

# Proyecto BBDD



Realizado por: Xabier Jauregi Bravo

# Índice

1. Introducción.....	Pág. 2
2. Glosario.....	Pág.2 a 3
3. Tema.....	Pág. 3 a 4
3.1. Motivación.....	Pág. 3
3.2. A quién va dirigida.....	Pág. 4
3.3. Alcance de la aplicación.....	Pág. 4
4. Manual Técnico.....	Pág. 5 a 9
4.1. Arquitectura de la BBDD.....	Pág. 5
4.2. Análisis.....	Pág. 6 a 8
4.3. Programación/Desarrollo.....	Pág. 9
5. Manual del Usuario.....	Pág. 10 a 15
6. Conclusión.....	Pág. 16

# 1. Introducción

Este documento recoge la documentación del proyecto de fin de curso para las asignatura de Bases de Datos y Entornos de Desarrollo.

El proyecto consiste en el diseño de una base de datos, el análisis de la aplicación y la programación.

El lenguaje de programación utilizado es Visual Basic (Lenguaje característico de Visual Studio).

SQLite es una biblioteca que implementa un Sistema de gestión de bases de datos transaccionales con el cual se ha desarrollado esta aplicación.

## 2. Glosario

Algunos términos como BBDD, Casos de Uso o Diagramas de Interacción te podrán resultar extraños si eres nuevo en el mundo de la informática.

Por ello abajo tienes una breve explicación de cada concepto.

**BBDD :** BBDD o Bases de Datos son un conjunto de información relacionada que se encuentra agrupada ó estructurada.

**Casos de Uso :** Un caso de uso es la descripción de una acción o actividad sobre un programa, como por ejemplo la acción Insertar Trabajador.

**Clases :** Una clase es una plantilla para la creación de objetos de datos según un modelo predefinido, cada clase define un conjunto de variables y métodos y cada objeto creado a partir de la clase se denomina instancia de la clase.

**Diagrama de Clases :** Un diagrama de clases en Lenguaje Unificado de Modelado es un tipo de diagrama que describe la estructura de un sistema mostrando las clases del sistema, sus atributos y funciones.

**Lenguaje Unificado de Modelado:** “Unified Modeling Language”, es el estándar que se ha adoptado para crear esquemas y diagramas en programas informáticos.

**Visual Studio :** Microsoft Visual Studio es un entorno de desarrollo integrado con el cual se ha desarrollado esta aplicación.

**CRUD :** En informática, CRUD es el acrónimo de "Crear, Leer, Actualizar y Borrar", que se usa para referirse a las funciones básicas en bases de datos.

## 3. Tema

### 3.1 Motivación :

La principal motivación que he tenido a la hora de realizar esta aplicación a parte de por mi mismo y mis estudios ha sido por mi padre.

Él trabaja en una empresa como a las que va dirigida esta aplicación y el haberle ayudado en en trabajo en algunos momentos me ha hecho interesarme como en se harían este tipo de aplicaciones.

Esta aplicación me ha servido para aprender a utilizar Visual Studio y su lenguaje de programación y como idea a una futura aplicación con el mismo objetivo pero más grande y con más funcionalidades..

### **3.2 A quién va dirigida :**

Como he comentado antes esta aplicación va dirigida para todas las empresas antiincendios, pero más concretamente para el usuario administrador o secretario/a de dicha empresa.

El objetivo de mi aplicación sería la gestión de una empresa en cuanto a pedidos, clientes y la realización de las facturas (por el momento en formato txt).

### **3.3 Alcance de la aplicación :**

Implementado:

La aplicación puede realizar operaciones básicas CRUD de Trabajadores, Almacenes, Productos, Proveedores, Clientes, Pedidos y Líneas a través de pequeños formulario.

La posibilidad de realizar búsquedas de todas las líneas de pedido de un pedido específico que introducimos en el buscador, realizar una Factura en un archivo en formato txt creando su propia carpeta y abrir ese archivo txt desde el propio programa. Ruta : "C:\Facturas\Facturas.txt"

Además la aplicación puede crear una carpeta con su base de datos en nuestro disco duro. Ruta : "C:\database\ProyectoBBDDExtintores.db".

No Implementado:

Por otro lado la aplicación no puede realizar seguimiento de el stock de los almacenes ni proveedores.

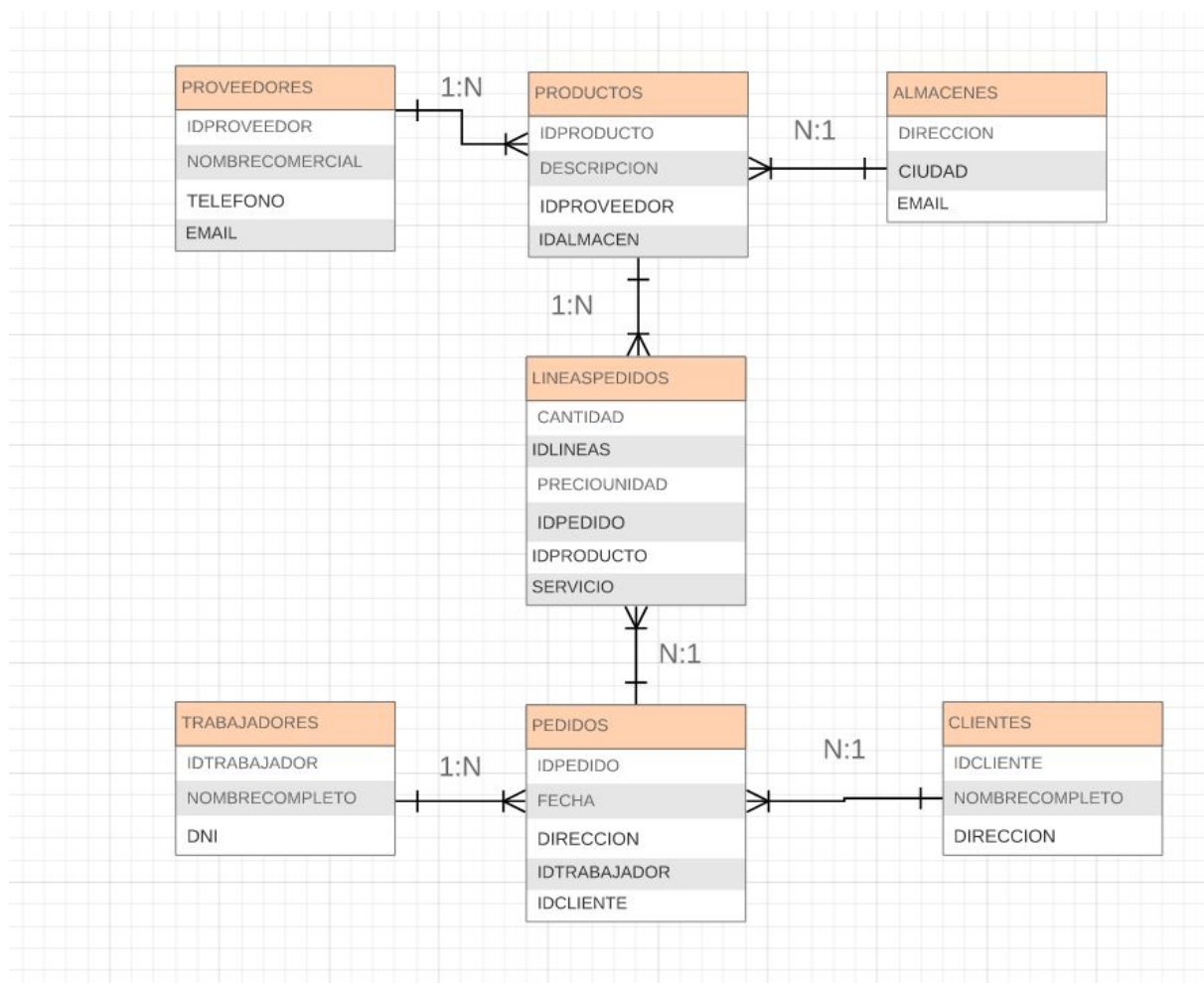
No tiene implementado un sistema que valide si los datos introducidos son correctos.

Además tampoco puede realizar nóminas a los trabajadores, son funcionalidades que me hubiera gustado introducir para que la aplicación fuese un poco más completa.

# 4. Manual Técnico

## 4.1 Arquitectura de la BBDD

En la siguiente imagen se muestra cómo está organizada la base de datos con la que trabaja la aplicación.

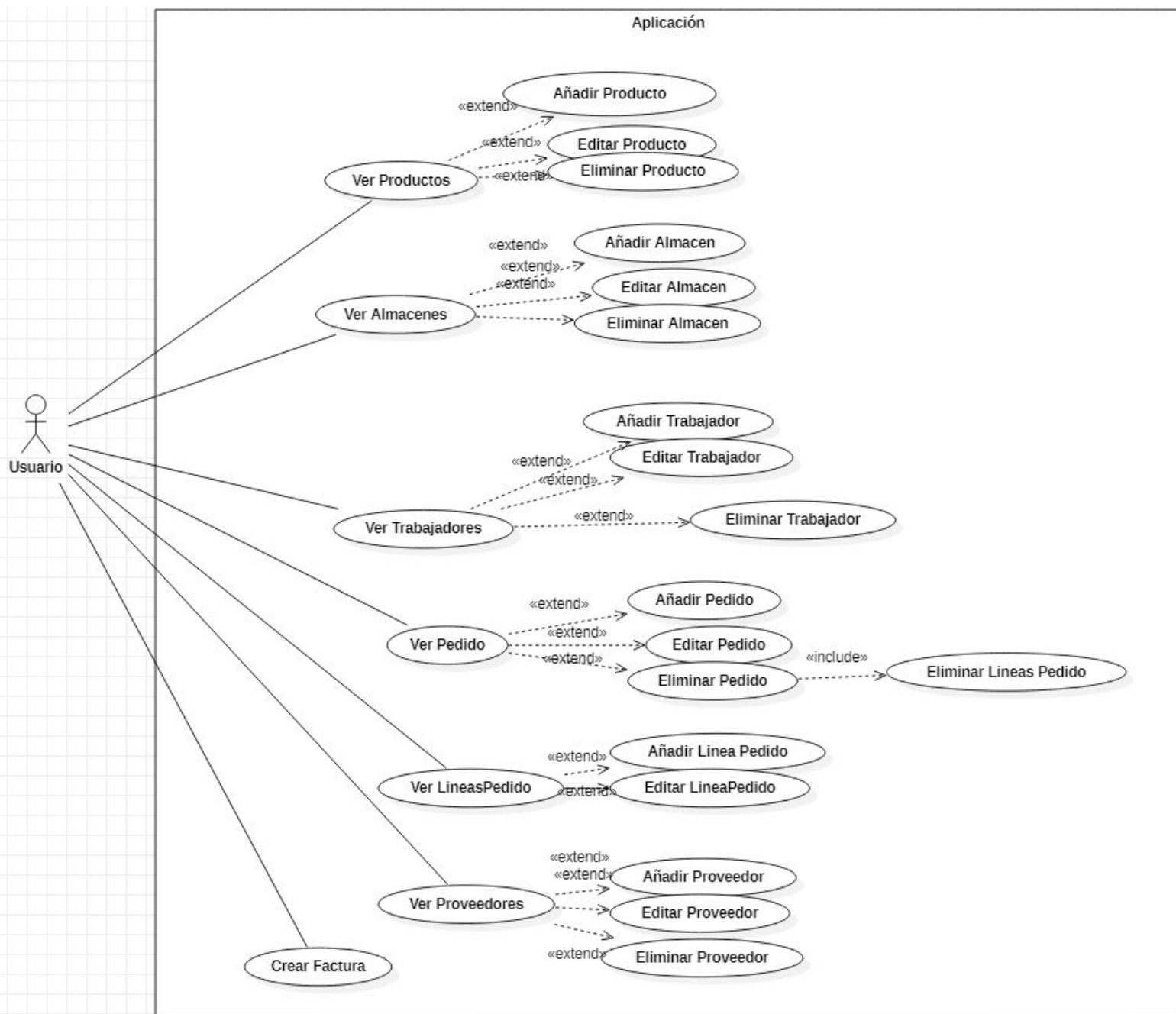


La entidad que más peso tienen sería Pedidos, ya que actualmente la aplicación se centra en la realización de facturas de los pedidos.

Como comente antes se podría desarrollar la gestión de stock de los almacenes con algunos cambios.

## 4.2 Análisis

El diagrama de casos de uso de la aplicación:



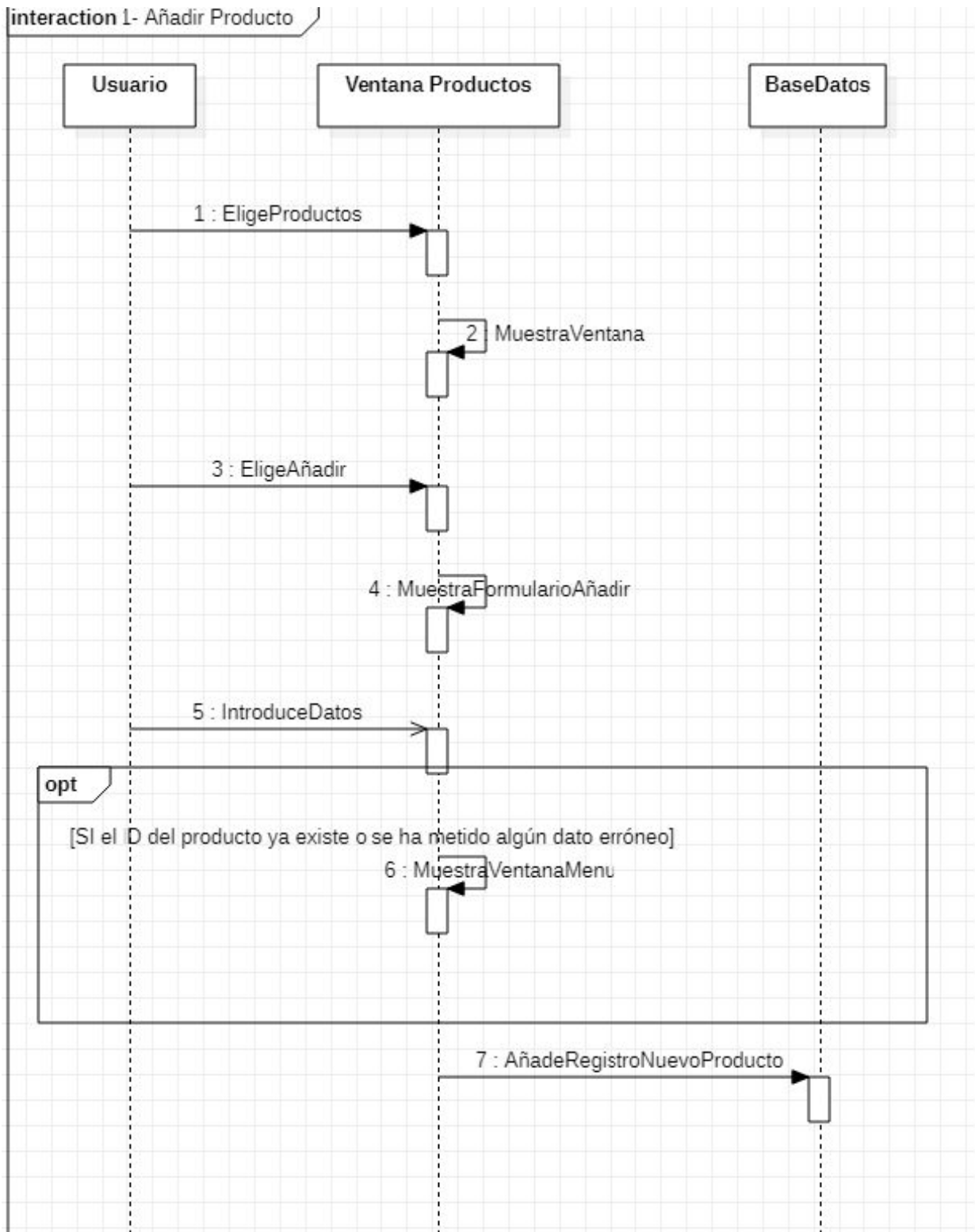
Por ejemplo la descripción de uso para el caso de uso Añadir producto sería el siguiente:

<b>Nombre: Añadir Producto</b>
<b>ID:</b> AP-01
<b>Descripción:</b>  El usuario quiere registrar un producto nuevo a la base de datos.
<b>Actores:</b> Usuario
<b>Precondiciones:</b> Ninguna
<b>Curso normal del caso de uso:</b> <ul style="list-style-type: none"><li>• El usuario inicia el programa y selecciona el botón "Productos"</li><li>• El sistema muestra la ventana de Productos.</li><li>• El usuario selecciona el botón "Añadir".</li><li>• El sistema muestra un formulario para meter los datos del Producto Nuevo.</li><li>• El usuario rellena el formulario y selecciona el botón "Añadir"</li><li>• El sistema añade el producto a la base de datos.</li></ul>
<b>Postcondiciones:</b>  Ninguna
<b>Alternativa 1:</b> <ul style="list-style-type: none"><li>• Si el ID del producto ya existe o se ha metido algún dato erróneo:<ul style="list-style-type: none"><li>• El sistema muestra un mensaje de error y vuelve al menú principal.</li></ul></li></ul>



Con su correspondiente diagrama de secuencia:

## 4.2 Programación / Desarrollo



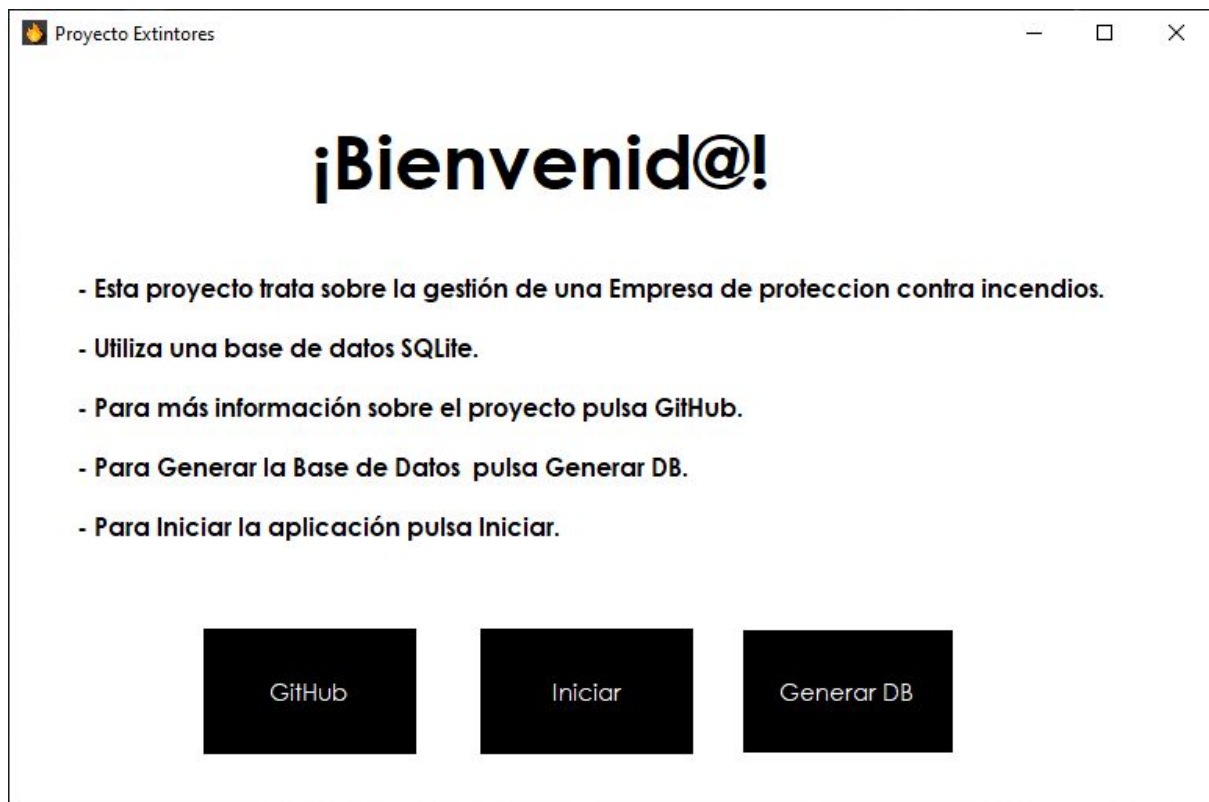
El lenguaje de programación utilizado es Visual Basic (Lenguaje característico de Visual Studio).

La arquitectura de la aplicación se basa en una clase `ControladorDatos` y una clase `CrearBD` que almacenan todas las funciones para que su uso en los formularios y su llamada sea más cómoda que creando la función en el propio formulario.

La parte más difícil para mí ha sido al principio del desarrollo de la aplicación, pero a medida que iba trabajando me desenvolvía mejor con el programa y el lenguaje.

## 6. Manual del Usuario

Al iniciar el programa nos encontraremos con esta pantalla:



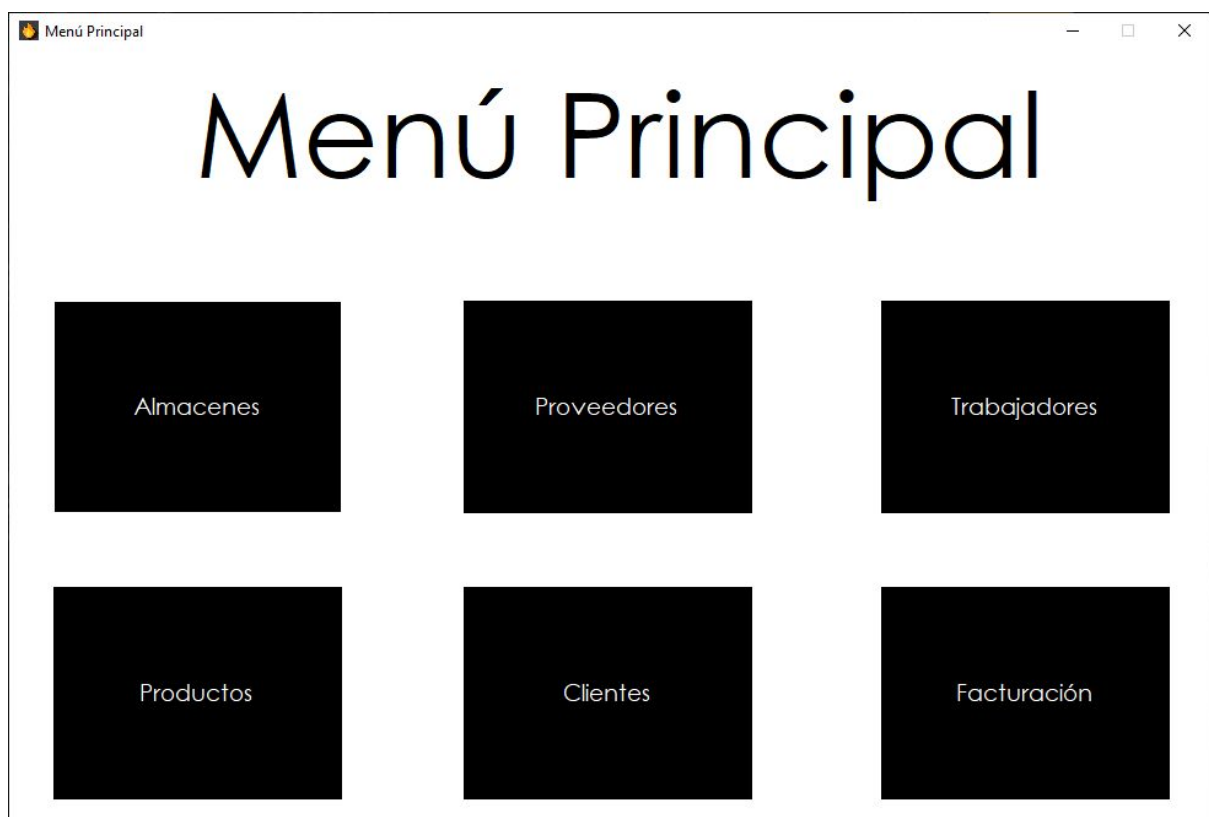
Esta pantalla no tenía pensado dejarla en el proyecto como ventana inicial, ya que pensaba en hacer una pantalla de inicio con un botón que llevará a esta otra pantalla.

En la pantalla de inicio se puede interactuar con 3 botones los cuales son “GitHub”, “Iniciar” y “Generar DB”.

Como se puede ver en la pantalla se da una breve explicación de lo que hace cada botón al pulsarlo.

Deberemos pulsar el botón “Generar DB” si es nuestra primera vez en esta aplicación para que el programa cree la carpeta “database” en C:\ y cree la base de datos dentro de esta carpeta.


A continuación pulsando el botón “Iniciar” nos muestra el siguiente menú.




Ahora tenemos 6 botones con los que interactuar, “Almacenes”, “Proveedores”, “Trabajadores”, “Productos”, “Clientes” y “Facturación”, viendo la imagen de la Base de Datos en 4.1 Arquitectura BBDD podemos hacernos una idea de lo que harán.


Pulsando cualquier botón excepto “Facturación” (el cual explicaré más adelante) nos mostrará una ventana en la que podremos realizar las operaciones CRUD hacia la base de datos.


Nos aparecerá una venta de este estilo con los datos de la tabla correspondiente al botón pulsado, probemos con “Proveedores”:


 Gestión Proveedores

	IDPROVEEDOR	NOMBRECOMERCIAL	TELEFONO	EMAIL
▶	PRO001	EXTFIRE	935948604	extfire@gmail.com
	PRO002	EXTFLAME	913123104	extflame@gmail.com
*				

 Actualizar Datos

 Añadir

 Editar

 Eliminar

Como vemos nos muestra los datos de la tabla correspondiente y una serie de botones con los que podremos realizar las operaciones CRUD.

Además de un botón “Actualizar Datos” para mostrar la tablas actualizada después de realizar cualquier operación.

Pulsamos “Añadir” y nos aparecerá el siguiente formulario junto al Gestión de proveedores :



Añadir Proveedor

### Nuevo Proveedor

IDPROVEEDOR

Nombre Comercial

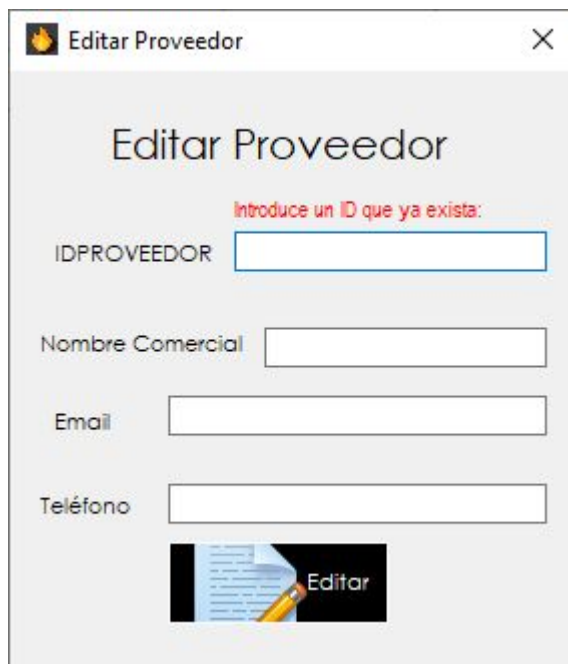
Email

Teléfono

  Añadir

Rellenamos los datos y pulsamos “Añadir”, si los datos introducidos son correctos al pulsar “Actualizar Datos” en la ventana de Gestión de proveedores nos aparecerá el registro.

Pulsamos ahora “Editar” y nos aparecerá el siguiente formulario junto al Gestión de proveedores :



Editar Proveedor



Introduce un ID que ya exista:

IDPROVEEDOR

Nombre Comercial

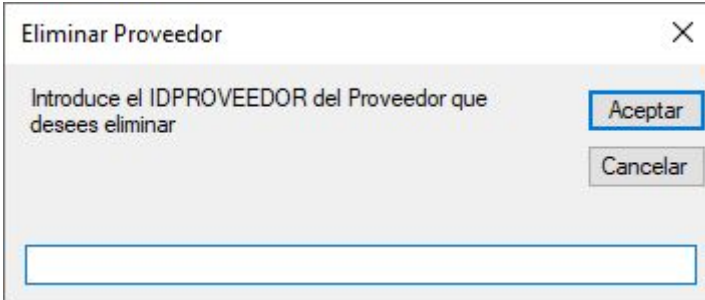
Email

Teléfono

  Editar

Rellenamos los datos y pulsamos “Editar”, si los datos introducidos son correctos al pulsar “Actualizar Datos” en la ventana de Gestión de proveedores se nos editará el registro.

Pulsamos ahora “Eliminar” y nos aparecerá el siguiente formulario junto al Gestión de proveedores :



Eliminar Proveedor

Introduce el IDPROVEEDOR del Proveedor que desees eliminar

Aceptar

Cancelar

Introducimos un Id de proveedor que ya existe y lo borrara de la base de datos.

Todos los botones del Menú Principal realizan la misma operación acorde a el botón pulsado.

Por ejemplo: Si seleccionamos “Almacenes” podremos gestionar la tabla Almacenes de nuestra base de datos, si seleccionamos “Proveedores” gestionaremos la tabla Proveedores de nuestra base de datos, etc...

Ahora hablando del botón del Menú Principal llamado “Facturación”, al pulsarlo nos aparece este Sub-Menú.



Gestión Pedidos

## Menú Gestión Pedidos

Pedidos

Productos Pedidos

Crear Factura

Abrir Factura

Si pulsamos “Pedidos” o “Productos pedidos” podemos gestionar operaciones CRUD de sus respectivas tablas en la base de datos Pedidos y Pedidos Líneas.

Además en “Productos pedidos” dispondremos de un buscador por IDPEDIDO para ver más claro las líneas de un Pedido.

	IDLINEAS	CANTIDAD	PRECIOUNIDAD	IDPEDIDO	IDPRODUCTO	SERVICIO
▶	LIN001	30	6	PED001	EXTPOLVO	REV
	LIN002	23	6	PED002	EXTCO2	VEN
*						

El botón “Crear Factura” nos preguntará el ID del pedido que queramos crear la factura y creará el documento y la carpeta Facturas en la siguiente ruta “C:\Factura\Facturas.txt”.

Facturacion Pedido

Introduce el IDPEDIDO del Pedido que desees facturar

Aceptar

Cancelar

Y por último el botón “Abrir Factura” nos abre el archivo creado donde se escriben las facturas.

Facturas: Bloc de notas

Archivo Edición Formato Ver Ayuda

---

Datos de la Factura

Factura con IDPedido: PED001

Fecha de Facturación: 28/05/2020

Datos Del Cliente

Nombre del Cliente: JESUS ALVAREZ MARTINEZ

Dirección del Cliente: C/Miranda Irun

Datos del Trabajador

Id del Trabajador: REV001

Atendido por: JUAN GONZALEZ MONTALVO

IDPRODUCTO	CANTIDAD	PRECIO	SERVICIO	SUBTOTAL
ETXPOLVO	30	6	REV	180

Total A Pagar: 180 Euros

---

Datos de la Factura

Factura con IDPedido: PED002

Fecha de Facturación: 28/05/2020

Datos Del Cliente

Nombre del Cliente: JOSE PEREZ VILLOSA

Ln 1, Col 1 100% Windows (CRLF) UTF-8



# 6. Conclusión

Me ha gustado el crear la aplicación, sobre todo el diseñar los formularios más que la parte de programación el proyecto aunque se que podría haberlo hecho mucho mejor de lo que es actualmente.

Me hubiera gustado implementar todo lo que no está implementado en 3.3 Alcance de la Aplicación antes de presentarlo aunque es fallo mio al organizarse mal el tiempo pensando que lo haría más rápido, pero me ha costado coger ritmo.