



**Universidad de Valladolid**



**Trabajo de fin de máster**

# **ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN**

Autor:

Trigueros Rey, José Miguel

Tutor:

Mateo, Manuel

Valladolid, julio 2021

## **RESUMEN**

En la fábrica de asientos Isringhausen, situada en IVECO, se observa una serie de dificultades a la hora de identificar y registrar las piezas defectuosas de mano de proveedor debido a la pérdida de información producida por la frecuente comunicación interdepartamental para hacerlo correctamente.

El objetivo principal de este trabajo consistirá en evitar los despilfarros producidos por un frecuente retrabajo a la hora de identificar, registrar y llevar a cabo el envío de piezas defectuosas por parte de los proveedores. Mediante esta implantación se pretende lograr que todos los operarios de la línea de montaje de asientos de Isringhausen identifiquen tanto la pieza que es defectuosa como su referencia, el propio defecto y su localización mediante la codificación de los distintos tipos de defectos posibles para cada pieza, llevando a cabo una estandarización de los mismos y logrando que cada codificación sea unívoca y así evitar que se puedan cometer errores.

A su vez, también se facilitará el trabajo llevado a cabo posteriormente, tanto por parte de calidad, a la hora de documentar el registro de defectos y elaborar el correspondiente informe para cada proveedor, como por parte de logística, ya que sabrá lo que ha de cargar en cada camión sin tener la necesidad de preguntar a fábrica por las piezas defectuosas y a calidad por la cuestión o reclamación a la que se asignan en cada caso.

Por tanto, mediante el uso de las herramientas de Excel y Visual Basic, se procederá a llevar a cabo una automatización y estandarización de todos los procesos que se llevan a cabo durante el proceso de gestión de devoluciones, procurando reducir al mínimo la pérdida de información y tiempo que podría producirse al tener que pasar a través de tres departamentos diferentes, reduciendo los desplazamientos de material innecesarios, así como liberar a los encargados dándole al programa la máxima flexibilidad y haciéndolo lo más intuitivo posible para que cualquier operario de línea pueda ejecutar dicho proceso.

## **ABSTRACT**

At IVECO's Isringhausen seat factory, there are a number of difficulties in identifying and recording defective supplier hand parts due to the loss of information caused by frequent interdepartmental communication in order to do this correctly.

The main objective of this work will be to avoid wastage caused by frequent rework when identifying, recording and carrying out the shipment of defective parts from suppliers. The aim of this implementation is to ensure that all the operators of the Isringhausen seat assembly line can identify both the defective part and its reference, the defect itself and its location by coding the different types of defects possible for each part, standardizing them and ensuring that each coding is univocal, thus avoiding errors.

In turn, it will also facilitate the work carried out later, both by quality, when documenting the record of defects and preparing the corresponding report for each supplier, and by logistics, since they will know what to load on each truck without having to ask the factory for the defective parts and quality for the issue or claim to which they are assigned in each case.

Therefore, through the use of Excel and Visual Basic tools, we will proceed to carry out an automation and standardization of all the processes that are carried out during the returns management process, trying to minimize the loss of information and time that could occur by having to go through three different departments, reducing unnecessary material movements, as well as freeing the managers by giving the program maximum flexibility and making it as intuitive as possible so that any line operator can execute this process.

## **AGRADECIMIENTOS**

En primer lugar, me gustaría agradecer al personal de la fábrica de asientos Isringhausen, que me propuso esta opción para poder llevar a cabo mi trabajo de fin de máster y me ayudó a conocer los entresijos de la logística, así como las particulares de la gestión de las devoluciones, y más particularmente a mi tutor, Luis González Sánchez, que me pudo guiar en todo momento para poder llegar al fondo de la programación de Visual Basic y a poder solventar todos los problemas que fueron surgiendo durante su ejecución. También me gustaría agradecer el apoyo de mi tutor del trabajo, Manuel Mateo, que pese a estar bastante ocupado pudo sacar tiempo para ayudarme a emprender esta causa. Por último, me gustaría agradecer a todos los profesores del máster por los conocimientos que me han ayudado a adquirir durante este extraño año de pandemia, logrando afianzar información en todos los campos relacionados con la logística. Si lo solucioné bien, atribúyase el mérito a mi equipo de apoyo. Si lo solucioné mal, acháquese a mis dificultades de comprensión. Sin más dilación, les dejo a continuación con el desarrollo del trabajo, que constará de una introducción, una parte teórica, un capítulo que hablará de la empresa en la que ha sido realizado, el desarrollo del propio programa y, por último, el estudio económico y las conclusiones y desarrollos futuros.

## Contenido

Capítulo 1: Introducción.....	1
1.1- Justificación.....	1
1.2- Objetivos.....	1
1.3- Método.....	3
1.4- Alcance.....	4
1.5- Estructura.....	4
Capítulo 2: Marco teórico.....	7
2.1- Logística inversa.....	7
2.2- Gestión de devoluciones.....	9
2.2.1 Gestión de reclamaciones.....	11
2.3- Programación Visual Basic y Excel.....	14
2.3.1-Historia.....	14
2.3.2-Características.....	17
2.3.3-Objetos y eventos.....	19
2.3.4-Ejemplos de código.....	22
2.3.5-Ventajas y desventajas de usar VBA.....	25
2.3.6-Alternativas a Visual Basic.....	26
Capítulo 3: Isringhausen.....	29
3.1- Historia de la empresa.....	29
3.2- Conceptos de asiento para la movilidad del mañana.....	34
3.3- Empresa y ubicación.....	36
Capítulo 4: Programa para gestión de devoluciones en Isringhausen.....	39
4.1- Manual de usuario.....	39
4.2- Análisis del Antes-Después y desarrollo del programa.....	41
4.3- Código de programación.....	52
Capítulo 5: Estudio económico.....	69
5.1- Factibilidad del programa para la gestión de devoluciones en Isringhausen.....	69
5.2- Desglose de costes de realización del proyecto de investigación.....	72
Capítulo 6: Conclusiones y desarrollos futuros.....	75
Bibliografía.....	77

## Índice de ilustraciones

Figura 2.1 Servicio de gestión de reclamaciones en una cadena de suministro [Ji, 2008] .....	11
Figura 2.2 Proceso de atención a reclamaciones. [Ji, 2008] .....	12
Figura 2.3 Logo de VisualBasic.Net [Saavedra, 2008] .....	14
Figura 2.4 Evolución de VisualBasic [Akkandi, 2009] .....	17
Figura 2.5 Clasificación de elementos utilizados en VBa [Ortíz, s.f.] .....	20
Figura 3.1 Isringhausen [We are Isringhausen - Isringhausen. (s.f.)] .....	29
Figura 3.2 Fundación [We are Isringhausen - Isringhausen. (s.f.)] .....	29
Figura 3.3 Traslado a Lemgo [We are Isringhausen - Isringhausen. (s.f.)] .....	29
Figura 3.4 Asientos modulares [We are Isringhausen - Isringhausen. (s.f.)] .....	30
Figura 3.5 Asientos automáticos [We are Isringhausen - Isringhausen. (s.f.)] .....	30
Figura 3.6 Primer cinturón de tres puntos [We are Isringhausen - Isringhausen. (s.f.)] .....	30
Figura 3.7 Adquisición por parte de Aunde [We are Isringhausen - Isringhausen. (s.f.)] .....	31
Figura 3.8 Primer asiento ajustable eléctrico [We are Isringhausen - Isringhausen. (s.f.)] .....	31
Figura 3.9 Ampliación del segmento [We are Isringhausen - Isringhausen. (s.f.)] .....	31
Figura 3.10 Asiento NTS [We are Isringhausen - Isringhausen. (s.f.)] .....	32
Figura 3.11 Cinturón de tres puntos [We are Isringhausen - Isringhausen. (s.f.)] .....	32
Figura 3.12 Último asiento NTS [We are Isringhausen - Isringhausen. (s.f.)] .....	32
Figura 3.13 Asientos Esteban [We are Isringhausen - Isringhausen. (s.f.)] .....	32
Figura 3.14 Metales para asientos [We are Isringhausen - Isringhausen. (s.f.)] .....	33
Figura 3.15 Nueva planta de turismos en Lemgo [We are Isringhausen - Isringhausen. (s.f.)] .....	33
Figura 3.16 Centenario de Isringhausen [We are Isringhausen - Isringhausen. (s.f.)] .....	33
Figura 3.17 Isringhausen en el mundo [We are Isringhausen - Isringhausen. (s.f.)] .....	34
Figura 4.1 Hoja de defectos por proveedor inicial .....	42
Figura 4.2 Excel inicial de calidad al que se exportan los datos .....	43
Figura 4.3 Tipos de asientos .....	44
Figura 4.4 Fundas de conductor .....	45
Figura 4.5 Piezas Metálicas de Conductor Europeo .....	45
Figura 4.6 Hoja de defectos por proveedor .....	46
Figura 4.7 Seleccionable por tipo de defecto .....	47
Figura 4.8 Seleccionable por ubicación .....	47
Figura 4.9 Hoja de calidad a la que se exporta .....	48
Figura 4.10 Tabla para exportar desde inicio .....	50
Figura 4.11 Macros individuales por botón .....	50
Figura 4.12 Tabla de asignación de botones a las referencias .....	51
Figura 4.13 Cuadro de búsqueda sencillo .....	52
Figura 4.14 Macro de comprobación inicial .....	53
Figura 4.15 Tabla de tamaños de lote y huecos disponibles .....	55

Figura 4.16 Macro para exportar valores desde el inicio.....	56
Figura 4.17 Macro para seleccionar pieza, referencia y proveedor.....	58
Figura 4.18 Macro para desplazamiento entre pestañas.....	59
Figura 4.19 Macro inicial de búsqueda por botones.....	60
Figura 4.20 Macro individual para búsqueda de botones .....	60
Figura 4.21 Macro de búsqueda de proveedor y referencia.....	61
Figura 4.22 Macro para buscar los valores desde un botón .....	62
Figura 4.23 Macro para imprimir .....	63
Figura 4.24 Macro inicial para exportar datos a calidad .....	64
Figura 4.25 Macro para exportar los datos a calidad.....	65
Figura 4.26 Macro para eliminar los valores editables .....	67
Figura 4.27 Macro para volver a la pestaña “ACCESO DIRECTO” .....	68
Figura 4.28 Macro para la realización del proceso completo .....	68
Figura 5.1 Informe de costes por tareas .....	72





# CAPÍTULO 1: INTRODUCCIÓN

## 1.1- Justificación

La necesidad de llevar a cabo este trabajo surge al percatarse la empresa del frecuente retrabajo que se produce en distintos lugares de la fábrica debido a la pérdida de información al pasar ésta de unas personas a otras, produciendo retrabajos y movimientos innecesarios.

Estas pérdidas repercuten en todos los departamentos de la empresa y, mediante la ejecución de este proyecto, se pretenderá lograr la reducción de comunicaciones redundantes entre los mismos. Al identificar fabricación una pieza defectuosa, ha de identificarla y registrarla, para después hacerle llegar la información tanto al departamento de calidad, que abrirá una reclamación que variará en función del proveedor, como a logística, que se pondrá a la disposición de la fábrica para programar el envío de las piezas en el próximo camión.

A mayores, este trabajo me supone de un gran interés, debido a que me satisface en gran medida automatizar y estandarizar procesos repetitivos, además de aprender a desenvolverme en un nuevo lenguaje de programación muy versátil, como lo es Visual Basic.

## 1.2- Objetivos

Los objetivos iniciales de este trabajo serán automatizar y estandarizar el proceso de toma de datos de los productos defectuosos que se envían a los distintos proveedores, haciendo que cada dato únicamente se introduzca una vez y evitando así el posible error humano.

De esta forma se podrá liberar de algo de trabajo a la gente de fabricación, que podrá sacar más tiempo o emplear el mismo para hacer procesos de 5S o dejar su puesto más

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

organizado, a la gente de calidad, que ya no tendrá necesidad de volcar todos los datos registrados, pudiendo centrarse en documentarlo todo en SAP y ponerse en contacto con el proveedor, y a la gente de logística, que tendrá identificado el contenedor desde el momento en que se cierre para poder localizarlo rápidamente y evitar que se produzcan movimientos innecesarios de los contenedores que les permitan identificar cuál era el que realmente se desea enviar. Los proveedores también saldrán beneficiados de este proceso, ya que la identificación posterior de las piezas será mucho más fácil e intuitiva.

El objetivo final de este trabajo por tanto consistirá en situar un ordenador a pie de línea al que puedan tener acceso todos los operarios y empleados, para que en el momento en el que identificaran una pieza defectuosa, puedan acercarse al mismo para asignarle un número a la pieza, que será el que finalmente la identificará cuando vaya junto a las demás en un contenedor cerrado.

Esto además provocará que el encargado no tenga que volver a revisar todas las piezas discriminadas para volver a identificar los defectos que ya habían identificado los operarios, y eliminará la necesidad de que tuviera que desplazarse hasta el puesto de trabajo del operario que la identificó para preguntarle de qué defecto se trataba en caso de no verlo.

A mayores, obligando al encargado a imprimir la hoja identificativa una vez se cierra cada contenedor, se logrará que todos los contenedores destinados a las devoluciones vayan identificados desde el primer momento y puedan ser desplazados por parte de los carretilleros a la zona de bloqueo, evitando por un lado tanto que el contenedor pueda equivocarse y mezclarse con las piezas de entrada para repetir todo el proceso de registro, como que por otro se desplace hasta la zona de bloqueo únicamente en el momento en el que se le añade la hoja identificativa.

Además, mediante la codificación y estandarización de los defectos se logrará un gran avance al tener proveedores de países muy diferentes en los que se hablan idiomas

## CAPÍTULO 1: INTRODUCCIÓN

distintos, que no tienen por qué necesariamente saber español y ya no tendrán que revisar en su nave cada pieza una a una para volver a identificar los defectos.

Otros de los stakeholders beneficiados serán calidad y logística, como ya se mencionó anteriormente, así como los camioneros que llevan a cabo los envíos, que ya no tendrán la necesidad de esperar hasta que los contenedores de calidad sean preparados para su posterior envío.

### 1.3- Método

Para llevar a cabo este proceso, se piensa que la utilización de Visual Basic para automatizar todos los procesos de Excel que se repiten constantemente puede llegar a ser una gran solución. De esta forma, mediante la estandarización y codificación de defectos y la utilización de macros se logrará que los datos introducidos por la gente de fábrica no tengan que ser manipulados posteriormente, y sean los que queden registrados, tanto en los informes de calidad como en las hojas identificativas de los distintos contenedores.

Para esto, se deberá conseguir que el Excel en el que el departamento de fábrica introduce las piezas defectuosas logre comunicarse con el que utiliza el departamento de calidad para identificar las mismas y crear una reclamación que variará en función del proveedor al que haga referencia, y su nomenclatura constará de una "Q" + Número de proveedor + Número de reclamación para dicho proveedor, haciendo que se genere de manera automática y se vaya actualizando como debería en todo momento.

Para la introducción de datos, se generarán distintos menús que variarán en función de la formación que tenga el empleado que lo ejecute, pasando desde un menú en el que el operario mete el tipo de pieza que va introducir y una macro le desplaza hasta el proveedor correspondiente, por un menú en el que se introducirán las referencias de cualquier

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

proveedor indistintamente e identificando la cantidad de piezas y el defecto asignado, hasta un intuitivo sistema de botones en el que se hará avanzando en función de la discriminación hasta llegar a la pieza deseada.

Así, se logrará que, pulsando un simple botón, la hoja de cada proveedor se imprima en el formato adecuado predispuesta para ya ser colocada en su contenedor correspondiente, se exporten todos los datos introducidos al Excel que explota calidad, y por último se restablezca a cero, haciendo un borrado de datos de la propia hoja que a su vez provocará que la reclamación avance a la siguiente.

### 1.4- Alcance

Este proyecto abordará la gestión de devoluciones de materia prima defectuosa por parte del proveedor desde el momento en que se detecten las piezas defectuosas en los distintos puestos de fabricación, pasando por su posterior identificación en el contenedor correspondiente, registro en el fichero de calidad y finalmente la realización del albarán para que los camioneros puedan llevarlo a su lugar de destino.

### 1.5- Estructura

Si hablamos de la estructura del proyecto, deberemos decir que está compuesta por el capítulo 2, el marco teórico, que hablará de la logística inversa y la gestión de devoluciones, haciendo un énfasis en la gestión de reclamaciones, y por último hablará de la programación de Visual Basic y Excel, enfocándose en su historia, sus características y la forma de programar con los mismos, así como en las ventajas y desventajas que pueden tener utilizar estos programas y algunos softwares alternativos que se podrían haber utilizado.

## CAPÍTULO 1: INTRODUCCIÓN

A continuación, en el capítulo 3, se hablará de Isringhausen como empresa en el mundo, tratando la historia de la misma y especificando cómo se fue desarrollando a lo largo del tiempo, hablando de algunos de sus productos principales y los pilares en que se basa para garantizarse un futuro próspero y estable, y posteriormente se hablará de la planta en concreto para la que se ha llevado a cabo este proyecto, situada en la fábrica de IVECO Pegaso en Valladolid, y que realiza los asientos para las furgonetas Daily de la misma marca.

El capítulo 4 se centrará en analizar el “Antes-Después” de la ejecución del proyecto y analizará lo que se tenía anteriormente y cómo se pudo mejorar, pasando a explicar los entresijos de la programación que va por debajo del programa y mostrando algunas capturas de pantalla para aclarar el contenido de la misma,

En el capítulo 5 se llevará a cabo un estudio económico del ahorro que supuso para la empresa implantar este sistema en su fábrica, estudiando el coste de manipulación innecesaria de los contenedores, así como el tiempo ahorrado tanto por el personal de planta como el de oficina.

Por último, en el capítulo 6, se hablará de las conclusiones y los posibles desarrollos futuros que surgen a partir de la implantación de este sistema.



## CAPÍTULO 2: MARCO TEÓRICO

### 2.1- Logística inversa

Las empresas han comenzado a darse cuenta de que la implementación de operaciones de logística inversa tiene el potencial de generar una gran cantidad de ingresos o ganancias al tiempo que minimiza los costes, para seguir siendo sostenibles y competitivos en el mercado [Morgan et al., 2018] asociados a las estrategias de reutilización, recuperación, reciclaje y remanufactura [Khor et al., 2016].

La logística inversa es contraria a la logística tradicional. El flujo se mueve desde los clientes o distribuidores hasta los fabricantes (desde el flujo descendente hacia el flujo ascendente en la cadena de suministro) porque algunas producciones perdieron su valor de uso obvio como el embalaje, perdieron su función por daños, son difíciles de vender en el mercado por resultar excedentes, o porque deben ser devueltos por alguna razón ligada a la calidad. La gestión de la logística inversa es el proceso de planificación, implementación y control de las actividades de logística inversa (Ji, 2008).

De Brito y Dekker (2004) concretan el término como el proceso de planificación, implementación y control de los flujos de retorno de materias primas, el inventario en proceso, embalaje y productos terminados, desde un punto de fabricación, distribución o uso, hasta un punto de recuperación o de disposición adecuada.

El Consejo de Profesionales de Gestión de Cadena de Suministro tiene definida la logística inversa como un segmento especializado de la logística que se centra en el movimiento y la gestión de productos y recursos después de la venta y de la entrega al cliente. Incluye devoluciones de productos para reparación y/o crédito [CSCMP, 2013]. Proceso consistente en una cadena de actividades que comienza con la recolección de productos devueltos de los consumidores y finaliza cuando esos productos devueltos son reprocesados y están listos para ser vendidos nuevamente [Safdar, 2020].

Govindan y Bouzon [2018], identifican tres barreras para la gestión de la logística inversa desde la perspectiva de los clientes:

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

1. Las dificultades entre los eslabones de la cadena de suministro, apoyo y coordinación reducidos por parte de los clientes al no compartir información de costes y el poco apoyo de toda la cadena a la implementación de prácticas de logística inversa.
2. Al llevar este tema a un contexto internacional, es relevante mencionar que las leyes de algunos países asignan más responsabilidad por la logística inversa a los fabricantes, mientras que las leyes de otros países adoptan una responsabilidad más compartida entre todos los actores de la cadena (incluidos los clientes y distribuidores). En consecuencia, estas diferencias en las leyes dan como resultados conjuntos de desafíos y oportunidades muy diferentes. Desde el lado descendente de la cadena, existe una influencia importante para las actividades de logística inversa porque los clientes se convierten en proveedores para las operaciones de devolución. En este sentido, existe una barrera para implementar sistemas de recuperación de producto, ya que la previsión y planificación de retornos son limitadas.
3. Finalmente, la percepción de un producto de peor calidad puede ser una barrera desde el punto de vista de los clientes. Los clientes pueden pensar que los productos remanufacturados o el uso de material reciclado dan como resultado estándares de calidad más bajos.

Una de las causas de la generación de logística inversa está determinada por la forma en que una empresa asume las reclamaciones de los clientes y las incorpora a la gestión de sus servicios, eliminando los obstáculos del flujo de retorno del producto o la información; en caso contrario, si una empresa ignora la queja del cliente, el servicio de logística inversa se deja de lado. Al analizar la información de las quejas del cliente de manera profesional (los resultados pull de logística inversa), se pueden descubrir algunos errores en el diseño del producto o en los procesos de producción y las empresas deben tomar la iniciativa de retirar esos productos [Ji, 2008].



## CAPÍTULO 2: MARCO TEÓRICO

### 2.2- Gestión de devoluciones

Las devoluciones de productos se pueden definir como logística inversa porque implican no solo la logística de avance, sino que también consideran los flujos inversos de los productos. Es un concepto de ciclos materiales que reemplaza progresivamente la percepción de economía "unidireccional". Muchos estudios realizados en el pasado han demostrado que los fabricantes de automóviles pueden obtener ventajas y oportunidades a través de la implementación de la devolución de productos. La industria automotriz en rápida expansión ha exigido que los actores de la industria presten más atención a los problemas ecológicos, como la gestión de devolución de productos [Zailani, 2017].

La devolución al proveedor consiste en retornar material que ha sido rechazado por el cliente o el departamento de inspección del comprador y está esperando ser enviado al proveedor para su reparación o reemplazo [CSCMP, 2013].

En su conjunto, logística inversa se puede clasificar por devolución y recuperación en una cadena de suministro. La devolución es una logística de "empuje" producida por todos los distribuidores o clientes finales; y la recuperación impulsada por los fabricantes se denomina logística de "extracción" [Chouinard et al., 2005]. La logística de "empuje" involucra principalmente productos e información de flujo inverso; la reclamación del cliente tiene como objetivo la retroalimentación de la información del producto [Ji, 2008].

La Asociación para la Gestión de la Cadena de Suministro [APICS, 2017] tiene modelada la gestión de devoluciones como proceso macro o de nivel uno de jerarquía. Establece que son los procesos asociados con el traslado de material de un cliente a través de la cadena de suministro para abordar defectos en el producto, pedidos o fabricación, o para realizar actividades de mantenimiento. A su vez están formados por subprocesos

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

de nivel dos donde están definidas de manera más detallada las actividades a realizar, las mejores prácticas, habilidades que deben poseer las personas encargadas de estas funciones, así como los indicadores de desempeño de cada una:

1. Devolución de productos defectuosos al proveedor: se establece según lo definido por los reclamos de garantía, retiro de productos, producto no conforme u otras políticas similares, incluido el reemplazo apropiado. El producto devuelto defectuoso admite cualquier tipo de producto que no se ajuste a las especificaciones (incluida la no conformidad con el pedido, como una entrega tardía o incorrecta). Las reglas comerciales de la empresa determinan la definición de "defectuoso". La disposición física del producto puede no ser parte del proceso de devolución. Entre las buenas prácticas que se aplican en el sector industrial para esta actividad se encuentran: el acceso móvil a la información, política de devolución incluida con el documento de envío y codificación de barras/RFID.
2. Devolución de producto para mantenimiento, reparación y operaciones (MRO) al proveedor: según lo definido por los planes de mantenimiento o la ocurrencia o anticipación del riesgo de falla. En general, se espera que los activos de la empresa administrados a través de un proceso de MRO se reacondicionen para que estén en condiciones de uso y se vuelvan a poner en servicio. La disposición física del producto puede no ser parte del proceso de devolución.
3. Devolución de producto en exceso al proveedor: La devolución de inventario en exceso o viejo o productos obsoletos según lo definido por los términos y condiciones de un contrato de cliente / proveedor. La intención de las devoluciones de productos en exceso es reasignar el inventario a una ubicación u organización que pueda vender el producto que se considera en exceso en la ubicación actual. La disposición física del producto puede no ser parte del proceso de devolución.

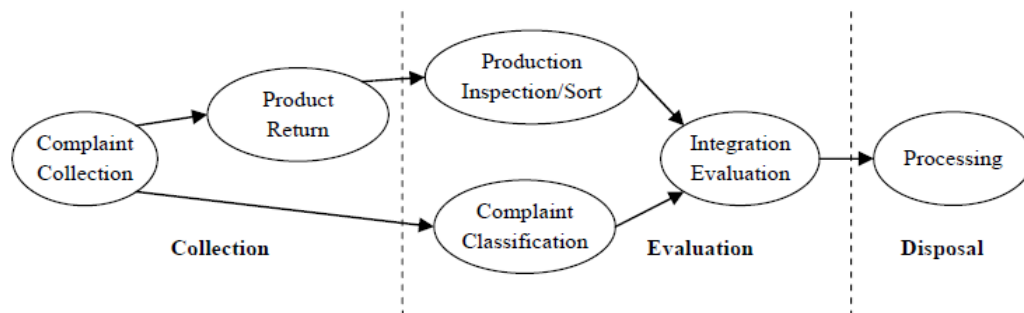
## CAPÍTULO 2: MARCO TEÓRICO

Estos son los procesos de devolución que regulan las actividades a realizar por clientes. Paralelos a cada uno de ellos están los procesos que rigen las actividades que deben desarrollar los proveedores para gestionar las devoluciones (autorizar, programar, recibir y transferir) que se les solicitan en dependencia de la característica del producto.

### 2.2.1 Gestión de reclamaciones

El desperdicio significa una mayor pérdida económica. Por lo tanto, creará más valor para la organización si las empresas de fabricación pueden reducir el desperdicio y utilizar plenamente todos sus recursos [Lai y Cheng, 2016]. Es importante asumir prácticas de análisis de defectos a los productos que envían los proveedores para que puedan eliminarse del producto o proceso y evitar futuros defectos en los mismos. Una que vez que han sido identificados se debe proceder a su reclamación y tratamiento.

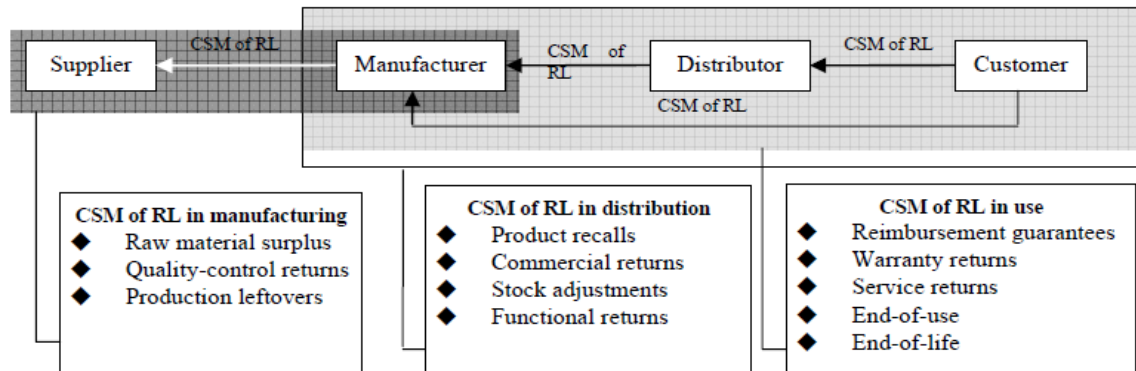
Ji et al. (2008) exponen que los diferentes patrones de aparición de logística inversa en una cadena de suministro afectan la aplicación del servicio de gestión de reclamaciones, que se puede dividir en tres aspectos: fabricación, distribución y uso, como muestra la **Figura 2.1**.



**Figura 2.1** Servicio de gestión de reclamaciones en una cadena de suministro [Ji, 2008]

El mismo autor argumenta que uno de los elementos importantes en la estrategia de logística inversa es la gestión de reclamaciones, que a través de un proceso de implementación logra tres funciones: recolección, evaluación y eliminación según muestra la **Figura 2.2**.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN



*Figura 2.2* Proceso de atención a reclamaciones. [Ji, 2008]

Como a los efectos de esta investigación es el apartado más interesante, sus subprocesos y buenas prácticas se muestran en la **Tabla 2.1**.

Subprocesos	Alcance	Buenas prácticas
1. Identificar la condición del producto defectuoso.	El proceso en el que el cliente utiliza políticas planificadas, reglas comerciales e inspección de las condiciones operativas del producto como criterios para identificar y confirmar que el material excede los requisitos defectuosos	<ul style="list-style-type: none"> <li>• Seguimiento de lotes</li> <li>• Selección perfecta guardada</li> <li>• Desarrollo de productos</li> <li>• Órdenes técnicas electrónicas y especificaciones del producto</li> </ul>
2. Disposición del producto defectuoso.	El proceso en el que el cliente determina si debe devolver el artículo defectuoso y la fuente de contacto adecuada para obtener una autorización de devolución.	<ul style="list-style-type: none"> <li>• Autorización de devolución remota</li> <li>• Seguimiento de lotes</li> </ul>

*Tabla 2.1* Actividades para devolver producto defectuoso al proveedor [APICS (2017)]

3. Solicitar autorización de devolución de producto defectuoso.	El proceso por el cual un cliente solicita y obtiene autorización, del último titular conocido o centro de devolución designado, para la devolución de un producto defectuoso. Además, el cliente y el último titular conocido o el centro de	<ul style="list-style-type: none"> <li>• Seguimiento de lotes</li> <li>• Política de devolución incluida con el documento de envío</li> </ul>
---	---	---

## CAPÍTULO 2: MARCO TEÓRICO

	<p>devolución designado discutirían las condiciones habilitantes tales como reemplazo o crédito de devolución, embalaje, manejo, transporte y requisitos de importación / exportación para facilitar la devolución eficiente del producto defectuoso.</p>	<ul style="list-style-type: none"> <li>• Se requiere autorización de devolución</li> <li>• Autorización de devolución proactiva</li> <li>• Autorización de devolución remota</li> <li>• Devoluciones preventivas</li> <li>• Piscina de repuestos rotativos</li> </ul>
<p>4. Programar envío de productos.</p>	<p>El proceso en el que el cliente desarrolla el cronograma para que un transportista recoja y entregue el producto defectuoso. Las actividades incluyen la selección del transportista y las tarifas, la preparación del artículo para la transferencia, la preparación de la documentación de programación y la gestión de la administración general de la programación.</p>	<ul style="list-style-type: none"> <li>• Seguimiento de devoluciones electrónicas</li> <li>• Autorización de devolución remota</li> <li>• Se requiere autorización de devolución</li> <li>• Optimización de la carga de retorno</li> </ul>
<p>5. Devolver producto.</p>	<p>El proceso en el que el cliente empaqueta y manipula el producto defectuoso en preparación para el envío de acuerdo con condiciones predeterminadas. Luego, el cliente proporciona el producto al transportista que transporta físicamente el producto y su documentación asociada al último poseedor conocido o centro de devolución designado.</p>	<ul style="list-style-type: none"> <li>• Seguimiento de lotes</li> <li>• Recuperación de proveedores</li> <li>• Seguimiento de devoluciones electrónicas</li> <li>• Piscina de repuestos rotativos</li> </ul>

*Tabla 2.1 Continuación*

# ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

## 2.3- Programación Visual Basic y Excel

### 2.3.1-Historia

Visual Basic es un programa desarrollado por Alan Cooper para Microsoft, que tiene un lenguaje de programación propio, llamado Basic. Dicho programa surgió por primera vez en 1991 como solución para problemas sencillos que fueron surgiendo, tales como la creación de programas gráficos. Como su propio nombre nos indica, se trata de una herramienta muy visual e intuitiva, podemos ver su logo en la **Figura 2.3**.



*Figura 2.3 Logo de VisualBasic.Net [Saavedra, 2008]*

Hoy en día, la aplicación únicamente funciona para Windows. Mediante su uso, nos facilitará la fácil creación de botones, menús, ventanas... solo arrastrando los elementos. A continuación, podremos modificar los comportamientos, apariencias y posiciones de los mismo, haciéndolo de forma visual, así como mediante lenguaje de programación.

Para ello, el programa se apoya en la utilización de paradigmas como el lenguaje orientado a objetos, así como el orientado a eventos. Usualmente este programa es considerado como un Rapid Application Development (RAD), ya que nos permitirá la creación de aplicaciones rápidamente, siendo esto muy útil por ejemplo para los prototipos [Secumate, 2011].

Visual Basic (VBa) es considerado un lenguaje de propósito general, ya que gracias a su versatilidad nos permite crear tanto juegos, como aplicaciones de escritorio, utilitarios, aplicaciones para

## CAPÍTULO 2: MARCO TEÓRICO

manejar bases de datos y mucho más. Muchas personas afirman que al querer abarcar tantos campos no acaba siendo muy fiable para ninguno de ellos, pero la realidad es que la aplicación acaba cumpliendo con cualquiera de las finalidades antes mencionadas. Como su propio nombre indica, se trata de un programa muy visual, que incluye una interfaz de desarrollo gráfico, pero, sin embargo, la gente puede llegar a pensar que la palabra BASIC significa que es un programa muy básico, pero esto no es así. Realmente el significado de esta palabra es “Beginner`s All-purpose Symbolic Instruction Code”, o lo que es lo mismo, código de instrucciones de uso universal para principiantes. Basándonos en esto, diremos que VBa se trata de un programa que actúa como intermediario entre el ordenador y el operador que lo explota. De esta forma, el computador traducirá lo que establezcamos que queremos que haga y le dará instrucciones específicas al ordenador para que pueda llevarlas a cabo, traduciéndolas a código binario.

Anteriormente comentamos que la creación del programa se le atribuye a Alan Cooper, pero ya se empezó a desarrollar en 1964 por parte de los estadounidenses John Kemeny y Thomas Kurtz, donde fue creciendo en aceptación gracias a la incorporación de herramientas muy potentes tales como Tiny BASIC y Microsoft BASIC, que ayudaron a catapultar este lenguaje ya que lo convirtieron en la primera lengua franca de los microordenadores. A partir de aquí, el lenguaje fue evolucionando y transformándose, puesto que su primer objetivo era ayudar a todo aquel público que quisiera adentrarse por primera vez en el mundo de la programación.

Posteriormente, en 1987, apareció una de las versiones más aclamadas del mismo, el QuickBasic, cuyas primeras versiones no eran estructuradas, sino interpretadas. Esta versión le hizo convertirse en un lenguaje de programación aún más popular, puesto que tenía los conceptos fundamentales de los otros lenguajes de programación propios de la época, pero se podía aprender de una manera mucho más sencilla y más aún a partir de 1991. Fue entonces cuando, tras muchas modificaciones iniciales, Alan Cooper lanzó la versión 1.0, que facilitaba en gran medida la creación de aplicaciones gráficas. Por esto se le considera el padre del lenguaje. Con esta versión, el desarrollo de aplicaciones para Windows dio un paso de gigante, tanto con la facilidad para creación de las mismas como para el tiempo ahorrado a la hora de crearlas.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

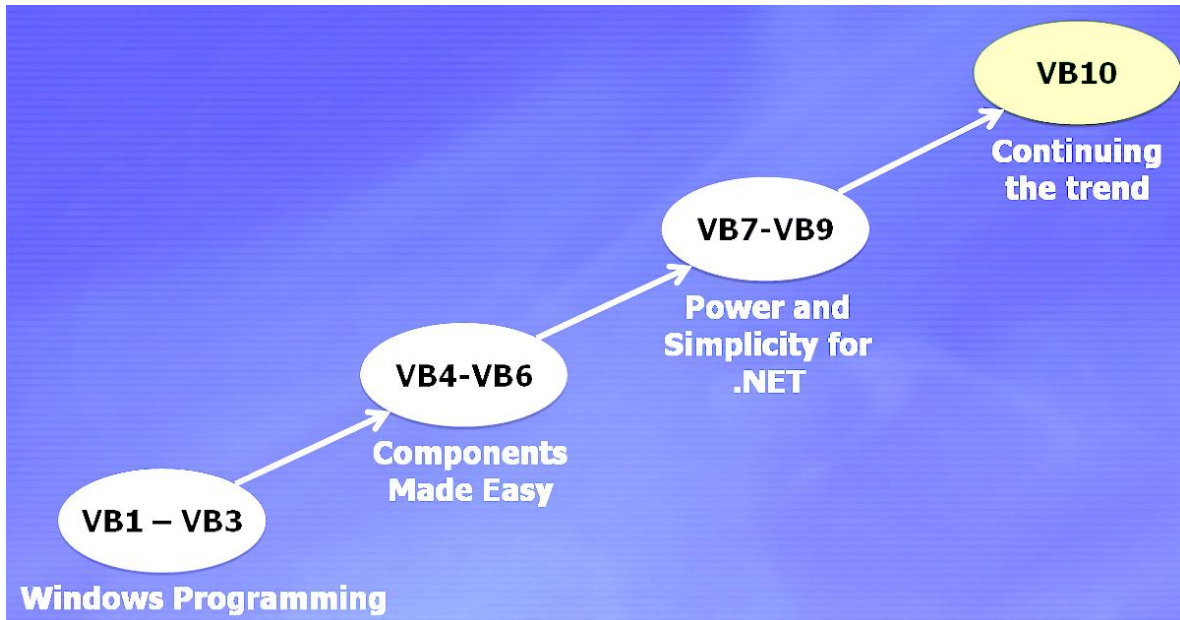
Cuando se iba a lanzar el producto al mercado, Microsoft tenía pensado incluir una gran cantidad de características, pero su equipo fue forzado a acortarla para hacerlo en un plazo corto de tiempo. Por esto mismo, las únicas modificaciones que esta versión incluyó fueron la tecnología Embedded Basic y un compilador que fue inicialmente desarrollado para Windows 3.0 pero no se llegó a incluir. Fue un año después cuando Microsoft desarrolló una herramienta muy potente para cubrir las exigencias del cliente, cuyo nombre era “Thunder”, pero durante esa época se produjo un gran impacto en la industria informática que afectó a Windows en gran medida, y no fue hasta diez años después que VBa logró un cambio muy significativo en el desarrollo de aplicaciones para Windows.

A partir de aquí surgió un grupo de seguidores que empezó a estandarizar la aplicación, creando una serie de bibliotecas de código y añadiéndolos como componentes de VBa. Son los llamados controles personalizados, expresados como VBXs. Únicamente cuando se aunaron tanto la aplicación como los componentes creados por sus seguidores, fue cuando la misma paso de ser un logro de software a un descubrimiento tecnológico, provocando a su vez que surgieran nuevas versiones con mucha mayor asiduidad. La versión 3.0 incluyó herramientas para facilitar el acceso a los datos además de una interfaz gráfica mucho más cómoda e intuitiva. La 4.0 ya incluía ejecutables de 16 y 32 bits. Hoy en día, la fuerza de poder combinar el lenguaje visual y la sencillez de BASIC permite llevar a cabo aplicaciones muy robustas que lo convierten en una alternativa muy viable para los programadores de más avanzado nivel.

El desarrollo de la aplicación fue continuo hasta la aparición de Visual Basic .NET, en 2002, podemos observar un pequeño esquema de la evolución del programa en la **Figura 2.4**. En esta versión se logró una orientación a objetos muy depurada, así como la posibilidad de crear servicios web, convirtiéndolo en un kit desarrollador altamente estratégico, basándose en simplificar y hacer aún más coherente el lenguaje de la aplicación, haciéndolo más sencillo de leer y entender, agregando nuevas características y ayudando a los programadores a evitar errores capacitándolos para crear aplicaciones más sólidas y mucho más fáciles de depurar [Saavedra, 2008].



## CAPÍTULO 2: MARCO TEÓRICO



*Figura 2.4 Evolución de VisualBasic [Akkandi, 2009]*

### 2.3.2-Características

La popularidad de este programa radica en la gran multitud de características que este tiene, tales como su lenguaje propio, Basic, que fue creado con miras a los que se inician en programación. También cabe destacar la gran adaptabilidad que este tiene, pudiendo ser utilizado para proyectos de una gran diversidad con respecto al propósito, al entorno operativo y a la complejidad. Como ejemplo de esto, podremos afirmar que la aplicación puede ser usada tanto para juegos y aplicaciones web llamativas como para simples utilidades de cinco líneas. Conociendo todas las características de este programa podrás saber si realmente te conviene utilizarlo o no para el proyecto que quieras emprender. Entre ellas destacaremos las siguientes:

**·Está orientado a los objetos**

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

Cuando programamos con Visual Basic deberemos enfocarnos en los objetos antes que en los procedimientos. Si se quiere recibir algo por parte de un usuario, lo haremos mediante el uso de un objeto, que a su vez puede contener otros objetos. Ejemplos de esto son los propios botones de comando, los botones de opción y los cuadros de texto.

### **·Es conducido por eventos**

Al realizar la programación orientada a los objetos, tendremos que decidir los eventos a los cuales cada uno de los objetos tiene que responder, así como posteriormente decidir cuáles serán las respuestas de estos objetos a cada uno de los eventos asociados. Como ejemplo de esto escribiremos el siguiente código respondiendo a la selección de un botón:

```
If (CommandButton1.Checked) Then  
  
MsgBox (“Has hecho una buena elección”)  
  
End If
```

### **·Está orientado a la familia**

Como ya hemos mencionado, cada objeto puede tener otros objetos asociados, haciendo que esta relación de familias de objetos permita una mejor comprensión del código de programación. Por ejemplo, el siguiente fragmento de código se desactivará y hará que todos, excepto uno de ellos, puedan incluirse en un formulario:

```
If No (TypeOf ct is Button) Then  
  
Ct.Allowed = No Ct. Allowed  
  
End If Next
```

Mediante esto podremos saber cuáles son los controles que están habilitados y cuáles no lo están.

·Diseñador de formularios WYSIWYG

## CAPÍTULO 2: MARCO TEÓRICO

Otra característica importante es que el propio programa incluye un diseñador de formularios WYSIWYG (What You See Is What You Get), mediante el uso del cual podremos crearlos de forma muy visual y rápida, mientras VB creará el código del mismo en un segundo plano. Esto se podrá llevar a cabo accediendo a través del menú proyecto → Agregar ventana de formulario. Podremos mover los controles del formulario mediante el uso de una paleta, para después darles formato y posición.

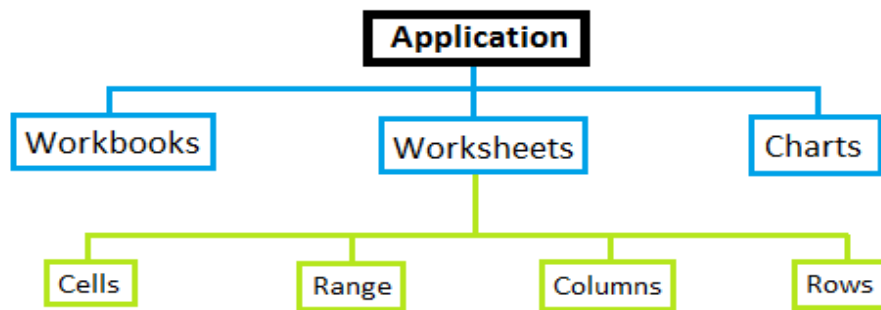
### **·Agregar códigos a cada objeto**

Si hacemos doble clic en algún objeto en nuestra ventana de diseñador, podremos revelar el esqueleto del código para dicho objeto. Una vez hecho esto, solo tendremos que rellenar los espacios en blanco para configurar la programación del mismo. Si por el contrario creamos un botón para rellenar un formulario, deberemos hacer doble clic en la ventana de diseño de formularios, accediendo a la subrutina del mismo. Aquí será donde establezcamos lo que queramos que ocurra cuando dicho botón sea pulsado [Koltow, s.f.].

### 2.3.3-Objetos y eventos

En Visual Basic, un objeto será aquello que represente algún elemento dentro de la aplicación, tales como una celda, un informe, un gráfico, una hoja de cálculo o un formulario, como se puede apreciar en la **Figura 2.5**. Para poder utilizarlo, se deberá identificar previamente, pudiendo aplicar uno de los métodos del mismo o cambiar el valor de las propiedades.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN



*Figura 2.5 Clasificación de elementos utilizados en VBa [Ortiz, s.f.]*

Si hablamos de una colección, nos referimos a un objeto que a su vez engloba varios objetos, que no tienen por qué ser necesariamente del mismo tipo. Por ejemplo, la colección `Workbooks` será un objeto que incluya todos los objetos `Workbook` abiertos, mientras que la colección `Forms` incluirá todos los objetos `Form`.

Identificaremos los elementos de una colección por número o por nombre. Si utilizamos el siguiente código identificaremos el primer objeto `Workbook` abierto:

```
Sub CloseFirst()  
Workbooks(1).Close  
End Sub
```

Sin embargo, si lo queremos identificar mediante el nombre, lo haremos de la siguiente manera [Microsoft, 2018]:

```
Sub CloseForm()  
Forms("MyForm.frm").Close  
End Sub
```

En nuestro ámbito, cuando hablemos de un evento, nos referiremos al mensaje que se envía desde un objeto de un programa al bucle principal del mismo, haciéndole saber que ha sucedido algo. Esto que sucede, puede variar ampliamente, siendo una referencia a que el

## CAPÍTULO 2: MARCO TEÓRICO

usuario ha hecho clic o incluso un reloj que llega a cero haciendo una cuenta atrás. Nuestro programa capturará lo sucedido y tomará una serie de decisiones a raíz de esto.

Existe una gran cantidad de eventos precodificados, pero también pueden establecerse desde el punto de vista del usuario. Los eventos pueden referirse a cualquier aspecto de programación o a la propia interacción entre distintos objetos del código. Entre estas interacciones, cabe destacar la que sucede entre el cursor y un control de formulario. Entre los distintos eventos posibles destacaremos los siguientes:

### **·Disparo**

Los posibles eventos pueden producirse en una gran cantidad de situaciones, algunos resultan de la interacción con el propio usuario, como al hacer clic o al realizar una entrada en el teclado. Si nos fijamos, por ejemplo, en el sistema operativo de Windows, nos daremos cuenta de que también envía eventos, al notificar cuando se minimiza una ventana o cuando se superpone otra a la ya abierta. A su vez, también pueden ser los objetos los desencadenantes de que suceda un evento, por ejemplo, al alcanzar cierta variable un determinado valor. También se pueden programar eventos para hacer algo, mediante algo a lo que denominaremos “raising”.

### **·Declaración**

Si queremos crear nuevos eventos, lo haremos mediante el uso de comandos de declaraciones. Un ejemplo de este tipo de comando sería el siguiente:

```
Public Event OptionChanged(ByVal Name As String, ByVal Number As Integer)
```

Programando este evento manualmente, lograremos que envíe dos valores al cambiar de opción, las cuales serían el nombre de la opción que cambiamos y el valor al que se establece.

### **·"Raising" o elevación de eventos**

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

Cuando hablamos de elevar un evento, nos referimos al hecho de provocar que este suceda. Podremos hacerlo mediante el uso de únicamente una línea de código, tal como la siguiente, que lanzará el evento enviando la información relativa al mismo en un procedimiento de controlador:

```
RaiseEvent OptionChanged("Length", 28)
```

### **Operarios informáticos**

El procedimiento que se lleva a cabo cuando un evento es lanzado, se denominará controlador. Este, será capaz de relacionar la información, tanto procesándola como haciéndosela llegar al propio usuario. Para poder llevar esto a cabo, será necesario que el evento esté interrelacionado con su operario informático, haciendo así posible que el programa sea consciente que se comunican entre sí [Koltow, s.f.].

### 2.3.4-Ejemplos de código

A continuación, vamos a exponer un sencillo ejemplo de código en VBA, en el que crearemos un programa que a su vez creará una variable de tipo string, destinada a hacer una llamada a un mensaje con el fin de mostrar un texto en pantalla. Para ello, crearemos un nuevo programa con el siguiente código:

```
Option Explicit On
```

```
Public Class Form1
```

```
    Dim mensaje as string
```

```
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
```

```
        Mensaje="Bienvenido a este programa"
```

## CAPÍTULO 2: MARCO TEÓRICO

```
MsgBox(mensaje)
```

```
End Sub
```

```
End Class
```

La sintaxis “Option Explicit On” servirá para forzar la declaración de todas las variables a nivel de módulo.

La sintaxis “Dim mensaje as string” servirá para declarar una variable de tipo alfanumérico que puede tener una longitud variable.

La sintaxis “Private Sub Form1\_Load()” se utilizará como procedimiento, para abrir un módulo de código de VBA.

Por lo tanto, la estructura de nuestro programa será Forzar declaración → Declaración de variables → Procedimientos. Cuando hablamos de procedimientos, nos referimos a una parte del código que se ejecutará en un momento concreto debido a una acción. Entre las acciones más significativas podemos encontrarnos que es llamado desde alguna parte del programa, o que sucede un evento que hace que este procedimiento se ejecute.

En nuestro caso, podemos observar que el programa se ejecuta cuando se lleva a cabo la carga del formulario, por lo que el programa imprimirá el mensaje cada vez que lo ejecutemos. La parte en la que escribimos mensaje = “Bienvenido a este programa” hace que nuestra variable alfanumérica declarada anteriormente guarde este contenido en su interior, y MsgBox (mensaje) provoca que el programa lo sitúe en el interior de una caja de mensajes y sea mostrado al usuario en cuestión [Rancel, s.f.].

A continuación, explicaremos otro sencillo ejemplo en el cual se deberá calcular la distancia que recorre un determinado automóvil. Para ello, utilizaremos el siguiente código:

```
Module Module1
```

```
Dim v, t, d, As Double
```

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

```
Sub Main()  
  
    Console.WriteLine("Introduzca velocidad")  
  
    v=Console.ReadLine  
  
    Console.WriteLine("Introduzca tiempo")  
  
    t=Console.ReadLine  
  
    d=v*t  
  
    Console.WriteLine("La distancia recorrida es {0}", d)  
  
    Console.ReadLine()  
  
End Sub
```

End Module

Según nuestro código, le consultaremos al usuario la velocidad y el tiempo durante el cual el vehículo se ha desplazado, y lo almacenaremos en las variables v y t. A continuación, realizaremos la multiplicación entre ambos para obtener la distancia de desplazamiento y le diremos a la consola que lo imprima en pantalla como respuesta final para nuestro usuario [Weebly, 2013]

Si lo que queremos es, por ejemplo, llevar a cabo un bucle que se repita hasta que una determinada expresión se cumpla, tendremos dos opciones, utilizar la sentencia For...Next o la sentencia Do...Loop. Por ejemplo, mediante el siguiente código, lograremos que un número inicial al que le daremos un valor de 9 unidades, vaya descontando una unidad tras cada repetición, hasta que el número adquiera el valor 0, momento en que se saldrá del bucle y nos aparecerá un mensaje en pantalla que afirmará "Se alcanzó el valor 0 tras 9 repeticiones", puesto que mostrará las dos variables que hemos declarado anteriormente, tanto número como contador [Programas Perú, 2018]

```
Sub EjemploDoWhile()  
  
    Dim contador As Integer  
  
    Dim numero As Integer
```



## CAPÍTULO 2: MARCO TEÓRICO

```
numero = 9
Do Until numero = 10
    If numero <= 0 Then Exit Do
        numero = numero - 1
        contador = contador + 1
Loop
MsgBox "Se alcanzó el valor " & numero & " tras " & contador & " repeticiones."
End Sub
```

### 2.3.5-Ventajas y desventajas de usar VBA

Entre las principales ventajas del uso de Visual Basic, podemos destacar que es relativamente fácil aprender cómo usarlo, que es uno de los lenguajes de programación más utilizados hoy en día, por lo que será mucho más sencillo a la hora de buscar documentación por si nos surgiera alguna duda de cómo emplearlo, que tiene acceso a casi todos los sistemas de la plataforma Windows y podrá comunicarse e interrelacionarse con todos ellos, que añade soporte para poder ejecutar scripts, haciéndonos mucho más fácil la lectura de documentos, y que como es de Windows, añade su implementación y diseño de formularios. Otra gran ventaja de nuestra aplicación, consiste en que es fácilmente extensible mediante librerías DLL y componentes ActiveX de otros lenguajes. Además, esta aplicación nos permite desarrollar aplicaciones grandes y complejas, pero a su vez también hace posible que sea un entorno adecuado para llevar a cabo pequeños prototipos rápidos.

Por el contrario, este programa también presenta algunas desventajas, tales como las librerías Runtime.dll, que tienen problemas de versionado. Además, aunque se trate de una aplicación que está orientada a objetos, tiene muy escaso soporte, y para llevar a cabo aplicaciones que se relacionen con otros sistemas de Windows, será necesario hacer llamadas a las mismas desde el propio programa. Entre otras desventajas, también cabe

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

destacar que depende de frágiles y complejas entradas de registro COM, que únicamente Microsoft puede decidir la evolución de este lenguaje, que solo genera ejecutables para Windows, que tiene una sintaxis poco flexible y además los ejecutables generados son relativamente lentos [Cárdenas, 2016], [Carrera, 2015], [Jesuïtes Educació, 2019].

### 2.3.6-Alternativas a Visual Basic

Entre las principales alternativas a Visual Basic, podemos destacar las siguientes:

#### **·VSCodium:**

Este programa nos permitirá descargar programas de código binario precompilado con código de VBA excluyendo las extensiones de Microsoft. A su vez, se elimina la traza de telemetría que Microsoft suele añadir a sus productos y muchas de las herramientas propias de la marca que se encuentran en el programa original. Hay que añadir, que esta opción es de fuente más abierta, pero por el contrario también es menos completa.

#### **·Code-OSS:**

Otra posible alternativa consiste en descargar el código del entorno de desarrollo de Microsoft. Una de las principales desventajas de este programa es que deberemos compilarlo nosotros mismos para finalmente obtener una versión prácticamente exacta a la de VSCodium.

#### **·Atom:**

Este es una de las aplicaciones de fuente abierta más populares. Inicialmente, se trataba de un proyecto independiente, pero posteriormente pasó a formar parte de Microsoft cuando esta adquirió GitHub en 2018. Se trataría de una aplicación muy similar a VSCodium, con la particularidad de que incluye métricas de seguimiento por defecto que se pueden

## CAPÍTULO 2: MARCO TEÓRICO

deshabilitar y no se imponen restricciones a las diferentes extensiones, lo que permitirá que no sea necesario adaptarse a una nueva modalidad de trabajo.

### **·Gnome Builder:**

Esta aplicación es propia de los sistemas operativos Linux y consiste en un programa muy sencillo y directo que además tiene soporte para docenas de lenguajes de programación. Como ventaja principal, cabe destacar que podremos encontrar una interfaz muy limpia, clara y minimalista.

### **·Geany:**

Se trata de un editor de texto muy sencillo que soporta más de cincuenta lenguajes de programación pero que no tiene muchas extensiones instaladas, pese a que alguna de ellas se pueda añadir mediante el uso de plugins.

### **·Brackets:**

Este también consiste en un editor de texto muy sencillo que ofrece soporte para tecnologías muy variadas. Gracias a esto, permitirá que los usuarios puedan llevar a cabo sus aplicaciones en casi cualquier lenguaje de programación.

### **·Che:**

Esta aplicación tiene su núcleo en la nube, por lo que será muy útil para desarrollar aplicaciones que estén liguen elementos que se encuentren subidos a esta [de Juana, 2020].



## CAPÍTULO 3: ISRINGHAUSEN

En este capítulo se tratará la historia de Isringhausen (**Figura 3.1**), pilares y fábrica.



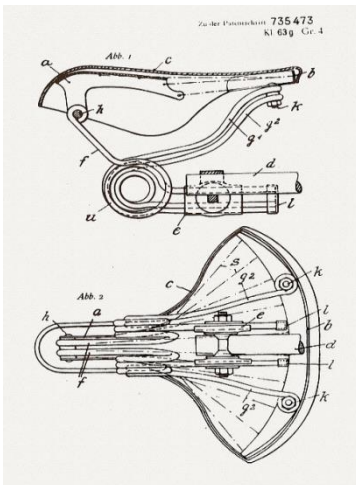
**Figura 3.1** Isringhausen [AUNDE Group SE, s.f.]

### 3.1- Historia de la empresa

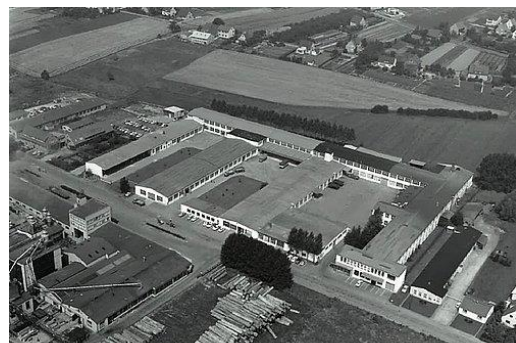
Isringhausen, también llamada Isri, es una empresa con más de 100 años de experiencia que se dedica a la construcción de asientos para vehículos industriales. A continuación, haremos un pequeño repaso por la historia de la empresa:

1919: *Se funda la compañía en Bielefeld*, como una empresa fabricante de asientos para bicicleta (**Figura 3.2**).

1957: *La compañía se traslada de Bielefeld a Lemgo* (**Figura 3.3**).



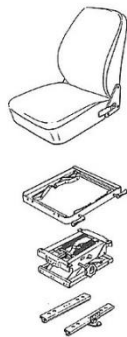
**Figura 3.2** Fundación [AUNDE Group SE, s.f.]



**Figura 3.3** Traslado a Lemgo [AUNDE Group SE, s.f.]

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

1970: *Desarrollo de sistemas de asientos modulares*. La introducción del principio modular en la tecnología de los asientos permite a ISRI ofrecer una nueva variedad de productos y una adaptación flexible a las diferentes necesidades del mercado (**Figura 3.4**).



**Figura 3.4** Asientos modulares  
[AUNDE Group SE, s.f.]



**Figura 3.5** Asientos automáticos  
[AUNDE Group SE, s.f.]

1987: *Primer cinturón de seguridad de tres puntos*. ISRI suministra el primer asiento del mundo para vehículos comerciales con un cinturón de seguridad de 3 puntos integrado (**Figura 3.6**).



**Figura 3.6** Primer cinturón de tres puntos [AUNDE Group SE, s.f.]

1991: *Adquisición por parte del grupo AUNDE*. El Grupo AUNDE, proveedor de materiales de superficie innovadores, adquiere el Grupo ISRINGHAUSEN (**Figura 3.7**).

## CAPÍTULO 3: ISRINGHAUSEN



*Figura 3.7 Adquisición por parte de Aunde [AUNDE Group SE, s.f.]*

1993: *Primer asiento ajustable eléctricamente para vehículos comerciales.* ISRI suministra el primer asiento ajustable eléctricamente del mundo para vehículos comerciales (**Figura 3.8**).

1994: *Asientos para furgonetas y vehículos comerciales ligeros.* Ampliación del segmento de productos: inicio del desarrollo y la producción de asientos para furgonetas y vehículos comerciales ligeros (**Figura 3.9**).



*Figura 3.8 Primer asiento ajustable eléctrico [AUNDE Group SE, s.f.]*



*Figura 3.9 Ampliación del segmento [AUNDE Group SE, s.f.]*

2001: *Generación de asientos ISRI NTS (nuevo asiento para camiones).* Inicio de la generación de asientos NTS - Uno de los productos más exitosos de ISRI (**Figura 3.10**).

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

2005: *La nueva generación del cinturón de tres puntos.* El cinturón de 3 puntos integrado sigue evolucionando (**Figura 3.11**). El ajuste mecánico de la altura proporciona un nuevo confort en el puesto de trabajo de cada conductor.



**Figura 3.10** Asiento NTS [AUNDE Group SE, s.f.]



**Figura 3.11** Cinturón de tres puntos [AUNDE Group SE, s.f.]

2008: *Fin de la generación de asientos NTS.* El asiento NTS número 4 millones (**Figura 3.12**), sale de la cadena de montaje.

2013: *Adquisición de Asientos Esteban por parte de Isringhausen.* Con la adquisición de ESTEBAN, la línea de productos de Asientos de Pasajeros (**Figura 3.13**), se integró bajo la marca ISRINGHAUSEN.



**Figura 3.12** Último asiento NTS [AUNDE Group SE, s.f.]



**Figura 3.13** Asientos Esteban [AUNDE Group SE, s.f.]

2017: *Adquisición de GMA.* Se produce la adquisición de las partes de la empresa en Bünde (D) y Kaplice (CZ). Con la adquisición de GMA, se refuerzan las competencias y capacidades



## CAPÍTULO 3: ISRINGHAUSEN

en el procesamiento de metales asociados al montaje de asientos (**Figura 3.14**), y se aumenta la profundidad del valor añadido.

2018: *Inauguración de la nueva planta de turismos en Lemgo*. ISRI suministra a sus clientes desde la nueva nave de producción JIT de la última norma industrial 4.0. Podemos ver la nueva planta de Lemgo en la **Figura 3.15**.



**Figura 3.14** Metales para asientos  
[AUNDE Group SE, s.f.]



**Figura 3.15** Nueva planta de turismos en Lemgo  
[AUNDE Group SE, s.f.]

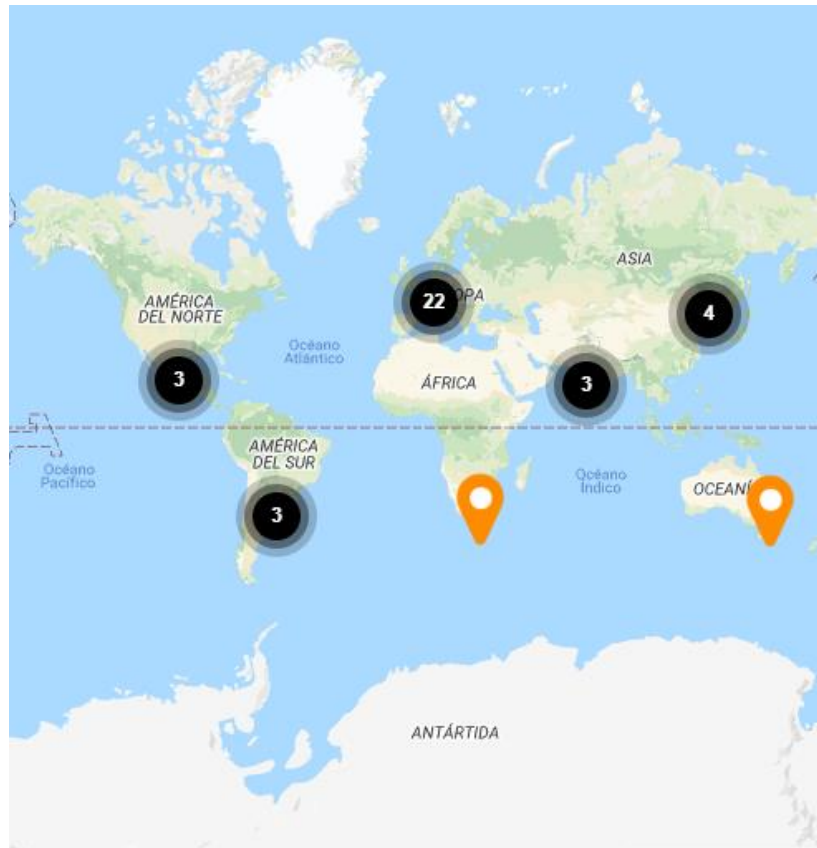
2019: *100 años de espíritu de equipo ISRI*. El ISRI celebra su centenario (**Figura 3.16**).

**ISRI**<sup>®</sup> **100** **JAHRE**  
ISRINGHAUSEN

**Figura 3.16** Centenario de Isringhausen [AUNDE Group SE, s.f.]

Actualmente, podemos encontrar plantas de la empresa en los cinco continentes, aunque predomine su presencia sobre todo en Europa, como se puede observar en la **Figura 3.17**, por lo que puede deducirse que el crecimiento de la empresa es continuo, ya que ha sabido transformarse a lo largo del tiempo para así poder ofrecerle a cada cliente lo que necesita, adaptándose a las necesidades de cada uno de ellos [We are Isringhausen - Isringhausen. (s.f.)].

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN



**Figura 3.17** Isringhausen en el mundo [AUNDE Group SE, s.f.]

### 3.2- Conceptos de asiento para la movilidad del mañana

Isringhausen ofrece un amplio catálogo de productos entre los que se incluyen los asientos de conductor, copiloto y pasajeros, tanto para camiones, como para furgonetas, autobuses e incluso maquinaria de trabajo.

La empresa siempre se ha centrado en ofrecer los más altos niveles de fiabilidad, calidad y eficiencia, procurando estar a la altura de los retos de sus clientes y contando con una experiencia excepcional en el desarrollo y fabricación, combinados con una continua innovación orientada al mercado. Partiendo de la idea original del producto, pasando por su estilo y desarrollo, y terminando con el prototipo y la producción en serie, ofrece a sus socios repartidos por todo el mundo servicios Just In Time (JIT) y Just In Sequence (JIS) a

## CAPÍTULO 3: ISRINGHAUSEN

medida. El objetivo principal de la misma será proporcionar a sus clientes la tecnología de los asientos del futuro.

Una frase destacada del jefe de operaciones de la empresa, la cual resume muy bien el espíritu de la empresa, es la siguiente; “En el futuro del transporte, el asiento seguirá siendo el elemento central del interior del vehículo. La funcionalidad y la personalización para el conductor y los pasajeros seguirán siendo el centro de desarrollo de los asientos.”

Los principales pilares en los que se basa la empresa son los siguientes:

·**Diseño y construcción:** Desarrollar soluciones de producto a medida en cooperación con los clientes, contando con un fuerte equipo de desarrollo con más de 200 personas con fuerte experiencia técnica en la empresa.

·**Ergonomía:** Los aspectos de salud, seguridad y confort son el núcleo del desarrollo de asientos de ISRI. Las funciones y necesidades biológicas del cuerpo humano junto con un cercano seguimiento de la situación del trabajo en concreto guía el diseño de los asientos.

·**Prototipos:** En su departamento de prototipos, un ejemplar de asiento real se crea con las especificaciones de cada cliente en concreto. Desde el diseño estructural y el modelado de la espuma, hasta el desarrollo de la cubierta.

·**Procesado del metal y tratamiento superficial:** En su producción en serie, ISRI cubre completamente la cadena de valor de los sistemas de asientos. Desde el taller de estampación y soldadura hasta el tratamiento superficial y ensamblado final.

·**Montaje de asientos:** Las nuevas tecnologías de producción y la amplia gama de fabricación de ISRI hacen posible la aparición de nuevas soluciones de producción para sus clientes.

·**Pruebas:** Para garantizar la máxima calidad y seguridad durante el desarrollo de los asientos, es necesario un laboratorio de medida y pruebas. Los bancos de pruebas de

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

resistencia a la flexibilidad y las pruebas de choque miden regularmente la fuerza estructural y durabilidad de sus productos.

**Sistemas de asientos:** Con sus tecnologías de producción basadas en el estándar de industria 4.0, los asientos a medida de conductor, copiloto y pasajero son fabricados en más de 50 plantas.

**Logística:** Manteniendo una cercana cooperación con los clientes y mediante un sistema de logística integral, los asientos son abastecidos de manera “Just in sequence”.

Por lo tanto, resumiendo todos estos puntos, podemos destacar que los asientos ISRI se desarrollan como equipamiento original en estrecha colaboración con los principales fabricantes de automóviles y vehículos comerciales para satisfacer las demandas específicas de cada tipo de vehículo y lugar de trabajo. La calidad, la fiabilidad y el confort son los elementos centrales del desarrollo de los asientos. Sus socios confían en sus más de 100 años de experiencia y en la fiabilidad de sus productos (*We are Isringhausen - Isringhausen*. (s.f.)).

### 3.3- Empresa y ubicación

Para nuestro estudio en particular, nos situaremos en la fábrica de montaje “Isringhausen S.A.” ubicada en la nave de IVECO Pegaso en Valladolid, destinada al montaje de asientos para su posterior abastecimiento sincronizado para la fabricación JIS (Just In Sequence) de las furgonetas IVECO Daily. Esta fábrica está compuesta por alrededor de 40 trabajadores repartidos en dos turnos para adecuarse al horario de apertura del cliente y en 2020 tuvo una facturación de 10.035.733€. A su vez, está compuesta por una fábrica de montaje y un almacén, en el que se guardan todos los productos entrantes mediante el uso de un WMS (Warehouse Management System) en los distintos huecos de las 13 posibles estanterías, posterior a haber descargado los mismos de los camiones de los distintos proveedores con la ayuda de dos carretilleros. Esto se llevará a cabo mediante un sistema de etiquetado

## CAPÍTULO 3: ISRINGHAUSEN

según el cual, al introducir el albarán, se crearán una serie de órdenes de trabajo mediante el programa informático que establecerá la ubicación en función de los diferentes tamaños posibles que pueda tener el embalaje.

Una vez almacenados los materiales, y para su posterior abastecimiento, se establece un sistema FIFO (First In First Out) para que al pedir los materiales los distintos carretilleros el propio sistema de WMS indique la ubicación del más antiguo al más nuevo. Una vez hecho esto, los carretilleros deberán preparar carros modulares con todos los productos semielaborados necesarios para llevar a cabo el picking según la demanda, adecuándose a un sistema de etiquetado que le advierte lo que tiene que poner en cada carro, para después traer todos ellos mediante un sistema Just In Time a la línea y que los operarios de montaje de la planta no queden desabastecidos en ningún momento. A su vez, se establece picking para los asientos de monoplaza, pero este se encuentra al lado del puesto, para que sean los propios operarios los que vayan cogiendo los materiales que les sean necesarios.

Entre los distintos proveedores principales de Isringhausen Valladolid, podemos destacar a Isringhausen Nibbia, situada en Italia, cerca de Milán, encargada de abastecer a la misma de todos los componentes metálicos necesarios para su posterior montaje, a excepción de las suspensiones, que las suministrará Isringhausen Lemgo, situada al sureste de Alemania. Además, el proveedor de fundas será Aunde, compañía dueña de la propia empresa Isringhausen y principal fabricante mundial de fundas para la industria del automóvil. A su vez, para el suministro de espumas para nuestros asientos, entrará en juego la empresa Toscana Gomma, situada también en Italia. Por último, añadiremos que el proveedor de los reposacabezas es Grammer.



## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

### 4.1- Manual de usuario

-Introducir datos:

A la hora de rellenar los diferentes campos para las piezas defectuosas achacables al proveedor, tendremos tres opciones diferentes:

·La **primera** será, en la hoja de inicio, introducir cualquiera de los tres campos, ya sea el tipo de pieza como la referencia o el proveedor. De esta forma, cuando le demos al OK, el Excel nos llevará a la hoja del proveedor correspondiente para que podamos introducir la referencia, el número de piezas defectuosas y el defecto que le corresponda.

·La **segunda** opción, de mayor complejidad, nos permitirá introducir una gran cantidad de piezas de cualquiera de los proveedores para llevar a cabo la exportación de todas ellas a la vez. Así, introduciremos los campos en blanco, los cuales son la referencia, la cantidad y el defecto, logrando que todos los demás se actualicen automáticamente. Cuando pulsemos el botón de exportar, lo primero que se llevará a cabo será una comprobación de que todos y cada uno de los materiales introducidos nos caben en los contenedores que tenemos abiertos. De no ser así, nos llegará un aviso por pantalla diciéndonos que el contenedor de un determinado proveedor está completo, y que debemos cerrarlo antes de poder continuar, haciendo visible la hoja del mismo para que únicamente tengamos que exportar dichos datos y borrarlos.

Si en algún momento se quisieran introducir más materiales de los que corresponden al tamaño de lote establecido, por ejemplo, añadiendo una lama a mayores a uno de los contenedores de Nibbia, deberemos hacer visible la “Hoja2” y modificar el tamaño de lote del propio proveedor para poder incluirlo.

·La **tercera** opción, algo más sencilla, consistirá en acceder desde la hoja de “ACCESO DIRECTO”. Este conjunto de hojas consiste en un sistema de botones por el que iremos avanzando y refinando nuestra búsqueda en función de si se trata de una pieza de biplaza

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

o conductor, metálica, funda o respaldo... hasta llegar a la pieza que estábamos buscando. Cuando pulsemos el botón de esta, nos desplazará hasta la hoja del proveedor correspondiente y se escribirá la referencia de la pieza seleccionada, no sin antes llevar a cabo una comprobación de que hay hueco en la caja. Una vez hecho esto, se deberán introducir las cantidades y los defectos observados antes de poder continuar con el registro de una nueva pieza. Para la elección del tipo de defecto, estaremos limitados a una serie de defectos que se repiten para cada proveedor específico, elegidos de un desplegable.

Si se quisiera incluir un otro botón para nuevos productos que comiencen a devolverse, habría que crear una hoja a la que acceder mediante los accesos directos de los botones previos en el campo específico que le corresponda, crear un "CommandButton" y copiar el código de alguno de los otros botones, modificando el número asignado por el que le corresponda según la lista de la "Hoja2".

A su vez, si se quisiera incluir un nuevo proveedor, habría que duplicar una de las hojas de los proveedores y cambiarla el nombre por la del nuevo, así como a su vez modificar los campos de la hoja que hagan referencia al proveedor, y establecer en la hoja 2 los posibles defectos correspondientes a las piezas proporcionadas por el mismo, modificando la columna de los mismos para que la misma también haga referencia a la nueva columna que hayamos insertado.

Es de suma importancia no dejar ningún campo vacío, ya que el registro quedaría incompleto y a la hora de exportar los datos se generaría un descuadre que haría que todas las siguientes referencias no fueran ligadas a sus defectos, fechas o cantidades.

-Cerrar contenedor y abrir uno nuevo:

Las hojas de los distintos proveedores tienen una serie de campos de escritura y otros de solo lectura. Los de escritura serán la referencia de la pieza, la cantidad y el defecto, donde



## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

deberemos registrar lo observado, y a partir de estos se generarán automáticamente los campos de número de pieza, descripción de la misma, fecha y número de reclamación. Una vez hayamos completado un cajón, deberemos proceder a imprimir la hoja para su correcta identificación, exportar los datos al Excel de calidad y eliminar los campos para poder empezar con un nuevo contenedor. Para esto se habilitan tres botones con los nombres “Imprimir”, “Exportar” y “Borrar datos”, respectivamente. Si pulsamos el botón “Borrar datos” el Excel nos lanzará una advertencia por pantalla que nos preguntará si estamos seguros. Para facilitar esta tarea, se crea otro botón que engloba estas tres acciones y que ya no nos preguntaría si estamos seguros, puesto que al ejecutarla es seguro que se hayan exportado los datos previamente.

Si en algún momento el número de reclamación no nos coincide con el que debería ser, deberemos situarnos en la celda k3 del proveedor en cuestión, que está oculta para evitar modificaciones involuntarias, y cambiarla por la que realmente corresponda. Haciendo esto, la próxima vez que cerremos el contenedor para abrir uno nuevo, este campo se actualizará automáticamente y no tendremos que volver a modificarlo.

### 4.2- Análisis del Antes-Después y desarrollo del programa

Antes de nada, comentaremos un poco lo que la empresa tenía antes de llevar a cabo esta intervención. Como se puede observar en la **Figura 4.1**, se trata de una hoja de Excel en la que los encargados van introduciendo manualmente uno a uno cada uno de los materiales defectuosos, teniendo que escribir tanto la referencia, como el nombre del material, la cantidad de productos defectuosos de la misma clase y con el mismo defecto, y por último el defecto que se presenta, discriminando todos ellos en una u otra hoja en función del proveedor asignado a cada pieza.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

ISRI		Lista de material bloqueado		Código: FORM 1341-03	
				Emitida por: Camino	
				Versión: 17.3.2020	
				Surtituye: 23.02.2011	
Devolver a proveedor:			Persona que selecciona:		
NIBBIA (cajón 1)			Fecha entrada zona bloqueo:		
N°	Referencia pi	Descripción pieza	Canti	Defecto	
1	01787000106	ESTRUCTURA MONO LV #W#WSUS DX	10	pasadas	
2	01786900106	ESTRUCTURA MONO LV #W#WSUS SX	5	pasadas	
3	01786900106	ESTRUCTURA MONO LV #W#WSUS SX	3	pasadas	
4	01787000106	ESTRUCTURA MONO LV #W#WSUS DX	11	pasadas	
5	01786900106	ESTRUCTURA MONO LV #W#WSUS SX	5	pasadas	
6		#H/D			
7		#H/D			
8		#H/D			
9		#H/D			
10		#H/D			
11		#H/D			

*Figura 4.1 Hoja de defectos por proveedor inicial*

Este proceso, tan repetitivo y lento, a su vez, provoca que los encargados se equivoquen cada vez más al rellenar esta hoja cada vez que haya una devolución, y registren un gran número de piezas englobándolas en una misma línea sin poder identificarlas de forma correcta, provocando que en proveedor el seguimiento de las piezas defectuosas también se convierta en un proceso más tedioso.

A partir de esto, el personal de calidad ha de recoger los datos aquí registrados para llevarlos a otro Excel representado en la **Figura 4.2** en el que registrará todos los datos introducidos anteriormente por el encargado, añadiendo la traducción del propio defecto en inglés e incluyendo a mayores el mes, la semana, el número de pieza asignado, asignándole un número de reclamación para hacérselo llegar al proveedor y el albarán en el que estas piezas estarán identificadas.

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

MES	SEMA	PI	REFERENCIA	tipo	CAUSA	col ni	ANTIC	RECLAMACIÓN	TRADA ZONA BLC	dev SAF	LIDA ZONA BLO	ÁN DE DE
1	5	57	017870D01006	M	thread damaged		6	Q21057	29/01/2021	*		DV2021007
1	5	1	017407D01010	M	dont fit jig		1	Q21057	29/01/2021	*		DV2021007
1	5	2	017407D01010	M	thread damaged		1	Q21057	29/01/2021	*		DV2021007
1	5	3	015769D02009	M				Q21057	29/01/2021	*		DV2021007
1	5	4	017870D01006	M	thread damaged		12	Q21057	29/01/2021	*		DV2021007
1	5	5	017408D01010	M	bad welded		1	Q21057	29/01/2021	*		DV2021007
1	5	6	013601D01000	M	unwelded		1	Q21057	29/01/2021	*		DV2021007
1	5	7	013601D01000	M	unwelded		1	Q21057	29/01/2021	*		DV2021007
1	5	8	017870D01006	M	thread damaged		2	Q21057	29/01/2021	*		DV2021007
1	5	9	015769D02009	M	deformed		1	Q21057	29/01/2021	*		DV2021007

**Figura 4.2** Excel inicial de calidad al que se exportan los datos

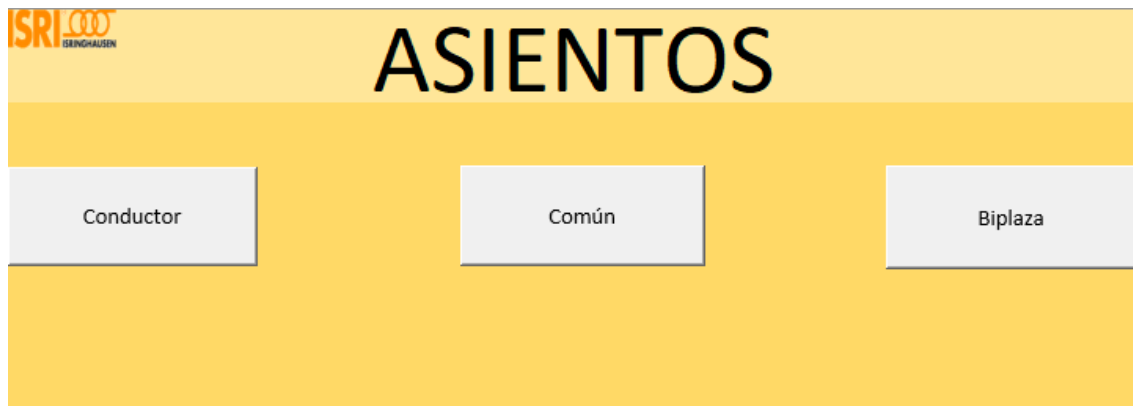
Ante la problemática que surge en la empresa debido a la falta de comunicación y a la pérdida de información en el cambio de turno, y viendo la posibilidad de automatizar todo este proceso repetitivo, aparece la necesidad de crear una aplicación para poder pilotar en todo momento las piezas defectuosas que se detectan en cada turno a causa de los distintos proveedores. Esto es debido a que en la empresa se utiliza un sistema de identificación de las mismas, en el cual se separan las piezas malas a causa de proveedor y las piezas malas porque se estropean durante el proceso de fabricación. Este sistema consiste en la utilización de un contenedor amarillo (yellow bin) y uno rojo (red bin), para situar en el amarillo las piezas malas por problemas en el proveedor, susceptibles de ser devueltas al mismo, y en el rojo las piezas malas por problemas en la línea.

De esta manera, cuando un operario reconoce una pieza mala, la aparta del proceso productivo situándola en uno de estos dos contenedores, para que a continuación el encargado haga una revisión y detecte los defectos que se han producido en cada una de ellas para poder identificarlas correctamente y así hacerle llegar los datos de las piezas no OK al técnico de calidad, para que este lleve a cabo el registro consiguiente de las mismos, y posteriormente al departamento de logística, con el fin de que pueda elaborar el albarán detallado de las piezas con la ayuda de los datos recibidos por parte del departamento de calidad.

Muchas veces, esto se convertía en una tarea muy pesada para el encargado al tener que registrar las piezas que anteriormente los operarios de la línea ya habían separado, y se podía observar un claro retrabajo. Mediante la creación de nuestra aplicación

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

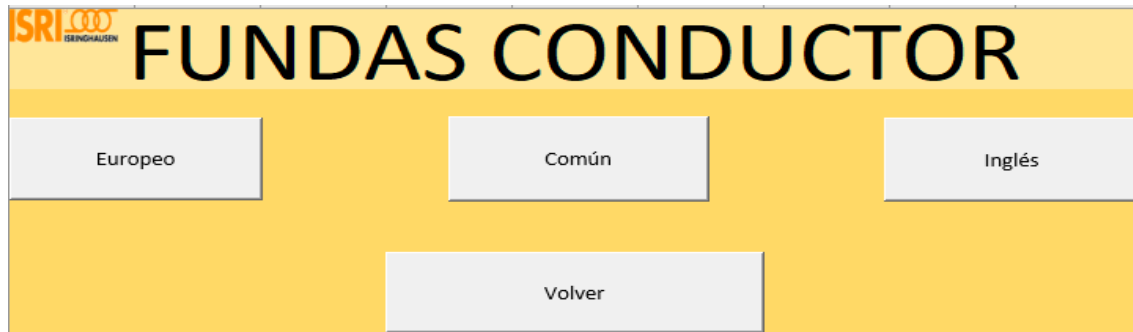
pretenderemos elaborar un programa de fácil comprensión para que los propios operarios, mediante el uso de una tableta, puedan identificar el defecto en el momento justo en que lo detectan. Para ello, se decidió crear un sistema de botones con la siguiente estructura, como se muestra en la **Figura 4.3**, para que pudieran entrar a la aplicación por el tipo de asiento en el que estaban trabajando y a partir de ahí ir seleccionando el tipo de pieza desplazándose por una serie de pestañas hasta llegar a un botón unívoco con el que pudieran seleccionar la pieza en concreto que les había salido defectuosa.



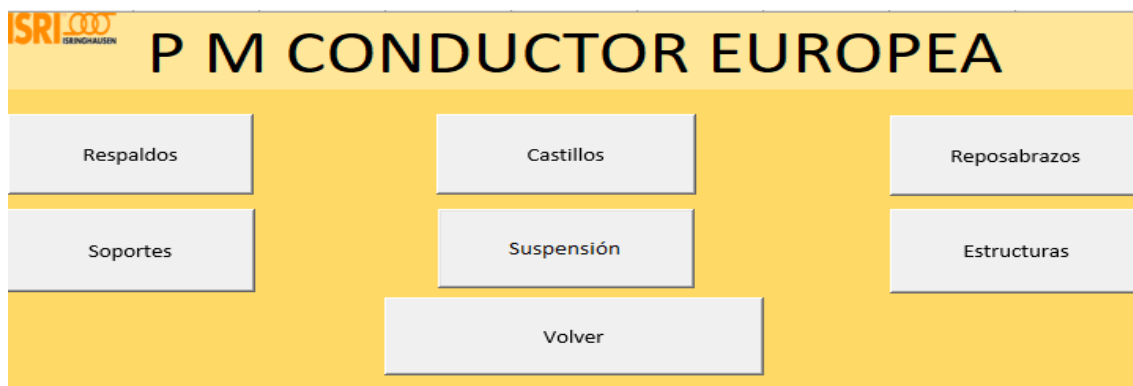
*Figura 4.3 Tipos de asientos*

Al hacer esto, se llegó a una complicación, debido a que existían algunas piezas que eran comunes para ambos tipos de asientos, como por ejemplo los reposacabezas. Para evitar tener que incluir dichas piezas por repetido en el listado de materiales, referencias y proveedores, se decidió crear un botón para incluir todas aquellas que se compartían entre conductor y biplaza. Asimismo, posteriormente surgía la misma problemática dentro de los menús de conductor y de biplaza, ya que existían determinados materiales comunes para los sistemas de conducción europeo e inglés, ya que muchos son iguales pero simétricos, como los reposabrazos, pero algunos son exactamente iguales, como alguna funda o espuma. De esta forma también se decidieron crear hojas para los sistemas comunes dentro de los apartados de algunas piezas de conductor o biplaza, como se puede observar en la **Figura 4.4** y en la **Figura 4.5**.

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN



*Figura 4.4 Fundas de conductor*



*Figura 4.5 Piezas Metálicas de Conductor Europeo*

Al pulsar este botón, una macro se desplazaría a la pantalla del proveedor de la pieza en cuestión, y les escribiría tanto la referencia del producto como el propio nombre, para que a continuación sólo tuvieran que añadir el número de piezas defectuosas y el defecto avistado, elegido de un desplegable con los defectos concretos posibles para esa familia de piezas, estandarizados para que las pequeñas diferencias de escritura no nos produzcan fallos posteriormente con el formato que se aprecia en la **Figura 4.6**. De esta forma se podrá eliminar el retrabajo llevado a cabo por parte del encargado.

Para eliminar también otro retrabajo identificado, se plantea la idea de incluir en la propia línea una etiquetadora que imprima números para que al situar la pieza en el contenedor amarillo se identifique la misma, para ya ir referenciada a su número y su defecto en particular dentro del contenedor al que se trasvasará a continuación. De esta forma,

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

tendríamos nuestro programa instalado en una tableta que podría comunicarse directamente con la etiquetadora, y cada vez que introduzcamos una pieza mala y su defecto, nos imprimiría dicho número, para que el operario no tuviera más que adherirlo a la pieza sin necesidad de hacer una identificación posterior por parte del encargado.

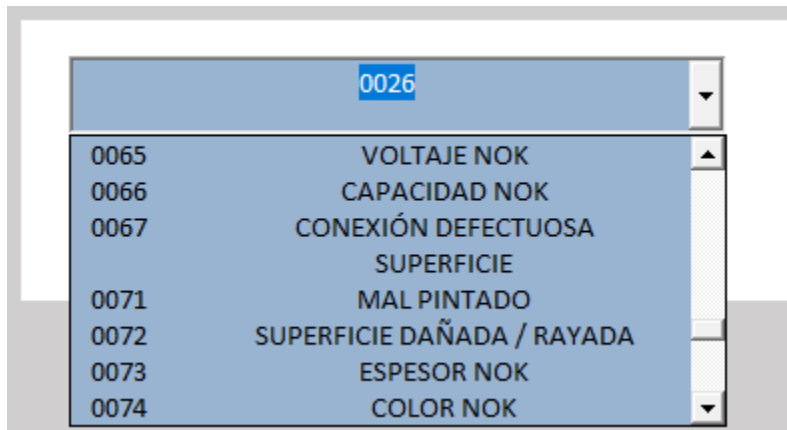
ISRI <small>GROUP</small>			Lista de material bloqueado			Código: FORM 1341-03 Emisión por: Camila Versión: 17.3.2020 Sustituye: 22.02.2019		
Devolver a proveedor:			Persona que selecciona:					
<b>NIBBIA</b>			Victor					
cajón			Fecha entrada zona bloqueo:					
			07/06/2021					
N°	Referencia	Descripción pieza	Cantid	Ubicaci	Tipo	Defecto		
1	321482-0100	PLACA PROTECCION DENTADO ESTRUCT. SW	1			desoldada		
2	118457-0102	REPOSABRAZOS IZQUIERDO	1			falta soporte		
3 a 4	017407D0110	ESTRUCTURA CASTILLO SUSPENSION L Y SX	2			sin agujero		
5 a 9	017407D0110	ESTRUCTURA CASTILLO SUSPENSION L Y SX	5			sin agujero		
10	321482-0100	PLACA PROTECCION DENTADO ESTRUCT. SW	1			desoldada		
11	321482-0100	PLACA PROTECCION DENTADO ESTRUCT. SW	1			desoldada		
12	118458-0102	REPOSABRAZOS DERECHO	1			deformada		
13	017407D0110	ESTRUCTURA CASTILLO SUSPENSION L Y SX	1			pasadas		
14 a 15	017407D0110	ESTRUCTURA CASTILLO SUSPENSION L Y SX	2			deformada		
16	015769D0210	ESTRUCTURA RESPALDO MONO REP+LUMB S	1			rosca mal		

*Figura 4.6 Hoja de defectos por proveedor*

Al hacerlo así, los operarios irán rellendo poco a poco las listas de material defectuoso en lugar de hacerlo el encargado todo a la vez al final del turno, para que, cuando esta esté completa, solamente tenga que pulsar un botón para imprimir la lista de materiales separados por proveedor y así después ubicarla directamente en el contenedor a enviar. Dentro del menú de cada proveedor, además contarán con la opción de eliminar todos los datos escritos una vez se haya impreso, pero al hacerlo, saltará un mensaje que le preguntará al encargado si realmente quiere borrar los datos antes de hacérselos

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

llegar al técnico de calidad. Además de esto, se incluirá un seleccionable desplegable que mostrará todos los tipos de defectos posibles codificados, para que el encargado pueda localizarlos con mayor rapidez se incluye la descripción junto a cada defecto, como podemos ver en la **Figura 4.7**.



*Figura 4.7* Seleccionable del tipo de defecto

Asimismo, crearemos otro desplegable que ayude a la persona que introduce los datos a escribir la ubicación o foco en la que se produce el defecto. Para ello, se numeran los posibles componentes de un asiento del 100 al 999, seccionando para diferenciarlos por conjuntos de componentes, como podemos observar en la **Figura 4.8**.



*Figura 4.8* Seleccionable por ubicación

Hilado con esto, también se crea otro botón con macros, que lo que hará será recoger todos los datos escritos en la lista para después exportarlos al Excel con el que trabaja calidad, el cual podemos observar en la **Figura 4.9**. Todo esto se hará automáticamente, además de

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

que, para mayor seguridad, el Excel de calidad se cerrará para así lograr evitar que los encargados puedan tocar algo que no deban y modificar los datos que el departamento de calidad luego ha de explotar. Al hacer esto, lograremos que en el Excel de calidad aparezcan las referencias, el número de piezas y los defectos identificados por parte de fabricación, con la peculiaridad de que estos se traducirán de forma automática al inglés, reduciendo lo máximo posible el trabajo que ha de llevar a cabo posteriormente el departamento de calidad, en el cual estará comprendido asignar un número a la reclamación, incluir la fecha en la que se realiza el envío y el albarán que las incluirá, y descontar las piezas en SAP. Haciéndolo así, eliminaremos la posibilidad de que suceda un error humano, ya que cada referencia irá ligada a su defecto y al número de piezas en el que se ha identificado.

11	46	50	01663SD01/04	Rosca pasada	Passed thread					1	10/11/2020	x			Q23144
11	46	51	01663SD01/04	Falta soporte cincha	Lack of webbing support					1	11/11/2020	x			Q23145
11	46	52	01663SD01/04	Falta soporte cincha	Lack of webbing support					1	12/11/2020	x			Q23146
11	46	53	01663SD01/04	Rosca pasada	Passed thread					5	13/11/2020	x			Q23147
11	46	54	01663SD01/04	Falta pieza	Missing part					1	14/11/2020	x			Q23148
11	46	55	01663SD01/04	Rosca pasada	Passed thread					2	15/11/2020	x			Q23149
5	21	56	01663SD01/04	Falta soporte cincha	Lack of webbing support					3	27/05/2021				Q23150
5	21	57	01663SD01/04	Rosca pasada	Passed thread					5	27/05/2021				Q23151
5	21	58	01663SD01/04	Falta pieza	Missing part					1	27/05/2021				Q23152
5	21	59	01663SD01/04	Rosca pasada	Passed thread					2	27/05/2021				Q23153
5	21	60	01663SD01/04	Falta pieza	Missing part					1	27/05/2021				Q23154

**Figura 4.9** Hoja de calidad a la que se exporta

Otro problema que se pretende solucionar con esto es el tiempo de espera de los camioneros para ser cargados, ya que estos, muchas veces, vienen a la empresa con la intención de cargar embalajes vacíos (two ways packaging), y estos viajes de vuelta al proveedor se aprovechan para enviar todo el material defectuoso. Pero, ¿qué sucede a veces?, que los contenedores de devoluciones no están preparados ni identificados, obligando a los trabajadores, tanto de fabricación, como de calidad o logística, a desviarse de sus tareas cotidianas para lograr identificar el material contenido en cada contenedor para poder identificarlo y así poder flejar el contenedor y enviarlo. Por estos problemas de comunicación se hace acuciante un sistema que logre automatizar todo este proceso y que



## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

obligue a los encargados a imprimir el listado de materiales de un cajón antes de poder empezar con el siguiente.

A mayores de esta búsqueda de piezas habilitada para los operarios, también se crea un menú principal para usuarios más avanzados, que tengan un conocimiento más amplio de las piezas y sepan cuál es la referencia de cada una de ellas. Usando este menú, podrán incluir en un listado la referencia de la pieza a identificar. Haciendo esto, se incluirá en la propia lista de forma automática tanto el nombre de la pieza como el proveedor al que pertenece, para que a continuación sólo tengan que incluir el número de piezas y el defecto identificado, que también se elegirá de un desplegable que variará en función de la familia de piezas a la que la seleccionada pertenezca. En este menú, representado en la **Figura 4.10**, se podrán incluir varias líneas, facilitando la tarea de introducción de datos, para que después, al pulsar exportar, un bucle vaya recorriendo cada una de ellas y lo almacene en el listado del proveedor referenciado por cada pieza. En esta hoja también se bloquearán todas aquellas celdas no modificables para evitar posibles pérdidas de información al tocar alguna de las fórmulas.

El único problema que tiene este sistema de introducción de datos, consiste en que la persona que los introduce ha de estar muy familiarizada con las piezas, ya que habrá de saberse las referencias de las mismas para poder llevar a cabo un registro eficaz. Por esto mismo, este sistema estará más enfocado a encargados o gente perteneciente al departamento de logística y no se recomienda su utilización a operarios que simplemente conozcan el nombre de la pieza, ya que es probable que perdieran una gran cantidad de tiempo para localizar la referencia de cada pieza en cuestión. En ese caso se recomienda utilizar el sistema de introducción consistente en movimiento entre pantallas mediante el uso de macros hasta llegar al material que se quiere introducir, como se describió anteriormente y representado en la **Figura 4.11**.



## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

Todo nuestro sistema funciona mediante una búsqueda de datos referenciada a una única hoja oculta, en la que se encuentran todas las referencias, nombres de piezas, proveedores y posibles defectos para cada familia de piezas. De esta forma, si en algún momento surge la aparición de un nuevo material, únicamente habríamos de incluir el mismo en nuestra lista, incluyendo a su vez la referencia y el proveedor, para que nuestro sistema se actualizara automáticamente y las macros pudieran acceder a los nuevos campos ya rellenados. Pero claro, si queremos incluir un nuevo botón para el sistema creado para los operarios, tendremos que recurrir a su programación. Para hacer esto de la forma más sencilla posible, se crea una función en la que solamente habrá que cambiar el nombre del botón, haciendo referencia en la hoja oculta al que se le ha asignado a la nueva pieza.

De esta forma, si en alguna de nuestras hojas queremos añadir un nuevo botón, solo deberemos copiar el código de cualquiera de los ya existentes y enlazarlo como se ve en la **Figura 4.12** al botón que le hayamos asignado en nuestra hoja oculta.

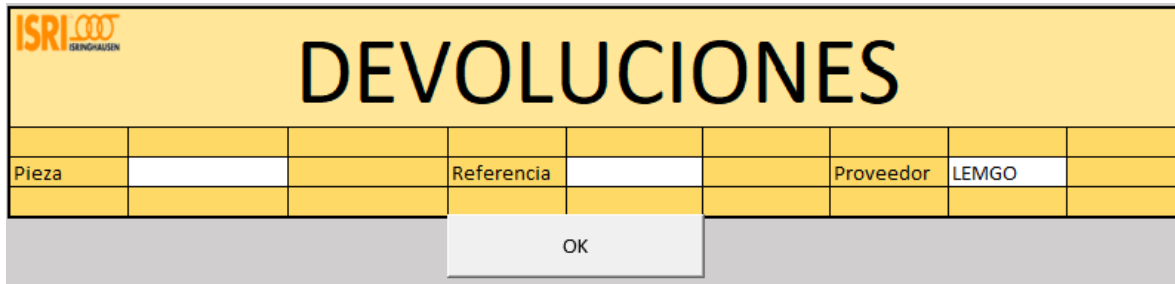
CommandButton22	947854/00	funda nero/blu 1P cus	AUNDE
CommandButton23	947864/00	Funda tabla bipo. Textil blu MY2016	AUNDE
CommandButton24	016646-01/03	REPOSACABEZAS	GRAMMER
CommandButton25	016635D01/04	SISTEMA DE SUSPENSIÓN SX	LEMGO
CommandButton26	016636D01/04	SISTEMA DE SUSPENSIÓN DX	LEMGO
CommandButton27	013571D01/05	ESTRUCTURA INTERIOR TABLA BIPLAZA BIS	NIBBIA cajón
CommandButton28	013601D01/00	BASE BANQUETA BIPLAZA SX	NIBBIA cajón

**Figura 4.12** Tabla de asignación de botones a las referencias

Asimismo, en el mismo lugar en el que se encuentra el menú de introducción de datos avanzada, se crea otro menú más sencillo para nuevos usuarios, en el que se podrá elegir de un desplegable tanto el nombre de la pieza, como la referencia, como el proveedor, logrando así, que cuando le demos al botón de aceptar, el Excel nos desplace a la hoja del proveedor que realiza la pieza identificada para buscar en el menú existente dentro de esta hoja el producto que queramos incluir para después rellenarlo manualmente. Sin embargo,

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

si se incluye una referencia en la hoja de un proveedor que no pertenece a este, saltará un mensaje de error y no te dejará incluirlo. Lo vemos en la **Figura 4.13**.



ISRI 000 ISRINGHAUSEN							
DEVOLUCIONES							
Pieza			Referencia			Proveedor	LEMGO

OK

**Figura 4.13** Cuadro de búsqueda sencillo

De esta forma, será el programa el que se adapte a las competencias de cada usuario en concreto, en lugar de ser el usuario el que tenga que adaptarse a un programa fijo, pudiendo elegir cuál de todas las opciones de búsqueda le es más útil.

### 4.3- Código de programación

A continuación, repasaremos la estructura de nuestro código en sus diferentes variantes.

#### **·Macro creada para exportar las diferentes líneas insertadas en la pestaña de “INICIO”:**

El primer paso para exportar las líneas insertadas en la pantalla de la hoja “INICIO”, será comprobar si en las hojas de los distintos proveedores hay hueco para incluir todas las líneas que hayamos añadido. En caso de no haberlo, la macro nos devolverá un mensaje de error y nos dirá que el lote del proveedor en cuestión está completo, y si queremos continuar, deberemos ir a la hoja del mismo para exportar los datos al proveedor e imprimirlo antes de poder empezar con el siguiente cajón. De este modo, si no hubiera hueco para las referencias introducidas para cualquiera de los proveedores, la macro no exportaría ninguna de ellas. Por el contrario, en caso de que sí haya hueco para todas las referencias, nuestra macro procederá a hacer una llamada que hará que se ejecute esta vez sí la macro

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

de exportación. Podemos ver la sintaxis de la macro de comprobación en la **Figura 4.14**, y a continuación pasaremos a explicarla detenidamente.

```
Sub CompruebaInicio()  
' CompruebaInicio Macro  
Dim i As Double  
Dim cantidad As String  
Dim prov As String  
Dim lote As String  
Dim hueco As String  
Dim response  
  
Set MYRANGE = Worksheets("Hoja2").Range("i3:l28")  
  
With Sheets(1)  
fin = Application.CountA(.Range("e:e")) - 2 + 13  
i = 13  
  
Sheets("Hoja2").Visible = True  
Sheets("Hoja2").Select  
Range("L3:L28").Select  
Selection.ClearContents  
Sheets("INICIO").Select  
  
Do While i <> fin  
  
prov = Cells(i, 7).Value  
cantidad = Cells(i, 5).Value  
Sheets("Hoja2").Visible = True  
Sheets("Hoja2").Select  
lote = Application.WorksheetFunction.VLookup(prov, MYRANGE, 2, False)  
  
If Application.WorksheetFunction.VLookup(prov, MYRANGE, 4, False) <> "" Then  
hueco = Application.WorksheetFunction.VLookup(prov, MYRANGE, 4, False) - cantidad  
Cells(Application.WorksheetFunction.VLookup(prov, MYRANGE, 3, False), 12) = hueco  
Else  
Sheets(prov).Visible = True  
Sheets(prov).Select  
hueco = lote - Range("E2").Value - cantidad  
Sheets("Hoja2").Select  
Cells(Application.WorksheetFunction.VLookup(prov, MYRANGE, 3, False), 12) = hueco  
End If  
Sheets("Hoja2").Visible = False  
Sheets("INICIO").Select  
If hueco < 0 Then  
Sheets(prov).Select  
response = MsgBox("Se ha excedido el tamaño de lote de " & prov & ", con capacidad  
Exit Sub  
Else  
i = i + 1  
End If  
Sheets(prov).Visible = False  
Loop  
End With  
Call exportInicio  
End Sub
```

**Figura 4.14** Macro de comprobación inicial

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

Como se puede observar, después de iniciar la macro, el primer caso será definir y declarar todas las variables que vayamos a utilizar. Para ello, utilizaremos la variable “i” como contador de nuestro bucle, la variable “cantidad”, que irá almacenando por línea las cantidades de las distintas referencias a introducir, la variable “prov”, que a su vez almacenará el proveedor de cada una de las líneas, la variable “lote”, que almacenará el tamaño de los contenedores para cada uno de las líneas, distinguiendo según cuál sea su proveedor, la variable “hueco”, que utilizaremos para comprobar si realmente las piezas introducidas van a caber en los contenedores ya existentes, y la variable “response”, la cual, en caso de no caber todas y cada una de las piezas introducidas, nos devolverá un mensaje para advertirnos el proveedor en el que se produce, el tamaño de lote del mismo, y el número de piezas en que lo hemos excedido.

El siguiente paso, será identificar el rango de la “Hoja2” en el que se van a introducir los tamaños de lote de los distintos proveedores. Como se puede observar en la **Figura 4.15**, especificada por el rango establecido “MYRANGE”, además del tamaño de lote incluiremos dos columnas más, de las cuales una nos servirá para lograr que la macro se posicione correctamente en la columna de hueco en función del proveedor al que hagamos la llamada, y la casilla de hueco se actualizará cada vez que ejecutemos la macro. En un primer momento esto no se llegó a pensar que hiciera falta, pero pensándolo mejor, si alguna vez se pretendieran añadir varias líneas del mismo proveedor, se necesitaría que la comprobación del hueco se actualizara, ya que si, por ejemplo, en un cajón de 4 huecos, introducimos dos líneas de una unidad, el programa nos lo iba a permitir, porque en la primera comprobación vería que hay un hueco disponible, y si no actualizamos este valor la segunda vez que introdujésemos una unidad también nos lo permitiría.

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

<u>Proveedor</u>	<u>Tamaño de lote</u>	<u>2</u>	<u>Hueco</u>
AUNDE	70	3	
GRAMMER	96	4	
LEMGO	20	5	
NIBBIA cajón	25	6	
NIBBIA biplaza	4	7	-1
TOSCANA GOMMA	30	8	

*Figura 4.15* Tabla de tamaños de lote y huecos disponibles

A continuación, se establecerá la hoja de trabajo, y el número de repeticiones que se llevarán a cabo en función del número de líneas que queramos añadir, utilizando la variable “i”. A su vez, también se restablecerá la tabla descrita a continuación a 0 en cuanto el número de huecos por contenedor. El próximo paso será iniciar nuestro bucle.

En nuestro bucle, haremos que la variable “prov” adquiera el valor de la columna 7, y la variable “cantidad” el de la columna 5. También buscaremos el tamaño de lote del contenedor en función de la referencia que se vaya especificando. A partir de aquí, si la columna de hueco de nuestra tabla está vacía se llevará un código que será muy diferente a si tiene algún valor (si ya se ha añadido alguna línea de productos a cada proveedor). Si no tiene ningún valor en dicha columna, el hueco disponible se almacenará como el tamaño de lote menos lo que ya hayamos introducido en el contenedor y restando también la cantidad que queramos introducir en esta nueva línea. Por el contrario, si esta tiene algún valor, el hueco existente se calculará restando la nueva cantidad introducida a lo a existente en la columna de hueco.

A partir de esto, si en el contenedor no hay hueco, la macro nos expulsará de la misma diciéndonos que no hay hueco disponible y nos dejará abierta la pestaña del proveedor para que podamos imprimirla y exportar los datos para después vaciarla y poder al fin introducirlos. En el caso de que sí que hubiera hueco, la macro haría una llamada a otra macro llamada “exportInicio”, representada en la **Figura 4.16** que ya procedería a exportar todos nuestros datos a las hojas de los distintos proveedores.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

```
Sub exportInicio()|
' exportInicio Macro
Dim i As Double
Dim ref As String
Dim cantidad As String
Dim defecto As String
Dim prov As String

With Sheets(1)
fin = Application.CountA(.Range("e:e")) - 2 + 13
i = 13

Do While i <> fin
ref = Cells(i, 3).Value
prov = Cells(i, 7).Value
cantidad = Cells(i, 5).Value
defecto = Cells(i, 6).Value

    Sheets(prov).Visible = True
    Sheets(prov).Select

        Range("c11").Activate
        If Range("c12").Value <> "" Then
            ActiveSheet.Range("c11").End(xlDown).Offset(1, 0).Select
        Else
            ActiveSheet.Range("c12").Select
        End If
        Selection.Value = ref

        Range("E11").Activate
        If Range("E12").Value <> "" Then
            ActiveSheet.Range("E11").End(xlDown).Offset(1, 0).Select
        Else
            ActiveSheet.Range("E12").Select
        End If
        Selection.Value = cantidad

        Range("H11").Activate
        If Range("H12") <> "" Then
            ActiveSheet.Range("H11").End(xlDown).Offset(1, 0).Select
        Else
            ActiveSheet.Range("H12").Select
        End If
        Selection.Value = defecto

        Call exportFechas
        Call exportReclamacion

        Sheets("INICIO").Select
        Sheets(prov).Visible = False
    i = i + 1
Loop
End With
End Sub
```

*Figura 4.16 Macro para exportar valores desde el inicio*

El primer paso de esta macro será iniciarla. Para ello utilizamos el comando Sub en la que la nombraremos, el cuál será “exportInicio”.



## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

Posteriormente, declararemos todas las variables que usaremos durante el desarrollo de la misma, entre las cuales estará “i”, que nos servirá para contar el número de filas existentes rellenas por los encargados y lo utilizaremos en el bucle; “ref”, que almacenará la referencia de cada uno de los productos introducidos; “cantidad”, que almacenará el número de piezas de cada referencia; y por último “prov”, que almacenará el proveedor en cuestión de cada una de las piezas.

El siguiente paso será seleccionar la hoja 1, “INICIO”, y establecer cuál será el fin del bucle, que contará todas las líneas existentes en la columna de cantidades, le restará dos unidades, debido a las celdas E7 y E12, que tienen contenido en su interior, y le sumará 13, puesto que esta será la fila en la que empezaremos a contar. Ligado a esto, estableceremos el valor de la variable “i” en 13 unidades.

Una vez hecho esto, empezaremos con el bucle, compuesto por un “Do While”, que se ejecutará siempre que “i” sea distinto del fin, logrando así que este avance hasta la última fila escrita. Estableceremos que la variable “ref” tiene el valor de la celda (i,6), siendo la i el número de fila y el 6 el número de columna. A su vez, “prov” tomará el valor de la celda (i,7), “cantidad” el de la celda (i,5) y defecto el de la celda (i,6). De esta manera, lograremos que el valor de todas nuestras variables se vaya actualizando a medida que avancemos en el bucle.

Cuando el valor de las variables ya se haya establecido, le diremos al Excel que abra la pestaña del proveedor que hemos guardado anteriormente, ya que esta página estará oculta por defecto, y posteriormente nos situaremos en ella. Una vez situados, nos posicionaremos en la celda C11 y entraremos a un “if” que lo que hará es comprobar si la primera celda a rellenar, C12, está vacía, de tal forma que si lo está se situará en esta misma celda, y si no lo está avanzará por la columna hacia abajo hasta la primera celda vacía de la página, y allí será donde se insertará la “ref” de nuestra primera pieza. A continuación, se

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

llevará el mismo proceso a cabo otras dos veces, tanto para insertar el valor de nuestra “cantidad” como el de nuestro “defecto”.

Una vez exportados nuestros tres valores a la hoja del proveedor correspondiente, se realizará una llamada a otras dos macros que servirán para introducir en la hoja del proveedor tanto el día en el que se han bloqueado cada una de las piezas como el número de reclamación que le corresponde a las mismas, y finalmente la macro nos devolverá automáticamente a la hoja de “INICIO”, y el valor de nuestra “i” se actualizará en una unidad, para así repetir el bucle con la siguiente fila de la hoja “INICIO”. Cuando se terminen de exportar todos los datos, es decir, que el valor de nuestra “i” coincida con el de nuestro “fin”, se saldrá del bucle y habremos terminado nuestro proceso.

### **·Macro para seleccionable por tipo de pieza, referencia y proveedor:**

Esta macro será una composición de un gran número de “if”, como se puede observar en la **Figura 4.17**, que decidirá en función de lo que elijamos en el desplegable, de entre cualquiera de los tres rangos habilitados para su modificación, la pestaña del proveedor encargado de su fabricación y nos desplazará a la hoja del mismo para que podamos introducir los datos.

```
If Range("F7") = "10200743D01/10" Then
Sheets("NIBBIA biplaza").Visible = True
Sheets("NIBBIA biplaza").Select
End If

If Range("I7") = "TOSCANA GOMMA" Then
Sheets("TOSCANA GOMMA").Visible = True
Sheets("TOSCANA GOMMA").Select
End If

If Range("C7") = "Suspensión" Then
Sheets("LEMGO").Visible = True
Sheets("LEMGO").Select
End If
```

*Figura 4.17 Macro para seleccionar pieza, referencia y proveedor*

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

### **·Macro para los desplazamientos entre pestañas siguiendo el sistema de botones:**

Este sistema tiene la peculiaridad de que se crea una pestaña individual según vamos avanzando en la búsqueda de nuestra pieza, por lo que obligatoriamente estará compuesto por una gran cantidad de macros individuales que harán visibles las hojas a las que nos desplazamos e invisibles aquellas de las que venimos, como se puede observar en la **Figura 4.18**.

```
Sub ESPUMAS_CONDUCTOR()  
'  
' ESPUMAS_CONDUCTOR Macro  
'  
'  
  
    ActiveSheet.Visible = False  
    Sheets("ESPUMAS CONDUCTOR").Visible = True  
    Sheets("ESPUMAS CONDUCTOR").Select  
End Sub
```

*Figura 4.18 Macro para desplazamiento entre pestañas*

### **·Macro para finalizar la selección de nuestro sistema de botones:**

Siguiendo la macro anterior, en el momento en que lleguemos al producto deseado mediante nuestro sistema de botones, llegaremos al último botón, al que estará asociado una macro que copiará la referencia de dicho producto y lo establecerá en la hoja del proveedor. Inicialmente, se pensó en llevar a cabo esto mediante la utilización de una macro individual para cada una de las referencias, que sería como se ve en la **Figura 4.19**.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

```
Sub ESPUMA_RESPALDO_BIS_SIN_TAB_EUROPEO()  
,  
' ESPUMA_RESPALDO_BIS_SIN_TAB_EUROPEO Macro  
,  
,  
  
    ActiveSheet.Visible = False  
    Sheets("TOSCANA GOMMA").Visible = True  
    Sheets("TOSCANA GOMMA").Select  
        Range("L18").Select  
        Selection.Copy  
        Range("c11").Activate  
            If Range("c12").Value <> "" Then  
                ActiveSheet.Range("c11").End(xlDown).Offset(1, 0).Select  
            Else  
                ActiveSheet.Range("c12").Select  
            End If  
    Selection.PasteSpecial paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _  
        :=False, Transpose:=False  
  
End Sub
```

*Figura 4.19 Macro inicial de búsqueda por botones*

Sin embargo, aunque esto se llevara a cabo así en un principio, se decidió que, para una mayor facilidad y con el fin de hacer la programación de la forma más intuitiva posible, facilitando la actualización de la hoja de datos ante la introducción de nuevas referencias y proveedores en el registro, lo mejor fuera unificar todas ellas de la siguiente forma.

Siguiendo la macro anterior, en el momento en que lleguemos al producto deseado mediante nuestro sistema de botones, llegaremos al último botón, al que estará asociado una macro que copiará la referencia de dicho producto y lo establecerá en la hoja del proveedor, como se aprecia en la **Figura 4.20**.

---

```
Private Sub CommandButton11_Click()  
    boton = CommandButton11.Name  
    Call abc  
    Sheets("FUNDAS BIPLAZA EUROPEO").Visible = False  
End Sub
```

*Figura 4.20 Macro individual para búsqueda de botones*

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

Para ello, como hemos explicado anteriormente, solo se deberá establecer en cada macro individual el nombre del botón seleccionado, para poder después llevar a cabo la búsqueda en nuestra hoja de datos. Para ello, deberemos establecer la variable “botón” como variable pública, ya que deberemos acceder a ella desde todas las pestañas de nuestro sistema de selección mediante botones. Como vemos en la imagen, nuestra macro a continuación hará una llamada a otra macro, llamada “abc”, que hará lo representado en la **Figura 4.21**.

```
Option Explicit
Public boton As String

Sub abc()
Worksheets("Hoja2").Visible = True
Worksheets("POR DET").Visible = True
Worksheets("POR DET").Activate
    ActiveSheet.Range("E2") = boton
    With Worksheets("POR DET").Range("A2")
        .Formula = "=IF(ISERROR(VLOOKUP(RC[4],Hoja2!R3C4:R70C7,3,FALSE)),\"\",VLOOKUP(RC[4],Hoja2!R3C4:R70C7,3,FALSE))"
    End With
Call BUSCARV
End Sub
```

**Figura 4.21** Macro de búsqueda de proveedor y referencia

Esta macro, hará visibles las pestañas “Hoja 2”, nuestra hoja de datos, y “POR DET”, hoja empleada para llevar a cabo la búsqueda de las distintas referencias y proveedores, y se situará en esta última. Haremos que se imprima en el rango E2 el valor de nuestro botón, el cual será CommandButton y un número que variará en función de la pieza que hayamos seleccionado. A continuación, en la celda A2, nos establecerá el nombre de la pieza que hemos seleccionado, buscando en nuestra hoja de datos mediante la utilización del comando “BUSCARV”, que se situará en el nombre del botón que hayamos pulsado y se desplazará horizontalmente hasta encontrarlo. Una vez hecho esto, nuestra macro realizará una llamada a otra macro llamada BUSCARV, la cual explicaremos en la **Figura 4.22**.

# ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

```
Sub BUSCARV()  
  
    Dim proveedor As String  
    Dim ref As String  
    Dim lote As String  
  
    'Seleccionamos la hoja listado  
    Worksheets("POR DET").Select  
    'Aplicamos la función buscarv para buscar el nombre y si no está que el resultado sea vacío  
    With Worksheets("POR DET").Range("B2")  
  
        .Formula = "=IF(ISERROR(VLOOKUP(RC[-1],Hoja2!R3C6:R70C7,2,FALSE)),\"\",VLOOKUP(RC[-1],Hoja2!R3C6:R70C7,2,FALSE))"  
        .Formula = .Value  
  
        Sheets(.Value).Visible = True  
  
    End With  
  
    With Worksheets("POR DET").Range("C2")  
  
        .Formula = "=IF(ISERROR(VLOOKUP(RC[-2],Hoja2!R3C6:R70C8,3,FALSE)),\"\",VLOOKUP(RC[-2],Hoja2!R3C6:R70C8,3,FALSE))"  
  
    End With  
  
    proveedor = Sheets("POR DET").Range("B2").Value  
    ref = Sheets("POR DET").Range("C2").Value  
  
    Set MYRANGE = Worksheets("Hoja2").Range("i3:j28")  
  
    lote = Application.WorksheetFunction.VLookup(proveedor, MYRANGE, 2, False)|  
    'Worksheets("Hoja2").Visible = False  
    Worksheets("POR DET").Visible = False  
  
    Sheets(proveedor).Select  
  
    If lote - Range("b2").Value > 0 Then  
  
        Range("C11").Activate  
        If Range("C12").Value <> "" Then  
            ActiveSheet.Range("C11").End(xlDown).Offset(1, 0).Select  
        Else  
            ActiveSheet.Range("C12").Select  
        End If  
        Selection.Value = ref  
  
        Call exportFechas  
        Call exportReclamacion  
  
        ActiveSheet.Range("E11").End(xlDown).Offset(1, 0).Select  
  
        Worksheets("Hoja2").Visible = False  
  
        Else: MsgBox "Se ha excedido el tamaño de lote en una unidad. Imprima hoja y repita el proceso.", vbCritical, "Hueco insuficiente": Exit Sub  
  
    End If  
End Sub
```

**Figura 4.22** Macro para buscar los valores desde un botón

Esta macro declarará tres variables, proveedor, ref y lote, que almacenarán respectivamente el proveedor, la referencia y el tamaño de lote del botón seleccionado. Para ello, nos situaremos de nuevo en la pestaña “POR DET”, en la que estableceremos nuestro cursor en la celda B2 para buscar el nombre de nuestro botón en la hoja de datos y almacenaremos el nombre del proveedor en nuestra primera variable, para poder después abrir la pestaña del mismo. A continuación, almacenaremos también el valor de nuestra referencia en la otra variable creada. La variable lote almacenará el tamaño de lote del

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

proveedor correspondiente para poder comprobar si cabe la referencia que queramos introducir, y cuando no sea así, nos mandará un mensaje de advertencia por pantalla.

Una vez hecho esto, en la hoja de nuestro proveedor nos situaremos en la primera celda no escrita y añadiremos el valor de la referencia seleccionada, para que automáticamente nos ponga el nombre de la pieza y únicamente tengamos que añadir la cantidad y el defecto identificado en la misma. A su vez, se añadirán también tanto la fecha correspondiente como el número de reclamación al que se hará referencia mediante la llamada de las macros ExportFechas y ExportReclamacion.

### **·Macros destinadas a explotar la lista de devolución una vez ya completa:**

Cuando la hoja del proveedor ya haya sido completada y podamos cerrar el cajón, se crea una primera macro para imprimir la lista, con la estructura representada en la **Figura 4.23**.

```
Sub Imprimir()  
'  
' Imprimir Macro  
'  
'  
  
    ActiveWindow.SelectedSheets.PrintOut Copies:=1, Collate:=True, _  
        IgnorePrintAreas:=False  
End Sub
```

*Figura 4.23 Macro para imprimir*

Esta macro seleccionará el área de impresión y realizará la misma de tal forma que se adapte a la hoja completa.

A continuación, se expondrá una macro que exporta los datos de las hojas al Excel de calidad, de la que ya se ha hablado anteriormente, que inicialmente tendría la estructura representada en la **Figura 4.24**.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

```
Sub exportNibbiaBip()  
,  
' exportNibbiaBip Macro  
,  
,  
  
    Range("C12").Select  
        If Range("C13").Value <> "" Then  
            Range(Selection, Selection.End(xlDown)).Select  
        End If  
    Selection.Copy  
  
Workbooks.Open "C:\Users\Usuario\Desktop\R 1341-05 2021 Gestión material no conforme.xlsm"  
Sheets("NIBBIA").Select  
  
    ActiveSheet.Range("D1").End(xlDown).Offset(1, 0).Select  
    ActiveSheet.paste  
  
    Windows("Lista material bloqueado encargados.xlsm").Activate  
    Range("E12").Select  
        If Range("E13").Value <> "" Then  
            Range(Selection, Selection.End(xlDown)).Select  
        End If  
    Selection.Copy  
    Application.WindowState = xlNormal  
    Windows("R 1341-05 2021 Gestión material no conforme.xlsm").Activate  
    ActiveSheet.Range("I1").End(xlDown).Offset(1, 0).Select  
    ActiveSheet.paste  
  
    Windows("Lista material bloqueado encargados.xlsm").Activate  
    Range("F12").Select  
        If Range("F13").Value <> "" Then  
            Range(Selection, Selection.End(xlDown)).Select  
        End If  
    Selection.Copy  
  
    Windows("R 1341-05 2021 Gestión material no conforme.xlsm").Activate  
    ActiveSheet.Range("G1").End(xlDown).Offset(1, 0).Select  
  
    Selection.PasteSpecial paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _  
        :=False, Transpose:=False  
  
    ActiveWorkbook.Save  
    ActiveWindow.Close  
End Sub
```

**Figura 4.24** Macro inicial para exportar datos a calidad

Logrando copiar todos los valores introducidos para después situarse en la primera fila no escrita de los valores de referencia, cantidad y defecto del proveedor en cuestión del libro de calidad, y los imprima en sus líneas correspondientes, guardando y cerrando dicho libro al finalizar. Como se pudo observar que los Excel de destino de calidad no estaban unificados, se crearon varias macros con un formato parecido estableciendo una por proveedor. Al ver que esto no le daba al programa la flexibilidad buscada a la hora de incluir



## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

nuevos proveedores o hacer una modificación en todos ellos, se decidió llevar a cabo una macro que unificara a todos los proveedores, con el formato representado en la **Figura 4.25**.

```
Dim fila As Integer
Range("C12").Select
    If Range("C13").Value <> "" Then
        Range(Selection, Selection.End(xlDown)).Select
    End If
Selection.Copy
abrirarchivoexterno
fila = Range("AA1").Value
Cells(fila, 4).Select
Selection.PasteSpecial paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
abrirarchivodevoluciones
Range("E12").Select
    If Range("E13").Value <> "" Then
        Range(Selection, Selection.End(xlDown)).Select
    End If
Selection.Copy
abrirarchivoexterno
Cells(fila, 10).Select
Selection.PasteSpecial paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
    abrirarchivodevoluciones
Range("H12").Select
    If Range("H13").Value <> "" Then
        Range(Selection, Selection.End(xlDown)).Select
    End If
Selection.Copy
abrirarchivoexterno
Cells(fila, 5).Select
Selection.PasteSpecial paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
    abrirarchivodevoluciones
Range("I12").Select
    If Range("I13").Value <> "" Then
        Range(Selection, Selection.End(xlDown)).Select
    End If
Selection.Copy
abrirarchivoexterno
Cells(fila, 11).Select
Selection.PasteSpecial paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
    abrirarchivodevoluciones
Range("J12").Select
    If Range("J13").Value <> "" Then
        Range(Selection, Selection.End(xlDown)).Select
    End If
Selection.Copy
abrirarchivoexterno
Cells(fila, 14).Select
Selection.PasteSpecial paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
ActiveWorkbook.Save
ActiveWindow.Close
    abrirarchivodevoluciones
Range("C12").Select
End Sub
```

*Figura 4.25 Macro para exportar los datos a calidad*

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

De esta forma, crearemos una variable nueva denominada “fila”, que almacenará la fila por la que nos lleguemos en el Excel de calidad en cada caso y así lograremos que, si por error alguna vez no se introduce alguno de los campos a rellenar, no nos produzca que al exportar los datos se nos copie en filas diferentes, para así tener todos los campos de cada referencia unidos en una misma línea. Eliminamos el posible error humano. Además, también incluiremos a mayores la fecha en el cual el material ha sido bloqueado y el número de reclamación correspondiente al mismo.

Otro de los comandos existentes en nuestras hojas de devolución, será el destinado a eliminar los datos introducidos para poder seguir introduciendo nuevos datos a continuación, ya que la cantidad de piezas que cabe en un cajón de un determinado proveedor está limitada según el tamaño de lote de dicho proveedor. De esta forma, entendemos que cada vez que se cierra un cajón se borrarán los campos en la hoja de cada proveedor para poder empezar con un nuevo cajón. Unido con esto, se decide que el cambio de reclamación se llevará a cabo cuando se cierre un cajón para abrir otro.

El sistema de nomenclatura de las reclamaciones estará compuesto por una Q, el número del proveedor y un número del 001 al 999 que irá avanzando por unidades, con una pequeña excepción. El de Nibbia biplaza ha de diferenciarse con el de Nibbia cajón, por lo que al primero se le asignarán los números pares de dicho proveedor y al segundo los impares. De esta forma, cada vez que cerremos un cajón, este indicador deberá avanzar de dos en dos. Por esto mismo realizamos la declaración de la variable reclamación y la sintaxis escrita al final de la macro, que diferenciará los proveedores y hará que las reclamaciones avancen de uno en uno en todos los casos excepto si el proveedor en cuestión es Nibbia, como ya hemos mencionado. La macro tendrá la estructura representada en la **Figura 4.26**.

## CAPÍTULO 4: PROGRAMA PARA GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

```
Sub AceptarCancelar()  
Dim pregunta As Integer  
Dim reclamacion As String  
Dim prov As String  
  
reclamacion = Range("m3").Value  
  
pregunta = MsgBox("ATENCIÓN, ¿Borrar datos antes de enviárselos al técnico de calidad?", vbOKCancel, "EXCELeINFO")  
  
If pregunta = 2 Then  
    MsgBox "Elegiste Cancelar": Exit Sub  
Else  
    Range("C12").Select  
    If Range("C13").Value <> "" Then  
        Range(Selection, Selection.End(xlDown)).Select  
    End If  
    Selection.ClearContents  
    Range("E12").Select  
    Range("E12").Activate  
    Range(Selection, Selection.End(xlToRight)).Select  
    If Range("E13").Value <> "" Then  
        Range(Selection, Selection.End(xlDown)).Select  
    End If  
    'Range("C12:C31,E12:F31").Select  
    'Range("E12").Activate  
    Selection.ClearContents  
  
    prov = Range("d9").Value  
  
    If prov = "NIBBIA" Then  
        Range("M3").Value = reclamacion + 2  
    Else  
        Range("M3").Value = reclamacion + 1  
    End If  
End If  
  
Range("C12").Select  
  
End Sub
```

*Figura 4.26 Macro para eliminar los valores editables*

De tal forma que lo primero que nos saldrá al pulsar ese botón será un recuadro de advertencia que nos preguntará si realmente queremos eliminar todos los datos introducidos. Si respondemos que sí, nos reseteará el cajón, mientras que, si pulsamos que no, no lo hará.

Por último, dispondremos de la macro “Volver”, que nos hará invisible la pestaña del proveedor y nos retornará a la pantalla de “ACCESO DIRECTO” mediante el uso de botones, que tendrá la estructura descrita en la **Figura 4.27**:

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

```
Sub VOLVER()  
,  
,  
,  
,  
,  
    ActiveSheet.Visible = False  
    Sheets("ACCESO DIRECTO").Select  
End Sub
```

*Figura 4.27* Macro para volver a la pestaña "ACCESO DIRECTO"

Además, dispondremos de otro botón que llevará a cabo el proceso de impresión, exportado y borrado completo, pero esta vez sin preguntar al usuario si realmente desea borrar los datos antes de enviarlos al técnico de calidad, ya que se da por supuesto que se llevará a cabo automáticamente y que está representado en la **Figura 4.28**.

```
Sub completo()  
  
Dim reclamacion As String  
Dim prov As String  
  
reclamacion = Range("M3").Value  
  
Imprimir  
export  
Range("C12").Select  
    If Range("C13").Value <> "" Then  
        Range(Selection, Selection.End(xlDown)).Select  
    End If  
    Selection.ClearContents  
    Range("E12").Select  
    Range("E12").Activate  
    Range(Selection, Selection.End(xlToRight)).Select  
    If Range("E13").Value <> "" Then  
        Range(Selection, Selection.End(xlDown)).Select  
    End If  
    'Range("C12:C31,E12:F31").Select  
    'Range("E12").Activate  
    Selection.ClearContents  
  
prov = Range("d9").Value  
  
If prov = "NIBBIA" Then  
    Range("M3").Value = reclamacion + 2  
Else  
    Range("M3").Value = reclamacion + 1  
End If  
  
Range("C12").Select  
End Sub
```

*Figura 4.28* Macro para la realización del proceso completo

## CAPÍTULO 5: ESTUDIO ECONÓMICO

### 5.1- Factibilidad del programa para la gestión de devoluciones en Isringhausen

A la hora de hacer el estudio económico, deberemos tener en cuenta varios factores, entre los que podemos destacar los siguientes:

·Tiempo ahorrado por el no retrabajo por parte de los encargados, ya que los operarios podrán identificar y registrar los defectos de manera autónoma. Si tenemos cuenta que entre los dos encargados consumen un total de cuatro horas y media semanales en identificar y registrar todas las piezas defectuosas que se producen durante una semana y, según el convenio del metal, el coste salarial de un encargado es de 27.555,55€ más un plus de 34,26€ mensuales por turnicidad, obtendríamos un coste total anual de 27.966,67€, y si dividimos esto entre las 744 horas que se trabaja al año, obtendríamos un coste horario de 37,59€. Por lo tanto, con la implantación de este proyecto se obtendrá un ahorro semanal de 169,15€ y mensual de 676,61€, lo cual nos supondría un ahorro total anual en coste salarial de 2.860,23€. A mayores, si se produce una disminución de la necesidad de comunicación entre los propios encargados tanto con el personal del departamento de calidad, ya que se exportarán los datos separados por contenedor de la manera que corresponda y esto se llevará a cabo cada vez que se introduzca una pieza, ahorrando la necesidad de que calidad tenga que interesarse por las mismas, y con el encargado de logística, para decirle qué piezas son, puesto que nada más se termine un contenedor, éste será identificado y desplazado a la zona de piezas bloqueadas para su posterior envío, tendremos un ahorro semanal de unos 40 minutos entre los dos turnos, suponiendo un total de 423,737€ anuales, y si sumamos este valor al anterior obtendremos un total de 3.283,967€ anuales y 234,57€ mensuales.

·Tiempo ahorrado por el departamento de calidad al eliminar la necesidad de comunicación con el personal de fábrica y a la hora de volcar todos los datos en el Excel de productos defectuosos. Si extraemos del convenio del metal el coste salarial que supone un ingeniero

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

responsable de calidad, podremos ver que esto supone 2.744,16€ al mes y 38.418,27€ anuales, y que realiza estas tareas de forma semanal, invirtiendo un total 60 minutos en llevar a cabo el volcado, incluyendo la comunicación previa con fabricación, y conseguimos reducir este tiempo empleado a 15 minutos, nos supondría un ahorro semanal de 45 minutos durante los cuáles lo único que tendrá que hacer será llevar a cabo el registro en SAP, lo cual, en términos monetarios, se traduciría en un total de 654,86€ anuales. A mayores, si incluimos el tiempo invertido por el mismo departamento cada viernes y último día de cada mes, para identificar las ppm de piezas defectuosas por proveedor, estimando el tiempo por cada vez en unos 12,5 minutos, tendríamos un total de 60 minutos al mes, suponiendo un ahorro en coste de 51,64€. Si calculamos el coste anual nos saldría un total de 722,92€, y si sumamos ambos valores 1.377,78€. Por lo tanto, el coste mensual sería un total de 98,41€.

·Si por el contrario nos enfocamos en el departamento de logística, miraremos por un lado el tiempo ahorrado por la mano de obra no directa, y por otro los carretilleros y camioneros afectados. Si cada vez que se realiza una carga, el encargado de logística pierde un total de 5 minutos para comunicarse con fabricación y otros 10 en coordinar todas las tareas necesarias para llevar a cabo el envío y registrar el albarán, esto nos supondría un total de 15 minutos por carga, pero si además ha de comunicarse con calidad y esperar a que este departamento le asigne una reclamación al envío para poder preparar tanto el albarán como el CMR, le sumaremos otros 5 minutos. El total de esto nos saldría unos 20 minutos por carga y si se realizan unas 3 cargas semanales, estableceremos la pérdida de tiempo en alrededor de una hora semanal. Según el convenio del metal, el coste salarial de un ingeniero responsable del departamento de logística es de 2.744,16€ mensuales y 38.418,27€ al año. Por tanto, si llevamos a cabo los cálculos correspondientes, obtendríamos un ahorro anual en coste salarial de 960,46€ y mensual de 68,60€. Si pensamos en los carretilleros, observaremos que tendrán los contenedores identificados

## CAPÍTULO 5: ESTUDIO ECONÓMICO

en el momento en que se cierran y preparados para llevar a la zona de bloqueo. Por lo tanto, esto nos ayudará a reducir la comunicación innecesaria entre fábrica y los carretilleros, además de a reducir o eliminar los movimientos innecesarios de piezas. Según el convenio del metal, un carretillero oficial de segunda cuesta 24.610,99€ anuales y 1.757,93€ mensuales, y pierde unos 10 minutos semanales en comunicarse con fábrica para saber cuándo puede mover los contenedores y otros 5 cuando lo hace por error, tendremos un total de 15 minutos semanales, y si lo multiplicamos por los dos turnos en que se fabrica, 30. Por lo tanto, esto nos supondrá un ahorro económico mensual de unos 19.98€, y anual de 279,67€.

Por lo tanto, si sumamos el coste total ahorrado de manera mensual, obtendremos un total de unos 401,58€, y si consideramos el coste anual, como se puede observar en la **Tabla 5.1**, el coste total ahorrado será de 5.901,88€.

Puesto	Coste mensual (€)	Coste anual (€)
Encargados	234,57€	3.283,967€
Encargado de calidad	98,41€	1.377,78€
Encargado de logística	68,60€	960,46€
Carretilleros	19,98€	279,67€
Total	401,58€	5.901,88€

*Tabla 5.1 Coste anual total ahorrado*

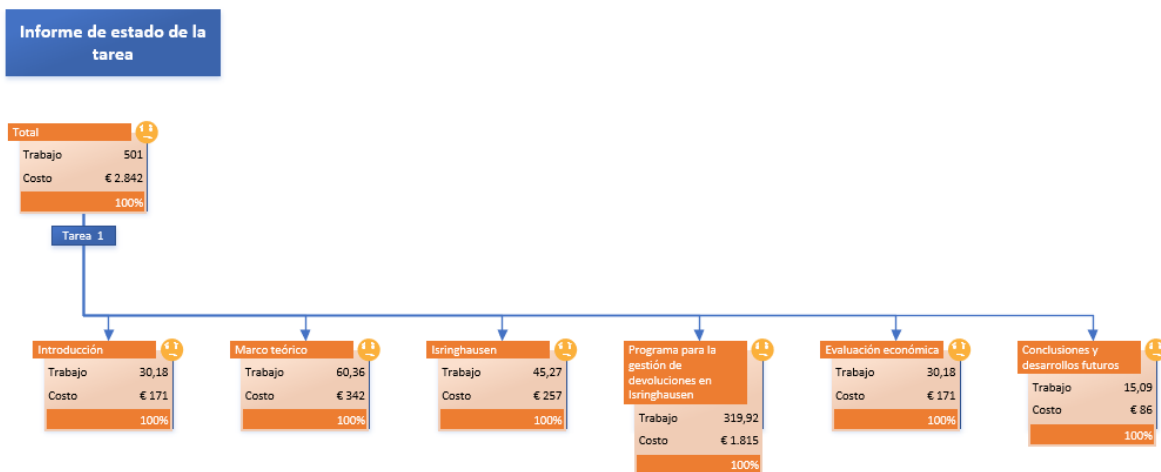
Si por el contrario consideramos el coste que ha supuesto llevar a cabo este proyecto, pensaremos en las horas de dedicación por parte de un ingeniero, las cuales han supuesto un total de 106, y como ya hemos dicho anteriormente, el coste salarial de un ingeniero supone 38.418,27€ anuales, obtendremos un coste total de realización del proyecto de 5.488,32€.

# ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

Observando el valor de algunos indicadores como el **payback**, que será **0,93**, se podrá comprobar que se trata de un proyecto muy viable en el cual se recuperará la inversión realizada en un tiempo inferior a un año, concretamente en 11,16 meses, por lo que este proyecto de estandarización estaría en la pole de cualquier empresa para lograr reducir los costes salariales y ahorrar una gran cantidad de tiempo para que sus empleados puedan dedicarse plenamente a sus quehaceres diarios sin la necesidad de malgastarlo en una tarea que, como ya hemos visto, puede estandarizarse y automatizarse.

## 5.2- Desglose de costes de realización del proyecto de investigación

El proyecto tuvo una duración de 166 horas, comenzando el día 22 de marzo de 2021 y terminando el 14 de julio de 2021. Se trabajó cinco días a la semana (los días laborables de lunes a viernes) durante dos horas al día de 18:00 a 20:00. Se trabajó según la siguiente estructura (**Figura 5.1**):



*Figura 5.1 Informe de costes por tareas*



## CAPÍTULO 5: ESTUDIO ECONÓMICO

- Introducción: 10 horas
- Marco teórico: 20 horas
- Isringhausen: 15 horas
- Programa para la gestión de devoluciones en Isringhausen: 106 horas
- Evaluación económica: 10 horas
- Conclusiones y desarrollos futuros: 5 horas

Para todo esto, se definen los siguientes recursos, algunos considerados como trabajos y otros como materiales:

- Ingeniero: Trabajo, 16€/h
- Microsoft Office: Trabajo, 0'01€/h
- Ordenador: Trabajo, 0'02€/h
- Electricidad: Material, 60'17€
- Agua: Material, 40'26€
- Internet: Material, 80'50€

Para entrar más en detalle, a continuación, se explicará de dónde salen cada uno de estos costes. El coste salarial de un ingeniero es de 5.488,32€, por lo tanto, realizando el prorrateo, obtendremos 16€/h. Se consideró que en un año se trabajan 300 días durante ocho horas al día. El resto de días son considerados vacaciones y festivos. Para la amortización lineal del software informático se considera un coste de 579€ y una duración del mismo de cinco años, por lo que el coste del mismo será de 1'66€. Por otro lado, para la amortización lineal del ordenador se considera que tiene un coste inicial de 800€ y una vida útil de 5 años, por lo que su depreciación durante la duración del proyecto será de un total de 3'32€. Para finalizar, se establecerá que los costes fijos de electricidad, agua e internet durante el período de realización del proyecto serán de 60'17, 40'26 y 80'5€, respectivamente.



## CAPÍTULO 6: CONCLUSIONES Y DESARROLLOS FUTUROS

Al observar el ahorro económico anual, se puede deducir fácilmente la viabilidad de realización de este proyecto, ya que se trata de una aplicación muy flexible que no tiene caducidad y permite una fácil actualización ante la necesidad de una introducción de nuevos productos, tanto de los proveedores actuales, como de otros nuevos.

Además, se observa un nada desdeñable ahorro de tiempo, sobre todo por parte de los encargados de fabricación, lo cual les permitirá emplearlo en otras labores, como una supervisión más concienzuda de sus trabajadores o poderles impartir una mayor formación que permita a la planta llegar al fin de cero defectos.

Los empleados empiezan a aprenderse las piezas por referencias y proveedores, lo cual les permite comunicarse entre ellos y con el personal de logística de una forma más rápida y efectiva, a la vez que aprenden a utilizar los campos de introducción de datos más complejos del propio Excel.

El personal de logística tiene más tiempo para preparar los albaranes y Cartas de Porte o CMRs (Convention relative au contrat de transport international de Marchandise par Route) sin necesidad de investigar por cuenta propia y terminar de preparar los envíos, disminuyendo el número de errores cometidos en el registro de las mercancías de los camiones. A mayores, aprenden a comunicarse con los carretilleros de manera mucho más eficaz, al haber nombrado cada contenedor con una reclamación. Ahora, en el momento de cargar las piezas, solo tienen que decirles a los carretilleros los números de reclamación de los contenedores a enviar, y estos pueden identificarlos de manera mucho más eficaz e intuitiva, ya que siempre tendrán los contenedores terminados en la zona de piezas bloqueadas y se eliminará la posibilidad de que por error vuelvan a meter las piezas a la línea.

## ESTANDARIZACIÓN Y AUTOMATIZACIÓN DEL PROCESO DE GESTIÓN DE DEVOLUCIONES EN ISRINGHAUSEN

A mayores, con la implantación de este proyecto, se conseguirá una mejora intangible en la calidad de trabajo, que se derivará de la disminución de piezas enviadas erróneamente y desembocará en una mejoría de la imagen de la empresa de cara a sus proveedores.

Como desarrollos futuros, se propone la inclusión de una etiquetadora junto al ordenador situado a pie de línea, para en lugar de que los operarios tengan que escribir el número de pieza con un bolígrafo en cada una que sea defectuosa, se le imprima automáticamente al introducirlas en el Excel el número correspondiente para las mismas. De esta manera se ahorraría una gran cantidad de tiempo y se evitaría el posible error humano.

Otro de los desarrollos futuros que se plantean, es la realización de un Excel con un formato similar, pero en este caso dedicado a las piezas que son chatarra, y de las cuáles también debe llevarse a cabo un registro, logrando tanto reducir el tiempo empleado por los operarios como por el encargado para su identificación y bloqueo.

## BIBLIOGRAFÍA

1. Akkandi, B. (2009). The Evolution of C#. Obtenido de <http://blog-mstechnology.blogspot.com/2009/11/evolution-of-c-40.html>
2. APICS. (2017). Supply Chain Operations Reference Model. *Logistics Information Management*, 1096.
3. AUNDE Group SE. (s.f.). We are Isringhausen. Obtenido de <https://www.isri.com/en/>
4. Capuano, B. (2010). Novedades En Visual Studio Team System 2010. Obtenido de <https://www.slideshare.net/elbruno/novedades-en-visual-studio-team-system-2010>
5. Cárdenas Ávila, J. D. (2016). Ventajas y desventajas de programar en VISUAL BASIC. Obtenido de [https://adictoalcodigo.blogspot.com/2016/07/ventajas-y-desventajas-de-programar-en\\_19.html](https://adictoalcodigo.blogspot.com/2016/07/ventajas-y-desventajas-de-programar-en_19.html)
6. Carrera Montenegro, W. (2015). Ventajas y Desventajas. Obtenido de <https://visualbasicwil2015.wordpress.com/ventajas-y-desventajas/>
7. Chouinard, M., D'Amours, S., & Aït-Kadi, D. (2005). Integration of reverse logistics activities within a supply chain information system. *Computers in industry*, 56(1), 105-124.
8. Closing the loop: Redesigning sustainable reverse logistics network in uncertain supply chains | Elsevier Enhanced Reader. (n.d.). Retrieved May 24, 2021, from <https://reader.elsevier.com/reader/sd/pii/S0360835220307634?token=B4EE3330ECC9F77A8E286CA87580A9C6BC87DAD18633AD9C4981CF2D6819D01E55A001D5A3956B7AD22EBE92EEAC7BCE&originRegion=eu-west-1&originCreation=20210524190148>
9. CSCMP. (2013). Supply chain management terms and glossary. Obtenido de [https://cscmp.org/CSCMP/Academia/SCM\\_Definitions\\_and\\_Glossary\\_of\\_Terms/CSCMP/Educate/SCM\\_Definitions\\_and\\_Glossary\\_of\\_Terms.aspx?hkey=60879588-f65f-4ab5-8c4b-6878815ef92](https://cscmp.org/CSCMP/Academia/SCM_Definitions_and_Glossary_of_Terms/CSCMP/Educate/SCM_Definitions_and_Glossary_of_Terms.aspx?hkey=60879588-f65f-4ab5-8c4b-6878815ef92)
10. de Brito, M. P., & Dekker, R. (2004). A framework for reverse logistics. In *Reverse logistics* (pp. 3-27). Springer, Berlin, Heidelberg.
11. de Juana, R. (2020). *Siete alternativas Open Source a Microsoft Visual Studio Code*. Obtenido de <https://www.muycomputerpro.com/2020/06/22/siete-alternativas-open-source-a-microsoft-visual-studio-code>

12. Govindan, K., & Bouzon, M. (2018). From a literature review to a multi-perspective framework for reverse logistics barriers and drivers. *Journal of Cleaner Production*, 187, 318-337.
13. Jesuïtes Educació. (2019). Visual Basic: ventajas y desventajas. Obtenido de <https://fp.uoc.fje.edu/blog/visual-basic-ventajas-y-desventajas/>
14. Ji, G. J. (2008). Reverse Logistics Operation Management Based on Virtual Enterprises and Complaint Service Management. *Journal of Service Science and Management*, 1(01), 51.
15. Khor, K.S., Udin, Z.M., Ramayah, T., Hazen, B.T., 2016. Reverse logistics in Malaysia: the contingent role of institutional pressure. *Int. J. Prod. Econ.* 175, 96e108. <https://doi.org/10.1016/j.ijpe.2016.01.020>.
16. Koltow, D. (s.f.). Características de Visual Basic. Obtenido de [https://techlandia.com/caracteristicas-visual-basic-sobre\\_90231/](https://techlandia.com/caracteristicas-visual-basic-sobre_90231/)
17. Lai, K. H., & Cheng, T. E. (2016). *Just-in-time logistics*. CRC Press.
18. Microsoft. (2018). Información sobre objetos, métodos, propiedades y eventos. Obtenido de <https://docs.microsoft.com/es-es/office/vba/language/concepts/getting-started/understanding-objects-properties-methods-and-events>
19. Morgan, T.R., Tokman, M., Richey, R.G., Defee, C., 2018. Resource commitment and sustainability: a reverse logistics performance process model. *Int. J. Phys. Distrib. Logist. Manag.* 48 (2), 164e182. <https://doi.org/10.1108/IJPDLM-02-2017-0068>.
20. Ortíz, M. (s.f.). Objetos, propiedades y métodos. Obtenido de <https://exceltotal.com/objetos-propiedades-y-metodos/>
21. Programas Perú. (2018). Ejemplos Do While en VBA. Obtenido de <https://www.excel-avanzado.com/ejemplos-do-while-vba>
22. Rancel, M. R. (s.f.). Ejemplo de programa básico en Visual Basic con Option Explicit On, Form, String, etc. Obtenido de [https://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=37&Itemid=61](https://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=37&Itemid=61)
23. Saavedra Gutierrez, J. A. (2008). Historia de Visual Básic. Obtenido de <https://jorgesaavedra.wordpress.com/2008/04/16/historia-de-visual-basic/>

## BIBLIOGRAFÍA

24. Safdar, N., Khalid, R., Ahmed, W., & Imran, M. (2020). Reverse logistics network design of e-waste management under the triple bottom line approach. *Journal of Cleaner Production*, 272, 122662.
25. Secumate. (2011). Definición e historia de Visual Basic. Obtenido de <https://sites.google.com/site/lestadmte/2-definicion-e-historia-de-visual-basic>
26. Weebly. (2013). Ejemplos en Visual Basic. Obtenido de <https://adrianita7.weebly.com/ejemplos/ejemplos-en-visual-basic>
27. Zailani, S., Govindan, K., Shaharudin, M. R., & Kuan, E. E. L. (2017). Barriers to product return management in automotive manufacturing firms in Malaysia. *Journal of cleaner production*, 141, 22-40.