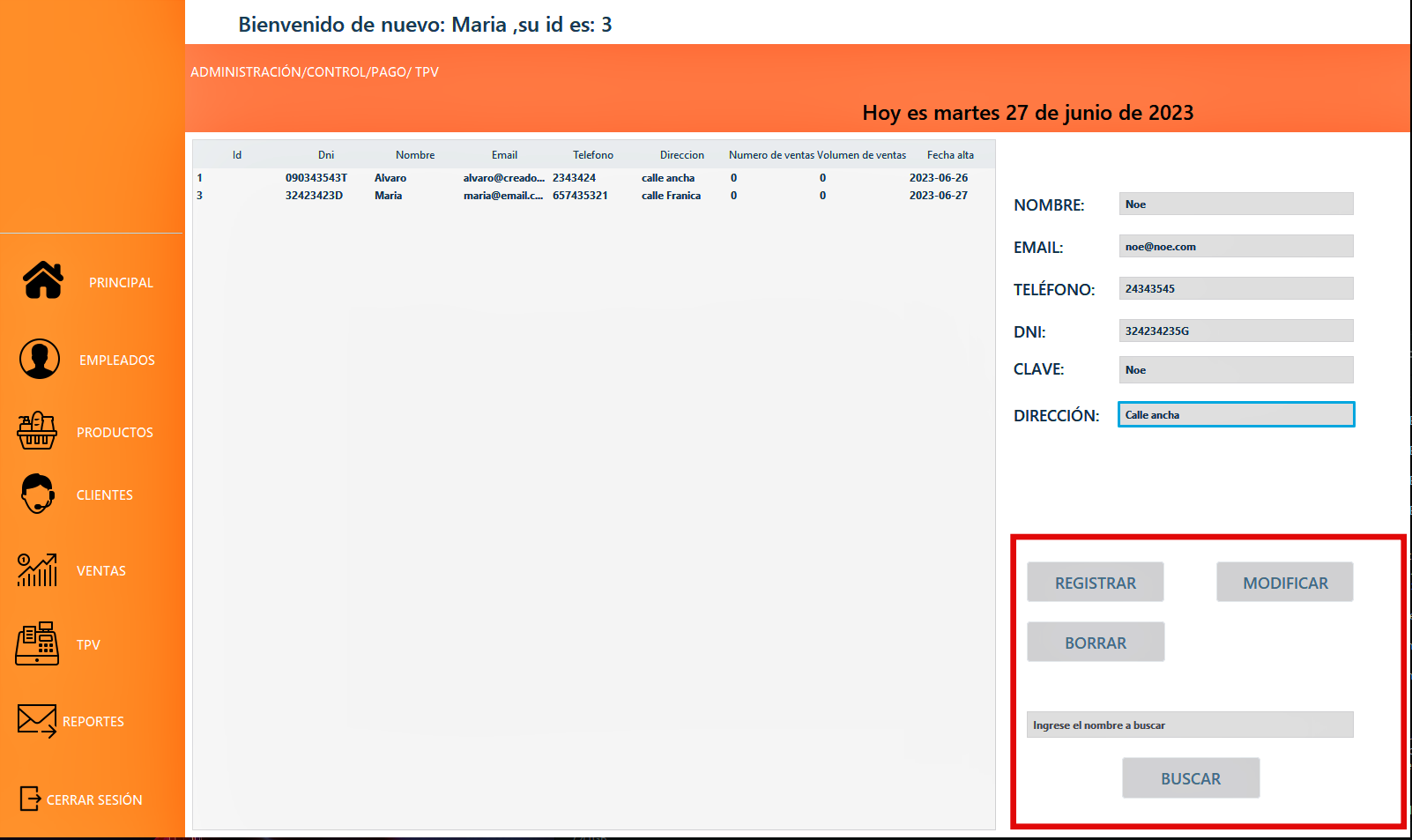
## Panel Principal



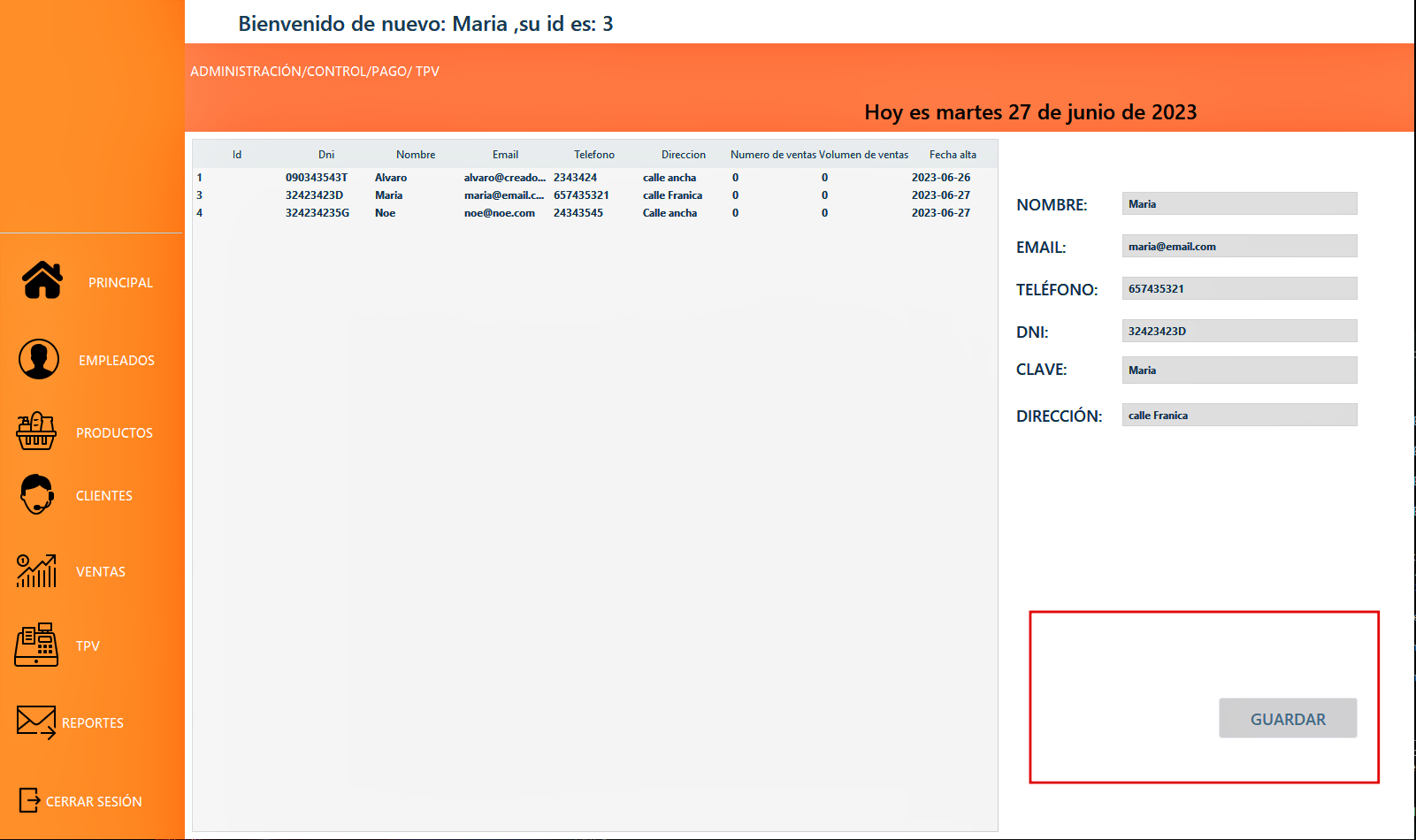
En el panel principal poder ver una leyenda de que hace la aplicación por encima, además e introducido elementos cambiantes. Por ejemplo, según con el usuario que te logues puedes ver tu nombre y tu id, aparte e la fecha, que va cambiando respecto al día que lo abras.

## Empleados

En la pantalla de empleados los botones no son estáticos, así mismo si quieres hacer un registro o una modificación los botones aparecerán y desaparecerán según sean necesarios, las tablas se cargan automáticamente, es decir si haces un registro en cuanto el código verifique que todo es correcto te sacara un JOptionPane diciéndote que el registro o la modificación se efectuó de manera exitosa y se cargara directamente en la tabla, pongo dos imágenes a modo de ejemplo.

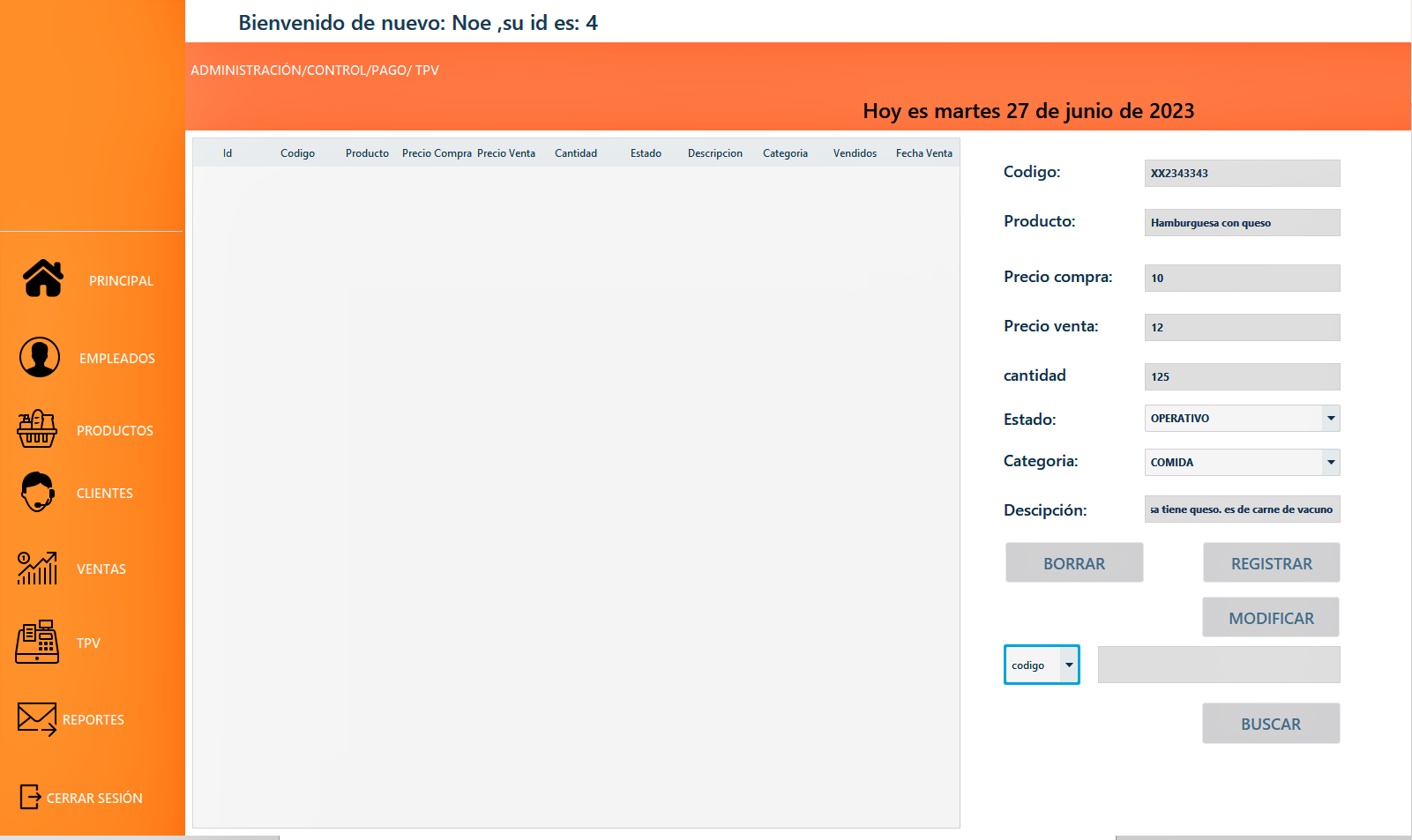


Vemos los botones de la fotografía en el registro aparte de su apartado de búsqueda.

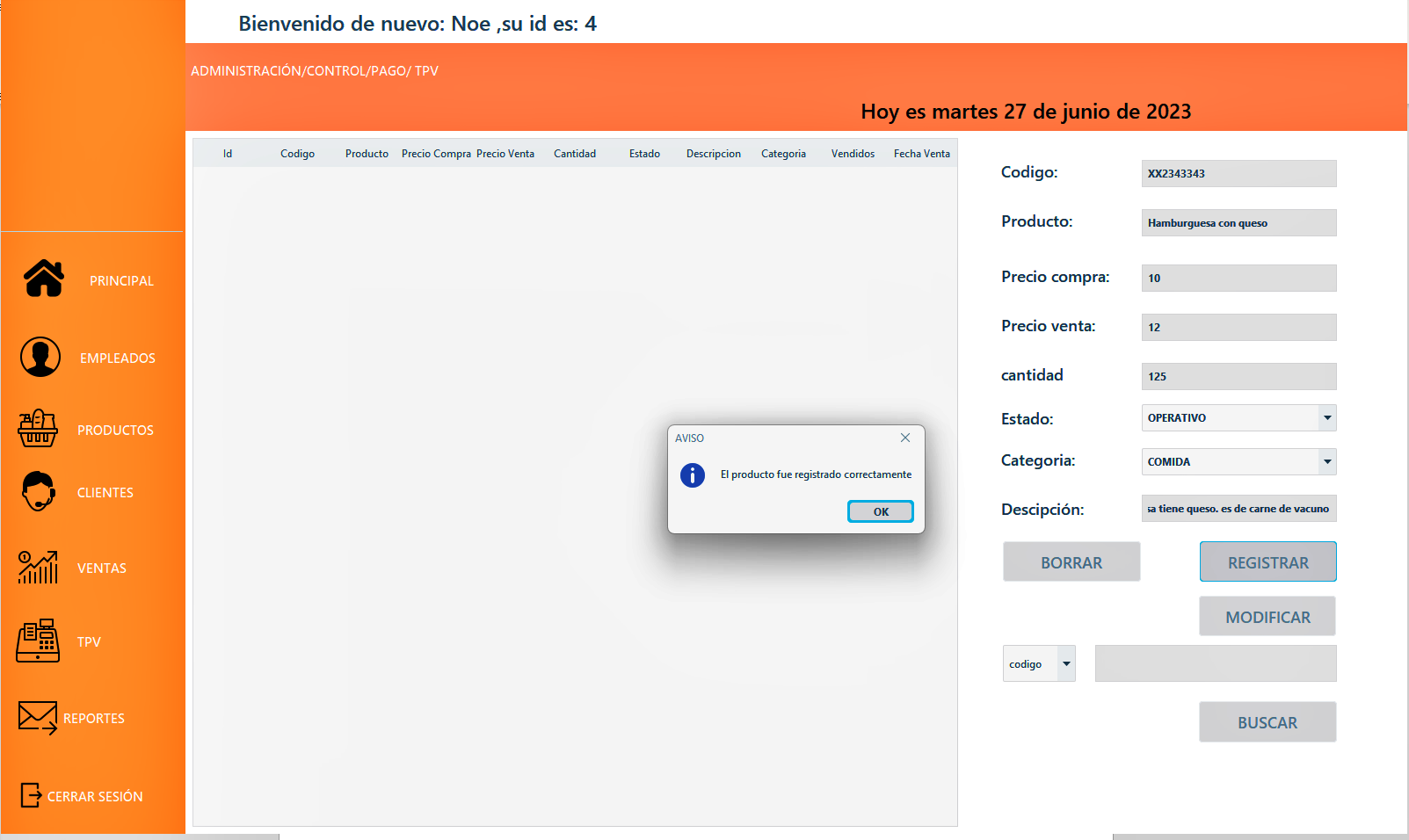
**

En esta imagen podemos ver que la disposición de los botones no es la misma que en la anterior, así mismo los demás botones han desaparecido para que solo se pueda guardar.

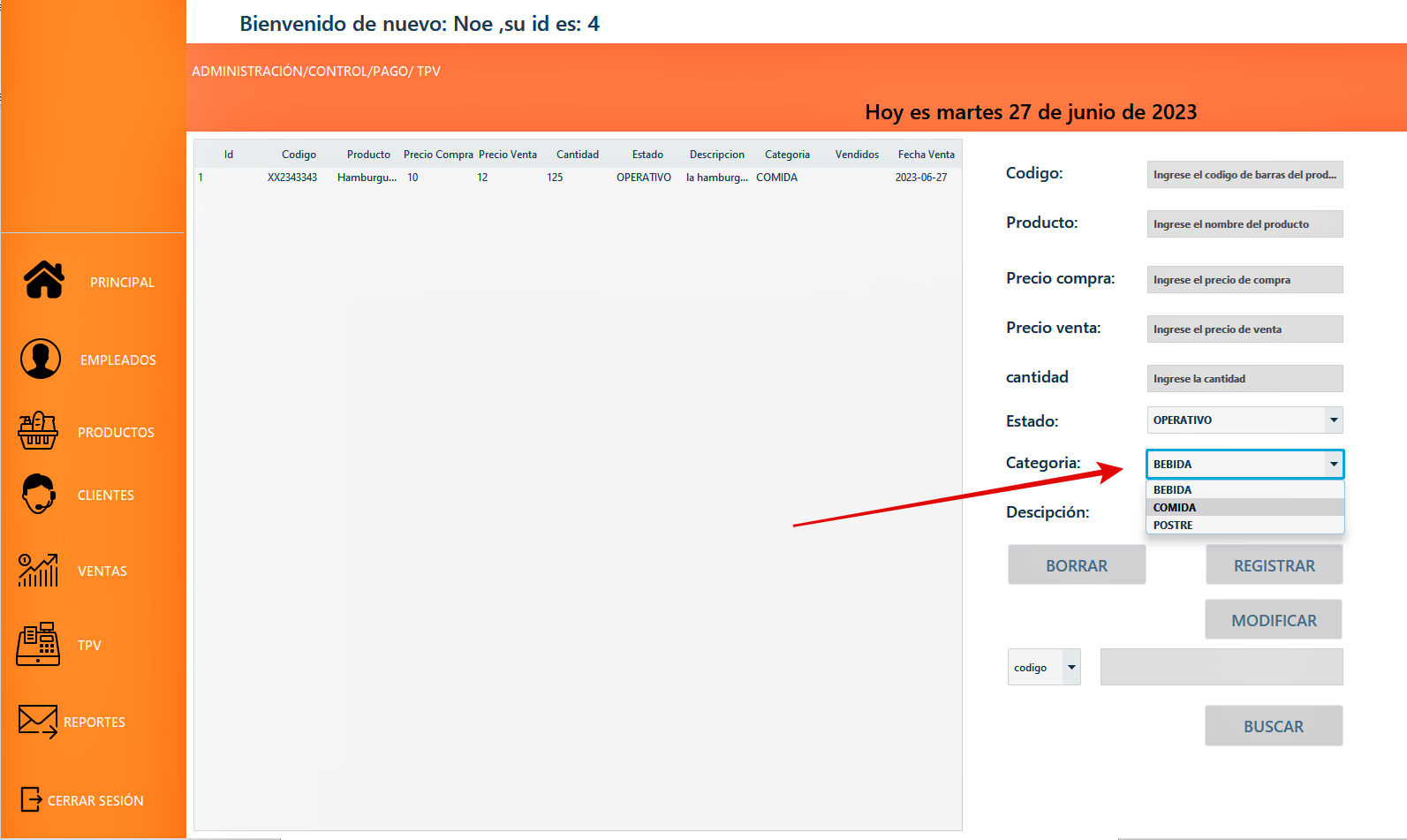
## Productos



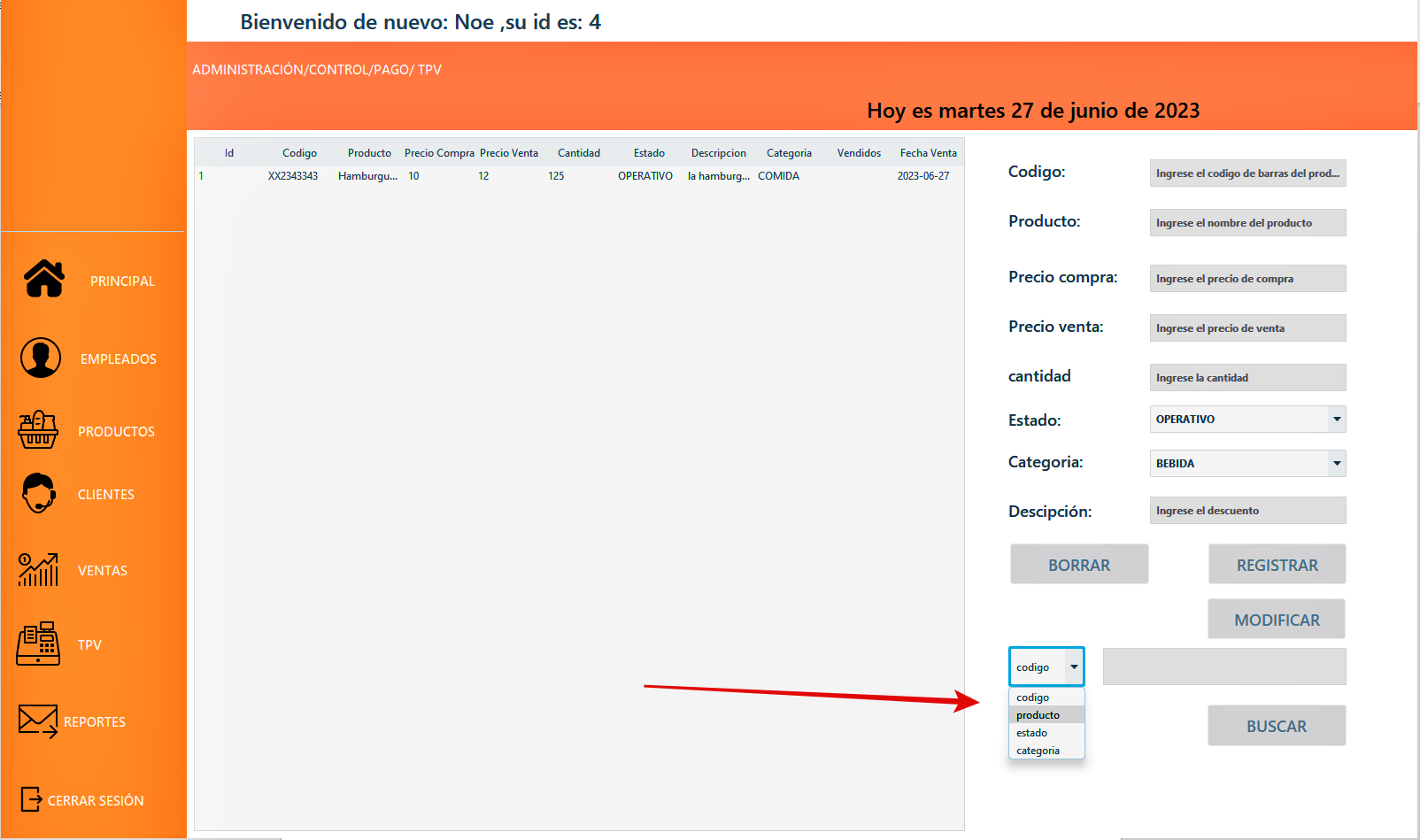
A la hora de registrar un producto podemos ver que nos pie valores estándar para un producto, lo cambiante de esta view es por ejemplo que en el campo ‘Precio compra’ ‘Precio venta’ & ‘Cantidad’, no nos dejara meter valores que no sean numéricos, al igual que en los precios nos dejara meter valores doubles en cantidad solo nos dejara meter valores INT, también decidí poner la categoría en selector para a la hora de usar la TPV no nos de error.



Cuando guardamos un producto de manera satisfactoria nos aparecerá este mensaje y se cargará seguidamente en la tabla, como podemos comprobar en la siguiente imagen.



El selector de la categoría solo nos permite escoger entre esos tres valores y asi quedaran guardados en la base de datos como podremos ver mas adelante en la TPV entenderemos la elección de selector.

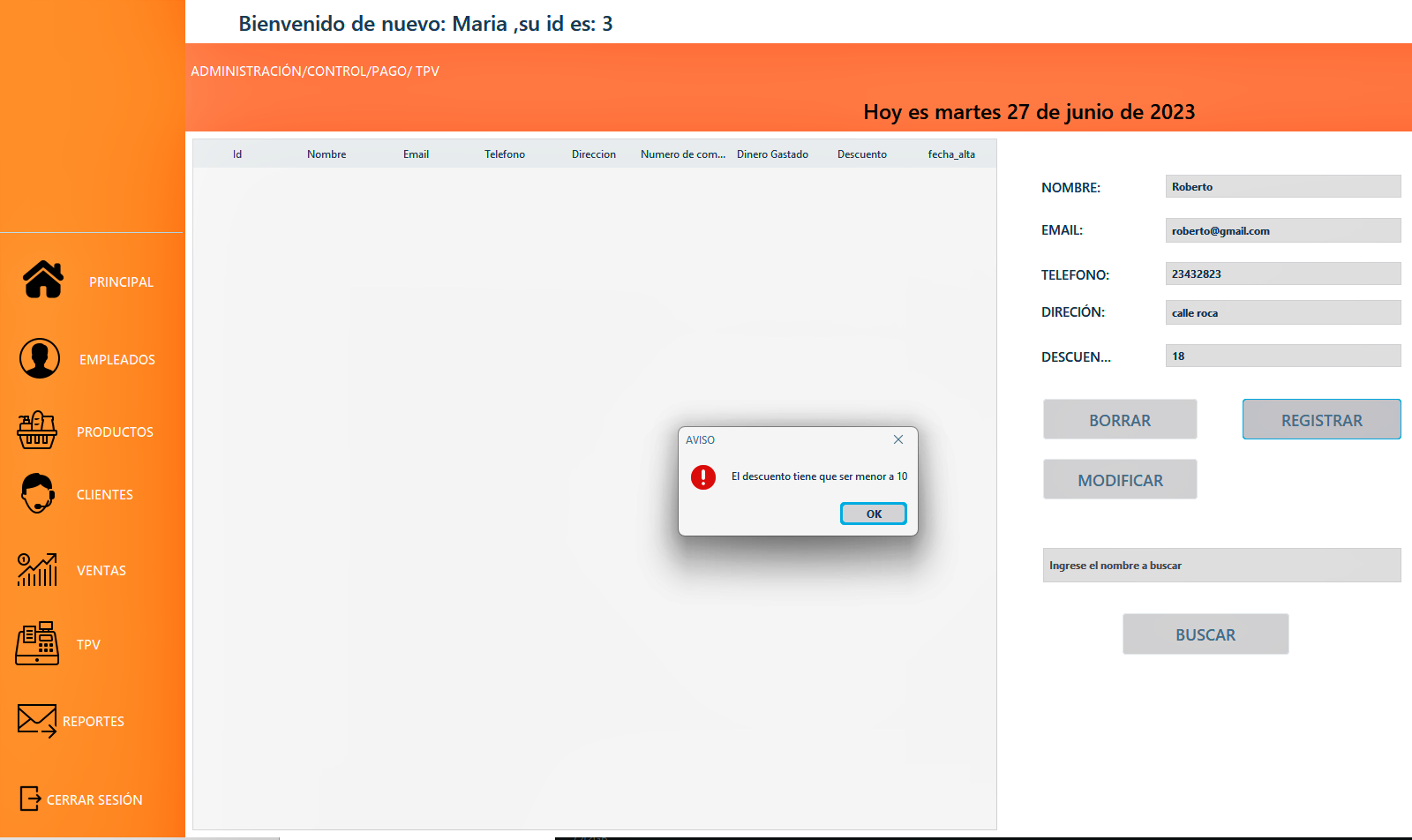


En el cuadro de búsquedas me decante por poner otro selector para que las búsquedas fuesen mas avanzadas, ya que en una empresa no siempre se buscan los productos de la misma manera.

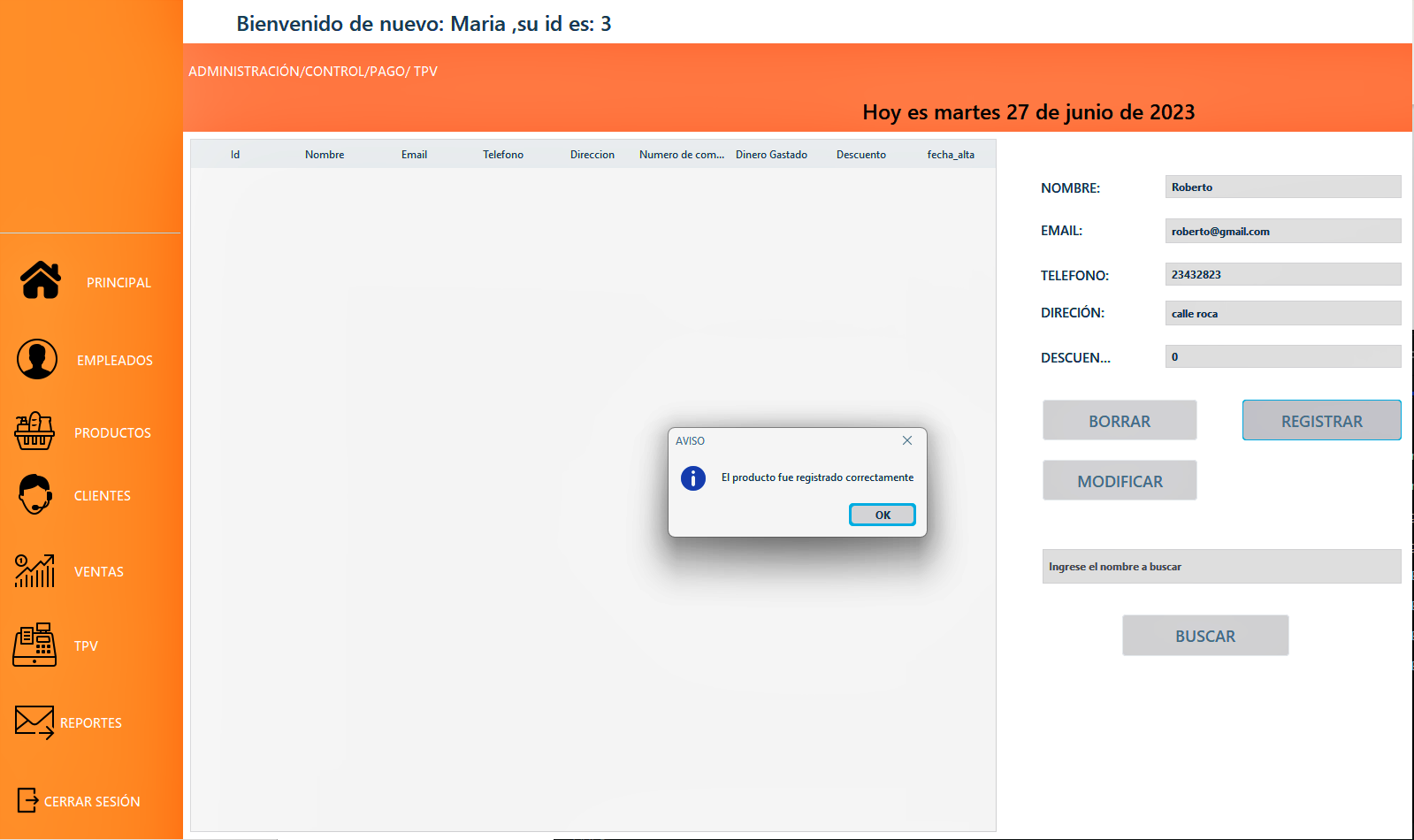
Los botones de borrar y modificar tienen la misma funcionalidad que en apartados anteriores.

## Clientes

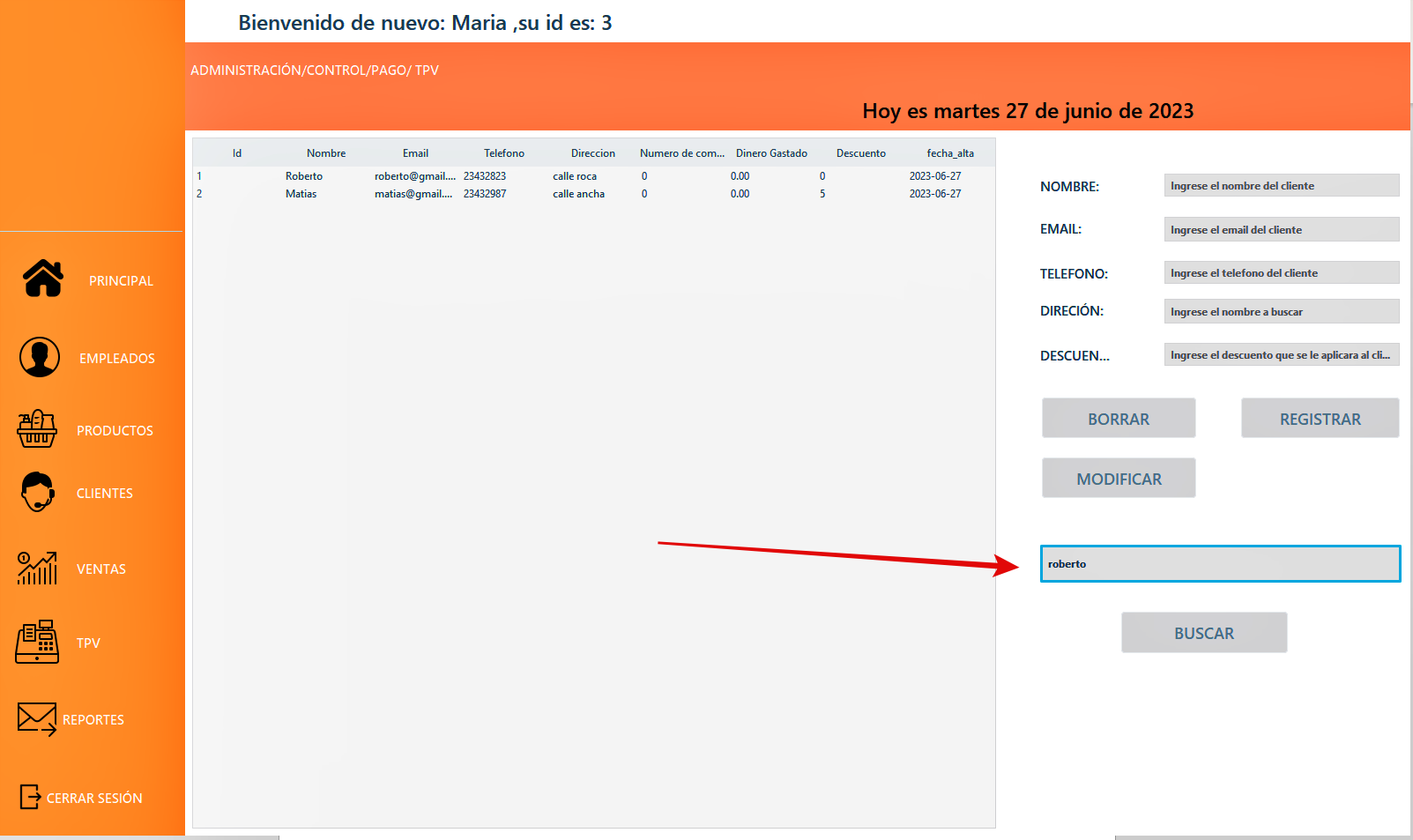
En la parte de clientes tengo configurado en el descuento un numero de compras en el cual, si un cliente adquiere un cierto nivel de compras recibirá un descuento, según valla subiendo las compras que haga le llegará a subir hasta un 10%, pero en el caso de que la empresa decidiera darle un descuento directamente podría ponerlo, aun así esta configurado para que no se pueda poner ni mas de 10 ni se pueda poner otro tipo de carácter.



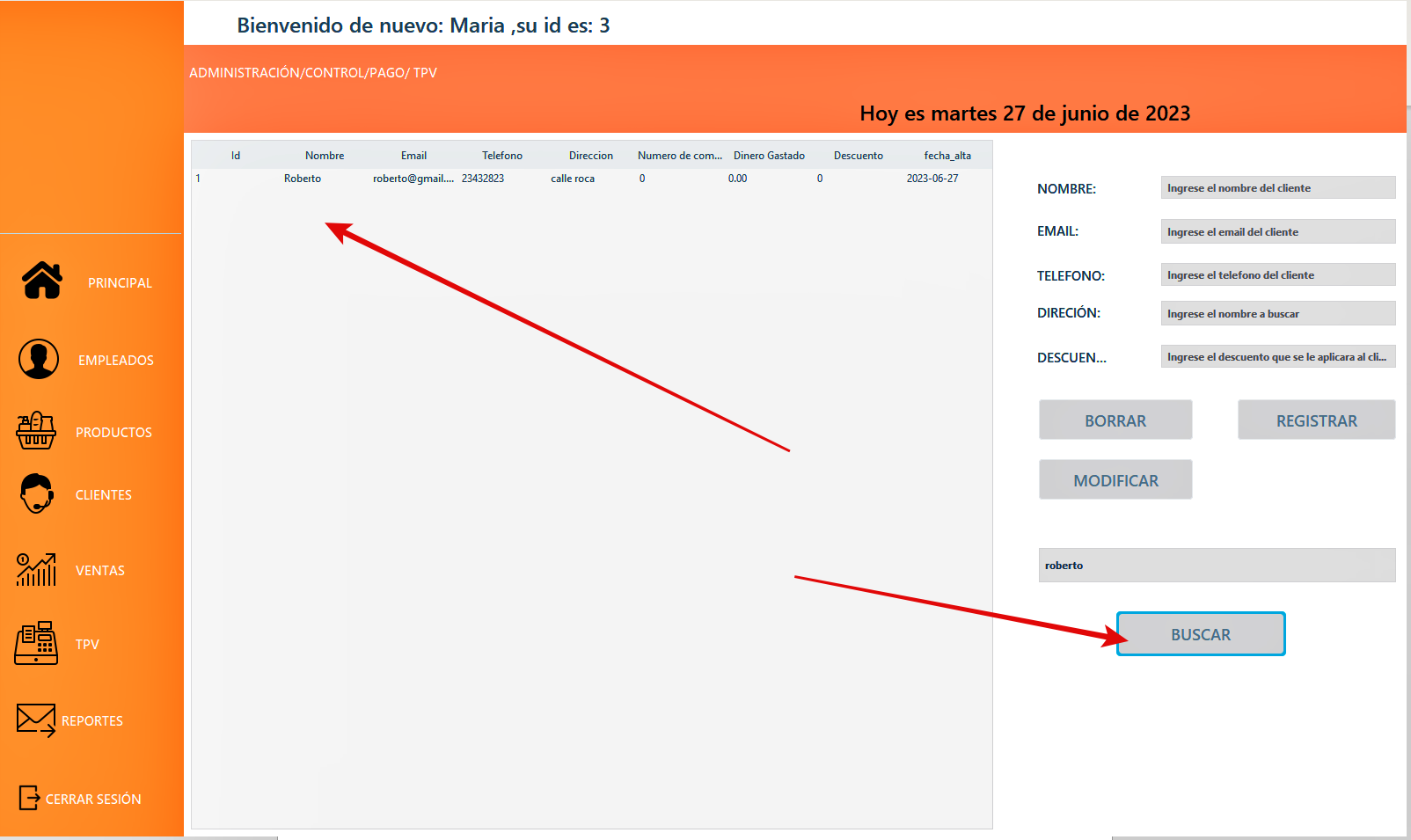
Este es el error que saldría si intentamos poner un descuento mayor de 10.

**

En caso de que todo este perfectamente nos saldrá un mensaje como este.

**

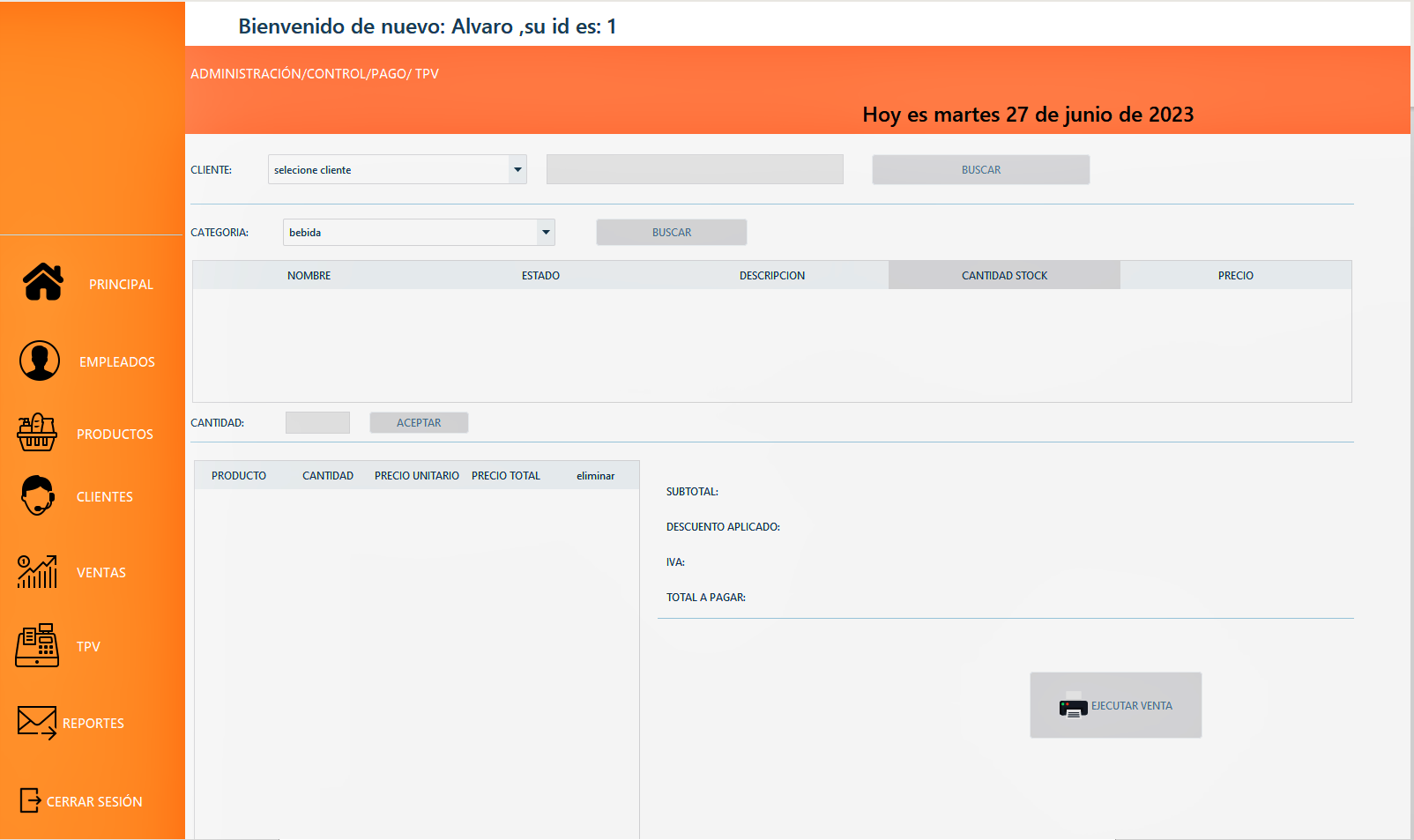
En la barra de búsqueda podemos filtrar por nombres de clientes, como vemos en esta imagen filtrare por el nombre de Roberto.

**

La tabla se refrescará a todas las búsquedas que contengan la palabra Roberto.

## TPV

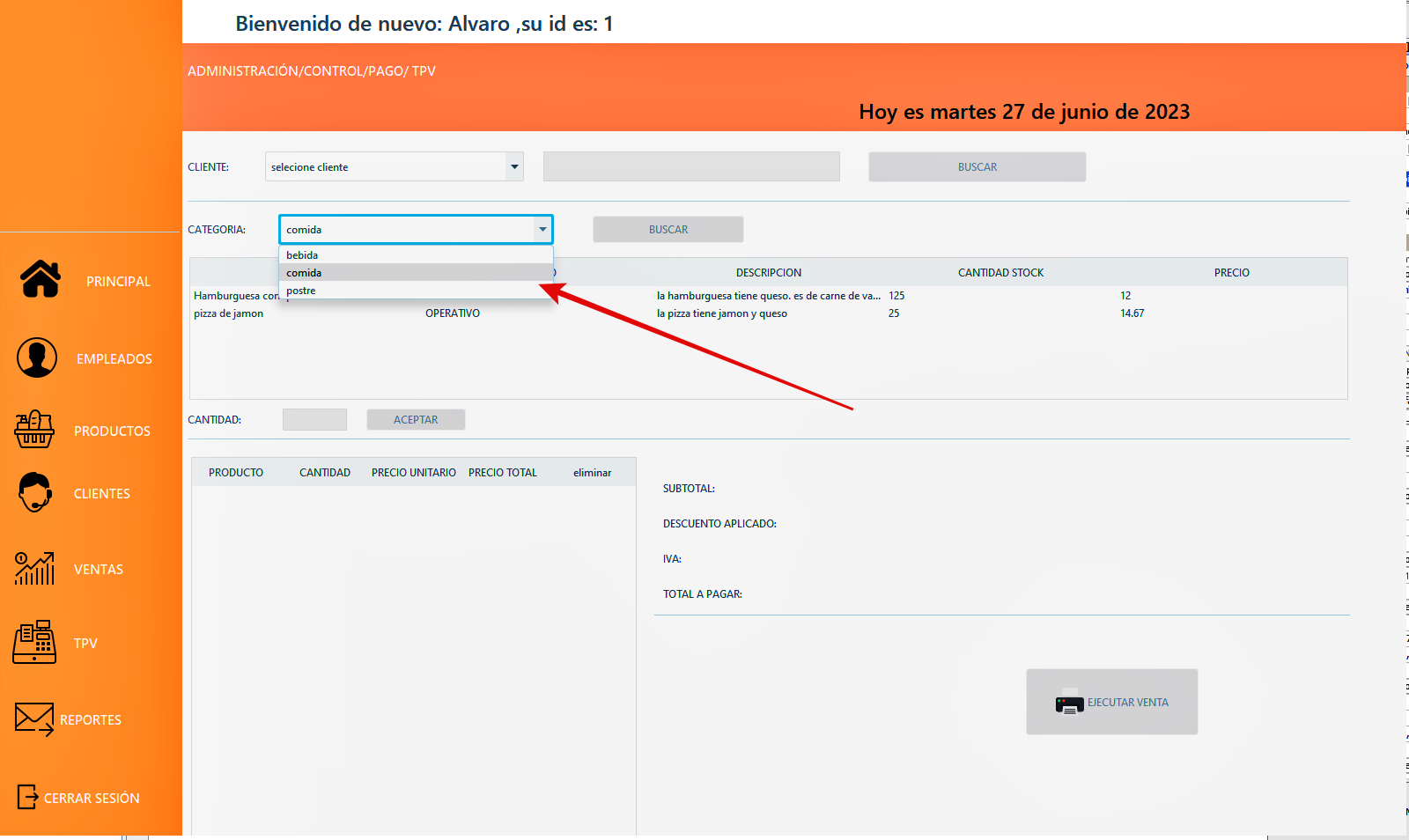
La tpv es una de las partes mas fundamentales de esta aplicación así como la más difícil de configurar, ya que para que la aplicación no cayese hay que hacer un montón de comprobaciones, ya que la tpv sube mucha información a la base de datos en todas las tablas existentes (así como en el id del cliente, el descuento de un cliente, el dinero gastado de un cliente, en la tabla de trabajadores, etc.), he intentado que toda la información de la base de datos venga lo mejor posible hacia esta parte para poder efectuar diferentes tipos de cuentas.



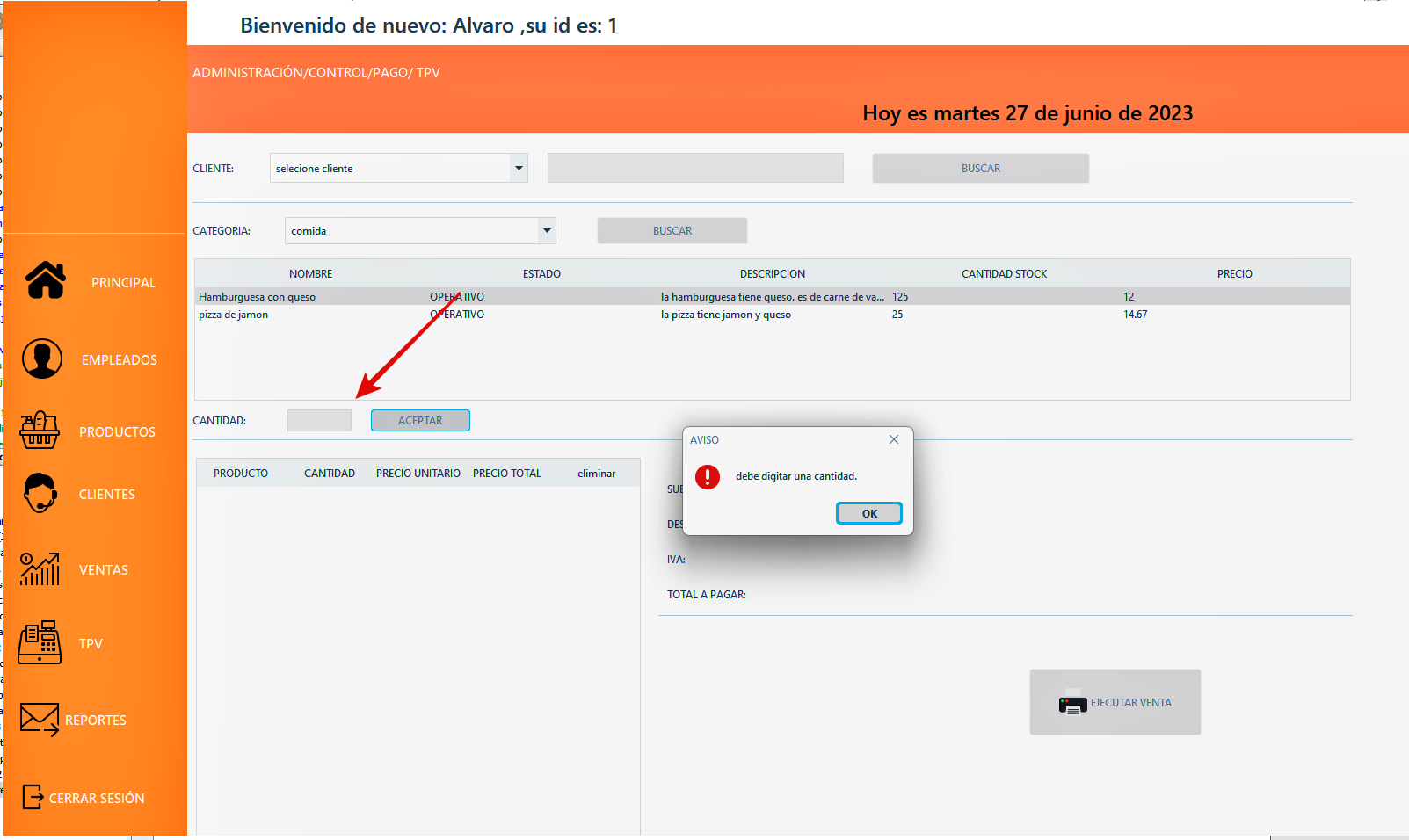
Esta es la interfaz que nos encontraremos al acceder a TPV.

### Sin cliente registrado

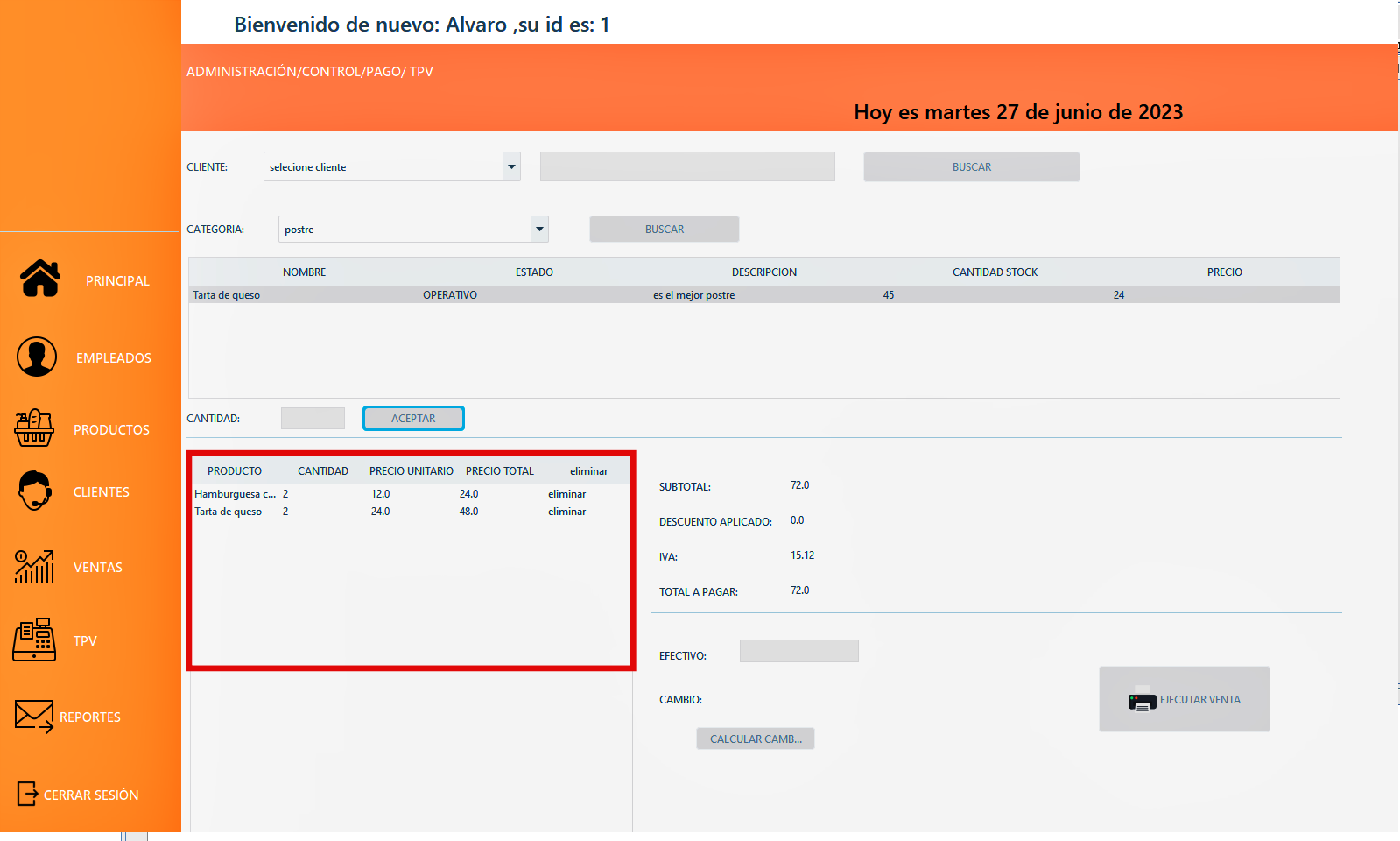
En el ejercicio establecía que al establecimiento podrían llegar clientes no registrados, hago con la compra sea con clientes no registrados.



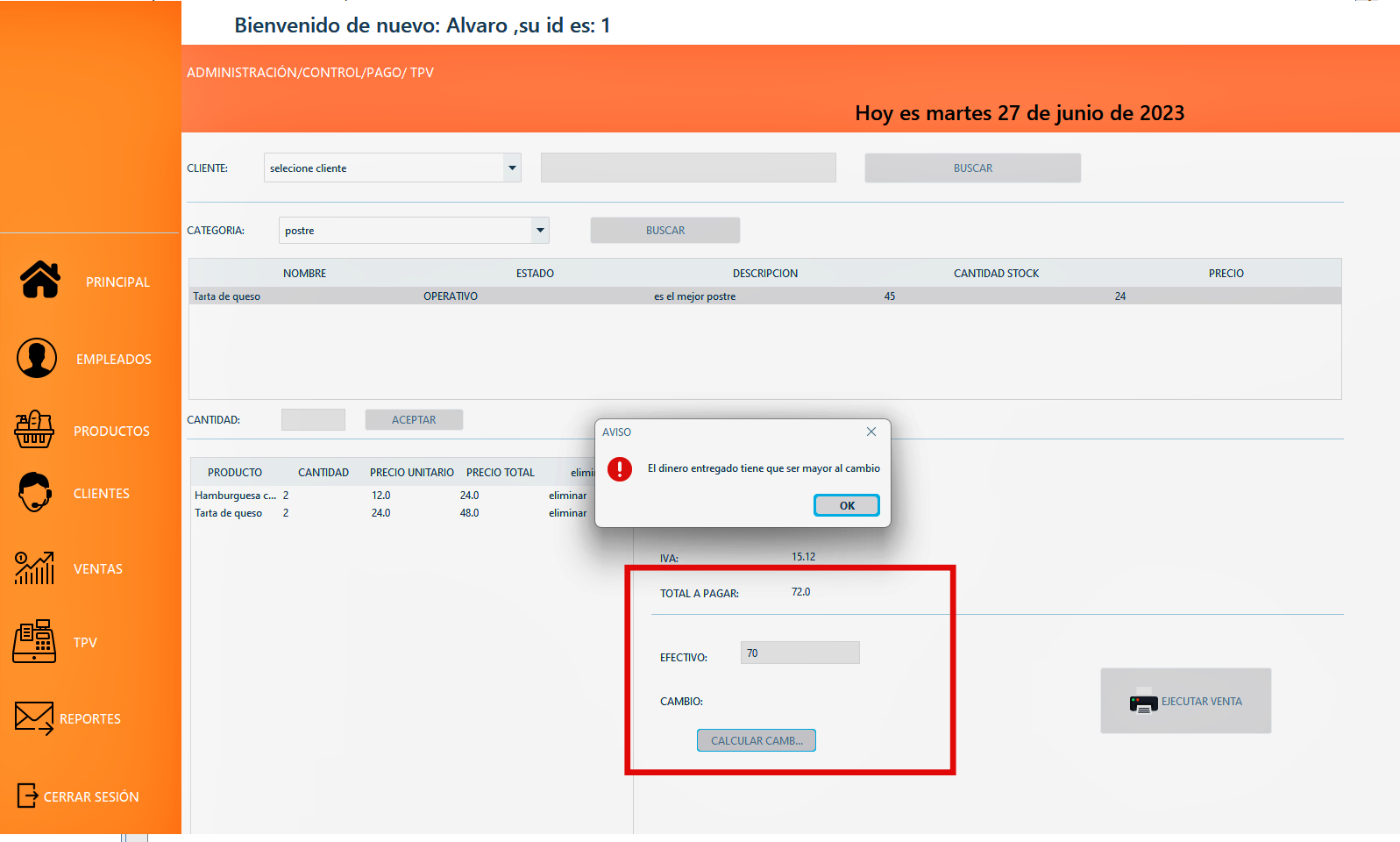
Aquí elegimos la categoría que queremos (por eso era decidí crear selector en el registro, para que no hubiese problemas con la base de datos).

**

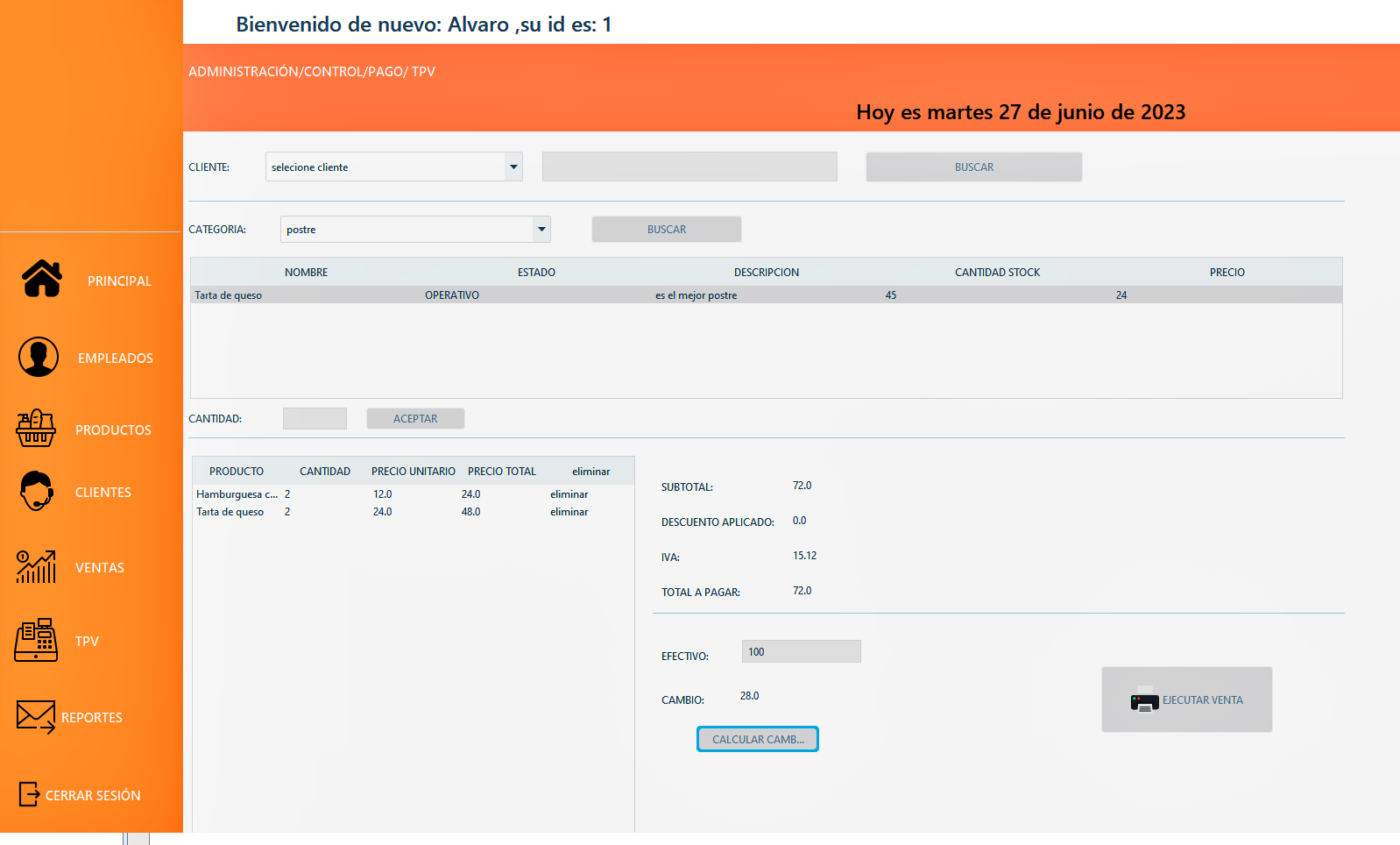
Vemos que en la tabla carga todo lo que tiene con esa categoría y tendremos que digitar la cantidad que queremos seleccionar si no nos saldrá un error.

**

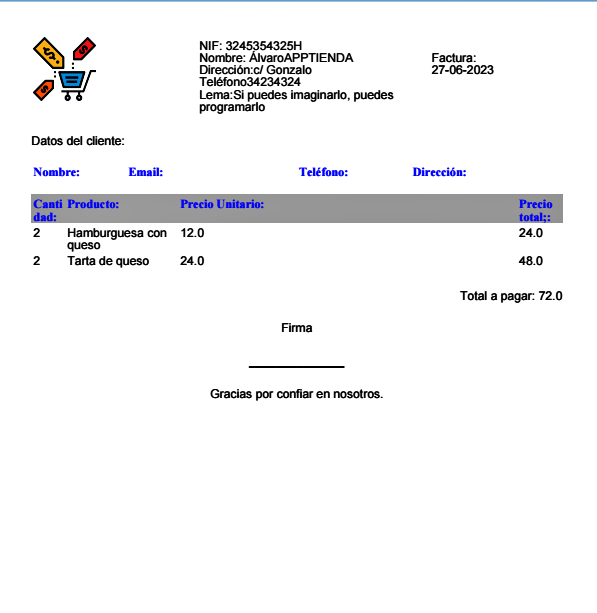
Vemos que según vamos añadiendo nos sale una tabla con todo lo seleccionado(la tabla tiene una función e eliminar en caso de equivocación).

**

También he añadido un campo para el cambio en el cual al digitar una cantidad mas pequeña que la cantidad total a pagar nos saldrá un error.

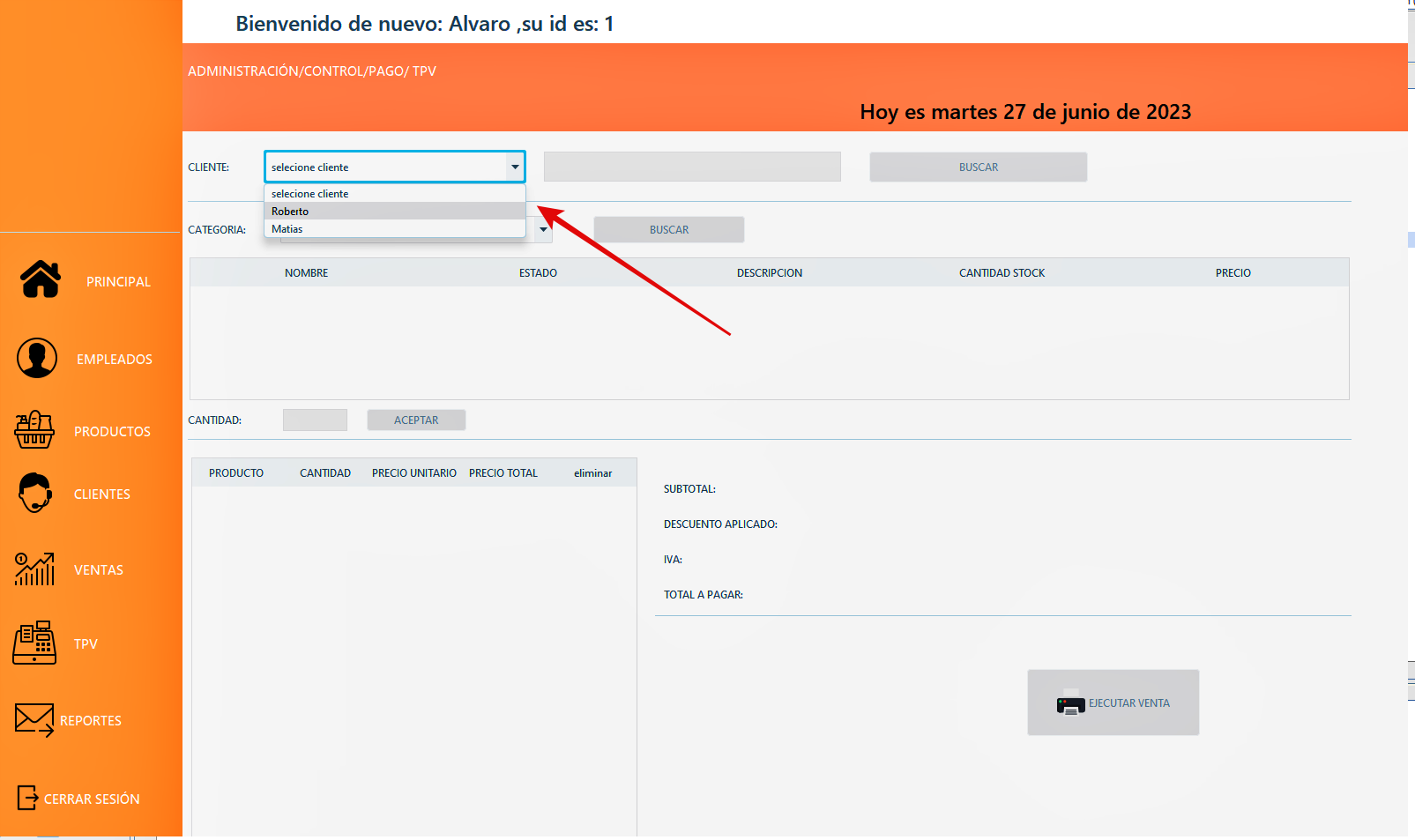
**

En caso de que todo sea correcto vemos que nos dice el cambio que tenemos que entregar al cliente, después si pulsamos sobre ejecutar venta nos saldrá una factura como la de la imagen de abajo.

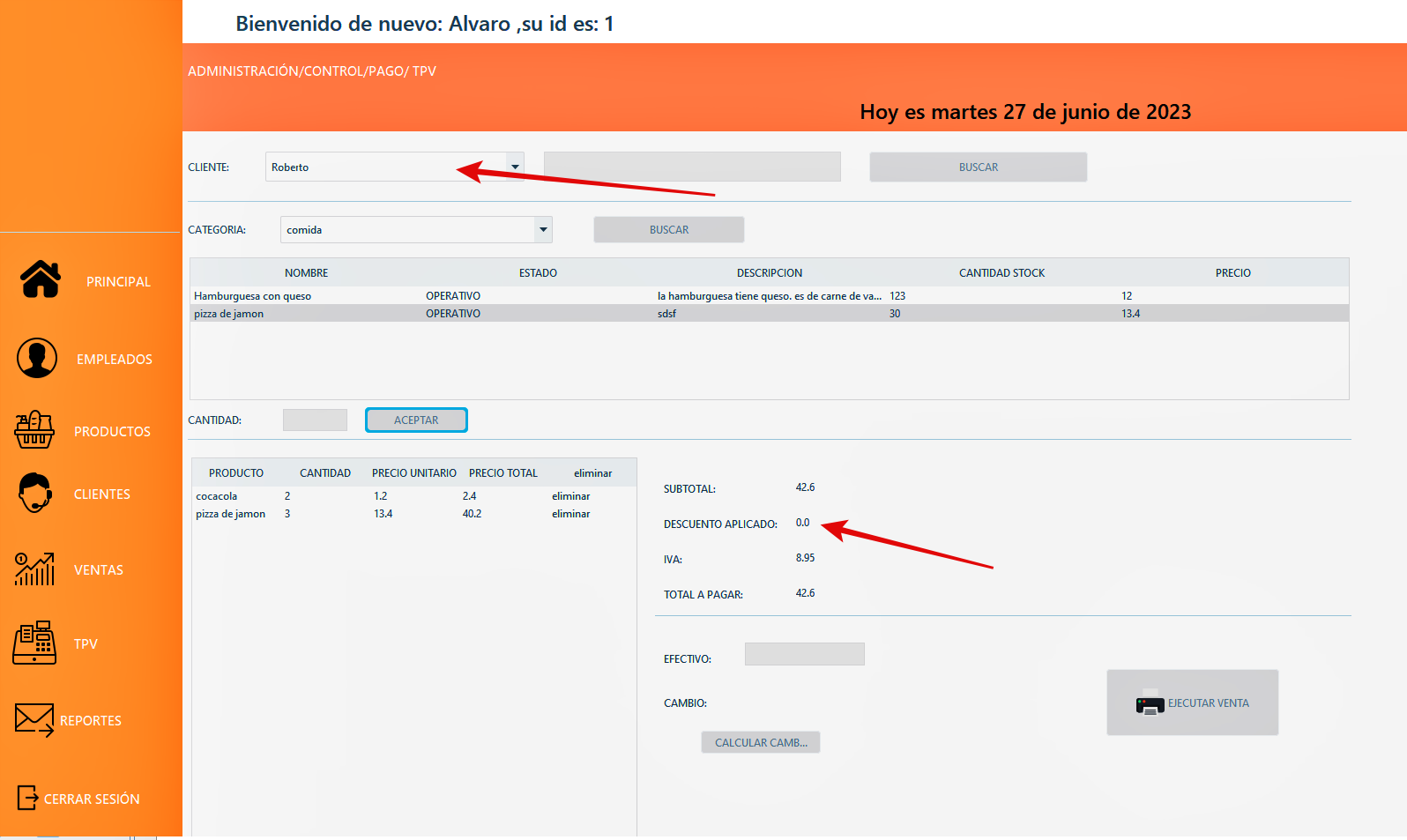
**

### Con cliente registrado

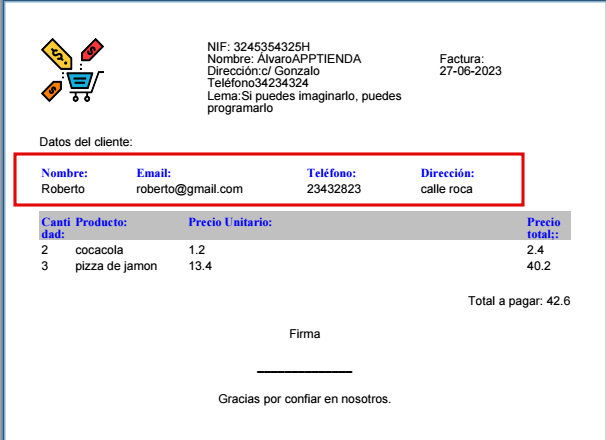
Cuando un cliente es registrado lo que hará en la base de datos es comprobar el descuento que tiene si lo tuviese, una vez efectuada la cuenta lo que hará es subir una venta en la base de datos al cliente y el dinero gastado.



En el selector de cliente están precargador los clientes, en caso de que fuesen muchos podríamos buscarlos mediante nombre en el buscador.

**

En la barra de descuento se cargaría el descuento del cliente si lo tuviese.

**

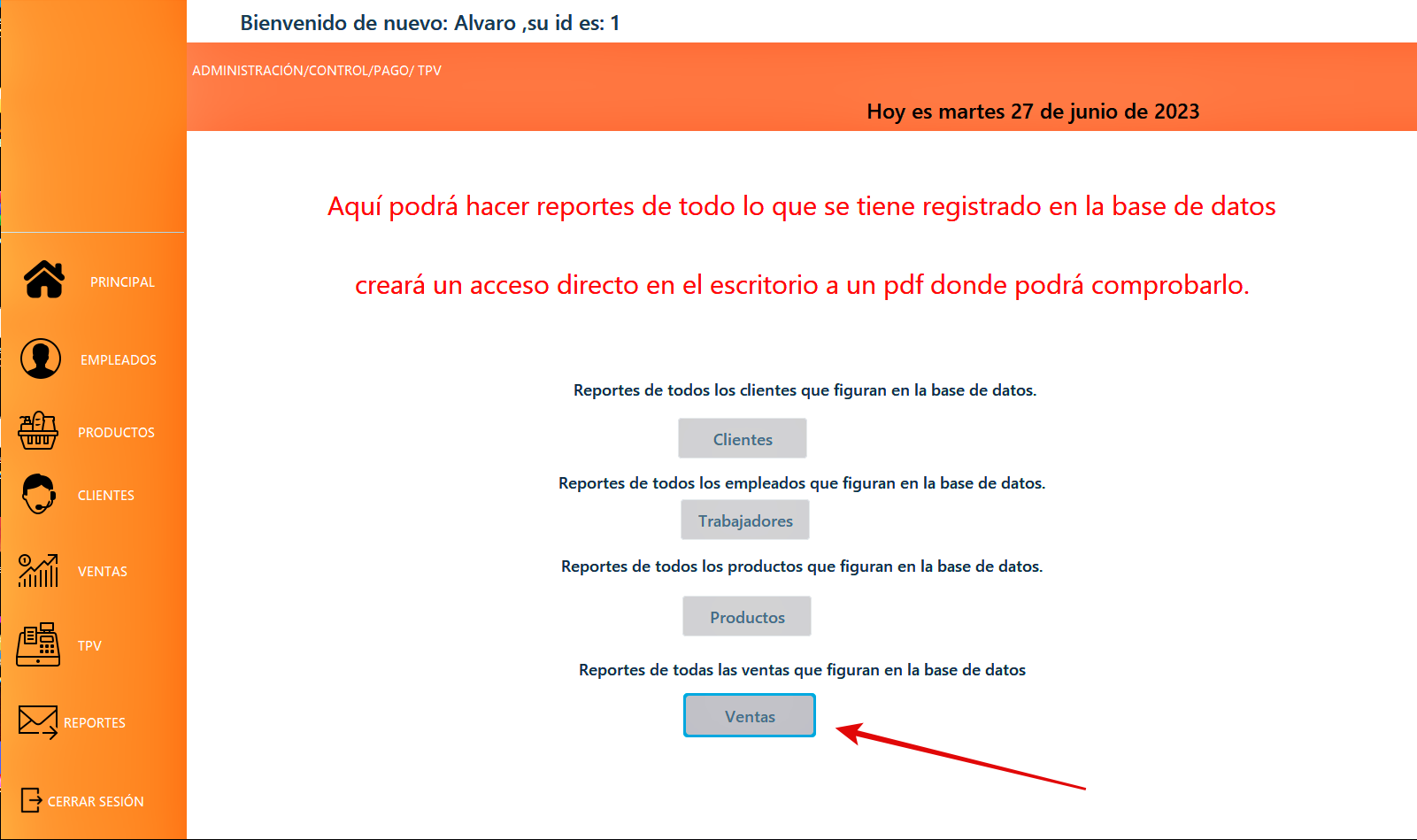
Aquí podemos comprobar que la factura esta con los datos del cliente.

### Notas de TPV

Las facturas quedan guardadas en la carpeta SRC del programa para que no haya perdida de información.

## Reportes

Aquí podemos comprobar que tenemos la opción de hacer reportes de las cosas más importantes de la empresa, por ejemplo, probamos con reportes ventas, lo que nos devolverá es un ‘.pdf’ en el escritorio en el que podremos ver todas las ventas que hemos efectuado, como vemos en el ejemplo son las dos que hicimos mas arriba, como podemos comprobar una de ellas esta asociada a un Id de cliente y la otra no.





## Ventas

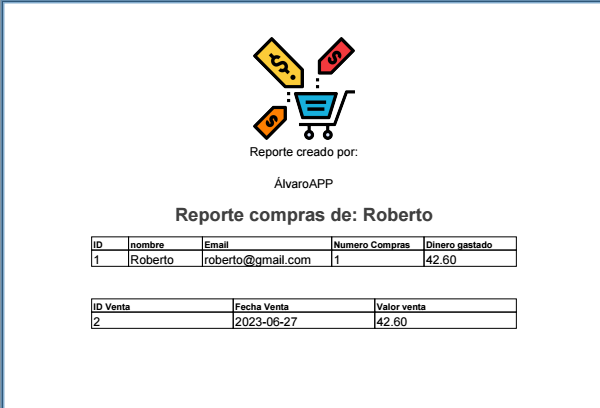
Como pedía el ejercicio que se pudiesen filtrar las ventas por fechas y por clientes he decido crear un menú bar. De ese modo podremos elegir por que queremos filtrar. En el caso de por cliente tendremos que hacerlo mediante nombre, pero como en el menú de tpv hay un selector con los clientes precargados, de las dos maneras generara un ‘.pdf’ en el escritorio con los resultados de la base de datos como podemos ver mas abajo en sendas fotografías.

### Ventas por cliente





Aquí tenemos clientes de la base de datos precargados, pero en el caso de que hubiese muchos podríamos filtrar por el nombre del cliente que buscamos.

**

Aquí podemos comprobar el reporte de las compras que hizo “Roberto” como hemos visto en el ejemplo con su nombre solo hicimos una.

### Ventas por fecha



Para añadir la fecha he usado Jcalendar que he añadido a la librería de mi aplicación, con ello podremos seleccionar la fecha desde un calendario para que no haya problemas a la hora de buscar en la base de datos.

**

El reporte igual que los otros reportes se crean en el escritorio, al haber solo dos ventas hechas en este momento seria el mismo que vimos en imágenes mas arriba.

## Cerrar sesión

En última instancia tenemos el botón de cerrar sesión, ya que como hemos podido comprobar no habría otra manera de cerrar el programa correctamente, pulsando este botón nos devolverá a la pantalla de Loguin ya que allí es de la única manera que podemos cerrar la aplicación.

