



MÁSTER UNIVERSITARIO EN INGENIERÍA INDUSTRIAL

TRABAJO FIN DE MÁSTER ELABORACIÓN DE HERRAMIENTA DE CONTROL DE INVENTARIO EN UNA START-UP

Autor: Manuel Puerto Baydal
Director: Ignacio Tornos de Inza

Madrid

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título
Elaboración de herramienta de control de inventario en una start-up
en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el
curso académico **2021/22** es de mi autoría, original e inédito y
no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido
tomada de otros documentos está debidamente referenciada.



Fdo.: Manuel Puerto Baydal

Fecha: 31/ 08 / 2022

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Ignacio Tornos de Inza

Fecha: 31/ 08 / 2022

Agradecimientos

Les doy las gracias a mi familia y amigos cercanos por todo el apoyo prestado durante la realización de este trabajo, especialmente a mi madre por su paciencia y ayuda. También deseo agradecer a mi tutor, Ignacio, por ejercer de guía a lo largo de este proceso y descubrirme las posibilidades que la tecnología ofrece. Finalmente, quiero dar las gracias a la empresa, sobre todo a Pedro y Jaime por sus consejos.

ELABORACIÓN DE HERRAMIENTA DE CONTROL DE INVENTARIO EN UNA START-UP

Autor: Puerto Baydal, Manuel.

Director: Tornos de Inza, Ignacio.

Entidad Colaboradora: ICAI – Universidad Pontificia Comillas

RESUMEN DEL PROYECTO

El desarrollo de este proyecto consiste en el diseño e implementación de un sistema de control de inventario, para su uso en una startup, haciendo uso de una base de datos SQL y un robot, que automatice las principales actividades relacionadas con el funcionamiento de la herramienta mediante RPA. Una vez finalizada, la herramienta es capaz de registrar los productos, anotar las compras y las ventas, actualizar los niveles de inventario de los productos y tomar decisiones en función de esos niveles.

Palabras clave: Base de datos, Automatización robótica de procesos, RPA, SQL, Inventario, Almacén, Control de inventario

1. Introducción

El presente trabajo trata sobre la elaboración de una herramienta de control de inventario para supervisar el funcionamiento del almacén de una empresa. El trabajo se ha desarrollado en colaboración con la misma. Se trata de una startup, una empresa joven, que se dedica a operar un *marketplace*, donde diferentes distribuidores se dedican a vender sus productos.

La empresa está planteando empezar a distribuir sus propios productos en el futuro. Para ello, necesitará su propio almacén, donde guardar los productos, y una herramienta para su gestión, que controle las entradas y salidas de material del almacén. En colaboración con el área de operaciones de la empresa, se ha definido la herramienta a utilizar, que constará de una base de datos SQL y un robot RPA [1].

2. Definición del proyecto

El orden que se ha seguido en el desarrollo de este proyecto es el mostrado en la Ilustración 1. Como se puede observar, se realizó, en primer lugar, una reunión con la empresa para conocer exactamente sus necesidades, el uso que se le iba a dar a la herramienta y qué idea tenían ellos previamente de lo que podían necesitar. A continuación, se estudiaron las posibles alternativas, viendo qué solución encajaba mejor para el caso planteado. La opción elegida fue crear una base de datos a medida combinada con un robot RPA, que recogiera toda la información que la empresa considerase crucial, y que fuera una solución escalable a medida que el volumen de pedidos crezca, sin suponer un gran desembolso para la startup. Se eligió utilizar Microsoft Azure para albergar la base de datos, puesto que la empresa ya lo utiliza para otros usos, por lo que no le supone un desembolso adicional. Algo que sí ocurriría si, por ejemplo, se decidiera contratar algún otro tipo de software adicional, como puede ser un ERP. Una vez decidida la solución a implementar, se creó la base de datos, con las distintas tablas consideradas de interés. Después de eso se diseñaron y programaron los diferentes flujos que automatizan las tareas de registro de información en la base de datos

asociadas a la entrada y salida de material del almacén. Finalmente, cuando estuvo todo implementado, realizaron diferentes pruebas para comprobar el correcto funcionamiento del sistema y se mostraron los resultados a los responsables de la empresa.

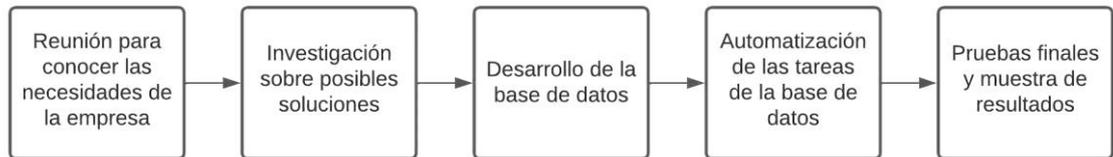


Ilustración 1. Orden de las actividades seguidas para realizar el trabajo

Los principales objetivos de este trabajo han sido:

- Crear una base de datos con SQL que permita albergar toda la información necesaria del inventario de productos de la empresa.
- Programar un robot usando RPA que automatice el proceso de la gestión del inventario cuando entren los pedidos.
- Definir qué métricas se usarán para medir el correcto funcionamiento del control de inventario.

3. Descripción del sistema

El sistema finalmente diseñado constará de dos partes. Por un lado, la base de datos albergada en *Microsoft Azure* [2]. Al estar almacenada en la nube permite a los usuarios poder acceder a la información independientemente de donde se encuentren. Por el otro lado, los flujos de automatización de las acciones de escritorio diseñados utilizando *Power Automate Desktop*, dado que al realizarse todo en un entorno Microsoft facilita la comunicación entre los distintos programas utilizados.

La base de datos creada consta de tres tablas diferentes, relacionadas entre sí por las referencias de los productos. La primera tabla, Control de Inventario, contendrá las referencias de los productos, su nivel de inventario y su estado, que viene determinado por el nivel de inventario. La segunda tabla, Compras y Ventas, registrará las entradas y salidas de productos del almacén, conteniendo los datos de los pedidos como el número de pedido, la referencia de los productos, las unidades, y la fecha de realización del pedido. Considerándose como compras, las adquisiciones de material que haga la empresa para reponer el inventario de productos en el almacén, y como ventas, las ventas de productos que se hagan a través de su plataforma a los clientes. La tercera tabla, Maestro de Artículos, guardará los datos de los diferentes productos con los que se trabaja. Estos datos incluirán la referencia asignada por la empresa, el nombre, el precio, el distribuidor, el fabricante y la referencia del fabricante, que puede ser distinta de la referencia interna. Los atributos en las tablas permitirán calcular métricas de interés como puede ser los tiempos de envío, para detectar anomalías, la rotación de inventario o el stock de seguridad.

Antes de empezar a operar con un producto es necesario registrarlo en la tabla Maestro de Artículos y en la tabla Control de Inventario. A partir de ese momento, los pedidos de

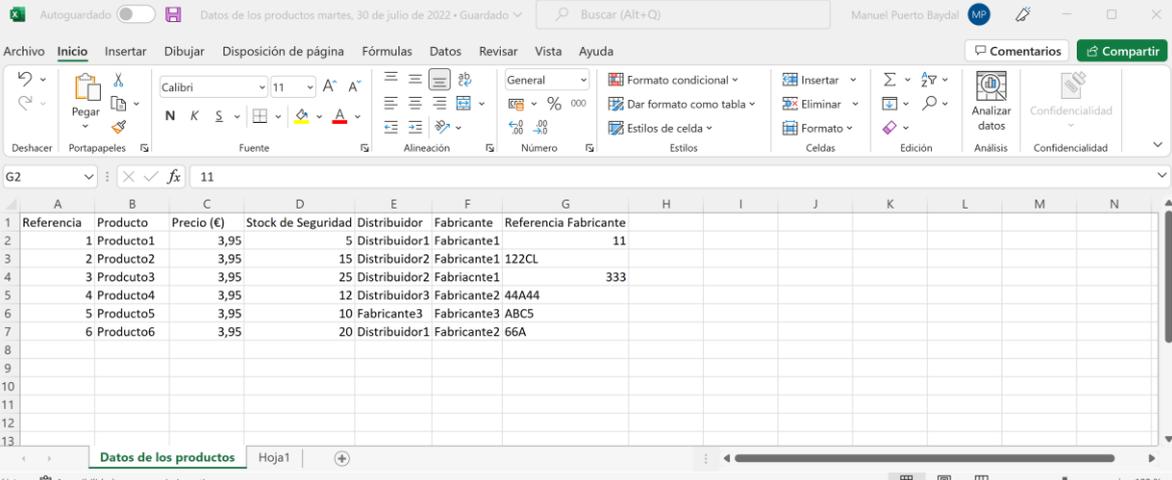
ese producto que entren se registrarán en la tabla Compras y Ventas, y se irá variando el nivel de inventario de manera acorde en el Control de Inventario.

Para poder automatizar las tareas descritas en el párrafo anterior, se han diseñado distintos flujos en *Power Automate Desktop* que permitan al sistema liberar al usuario de tener que realizarlas [3]. Cada flujo está diseñado para realizar diferentes tareas, siendo las principales:

- Adición de nuevos productos a la base de datos.
- Anotación de los registros de nuevas compras y nuevas ventas y actualización del inventario.
- Recepción y redacción de correos.
- Actualización de los registros de las tablas en función de la información recibida y los cálculos realizados.
- Visualización de datos, que permitan evaluar el rendimiento del almacén y generar diferentes análisis.

4. Resultados

Una vez implementado todo el sistema la herramienta está lista para empezar a operar. Se ha comprobado su correcto funcionamiento mediante el uso de algunos ejemplos, como el mostrado en la Ilustración 2.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Referencia	Producto	Precio (€)	Stock de Seguridad	Distribuidor	Fabricante	Referencia Fabricante							
2	1	Producto1	3,95	5	Distribuidor1	Fabricante1	11							
3	2	Producto2	3,95	15	Distribuidor2	Fabricante1	122CL							
4	3	Producto3	3,95	25	Distribuidor2	Fabricante1	333							
5	4	Producto4	3,95	12	Distribuidor3	Fabricante2	44A44							
6	5	Producto5	3,95	10	Fabricante3	Fabricante3	ABC5							
7	6	Producto6	3,95	20	Distribuidor1	Fabricante2	66A							
8														
9														
10														
11														
12														
13														

Ilustración 2. Datos extraídos de la tabla Maestro de Artículos

Como se puede observar, se han añadido diferentes productos a la base de datos, productos con diferentes tipos de datos, mediante el uso de un flujo. Posteriormente, se ha utilizado otro flujo para obtener la información de la base de datos y plasmarla en una hoja Excel. Verificando así que los datos se han registrado correctamente en la base de datos y que se pueden extraer con facilidad.

5. Conclusiones

La realización de este proyecto ha servido para que el alumno pueda aplicar los conocimientos adquiridos durante el transcurso del máster, así como adquirir nuevos en relación a SQL y RPA, dos tecnologías muy útiles en la actualidad y que cada día cobran

mayor importancia en la industria. Además, ha podido involucrarse en la actividad diaria de la empresa y familiarizarse con las decisiones estratégicas relacionadas con el objetivo del proyecto.

Por último, hay que considerar las actividades que han quedado fuera de la realización de este trabajo, pero que podrían ser de interés si se deseara mejorar la herramienta. Por ejemplo, la inclusión de flujos *DPA* que permitan al sistema trabajar de manera completamente autónoma.

6. Referencias

- [1] C. A. Bermúdez Irreño, «RPA - AUTOMATIZACIÓN ROBÓTICA DE PROCESOS: UNA REVISIÓN DE LA LITERATURA,» Revista Ingeniería, Matemáticas y Ciencias de la Información, vol. 8, nº 15, pp. 111-122, 2021.
- [2] Microsoft, «Azure SQL Database: servicio de base de datos en la nube administrada | Microsoft Azure,» 2022. [En línea]. Available: <https://azure.microsoft.com/es-es/products/azure-sql/database/>.
- [3] Microsoft, «Automatización de procesos robótico (RPA) | Microsoft Power Automate,» 2022. [En línea]. Available: <https://powerautomate.microsoft.com/es-es/robotic-process-automation/>.

DEVELOPMENT OF A STOCK CONTROL TOOL FOR A STARTUP

Autor: Puerto Baydal, Manuel.

Director: Tornos de Inza, Ignacio.

Entidad Colaboradora: ICAI – Universidad Pontificia Comillas

ABSTRACT

The development of this project consists of the design and implementation of a stock control system, intended to be used in a start-up. The system has been developed using a SQL database and an RPA robot to automate the main tasks of running the database. Once finished, the system is able to register the movement of products in and out of the warehouse, track the levels of inventory and take actions based on those levels.

Keywords: Database, Robotic process automation, RPA, SQL, Stock, Inventory, Warehouse, Stock control

1. Introduction

The present work has been done in collaboration with a start-up to develop a stock control tool to be used by the company in the future to aid them in their warehouse management. The startup, a young company, is dedicated to developing their own marketplace, where different vendors offer their products to customers.

As a startup, they have plans of developing new business lines in the future, one of them being the possibility of distributing their own products. To achieve that, they would need a warehouse, to store the material, and some kind of software to manage it, including the movement of products in and out of the warehouse. In collaboration with the company's operations department, the tool to be used has been defined, which will consist of an SQL database and an RPA robot [1].

2. Definition of the project

The main activities that have been followed in the development of this project is shown in Figure 1. In the first place, a meeting was first held with the management of the company to find out its exact needs, what they were aiming to achieve once the tool was in place and what previous ideas they had about the implementation. In the second place, the possible alternatives were studied, seeing which solution fit best for the case at hand. The chosen option was to create a customized database combined with an RPA robot, which would collect all the information that the company considered crucial, and which would be a scalable solution as the volume of orders grows, without involving a large investment for the startup. Microsoft Azure was chosen as the option to host the database, since the company already uses it for other purposes, so it does not mean any additional cost for them. Something that would happen if, for example, it was decided to pay for some other type of additional software, such as an ERP. Once the solution to be implemented was decided, the database was created, with the different tables considered of interest. After that, the different flows that automate the tasks in the database associated with the entry and exit of material from the warehouse were designed and programmed. Finally, when everything was implemented, different tests were carried out

to verify the correct functioning of the system and the results were shown to the company managers.



Figure 1. Work development for the project

The main objectives of this work have been:

- Create a database with SQL that allows hosting all the necessary information of the company's product inventory.
- Programming a robot using RPA that automates the inventory management process when orders come in.
- Define what metrics will be used to measure the proper functioning of stock control.

3. Description of the system

The final design of the system consists of two parts. On the one hand, the database hosted in Microsoft Azure [2]. Being stored in the cloud allows users to access information regardless of where they are located. On the other hand, the automation flows of desktop actions designed using Power Automate Desktop, as everything is done in a Microsoft environment, it makes communication between the different programs used easier.

The created database consists of three different tables, related to each other by the product references. The first table, Inventory Control, will contain the references of the products, their inventory levels and their status, which is determined by the inventory level. The second table, Purchases and Sales, will record the inputs and outputs of products from the warehouse, containing the order information such as the order number, the product reference, the units, and the date the order is placed. Considering as purchases, the material acquisitions made by the company to replenish the inventory of products in the warehouse, and as sales, the sales of products made through its platform to customers. The third table, Article Master, will store the data of the different products with which the company works. This data will include the reference assigned by the company, the name, the price, the distributor, the manufacturer, and the manufacturer's reference, which may be different from the internal reference. The attributes in the tables will allow calculating metrics of interest such as shipping times, to detect anomalies, inventory turnover or safety stock. This data can be leveraged to optimize supply chain operations. Also, those KPIs defined by the company, help to assess their operations and propose future changes that can improve their warehouse operations and logistics.

Before starting to operate with an item, it is necessary to register it in the Product Master table and in the Inventory Control table. From that moment on, the incoming orders for that product will be recorded in the Purchases and Sales table, and the inventory level will vary accordingly in the Inventory Control.

In order to automate the tasks described in the previous paragraph, different flows have been designed in Power Automate Desktop that allow the system to free the user from having to perform them [3]. Each flow is designed to perform a specific task, the main ones being:

- Addition of new products to the database.
- Record of new purchases and new sales and updates of inventory levels.
- Reception and writing of emails.
- Update of the table records based on the information received and the calculations made.
- Visualization of data, which allows evaluating the performance of the warehouse and generating different analyses.

4. Results

Once the entire system has been implemented, the tool is ready to start operating. Its correct operation has been verified through the use of some examples, such as the one shown in Figure 2.

Referencia	Producto	Precio (€)	Stock de Seguridad	Distribuidor	Fabricante	Referencia Fabricante
1	Producto1	3,95	5	Distribuidor1	Fabricante1	11
2	Producto2	3,95	15	Distribuidor2	Fabricante1	122CL
3	Producto3	3,95	25	Distribuidor2	Fabricante1	333
4	Producto4	3,95	12	Distribuidor3	Fabricante2	44A44
5	Producto5	3,95	10	Fabricante3	Fabricante3	ABC5
6	Producto6	3,95	20	Distribuidor1	Fabricante2	66A

Ilustración 3. Data obtained from the database table Products Master

As it is shown, different products have been added to the database, products with different types of data, by using the appropriate Power Automate flow. Subsequently, another flow has been used to obtain the information from the database and copy it into an Excel sheet. Thus, verifying that the data has been correctly registered in the database and that it can be easily extracted.

5. Conclusions

The realization of this project has served the student to apply the knowledge acquired during the master's degree, as well as acquire new ones in relation to SQL and RPA, two very useful technologies today and that are becoming more important nowadays in the industry. In addition, being able to get involved in the daily activity of the company and become familiar with the strategic decisions related to the product distribution project.

Finally, it is necessary to consider the activities that have been excluded of the realization of this work, but that could be of interest if there was a desire to improve the tool. For

example, the inclusion of DPA flows that allow the system to work completely autonomously.

6. Bibliography

[1] C. A. Bermúdez Irreño, «RPA - AUTOMATIZACIÓN ROBÓTICA DE PROCESOS: UNA REVISIÓN DE LA LITERATURA,» Revista Ingeniería, Matemáticas y Ciencias de la Información, vol. 8, n° 15, pp. 111-122, 2021.

[2] Microsoft, «Azure SQL Database: servicio de base de datos en la nube administrada | Microsoft Azure,» 2022. [En línea]. Available: <https://azure.microsoft.com/es-es/products/azure-sql/database/>.

[3] Microsoft, «Automatización de procesos robótico (RPA) | Microsoft Power Automate,» 2022. [En línea]. Available: <https://powerautomate.microsoft.com/es-es/robotic-process-automation/>.

Índice de la memoria

Capítulo 1. Introducción	5
Capítulo 2. Descripción de las Tecnologías.....	9
2.1 La base de datos	9
2.2 automatización de las tareas de la base de datos	12
Capítulo 3. Estado de la Cuestión	15
Capítulo 4. Definición del Trabajo	21
4.1 Justificación.....	21
4.2 Objetivos	22
4.3 Metodología.....	24
4.4 Planificación y Estimación Económica	24
Capítulo 5. Diseño de la base de datos.....	29
5.1 Consideraciones generales	29
5.2 Base de datos SQL	30
5.2.1 Creación de las tablas	32
5.2.2 Adición de nuevos registros en las tablas	39
5.2.3 Eliminación de registros y elementos duplicados	40
5.2.4 Actualización de registros	42
5.2.5 Realización de consultas	43
5.2.6 Modificación de las tablas.....	44
Capítulo 6. Automatización de procesos	45
6.1 Automatización de la base de datos.....	45
6.2 Gestión de pedidos	47
6.2.1 Gestión de correos electrónicos	49
6.2.2 Gestión de documentos.....	50
6.2.3 Registro del pedido en la base de datos	53
6.2.4 Actualización del inventario.....	54
6.2.5 Actualización del estado del inventario del producto	56
6.3 Gestión de consultas.....	62

6.4	Adición de nuevos productos al almacén	65
6.5	Modificación de productos en el maestro de artículos	68
6.6	Eliminación de referencia de la base de datos	70
6.7	Cálculo del stock de seguridad	72
Capítulo 7. <i>Análisis de Resultados</i>		75
Capítulo 8. <i>Conclusiones y Trabajos Futuros</i>		81
Capítulo 9. <i>Bibliografía</i>		83
ANEXO	87	

Índice de figuras

Figura 1. Funciones que puede desarrollar un robot RPA (Deloitte, 2016)	13
Figura 2. Flujo de gestión de pedidos	48
Figura 3. Flujo de gestión de correos	50
Figura 4. Formatos de documentos utilizados para pedidos.....	51
Figura 5. Flujo de gestión de documentos	52
Figura 6. Flujo de registro de pedidos en la tabla Compras y Ventas	53
Figura 7. Flujo de actualización del valor del inventario de un producto	55
Figura 8. Flujo de actualización de estados de la tabla Control de Inventario	61
Figura 9. Flujo de gestión de consultas de la base de datos	64
Figura 10. Flujo de adición de nuevos productos de la base de datos	66
Figura 11. Formulario de adición de nuevos productos	68
Figura 12. Flujo de modificación de atributos de un producto.....	69
Figura 13. Flujo de eliminación de productos de la base de datos	71
Figura 14. Flujo del cálculo del stock de seguridad	73
Figura 15. Formulario de adición de productos rellenado	76
Figura 16. Documentos utilizados para simular el pedido	77
Figura 17. Datos de Compras extraídos de la base de datos a una hoja Excel	78
Figura 18. Datos de los artículos contenidos en la tabla Maestro de Artículos, extraídos a una hoja Excel	78

Índice de tablas

Tabla 1. Presupuesto mensual estimado de Azure (Microsoft, 2022)..... 27

Capítulo 1. INTRODUCCIÓN

El presente proyecto se ha desarrollado en el marco de una empresa. Se trata de una compañía joven, una *startup* con menos de dos años de antigüedad. Por este motivo, todavía se encuentra definiendo ciertos aspectos de su modelo de negocio, entre los que existen diversas líneas posibles que tiene aún pendientes de desarrollar. Actualmente, consiste en un *marketplace*, donde los distintos distribuidores que posee venden sus productos a los clientes a través de la plataforma. La empresa se encarga únicamente de gestionar la plataforma que conecta a distribuidores y clientes, pero no compite como un proveedor más dentro de su propia plataforma.

En el futuro, una de las posibles líneas estratégicas de ampliación podría ser la distribución de productos, ya sea comprando productos de marcas ya establecidas y vendiéndolos a los clientes, actuando solo como intermediario, o mediante la creación de su propia marca blanca, que le permita ofrecer a los clientes los productos más demandados a un precio reducido. Este cambio de enfoque implicaría variar ciertos perfiles en la empresa, al alejarse de su actividad principal.

Si desea ofrecer el servicio adicional de distribuidor, la empresa necesitaría operar su propio almacén, en el que mantener el inventario de los productos ofertados. Para gestionar ese almacén de forma eficiente y suministrar los pedidos de forma correcta a sus clientes, se requiere controlar el movimiento de los productos del almacén, ya que esto permite conocer qué cantidad de productos hay en cada momento, y cómo se desarrolla la demanda de los distintos productos disponibles a lo largo del tiempo. Por ello, es necesario disponer de algún tipo de herramienta que permita realizar estas tareas de la manera óptima. El desarrollo de esta línea estratégica también implicaría otras consideraciones como la elección de la ubicación del almacén, y la distribución y operación del mismo, pero estos aspectos, por su complejidad, quedan fuera del ámbito de aplicación de este proyecto.

Es interesante destacar como la introducción de esta línea de negocio convertiría una empresa puramente tecnológica en una empresa con una rama de logística propia. La logística abarca transporte, empaquetado y almacenaje de productos, así como la gestión del almacén y sus capacidades. En resumen, controlar todo el movimiento estratégico de “adquisición, movimiento y almacenamiento de productos, así como todo el flujo de información asociados” [1].

Debido a las características de la empresa, una empresa joven en crecimiento, con poco capital para invertir en gastos fijos adicionales, se ha creído conveniente desarrollar una herramienta propia para la gestión del inventario. Esta decisión presenta dos ventajas fundamentales. Por una parte, proporciona flexibilidad y permite personalizar la solución, y por otra, supone un ahorro en gasto, lo que permite dedicar el dinero ahorrado, que llevaría contratar una solución tecnológica comercial, a otros proyectos estratégicos. Hay que destacar que, para poder asegurar la utilidad de la herramienta a medida que crecen la cantidad de referencias en el almacén, es importante diseñarla teniendo en cuenta su escalabilidad. De lo contrario, podría quedar obsoleta en poco tiempo. Por eso, se ha decidido programar una base de datos con SQL y hacer uso de un robot que permita automatizar gran parte de las tareas informáticas asociadas a gestionar el movimiento de productos dentro del almacén.

Para definir correctamente las funcionalidades de la herramienta, se ha llevado a cabo un estudio, junto con los responsables de Operaciones de la empresa, ya que será el departamento encargado de gestionar el almacén cuando este esté en funcionamiento. Se ha buscado su aportación para conocer qué es lo que se pretende lograr una vez implementada la herramienta, qué información es importante tener en cuenta, y qué métricas se pretenden aplicar sobre esa información. Todo esto permitirá tener una herramienta lo más completa posible, y lo más ajustada a las necesidades de la empresa.

La solución tecnológica planteada ha de tener en cuenta las peculiaridades propias de la empresa para la que está siendo diseñada. El *marketplace* tiene una marcada variación estacional en la demanda de productos, coincidiendo los puntos más bajos de la demanda

con las vacaciones y festividades en España. Es notable cómo, a medida que se acercan los días festivos, la demanda se reduce, porque los clientes ya tienen en cuenta que van a realizar un menor consumo de materiales durante esas fechas y simplemente intentan agotar el stock que ya tienen antes de cerrar por vacaciones. Este aspecto será clave para no rellenar innecesariamente el almacén antes de esas fechas, sabiendo que las ventas durante ese periodo se reducirán y, por lo tanto, no se le va a dar salida al stock almacenado hasta que pasen esas fechas.

Capítulo 2. DESCRIPCIÓN DE LAS TECNOLOGÍAS

Como se ha indicado en el capítulo anterior, el trabajo para gestionar el almacén ha consistido fundamentalmente en el desarrollo una base de datos SQL, así como de un robot que permita automatizar gran parte de las tareas informáticas asociadas a la gestión de los productos del almacén. De común acuerdo con la empresa, las herramientas seleccionadas para este proyecto han sido principalmente *Microsoft Azure*, para crear la base de datos SQL en *Microsoft SQL Server*, *Azure Data Studio*, que permite interactuar con bases de datos SQL en *SQL Server*, y *Microsoft Power Automate Desktop*, para automatizar los procesos relacionados con la gestión de la base de datos generada. Para la base de datos en SQL se usará *Microsoft Azure*, *Azure SQL Database*, y *Azure Data Studio*, debido a que ya es actualmente utilizado por la empresa para realizar otras funciones, aunque gracias a lo unificado que está el lenguaje de SQL se podría usar otra opción y después adaptarlo sin grandes problemas. Asimismo, se ha decidido utilizar *Power Automate Desktop* por la sencillez para automatizar procesos que incluyan otras aplicaciones de Microsoft, como es el caso de *SQL Server*, facilitando la operación de la herramienta diseñada.

2.1 LA BASE DE DATOS

SQL es un lenguaje de programación pensado para interactuar con bases de datos relacionales, permitiendo a los usuarios administrar y obtener la información de sistemas de gestión de bases de datos relacionales [2]. En nuestro caso su uso será fundamental, ya que se pretende trabajar grandes cantidades de información, que será crucial para el correcto funcionamiento del control de inventario. Es necesario que la información esté actualizada en todo momento y sea veraz, reduciendo al mínimo el riesgo de que se introduzcan datos corrompidos, o se alteren erróneamente los datos existentes, que alteren el correcto funcionamiento de nuestro sistema.

Una base de datos relacional consiste en un tipo de base de datos que almacena y permite el acceso a distintos puntos de datos relacionados entre ellos [3]. La información se organiza en tablas, en las que cada una de las columnas es un campo y cada una de las líneas constituye un registro. Cada uno de los campos contiene un atributo distinto de los datos registrados, teniendo, generalmente, un valor diferente para cada uno de sus atributos, pudiendo así diferenciar fácilmente los distintos registros contenidos en una tabla. Que los atributos de los diferentes registros sean distintos facilita el operar con las bases de datos relacionales, pues conociendo un dato de un registro de una tabla, se puede acceder a otra tabla que tenga ese mismo atributo y ver el registro almacenado allí. Un ejemplo podría ser una tabla que contenga la información de los pedidos de la empresa, indicando el número de pedido, los productos, las unidades, el coste, y el nombre del cliente, y una segunda tabla que contenga los datos de los clientes, nombre del cliente, dirección de envío, dirección de facturación, número de teléfono y correo electrónico. Conociendo el número de pedido, se podría saber el nombre del cliente y con esa información acceder a la segunda tabla para obtener la información necesaria de ese cliente.

Para poder facilitar la introducción de datos en las tablas, se trabaja con distintos tipos de datos en SQL, previamente definidos en el momento de creación de las tablas. Esto permite acotar el tipo de información que se introduce posteriormente. En este proyecto, trabajaremos con diferentes tipos de datos en función del tipo de atributo que queramos registrar o modificar:

- **Char:** Una cadena de tamaño fijo que puede estar formada por letras, números y caracteres especiales. Se utiliza para definir atributos que siempre tienen la misma longitud, y cuya longitud es conocida. Puede resultar útil para acotar las entradas de información y así estandarizar las operaciones, haciendo más sencilla la lectura de información entre diferentes usuarios. De esta manera, las operaciones utilizarán siempre las mismas cadenas para ser definidas, no se puede variar la información introducida porque no se adaptaría al formato definido.
- **Varchar:** Una cadena de tamaño variable que puede incluir letras, números y caracteres especiales. Será útil para datos como las referencias de los productos,

el nombre de los productos, el proveedor o el fabricante, porque cada registro tendrá una longitud diferente.

- **Int:** Permite introducir números enteros. El número de unidades almacenadas será un ejemplo de su uso. Algunos programas de SQL, como *MySQL*, permiten definir el signo de los valores numéricos usando `int UNSIGNED`, acotando la entrada a valores numéricos positivos. No es el caso de *Microsoft Azure Data Studio*, por lo que no podemos hacer esa distinción. Sería una gran ventaja que nos permitiría evitar errores a la hora de introducir valores que no pueden ser negativos, como por ejemplo el número de unidades en la tabla `CONTROL_DE_INVENTARIO`, ya que en ningún caso las unidades almacenadas pueden ser inferiores a 0.
- **Date:** Permite introducir datos con formato de fecha. Es relevante para conocer cuándo se realizan los pedidos, fechas de las compras, ventas y los tiempos de envío. Para poder introducir fechas en SQL deben tener el formato `MM/DD/AAAA`, al contrario de la manera habitual que tenemos de representar las fechas en España, que sería `DD/MM/AAAA`.

Existen infinidad de sistemas de gestión de bases de datos con soporte SQL, algunos ejemplos serían *MySQL*, *IBM DB2* o *Microsoft SQL Server*. Dichos sistemas se conocen como sistemas de gestión de bases de datos relacionales (RDBMS), que “es el software que ejecuta consultas sobre los datos, incluida la adición, actualización y búsqueda de valores”, pudiendo incluso mostrar una representación gráfica de los datos [4]. Existen pequeñas diferencias entre los distintos sistemas de gestión, pero el funcionamiento está bastante estandarizado, siendo sencillo pasar a trabajar de un sistema a otro. Son soluciones que permiten almacenar la información en la nube, albergando los datos en los servidores del proveedor informático y proporcionando acceso remoto, por lo que pueden ser accedidos desde cualquier lugar del mundo.

Como se ha mencionado al principio, en este proyecto se utilizará *Microsoft Azure* y, más en concreto, para implementar la base de datos, *Azure SQL Server*, que cumple con los requisitos de estabilidad, escalabilidad y seguridad, para prevenir la corrupción de los datos

almacenados [5]. *Microsoft Azure* es el sistema de computación en la nube creado por Microsoft para el desarrollo de aplicaciones, utilizando para ello sus centros de datos [6]. Entre los servicios proporcionados, se encuentra *Azure SQL Server*, que permite tener la base de datos albergada en la nube, con las ventajas que eso supone para facilitar el acceso a la información [7]. Con los planes disponibles, Azure asegura un acuerdo de nivel de servicio (SLA) de al menos el 99,99% para su suscripción básica, siendo incluso superior para otros planes. Esto asegura que el máximo tiempo que el servicio puede estar interrumpido es de 5 minutos por año, demostrando así su fiabilidad [8].

2.2 AUTOMATIZACIÓN DE LAS TAREAS DE LA BASE DE DATOS

La automatización robótica de procesos (RPA) es una manera de automatizar procesos, mediante la cual, se hace uso de un software, un robot, que permite automatizar procesos sin necesidad de tener conocimientos de programación, replicando los pasos que haría una persona interactuando con la interfaz gráfica de un sistema informático, mejorando la ejecución de los flujos de trabajo [9]. Este sistema permite automatizar operaciones que de otra manera tendrían que ser llevadas a cabo por un operario, liberando a la persona de tareas informáticas tediosas y repetitivas, generando mayor satisfacción entre los empleados que tienen tiempo para dedicarse a tareas más enriquecedoras [10]. Estas operaciones suelen tener dos características en común, la estandarización de los pasos a realizar, dado que se trata de tareas repetitivas, y el volumen de operaciones que hay que realizar, cuanto mayor es el volumen de una misma tarea, mayor sentido tiene intentar automatizar el proceso. De esta manera ahorramos tiempo y prevenimos errores provenientes de la acción humana.

Es importante destacar que cuando hablamos de robots no nos referimos a robots físicos (hardware), como se puede asociar instintivamente cuando escuchamos el término. Como ya se ha mencionado, se trata de un software configurado para copiar las actividades que haría una persona delante de un ordenador. Se compone de 4 partes:

- El robot, refiriéndonos al software encargado de realizar las tareas.
- Una interfaz gráfica, interactiva con el usuario para poder programar con facilidad.

- Un hardware, virtual o físico, en el que desplegar el robot.
- Compatibilidad con diversas plataformas, a las que puede acceder el robot al igual que lo haría una persona.

Una de sus ventajas más evidentes es que no es necesario saber de programación para poder programar el robot, se configura utilizando pasos demostrativos, realizando la tarea paso a paso que más adelante realizará el robot por su cuenta. Esta característica permite que cualquier persona, por muy tecnófoba que sea, pueda servirse de los robots para automatizar tareas repetitivas. Es importante saber analizar correctamente qué procesos pueden ser automatizados y cuáles no, una selección incorrecta podría llevar a redundancias, no todos los procesos deben ser automatizados, o a pérdida de calidad en el trabajo, si son tareas que requieren de un análisis y toma de decisiones que solo una persona puede proporcionar. Como se puede observar en la Figura 1, algunos ejemplos relevantes de tareas que es capaz de realizar un RPA son abrir correos y descargar los archivos adjuntos, para poder registrar los pedidos que van entrando al sistema, generar correos, para avisar al responsable cuando sea necesario realizar un pedido para reponer el inventario, leer y rellenar formatos, obteniendo la información de las facturas y volcando esa información en la base de datos, realizar cálculos y obtener información, sacando la información de la base de datos y obteniendo las métricas que nos permitirán mantener actualizados nuestros KPI.

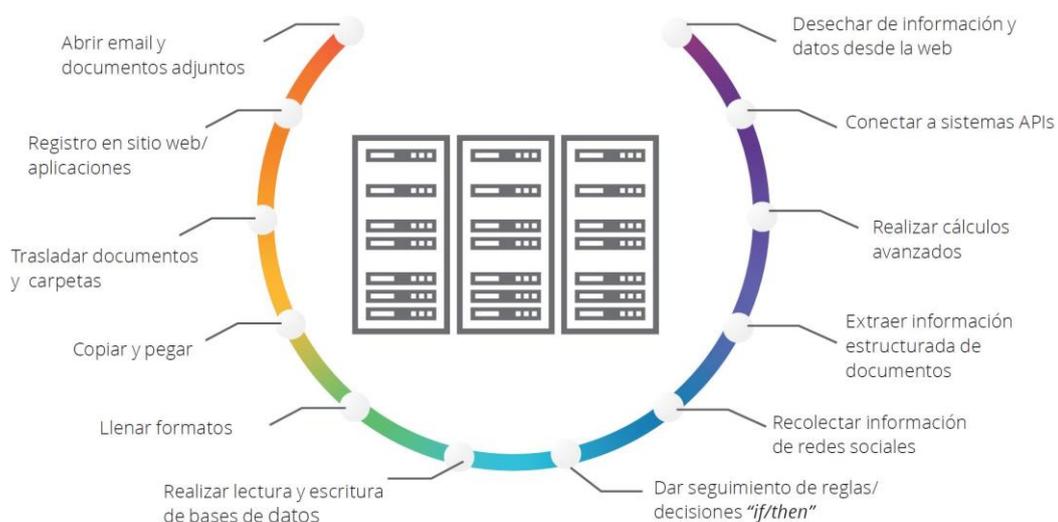


Figura 1. Funciones que puede desarrollar un robot RPA [9]

Al ser una tecnología relativamente reciente, será interesante ver cómo se desarrolla en el futuro, permitiendo automatizar procesos que en la actualidad quedan fuera del espectro de aplicación de RPA por su complejidad. La tendencia de crecimiento en los últimos años hace presagiar que cada vez será una tecnología más demandada por parte de la industria, favoreciendo el desarrollo de nuevas funcionalidades por parte de los proveedores de software [11].

Para este proyecto, se ha decidido utilizar *Power Automate Desktop*, antes conocido como *Microsoft Flow*, en la automatización de los procesos necesarios para la gestión de la base de datos [12]. Otras opciones comerciales de RPA incluyen *UiPath* o *Automation Anywhere* [13]. Además de la versión para automatizar procesos de escritorio, también existe *Power Automate* en la nube, cuyo objetivo es la automatización de procesos digitales (*DPA*) [14]. En nuestro caso, el uso de *Power Automate* está plenamente justificado porque, al utilizar otros servicios de Microsoft para la base de datos, la integración resulta más sencilla e intuitiva, dado que *Power Automate* posee una serie de acciones pensadas para introducir en sus flujos, que facilitan la conexión a otros servicios también creados por Microsoft.

Capítulo 3. ESTADO DE LA CUESTIÓN

El stock son las existencias, la mercancía guardada en un almacén [15]. Por otro lado, el inventario es un “asiento de los bienes y demás cosas pertenecientes a una persona o comunidad, hecho con orden y precisión” [16]. ¿Cómo podemos hacer para que exista ese orden y precisión? ¿Cómo aseguramos que exista concordancia entre lo registrado en el inventario y el stock real? Para poder controlar la mercancía del almacén es necesario algún tipo de sistema que permita mantener actualizado el inventario en todo momento y refleje los niveles de stock reales en todo momento. Es aquí cuando surge la necesidad de implantar sistemas de control de inventario para solventar ese problema.

El control de inventario es una de las principales funciones dentro del área logística de una empresa. Su cometido es supervisar la cantidad de unidades de cada producto disponibles que posee la empresa, entendiendo que el conjunto global de productos almacenados compone el stock de la empresa. Todo esto con el objetivo de mejorar la eficiencia en la cadena de suministro lo máximo posible. De esta manera, es posible suministrar los pedidos a tiempo y que no se produzcan fallos en la cadena de suministro por errores en el inventario. Los productos almacenados pueden dividirse en dos tipos, materias primas, empleadas en empresas de producción, que utilizan los recursos como entradas para producir bienes, y productos terminados, que pueden estar almacenados en la fábrica una vez se ha concluido su proceso de producción o en almacenes de distribución.

Las principales características que debería poseer un software de gestión de inventario han de ser, con algunas variaciones en función del inventario a manejar y el tipo de producto:

- **Identificación del producto:** Conocer las características que permiten diferenciar el producto. Referencia, nombre del producto, fabricante, distribuidor, y precio de compra y de venta, pueden ser ejemplos de datos que permitan catalogar correctamente los productos. De ahí se pueden obtener los datos que permiten gestionar las compras, sabiendo la referencia del producto, el distribuidor y el

fabricante, que en el caso de que sean productos comprados directamente al fabricante es la misma empresa. Con el histórico del precio podemos conocer la evolución del precio a lo largo del tiempo.

- **Punto de pedido:** Permite saber y avisar de en qué momento es necesario realizar un pedido de un producto para reponer el inventario. Para ello se programa un cierto nivel de inventario, de manera que cuando el número de unidades disponibles disminuya por debajo de ese número, genere una alerta a la persona disponible, avisándola de que es necesario realizar un pedido para reponer el stock.
- **Localización de productos:** Para identificar dónde se encuentra almacenado un determinado producto dentro del almacén. Existen varias tecnologías que permiten señalar la localización de un producto. Por ejemplo, se pueden utilizar códigos de barras, que además permiten conocer los datos de identificación de los productos almacenados.
- **Coste del material:** Saber cuál es el coste de los distintos productos almacenados y, por lo tanto, del coste total de las materias almacenadas. Aunque sea una herramienta pensada para la gestión logística, más departamentos de la empresa se pueden beneficiar de los beneficios del correcto uso del software de gestión del inventario.
- **Optimización de tareas:** Al tener toda la información centralizada y catalogada, es más fácil obtener la información relevante requerida en todo momento. Esa información también permite optimizar actividades dentro del propio almacén, por ejemplo, facilidad y rapidez para localizar un producto. También permite obtener distintas métricas como pueden ser el cálculo del punto de pedido, la cantidad óptima de unidades que se han de pedir, la demanda durante un periodo específico de tiempo, o la duración estimada del inventario, cuánto tiempo falta para que se agote el inventario actual, en caso de que no se hiciera un pedido para reponerlo.
- **Control en tiempo real:** Una de las funciones más importantes es tener actualizados los niveles de inventario en todo momento, para evitar problemas

con los pedidos y el suministro, dando por válidos pedidos cuando no hay inventario suficiente para poder suministrarlos o, a la inversa, bloqueando pedidos de productos que sí que tienen inventario suficiente y podrían ser servidos adecuadamente.

- Creación de informes: La información contenida en el sistema de gestión permite extraer los datos relevantes que permiten medir distintos indicadores clave de rendimiento (KPI, *Key Performance Indicators*) y ver el desarrollo del negocio. Permitiendo así evaluar la idoneidad del negocio, sabiendo qué productos tienen mayor demanda y a los que se les tendría que dar prioridad en el negocio.

En la actualidad, existen diferentes soluciones comerciales que permiten gestionar un inventario de manera eficiente. Las posibilidades varían desde sencillas soluciones como generar un modelo en *Microsoft Excel* o *Google Sheets*, que permita ir anotando las variaciones del inventario, hasta complejos programas informáticos pensados específicamente para la gestión de inventario. Existe gran cantidad de software pensado para controlar el stock en las empresas, como pueden ser *ABC Inventory*, *RightControl*, *SIGA* o *Easy WMS* [17]. Otro ejemplo de software de control de stock, aunque algo más alejado de lo que se busca en este trabajo, podría ser *ON!Track*, el programa de la empresa Hilti [18]. Un software muy completo que permite la gestión de inventario consumible, así como de herramientas y otros productos que pueden ir variando su localización y su estado, siendo ideal para empresas con operarios que se desplazan entre distintos lugares de trabajo, como pueden ser los técnicos de empresas de construcción industrial.

Los softwares de gestión de stock suelen estar pensados para pequeñas y medianas empresas, por el contrario, las empresas de gran tamaño es más común que utilicen un sistema de planificación de recursos empresariales, que agrupa todo tipo de información en un único sitio. Este tipo de software permite a las empresas agrupar todo tipo de información en un mismo lugar.

Un sistema de planificación de recursos empresariales (ERP, *Enterprise Resource Management*) podría ser una solución integral que permitiría gestionar el almacén y el

inventario de productos almacenados. Estos sistemas permiten gestionar los recursos de la empresa, abarcando muchos más aspectos de los planteados en este proyecto. Son sistemas que ayudan a unificar todos los flujos de información derivados de las operaciones de una empresa, asegurando la trazabilidad de todos sus procesos. Suelen estar divididos en distintos módulos en función de las necesidades concretas de la empresa, dependiendo su coste y complejidad de los módulos que se contraten. Sus características principales son:

- Agrupar y trazar todas las operaciones en un único sistema.
- Optimizar los procesos de la empresa.
- Planificar los recursos necesarios futuros.
- Automatizar los procesos entre distintos departamentos de la empresa.
- Fácil acceso a los datos y presentación estructurada de información.
- Facilidad para compartir información.
- Eliminación de información y procesos innecesarios.

Un ejemplo podría ser SAP Business Suite y su módulo de Software de Gestión de la cadena de suministro de SAP (SAP SCM), que dan acceso a *SAP Extended Warehouse Management* (SAP EWM), permitiendo la gestión de almacenes [19]. Su software de gestión de almacenes permite la integración con la aplicación de *myWarehouse*, la cual permite a los operarios realizar las tareas del almacén mediante el uso del teléfono móvil [20]. Otro ejemplo podría ser Holded, que entre las opciones que oferta incluye software de gestión de inventarios [21]. Este tipo de soluciones son perfectas si ya se usan otros módulos del ERP, porque permiten la comunicación entre módulos. Una mayor integración del ERP, contratando más módulos, consigue que se tenga toda la información centralizada dentro de la empresa.

Estas soluciones están pensadas para gestiones complejas y confieren gran versatilidad al cliente. El problema de estas soluciones es que suelen tener un coste económico elevado y solamente son rentables si se manejan grandes presupuestos y cantidades ingentes de inventario. En nuestro caso, buscar una solución a medida que se adapte a las necesidades de la empresa, que sea escalable a medida que se va desarrollando el proyecto de albergar su propio inventario, y que permita medir las métricas concretas solicitadas por la empresa,

hacen preferible la idea de desarrollar desde cero la herramienta para el control del inventario y tener la libertad de poder diseñar una solución adaptada específicamente a las características del contexto.

Capítulo 4. DEFINICIÓN DEL TRABAJO

4.1 JUSTIFICACIÓN

La realización de este trabajo viene motivada por la posibilidad de realizar una herramienta personalizada que contribuya al desarrollo de uno de los proyectos estratégicos que tiene la empresa en el medio plazo. No siendo urgente, sí que es interesante desarrollarlo cuanto antes para así tener todos los componentes necesarios disponibles cuando se decida acometer la implantación del almacén propio. Además, permite al alumno aplicar los conocimientos adquiridos a lo largo del máster, así como involucrarse en las operaciones de la empresa en mayor medida y conocer mejor el funcionamiento interno de esta. No solamente permitiéndole profundizar en los detalles técnicos, sino también en otros aspectos de carácter más empresarial. Este trabajo es el comienzo de una posible rama estratégica dentro de la empresa, y su correcto desempeño ayudará en la decisión de hasta qué punto resulta relevante su desarrollo. Además, la solución escogida aporta flexibilidad y ahorro económico para la empresa.

La posibilidad de poder colaborar estrechamente con la empresa y poner en práctica los conocimientos adquiridos durante los estudios del máster son una gran oportunidad para el desarrollo académico, profesional y personal. Permitiendo involucrarse de manera palpable en el desarrollo de la empresa, dentro de su área de operaciones, y asistir en la toma de decisiones estratégicas de cara al futuro. Para desarrollar este trabajo han resultado relevantes los conocimientos adquiridos en asignaturas como Sistemas de Producción y Fabricación (SPF) y Dirección de Operaciones (OM), al ser aplicados, en este trabajo, aspectos como el cálculo del inventario, la tasa de rotación de inventario, o el stock de seguridad.

La decisión de desarrollar un modelo propio proviene de la idea de diseñar una solución a medida que sea eficaz, escalable, y que aporte mayor flexibilidad a la empresa sobre qué momento es más propicio para tomar la decisión de desarrollar el proyecto. La idea propuesta

permite aportar una solución intermedia entre las soluciones actuales existentes, aunando como potenciales virtudes que sea una solución económica, escalable e intuitiva. Por una parte, no requiere pagar por un sistema de control de inventario comercial, que puede resultar demasiado caro al principio, cuando no se están manejando grandes volúmenes de inventario y no resulta rentable contratar un programa para gestionarlo. Por otra parte, no implica usar un sistema barato, o incluso gratuito, pero que sea muy rudimentario y quede rápidamente obsoleto en cuanto los volúmenes de inventario o la variedad de productos manejados aumenten.

Una solución industrial convencional podría ser demasiado cara para el plan inicial de gestionar un inventario muy limitado de productos, por el contrario, un Excel podría ser una buena solución para ese estadio inicial, pero se quedaría rápidamente obsoleto y obligaría a la empresa a cambiar a otro programa pronto, con el consiguiente problema causado por tener que migrar toda la información al nuevo programa mientras el almacén está en funcionamiento, con sus consiguientes variaciones diarias de inventario. La opción de Excel no es viable cuando ya se manejen grandes volúmenes de información debido a la facilidad con la que se podrían introducir datos erróneos y corromper la información, haciendo inútil el sistema de control de inventario. Por eso es conveniente aunar una base de datos en SQL, que permite manejar la información de manera económica y escalable, a medida que aumenta el volumen de información, con un RPA que automatice, dentro de lo posible, los distintos pasos para registrar los pedidos, haciendo la gestión de los procesos lo más sencilla e intuitiva posible.

4.2 OBJETIVOS

Mediante la realización de este proyecto se pretende diseñar e implementar un sistema de gestión de inventario en la empresa. Llegando así a conseguir los 3 principales objetivos enumerados a continuación:

- Crear una base de datos con SQL que permita albergar toda la información necesaria del inventario de productos de la empresa.

- Diseñar cómo, mediante la información contenida en la base de datos, se ha de poder gestionar toda la información relacionada con las compras y las ventas, indicando las fechas de cada una de las acciones y las cantidades implicadas, para ver cómo afectan a la variación del inventario a lo largo del tiempo.
- Idear la mejor manera de que la información dentro de la base de datos esté bien catalogada, definiendo los tipos de datos en las diferentes filas y columnas, para prevenir valores erróneos que corrompan la información.
- Estructurar la base de datos de forma que permita facilitar el flujo de información entre diferentes departamentos. Por ejemplo, cuántos proveedores existen, o con qué número de fabricantes se trabaja.
- Programar un robot usando RPA que automatice el proceso de la gestión del inventario cuando entren los pedidos.
 - Automatizar los distintos pasos que tendría que realizar una persona para registrar la información al solicitar nuevos productos, en el momento de recibirlos, y cuando se procesa un pedido de productos del almacén.
 - Registrar las variaciones en el inventario y proporcionar los avisos adecuados cuando se requiera. Por ejemplo, cuando el stock de un producto descienda por debajo de un cierto nivel.
- Definir qué métricas se usarán para medir el correcto funcionamiento del control de inventario.
 - ¿Qué información relevante se espera obtener?
 - ¿Cuál será el stock de seguridad para cada producto?
 - ¿Cómo ha de medirse qué productos son los más demandados y son los prioritarios?
 - Para poder medir el desarrollo del inventario a lo largo del tiempo, ¿con qué periodicidad habrá que obtener esa información?

4.3 METODOLOGÍA

En primer lugar, se estudiará qué objetivos concretos pretende lograr la empresa una vez esté finalizado el trabajo, definiendo cuáles son las funciones exactas que ha de realizar la herramienta, y por tanto fijando el alcance real del proyecto. Es importante tener claras las funcionalidades exactas que se espera obtener de la herramienta una vez esté finalizado el proyecto. Para este paso es fundamental estar en contacto constante con los responsables de la empresa para poder plasmar sus requerimientos de la mejor manera posible, siempre teniendo presente que se han de cumplir los criterios técnicos.

Más adelante, se creará una base de datos en SQL con toda la información que hayan de contener los distintos productos, campos como referencia, nombre, unidades almacenadas, precio y distintas métricas como el stock de seguridad, fecha del último pedido y coste del inventario almacenado, entre otras. Partiremos de un modelo simplificado elaborado en Excel que permita poner en común las necesidades de la empresa y poder mostrar cuál sería el resultado final de lo que se pretende lograr de una manera sencilla e intuitiva.

Por último, se programará un robot, utilizando RPA, que permita automatizar el proceso de actualizar la base de datos y las consecuencias derivadas, cuando se realice un pedido. Será necesario descontar una unidad del inventario actual, y avisar al encargado de que ha entrado un pedido. En caso de que el stock resultante sea inferior al stock de seguridad, se notificará al responsable de que es necesario realizar un nuevo pedido al proveedor. También es importante que se pueda extraer periódicamente información relevante de la base de datos. ¿Cuántos productos se han vendido en un cierto periodo de tiempo? ¿Qué productos tienen una mayor demanda? ¿Ha habido alguna modificación en los productos más demandados desde la última vez que se comprobó el listado?

4.4 PLANIFICACIÓN Y ESTIMACIÓN ECONÓMICA

En los cálculos del coste de desarrollo del proyecto, mostrado a continuación, es importante tener en cuenta que las opciones seleccionadas se han basado en los servicios ya contratados

por la empresa, por lo que, en el caso concreto de desarrollo de este proyecto, no supone ningún gasto adicional para ellos. Los costes mostrados en este apartado, que pueden variar según los servicios de los que se desee disponer, son los que se tendrían si se quisiera contratar estos servicios por primera vez. En función de las necesidades específicas, existen distintos planes a los que suscribirse, que abarcan diferentes cantidades de los servicios que oferta Microsoft. Dentro del cálculo de los costes, únicamente se ha contabilizado el coste de los servicios necesarios para poner en funcionamiento la infraestructura de la herramienta de control de inventario y mantenerla, de ahí que se haya calculado el coste mensual. No se ha incluido en los cálculos el coste de personal que podría suponer, porque todavía no se tiene decidido, por parte de la empresa, quién ha de ser el encargado de supervisar y gestionar el correcto funcionamiento del sistema.

Es necesario disponer de un servidor que albergue la base de datos SQL y de un programa de automatización RPA. Como se ha mencionado en el Capítulo 2, la empresa ha decidido albergar el servidor en la nube, más concretamente en *Microsoft Azure*. Por facilidad se ha decidido trabajar todo en un entorno Microsoft, que adicionalmente ofrece mayor facilidad al usuario de interoperabilidad dentro de sus aplicaciones.

El presupuesto mostrado en la Tabla 1 proviene de la calculadora de precio de *Microsoft Azure*, que nos permite. Es importante resaltar que, como se indica en la página, “los precios son solo estimaciones y no pretenden ser presupuestos reales. Los precios reales pueden variar según el tipo de contrato acordado con Microsoft, la fecha de compra y el tipo de cambio. Los precios se calculan en dólares estadounidenses y se convierten con las tasas de referencia de Thomson Reuters, actualizadas el primer día de cada mes natural.” Pero, los datos de la tabla permiten conocer una estimación aproximada del coste de contratación, que variaría en función de las necesidades específicas y del uso que se le dé. Como se muestra, el pago se realiza por uso, no es necesario pagar nada por anticipado, y el importe final irá acorde al uso [22].

Con respecto a las tarifas de *Power Automate*, existen distintas opciones en función de las necesidades. En el caso de este proyecto, se opta por el Plan por usuario con RPA atendida,

con un coste de 33,70 € por usuario y mes, que permite a los usuarios utilizar flujos en la nube y en el escritorio [23].

Categoría	Tipo de servicio	Región	Descripción	Coste mensual estimado	Pago adelantado estimado
Bases de datos	Azure SQL Database	Europa Oeste	Base de datos única, Núcleo virtual, Almacenamiento de copia de seguridad RA-GRS, Crítico para la empresa, Aprovisionado, Serie Estándar (Gen 5), 1 - 8 vCore instancias, reserva de 1 años, 5 GB de almacenamiento, 1 GB de almacenamiento de copia de seguridad	3,422.24 €	0.00 €
Soporte	Soporte Licencia		Microsoft Customer Agreement (MCA)	0.00 €	
	Total			3,422.24 €	

Tabla 1. Presupuesto mensual estimado de Azure [22]

Capítulo 5. DISEÑO DE LA BASE DE DATOS

En este apartado se explicará el proceso seguido para la estructuración del trabajo y el desarrollo de la base de datos SQL que albergará la información para la gestión del control de inventario.

5.1 CONSIDERACIONES GENERALES

Antes de comenzar a diseñar la herramienta, es importante reflexionar sobre las condiciones indispensables a tener en cuenta para crear una solución que cumpla con los requisitos y tenga en consideración las condiciones específicas que afronta la empresa en su gestión diaria, y lo que desea conseguir una vez implementada la herramienta de control de stock.

Por lo tanto, conviene plantearse una serie de preguntas ¿Qué desea medir la empresa? ¿Qué información considera vital recoger y mostrar? ¿Cómo ha de estar distribuida la información en la base de datos para lograr los objetivos?

En primer lugar, se desea medir el inventario, lo que resulta fundamental para poder mantener, en todo momento, actualizada la cantidad de unidades disponibles de cada producto dentro del almacén. Solo de esta manera será posible saber con seguridad si somos capaces de suministrar un pedido entrante, mientras la cantidad de unidades solicitadas sea igual o inferior a las unidades disponibles en nuestro almacén, y en qué momento deberemos realizar un pedido para reponer el inventario, antes de que se agote. Además de eso, el sistema de control de inventario puede ser útil para proporcionar información adicional a otros departamentos de la empresa, como puede ser conocer el valor de la mercancía almacenada.

La herramienta desarrollada también debe permitir a la empresa disponer de otra información de utilidad que sirva de base para poder medir diferentes métricas de interés. Por ejemplo, cuánto tiempo tardan los envíos, para rastrear cualquier anomalía y poder

conocer sus causas, tener accesible en todo momento la información relevante de los productos, como puede ser la información sobre fabricantes y distribuidores, poder calcular qué productos son los más demandados y cuál es su rotación de inventario, para detectar tendencias de compra y la asiduidad con la que se compran los productos.

Al principio, el almacén de la empresa se centrará solo en productos con gran demanda, por tanto, con gran consumo por parte de los clientes, por lo que podemos asumir que la rotación de inventario será elevada. Principalmente, serán artículos que la empresa haya detectado que tienen una mayor demanda en el *marketplace*, por lo que puede resultarle más rentable distribuirlos por la propia empresa. Es presumible que, al ser productos de gran consumo, haya un gran número diario de interacciones con la base de datos, con su consiguiente variación del inventario. Por lo tanto, el escenario inicial planteado es poca variedad de productos, pero gran variación de los niveles de inventario semanalmente. A medida que avance el tiempo, se irá expandiendo la cantidad de productos ofertados, añadiendo más adelante otros artículos de mayor precio y menor demanda, por lo que las variaciones de inventario serán menos pronunciadas, haciendo que esos productos tengan una menor tasa de rotación del inventario.

Por las características de la empresa, existe gran variación estacional en la demanda de productos, siendo las vacaciones de verano, Navidad y Semana Santa, momentos en los que la demanda será más baja. Esto es importante preverlo cuando se haya de reponer un producto. Una política de compras inapropiada podría derivar en tener los productos almacenados en el almacén más tiempo del necesario, lo que supone un riesgo de deterioro del material y un coste para la empresa, al tener invertido el dinero en los productos. Inversión que se podría destinar a otras operaciones.

5.2 BASE DE DATOS SQL

Tras haber consultado las necesidades de la empresa, y de común acuerdo con ellos con respecto a la distribución y organización de la solución adoptada para el control del stock, se usan tres tablas diferentes conectadas en una base de datos en *Microsoft Azure*.

Se ha seleccionado la opción de realizar el trabajo en *Microsoft Azure*, frente a otras opciones, por simplicidad, puesto que la empresa lo emplea ya para otras funciones. Así pueden estar todos los datos unificados en un mismo sitio desde el principio. Sin necesidad de tener que replicar lo expuesto en el futuro para poder utilizarlo. Se ha creado una base de datos llamada Control-stock, a la cual se accede mediante *Azure Data Studio*. De esta manera se pueden crear tres tablas independientes, una que recoja los datos del inventario de los productos almacenados, la otra los datos de las compras y las ventas, registrando las entradas y salidas de material, a su vez sirviendo de ayuda para calcular el inventario y proveer de datos a la primera tabla, y la última tabla que contenga toda la información relevante de los productos contenidos en el almacén.

El funcionamiento de las tablas consistirá en que, se registrarán como entradas los productos recibidos cada vez que se haga una compra y como salidas los productos enviados cuando se realice una venta. El total de unidades almacenadas se obtendrá restando las entradas menos las salidas. Cada vez que se introduzca un nuevo producto en el catálogo del almacén habrá que incluirlo en el listado. Además, es interesante registrar las fechas de realización de los pedidos y la fecha de recepción, tanto cuando se hace un pedido para el almacén como cuando un cliente realiza un pedido, para así conocer la evolución de los plazos de envío. Con los datos registrados podremos medir 3 KPI de gran interés. En primer lugar, la evolución de los niveles de stock y la tasa de reposición. En segundo lugar, el stock de seguridad.

A continuación, se pasará a desarrollar todas las actividades relacionadas con la base de datos SQL que se consideran de interés para el proyecto y la correcta utilización del sistema de control de stock, una vez esté en funcionamiento por parte de la empresa. Esto incluye:

- Creación de las nuevas tablas.
- Inserción de registros en las tablas.
- Eliminación de registros duplicados en las tablas.
- Actualización de uno o varios atributos de los registros ya existentes.

- Realización de consultas, para extraer y visualizar información de la base de datos.
- Modificación del formato de las tablas, que puede consistir en añadir o quitar columnas o modificar el tipo de dato de la columna.

Es conveniente especificar que la mayoría de los casos mostrados a continuación en este apartado, a excepción del código para crear las tres tablas utilizadas en el proyecto, constituyen ejemplos sencillos de casos que pueden suceder cuando se opere con las tablas creadas, algunos de los comandos mostrados serán más adelante implementados en *Power Automate Desktop* para la operación automatizada de la base de datos. Han de tomarse como casos de referencia, por lo tanto, perfectamente pueden aparecer casos más complejos que requieran un mayor conocimiento de la gestión de bases de datos SQL. Aunque con lo ilustrado en estos ejemplos, y los flujos desarrollados más adelante en el apartado de automatización, debería ser más que suficiente para que un usuario con conocimientos básicos pudiera operar el sistema de control de inventario sin complicaciones. Por ese motivo, se considera que la inclusión de los ejemplos en este proyecto cumple una doble función, ilustrar el código posteriormente implementado, y guiar a un usuario que necesite interactuar directamente con la base de datos, sin necesidad de hacer uso de los flujos de *Power Automate* previamente implementados.

5.2.1 CREACIÓN DE LAS TABLAS

La primera tabla CONTROL_DE_INVENTARIO, sirve para medir los niveles actuales de inventario. Para facilitar la lectura de la información, y el manejo de la tabla, únicamente contendrá las referencias de los productos, el nombre de los productos, el nivel de inventario del producto, y el estado del inventario. Cuando hablamos de referencias, nos referimos a las referencias internas asignadas por la empresa a cada uno de sus productos. Estas referencias pueden variar con respecto a la referencia adjudicada por el proveedor o por el fabricante a ese mismo producto. Se puede dar el caso de que 2 fabricantes distintos asignen la misma referencia a productos que no tienen nada que ver entre ellos, por eso trabajaremos con la referencia interna para asegurar que no existe ninguna duplicidad en la asignación de

referencias. El atributo estado del inventario ayudará a aportar claridad a la hora de visualizar la tabla. Por ese motivo se ha elegido que sea un tipo de dato char, que tenga siempre la misma longitud de palabra, para asegurar que todos los registros de estado están estandarizados. Permitirá conocer cómo se encuentra el inventario de cada uno de los productos, aportando además información adicional al nivel de inventario. Se han definido 6 estados predeterminados, 6 casos que se pueden dar para definir cómo se encuentra el inventario de cada producto. Los estados son:

- **Stock correcto (OK):** El inventario es suficiente y se encuentra por encima del stock de seguridad con suficiente margen para cumplir con los pedidos que puedan ir entrando.
- **Stock bajo (SB):** El inventario se sigue encontrando por encima del stock de seguridad, pero ha disminuido por debajo del umbral situado en el 110% del stock de seguridad. Este estado permite tener controlados que productos puede haber necesidad de pedir próximamente, aunque de momento no haga falta. Simplemente ayuda a que el encargado pueda estar atento y poner el foco de atención en esos productos.
- **Reponer stock (RS):** El stock ha disminuido por debajo del stock de seguridad y es necesario generar un aviso para alertar al responsable que se debe realizar un pedido de material para reponer el stock.
- **Aviso generado (AG):** Se ha enviado un aviso al responsable y el sistema está en espera de registrar la compra que aumente el stock. Este estado es fundamental para evitar que se estén generando avisos continuamente, mientras el inventario esté por debajo del stock de seguridad. En caso de no existir este estado, cada vez que hubiera una nueva venta y el inventario se encontrase por debajo del stock de seguridad, se generaría un nuevo aviso, pudiendo darse el caso de múltiples avisos que simplemente generarían confusión, siendo contraproducentes.
- **Pedido realizado (PR):** Se ha registrado ya un pedido que reponga el stock de un producto, pero al no haberse recibido todavía, el stock disponible sigue siendo inferior al stock de seguridad. Para que el sistema pueda saber que el pedido de reposición ya se ha realizado, se debe comprobar que la suma de las unidades

pedidas de las compras de ese producto es superior a la suma de las unidades recibidas de las compras del producto, quedando patente que existe una desigualdad entre los valores de las unidades pedidas y las unidades recibidas de las compras del producto. Eso implica que el pedido de compra ya se ha realizado y se ha registrado en el sistema, pero todavía no ha sido recibido en el almacén. Por ese motivo existe esa diferencia entre las unidades pedidas y las unidades servidas del producto.

- Stock agotado (SA): El inventario ha llegado a 0, no quedaría ninguna unidad disponible en el almacén para la venta. Este estado debe intentar evitarse, dado que alcanzarlo implica dos cosas importantes. La primera, una posible mala planificación que no ha permitido reponer el stock a tiempo, y la segunda, que el producto ha de ser marcado en la plataforma como no disponible por falta de stock, generando la pérdida de potenciales ventas.

El uso de los estados tiene su origen en intentar dotar al sistema de cierta planificación temporal. Como se describe en la descripción del estado “Aviso generado”, se desea poder registrar que ya se ha enviado un aviso al responsable al estilo de los sistemas de planificación de los requerimientos de material (MRP, *Material Requirements Planning*). Un sistema MRP no solo tiene en cuenta el estado actual de los inventarios y las necesidades de material en ese momento, sino que evalúa a lo largo de un período de tiempo cómo se desarrolla la evolución temporal de los requisitos de material para producción. Cuantas unidades o cantidad de cada materia prima se van a necesitar, cuántas de ellas se tienen disponibles en stock al inicio del periodo evaluado y qué pedidos está planificado que se reciban durante el periodo de evaluado [24].

A continuación, se muestra el código para la creación de la tabla de Control de inventario, donde puede verse el listado de los atributos definidos en la tabla (REFERENCIA, PRODUCTO, UNIDADES, ESTADO):

Creación tabla CONTROL DE INVENTARIO

```
drop table CONTROL_DE_INVENTARIO;  
create table CONTROL_DE_INVENTARIO(  
  
    REFERENCIA varchar(15) PRIMARY KEY NOT NULL,  
    PRODUCTO varchar(60) NOT NULL,  
    UNIDADES int,  
    ESTADO char(2)  
  
);
```

Conviene resaltar como en el atributo REFERENCIA se indica la restricción PRIMARY KEY, esto implica que no puede haber dos productos con la misma referencia en la tabla de Control de inventario. Las claves primarias, o principales, implican que el atributo debe contener un valor único, no puede haber dos registros con un mismo valor para ese atributo dentro de la misma tabla, y además no puede contener valores NULL, se debe introducir un valor para ese atributo siempre que se añada un registro [24]. Por lo tanto, siempre que se defina una clave primaria se debe especificar como NOT NULL. Si se intentase introducir un registro con una referencia ya existente en la tabla, daría error y no se añadiría el registro a la tabla. No se ha fijado ningún otro atributo con ese criterio para evitar el riesgo de que exista una coincidencia. Los dos casos más probables de existir una coincidencia en el valor de los atributos en las tablas creadas son el PRODUCTO, que aparece en la tabla de Control de inventario y en el Maestro de artículos, y la REF_FARBICANTE, que se verá a continuación cuando se explique cómo se ha creado la tabla que contiene el Maestro de artículos. En el caso del nombre del producto, puede suceder que dos productos tengan el mismo nombre, si, por ejemplo, se realiza una actualización de este, que mantiene el nombre, pero cambia la referencia. En cuanto a la referencia del fabricante, como ya se ha comentado, podrían existir dos marcas que hayan asignado la misma referencia para dos productos diferentes, de sus respectivos catálogos.

La segunda tabla, COMPRAS_Y_VENTAS, registra cada una de las compras y las ventas realizadas con respecto a los productos almacenados. Las compras las realizará la empresa para poder abastecer su almacén de productos, que después pondrá a la venta en su web.

Servirán para crear un stock de productos, y más tarde ir reponiendo ese stock a medida que se vaya agotando debido a las ventas. Las ventas recogerán los pedidos que harán los clientes a la empresa a través de su web.

A continuación, se muestra el código para la creación de la tabla de Compras y ventas, donde puede verse el listado de los atributos definidos en la tabla (TIPO, REFERENCIA, N_PEDIDO, FECHA_PEDIDO, FECHA_RECEPCION, UNIDADES_PEDIDAS, UNIDADES_SERVIDAS):

Creación tabla COMPRAS Y VENTAS

```
drop table COMPRAS_Y_VENTAS;  
create table COMPRAS_Y_VENTAS(  
  
    TIPO char(1),  
    REFERENCIA varchar(15) NOT NULL,  
    N_PEDIDO int,  
    FECHA_PEDIDO date,  
    FECHA_RECEPCION date NULL,  
    UNIDADES_PEDIDAS int NOT NULL,  
    UNIDADES_SERVIDAS int NULL,  
  
);
```

En primer lugar, se indicará de qué tipo de operación se trata, una compra de material (C) o una venta (V). Se recogerán los datos del producto mediante su referencia, gracias a la cual podremos conocer de qué producto de los existentes en la tabla de Maestro de artículos (tercera tabla) se trata. El número de pedido servirá para poder corroborar con la plataforma que el producto introducido es el correcto, y que concuerda lo que hay registrado con las facturas emitidas por los proveedores. La fecha del pedido y la fecha de recepción permiten conocer cuántos días se ha tardado en servir un pedido. Esto es útil en ambos casos, para las compras permite saber cuántos días tardan los proveedores en servir un pedido de media, ver si ha sucedido un problema con algún pedido, y conocer cuándo se ha de realizar un pedido para reponer el stock de un producto, con el tiempo suficiente para evitar que el stock de ese producto se agote o si fuera necesario aumentar el stock de seguridad de un determinado producto, para prevenir el desabastecimiento del producto, y para las ventas permite a la

empresa medir si son eficaces sirviendo un pedido a sus clientes, y tomar acciones correctivas en caso de que los datos obtenidos no sean satisfactorios. Por último, las unidades pedidas y las unidades servidas permitirán conocer si un pedido realizado se ha podido servir al completo. Cuando se introduzcan las unidades, se diferenciará entre compras y ventas, siendo las compras positivas y las ventas negativas. Se ha decidido hacerlo de esta forma para que resulte más intuitivo y que en el momento que se extraigan los datos para visualizarlos sea más gráfico y se pueda diferenciar con mayor facilidad.

Para la tabla de Compras y ventas se ha valorado la opción de poner una clave principal formada por la combinación de la referencia y el número de pedido, como la mostrada a continuación. La combinación de dos o más atributos dentro de la clave principal sirve para generar una clave compuesta. La clave compuesta garantiza que la combinación de atributos es única, pero por separado los valores de los distintos atributos que componen la clave pueden estar repetidos.

```
drop table COMPRAS_Y_VENTAS;  
create table COMPRAS_Y_VENTAS(  
  
    TIPO char(1),  
    REFERENCIA varchar(15) NOT NULL,  
    N_PEDIDO int,  
    FECHA_PEDIDO date,  
    FECHA_RECEPCION date NULL,  
    UNIDADES_PEDIDAS int NOT NULL,  
    UNIDADES_SERVIDAS int NULL,  
    PRIMARY KEY (REFERENCIA, N_PEDIDO)  
  
);
```

Se descarta finalmente esta opción debido a la posibilidad de que existan pedidos que se sirvan incompletos, y haya que añadir más de un registro, para la misma compra, con números de pedido y referencias idénticas. En este caso, ese segundo pedido tendrá las unidades solicitadas con el valor 0, así podemos comprobar si un pedido está completo, al sumar las unidades pedidas y las unidades servidas de ambos registros y ver si la suma de

unidades pedidas y de unidades servidas coincide. Esto se verá con más detalle en el apartado 5.2.3.

La tercera tabla consiste en el MAESTRO_DE_ARTICULOS. Aquí se tiene registrada toda la información pertinente a los distintos artículos con los que se trabaja en el almacén. Tener la información unificada en una única tabla, no habiendo datos duplicados, nos permite que el inventario siga funcionando sin alteraciones en caso de que haya una modificación en alguno de los productos. Si se decide modificar el proveedor de un producto en concreto, únicamente se necesita modificar el registro pertinente en el maestro de artículos, las otras tablas no se ven alteradas. En caso de que se desee obtener algún dato de los productos, se puede acceder a esta tabla y sacar el atributo que contenga lo que se desea saber.

A continuación, se incluye el código necesario para la creación de la tabla Maestro de artículos, donde puede verse el listado de los atributos definidos en la tabla (REFERENCIA, PRODUCTO, PRECIO, STOCK_SEGURIDAD, DISTRIBUIDOR, FABRICANTE, REF_FABRICANTE):

Creación tabla MAESTRO DE ARTÍCULOS

```
create table MAESTRO_DE_ARTÍCULOS(  
  
    REFERENCIA varchar(15) PRIMARY KEY NOT NULL,  
    PRODUCTO varchar(60) NOT NULL,  
    PRECIO decimal(7,2),  
    STOCK_SEGURIDAD int,  
    DISTRIBUIDOR varchar(50),  
    FABRICANTE varchar(50),  
    REF_FABRICANTE varchar(15),  
  
);
```

Al igual que en el caso de la tabla de Control de inventario, la referencia se ha marcado con la restricción PRIMARY KEY, ya que es un atributo único que no puede estar duplicado en la tabla. Esto facilita buscar los datos de los productos en la tabla. Conociendo la referencia, podemos acceder al registro y obtener cualquier otro atributo. Un dato clave en esta tabla será el stock de seguridad, que marcará cuándo hay que realizar un pedido de compra para reponer el stock de un producto. Inicialmente, al no disponer de datos históricos para fijar

su valor, habrá que definirlo manualmente, hasta tener un mínimo de pedidos que permitan fijar un valor razonable. Si se calculara inicialmente, estaría sujeto a grandes variaciones, porque cada registro de la tabla COMPRAS_Y_VENTAS perteneciente al producto tendría un peso excesivo en el cálculo del stock de seguridad. A medida que se tengan más valores, el peso de cada registro disminuye, permitiendo medir con mayor exactitud los valores.

5.2.2 ADICIÓN DE NUEVOS REGISTROS EN LAS TABLAS

Para poder añadir nuevos registros en cada una de las tablas, será necesario su respectivo código que incluya los atributos indispensables. Como se ha indicado antes, en el código ciertos atributos no pueden quedar vacíos al introducir un registro, y se les tiene que adjudicar un valor para poder insertarlo correctamente.

Para añadir nuevos productos a la tabla de Control de Inventario, podemos utilizar el siguiente código, asignando el valor correspondiente a cada uno de los atributos.

```
insert into CONTROL_DE_INVENTARIO (REFERENCIA, PRODUCTO, UNIDADES, ESTADO)
values (1, 'producto1', 1, 'OK'),
       (2, 'producto2', 1, 'OK')
;
```

Cada fila corresponderá con un producto único, no pudiendo existir duplicados. Si intentásemos añadir otro producto distinto con la misma referencia, recibiríamos un error y no se añadiría el registro.

Para añadir un registro en la tabla Compras y Ventas, cada vez que se haga una compra o una venta, usaremos el siguiente código, indicando sus atributos.

```
insert into COMPRAS_Y_VENTAS (TIPO, REFERENCIA, N_PEDIDO, FECHA_PEDIDO, FECHA_RECEPCION,
UNIDADES_PEDIDAS, UNIDADES_SERVIDAS)
values ('C', 1, 1, '06/21/2022', '06/23/2022', 5, 5)
;
```

```
insert into COMPRAS_Y_VENTAS (TIPO, REFERENCIA, N_PEDIDO, FECHA_PEDIDO, FECHA_RECEPCION,
UNIDADES_PEDIDAS, UNIDADES_SERVIDAS)
values ('V', 1, 1, '06/21/2022', NULL, 3, NULL)
;
```

No todos los datos podrán ser introducidos directamente. Para saber la fecha de recepción y las unidades servidas, habrá que esperar a que se reciba, y se dé de alta en el almacén, el pedido. En ese momento, se podrá actualizar el registro con los datos que faltaban por introducir. Puede darse el caso de que, por falta de stock de los proveedores, un pedido se entregue incompleto, en ese supuesto, hay que poder registrarlo, anotando las unidades recibidas y creando un nuevo registro cuando se reciban las unidades restantes.

En el código anterior, el primer ejemplo muestra el código de una compra, y el segundo, una venta. En el ejemplo de la compra, podemos ver como se introducen todos los campos. Por el contrario, en la venta se dejan vacíos la fecha de recepción y las unidades servidas, a la espera de finalizar con éxito el pedido. Hay que resaltar que lo habitual cuando se produzca una compra o una venta será dejar esos dos campos vacíos hasta que se finalice la entrega, y que el caso mostrado en el ejemplo de la compra es simplemente ilustrativo.

En la tabla de Maestro de artículos se añadirán los nuevos registros usando el código:

```
insert into MAESTRO_DE_ARTÍCULOS (REFERENCIA, PRODUCTO, PRECIO, STOCK_SEGURIDAD, DISTRIBUIDOR, FABRICANTE, REF_FABRICANTE)
values (1, 'producto1', 5, 1, 'distribuidor1', 'fabricante1', 11)
;
```

5.2.3 ELIMINACIÓN DE REGISTROS Y ELEMENTOS DUPLICADOS

La eliminación de registros de la base de datos puede ser necesaria en dos casos primordiales, que existan registros duplicados o que se desee borrar un registro correspondiente a un producto. Cumpliendo con esos dos supuestos, es posible mantener la base de datos actualizada y sin registros que corrompan la veracidad de los datos, o que generen información redundante que dificulte visualizar lo relevante.

Eliminar los elementos duplicados es fundamental para evitar que se corrompan los datos de la tabla de Compras y ventas. Al no tener ningún tipo de clave primaria, se podría introducir un registro en más de una ocasión, sin ningún tipo de restricción. Como se ha indicado en el apartado 0, no se ha implementado la opción de incluir la clave primaria combinando los atributos pedido y referencia por el riesgo de que haya un pedido incompleto y sea necesario

añadir un registro adicional para complementar los datos del registro adicional. En el supuesto planteado, se añadiría un registro adicional con los datos idénticos al original, simplemente registrando las unidades pedidas como 0 y las unidades recibidas y la fecha de recepción con los datos del segundo envío que completa al primero. Como ya se ha mencionado, en las otras dos tablas de control de inventario no debería ser posible gracias al uso de PRIMARY KEY en el atributo referencia. Esto nos permite asegurar que no existen registros duplicados en ninguna de las 3 tablas.

El código siguiente permitiría eliminar los registros duplicados de la tabla de Compras y Ventas:

```
with cte as (  
    select TIPO, REFERENCIA, N_PEDIDO, FECHA_PEDIDO, FECHA_RECEPCION, UNIDADES_PEDIDAS,  
    UNIDADES_SERVIDAS,  
  
        ROW_NUMBER() over (  
            partition by  
  
TIPO, REFERENCIA, N_PEDIDO, FECHA_PEDIDO, FECHA_RECEPCION, UNIDADES_PEDIDAS,  
UNIDADES_SERVIDAS  
order by TIPO, REFERENCIA, N_PEDIDO, FECHA_PEDIDO, FECHA_RECEPCION, UNIDADES_PEDIDAS,  
UNIDADES_SERVIDAS  
        ) as row_num  
    from  
        COMPRAS_Y_VENTAS  
)  
delete from cte  
where row_num > 1  
  
select * from COMPRAS_Y_VENTAS;
```

El otro supuesto planteado, eliminar un registro correspondiente a un producto puede suceder por varios motivos, aunque el principal será cuando se desee eliminar un registro obsoleto. Una vez un producto es descatalogado y su stock ha sido agotado, es conveniente retirarlo de la base de datos, para liberar ese espacio y evitar confusiones futuras. Así aseguramos que solamente almacenamos en la base de datos la información sobre los productos activos. Para ello, habrá que eliminar el registro asociado a su referencia tanto en la tabla de Control de inventario como en la del Maestro de artículos, como se realiza en los ejemplos mostrados a continuación.

```
delete from MAESTRO_DE_ARTÍCULOS where REFERENCIA = 0045;
```

```
delete from CONTROL_DE_INVENTARIO where REFERENCIA = 0045;
```

5.2.4 ACTUALIZACIÓN DE REGISTROS

Actualizar un registro nos permitirá modificar los valores de sus atributos. El ejemplo más evidente sería actualizar los valores de los atributos de los pedidos cuando se reciban, modificando la fecha de recepción y las unidades servidas, completando así el registro. Además, cada vez que se reciba un pedido habrá que recalcular y actualizar el valor del inventario disponible. Por último, porque su probabilidad de suceder es menor, podemos necesitar modificar los atributos de un producto en la tabla de Maestro de Artículos. Este caso podría suceder si se modifica el distribuidor de un producto o si periódicamente se decide revisar el stock de seguridad de los productos. El ejemplo siguiente muestra el código que permitirá actualizar los atributos de un registro de la tabla de Compras y Ventas y de la tabla de Control de Inventario:

```
update COMPRAS_Y_VENTAS
set UNIDADES_SERVIDAS = 3,
    FECHA_RECEPCION = '06/26/2022'
where TIPO = 'V'
;

update CONTROL_DE_INVENTARIO
set UNIDADES = 3 where REFERENCIA = 1
;
```

El uso del *where* es fundamental, en caso contrario, no se especificaría qué registro o registros han de ser actualizados, se actualizarían todos los registros de la tabla. En las tablas de Control de Inventario y de Maestro de Artículos es suficiente con indicar la referencia del producto en el *where*, debido a las claves primarias implementadas al crear las tablas, pero para la tabla de Compras y Ventas necesitaremos indicar el tipo de actividad, el número de pedido y la referencia del producto, como mínimo, para asegurarnos que estamos modificando el registro correcto.

El momento en que se actualizan las unidades dentro de la tabla de Control de Inventario resulta crítico para asegurar que no existen fallos dentro de la cadena de suministro. Por ese motivo, no podemos actuar de la misma manera si se trata de una compra de material que si se trata de una venta. En las compras solamente se deben sumar los productos una vez que se reciban en el almacén, que es cuando ese stock pasa a estar disponible para la empresa y puede suministrarlo a los clientes. En las ventas funciona al contrario, el inventario debe ser actualizado en el instante en que se realiza una venta, pese a que las unidades del producto todavía se encuentran dentro del almacén pendientes de ser enviadas. Esto es debido a que, si no se hiciera de ese modo, se podría vender más unidades de un producto de las que realmente se disponen, porque se estarían contando como disponibles unas cantidades de un producto que ya están comprometidas con un determinado cliente.

Debido a lo poco común que serán las actualizaciones de los registros en la tabla Maestro de Artículos, en comparación con las otras dos tablas, en las que las actualizaciones serán diarias, las modificaciones en esta tabla se harán manualmente. Aun así, incluimos un ejemplo para ilustrar una posible actualización:

```
update MAESTRO_DE_ARTÍCULOS  
set FABRICANTE = 'fabricante3' where REFERENCIA = 1  
;
```

5.2.5 REALIZACIÓN DE CONSULTAS

Si se desea realizar una consulta de la información contenida en las tablas para obtener los datos almacenados, es necesario ejecutar un código especificando la tabla de la que se pretende extraer la información, y las condiciones para filtrar los datos si no se desea la tabla completa. Las consultas devuelven los datos de las tablas, ya sean totales o con alguna condición específica, como podría ser ver qué productos tienen el stock bajo, las ventas de un producto concreto durante el último mes, o todos los productos que provienen de un fabricante. Al estar en funcionamiento, el estado de los datos es cambiante, por lo que, en función del momento en que se ejecute la consulta, los resultados serán distintos. A continuación, se presenta el código para realizar las consultas de los ejemplos nombrados:

```
select REFERENCIA from CONTROL_DE_INVENTARIO where ESTADO = 'SB';
```

```
select * from COMPRAS_Y_VENTAS where REFERENCIA = 1 and FECHA_PEDIDO between '06/20/2022' and '07/20/2022';
```

```
select * from MAESTRO_DE_ARTÍCULOS where FABRICANTE = 'fabricante1';
```

5.2.6 MODIFICACIÓN DE LAS TABLAS

Finalmente, podría darse el caso que, para variar el tipo de información recogido en las tablas o porque con el funcionamiento se detecte alguna necesidad no prevista anteriormente, se desee modificar el formato de la tabla. Bien podría ser para añadir, o quitar, una columna, porque se ha descubierto la importancia de recoger algún dato que no se estaba registrando hasta el momento, o, por el contrario, se ha decidido que no tiene interés seguir recogiendo algún dato que carece de utilidad. Otro posible escenario sería modificar el formato de un atributo en la tabla, porque el formato actual no permite recoger correctamente los datos. Para cualquier modificación de este tipo se utilizará el comando ALTER TABLE. A continuación, veremos algunos ejemplos:

Añadir una columna nueva

```
alter table MAESTRO_DE_ARTÍCULOS  
add TELEFONO BIGINT;
```

Eliminar una columna existente

```
alter table MAESTRO_DE_ARTÍCULOS  
drop column TELEFONO;
```

Modificar el formato de una columna

```
alter table CONTROL_DE_INVENARIO  
alter column INVENTARIO bigint;
```

Capítulo 6. AUTOMATIZACIÓN DE PROCESOS

En este capítulo se plantea el método para poder gestionar los procesos de la base de datos de manera desatendida, pudiendo funcionar sin supervisión y liberando, al encargado de gestionar el almacén, de tareas tediosas. Para ello se propone la manera de automatizar la base de datos previamente diseñada, haciendo uso de *Power Automate*.

6.1 AUTOMATIZACIÓN DE LA BASE DE DATOS

Una vez se ha concluido el desarrollo de la base de datos Control de stock en SQL, y las distintas tablas que la componen, Control de Inventario, Compras y Ventas, y Maestro de Artículos, es el momento de automatizar las operaciones para poder gestionarla. La automatización de los procesos permitirá al sistema operar de manera independiente en muchas de sus funciones. En este caso se simularán las acciones que llevaría a cabo una persona para poder gestionar correctamente el inventario. Se registrarán los pedidos de compras y ventas, y se modificará de forma acorde el nivel de inventario de cada uno de los productos afectados.

Es importante recalcar que no todos los productos tendrán el mismo nivel de rotación de inventario, lo cual significa que no tendrán el mismo nivel de demanda ni aparecerán con la misma asiduidad en los pedidos gestionados. Mayor nivel de demanda implicará que el stock del producto se tendrá que reponer más a menudo, o almacenar mayores cantidades del artículo, aunque eso tendrá los inconvenientes de necesitar mayor cantidad de espacio y dinero invertido en el stock almacenado.

En el proceso de automatización de la herramienta se ha decidido utilizar *Power Automate Desktop*. Con esta aplicación se logra automatizar gran cantidad de las actividades de escritorio. Una de las principales ventajas de su uso proviene de la facilidad para enlazar acciones con distintos programas creados por Microsoft, como puede ser la gestión de correos con Outlook, gestionar hojas de Excel y documentos Word, o conectar los flujos

generados en *Power Automate* con *Microsoft SQL Server*, entre otras funciones posibles. Esto será de gran utilidad a la hora de enlazar los flujos generados con la base de datos creada en el apartado anterior, y automatizar algunas de las actividades relacionadas con su correcta gestión.

Para poder conectar los flujos diseñados de *Power Automate Desktop* a la base de datos SQL ha sido necesario descargar e instalar en el ordenador el *Microsoft ODBC Driver para SQL Server*. En caso de no haber instalado el controlador antes de ejecutar *Power Automate*, este no logra enlazar a la base de datos, el flujo no se ejecuta correctamente, y se genera un error. Una vez instalado correctamente, no es necesario realizar ningún paso más, automáticamente enlaza *Power Automate* con la base de datos, y se puede empezar a realizar flujos que contengan intercambio de información con la base de datos.

Para la gestión de la base de datos, las principales funciones que se desea automatizar, utilizando *Power Automate Desktop*, son:

- Anotación de los registros de las nuevas compras y las nuevas ventas en la tabla de Control de Inventario, con los datos relevantes sobre el producto, unidades pedidas y recibidas, y fechas de compra y de recepción.
- Actualización del inventario de la tabla de Control de Inventario, calculando los niveles de stock para cada producto.
- Recepción de correos, extrayendo la información relevante de los mismos y de sus archivos adjuntos, descargando y almacenando, en la ubicación deseada, los archivos de interés.
- Redacción de correos para generar avisos automáticos, con los que mantener informado al responsable del estado del inventario.
- Actualización de los registros de las tablas en función de la información recibida y los cálculos realizados.
- Visualización de datos, que permitan evaluar el rendimiento del almacén y generar diferentes análisis.

De las funciones descritas, no todas tendrán su propio flujo de manera independiente, por cuestiones de interés se agruparán ciertas actividades, componiendo flujos más largos con distintos subflujos. Esto es de especial utilidad, ya que permite relacionar tareas, que si bien son independientes, adquieren mayor relevancia cuando se realizan en conjunto. Revisar el correo electrónico y descargar los archivos adjuntos no está directamente relacionado con rellenar la tabla de SQL, pero si juntamos las tareas y aprovechamos los datos obtenidos de los archivos adjuntos para rellenar las órdenes de SQL, se unifican en un solo proceso tareas que, en principio, son independientes.

El cálculo automático del stock de seguridad se excluye inicialmente, y los valores se definirán manualmente por el encargado. Esto es debido a que se quiere evitar que, habiendo pocos pedidos registrados en el sistema, el valor del stock de seguridad sufra grandes fluctuaciones, creando disrupciones en la gestión del almacén.

6.2 GESTIÓN DE PEDIDOS

El principal objetivo de la automatización de la base de datos ha de ser lograr que las variaciones de inventario, con todos los procesos asociados que eso conlleva, se ejecuten de manera automática, para que el encargado no tenga que estar constantemente pendiente de supervisar y registrar cada una de las interacciones que sucedan con la base de datos. Esto se hace para aportar mayor claridad y que se pueda seguir el proceso de manera más sencilla. El proceso principal es un flujo que se ha dividido en diferentes subflujos para así poder operar con mayor facilidad, como se observa en la **¡Error! No se encuentra el origen de la referencia.** Los subflujos presentados en el siguiente listado siguen el orden de ejecución mostrado:

1. Gestión de correos electrónicos
2. Gestión de documentos PDF
3. Registro del pedido en la base de datos
4. Actualización del valor de inventario de los productos incluidos en el pedido

5. Actualización del estado asociado al inventario de los productos incluidos en el pedido

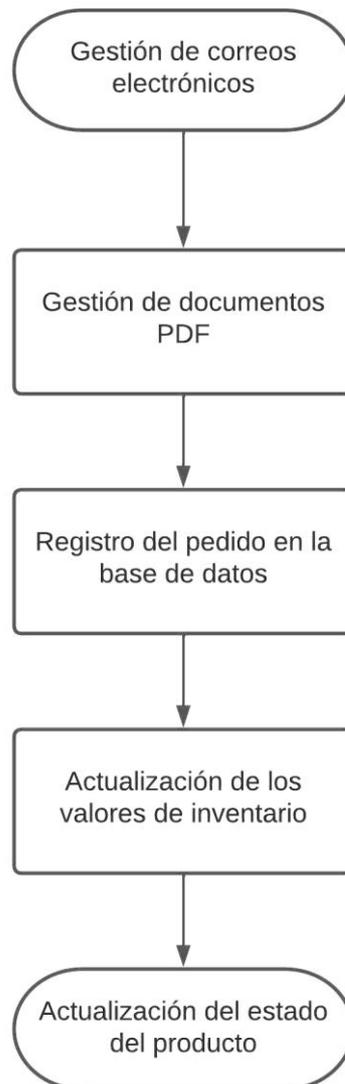


Figura 2. Flujo de gestión de pedidos

El flujo se ha de ejecutar cada vez que entre un correo que contenga un pedido, indistintamente si es de compra o de venta. Debido a la variedad de procesos asociados, desde que entra el correo con el pedido, hasta que finalmente se actualiza el inventario y el estado de los productos, se ha decidido desglosar la explicación de los distintos

subflujos que componen el flujo principal, en los subapartados que se muestran a continuación. Es conveniente aclarar que hay procesos repetidos en algunos subflujos, como puede ser abrir o cerrar las conexiones con la base de datos de *Microsoft SQL Server*, que se han presentado cada vez que son necesarios en los diferentes subflujos, para así aportar mayor claridad. Pero en la realidad solamente se ejecutan una vez, cuando es necesario establecer la conexión por primera vez dentro de flujo, y permanecen abiertos dentro del proceso mientras se requiere su uso en los distintos subflujos, cerrándose una vez ya no se vuelve a necesitar.

6.2.1 GESTIÓN DE CORREOS ELECTRÓNICOS

Una de las principales funciones a automatizar será la de gestionar los correos electrónicos. A medida que se vayan recibiendo, se almacenarán en su respectiva subcarpeta dentro de la Bandeja de entrada, una subcarpeta que se ha nombrado Registro Compras y Ventas. De esta manera se puede revisar que los pedidos estén entrando al correo asociado a la gestión del inventario de manera correcta. Los correos movidos a la carpeta se marcarán como no leídos para que la persona responsable pueda ver más fácilmente que correos ha revisado de manera manual posteriormente. El proceso de recepción y gestión de los correos se muestra en la Figura 3.

Para poder catalogar el correo de manera automática, es necesario estandarizar el contenido. En nuestro caso, definiremos un formato estándar para el asunto del correo. De esta manera, Power Automate buscará palabras clave en el asunto del correo y reconocerá qué correos son los que interesa mover a Registro Compras y Ventas. Los correos que contengan una compra deberían tener el formato, en el asunto, Compra seguido del número de pedido, y las ventas el formato Venta seguido del número de pedido, que es lo que se ha indicado a *Power Automate* que debe buscar para catalogar el correo. Además, se descargarán los archivos adjuntos en el correo y se almacenarán en una carpeta que se ha llamado Compras y Ventas, presumiblemente los correos contendrán un PDF adjunto con el pedido a realizar. De ese archivo adjunto se obtendrá la información para introducir nuevos registros en la tabla de Compras y Ventas. Conociendo el formato del documento, es posible hacer que Power

Automate lo reconozca y sepa dónde ha de localizar los distintos datos a extraer del documento.

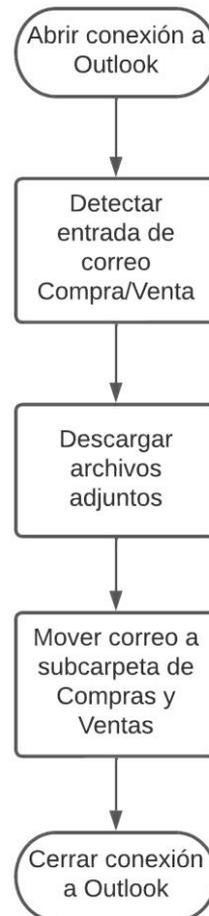


Figura 3. Flujo de gestión de correos

6.2.2 GESTIÓN DE DOCUMENTOS

Si se desea obtener la información contenida en el documento que se ha descargado del correo, es necesario estandarizar el formato del documento para que *Power Automate* pueda identificarlo, saber de dónde ha de extraer los datos y que representan esos datos extraídos. Con esa información se podrá acceder posteriormente a la base de datos y registrar los pedidos en la tabla de Compras y Ventas. Para ello se ha definido un formato de factura, que

contiene una tabla con la información de los productos, partiendo del que usa actualmente la empresa en sus pedidos.

Presupuesto de Pedido			
Tipo	Pedido	Fecha	
Compra	1	23/07/2022	
Referencia	Producto	Unidades	Precio Unitario (€)
1	Producto1	1	5,00 €
2	Producto2	2	4,00 €
3	Producto 3	4	2,00 €
IMPORTE TOTAL SIN IVA (€)			21,00 €

Factura de Pedido			
Tipo	Pedido	Fecha	
Compra	1	23/07/2022	
Referencia	Producto	Unidades	Precio Unitario (€)
1	Producto1	1	5,00 €
2	Producto2	2	4,00 €
3	Producto 3	4	2,00 €
IMPORTE TOTAL SIN IVA (€)			21,00 €

Figura 4. Formatos de documentos utilizados para pedidos

El formato de los documentos, mostrados en la Figura 4, será el mismo tanto para compras como para ventas, con la única diferencia que se muestra en el título de la hoja. Esto depende de si el documento muestra un pedido que se ha realizado, usando la hoja de presupuesto, o el albarán generado una vez se ha entregado el pedido, usando la hoja de factura. Se indica en el propio documento de qué tipo de pedido se trata, el número de pedido y la fecha de realización del pedido. Esa información está situada al inicio del documento, en una primera tabla, encima de la siguiente tabla que contiene la información de los productos. La tabla de los productos consta de varias columnas, que contienen las referencias, los nombres de los productos, las unidades y el precio del producto. Cada producto aparece en una línea separada, para que sea sencillo diferenciarlos, y no se confunda la información entre ellos si

algún dato es muy largo. Generalmente ese problema podría surgir si el nombre del producto es muy largo, aunque en nuestro caso no existe el riesgo porque en los pedidos solamente se registrarán en la tabla de Compras y Ventas la referencia y las unidades de los productos de la tabla.

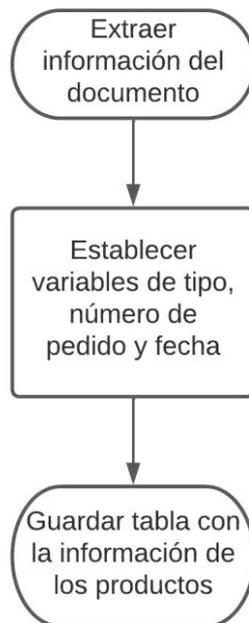


Figura 5. Flujo de gestión de documentos

Como se puede observar en la Figura 5, el primer paso será extraer la información de las tablas contenidas en el documento. Se obtienen las dos tablas y, posteriormente, se ejecuta un bucle que permita almacenar la información de una manera más sencilla para poder operar con ella después, que será diferente en función de la tabla en la que se encuentre. Para ello se plantean dos casos. En el caso de la primera tabla, se obtendrán variables que almacenen el tipo de pedido, el número de pedido y la fecha del pedido. En la segunda tabla, únicamente se obtendrá la variable `TablaProductos`, que almacenará toda la información de la tabla en su conjunto. Esto es debido a que tener toda la información junta nos facilitará añadirla a la base de datos mediante el uso de un bucle. Además de ello, se obtendrá el tipo de documento,

si se trata de un presupuesto o una factura, que será relevante en el momento de añadir la información a la base de datos.

6.2.3 REGISTRO DEL PEDIDO EN LA BASE DE DATOS

El siguiente proceso, que irá enlazado con los anteriores, será introducir la información contenida en la variable TablaProductos del subflujo anterior, en la base de datos. Los pasos a realizar para registrar los productos del pedido en la tabla de Compras y Ventas se muestran en la Figura 6.

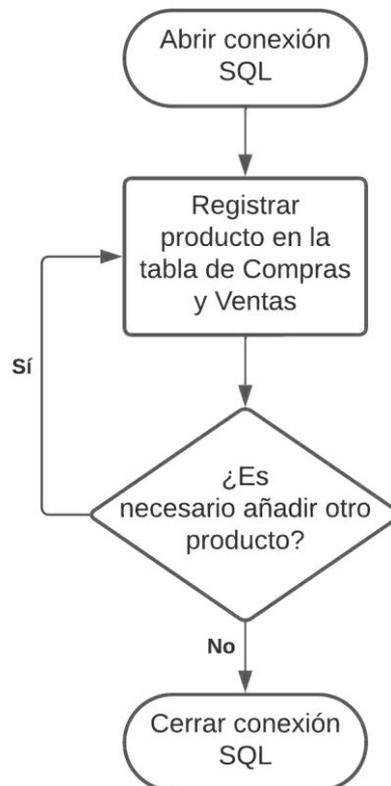


Figura 6. Flujo de registro de pedidos en la tabla Compras y Ventas

Para ello se ejecutará un flujo con un bucle que añada cada una de las filas contenidas en la tabla a la tabla de Compras y Ventas. Las variables registradas en el apartado anterior se utilizarán para definir el tipo de pedido, si es una compra o una venta, el número de pedido

y la fecha del pedido. Estos 3 datos serán los mismos para todos los registros introducidos, dado que todos provienen del mismo documento. La única diferencia que existirá vendrá marcada por lo comentado en el apartado anterior. Si se trata de un presupuesto el documento del que se ha extraído la información, hay que añadir un nuevo registro para cada uno de los productos de la tabla. Por el contrario, si se trata de una factura, simplemente hay que actualizar los registros, que se han realizado previamente al registrar el presupuesto, con los datos de la fecha de entrega y las unidades servidas. Cuando se termine de registrar el pedido en la base de datos se podrá pasar a actualizar el inventario.

6.2.4 ACTUALIZACIÓN DEL INVENTARIO

La actualización del stock de un producto se deberá calcular cada vez que se produzca una compra o una venta, una vez registrada en la base de datos la información contenida en el documento que acredite ese pedido. Es decir, cualquier movimiento de material que afecte a la cantidad de unidades almacenadas. Para poder conocer la cantidad de unidades disponibles en el almacén, una vez ejecutado el flujo, primero obtendremos la cantidad de unidades recibidas en las compras y la cantidad de unidades pedidas en las ventas. Se ha decidido hacerlo de esta manera para asegurar que únicamente se contabilizan las unidades en ese momento dentro del almacén, y evitar que se puedan vender más unidades de las que se tienen. Una vez obtenidos los dos valores, los sumaremos para conocer la diferencia entre ambos. Los dos valores se suman debido a que se ha decidido introducir el valor de las unidades en las compras como positivo en la base de datos, se están añadiendo unidades al almacén, y las unidades en las ventas como negativo, dado que se están extrayendo unidades del almacén. De esta manera, si se extrae la información de la base de datos para poder realizar informes, se puede visualizar más fácilmente lo que está pasando en el almacén. Si el sistema de control de stock está funcionando correctamente, la cantidad siempre ha de ser mayor o igual a 0. En caso contrario, si el número fuera negativo, significaría que se han vendido más unidades de las que se tienen disponibles en el almacén. Se puede observar el proceso en la Figura 7.

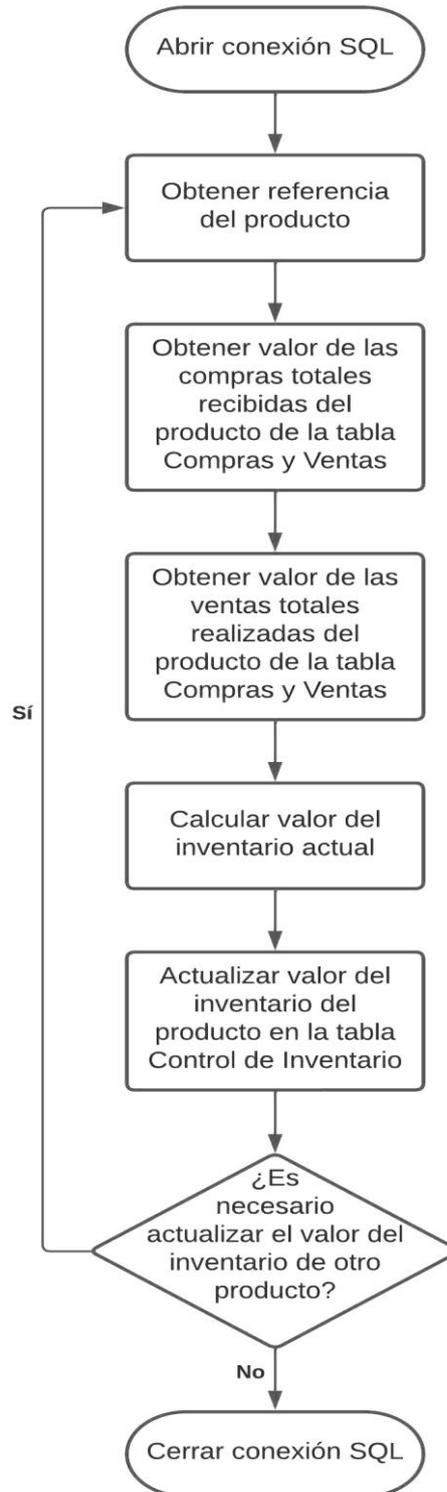


Figura 7. Flujo de actualización del valor del inventario de un producto

Para lograrlo, se abre una conexión a la base de datos. De esta manera, es posible interactuar con la base de datos desde *Power Automate*, ejecutando consultas que permitan obtener la información de las compras y las ventas para uno, o varios, productos, en función de lo que se desee calcular. Se guardan los valores extraídos de la tabla en sendas variables llamadas Compras y Ventas, respectivamente. Después se realiza el cálculo en una tercera variable llamada Inventario, sumando las dos variables anteriores de compras totales y ventas totales del producto, y, por último, se actualiza el valor de las unidades del producto seleccionado en la tabla de Control de Inventario, fijando las unidades como el valor que se ha obtenido para la variable Inventario, y se cierra la conexión SQL. En caso de que se estuvieran actualizando varios registros, las variables Compras, Ventas, e Inventario serían una tabla de una única columna, en lugar de variables numéricas. Al realizar los cálculos dentro del flujo de *Power Automate* se facilita las consultas de SQL, simplificando el código utilizado y facilitando la comprensión futura del trabajo desarrollado.

En todos los flujos generados en *Power Automate* se ha incluido una acción del tipo En error del bloque, colocada detrás de la acción abrir conexión SQL y terminada justo antes de la acción de cerrar conexión SQL. Esto permite asegurar que, en caso de que ocurra un error, el flujo se va a terminar de ejecutar hasta el final y la conexión se va a cerrar antes de terminar. El objetivo de esta acción es evitar que se queden conexiones abiertas y se puedan producir modificaciones involuntarias en los programas asociados. En el caso de la base de datos SQL, un error que generara modificaciones incontroladas de los datos podría ser fatal e inutilizar toda la estructura de control de inventario.

6.2.5 ACTUALIZACIÓN DEL ESTADO DEL INVENTARIO DEL PRODUCTO

Al igual que se actualiza el inventario, es necesario actualizar el estado del inventario de ese producto. Como ya se menciona en el apartado Tras haber consultado las necesidades de la empresa, y de común acuerdo con ellos con respecto a la distribución y organización de la solución adoptada para el control del stock, se usan tres tablas diferentes conectadas en una base de datos en *Microsoft Azure*.

Se ha seleccionado la opción de realizar el trabajo en *Microsoft Azure*, frente a otras opciones, por simplicidad, puesto que la empresa lo emplea ya para otras funciones. Así pueden estar todos los datos unificados en un mismo sitio desde el principio. Sin necesidad de tener que replicar lo expuesto en el futuro para poder utilizarlo. Se ha creado una base de datos llamada Control-stock, a la cual se accede mediante *Azure Data Studio*. De esta manera se pueden crear tres tablas independientes, una que recoja los datos del inventario de los productos almacenados, la otra los datos de las compras y las ventas, registrando las entradas y salidas de material, a su vez sirviendo de ayuda para calcular el inventario y proveer de datos a la primera tabla, y la última tabla que contenga toda la información relevante de los productos contenidos en el almacén.

El funcionamiento de las tablas consistirá en que, se registrarán como entradas los productos recibidos cada vez que se haga una compra y como salidas los productos enviados cuando se realice una venta. El total de unidades almacenadas se obtendrá restando las entradas menos las salidas. Cada vez que se introduzca un nuevo producto en el catálogo del almacén habrá que incluirlo en el listado. Además, es interesante registrar las fechas de realización de los pedidos y la fecha de recepción, tanto cuando se hace un pedido para el almacén como cuando un cliente realiza un pedido, para así conocer la evolución de los plazos de envío. Con los datos registrados podremos medir 3 KPI de gran interés. En primer lugar, la evolución de los niveles de stock y la tasa de reposición. En segundo lugar, el stock de seguridad.

A continuación, se pasará a desarrollar todas las actividades relacionadas con la base de datos SQL que se consideran de interés para el proyecto y la correcta utilización del sistema de control de stock, una vez esté en funcionamiento por parte de la empresa. Esto incluye:

- Creación de las nuevas tablas.
- Inserción de registros en las tablas.
- Eliminación de registros duplicados en las tablas.
- Actualización de uno o varios atributos de los registros ya existentes.

- Realización de consultas, para extraer y visualizar información de la base de datos.
- Modificación del formato de las tablas, que puede consistir en añadir o quitar columnas o modificar el tipo de dato de la columna.

Es conveniente especificar que la mayoría de los casos mostrados a continuación en este apartado, a excepción del código para crear las tres tablas utilizadas en el proyecto, constituyen ejemplos sencillos de casos que pueden suceder cuando se opere con las tablas creadas, algunos de los comandos mostrados serán más adelante implementados en *Power Automate Desktop* para la operación automatizada de la base de datos. Han de tomarse como casos de referencia, por lo tanto, perfectamente pueden aparecer casos más complejos que requieran un mayor conocimiento de la gestión de bases de datos SQL. Aunque con lo ilustrado en estos ejemplos, y los flujos desarrollados más adelante en el apartado de automatización, debería ser más que suficiente para que un usuario con conocimientos básicos pudiera operar el sistema de control de inventario sin complicaciones. Por ese motivo, se considera que la inclusión de los ejemplos en este proyecto cumple una doble función, ilustrar el código posteriormente implementado, y guiar a un usuario que necesite interactuar directamente con la base de datos, sin necesidad de hacer uso de los flujos de *Power Automate* previamente implementados.

Creación de las tablas Tras haber consultado las necesidades de la empresa, y de común acuerdo con ellos con respecto a la distribución y organización de la solución adoptada para el control del stock, se usan tres tablas diferentes conectadas en una base de datos en *Microsoft Azure*.

Se ha seleccionado la opción de realizar el trabajo en *Microsoft Azure*, frente a otras opciones, por simplicidad, puesto que la empresa lo emplea ya para otras funciones. Así pueden estar todos los datos unificados en un mismo sitio desde el principio. Sin necesidad de tener que replicar lo expuesto en el futuro para poder utilizarlo. Se ha creado una base de datos llamada Control-stock, a la cual se accede mediante *Azure Data Studio*. De esta manera se pueden crear tres tablas independientes, una que recoja los datos del inventario de los

productos almacenados, la otra los datos de las compras y las ventas, registrando las entradas y salidas de material, a su vez sirviendo de ayuda para calcular el inventario y proveer de datos a la primera tabla, y la última tabla que contenga toda la información relevante de los productos contenidos en el almacén.

El funcionamiento de las tablas consistirá en que, se registrarán como entradas los productos recibidos cada vez que se haga una compra y como salidas los productos enviados cuando se realice una venta. El total de unidades almacenadas se obtendrá restando las entradas menos las salidas. Cada vez que se introduzca un nuevo producto en el catálogo del almacén habrá que incluirlo en el listado. Además, es interesante registrar las fechas de realización de los pedidos y la fecha de recepción, tanto cuando se hace un pedido para el almacén como cuando un cliente realiza un pedido, para así conocer la evolución de los plazos de envío. Con los datos registrados podremos medir 3 KPI de gran interés. En primer lugar, la evolución de los niveles de stock y la tasa de reposición. En segundo lugar, el stock de seguridad.

A continuación, se pasará a desarrollar todas las actividades relacionadas con la base de datos SQL que se consideran de interés para el proyecto y la correcta utilización del sistema de control de stock, una vez esté en funcionamiento por parte de la empresa. Esto incluye:

- Creación de las nuevas tablas.
- Inserción de registros en las tablas.
- Eliminación de registros duplicados en las tablas.
- Actualización de uno o varios atributos de los registros ya existentes.
- Realización de consultas, para extraer y visualizar información de la base de datos.
- Modificación del formato de las tablas, que puede consistir en añadir o quitar columnas o modificar el tipo de dato de la columna.

Es conveniente especificar que la mayoría de los casos mostrados a continuación en este apartado, a excepción del código para crear las tres tablas utilizadas en el proyecto, constituyen ejemplos sencillos de casos que pueden suceder cuando se opere con las tablas

creadas, algunos de los comandos mostrados serán más adelante implementados en *Power Automate Desktop* para la operación automatizada de la base de datos. Han de tomarse como casos de referencia, por lo tanto, perfectamente pueden aparecer casos más complejos que requieran un mayor conocimiento de la gestión de bases de datos SQL. Aunque con lo ilustrado en estos ejemplos, y los flujos desarrollados más adelante en el apartado de automatización, debería ser más que suficiente para que un usuario con conocimientos básicos pudiera operar el sistema de control de inventario sin complicaciones. Por ese motivo, se considera que la inclusión de los ejemplos en este proyecto cumple una doble función, ilustrar el código posteriormente implementado, y guiar a un usuario que necesite interactuar directamente con la base de datos, sin necesidad de hacer uso de los flujos de *Power Automate* previamente implementados.

Creación de las tablas, existen diferentes estados, que se han definido para aportar una manera más gráfica del estado del producto, así el responsable puede conocer cómo se encuentra el estado de cada producto de forma rápida. Como ese estado depende del inventario, cada vez que exista una variación, se debe analizar para ver si es necesario actualizar el estado.

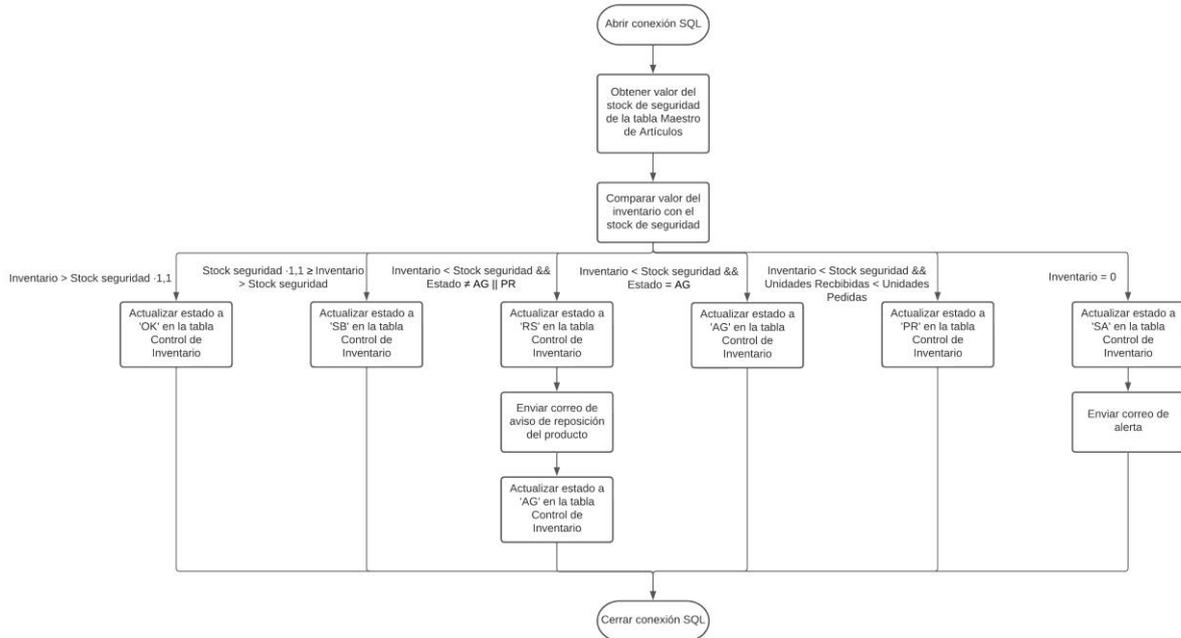


Figura 8. Flujo de actualización de estados de la tabla Control de Inventario

Como puede verse en la **¡Error! No se encuentra el origen de la referencia.**, lo primero que se ha de hacer es obtener el valor del stock de seguridad de los productos que han sido modificados, para poder compararlo con el valor del inventario, que se ha obtenido en el apartado anterior, y obtener el estado actual. El estado actual, el stock de seguridad y el valor del inventario se utilizan para que, comparando ambos valores numéricos, el inventario y el stock de seguridad, y dependiendo del valor que tenga el estado en ese momento, ejecutar un caso u otro. En el flujo se ha implementado un case-switch con distintos supuestos, en función de los valores obtenidos. Los distintos casos se presentan a continuación:

- El inventario es mayor al stock de seguridad multiplicado por 1,1 entonces se actualiza a OK el estado.
- Si el inventario es mayor que el stock de seguridad y menor que el stock de seguridad multiplicado por 1,1, entonces el estado se actualiza a stock bajo (SB).
- Si el inventario es menor que el stock de seguridad y el estado es OK o stock bajo, entonces se actualiza al estado de realizar pedido (RP), se genera un aviso por correo

que avisa al responsable de que es necesario reponer el stock del producto y se vuelve a actualizar el estado a aviso generado (AG).

- Si el inventario es menor que el stock de seguridad y el estado es igual a avisado generado, no se actualiza nada, dado que ya se ha enviado el aviso pertinente.
- Si el inventario es menor que el stock de seguridad y el número de unidades recibidas es menor que el de las unidades pedidas, se actualiza el estado a pedido realizado (PR).
- Si el inventario es igual a 0, se actualiza el estado a stock agotado (SA) y se envía un correo de alerta, notificando al responsable de que no quedan unidades en el almacén y es urgente reponerlas.

Una vez actualizado el estado, concluiría el flujo, siendo este el último subflujo asociado a la gestión de pedidos. La información se ha registrado en la base de datos y se han actualizado los atributos relacionados.

6.3 GESTIÓN DE CONSULTAS

Se ha creado un flujo, que se puede observar en la Figura 9, que permita realizar consultas de la información contenida en las distintas tablas. Para ello, se seleccionará de un cuadro de texto con un desplegable la opción que se desea obtener. Inicialmente se han planteado 5 opciones, que pueden ser fácilmente ampliadas en función de las necesidades detectadas:

1. **Datos del inventario:** Permite conocer el estado del inventario en ese momento. Se mostrarán los datos de la tabla Control de Inventario. Se mostrará una captura momentánea del estado del inventario
2. **Compras:** Al contrario que los datos del inventario, que eran una instantánea en un momento concreto, la consulta permite obtener un histórico de todas las compras de aprovisionamiento de productos para el almacén. Se mostrarán los datos de compras de la tabla Compras y Ventas, ordenados por orden cronológico y número de pedido.

3. Ventas: Al igual que en el caso anterior, permite obtener un histórico de todas las ventas de los productos almacenados que se han realizado. Se mostrarán los datos de ventas de la tabla Compras y Ventas, ordenados por orden cronológico y número de pedido.
4. Compras y ventas: Mostrará toda la información recogida, en orden cronológico y de número de pedido, de la tabla Compras y Ventas. Es una combinación de las dos opciones anteriores, permitiendo obtener todos los movimientos registrados en el almacén. Para las tres opciones disponibles que se pueden obtener de la tabla de Compras y Ventas, conocer el histórico resulta interesante para ver tendencias en el tiempo, qué productos se venden más a menudo, cada cuánto hay que realizar una compra para reponer el stock de un determinado producto, qué productos se venden en mayor cantidad por pedido o qué productos los clientes suelen comprar juntos.
5. Datos de los productos: Permite conocer la información relevante de los productos almacenados. Mantener un listado actualizado disponible para consulta puede resultar útil para agilizar las compras de productos a los proveedores. El listado vendrá ordenado en orden según la referencia.

En las consultas que extraigan información de la tabla de Compras y Ventas (opciones anteriores 2, 3 y 4), se podrá seleccionar las fechas entre las que se quiere extraer la información. Esto se debe a que, con el tiempo, previsiblemente se manejará una ingente cantidad de información, que acabará siendo difícil de manejar para el usuario, si se procesa en su totalidad. Se seleccionará una fecha de inicio y otra fecha de fin, marcando el periodo para el cual se extraerán los datos de la tabla. Esas fechas seleccionadas se guardarán en sendas variables, FechaInicio y FechaFin, que posteriormente se introducirán en el código que se ejecutará en la instrucción de SQL.

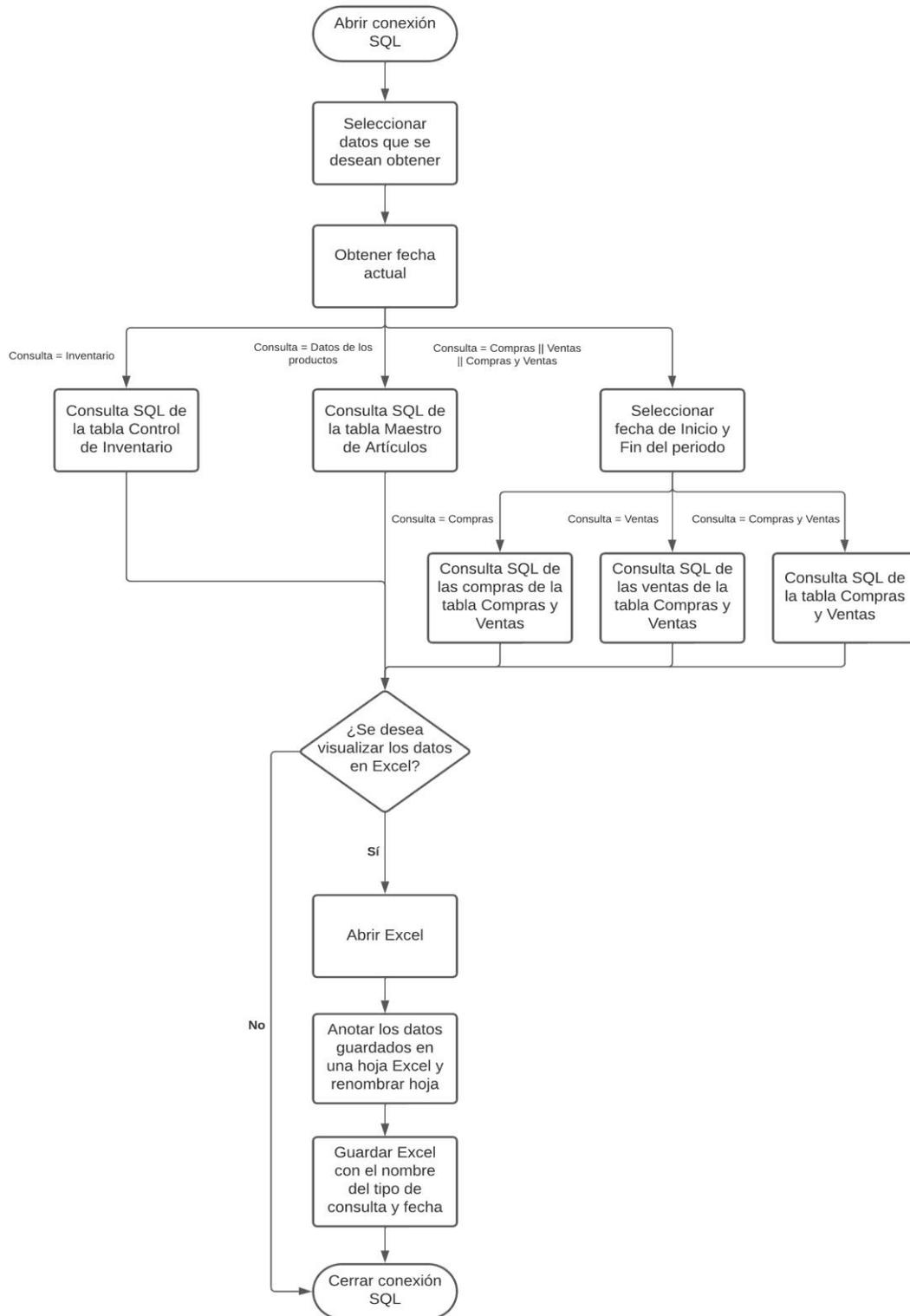


Figura 9. Flujo de gestión de consultas de la base de datos

Para poder obtener los datos de la base de datos Control de stock, se realizará una conexión a SQL. En función de la opción seleccionada en el cuadro de texto, se asignará un valor específico a una variable de salida, Consulta, cuyo valor se comprobará más adelante utilizando una instrucción switch. Cada una de las opciones planteadas en el desplegable constituye uno de los casos planteados en el switch. Cada caso contiene una acción para ejecutar un código específico de SQL que permite obtener la información deseada. Esa información se almacena la variable xxx, del tipo DataTable. Finalmente, podemos decidir qué hacer con la información en la tabla generada, si deseamos visualizar la información en una hoja Excel o no. Si se decide que se quiere pasar la información a Excel, se creará un nuevo archivo para pasar la información, Esto se hará generando una hoja nueva en el archivo, con el nombre de la información que se está visualizando. La información se presentará en distintas columnas, siendo las distintas columnas rellenas las equivalentes a los atributos de las tablas originales de SQL.

6.4 ADICIÓN DE NUEVOS PRODUCTOS AL ALMACÉN

Otro flujo interesante para desarrollar es la adición de nuevos productos a la base de datos, productos con los que no se haya trabajado hasta ese momento en el almacén, y que se desee comenzar a distribuir. Para registrar un nuevo producto correctamente, es fundamental que sus datos estén creados en un registro dentro del Maestro de artículos, y su correspondiente registro asociado esté en la tabla de Control de inventario. Cada vez que se comience a distribuir un nuevo producto por parte de la empresa, y vaya a estar guardado en el almacén, es necesario incluirlo en la base de datos para la monitorización de su nivel de inventario, antes de empezar a almacenarlo. El flujo mostrando la adición de nuevos productos se muestra en la Figura 10.

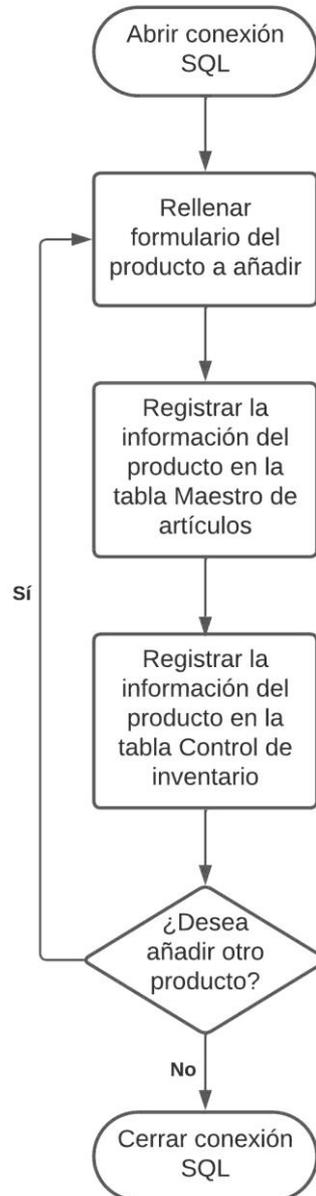


Figura 10. Flujo de adición de nuevos productos de la base de datos

Para introducir nuevos productos, se presentará al usuario un formulario con distintas ventanas solicitando la introducción de datos correspondientes a los atributos de la tabla. Estos datos generarán diferentes variables, que guardan el valor de los datos introducidos, para posteriormente usar esta información para rellenar el registro que se añade en SQL. Poder introducir código simplemente rellenando un formulario facilita su uso, si el usuario

no está familiarizado con SQL. Se iniciará una conexión a la base de datos y ejecutarán dos instrucciones SQL, una para añadir un registro en la tabla de Maestro de artículos y otro en la tabla Control de inventario. Los datos de referencia y el producto serán comunes en las dos tablas. Para el Control de inventario, las unidades y el estado se introducirán como NULL, solamente recibirán un valor cuando se haya recibido la primera compra. Esto es para evitar asignar un valor de 0 a las unidades, que el sistema detecte que el inventario está agotado y, por tanto, por debajo del stock de seguridad, genere una alerta innecesaria de que es necesario reponer el stock. Es necesario añadir el registro correspondiente al producto antes de registrar la primera compra, de lo contrario el inventario no se actualizará correctamente porque no se encontrará la referencia dentro de la tabla Control de inventario, cuando se ejecute el flujo correspondiente al recibir el aviso de la compra. Una vez introducidos los datos, se concluye el flujo cerrando la conexión a la base de datos SQL. En el caso de que la referencia que se está intentando introducir ya estuviera presente en las tablas, no se incluiría el registro.

El formulario que se va a utilizar es el mostrado a continuación en la Figura 11, mostrando la composición del mismo, que será la forma en que aparecerá delante del usuario cada vez que se ejecute para que se rellene. Consiste en la referencia, que al igual que en la tabla SQL se ha definido como un valor numérico, el nombre del producto, que será una entrada de texto, el precio, el stock de seguridad, el distribuidor, el fabricante y la referencia del fabricante, que se ha establecido como una entrada de texto, en lugar de numérica, debido a que es común que las referencias asignadas por las empresas cuenten con una combinación de números, letras y símbolos. Si se deja vacío alguno de los campos que se ha marcado obligatorio para rellenar y se intenta validar el formulario, dará un error y no avanzará el flujo, indicando qué campo queda por rellenar.



Referencia *

Nombre del producto *

Precio (€)

Stock de seguridad *

Distribuidor

Fabricante

Referencia del fabricante

Ok

Figura 11. Formulario de adición de nuevos productos

Todas las entradas numéricas incluidas en el formulario, referencia, precio y stock de seguridad se han definido con valor mínimo de 0, por lo cual no se podrán introducir números negativos, necesariamente han de ser un valor numérico superior o igual a 0. Además, se ha impuesto la condición de validación a la referencia, el nombre del producto y el stock de seguridad, esto implica que estos campos no se pueden dejar vacíos. En caso de no rellenarse, se generaría un aviso de error, avisando de que ese campo no se puede dejar vacío.

6.5 MODIFICACIÓN DE PRODUCTOS EN EL MAESTRO DE ARTÍCULOS

Si se desea modificar los atributos de un producto incluido en el Maestro de artículos, se dispone de un flujo que, al ejecutarse, permitirá al usuario ir añadiendo la información que se desea modificar. La información solicitada se irá presentando en distintos cuadros de mensaje que irán apareciendo para guiar al usuario sobre la información necesaria. Este flujo puede ser de utilidad, por ejemplo, si cambia el fabricante o el distribuidor de un producto y

hay que reflejar esa modificación en la información recogida en la base de datos. El desarrollo del flujo se muestra en la Figura 12.

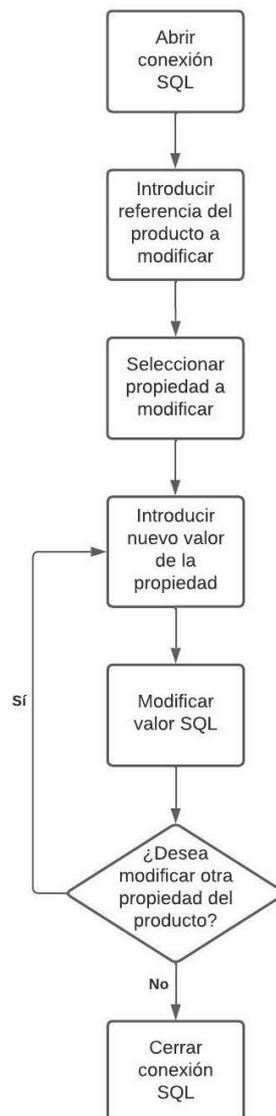


Figura 12. Flujo de modificación de atributos de un producto

El flujo comienza abriendo la conexión a la base de datos. Seguido de los distintos mensajes mostrados para recoger la información sobre el producto y el atributo que se desea modificar. En primer lugar, se solicita al usuario que introduzca la referencia del producto que se desea modificar, cuyo valor se almacena en una variable Referencia, que será utilizada

posteriormente en el comando SQL. A continuación, se solicita al usuario que elija el atributo que desea modificar, presentando las opciones en una lista desplegable. Esa elección se guarda en una variable `AtributoSeleccionado`. Por último, se solicita al usuario que introduzca el nuevo valor que desea asignar a la variable seleccionada, que se almacena en la variable `ValorIntroducido`. Con la información recogida en las variables mencionadas, se ejecuta un comando que actualiza el valor del atributo seleccionado con el valor introducido. Antes de concluir el flujo, se muestra un cuadro de mensaje preguntando al usuario si desea modificar otro atributo de ese producto. En caso de que se seleccione que sí que se desea modificar otra propiedad del producto, el flujo vuelve al paso de mostrar el cuadro de mensaje con el desplegable de los atributos. En caso de que la respuesta sea negativa, porque ya se han concluido las modificaciones que se desea realizar para ese producto, el flujo finaliza, cerrando la conexión a la base de datos.

6.6 ELIMINACIÓN DE REFERENCIA DE LA BASE DE DATOS

Puede suceder que se desee eliminar una referencia del sistema porque se trata de un producto obsoleto que ya no se comercializa. Eliminar la referencia puede tener sentido por varios motivos. En primer lugar, para mantener actualizada la base de datos, aportando claridad cuando se visualice y no teniendo datos almacenados que ya no poseen ningún valor para el usuario. En segundo lugar, puede querer liberarse la referencia para poder utilizarla en el futuro con otro producto, y no se podría introducir el nuevo producto a menos que primero se elimine el producto antiguo de la base de datos. Por último, permite evitar fallos que falseen el inventario. Si por algún motivo se añadieran al inventario productos de una referencia que no existe, el sistema daría error y no añadiría los productos. El flujo seguido para eliminar los productos se muestra en la Figura 13.

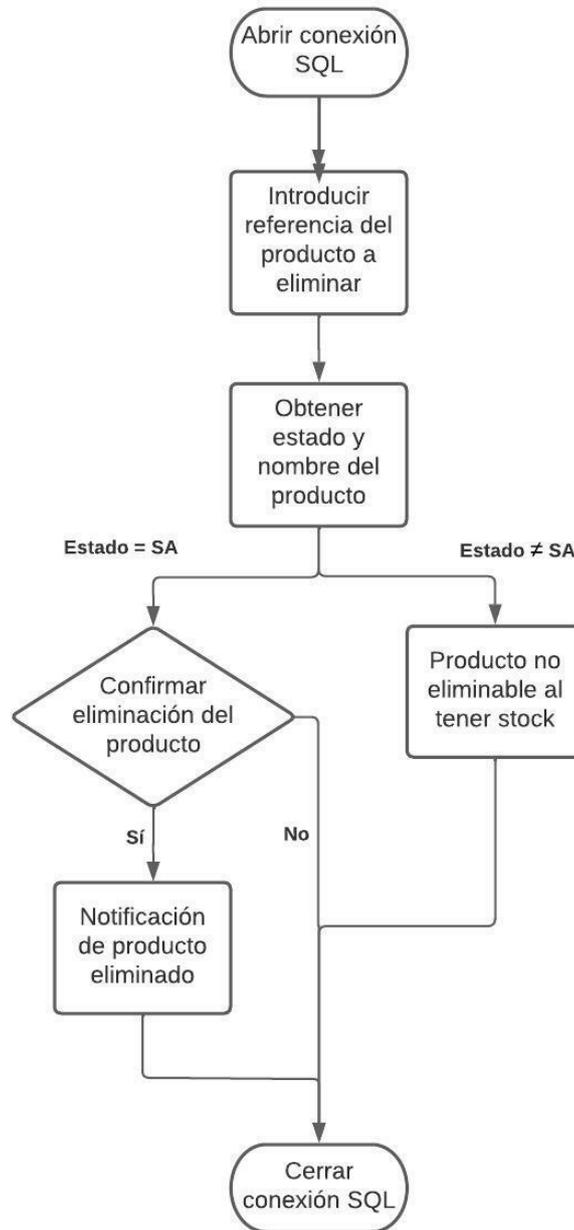


Figura 13. Flujo de eliminación de productos de la base de datos

El flujo comienza abriendo la conexión a la base de datos y pidiendo al usuario que introduzca la referencia del producto a eliminar. A continuación, se comprueba el estado del producto en la tabla de Control de inventario. Es necesario que el stock del producto esté a 0, y su estado sea de Stock Agotado, para asegurar que ya no quedan unidades en el almacén

y no se va a seguir operando con el producto. De lo contrario, el producto no se eliminará y habrá que esperar hasta que se hayan vendido todas las unidades, y se concluirá el flujo cerrando la conexión a la base de datos. Después de confirmar que el producto es el indicado, se procede a eliminarlo de la base de datos y a notificar al usuario. Para concluir, se cierra la conexión a la base de datos y termina el flujo.

Se ha considerado fundamental que el flujo diseñado incluya una confirmación de que se desea realmente eliminar la referencia que se ha introducido, para evitar así cualquier error humano en la introducción de datos. Hay que estar seguro de que se está eliminando el producto correcto. Las condiciones necesarias para eliminar un producto serán que se confirme el producto en un cuadro de mensaje y que, además, el stock del producto esté agotado, por lo que se puede eliminar porque no quedan unidades restantes en el almacén. Si no se cumplieran alguna de las dos condiciones, no se llevaría a cabo la eliminación del producto.

6.7 CÁLCULO DEL STOCK DE SEGURIDAD

Como se ha comentado al principio del capítulo, se ha diseñado el flujo del cálculo del stock de seguridad, pero no se implementa, debido a que inicialmente se considera que puede ser perjudicial para el funcionamiento del sistema. Al haber pocos pedidos, cualquier variación tendría gran impacto sobre el cálculo y generaría posibles interrupciones en la gestión de los productos almacenados. Interrupciones que podrían suponer en riesgo en la gestión de las compras del almacén, realizando compras para reponer el stock antes de lo que realmente sería necesario. Por lo cual, se ha decidido posponer la implantación del cálculo automático del stock de seguridad hasta que la empresa considere que dispone de suficiente cantidad de pedidos para que se pueda calcular de manera veraz. Se plantea cómo ha de calcularse en la Figura 14, pero no se implementará de común acuerdo con los responsables de la empresa. Se muestra el flujo a seguir, que se debería ejecutar periódicamente. No se cree necesario que se ejecute cada vez que existe un nuevo pedido, debido a que las variaciones deberían ser mínimas una vez alcanzado un nivel razonable de fiabilidad por parte de los datos

empleados en el cálculo. Para el stock de seguridad y la reposición del stock hay que tener en cuenta otro factor, que es la estacionalidad de la demanda que tiene la empresa.

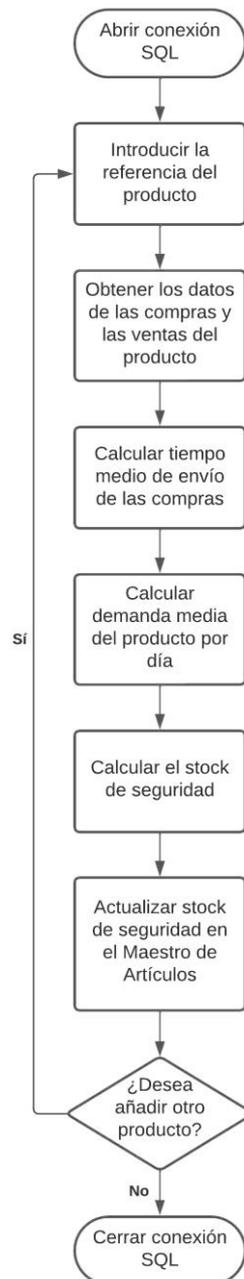


Figura 14. Flujo del cálculo del stock de seguridad

El flujo comienza estableciendo la conexión con la base de datos, para poder acceder a los datos que se van a manipular, y pidiendo al usuario que introduzca la referencia del producto para el que se desea recalcular su stock de seguridad. Será necesario obtener la referencia del producto que se desea actualizar para poder obtener los datos del producto. De esta manera se pueden realizar sucesivas consultas a la base de datos que permitan obtener los datos de los envíos de las compras del producto, para conocer los tiempos de entrega, y las compras, para conocer la demanda diaria media. El tiempo de entrega medio se obtendrá restando la fecha de recepción de la fecha de pedido y calculando el valor medio para los envíos del último año. La demanda media se obtiene con el número de unidades vendidas dividido por el número de días. A continuación, se calcula el stock de seguridad con los valores obtenidos y se ejecuta el comando para actualizar el valor del stock de seguridad del producto en la base de datos. Antes de concluir el flujo, se muestra un mensaje al usuario preguntando si desea actualizar el stock de seguridad de otro producto. Al haber incluido un bucle en el flujo, si el usuario indica que sí, se vuelve al principio del flujo, solicitando al usuario que introduzca la referencia del producto que se desea actualizar. En caso contrario, se cierra la conexión a la base de datos.

Capítulo 7. ANÁLISIS DE RESULTADOS

Tras concluir el diseño e implementación de la herramienta completa, incluyendo la creación de la base de datos y los flujos de automatización, se ha comprobado que todo funciona según lo establecido. Se han probado todos los flujos, ejecutándolos por separado y revisando que cumplen con lo esperado y que la base de datos cumple con lo establecido. Además, ha habido una reunión final con los responsables de la empresa para mostrarles el resultado final, explicarles el funcionamiento y dejar todo el sistema preparado para su futuro uso.

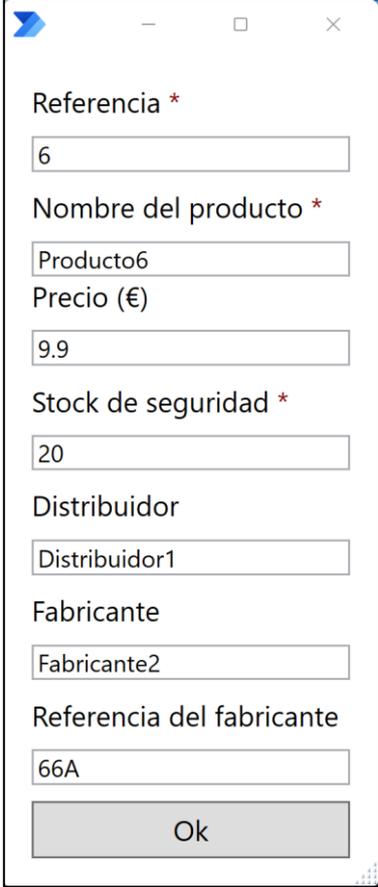
Con el objetivo de mostrar el correcto funcionamiento de este proyecto, se va a mostrar un ejemplo que muestre que los flujos diseñados, y la base de datos, funcionan tal y como se presupone. Para ello, inicialmente, se han dado de alta 5 productos en la base de datos, asociados a 3 fabricantes ficticios (Fabricante1, Fabricante2 y Fabricante3). Posteriormente, vamos a ver de forma detallada cómo se incluye un nuevo producto en la base de datos, cómo se efectúa un nuevo pedido de este y cómo se exportan los datos de la base de datos a una hoja de Excel.

El producto que se ha decidido incluir tendrá las siguientes características:

- Referencia: 6
- Nombre del producto: Producto6
- Precio: 9,90€
- Stock de seguridad: 20
- Distribuidor: Distribuidor1
- Fabricante: Fabricante2
- Referencia del fabricante: 66A

Como se puede ver en el formulario mostrado a continuación en la Figura 15, se rellena el formulario con los datos del producto, como parte del flujo de adición de nuevos productos

a la base de datos. Una vez añadido, el artículo queda registrado en la tabla y ya se puede empezar a operar con ese producto.



Referencia *

Nombre del producto *

Precio (€)

Stock de seguridad *

Distribuidor

Fabricante

Referencia del fabricante

Ok

Figura 15. Formulario de adición de productos rellenado

Asimismo, se ha simulado el registro de un pedido del producto, que incluye el documento del pedido y de la factura cuando lleguen los productos al almacén y haya que actualizar la base de datos, dando de alta las unidades. Para ello, en primer lugar, se envía un primer correo con el presupuesto del pedido, como se haría en una situación normal, y se ejecuta el flujo de gestión de pedidos. En segundo lugar, se envía un segundo correo con la factura del pedido y se vuelve a ejecutar el flujo de gestión de pedidos. De esta manera se consigue registrar por completo el pedido y actualizar el estado del producto. En la Figura 16, mostrada a continuación, se pueden ver los documentos del pedido simulado.

Presupuesto de Pedido			
Tipo	Pedido	Fecha	
Compra	2	24/07/2022	
Referencia	Producto	Unidades	Precio Unitario (€)
6	Producto6	200	9,90
IMPORTE TOTAL SIN IVA (€)			1980,00 €

Factura de Pedido			
Tipo	Pedido	Fecha	
Compra	2	24/07/2022	
Referencia	Producto	Unidades	Precio Unitario (€)
6	Producto6	200	9,90
IMPORTE TOTAL SIN IVA (€)			1980,00 €

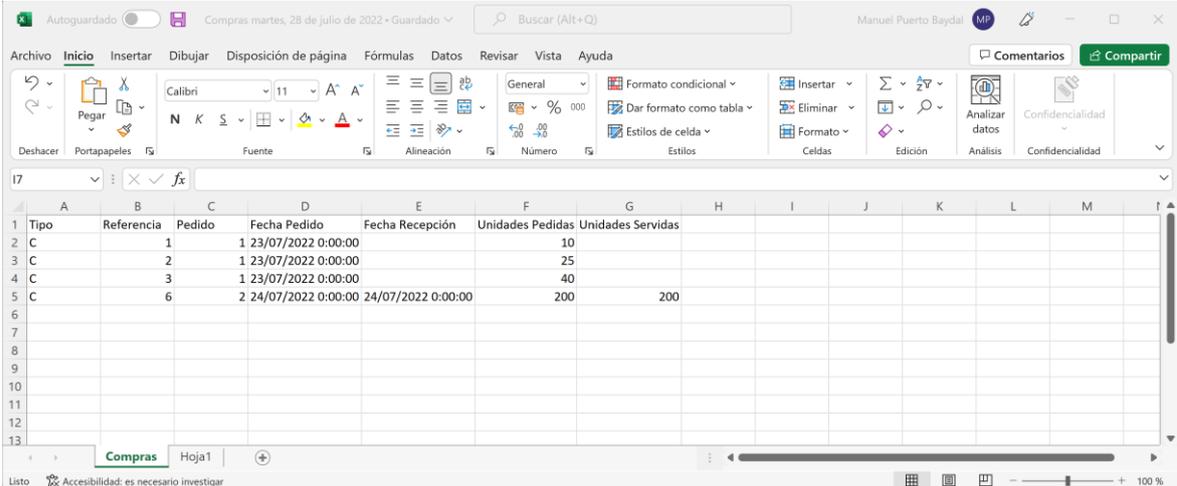
Figura 16. Documentos utilizados para simular el pedido

Los documentos siempre deben tener el formato mostrado en la Figura 16. De lo contrario, es posible que *Power Automate Desktop* no detecte la información de la manera correcta, y no se pueda registrar la información en la base de datos. Cualquier modificación del formato del documento debe corresponderse con una revisión del flujo, para asegurar que los datos contenidos en el documento se siguen recogiendo de la manera correcta.

Por último, una vez se ha ejecutado el flujo de gestión de pedidos dos veces para añadir el pedido completo a la base de datos, se sacan los datos contenidos en las tablas a una hoja Excel utilizando el flujo de gestión de consultas, introducido en el apartado 6.3. Esto se hace para poder comprobar que todo se ha registrado correctamente y funciona como se presupone.

En la Figura 17 podemos ver las capturas de las hojas de Excel que muestran los resultados. Como se observa, además del pedido del producto6, se ha realizado otro pedido (pedido

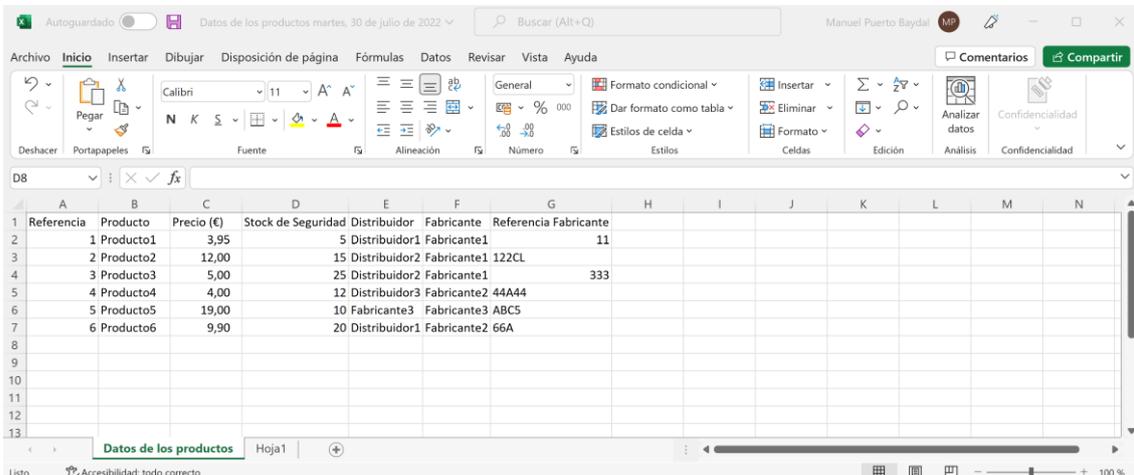
número 1) para comprobar que el funcionamiento del bucle incluido en el subflujo de registro de pedidos cumple con lo establecido cuando un pedido incluye más de un producto.



Tipo	Referencia	Pedido	Fecha Pedido	Fecha Recepción	Unidades Pedidas	Unidades Servidas
C	1	1	23/07/2022 0:00:00		10	
C	2	2	23/07/2022 0:00:00		25	
C	3	3	23/07/2022 0:00:00		40	
C	6	6	24/07/2022 0:00:00	24/07/2022 0:00:00	200	200

Figura 17. Datos de Compras extraídos de la base de datos a una hoja Excel

Además de la hoja Excel de los datos de las compras, se ha obtenido una segunda hoja mostrando los datos contenidos en el Maestro de Artículos, para comprobar que los datos del producto se han añadido correctamente. Si se observa la Figura 18, además de los datos del producto6, se pueden ver los otros productos añadidos, probando diferentes tipos de referencias de fabricante y precios, verificando así que el flujo añade los artículos según lo diseñado.



Referencia	Producto	Precio (€)	Stock de Seguridad	Distribuidor	Fabricante	Referencia Fabricante
1	Producto1	3,95		5 Distribuidor1	Fabricante1	11
2	Producto2	12,00		15 Distribuidor2	Fabricante1 122CL	
3	Producto3	5,00		25 Distribuidor2	Fabricante1	333
4	Producto4	4,00		12 Distribuidor3	Fabricante2 44A44	
5	Producto5	19,00		10 Fabricante3	Fabricante3 ABC5	
6	Producto6	9,90		20 Distribuidor1	Fabricante2 66A	

Figura 18. Datos de los artículos contenidos en la tabla Maestro de Artículos, extraídos a una hoja Excel

Mediante este ejemplo se ha mostrado un caso práctico del uso de los distintos flujos diseñados. Además, se ha podido comprobar que la herramienta funciona tal como se había planeado. Los flujos de ejecución se han ejecutado manualmente y se ha revisado que los cuadros de mensaje se muestran en el momento apropiado, se captura la información correctamente, se realizan las conexiones adecuadas con los demás programas de Microsoft y se plasma la información como es debido. A causa de que los flujos no tienen una periodicidad fija para ser ejecutados, depende más de en qué momento le sea conveniente al usuario, se han mantenido como flujos manuales, explicando a los encargados de la empresa su funcionamiento para que estén familiarizados con la función que cumple cada uno de los flujos.

Con la herramienta ya diseñada y montada, ha habido una reunión con la empresa para mostrar el resultado final. En la reunión, se han comentado con los responsables las funcionalidades y se ha realizado una demostración práctica de los flujos para que se familiaricen con su funcionamiento, y así poder recibir sus opiniones sobre el resultado final del proyecto. Las explicaciones han incluido un desglose de los distintos pasos de *Power Automate Desktop* y sus capacidades. Puesto que no habían utilizado previamente el programa, aunque sí estaban familiarizados con la tecnología RPA en general, era necesario explicarles en qué consistía el programa exactamente y cuáles eran sus principales funciones.

La explicación sobre las funcionalidades y los pasos aplicados en los distintos flujos resultaba fundamental para entender lo que está pasando en cada momento y las posibles modificaciones que se podrían realizar en el futuro. Al tener un mayor conocimiento de *Power Automate* se pueden idear cambios que mejoren la herramienta, si se detecta alguna funcionalidad de la herramienta que puede ser enfocada de otra manera, que aporte mayor valor.

Una vez la herramienta haya sido implantada en la empresa, el sistema quedará listo y disponible para empezar a funcionar cuando la empresa considere que es el momento oportuno de empezar a distribuir sus propios productos y gestionar su propio almacén. Con

el uso de la herramienta se podrá gestionar la entrada y salida de artículos, como se ha visto en este apartado.

Capítulo 8. CONCLUSIONES Y TRABAJOS FUTUROS

En este capítulo se recogen las principales conclusiones sobre el aprendizaje que se ha obtenido tras concluir el trabajo, los objetivos alcanzados y los conocimientos puestos en práctica. Así como una serie de ideas, que se podrían llevar a cabo en el futuro si se deseara seguir ampliando el sistema.

El desarrollo del proyecto ha conseguido cumplir los principales objetivos que se habían planteado, logrando diseñar una herramienta que permite controlar el inventario según lo establecido, registrando de manera automática los pedidos cuando llega el aviso al correo, así como almacenar una serie de datos que permitan analizar los KPI del almacén con facilidad. Es posible medir los plazos de entrega de forma sencilla, con las fechas registradas en el almacén, conocer la rotación de inventario al saber la cantidad de productos pedidos y con qué frecuencia es necesario reponer el inventario, así como conocer los productos más demandados en función de las ventas realizadas.

Otro factor importante, es la estrecha colaboración que ha habido entre empresa y alumno, puesto que en todo momento se ha mantenido a la empresa informada de los avances. Además, se han tenido en cuenta las opiniones y peticiones de los responsables a la hora de desarrollar la herramienta, pudiendo profundizar en las necesidades de la empresa y su plan estratégico con respecto a la distribución de productos.

A nivel personal siento que la realización de este trabajo ha sido una oportunidad excelente para desarrollar mi curiosidad sobre el uso de una tecnología en auge, como es RPA, y ha servido para reforzar mis conocimientos previos de SQL. Después de concluir el trabajo, estoy convencido de que todo lo puesto en práctica en este proyecto, los conocimientos adquiridos a lo largo del máster y los específicos que ha sido necesario aprender para realizar el trabajo, serán de gran utilidad en el futuro.

Por último, es conveniente mencionar los siguientes pasos que se podrían dar con el objetivo de seguir mejorando la herramienta. Las principales tareas podrían ser la incorporación de flujos DPA de *Power Automate* en la nube, con el objetivo de que los flujos trabajasen de manera completamente desatendida, y la implementación del cálculo automático del stock de seguridad. El desarrollo de flujos DPA permitiría conectar los flujos desarrollados en *Power Automate Desktop* a procesos en la nube, que funcionen incluso cuando el usuario no está presente. Con respecto al cálculo del stock de seguridad, sería necesario implementarlo y afinar el método de cálculo que se ha utilizado en la propuesta realizada en este proyecto, pudiendo ser necesario establecer un plazo diferente a un año para los datos utilizados, que permita tener en cuenta la variación estacional de la demanda. Además de los datos del propio almacén, la empresa podría utilizar los datos generales del *marketplace* para hacer previsiones de demanda de los productos del almacén. Para implementar esas dos funcionalidades a la herramienta sería conveniente esperar a que esté en uso por parte de la compañía y que el volumen de negocio justifique su implantación. Un último aspecto, que podría ser conveniente tratar para la empresa, sería la manera en la que se comunica la plataforma de la empresa con la herramienta de control de inventario, asegurando que en la plataforma está registrado el valor actualizado del inventario y es capaz de verificar si un pedido se puede realizar antes de enviar el aviso por correo electrónico a la base de datos.

Capítulo 9. BIBLIOGRAFÍA

- [1] A. Molins de la Fuente, «Logística Internacional,» Escuela de Organización Industrial, Madrid, 2011.
- [2] F. A. Morteo y N. L. E. Bocalandro, Un enfoque práctico de SQL, Buenos Aires: Ediciones Cooperativas, 2004.
- [3] Oracle, «¿Qué es una base de datos relacional?,» 2022. [En línea]. Available: <https://www.oracle.com/es/database/what-is-a-relational-database/>.
- [4] TechLib, «Definición de RDBMS (Sistema de gestión de bases de datos relacionales),» 2022. [En línea]. Available: <https://techlib.net/definicion/rdbms.html>.
- [5] A. H. D. N. J. R.-J. A. S. D. R. B. W. Kellyn Gorman, Introducing Microsoft SQL Server 2019: Reliability, scalability, and security both on premises and in the cloud, Birmingham: Publishing Ltd., 2019.
- [6] Microsoft, «Servicios de informática en la nube | Microsoft Azure,» 2022. [En línea]. Available: <https://azure.microsoft.com/es-es/>.
- [7] Microsoft, «Azure SQL Database: servicio de base de datos en la nube administrada | Microsoft Azure,» 2022. [En línea]. Available: <https://azure.microsoft.com/es-es/products/azure-sql/database/>.
- [8] Microsoft, «SLA para Azure SQL Database | Microsoft Azure,» Abril 2022. [En línea]. Available: https://azure.microsoft.com/es-es/support/legal/sla/azure-sql-database/v1_8/.

- [9] Deloitte, «La era de la Automatización. Implementación de Robotics en los Centros de Servicios Compartidos,» Deloitte Consulting Group S.C., México, 2016.
- [10] C. A. Bermúdez Irreño, «RPA - AUTOMATIZACIÓN ROBÓTICA DE PROCESOS: UNA REVISIÓN DE LA LITERATURA,» *Revista Ingeniería, Matemáticas y Ciencias de la Información*, vol. 8, nº 15, pp. 111-122, 2021.
- [11] Gartner Inc., «Gartner Says Worldwide RPA Software Spending to Reach \$2.9 Billion in 2022,» p. 1, 1 Agosto 2022.
- [12] Microsoft, «Automatización de procesos robótico (RPA) | Microsoft Power Automate,» 2022. [En línea]. Available: <https://powerautomate.microsoft.com/es-es/robotic-process-automation/>.
- [13] G2, «Top 10 Microsoft Power Automate Alternatives 2022 | G2,» 2022. [En línea]. Available: <https://www.g2.com/products/microsoft-power-automate/competitors/alternatives>.
- [14] Microsoft, «Automatización digital de procesos | Microsoft Power Automate,» 2022. [En línea]. Available: <https://powerautomate.microsoft.com/es-es/digital-process-automation/>.
- [15] RAE, *REAL ACADEMIA ESPAÑOLA: Diccionario de la lengua española, 23.ª ed.*, Madrid, 2022.
- [16] RAE, *REAL ACADEMIA ESPAÑOLA: Diccionario de la lengua española, 23.ª ed.*, Madrid, 2022.
- [17] A. P. Silverio, 2020. [En línea]. Available: <https://asesorias.com/empresas/programas-gratis/software-gestion-almacenes/>.

- [18] Hilti, «ON!Track: Toma el control de tus recursos,» 2022. [En línea]. Available: <https://www.hilti.es/content/hilti/E2/ES/es/servicios/herramientas/gestion-herramientas-inventario-on-track.html>.
- [19] SAP, «SAP Extended Warehouse Management | WMS | SAP,» 2022. [En línea]. Available: <https://www.sap.com/spain/products/scm/extended-warehouse-management.html>.
- [20] On Device Solutions Ltd. , «myWarehouse | Enterprise Asset Management,» 2022. [En línea]. Available: https://store.sap.com/dcp/en/product/display-0000056225_live_v1/myWarehouse?gclid=CjwKCAjwo_KXBhAaEiwA2RZ8hL_-A_9V5y6SLPH7v3thsE2l43RIukCoSEAWgDuEznrVf77cAWcmWhoCeRMQAvD_BwE.
- [21] Holded, «Software de gestión de inventario | Holded,» 2022. [En línea]. Available: <https://www.holded.com/es/gestion-de-inventarios>.
- [22] Microsoft, «Calculadora de precios | Microsoft Azure,» 2022. [En línea]. Available: <https://azure.microsoft.com/es-es/pricing/calculator/>.
- [23] Microsoft, «Precios | Microsoft Power Automate,» 2022. [En línea]. Available: <https://powerautomate.microsoft.com/es-es/pricing/>.
- [24] J. Heizer y B. Render, Dirección de la producción y de operaciones. Decisiones estratégicas, Pearson, Prentice Hall, 2007.
- [25] Microsoft, «Creación de claves principales en SQL Server - SQL Server | Microsoft Docs,» 2022. [En línea]. Available: <https://docs.microsoft.com/es-es/sql/relational-databases/tables/create-primary-keys?view=sql-server-ver16>.

ANEXO

Alineación con los Objetivos de Desarrollo Sostenible (ODS)

Este proyecto se considera alineado con una variedad de los Objetivos de Desarrollo Sostenible (ODS). Principalmente, se alinea con el objetivo 9, Industria, Innovación e Infraestructura, debido al marcado carácter tecnológico que posee el desarrollo del proyecto, y con el objetivo 8, Trabajo Decente y Crecimiento Económico, por su contribución al desarrollo de un nuevo negocio dentro de la empresa que permitirá expandir sus actividades y diversificar los perfiles contratados. Aunque en menor medida, el objetivo 12, Producción y Consumo Responsable, también se puede considerar que está relacionado con este proyecto debido a que los sistemas de control de inventario promueven una gestión más eficiente del stock almacenado, siendo consciente de las existencias exactas que se poseen.

Con respecto a la vinculación del proyecto con el objetivo 9, la implementación del sistema de control de inventario supondrá el desarrollo de una herramienta de software que permita gestionar el futuro inventario de la compañía de la manera más eficiente posible. Con esta herramienta, se mejorará la gestión del almacén, haciendo que los empleados puedan manejar el inventario con mayor eficacia, y precisión, y prevenir el desperdicio del material almacenado por fallos en la previsión. Cualquier herramienta que pueda facilitar una mejora tecnológica dentro del sector logístico es capaz de favorecer la innovación y el desarrollo dentro de dicho sector.

Además, se puede considerar que este proyecto también está ligado al objetivo 8 porque, al desarrollar el trabajo, se habilitará a la empresa a desarrollar una de sus futuras líneas estratégicas. Esto les permitirá diversificar y aumentar su negocio, creando mayores ingresos para la empresa y necesidades de personal para la gestión de su almacén, lo cual generará más puestos de trabajo y más contrataciones para solucionar esas necesidades de mano de obra. Para poder gestionar el almacén correctamente, se necesitarán una variedad de perfiles, tanto cualificados como no, que diversificarán las características de los empleados

contratados por la empresa. Se diferenciará de una mayoría de perfiles tecnológicos actualmente presentes en la empresa, dando oportunidad de acceso al mercado de trabajo a un mayor número de personas. La implantación de un sistema de control de inventario supone que la tarea puede ser elaborada con precisión sin necesidad de ser realizada manualmente. En caso de no tener en uso este tipo de software, sería necesario que una persona llevara el control de forma manual, teniendo que hacer un inventario periódicamente, siendo una tarea que se dificulta en exceso cuando se manejan grandes cantidades de stock. Por lo tanto, también es de suponer que, la adopción de medidas tecnológicas en el apartado logístico pueda venir aparejada con una mejora de la productividad y la eficiencia en la gestión del almacén.

Por último, la vinculación del proyecto con el objetivo 12 viene de la mejora en eficiencia en la logística de una empresa derivada de la implantación de un sistema de control de inventario. Implementar este tipo de sistema permite un mayor control sobre las existencias que se poseen, esto evita la sobrecompra de productos y el posible deterioro de productos que quedan olvidados en el almacén por errores al contabilizar el inventario disponible.