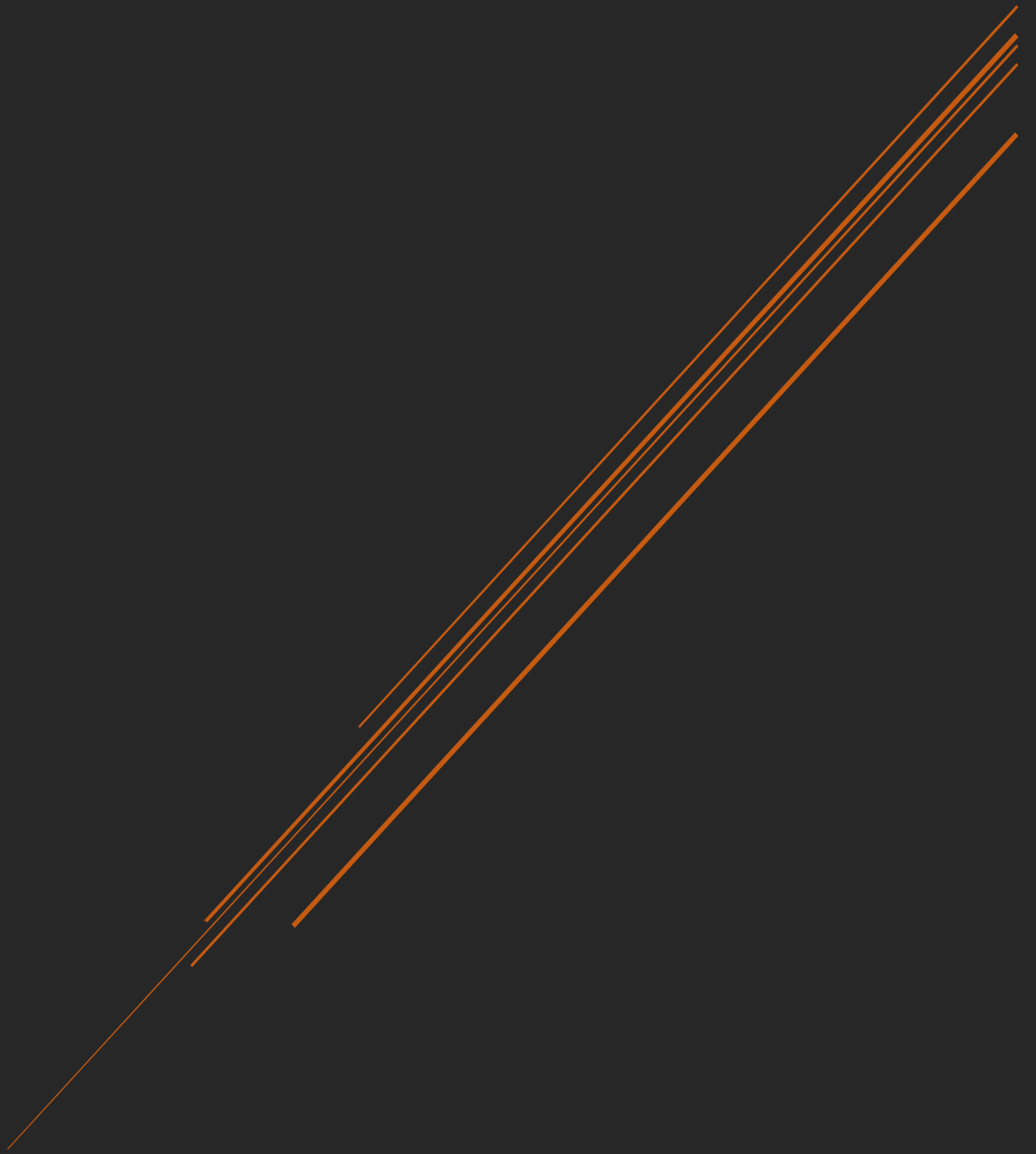


E MANAGEMENT Z

La aplicación de gestión para todos.



Adrián García de la Fuente

ÍNDICE

INTRODUCCIÓN A E MANAGEMENT Z	2
LICENCIA DE LA APLICACIÓN.....	2
DESCRIPCIÓN DE LA APLICACIÓN.....	2
DESARROLLO DE LA APLICACIÓN	3
DIAGRAMA DE GRANT	4
ESTRUCTURA DE LA APLICACIÓN	5
ESTRUCTURA DE LA BASE DE DATOS	6
INDICE DE MERCADO	6
FUNCION A NIVEL DE CODIGO DE LAS VENTANAS.....	8
FUTURAS MEJORAS Y PROBLEMAS ENCONTRADOS	11
BIBLIOGRAFÍA	12

INTRODUCCIÓN A E MANAGEMENT Z

E Management Z es una aplicación de gestión pensada para todos. El nombre de E Management Z es un juego de palabras, que mezcla la abreviatura inglesa "EZ" (Que significa fácil, normalmente en juegos), haciendo que el nombre sea una referencia a una gestión fácil.

La idea de la aplicación es hacer que la gente que tiene un negocio muy pequeño o que gente que hace su propio negocio de forma autónoma, pueda gestionar de manera fácil los datos de sus clientes, proveedores o productos. Con esto, no es necesario que tengan que lidiar con aplicaciones de gestión que tienen miles de funciones que nunca se usan.

Además, con esta aplicación podemos crear pequeñas en texto plano, fácilmente visualizables en casi cualquier dispositivo, lo cual lo hace perfecto para compartir con cualquier persona y que esta la pueda ver.

LICENCIA DE LA APLICACIÓN

Esta aplicación es una app privada. Está prohibida su distribución, su modificación y por supuesto su venta. Cualquier persona que incumpla estas condiciones esta expuesta a acciones judiciales.

DESCRIPCIÓN DE LA APLICACIÓN

E Management Z es una aplicación de gestión, que consta de una base de datos propia y todas las ventanas de gestión incluidas en la aplicación.

Para comenzar tenemos un login donde entrar a la app, además de los correspondientes apartados de registro de usuarios y recuperación y cambio de contraseñas. Desde ahí el usuario podrá identificarse para poder usar la aplicación o podrá registrarse y/o reestablecer sus contraseñas.

Además, el sistema de recuperación de la contraseña se realiza mediante un código generado en la aplicación que se envía por email al usuario en cuestión.

Contamos también con varias ventanas, que podemos dividir en 5:

- Ventanas de Proveedores, desde las cuales podremos crear, visualizar o eliminar proveedores a nuestra base de datos.
- Ventanas de Clientes, desde las cuales se podrán crear, visualizar y eliminar clientes.
- Ventanas de Productos, desde las cuales se podrán crear, visualizar y eliminar productos. Además, estas cuentan con la posibilidad de añadir una imagen para la fácil identificación del producto.
- Ventana de Facturas, la cual con los datos que añadamos en las otras ventanas podremos generar un documento de texto con los datos para dárselo al cliente o a quien nos lo pida
- Ventana de usuarios, la cual es solo para usuarios administradores y que sirve para manejar todo lo relacionado con los usuarios de la aplicación.

Para saber más sobre las ventanas y su uso, se puede ver su funcionamiento en el PDF llamado "Instrucciones de EmZ" en la carpeta de la aplicación.

DESARROLLO DE LA APLICACIÓN

La aplicación está enteramente programada en Python excepto la base de datos, la cual está creada en SQLite. Puesto que Python de por sí no es un lenguaje específicamente creado para programar interfaces gráficas, así que por ello se han usado varios módulos que han ayudado en el proceso.

El módulo principal ha sido Tkinter, un módulo de gran tamaño encargado de la creación de toda la parte visual, que lo complementa PySimpleGUI, el cual se encarga de coger lo que hace Tkinter y convertirlo a funciones para hacer su uso más fácil.

El siguiente modulo que se ha utilizado ha sido Pandas. Este módulo es el encargado del trabajo con grandes cantidades de datos, en este caso, los añadidos y recuperados de la base de datos. Con este módulo se pueden realizar tambien todo tipo de consultas a las BBDD, además de mostrar todo en forma de tablas y permitiéndonos manejar de forma mucho más fácil esos datos. Todo, desde añadir, filtrar, modificar o eliminar se puede realizar con este módulo.

Por último, pero no menos importante, tenemos Pillow. Este pequeño modulo es el encargado de que en las ventanas de productos podamos mostrar fotografías. Este se encarga de la apertura, codificación y mostrado de las imágenes.

Para saber más sobre estos módulos se dejan los enlaces a estos:

Pandas: <https://pandas.pydata.org/>

PySimpleGUI: <https://www.pysimplegui.org/en/latest/>

Pillow: <https://pillow.readthedocs.io/en/stable/>

DIAGRAMA DE GRANT

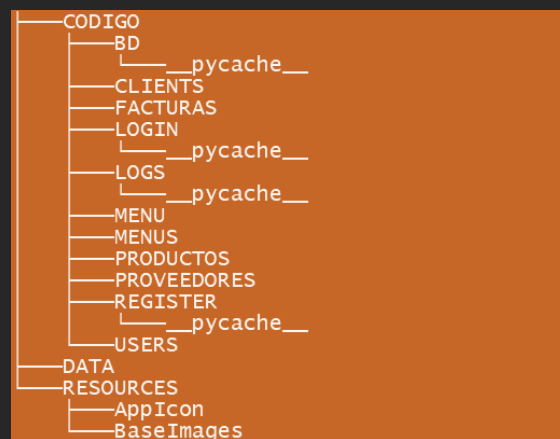
Aquí se puede ver un pequeño diagrama en base a 2 meses, donde podemos ver como ha sido de una forma aproximada el desarrollo de esta aplicación. Cabe recordar que esto tiene un fin orientativo, pero más o menos este podría ser un resumen de cómo ha sido el desarrollo de la aplicación

Actividad / Día	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
Investigación de mercado	✓	✓	✓	✓																
Planificación de la aplicación				✓	✓															
Plan de mercado					✓	✓	✓													
Diseño							✓	✓												
Desarrollo de la aplicación								✓	✓	✓	✓	✓	✓	✓	✓	✓				
Testing de la app																	✓	✓	✓	
Documentación de la app																	✓	✓	✓	✓

ESTRUCTURA DE LA APLICACIÓN

La estructura del código de la aplicación se divide en 3 partes:

- CODIGO
- DATOS
- RECURSOS



La carpeta de CODIGO es la carpeta donde se encuentra la parte importante de la aplicación. Aquí podemos encontrar tanto la base de datos como todas las ventanas de la app. Cada ventana esta almacenada en su correspondiente carpeta.

La carpeta de DATA es la carpeta en la que se almacenan las imágenes que nosotros seleccionamos para los productos. Esta carpeta esta únicamente destinada a este fin.

La carpeta RESOURCES es la carpeta donde se almacenan todos los recursos de imagen que utiliza la aplicación, como por ejemplo el icono de esta o las imágenes base de la ventana de productos.

En esta parte no se incluyen los que, por supuesto son tambien necesarios, archivos de la compilación en EXE y todo lo necesario para que este funcione.

ESTRUCTURA DE LA BASE DE DATOS

La base de datos esta creada en SQLite. Esto es debido a la fácil integración que tiene con Python, lo que hace que la podamos generar incluso por código. El archivo de la BBDD va integrado en la propia aplicación, en la carpeta BD que podemos ver en el apartado de estructura de archivos de la aplicación.

Aunque en la BBDD hay 6 tablas, actualmente se usan 4. Esto es debido a una futura actualización que se tiene en mente.

Las tablas son las siguientes:

- Usuarios: En la cual se almacena todo lo relacionado con los usuarios de la aplicación
- Clientes: Tabla en la que se almacena todos los datos de los clientes, que serán utilizados después para la creación de las facturas
- Productos: Tabla en la que se almacena toda la información de los productos, que serán utilizados después para la creación de las facturas
- Proveedores: Tabla en la que se almacena todos los datos de los proveedores, que serán utilizados después para la creación de las facturas

Además de estas tablas, hay una tabla la cual se ha explicado que esta creada para una futura actualización, y es que, en un futuro, nos gustaría crear una estructura Cliente – Servidor haciendo un control de acceso, el cual se almacenaría en la base de datos, de ahí que haya una tabla llamada INFORMACIÓN

INDICE DE MERCADO

¿A quién va dirigida la aplicación?

Esta aplicación va dirigida a cualquier persona que quiera tener un control de su negocio. La principal virtud de esta app es que sea fácil de usar y que cualquier persona que sepa encender un ordenador pueda usarla.

¿Tiene cabida esta aplicación en el mercado?

Si. Aunque el mercado este lleno de aplicaciones similares, y muchas veces iguales, creemos que esta aplicación puede tener futuro. Como hablamos anteriormente, muchas de esas aplicaciones que podemos encontrar a golpe de clic vienen cargadas de cientos y a veces miles de funciones que, para un particular o una pequeña empresa normalmente son completamente prescindibles.

No queremos competir con las empresas grandes o punteras del sector, porque sabemos las limitaciones de nuestra app, pero queremos ser esa aplicación que cuando una persona diga: - *"-Quiero algo fácil de usar"* piensen en nosotros y nos recomienden.

¿Qué precio de venta tiene esta aplicación?

Hemos calculado que, por el tiempo de horas de desarrollo de la misma, y más o menos viendo al precio al que están el resto de aplicaciones el precio de venta de esta aplicación será de 2200€.

A medida que vayamos añadiendo funcionalidades o veamos que obtenemos suficientes beneficios podremos plantearnos o subir o bajar el precio. Pero inicialmente hemos determinado esa cantidad.

Quizás incluso, en un futuro, teniendo en cuenta como es el mercado, podríamos plantearnos el crear una versión *freemium*, dando características gratuitas y con varias de pago.

¿Cómo será el proceso de distribución de la aplicación?

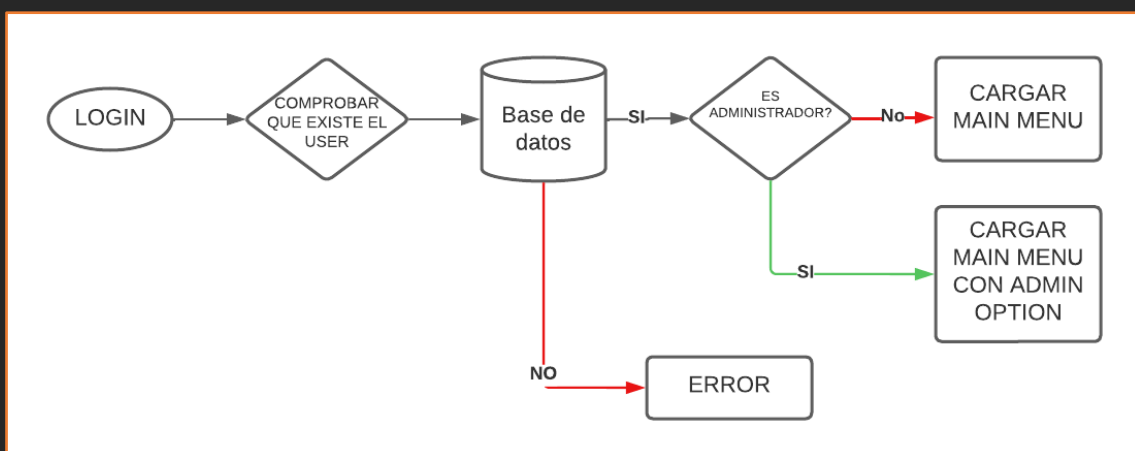
Actualmente, y teniendo en cuenta que nuestra aplicación va dirigida la app, la distribución se hará de forma manual, es decir, a cada cliente de nuestra app se le enviara el programa.

La idea futura es que cuando tengamos nuestra estructura cliente – servidor, si la aplicación pasa a un formato freemium, que esta esté alojada en algún repositorio público o en un servidor web, de modo que no haga falta instalación de nada

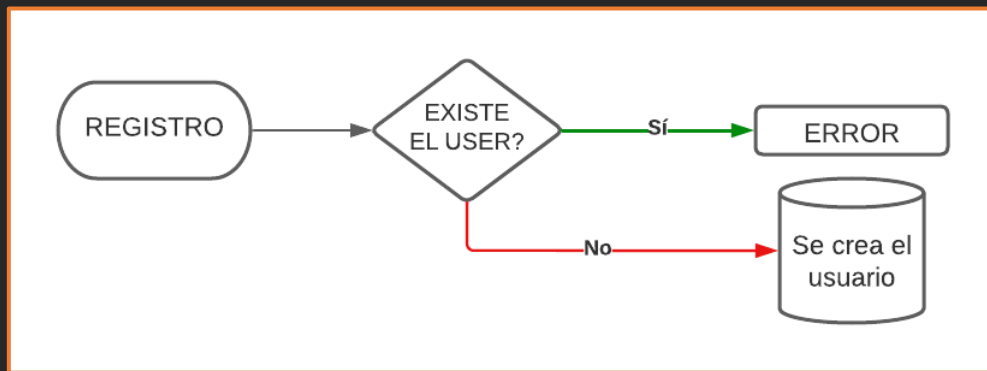
FUNCION A NIVEL DE CODIGO DE LAS VENTANAS

Aunque esta aplicación solo se distribuye, y en ningún caso puede ser modificada por otra persona, en el caso de que alguien quiera saber como funciona a nivel interno la app, aquí se realiza un pequeño resumen de algunas de las ventanas de la aplicación.

La ventana de Login tiene un funcionamiento bastante simple. En esta, una vez cubiertos los campos, en el momento de hacer el inicio de sesión, la aplicación hace una conexión a la base de datos para saber que ese usuario existe con esa contraseña. En caso de no existir, el inicio de sesión no se podrá realizar. Además, tambien se comprueba que si al iniciar sesión el usuario se administrador o no. Aquí se ve un pequeño esquema:

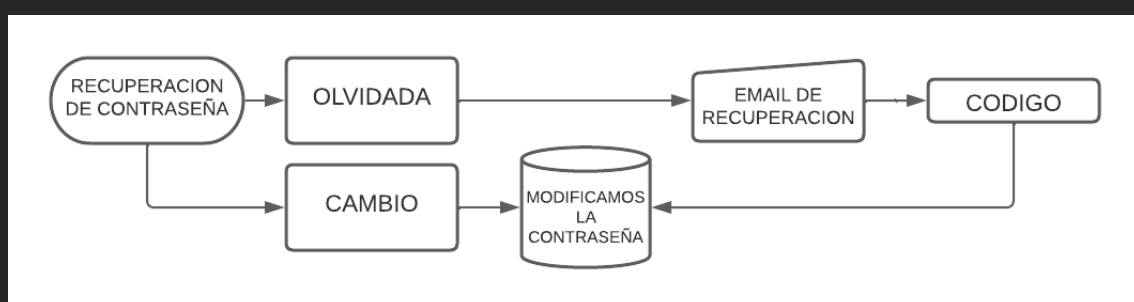


Para la ventana de registro es mucho mas sencillo. Se intenta realizar un registro. Para ello, primero se cogen los datos de registro y se comprueba que el usuario no exista previamente en la base de datos. En el caso de existir, no se podrá realizar el añadido. En caso de no existir, se realiza el añadido sin ningún problema



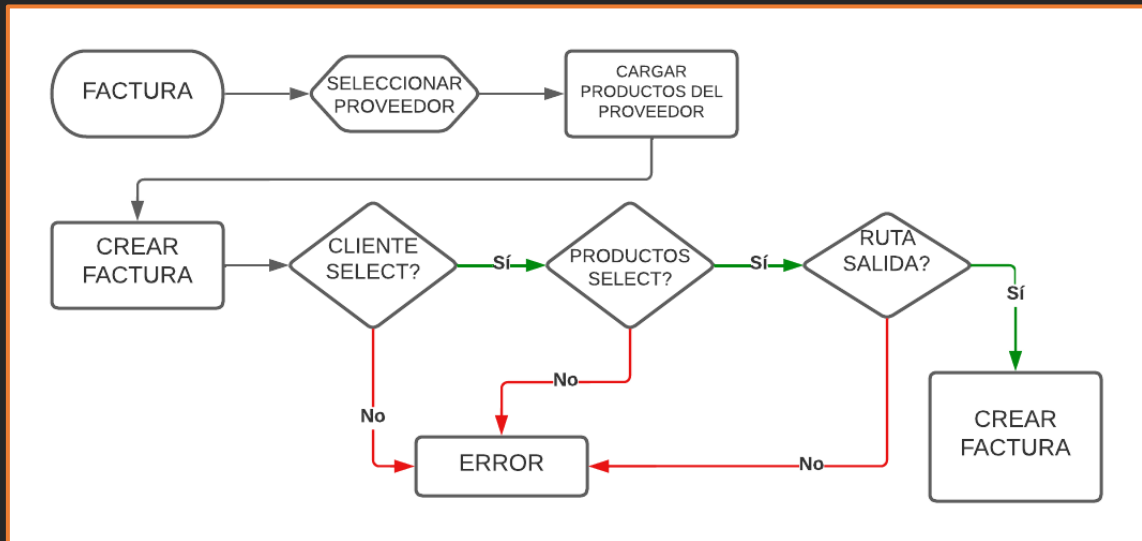
Para la recuperación de la contraseña cambia un poco. Tenemos que diferenciar entre la opción de cambiar la contraseña o recuperar la contraseña. En caso de ser la opción de contraseña olvidada, tenemos que coger los datos y comprobar que existen, y entonces realizar el envío del email con el código. Después, con los datos nuevos hacemos el cambio de contraseña.

Para cambiarla es mas fácil, los datos viejos, los nuevos y hacer el cambio en la BBDD

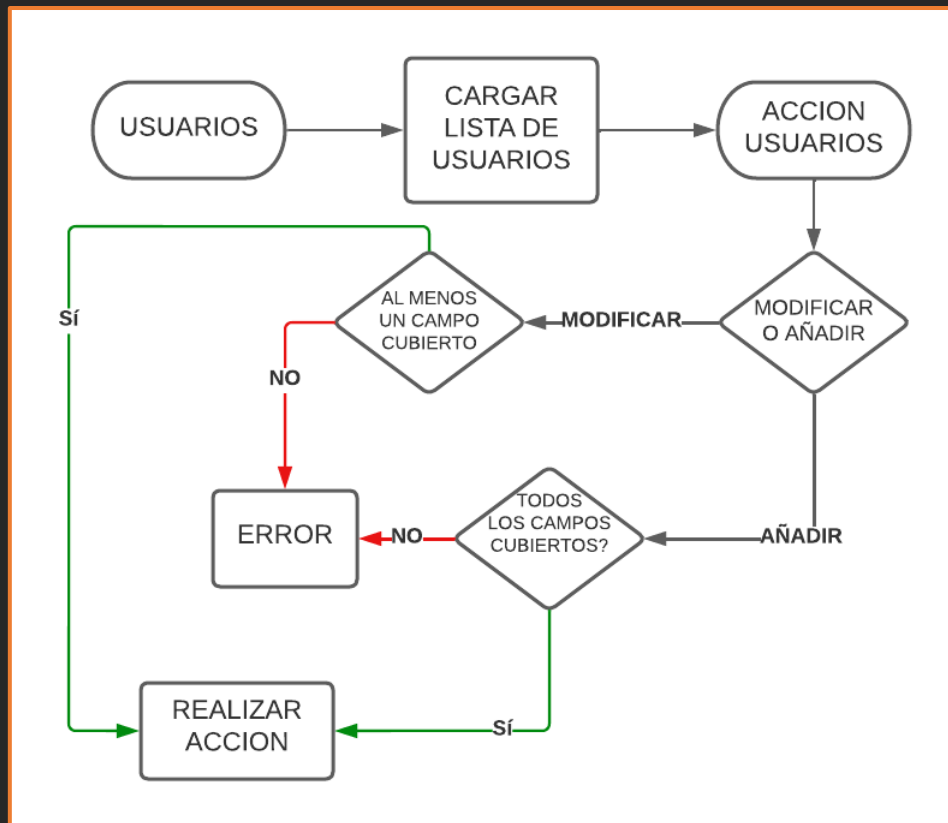


Para la ventana de facturas tenemos que tener varios factores en cuenta. Primero tenemos que seleccionar un proveedor, con lo que haremos que se generen los datos para la tabla de productos que vayan asociados a ese proveedor. Una vez seleccionado lo que queremos, la aplicación hace una comprobación de que haya un cliente seleccionado, haya al menos un

producto seleccionado y que haya ruta de salida del archivo seleccionada. Si todo eso es correcto, se generará una factura. Aquí se puede ver un pequeño esquema.



La ventana de usuarios tiene un funcionamiento especial. Primero tenemos una lista con todos los usuarios de la aplicación. En el momento de realizar la acción de modificar/añadir tenemos que tener en cuenta cosas diferentes. En caso de añadir, tenemos que comprobar que todos los campos estén cubiertos para añadir el usuario, pero en caso de modificar, hay que tener en cuenta que al menos un campo esté cubierto y tengamos un usuario seleccionado en la tabla. Si no, nos devolverá un mensaje de error



Las ventanas de añadir son todos iguales, se cogen los campos y se intenta realizar el insertado en la base de datos, al igual que las de mostrar, que con un Select se recuperan los datos y se muestran por pantalla. Para la ventana de eliminado, se muestra una tabla de la que se podrán borrar uno o todos los datos que aparecen.

En la tabla de proveedores, para poder borrar un proveedor, se comprueba que no haya ningún producto asociado, si no, no se podrá borrar.

FUTURAS MEJORAS Y PROBLEMAS ENCONTRADOS

Esta aplicación ha sido fácil de programar en el aspecto de la estructura. El mayor de los problemas que se ha encontrado en el proceso de creación es el de la distribución de la misma. Al ser un lenguaje que no está pensado para las interfaces gráficas, el hecho de compilar el proyecto y enviarlo como un único

EXE no ha sido posible, de ahí que se adjunten las instrucciones de instalación de la propia aplicación.

En un futuro sería bueno que la aplicación pasase a un modelo cliente – servidor de forma que los datos estuviesen en un sistema central y cada cliente accediese de forma remota.

Esto implicaría también el cambio de lenguaje de la aplicación, seguramente a un formato de aplicación web. Esto haría todo mucho más sencillo, tanto para la programación como para la distribución, aparte de que es un lenguaje al alza.

BIBLIOGRAFÍA

Los recursos que se han usado para la creación de esta aplicación han sido obtenidos desde:

<https://stackoverflow.com/>

<https://www.python.org/doc/>

<https://pandas.pydata.org/>

<https://www.pysimplegui.org/en/latest/>

<https://www.reddit.com/r/Python/>

Cursos formativos en <https://www.viewnext.com/>

<https://www.flaticon.es/>

<https://www.freepik.es/>