goxo denda app

Aplicación para la venta de una tienda de chuches

Desarrollo de Aplicaciones Web. Plaiaundi BHI

Eneko Santa Cruz Galardi

# ÍNDICE

# INTRODUCCIÓN

En este documento se refleja el trabajo realizado durante este año tan difícil e inverosímil, dadas las circunstancias que hemos vivido no sólo en el centro ni en casa, sino en todo el planeta. Dejando la pandemia a un lado, no ha sido un trabajo fácil, hemos tenido que pasar de intentar entender que el mundo se puede organizar perfectamente dentro de unas tablas, hasta programar en un lenguaje totalmente nuevo.

Han sido semanas difíciles, todavía recuerdo el día que no sabía diferenciar una entidad de un atributo. Cualquiera que sepa sobre BBDD se tiraría de los pelos, incluso se reirían, pero al fin y al cabo, todos empezamos así. El plan siempre ha sido tirar hacia delante, poco a poco, y he llegado a conseguir el objetivo que me puse desde que empecé este curso: hacer un programa “chapuza”, pero funcional.

Este documento recoge todo sobre el producto creado de la unión de dos asignaturas, Bases de Datos y Entornos de Desarrollo. El proyecto consiste primero en diseñar la base de datos, organizando todos los procesos en unas aparentemente simples tablas, de las que tanto dolor de cabeza nos han dado durante todo el año. Hay que tener claro cómo relacionar las tablas, pues todo depende de su correcta estructura.

Una vez hecha la estructura de mi BBDD, viene la parte de Entornos de Desarrollo, en la cual, organizamos o creamos el esqueleto de lo que va a ser nuestro programa, diseñando los casos de uso, los diagramas de clase y los diagramas de secuencia.

Sin más dilación, vamos a averiguar de qué trata este proyecto.

# GLOSARIO

BBDD: Base de datos

Programación por capas: La programación por capas es un modelo de desarrollo software en el que el objetivo primordial es la separación (desacoplamiento) de las partes que componen un sistema software o también una arquitectura cliente-servidor: lógica de negocios, capa de presentación y capa de datos.

Visual Studio 2019: Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para Windows, Linux y macOS. Es compatible con múltiples lenguajes de programación, tales como C++, C#, Visual Basic .NET, F#, Java, Python, Ruby y PHP, al igual que entornos de desarrollo web, como ASP.NET MVC, Django, etc., a lo cual hay que sumarle las nuevas capacidades en línea bajo Windows Azure en forma del editor Monaco.

Microsoft Access: Microsoft Access es un sistema de gestión de bases de datos incluido en el paquete ofimático denominado Microsoft Office, sucesor de Embedded Basic.

Query: Query string o, en español, cadena de consulta es un término informático que se utiliza para hacer referencia a una interacción con una base de datos. Es la parte de una URL que contiene los datos que deben pasar a aplicaciones web como los programas CGI.

# LA APLICACIÓN

He de mencionar que la aplicación está desarrollada en vb.net, un lenguaje que ni siquiera sabíamos de su existencia en la primera mitad de la segunda evaluación.

La aplicación, como bien dice el título, es sobre la venta de una tienda de chuches. La idea vino de una amiga que empezó hace no mucho en una tienda de chuches de mi barrio, la típica tienda de toda la vida, en la que por problemas de salud, pudo entrar ella a trabajar. Sí, es una amiga. La cuestión es que siempre la veía haciendo las cuentas a papel y lápiz. No estoy en contra del papel y lápiz, solo que en esta sociedad tan modernizada que ya existen desde coches eléctricos, hasta otros aparatos eléctricos de mucho uso, me pregunté: ¿por qué no?

Mi padre trabaja en una carnicería y tiene una báscula conectada a un ordenador con una interfaz gráfica (no mucho mejor que la mía), así que ya tenía un modelo a seguir. El resultado, lo veremos más adelante.

La principal de esta aplicación era gestionar la tienda de chuches entera, desde la entrada de los encargos de los proveedores, hasta la venta de un chicle de cinco céntimos al hijo de la vecina. Sin embargo, el alcance de esta aplicación se ha visto drásticamente reducida por el tema del Covid, el impacto social que ha tenido y realmente porque todos los profesores se han puesto de acuerdo para mandar proyectos muy potentes en un plazo muy corto.

Como el tiempo se me echaba encima, finalmente me he centrado en la venta de los productos de la tienda, crear pedidos, añadir artículos, modificarlos… La sección de los proveedores se ha quedado paralizada, por el momento.

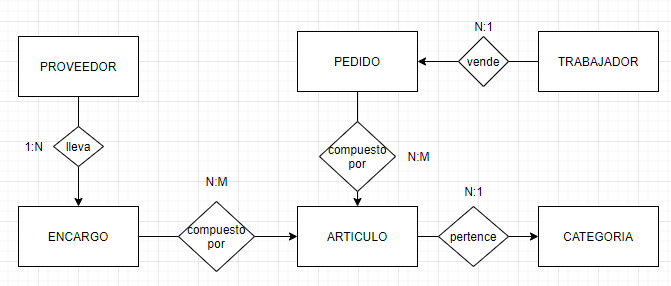
Así que desde ese momento, me he centrado en el correcto funcionamiento de una compra ordinaria, como cuando vas a comprar el pan. La aplicación es capaz de guardar la compra en una tabla, quitar algún artículo del pedido, añadir nuevos artículos… Se puede decir que hace todo menos la impresión final del ticket.

La aplicación va dirigida a aquellos que tienen una tienda de chuches pequeñita, para facilitar la gestión y las ventas para verlas de una manera más visual.

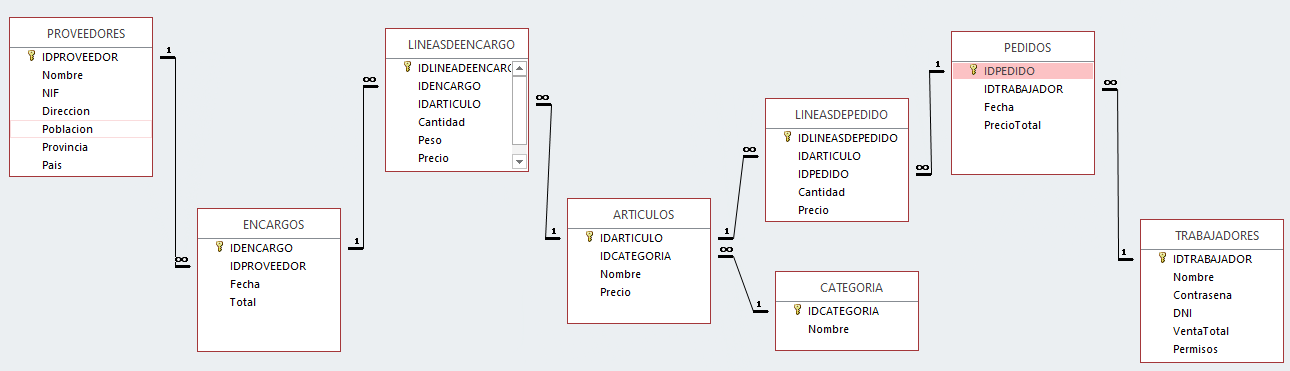
# MANUAL TÉCNICO

## Arquitectura de la BBDD

Modelo Entidad/Relación:



Modelo relacional:



Como podemos observar, la BBDD no tiene mucha complicación. Tenemos una tabla de trabajadores, donde se van a registrar los datos que necesitaremos de los mismos. Estos trabajadores crearán pedidos, la parte fundamental de mi aplicación. Los pedidos estarán compuestos por una o varias líneas de pedido, que a su vez son compuestas por los diferentes artículos que tengamos registrados.

Los artículos pertenecen a una categoría, por comodidad y rapidez. Después veremos la importancia que tienen las categorías en la interfaz gráfica. Estos artículos los pedimos de diferentes proveedores, los cuales vamos a registrarlos en la tabla de proveedores, con sus respectivos datos.

Como en la parte de los pedidos que hacen los trabajadores, tenemos este mismo formato para los proveedores. Nosotros hacemos encargos a estos proveedores, que están compuestas de líneas de encargo y a su vez, de artículos. Como he mencionado antes, no es una BBDD muy compleja a la hora de entenderlo.

Tras crear la BBDD, mi objetivo era reunir y controlar todos los campos y tablas mencionados anteriormente, pero se puede decir que la aplicación se ha quedado en una fase beta, y he enfocado el objetivo del mismo a la venta de artículos de un trabajador. No he querido meterme mucho con los proveedores, por falta de tiempo más que nada.

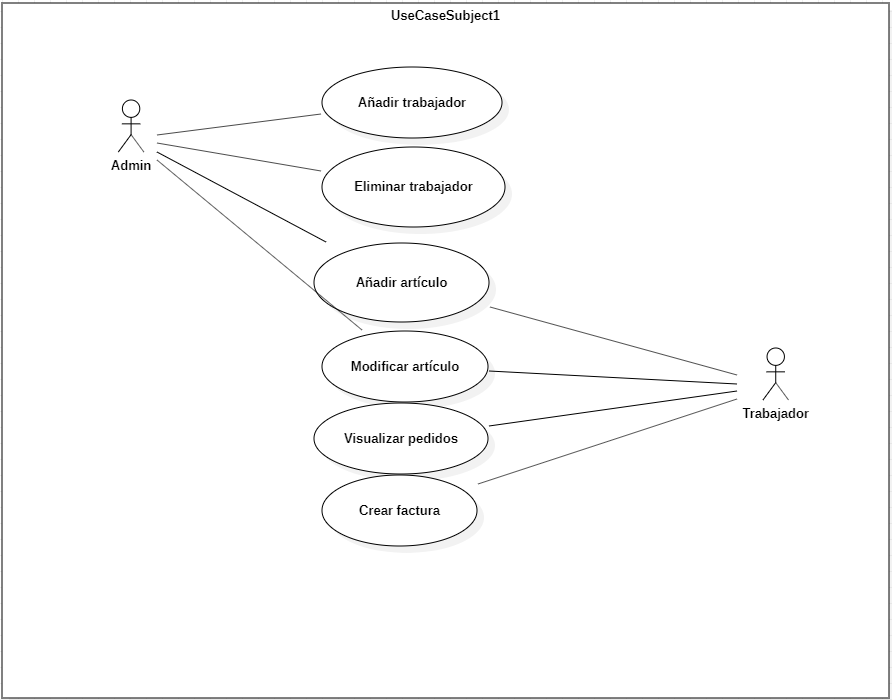
Por lo tanto, se podría decir que el objetivo principal de esta aplicación es la gestión de ventas de una tienda de chuches. Podría haber creado una tabla de clientes para guardar sus datos para futuras compras, pero lo he descartado completamente, no es algo sea muy importante, por lo menos en este ámbito.

## ANÁLISIS

En esta aplicación, el administrador tendrá control sobre los trabajadores. Podrá tanto añadir como eliminar trabajadores, y también podrá modificar y añadir nuevos artículos.

El trabajador, por otro lado, solamente podrá gestionar artículos y crear pedidos. No podrá acceder a la sección de trabajadores, por cuestiones de seguridad. De todas formas, la persona que lleve la aplicación tendrá permisos de administrador para que lleve la gestión de su tienda, y si hace falta, añadir nuevos trabajadores.

En este diagrama de Casos de Uso se refleja lo anteriormente mencionado:

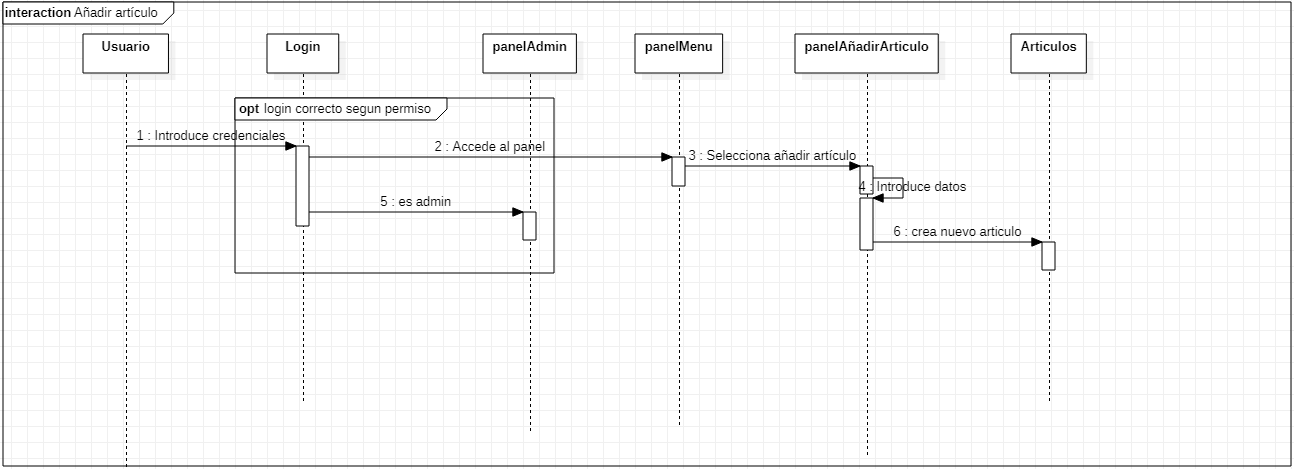


La idea era inicial era que el programa tuviese 4 apartados: Trabajadores, Pedidos, Artículos y Proveedores. Si el usuario tiene permisos de administrador, podría entrar en la sección de trabajadores. Como he mencionado antes, me he enfocado en la gestión de ventas de artículos, la creación del pedido… Por ello, las secciones de trabajadores y proveedores se encuentran deshabilitadas y los casos de uso “Añadir trabajador” como “Eliminar trabajador” no se encuentran operativas (aunque sí implementadas).

Para entender un poco más de cómo es el funcionamiento de los Casos de Uso, dejo aquí una breve explicación del procedimiento:

|  |
| --- |
| **Nombre:** **Añadir Artículo** |
| **Descripción:**  El trabajador/administrador solicita añadir un artículo. |
| **Actores:** Trabajador/administrador |
| **Precondiciones:**  Se requiere que el trabajador/administrador esté registrado en el sistema. |
| **Curso normal del caso de uso:**   1. El trabajador/administrador selecciona añadir artículo. 2. El sistema le devuelve campos para rellenar. 3. El trabajador/administrador rellena los campos. 4. El trabajador/administrador pulsa el botón de añadir. 5. El sistema comprueba que los campos rellenados son válidos. 6. El sistema añade el nuevo artículo. |
| **Postcondiciones:**  El sistema añade el nuevo artículo. |
| **Alternativa:**   1. Los campos rellenados son incorrectos. 2. Fin de caso de uso. |

En el diagrama de secuencia que está a continuación, se ve el proceso de “vida” del caos de uso “Añadir artículo”.



El usuario se identifica con sus credenciales y si tiene permisos de administrador, va al panel del menú principal. Si no tiene permisos, accede al panel de gestión de ventas. En ese panel, podrá clicar en el botón “Añadir artículo” y le saldrá unos campos a rellenar (nombre, ID, categoría y precio). Una vez los rellene, si todos los datos son correctos, el programa creará un nuevo artículo con sus respectivos datos.

## PROGRAMACIÓN Y DESARROLLO

Para llevar a cabo la programación de esta aplicación, he necesitado dos cosas esenciales: primero, una base de datos donde guardar toda mi información; y segundo, un entorno donde desarrollar y crear código para la aplicación.

La aplicación está desarrollada en Visual Studio 2019, en el lenguaje vb.net, con un intento de hacerlo por capas. Está compuesta por varios formularios, cada uno con sus funciones dentro de su código.

La BBDD es de formato .accdb, propio de Microsoft Access. Tuve mis problemas al principio a la hora de conectarla, pero al final ha resultado ser un formato más que suficiente para llevar adelante el proyecto.

Primero tuve que visualizar lo que quería un vendedor, una pantalla simple pero funcional y completa a su vez. Así que el primer formulario no podía ser otra que el panel de ventas. Sin duda esta es la que más trabajo ha dado, a simple vista parece un formulario normal, pero hay mucho código por debajo.

He intentado jugar con colores para que la interfaz sea más atractiva e intuitiva. Cada una de las categorías, por ejemplo, las representa un color con el fin de facilitar la detección de la categoría que quiera seleccionar el vendedor.

En cuanto a las dificultades, han habido unas cuantas durante todo el proceso, pero la más difícil sin lugar a dudas ha sido el tiempo. No estoy culpando la metodología, dado que mi organización tampoco ha sido la mejor.

En cuanto a la programación, ha habido algunos problemas que me han llevado mucho tiempo corregir, y resulta que eran una auténtica tontería. Por lo visto, Microsoft Access tiene algunas “limitaciones” o “manías” a la hora de recibir información desde una aplicación. Tardé dos horas en ver qué es lo que estaba mal. Revisé el código 20 veces, la función estaba bien escrita, la query estaba perfecta, pero el resultado siempre era el mismo, no hacía nada. Tras dos horas, me di cuenta de que en una query que le mandaba a la BBDD, los parámetros que le pasaba, estaban en distinto orden a la que decía al enunciado de la query. Según he podido escuchar, SQLite y SQL no tienen esas “manías” de recibir información de manera ordenada.

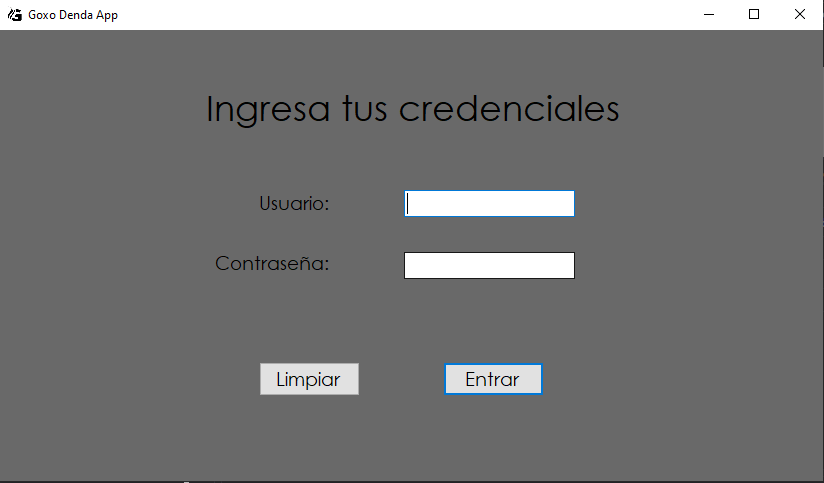
Por otro lado, si me encontraba en algún problema de código o de implementar nuevas funcionalidades, bastaba con mirar un poco por internet para conseguir solucionar las dudas.

En resumen, la aplicación no ha sido difícil de hacer, sino que ha faltado tiempo para poder exprimir el contenido al máximo.

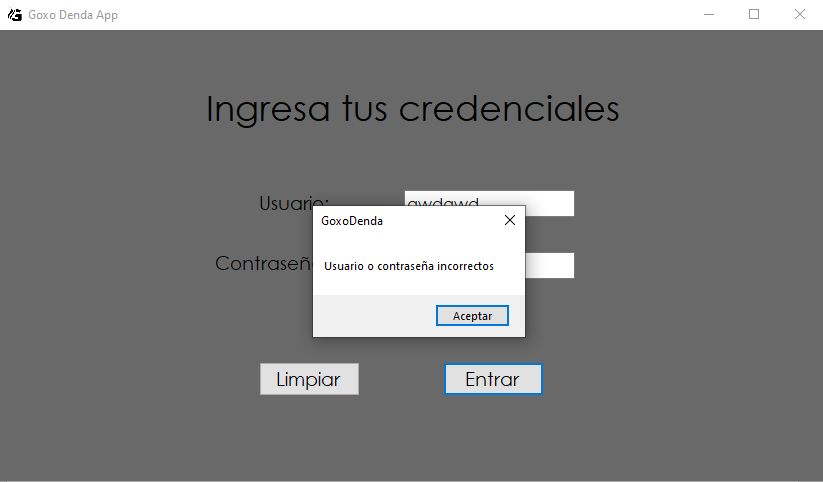
# MANUAL DE USUARIO

Nada más ejecutar el programa, nos saldrá la ventana de login. Para poder entrar, tendremos que tener una cuenta. Para ello, el “admin” podrá crear todos los trabajadores que quiera.

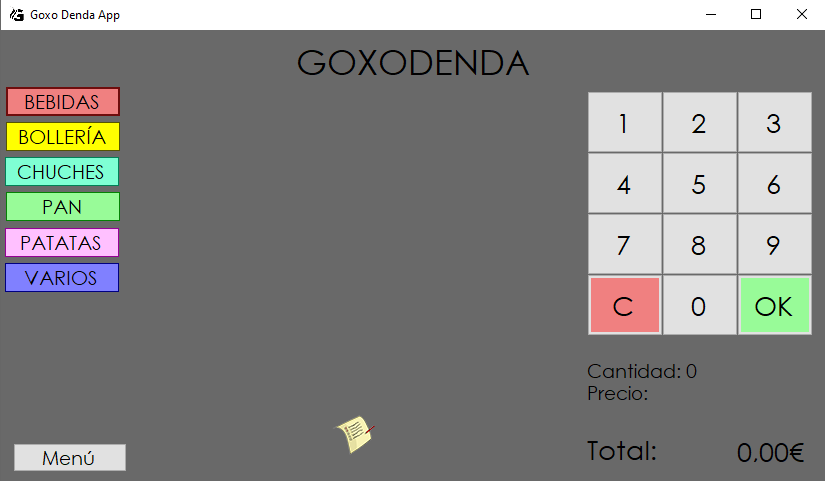
## PANEL DE LOGIN



Si introducimos una cuenta inexistente, nos saldrá el siguiente mensaje.



Sin embargo, si logueamos con una cuenta existente, nos llevará directamente al panel de venta, si no tenemos permisos de administrador.



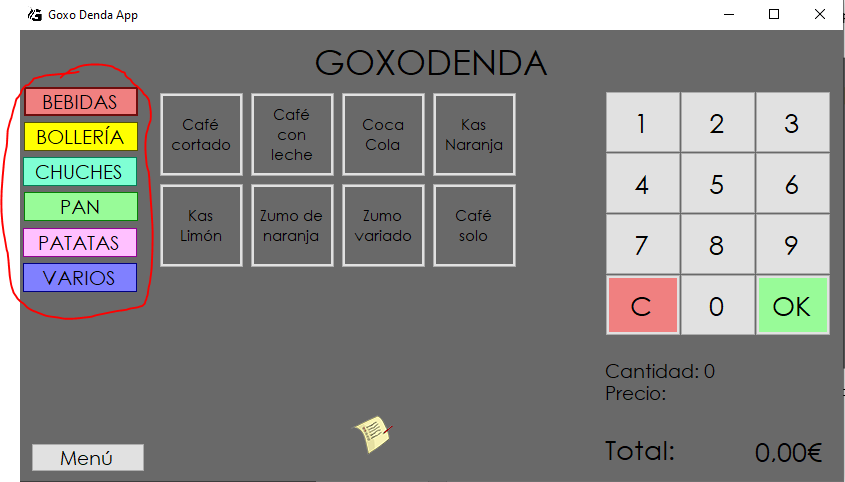
Si tuviésemos permisos de administrador, aparecería el siguiente panel:



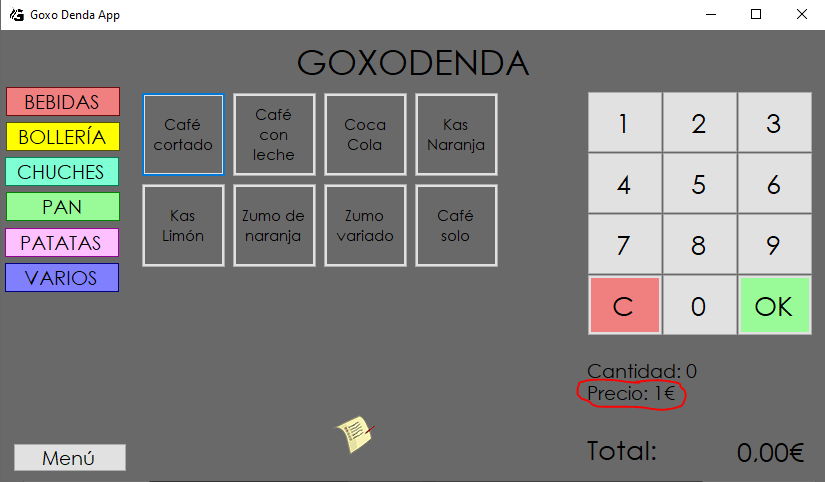
Todos los usuarios pueden acceder a este panel, pero solamente aquellos que tengan rango de administrador podrán entrar en la sección de Trabajadores.

## PANEL DE VENTA

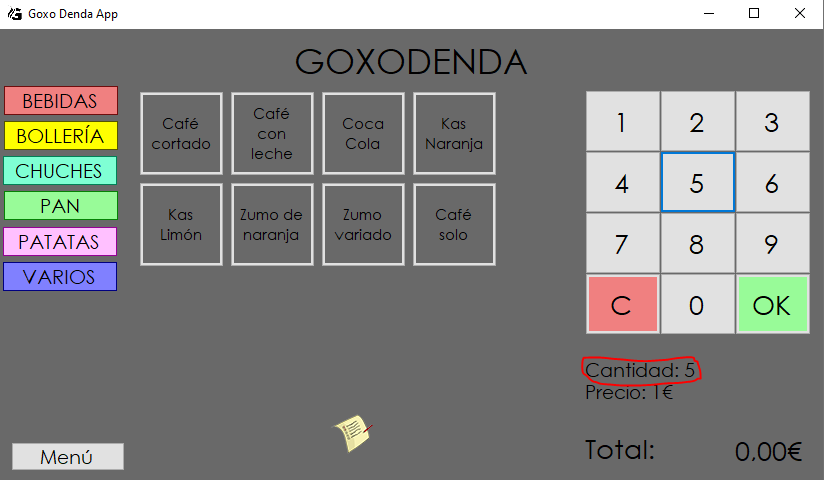
El panel de venta es muy intuitivo. A la izquierda aparecen las diferentes categorías de los artículos. Si clicamos en una de ellas, en el panel del medio se cargarán todos los artículos de dicha categoría.



Si seleccionamos un artículo, el botón del artículo se pondrá con el borde de color azul para saber que lo hemos seleccionado. Acto seguido, se mostrará el precio del artículo en la esquina inferior derecha de la pantalla.



En el panel de la derecha, donde aparecen números del 0 al 9, seleccionaremos la cantidad de artículos que queremos vender. Para ello, pulsaremos dicho botón y nos mostrará la cantidad que hemos seleccionado.



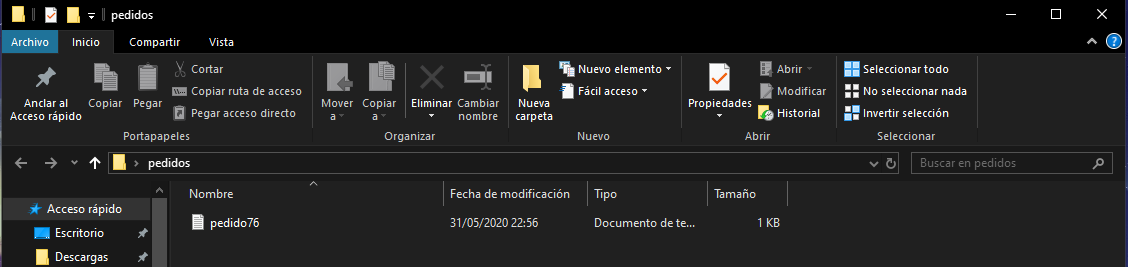
Una vez elegido el artículo y la cantidad, solo nos queda añadirlo al pedido. Para ello, clicaremos en el botón verde “OK” y se añadirá a la lista. Acto seguido, nos mostrará el total que lleva el pedido.



Seguiremos así hasta que terminemos con el pedido. Después, solo tendremos que pulsar el botón donde aparece la imagen de la cuenta, y se cargarán los datos.



Si el cliente se arrepiente de comprar algo, o si quiere añadir una cantidad menor del artículo, se puede borrar la línea clicando en el botón rojo “Eliminar”. Si el cliente está satisfecho, pulsaremos el botón de finalizar y se creará el pedido.

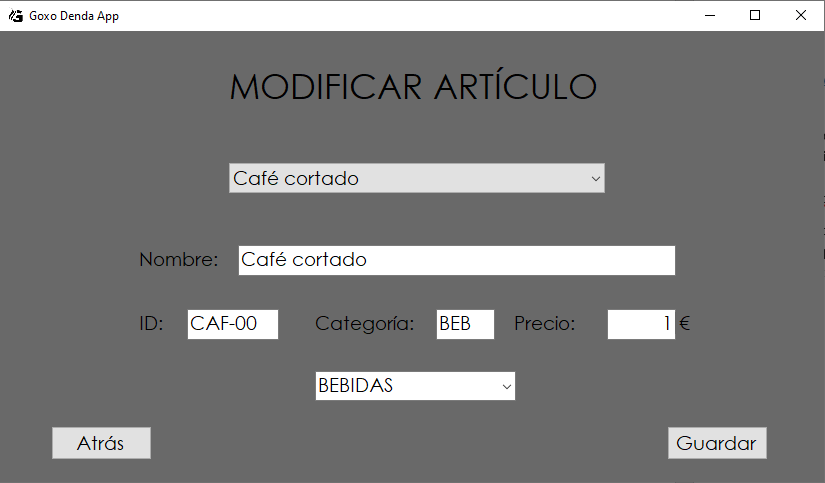


Se creará un .txt por cada pedido que hagamos, con su ID del pedido. Si abrimos el archivo, debería tener toda la información sobre el pedido.

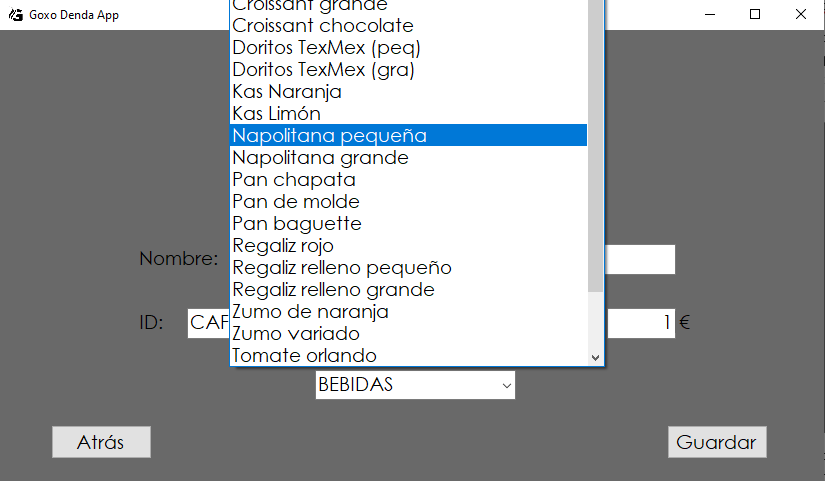
## PANEL MENÚ

Podemos acceder al panel de menú desde el botón “Menú” desde el panel de ventas, ubicado en la esquina inferior izquierda. Las secciones de Trabajadores y Proveedores no están implementadas, pero la idea es que solamente el usuario con permisos de “admin” pueda acceder a la sección de trabajadores por cuestiones de seguridad.

Si clicamos en el botón de “Artículos”, nos aparecerá dos nuevos botones. Tendremos la posibilidad de modificar o añadir un nuevo artículo. Si seleccionamos en modificar artículo, podremos ver la siguiente pantalla:



Si despegamos el combobox, tendremos la lista entera de todos los artículos. En cuanto cambiemos la selección, se cargarán sus datos. Si queremos modificar, solamente tendremos que cambiar la información de los paneles, y clicar en guardar.



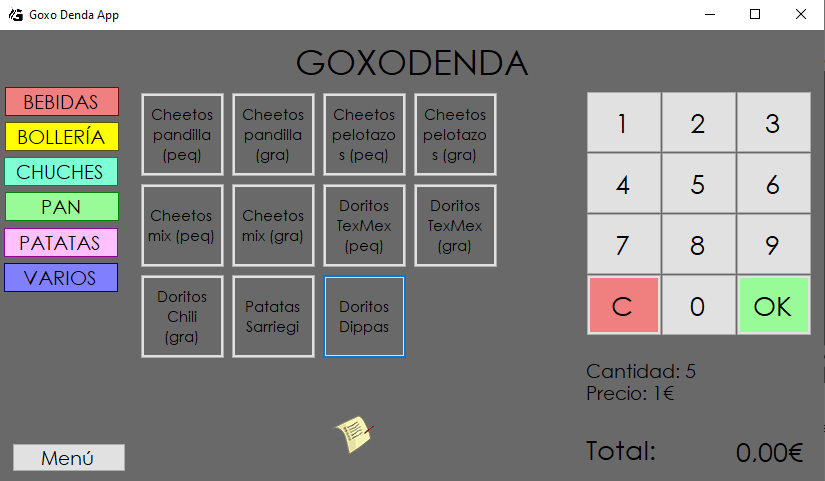
Si en el panel de modificar artículo seleccionamos “Añadir artículo”, podemos apreciar la siguiente pantalla:



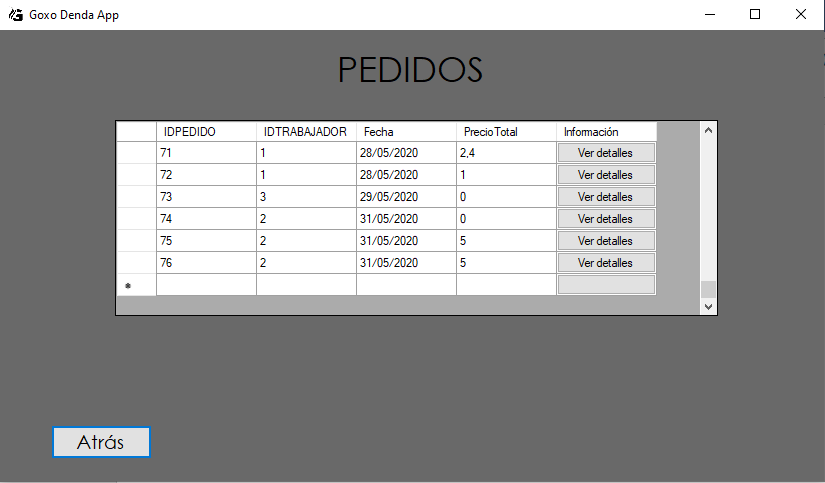
En este panel podremos añadir un nuevo artículo, con sus respectivos campos. Si creamos un artículo, se guardará en la BBDD automáticamente.



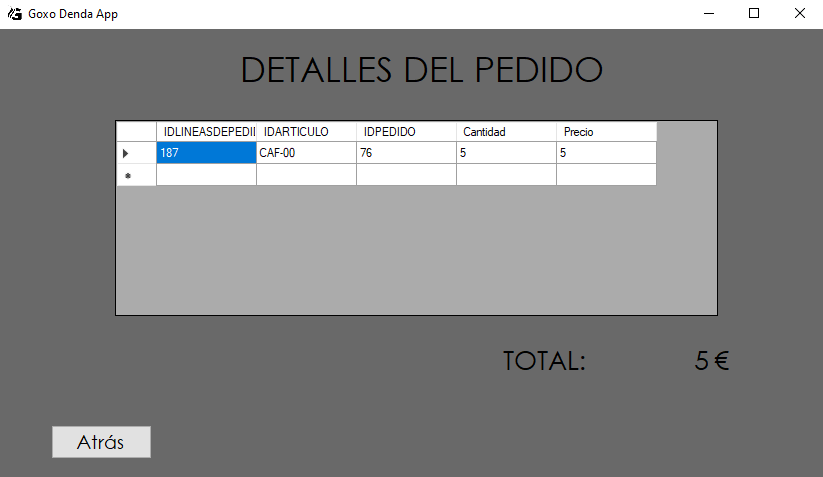
Introducimos los datos y clicamos en finalizar. Si vamos a la sección de ventas, podemos ver que se ha creado el artículo exitosamente.



## PANEL DE PEDIDOS

Si entramos en el panel de pedidos, se podrán visualizar todos los pedidos que se han hecho hasta la fecha. Todavía está por implementar la función de escoger los pedidos que se hayan hecho en *x* fecha. 

Si clicamos en el botón de “ver detalles”, nos sacará un resumen del pedido.



# CONCLUSIONES

Este programa me ha servido para ver realmente de lo que soy capaz. Si alguien me hubiese dicho en septiembre que al final del curso desarrollaría una aplicación, seguramente le tomaría por loco.

Estoy muy contento con el resultado, aunque me hubiera gustado meter muchas más funcionalidades. Al ser el primer proyecto individual, no sabía cuántas horas necesitaría para hacer una aplicación, y ese sin duda ha sido un gran fallo. No he sabido gestionar bien las horas y por ende, creo que ha quedado una aplicación “simple”. La parte más negativa creo que ha sido el dejar de asistir a clase.

Realmente me hubiera gustado dedicar más horas a la aplicación, he elegido un tema que me gusta y que podía lograr algo bonito, pero el confinamiento y la carga de trabajo adicionales han dificultado la tarea. Estas dos últimas semanas las he pasado muy mal, han sido puro trabajo y he llegado a estresarme (hace falta mucho para que yo me estrese), pero ya que hemos llegado hasta aquí, ¿por qué no seguir un poco más? He empezado a pensar que la desesperación o por lo menos el: “Soy tonto, ¿cómo no me he dado cuenta antes?” es el día a día de un programador. Cuántos errores habré tenido que era una auténtica tontería, que he tardado más de una hora buscando la solución, y resulta que era tan sencillo como intercambiar las líneas de orden. Lo mejor de esto es que errores tan “tontos” se me quedan grabados en la cabeza y no creo que vuelvan a ocurrir. Y si suceden, por lo menos, ya tengo la solución.

Me ha gustado la aplicación de Visual Studio, he tenido mis peleas al principio sobre cómo gestionar todas las funcionalidades que tiene, pero al final le he cogido cariño. No tenía ni idea hasta que he empezado a fisgonear por el programa. Las búsquedas de Google han servido muchísimo, aunque a veces costaba un poco encontrar la solución correcta.

Aunque haya entregado el proyecto, aún quedan pendientes algunos apartados de la aplicación. Ahora ya que no tengo la presión de entregarlo para un día en concreto, y que me siento bastante liberado, la intención es seguir trabajando durante el verano para terminar la aplicación al 100%.

En resumen, ha sido una experiencia muy bonita y reconfortante. Quizá no sea una gran aplicación, pero es la primera que hago y estoy muy orgulloso.