



GRADO EN INGENIERÍA INFORMÁTICA
TRABAJO DE FINAL DE GRADO

**Dashboard mediante tecnología Power
BI, lenguaje DAX y librería pbiviz**

Autor:

Iván Jesús

MONFERRER MARZÁ

Supervisor:

Francisco José

MILLÁN SANTACREU

Tutor:

Ángel Pascual

DEL POBIL FERRÉ

Fecha de lectura: 25 de junio del 2020

Curso académico 2019/2020

A Sofía y Atenea

Por su compañía a Lorena

A mis padres y mi hermano por su apoyo y su comprensión

A todos los maestros incansables que inspiran con su intención

A los componentes del departamento IT de UBE por su cálido
acogimiento y la experiencia

Paco, Germán, Miguel, Antonio, Pedro y Eva

Gracias

Resumen:

Los Tableros inteligentes permiten obtener una gran cantidad de información en un corto espacio de tiempo. Power BI se trata de una herramienta con la que generar tableros utilizando fuentes de datos diferentes, campos calculados o la creación de objetos visuales propios. Es a través del lenguaje DAX que podremos realizar diferentes cálculos sobre el modelo que creamos en Power BI y la librería pbiviz la que nos permitirá crear nuestro propios objetos visuales.

Este documento trata de recompilar la suficiente información para conocer a fondo POWER BI y DAX, siguiendo un proceso de ingeniería de software sobre la creación de un Dashboard de rotación de inventario. Además, se dan pinceladas sobre qué encuentra un programador al comenzar a utilizar pbiviz.

Palabras clave:

Tablero, Power BI, Ingeniería de requisitos, DAX

Keywords:

Dashboard, Power BI, Engineering requirements, DAX

Índice de contenido

1. Introducción	7
2. Descripción del proyecto.....	9
2.1 Alcance y Objetivos	9
2.2 Estado del arte	10
2.3 Producto: tablero de rotación de inventario	10
2.4 Restricciones y gestión de riesgos.....	11
2.5 Recursos	12
2.6 Estimación de costes	15
2.7 Metodología	15
2.8 Planificación temporal: Pila del producto	16
2.9 Seguimiento del proyecto	19
3. Documentación de las tecnologías	23
3.1 Power BI	23
3.2 Data Analysis Expressions (DAX).	32
3.3 Paquete de objetos visuales pbviz	40
4. Análisis del sistema	45
4.1 Entrevistas a usuario	45
4.2 Requisitos del sistema principal	49
4.3 Requisitos del subsistema gráfico	51
4.4 Modulación del sistema	52
5. Diseño del sistema	53
5.1 Modelo de datos	53
5.2 Modulo DAX: Cálculo temporal	61
5.3 Modulo DAX: Cálculo de rotación	62
5.4 Diseño de la interfaz.....	63
5.5 Diseño del gráfico.....	66
6. Implementación y pruebas.....	67
6.1 Implementación del tablero.....	67
6.2 Implementación del gráfico	68
7. Conclusión	71
8. Bibliografía	73

1. Introducción

El propósito del presente documento es la documentación, organización y realización de un proceso de ingeniería del software para obtener un producto acorde con las necesidades de un usuario. Este documento forma parte de un proyecto más amplio y trata de servir como apoyo para el trabajo realizado en éste pero también como un archivo con finalidad didáctica. Este trabajo se desarrolla en el marco de la asignatura *EI1054 – Pràctiques externes i projecte de final de grau* del Grado en Ingeniería Informática de la Universidad Jaume I (Castellón). La empresa donde se realiza la estancia de prácticas y se desarrolla el trabajo de fin de grado es UBE Castellón.

El entorno UBE es la suma de un conjunto de empresas con como producto fundamental la producción y su futura distribución de productos químicos. Muchos de ellos con larga cadena de procesado, tales como: caprolactama, fertilizantes, nylon, copoliamidas o polibutadieno. Además de los productos químicos, también mantienen un continuo crecimiento en I+D+I con el fin de desarrollar soluciones tecnológicas y sostenibles para satisfacer las demandas de sus clientes y tendencias de mercado.

Se trata de un grupo internacional con delegaciones y oficinas en: Castellón, Madrid, Düsseldorf, Sao Paulo y México DF. Todas ellas integradas en el grupo japonés: UBE Industries, Ltd. Siendo este último, un grupo creado hace 120 años en la ciudad de Ube. Junto a la maquinaria y la química se unen también como base principal de ésta: los productos farmacéuticos o la energía ^[1].

UBE cuenta con una amplia plantilla de especialistas en sistemas de la información y de computación, dentro del subgrupo de IT se encuentra el equipo de Aplicación, será en éste donde se desarrollarán las prácticas. El trabajo es tutorizado por Francisco José Millán, responsable de sistemas del departamento IT de UBE.

El producto final que se realiza es un *Business intelligence Dashboard*, un panel con datos clave en tiempo real e indicadores de rendimiento. Con el formato, la consistencia y el enfoque adecuados se trata de una herramienta que ayuda a la toma de decisiones en el ámbito ejecutivo. Además, se espera que este documento recoja conceptos y lógicas suficientes para poder servir como guía en las tecnologías y metodologías utilizadas ^[2].

La organización de este documento sigue el siguiente orden: El primer punto del documento es esta introducción. Como punto segundo se describe el proyecto: su alcance, las tecnologías del producto y la metodología. Es al final de éste donde se realizará el seguimiento del trabajo mediante los sprints. El tercer apartado es donde se realizará la documentación de las tecnologías específicas que han sido utilizadas durante el desarrollo de estas prácticas. Como cuarto punto, se realiza un análisis, en él se profundizará en los requisitos del cliente y se estudiarán sistemas parecidos. El quinto y el sexto punto mostraran el diseño y la implementación respectivamente. Tras esto, se cerrará el documento con una reflexión a modo de conclusión tanto a nivel tecnológico como personal.

2. Descripción del proyecto

La sección descripción del proyecto trata de mostrar cuál es la finalidad del proyecto desarrollado, sus características y la metodología de trabajo que se llevó a cabo. En este último apartado se describirá también el seguimiento que se ha hecho para cada sprint.

2.1 Alcance y Objetivos

Alcance:

El alcance del proyecto incluye la creación de un conjunto de Dashboards conforme a las necesidades que describe el cliente. Utilizando para ello tanto diferentes lógicas programables como gráficos. También se incluye documentar las tecnologías utilizadas a través de este documento.

A nivel organizativo: El informe de rotación será utilizada por analistas a fin de obtener de forma efectiva información que sea de ayuda a la toma de decisiones. Se espera que la documentación desarrollada en este documento también pueda servir para desarrollar futuros Dashboards en la empresa.

A nivel funcional: Ha de permitir tener todos los ítems considerados necesarios para realizar un Dashboard correcto de la rotación de inventarios. Además de un documento de formación sobre las tecnologías utilizadas. A nivel informático: Sistema desarrollado sobre el marco de Dashboards existente en UBE.

Objetivo principal:

Obtener el conjunto de requisitos de los clientes objetivos, analizar estos a fin de generar un conjunto de dashboards mediante la aplicación POWER BI y sus herramientas anexas. Documentando cada tecnología utilizada a fin de servir este documento como archivo docente.

Desglosado en los siguientes objetivos específicos:

- Realizar una correcta Ingeniería de Requerimientos. Esto es, citando a la IEEE (Institute of Electrical and Electronics Engineers) *“Condición o capacidad que debe satisfacer o poseer un sistema [...] para satisfacer una especificación u otro documento formalmente impuesto”* ^[3]. Incluyendo para este fin: Entrevistas con los clientes, Análisis y Diseño.
- Aprendizaje de la aplicación POWER BI.
- Documentar el lenguaje de fórmulas DAX (Data Analysis Expressions).
- Modelar los datos necesarios para el Dashboard y comprender su lógica y cardinalidad.
- Utilizar lenguaje DAX para diseñar las soluciones a los cálculos del Tablero.
- Conocer qué datos y cómo se encuentran en el SAP interno de UBE. Para descubrir cuáles serán útiles para la representación del modelo.
- Creación de un informe de rotación de inventario en un formato funcional e intuitivo.
- Creación de objetos visuales mediante el paquete pbviz utilizando el lenguaje TypeScript. Y generar documentación sobre su implementación.

2.2 Estado del arte

Un Tablero de control representa de manera visual información y umbrales específicos utilizando medidas de tiempo adecuadas. Esto es equivalente a que permite la representación de lo fundamental de un objetivo o la monitorización del comportamiento de una organización. Cuando un Dashboard está realizado correctamente es capaz de agilizar la toma de decisiones de los usuarios.

La clave de la realización de un Dashboard funcional está en centrar el proceso en el usuario y eliminar para éste las barreras cognitivas. Es decir, se consigue que el usuario pueda acceder a los datos que quiere sin ninguna dificultad. Para poder llegar a tener un razonamiento sobre una situación actual determinada. Dicho de otra manera: Conocer de forma ágil la información de un estado y poder llevar decisiones que requiere sin gran dificultad ^[4].

2.3 Producto: tablero de rotación de inventario

El Tablero a desarrollar en este proyecto gira en torno a la rotación de inventario. Específicamente en la rotación de inventario de tres conjuntos: Producto acabado (aquél que está dispuesto ya a venta o consumo), embalajes (útiles para el transporte y almacenamiento) y materia prima y aditivos (utilizados como base para la creación de un producto acabado). Un elemento podría ser al mismo tiempo producto acabado y materia prima.

La rotación de inventario es la frecuencia con la que una compañía vende sus productos físicos, también incluye otros tipos de salidas de estos. Es decir, determina cuán rápido sale de almacén un producto. El almacenaje de un producto lleva consigo un coste, además el valor en el mercado puede disminuir, produciendo una responsabilidad financiera a asumir ^[5]. La métrica que se utilizará para determinar esta rotación es: la previsión en cantidad de días que tardará en salir el stock actual siguiendo la tendencia de su salida a 12, 6 y 3 meses.

Una **rotación baja** implica que no hay mucha demanda o tiene el precio por encima de lo que los clientes están dispuestos a pagar. Una **rotación alta** implica un precio bajo o que la compañía no fabricó suficiente para cubrir la demanda.

La información que debe mostrarse tiene como eje aquellos ítems que son almacenados. Todos ellos tienen un código de identificación, además, de campos que se asocian al tipo de material y, por otro lado, aquéllos que identifican el ítem específico. Será en el apartado de Análisis del sistema donde se recogerá qué se necesita y, en el de Diseño, cómo conseguirlo con detalle

2.4 Restricciones y gestión de riesgos

A través de la gestión de riesgos se espera identificar éstos y solventarlos lo antes posible y de la mejor manera. Siendo el fin: maximizar la probabilidad de que el producto llegue al éxito. Se presentará primero los riesgos identificados, después su gestión y por último las restricciones que presenta el proyecto ^[6].

Identificación de riesgos

R01: Estimaciones temporales incorrectas. Debido a no conocer cuánto costará desarrollar actividades tipo sobre las que se realiza un primer enfrentamiento.

R02: Error en identificación de requisitos. La falta de pericia al enfrentar por primera vez entrevistas, puede dejar algunos requisitos sin recogerse.

R03: El alcance puede no estar bien definido. Asociado al R02 u otras circunstancias, el alcance puede quedarse corto.

R04: Existen incoherencias en el modelo realizado. Es posible que los datos identificados en un primer momento no se ajusten a los que después se puedan obtener desde las fuentes de datos.

Gestión de riesgos identificados

R01: Cada vez que se realice una tarea nueva, temporalizara y modificar la estimación de tareas parecidas. Si llega a producirse un desbarajuste alto, reestructurar la pila del producto.

R02: Para evitar que se produzca se han de mejorar las competencias en elicitación de manera continua. Con especial importancia el análisis posterior a la entrevista con el cliente. Si se produce se habrá de crear nuevos de elementos de la pila del producto.

R03: En cada momento en el que aparezca la oportunidad se habrá de refrescar los requisitos al cliente, obtener su feedback y refinarlos cada vez más. Si aparece la situación, marcar una reunión con los clientes a fin de asegurar que los requisitos actuales sí son los esperados.

R04: Para evitar que se produzca se ha de tratar de estar atento al realizar cada modificación y al realizar cada nueva consulta dentro del sistema. Si llega a producirse, será de vital importancia entender correctamente la naturaleza de la consulta y redefinirla en el modelo.

Restricciones

Se identifican restricciones a tres niveles en la realización del proyecto:

Temporal. La realización de todo el proyecto tiene un plazo de 300 horas. Si se produce alguna situación imprevista que fuerce a un retraso en la planificación será necesario recalcular la planificación.

Recursos humanos. Proyecto realizado por una sola persona.

Trabajo telemático. Puesto que en el desarrollo de este proyecto se produjo la situación de alarma causada por Covid-19, todo el desarrollo del mismo, incluyendo las reuniones con clientes se realizó de manera telemática.

2.5 Recursos

En este apartado se recoge qué recursos divididos a nivel de herramientas y de tecnologías han sido necesarias para llevar a cabo el proyecto. En la segunda parte del mismo se realizará una estimación de los costes tanto de los recursos materiales como humanos.

A nivel de **Herramientas** se contemplan aquéllos recursos físicos reutilizables:

- Ordenador hp portátil corporativo
- Periféricos para ordenador: Monitor también de marca hp, teclado hp y ratón Gamer Trust Gixt.

A nivel de **Tecnologías** se presentan los diferentes recursos que se han utilizado en el proyecto de manera breve, en el caso del Microsoft Power BI, el DAX y el paquete pbiviz se realiza un estudio de estas en el apartado *Documentación de las tecnologías*.

Conexión remota a servidor de UBE

Se utiliza un ordenador corporativo con conexión remota a la red de UBE mediante VPN. Una red privada virtual (en inglés Virtual Private Network) se trata de una tecnología que permite conectar dispositivos a una red privada utilizando internet.

Microsoft Power BI

Se trata de un servicio de Microsoft con aplicación de escritorio, móvil y también web. A través de ella se pueden realizar tableros con diferentes orígenes, a nivel de datos y formato, e interrelacionarlos. Además se mantiene siempre actualizado con el origen.

Permite el desarrollo de tableros con diferentes gráficos que permiten realizar un análisis empresarial. Estos permiten mostrar diferente información y compartirla tanto a través de aplicaciones propias como del sitio web de la empresa.

Cuenta con una gran cantidad de objetos visuales que, además, podrán extenderse a través de un paquete propio de Power BI (pbiviz). Todo informe es interactivo, es decir, permite la selección de algún ítem de sus elementos gráficos y la actualización del resto respecto a esta selección.

Data Analysis Expressions (DAX)

Se trata de una librería de funciones y operadores que pueden combinarse para obtener expresiones tanto en Power BI como Excel (mediante el uso de Power Pivot) o en los modelos tabulares de SQL Server Analysis Services.

DAX también puede definirse como lenguaje funcional (orientado a funciones) dirigido al análisis de datos. Dentro de su funcionalidad se incluyen medidas (que pueden cambiar su comportamiento dependiendo del contexto), columnas calculadas o tablas calculadas. En la sección documentación se dará a conocer más sobre este lenguaje.

Visual Studio Code

Se trata de un editor de código gratuito desarrollado por Microsoft. VSC es realmente ligero al solo tener aquéllas herramientas que se configuran a través de plugins o peticiones de terminal. Estos complementos permiten utilizar el editor con comportamientos de compilador, intérprete, depurador u otros programas relevantes a la hora de desarrollar Software. El código de la aplicación se desarrolla a través de Github donde se permite a programadores crear issues (o notificaciones) para reportar peticiones.

Cuenta con un archivo de configuración de editor con una gran variedad de posibilidades, incluyendo la reescritura de los atajos. Esto unido a la posibilidad de poder trabajar con una amplia gama de lenguajes y depuradores hace de Visual Studio Code una herramienta muy potente.

Para el desarrollo de este proyecto se utilizará el editor para desarrollar elementos gráficos de Power BI a través de su librería específica. Para hacerlo, será necesario preparar el entorno con node.js y utilizar el lenguaje Typescript.

JavaScript

Se trata de un lenguaje *first-class function*, es decir, orientado a funciones: Éstas podrán ser asignadas a variables, utilizadas como argumento o como valor de vuelta de otras funciones. Se trata de un lenguaje fundamental en la programación web que ha cobrado especial relevancia en los últimos años. Así, alrededor del 65% de los programadores utilizaron JavaScript en algún proyecto en el 2018 ^[7].

En la programación de los gráficos se utilizará el lenguaje TypeScript pero éste, será transpilado a JavaScript antes de su ejecución. Este proceso es parecido al de la compilación con la diferencia de que el resultado sigue siendo un lenguaje de alto nivel. Este último llevará consigo la misma funcionalidad que la realizada en TypeScript.

TypeScript

Se trata de un lenguaje de programación de código abierto desarrollado por Microsoft. TypeScript funciona como una capa sobre JavaScript, permitiendo toda la expresión de éste. TypeScript permite, entre otros, formalizar los tipos de los archivos .js, esto nos permite definir variables o funciones tipadas, disminuyendo la probabilidad de errores de ejecución.

Este lenguaje funciona como un superconjunto de JavaScript, es decir, su código puede presentar sus formas propias pero también escribirse con el formato habitual de JavaScript. Todo

aquello realizado mediante TypeScript se transpila en JavaScript antes de poder ser utilizado en la máquina.

Entorno Node.js

Node.js es posiblemente la causa de que haya aumentado el uso de JavaScript. Así esta librería permite que código que solo podía ser ejecutado a nivel de navegadores a código que puede ejecutarse a nivel de ordenadores como una aplicación independiente.

A través de este entorno se permite un modelo de entrada y salida sin bloqueo permitiendo referir cualquier operación de forma ligera y eficiente. Node.js funciona como un solo hilo controlado por eventos que permite el manejo simultáneo de peticiones. Su meta es: “proporcionar una manera fácil para construir programas de red escalables” ^[8].

Paquete pbiviz

Power BI cuenta con la posibilidad de crear objetos visuales adicionales para utilizarlos en paneles e informes. Cada uno de los que cuenta la versión por defecto de la aplicación se encuentra en GitHub y sirve de base para la creación de nuevos. En la documentación de tecnologías se describirá la funcionalidad de este paquete.

Less CSS

El lenguaje Less CSS es un lenguaje de diseño gráfico que compila antes de utilizarse en una hoja de estilos CSS. A través del lenguaje JavaScript, la librería node.js y la librería propia (Less) lo hará de manera automática. De la forma que se ha utilizado en el proyecto no requiere de ningún prerequisite específico por estar dentro de las dependencias del esqueleto de un proyecto de pbiviz.

Las mejoras que nos da Less son la creación de variables, *mixins* que nos permite definir clases de estilo como funciones, anidar clases, el uso de operadores (también usando variables), funciones propias de Less (lógicas, de tipos básicos, de imágenes) o también importar desde otro archivo Less ^[9].

2.6 Estimación de costes

Los costes humanos del desarrollo del producto son igual al trabajo de un profesional durante 300 horas. A nivel de licencias para extraer todo el potencial de la aplicación, sería necesario obtener la Power BI Pro con el coste de 9.99\$ por usuario. Para este último ítem se considerarán 4 meses por ser el tiempo en que se utilizó la aplicación.

También ha de sumarse el coste del equipo que se ha utilizado para desarrollar el producto más todos sus periféricos necesarios que hace un total de 900€. A lo anterior también se ha de agregar el coste de servicios de mantenimiento y desarrollo: limpieza, luz... Este último se fija a 120€/mes teniendo en cuenta la extrapolación de los costes entre varios departamentos y despachos. El coste total será de 5036,4€ véase el desglose en la Figura 1.

Objeto	Coste económico
Profesional 300h	$300h * 11€/h = 3300€$
Licencia Power Bi Pro 9.99\$ al mes	$9.99\$/mes * 4meses * 0,91\$/€ = 36,4€$
Equipo y periféricos	900€
Costes varios: limpieza + luz	$200 * 4 meses = 800€$
TOTAL	5036,4€

Factor de conversión Dólar estadounidense a Euro a 28/05: 0,91

Figura 1: Cálculo de costes

2.7 Metodología

La metodología de este proyecto se llevará a cabo utilizando la metodología ágil, más concretamente: **Scrum**. Para ello se realizará en primera instancia una pila del producto que será revisada al finalizar cada sprint. Además, se dividirá el trabajo en sprints de bloques de 15 días laborables, haciéndolos coincidir así con los envíos periódicos propuestos para esta asignatura. Esto significa que habrá un total de tres sprints completos y un cuarto sprint que será de 10 días laborables. Se ha decidido utilizar la metodología Scrum dado que ^[10] ^[11]:

- Los requisitos no están definidos de forma clara. Scrum nos da un formato con el que comenzar a trabajar sin tener todavía todos los datos definidos.
- Son probables los cambios durante el desarrollo. Así, una buena parte del proyecto pivotará al conocer las necesidades del cliente y las herramientas a utilizar.
- La aprobación del cliente, al que se podrá acceder de manera semanal, es clave para la realización correcta del producto.

2.8 Planificación temporal: Pila del producto

La pila del producto (o Product Backlog) se trata de, en diferentes posibles formatos, una lista de tareas y requisitos que necesita el producto final. Ésta se encuentra en constante evolución, modificándose para adaptarse a la nueva situación del proyecto. Así, esta pila ha ido mejorándose una vez cerrado el apartado del Análisis (Sprint 2). Las diferentes tareas pueden agruparse en macros o épicas que permiten abstraer el significado de un conjunto de tareas. En la pila se presentan las tareas con la siguiente información ^[12] ^[13]:

- **Identificador.** Un código único que permite identificar de forma inequívoca una tarea.
- **Descripción.** Unas pocas palabras que sirvan para describir el trabajo que se desarrollará en una tarea específica.
- **Estimación** en puntos de historia. Se toma como punto de esfuerzo, un bloque de trabajo propio seguido que se ha asociado a aproximadamente 2 horas.
- **Valor** para el negocio ^[14]. Se utiliza la escala cualitativa MoSCoW para dar la importancia necesaria a cada requisito. Así, para cada elemento de la pila del producto, se ha valorado: Aquello que debe estar presente en el producto (MUST), aquello que debería estar (SHOULD), lo que sería conveniente (COULD) y aquello que no está presente en los planes pero puede pasar a otro estado en iteraciones futuras (WON'T).
- Se establece un orden basado en prioridad. Con los dos puntos anteriores: estimación de coste y valor.

Se definen cinco épicas para el proyecto, numeradas del 1 al 5. Cada una de las tareas se identificará con un número que referenciará una de ellas y una letra que la definirá de forma única. Las épicas son las siguientes:

1. Documentación de con herramienta Power BI
2. Documentación del lenguaje de fórmulas de Power BI: DAX
3. Documentación de librería de Power BI para *TypeScript* que permite la creación de gráficos personalizados.
4. Realización de Ingeniería de requisitos
5. Realización de Prototipado Evolutivo

Para la estimación de los puntos de esfuerzo se utiliza una previsión utilizando los número de Fibonacci: 1, 2, 3, 5, 8, 13... Para cada tarea se valora el tiempo invertido en tareas que se hicieron con anterioridad de naturaleza parecida. Con esto, se aproxima a la alta a razón de un punto de esfuerzo – dos horas de trabajo.

La escala cualitativa MoSCoW se utiliza del siguiente modo: Aquello que debe estar se le da un valor entre 80 y 100 ambos incluidos. Aquello que debería estar se le da un valor entre 50 y 70; lo que es conveniente entre 30 y 50 y donde el valor percibido es menor de 30 se trata de aquello que se encuentra fuera de la planificación inicial pero puede cambiar su estado en el desarrollo de un sprint a otro. Puede verse el resultado de la realización de la pila del producto en la figura 2.

Id.	Descripción	Estimación	Valor
1a	Aprender formato básico de inserción de visualizaciones	2	40
1b	Documentar funcionalidad básica Power BI	5	50
1c	Documentar configuración de tablas y tipos	3	40
1d	Recoger diferentes Power BI visuals	8	30
1e	Obtener datos de diferentes orígenes para realizar prototipos	3	50
1f	Especificar modelo sobre documento EXCEL	3	80
1g	Conexión con datos de UBE	8	100
2a	Documentar tecnología DAX	13	80
2b	Generar archivo pbix con diferentes funcionalidades de análisis de datos	13	60
2c	Generar las soluciones para el proyecto con DAX	8	100
3a	Preparar entorno de programación	3	80
3b	Documentar formato de archivos de la librería pbiviz	8	60
3c	Conectar PowerBI Desktop con Power BI Web para poder mostrar los gráficos en remoto	5	20
3d	Documentar librería pbiviz	13	60
3e	Probar gráficos base del repositorio de PowerBI	21	30
3f	Generar prototipo de gráfico objetivo	54	50
4a	Preparación de entrevista inicial	2	90
4b	Realización de entrevista inicial	1	100
4c	Recompilar información de entrevista inicial	2	80
4d	Preparación, realización y análisis de entrevista02	5	100
4e	Validar entrevista inicial en entrevista02	1	80
4f	Preparación, realización y análisis de entrevista03	5	100
4g	Validar entrevista02 en entrevista03	1	80
4h	Preparación, realización y análisis de entrevista04	5	100
4i	Validar entrevista03 en entrevista04	1	80
4j	Preparación, realización y análisis de entrevista05	5	100
4k	Validar entrevista04 en entrevista05	1	80
5a	Análisis de requisitos de producto	13	90
5b	Soluciones para módulo temporal	5	90
5c	Diseño y prueba de prototipo01	13	90
5d	Preparar muestra de prototipo01	2	70
5e	Obtener feedback de prototipo01	5	100
5f	Refinamiento del Análisis	5	90
5g	Diseño y prueba de producto final	13	90
5h	Preparar muestra de producto final	2	70
5i	Feedback final	5	100

Figura 2: Pila de producto

En la Figura 2, se ha decidido utilizar un marco de colores para identificar aquello que se propone para cada sprint. Se espera que para cada sprint de tiempo a realizar las tareas asignadas para cada uno de ellos. Que quedaría del siguiente modo:

- Sprint 1: Verde
- Sprint 2: Amarillo
- Sprint 3: Naranja
- Sprint 4: Azul

2.9 Seguimiento del proyecto

El tamaño de cada sprint se marca en 15 días, sincronizándolo de esta manera con los envíos periódicos propuestos para esta asignatura. Será así para todos los sprints exceptuando el cuarto que será de 10 días.

En cada sprint se ha realizado una tabla para representar las tareas que se han desarrollado. El primero se encuentra en la figura 3, el sprint 2 en la figura 4, el Sprint 3 en la figura 5 y el sprint final en la figura 6.

Para cada ítem de la pila de productos pasado a tareas del sprint se añade la columna "Observaciones". Ésta se encuentra vacía al iniciar el sprint, cuando una tarea se considera acabada se marca a "COMPLETADA", si está acabada pero será necesario hacer una revisión a fondo conforme evolucione el proyecto se marca a "A REVISIÓN", si no se ha conseguido acabar una de las tareas del sprint se marca a "PASADO A sprint siguiente". Aquéllas tareas que no se han podido terminar en un sprint se ha documentado en él.

Sprint 1

Del 04/03/2020 a 31/03/2020

Id.	Descripción	Observaciones
1a	Aprender formato básico de inserción de visualizaciones	COMPLETADA
1b	Documentar funcionalidad básica Power BI	COMPLETADA
1c	Documentar configuración de tablas y tipos	A REVISIÓN
1d	Recoger diferentes Power BI visuals	PASADO A Sprint2
1e	Obtener datos de diferentes orígenes para realizar prototipos	COMPLETADA
2a	Documentar tecnología DAX	A REVISIÓN
4a	Preparación de entrevista inicial	COMPLETADA
4b	Realización de entrevista inicial	COMPLETADA
4c	Recompilar información de entrevista inicial	COMPLETADA
4d	Preparación, realización y análisis de entrevista02	COMPLETADA

Figura 3: Tareas Sprint 1

Desviaciones y corrección:

La tarea con menor valor para el proyecto para este sprint: Recoger diferentes Power BI visuals, se dejó para el final del sprint y finalmente no pudo completarse. Así, se espera desarrollarla en el siguiente sprint. Además, se espera perfilar las tareas 1c y 2a en el resto de sprints.

Sprint 2

Del 31/03/2020 a 23/04/2020

Id.	Descripción	Observaciones
1d	Recoger diferentes Power BI visuals	PASADO A Sprint3
1f	Especificar modelo sobre documento EXCEL	COMPLETADA
1g	Conectar con datos de UBE sobre modelo	PASADO A Sprint3
2b	Generar archivo pbix con diferentes funcionalidades de análisis de datos	A REVISIÓN
3a	Preparar entorno de programación	COMPLETADA
3b	Documentar formato de archivos de la librería pbviz	A REVISIÓN
3c	Conectar PowerBI Desktop con Power BI Web para poder mostrar los gráficos en remoto	COMPLETADA
4e	Validar entrevista inicial en entrevista02	COMPLETADA
4f	Preparación, realización y análisis de entrevista03	COMPLETADA
4g	Validar entrevista02 en entrevista03	COMPLETADA
4h	Preparación, realización y análisis de entrevista04	COMPLETADA
4i	Validar entrevista03 en entrevista04	COMPLETADA
5a	Análisis de requisitos de producto	A REVISIÓN

Figura 4: Tareas Sprint 2

Desviaciones y corrección:

La tarea 1d: Recoger diferentes Power BI visuals, se le da poca prioridad y de nuevo se pasa al siguiente sprint. La tarea 1g se encuentra aplazada al siguiente sprint, siendo que una vez especificado el modelo de datos, se tuvieron que hacer algunas modificaciones de acceso sobre el sistema de archivos de UBE, esto significó llevar esta tarea al siguiente sprint.

Se espera mejorar la funcionalidad de análisis de datos de la tarea 2b y 3b. Desde este sprint donde se pusieron las bases hasta el final del proyecto habrá de realizarse una mejora continua del análisis (tarea 5a).

Sprint 3

Del 24/04/2020 al 15/05/2020

Id.	Descripción	Observaciones
1d	Recoger diferentes Power BI visuals	COMPLETADA
1g	Conexión con datos de UBE	A REVISIÓN
2c	Generar las soluciones para el proyecto con DAX	COMPLETADA
3d	Documentar libería pbiviz	
3e	Probar gráficos base del repositorio de PowerBI	COMPLETADA
4j	Preparación, realización y análisis de entrevista05	COMPLETADA
4k	Validar entrevista04 en entrevista05	COMPLETADA
5b	Soluciones para módulo temporal	Pasado a sprint 4
5c	Diseño y prueba de prototipo01	COMPLETADA
5d	Preparar muestra de prototipo	COMPLETADA
5e	Obtener feedback de prototipo01	COMPLETADA
5f	Refinamiento del Análisis	COMPLETADA

Figura 5: Tareas Sprint 3

Desviaciones y corrección:

Se inserta en la pila la tarea 5f al ser necesario realizar mejoras sobre el análisis del sistema que implicaron una cantidad de tiempo significativa. También se insertan en la pila las tareas: 5g, 5h y 5i.

La tarea 5b no dio tiempo a resolverla en este sprint, como estaba planeado, se pasa al último sprint. También se valor hacer una revisión sobre la conexión de los datos de UBE.

Sprint 4

Del 18/05/2020 al 29/05/2020

Id.	Descripción	Observaciones
3f	Generar prototipo de gráfico objetivo	SIN CONCLUIR
5b	Soluciones para módulo temporal	COMPLETADA
5g	Diseño y prueba de producto final	COMPLETADA
5h	Preparar muestra de producto final	COMPLETADA
5i	Feedback final	COMPLETADA

Figura 6: Tareas Sprint 4

Desviaciones y corrección:

Se cierran las revisiones sobre el análisis. Se mejora la documentación de DAX. Se revisa la conexión con los datos internos y se utiliza solo aquello necesario. Ahorrando en la cantidad de consultas.

Se invierte mucho tiempo en conocer cómo funciona la librería pbiviz, se comienzan a dar pasos pero no puede completarse el gráfico propuesto.

3. Documentación de las tecnologías

Este apartado documenta las tecnologías que más se han utilizado e investigado en el desarrollo del proyecto. En él se describen, se da ejemplos de su uso y su funcionalidad. Para realizar esta documentación, se han tomado diferentes vías que se han añadido a la bibliografía. Estas tecnologías son: Power BI, DAX y el paquete pbviz.

3.1 Power BI

El proyecto se dirige a la realización de un conjunto de Dashboards utilizando la herramienta de Microsoft: Power BI. Se trata de un servicio de análisis empresarial, es decir, la recopilación de datos a nivel de funciones y procesos empresariales que permita obtener información (que pueda ayudar a la toma de decisiones) a los usuarios finales.

Power BI permite utilizar en su interfaz diferentes lenguajes que aumentan su funcionalidad: Python, R o DAX. De ellos, se utilizará el DAX (Data Analysis Expressions) para realizar comportamientos complejos de los datos que han de generarse para obtener el modelo. También se utilizará a fin de generar filtrados especiales en los gráficos.

Funcionamiento básico de Power BI

Al abrir la aplicación Power BI desktop tenemos tres pestañas a su izquierda, cada una se identifica con uno de los conjuntos de herramientas básicos de Power BI. Por su orden de aparición: informe, datos y modelo ilustrado en la figura 7. El primero hace referencia a la visión final del **informe**, en la pestaña de **datos** podemos ver ejemplos de los datos que tenemos cargados (para ver que realmente se comportan cómo queremos) y en el último (**modelo**) tenemos las tablas que hemos cargado desde la opción “Obtener datos”, es aquí donde también podremos realizar las diferentes interacciones entre datos de diferentes orígenes.

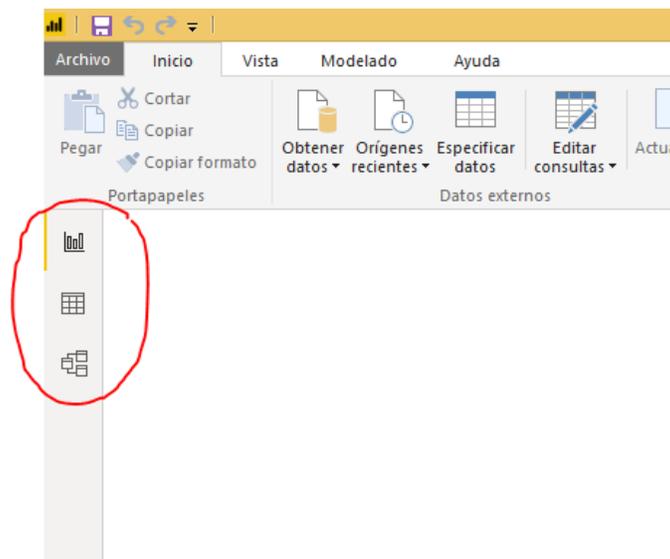


Figura 7: Power BI - Pestañas

Para comenzar a trabajar con esta herramienta, realizaremos una carga básica de datos. Pese a que tiene muchas posibilidades (SQL, Oracle Server, SAP, Web) como formato de entrada, se utilizará un Excel descargado de la propia página de Microsoft. Se trata de un ejemplo al que se alude en muchos de los hilos de preguntas y respuestas de la comunidad. Puede encontrarse en el siguiente enlace: <https://docs.microsoft.com/es-es/power-bi/sample-financial-download>

Una vez cargados los datos en el Power BI es el momento de entrar en la ventana de datos para asegurar que todos los datos cumplen con el tipo esperado (véase figura 8). En esta vista, podremos cargar las diferentes tablas y modificar su tipo (cadena de texto, decimal, monetario...). Así como la creación de tablas calculadas y medidas (a través de DAX).

Segment	Country	Product	Discount Band	Units Sold	Manufacturing Price	Sale Price	Gross Sales	Discounts	Sales	COGS	Profit	Date	Month Number
Government	Germany	Carretera	None	1513	3	350	529550	0	529550	393380	136170	lunes, 1 de diciembre de 2014	12
Government	Germany	Paseo	None	1006	10	350	352100	0	352100	261560	90540	domingo, 1 de junio de 2014	6
Government	Canada	Paseo	None	1725	10	350	603750	0	603750	448500	155250	viernes, 1 de noviembre de 2013	11
Government	Germany	Paseo	None	1513	10	350	529550	0	529550	393380	136170	lunes, 1 de diciembre de 2014	12
Government	Germany	Velo	None	1006	120	350	352100	0	352100	261560	90540	domingo, 1 de junio de 2014	6
Government	France	VTT	None	1527	250	350	534450	0	534450	397020	137430	domingo, 1 de septiembre de 2013	9
Government	France	Amarilla	None	2750	260	350	962500	0	962500	715000	247500	sábado, 1 de febrero de 2014	2
Government	Mexico	Carretera	Low	1210	3	350	423500	4235	419265	314600	104665	sábado, 1 de marzo de 2014	3

Figura 8: PowerBI - Pestaña Datos

Además de la pantalla anterior, existe la pantalla editor de **Query** que se abrirá al seleccionar Inicio-Editar Consultas. En esta pantalla podemos hacer configuraciones complejas sobre los datos que hemos añadido, tales son: modificar el formato en el que se presentan las fechas, modificar el valor de una fila o eliminar ciertos datos, tratándolos como nulos. O incluso modificar del mismo fichero las selección de los datos.

Una de las características más interesantes de Power BI es que permite la relación entre datos de diferentes orígenes. El *Sample financial* se trata de un EXCEL proporcionado por Microsoft, aprovechando que tiene el campo país, ilustraremos la conexión con datos de otro origen. En este caso, conectaremos el modelo con una tabla que relaciona cada país con una tabla para cada país su región ^[16].

Para extraer este dato utilizaremos Obtener datos – Web, en la pestaña siguiente nos saldrán más de una tabla (Figura 9), seleccionaremos aquélla que nos interesa y la cargaremos en el modelo.

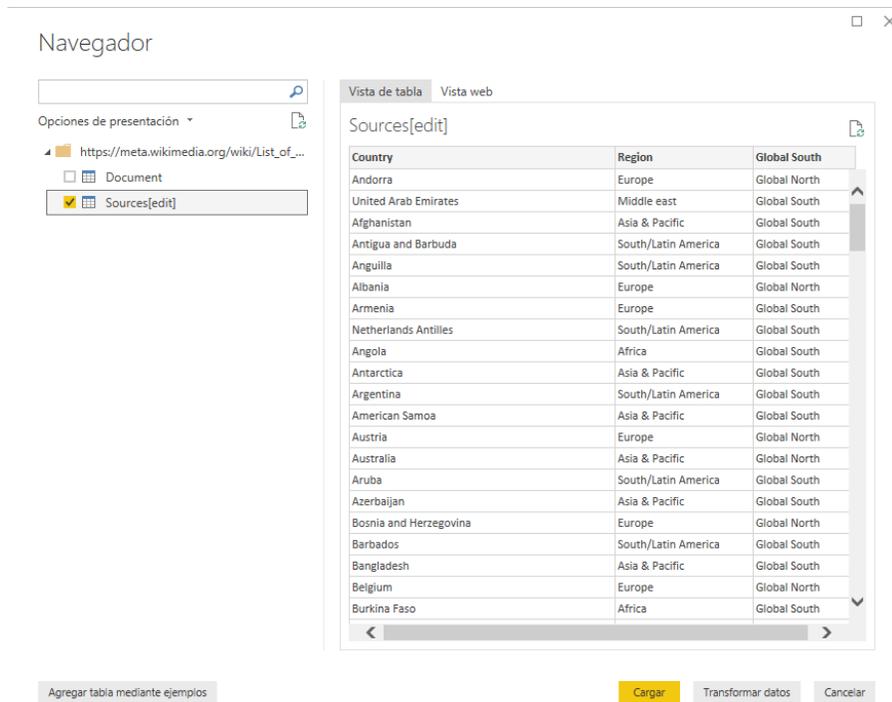


Figura 9: Power BI - Obtención de datos Web

Como último paso para poder utilizarlas, iremos a la pestaña modelo y relacionaremos ambas, en este caso, con una relación muchos a uno. Esta relación se ve marcada en el modelo del Power BI como podemos observar en la figura 10. Una vez realicemos esta acción, para cualquier muestra de datos podrá emplearse *Region* para que se agrupe un subconjunto del contenido de la tabla *financials* por éste.

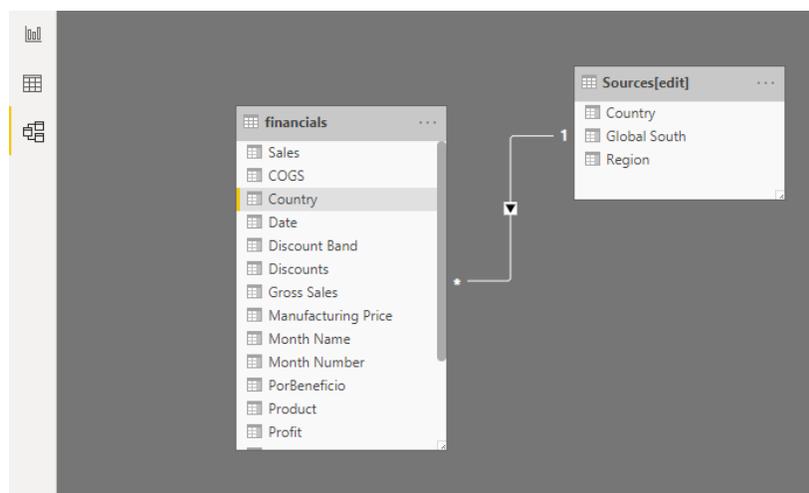


Figura 10: Power BI - Obtención de datos

Cuando todos los datos estén correctamente configurados el siguiente paso es comenzar a configurar el tablero a través de elementos gráficos. En la vista de informe tenemos tres pestañas a la izquierda (figura 11). De derecha a izquierda tenemos: los campos que hemos configurado en nuestro modelo, las opciones para los objetos visuales y los filtros.

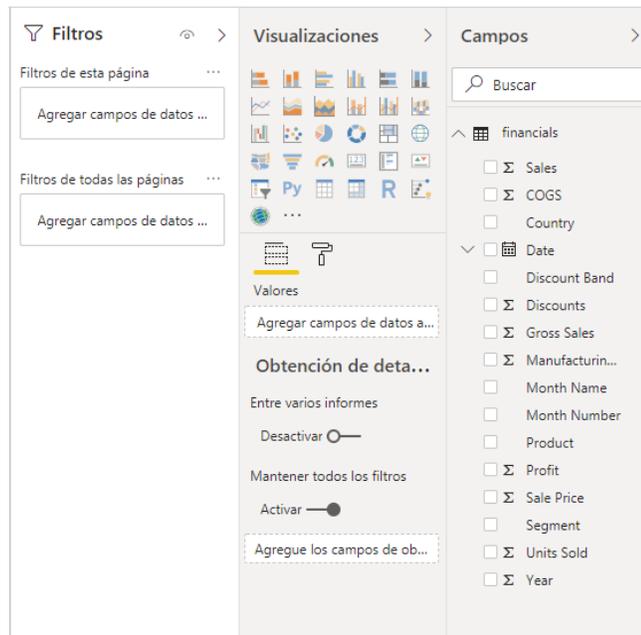


Figura 11: PowerBI – Campos de Informe

En los campos puede verse que cada elemento seleccionable es una de las columnas de nuestras bases de datos ya configuradas con los tipos en la pestaña datos. Tras el check button, como puede verse, hay diferentes símbolos: **sigma** identifica que se resume con el sumatorio total de elementos, el **calendario** que se trata de una columna de fecha y los elementos **sin símbolo** que identifican otra información por la que los anteriores se verán agrupados.

Si clicamos en Manufacturing Price y Sales Price obtendremos una gráfica con dos barras que representarán el total de ambas. Si además seleccionamos el país, se modificará el contexto y con ello, se actualizará la tabla, de forma que: para cada país existirán dos gráficas la de Manufacturing Price y la de Sales Price. Este cambio de contexto en un gráfico puede observarse en la figura 12.

Manufacturing Price y Sale Price

● Manufacturing Price ● Sale Price

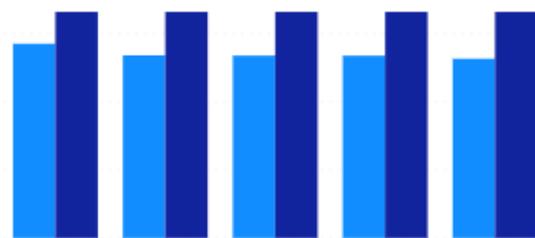
100 mil
80 mil
60 mil
40 mil
20 mil
0 mil



Manufacturing Price y Sale Price por Country

● Manufacturing Price ● Sale Price

20 mil
15 mil
10 mil
5 mil
0 mil



Country

Figura 12: Power BI – Cambio de contexto en gráfico

En la pestaña de visualizaciones podemos insertar los gráficos y, también, elegir sobre el formato de su texto o la paletilla de colores que utiliza. Para este último ítem, existe la posibilidad de importar un tema, que dará valores por defecto a cada color, seleccionando “cambiar tema”. Tras visitar la página, se descarga el archivo .json del tema elegido y de nuevo en la opción “cambiar tema” podremos importarlo.

Tras la importación de “Flat Color”, seleccionando ventas y país en un gráfico circular y realizando unas pequeñas modificaciones de estilo (Título centrado y márgenes existentes) se consigue un gráfico con temática observable en la figura 13.

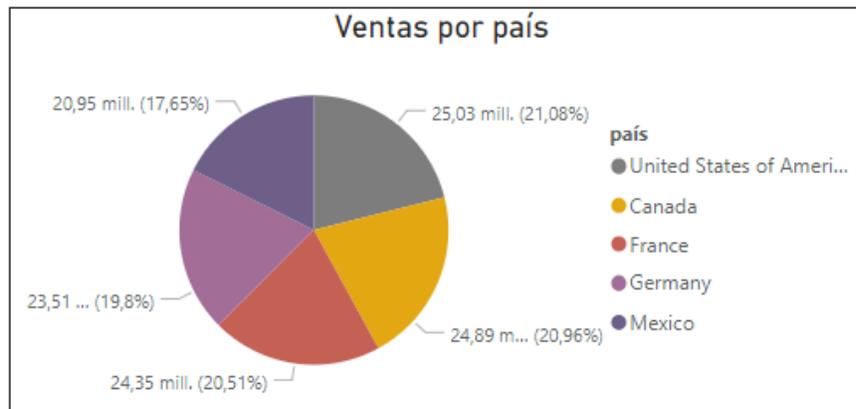


Figura 13: PowerBI – Gráfico circular con Tema Flat Color

Hasta aquí lo más relevante de lo utilizado en Power BI Desktop respecto a sus tres ventanas y su funcionalidad básica.

Vistas disponibles en Power BI

Power BI Desktop cuenta con vistas básicas, entre ellas ^[17]:

- **Gráfico lineal.** Presente una o más líneas que son valores cuantitativos de una determinada categoría. Es el elegido para mostrar diferentes tendencias a lo largo del tiempo.
- **Gráfico de columnas.** Las columnas o diagrama de barras presentan una longitud variable que representa un valor. Son utilizadas para comparar valores entre categorías: como estimación/ real.
- **Mapas de puntos.** Permiten visualizar conjuntos de datos en puntos geográficos, el elemento gráfico que representa la cantidad tendrá tamaños diferentes para mostrar la cantidad.
- **Gráficos de burbujas.** No utiliza puntos, sino burbujas que representan una dimensión adicional de los datos. Son útiles para cuadrantes (utilizan los cuatro regiones del sistema de coordenadas Cartesianas) o si los datos tienen tres series de datos con valores.
- **Gráficos de anillo.** Se trata de una corona circular donde cada sección con un color específico representa un grupo y el tamaño de una franja la cantidad de cada grupo.

Para relacionar fechas en un gráfico, tendremos que desplegar el seleccionable, de tal manera que nos aparecerán los diferentes formatos de fecha que se han guardado en la columna. Los básicos son: Año, trimestre, mes y día. Para crear otros formatos tendremos que utilizar el DAX. En la figura 14 podrá verse el resultado de un gráfico lineal con el total de descuentos y beneficios a lo largo de los meses.

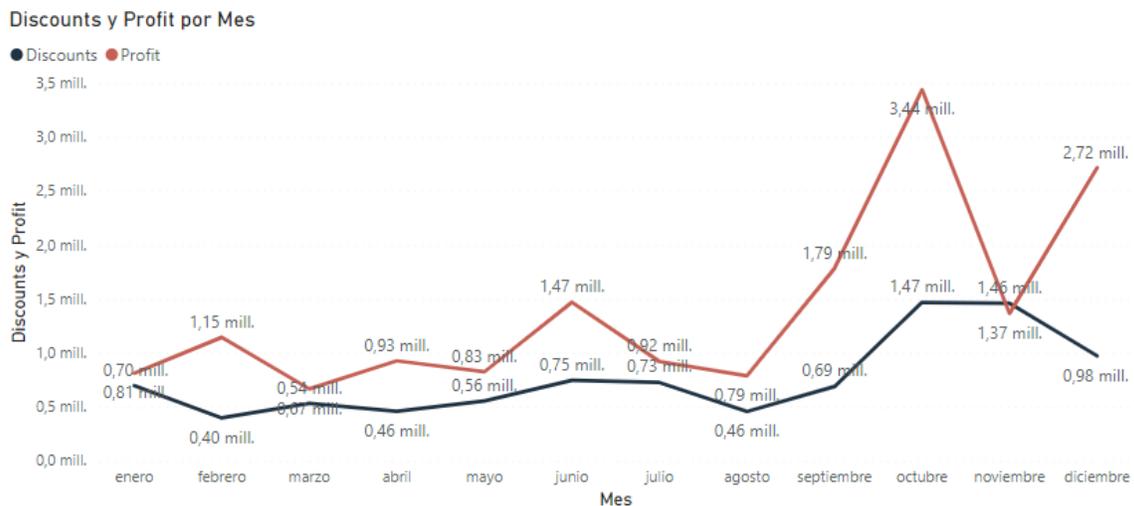


Figura 14: PowerBI – Gráfico lineal con temporalidad

Además de los las visualizaciones básicas, también se pueden importar desde un archivo propio (construido a través de la librería pbiviz) o descargar desde la Marketplace. Para poder utilizar los gráficos que la comunidad ha realizado debemos clicar en los tres puntos que aparecen en la pestaña de Visualización (Figura 15).

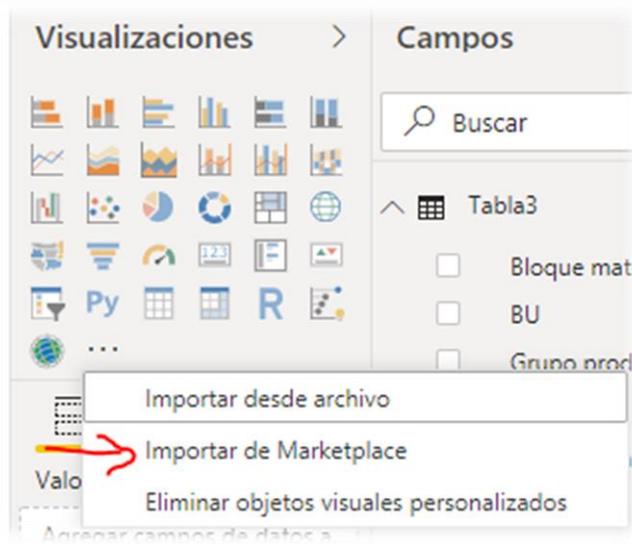


Figura 15: PowerBI – Importar visualizaciones

Dentro de la Marketplace podemos encontrar gráficos personalizados generados por la comunidad de Power BI, entre ellos encontramos uno con la siguiente referencia:

Nombre:	Infographic Designer 1.8.7
Id.:	PBI_CV_73744D90_4DC9_4F18_8BA5_EE8FA5C98035
Versión:	1.8.700
Origen:	AppSource.
Soporte técnico:	http://www.microsoft.com/en-

Con este gráfico podemos transformar las barras básicas en iconos que representen la información, siendo la representación de estos muy configurable. También tiene la posibilidad de subir nuestros propios iconos. Por ejemplo, la figura 16 muestra la salida en el caso de estar seleccionados Sales y Country y de que utilizamos el icono de la apilación de monedas.



Figura 16: Power BI – Infographic Designer

También existen en la store otros gráficos útiles para observar comparaciones en dos dimensiones. En el siguiente gráfico las categorías se reparten por los vértices de un polígono regular del orden de los tipos de la categoría. Podremos colocar diferentes campos (serán dimensiones) que se sobrepondrán, pudiendo comparar para cada categoría tanto cada uno de los campos como entre ellas. Este gráfico tiene como referencia:

Nombre:	Radar Chart 2.0.2
Id.:	RadarChart1446119667547
Versión:	2.0.2
Origen:	AppSource.
Soporte técnico:	https://community.powerbi.com

El resultado del gráfico podemos observarlo en la figura 17, utilizando la categoría de tipos de producto y como dimensiones los descuentos y el beneficio.

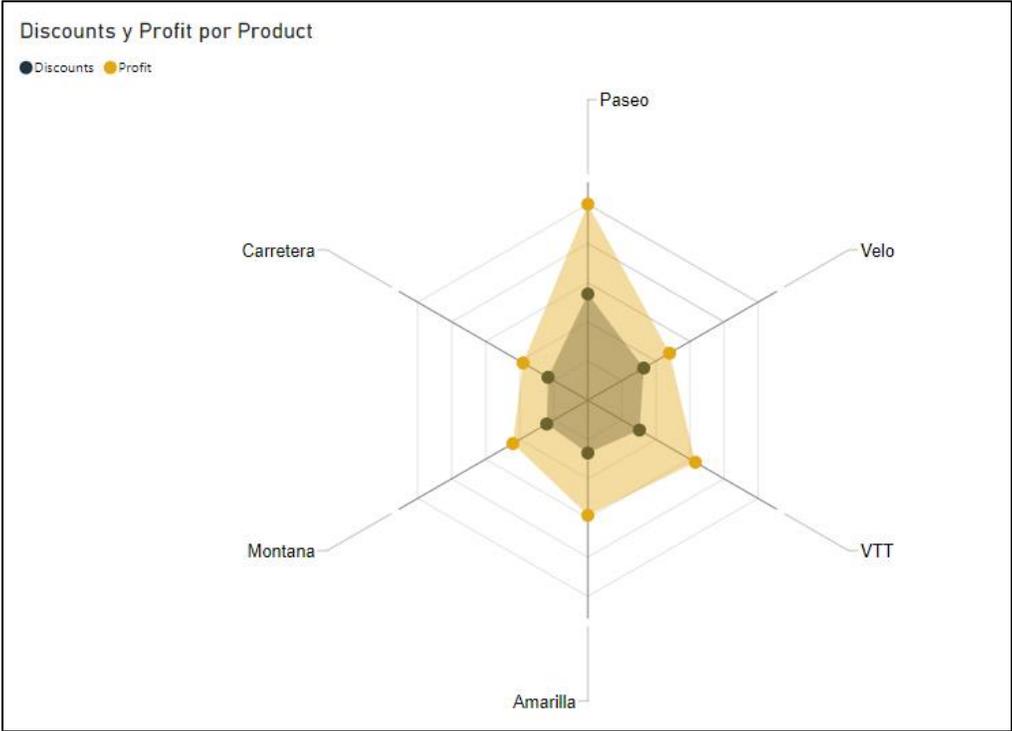


Figura 17: Power BI – Radar Chart

3.2 Data Analysis Expressions (DAX).

DAX son las siglas de Data Analysis Expressions, se trata del lenguaje que permite calcular nuevos datos o modificar comportamientos en elemento de Power BI.

DAX trata de un lenguaje funcional, es decir, su entorno de trabajo será fundamentalmente la creación de funciones, evitando datos mutables o compartir estado entre funciones. Las funciones serán el actor principal y podrán tanto ser asignadas a variables como ser input u output de otras funciones ^[18].

La lógica del funcionamiento de DAX es: una función que realiza un cálculo dependiendo de sus argumentos. El flujo normal de este lenguaje es el uso de funciones que en sus argumentos llaman a otras funciones hasta alcanzar aquélla que no realiza una llamada a otra función. Después el resultado de las funciones que son tratadas como argumento se irán resolviendo para, finalmente, la primera función resolverse y devolver su resultado.

DAX es un lenguaje tipado y basado en tablas, las tablas se dividen en columnas y cada una de estas tiene un tipo de datos bien determinado.

Lenguaje tipado

Existen dos grandes bloques en sus tipos:

- Los números, que incluyen: Enteros (Integer), decimales (Decimal), dinero (Currency), fecha y hora (DateTime) y booleanos (TRUE / FALSE)
- Otros, que incluyen las cadenas de texto (String) y objetos binarios (BinaryObjects)

Con el tipado, sus funciones base son de dos tipos: aquéllas que funcionan para cualquier número y aquéllas que funcionan para cualquier valor. Si una función es del primer tipo, funciona para todos los tipos de número, incluyendo un booleano pero nunca una cadena.

Se permite la sobrecarga de operadores, la conversión sucede cuando es necesario. Si tomamos dos cadenas con un número dentro: "5" + "4" = 9 → transforma ambos dígitos en un entero que es la suma de ellos al pasarlos de string a número.

Los valores temporales de DAX funcionan con enteros, comprender esto hará más sencillo realizar cálculos sobre ellos. Así, las fechas (DATE) se representan como un entero. Este entero apunta a una fecha que es resultado de la suma en días de éste al día 30/12/1899. Por su parte, el formato de tiempo (TIME) utiliza una conversión en la que la 0 representa el inicio del día: 12AM. Y a partir de aquí se calcula respecto a la unidad el valor transpolado de cada una de las horas. La 1AM, como 1/24 del día, 0.04 y las 12PM, como medio del día, 0.5.

Columnas calculadas, tablas calculadas y medidas

Las funciones de DAX tienen dos outputs fundamentales o bien columnas calculadas o bien medidas calculadas. Las primeras hacen referencia a las columnas que serán mostradas como modelo de los datos. Las medidas se calculan sobre la marcha al ser aplicadas sobre las visualizaciones ^[19].

Podemos crear una nueva columna, para ello iremos a la ventana de datos del proyecto de Power BI seleccionaremos en modelado --> Nueva columna. En mi caso crearé la columna "PorBeneficio" que es el porcentaje de beneficio respecto al total de venta de un producto y cliente. En la figura 18 se muestra cómo se realizaría ésta.

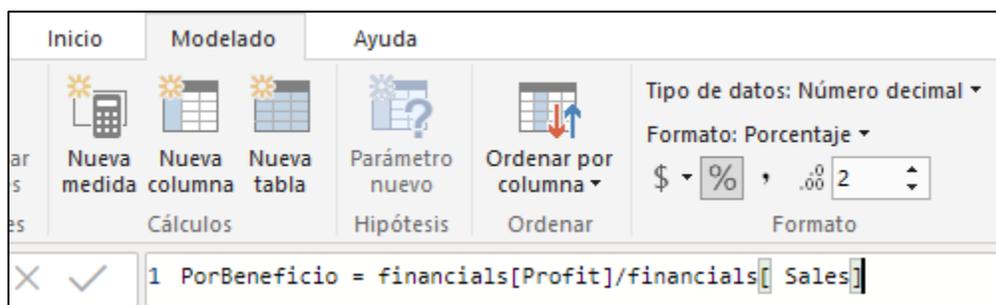


Figura 18: DAX – Creación de columna

La nueva columna tendrá un formato decimal con muchos decimales, para arreglar esto, le damos el Formato porcentaje y limitamos a 2 el total de decimales.

Las columnas calculadas se obtienen a través del cálculo fila a fila pero no nos permiten realizar cálculos sobre el total de elementos. Es decir si quisiéramos para cada entrada de la fila conocer qué porcentaje de ventas respecto del total presenta, necesitaríamos una medida. Para obtener el cálculo que se ha enunciado deberemos crear una nueva medida y hacer:

```
MargenTotalVentas = SUM (financials [Sales]) /  
SUMX (ALL (financials); financials [Sales])
```

La sentencia anterior se traduce a: para cada fila existente muéstrame sus ventas y divídelas por la suma de las ventas de toda la tabla, independientemente de los filtros aplicados. La lógica de la sentencia se estudiará en el siguiente apartado.

La medida aparecerá en la pestaña de Campos pero no aparecerá en la tabla, para poder probarla deberemos ir a la ventana de informes. Para ello se seleccionan Sales, Country y MargenTotalVentas obteniendo la salida mostrada en la figura 19.

Country	Sales	MargenTotalVentas
Canada	24.887.654,88	0,21
France	24.354.172,28	0,21
Germany	23.505.340,82	0,20
Mexico	20.949.352,11	0,18
United States of America	25.029.830,17	0,21
Total	118.726.350,26	1,00

Figura 19: DAX – Medida en Gráfico

Las medidas se recalculan dependiendo de las columnas que están en el seleccionadas en un momento específico. Es decir, si seleccionamos ahora “Segment” se alargará la tabla para dar nuevas combinaciones. Veremos también que la columna de “Sales” se recalculará con los nuevos datos, tal y como puede verse en la figura 20a ^[20].

Segment	Country	Sales	MargenTotalVentas
Government	France	12.127.782,72	0,10
Small Business	United States of America	11.456.559,00	0,10
Government	Germany	11.452.895,94	0,10
Government	Canada	10.741.236,52	0,09
Government	Mexico	9.791.599,38	0,08
Small Business	Canada	9.177.549,00	0,08
Government	United States of America	8.390.746,11	0,07
Small Business	France	7.369.606,50	0,06
Small Business	Germany	7.327.848,00	0,06
Small Business	Mexico	7.096.356,00	0,06
Enterprise	United States of America	4.350.605,00	0,04
Total		118.726.350,26	1,00

Figura 20a: DAX – Cambio de contexto en medida

Dependiendo de la finalidad del cálculo y sobre qué se aplicará, DAX tiene un total de tres tipos de expresiones:

- Columnas calculadas: Son calculadas fila a fila. Se utilizan cuando se quiere recortar o filtrar el valor.
- Tablas calculadas: Del mismo modo que se han realizado las columnas calculadas pueden realizarse tablas completas. Es decir, referenciando cada una de las columnas que la formarán a otras columnas. También existen funciones como CALENDARAUTO que tendrá como salida una tabla.

- **Medidas:** Se calculan mediante concatenación de fórmulas (que permiten la agregación de valores) u otras medidas. Son utilizadas por ejemplo para el cálculo de porcentajes, el radio o agregaciones complejas.

Funciones generales de DAX

Tanto al utilizar medidas como al generar columnas o tablas, toda la lógica de éstas se expresa mediante funciones. Una función puede ser argumento para otra función.

Algunas de las funciones de DAX tienen al menos siempre dos formatos, uno que actúa a nivel de la fila y otro (Agregadores) que funciona como un iterador, permitiendo el trabajo a nivel de tabla. Estos últimos se nombran con el nombre de la función básica (SUM) seguido de una X (SUMX). Siempre utilizan como argumentos: una tabla donde se producirá la acción y una expresión que se evaluará para cada fila de la tabla.

Se presenta a continuación, la clasificación de funciones utilizados por este lenguaje y con explicación de algunas de las funciones más utilizadas. En el apartado siguiente se pondrán en práctica con algunos scripts. DAX cuenta con un total de 9 grupos de funciones ^[21]:

- **Funciones de fecha y tiempo.**
 - Contiene fechas para la creación de calendarios (CALENDARAUTO)
 - Las que permiten la extracción de alguno de los elementos de una fecha específica: YEAR, MONTH, HOUR, MINUTE.
- **Funciones de inteligencia de tiempos.** Son funciones que abstraen el comportamiento temporal. Todas ellas son obtenibles empleando otros conjuntos de funciones sobre las de fecha y tiempo y operadores básicos.
 - DATESINPERIOD(<dates>, <start_date>, <number_of_intervals>, <interval>)
- **Funciones de filtrado.**
 - CALCULATE (<expression>, <filter1>, <filter2>...) → *Mediante esta función sobre la expresión (que será una columna específica) se aplican los filtros indicados a continuación. Un apunte importante es que pese a que cambie el contexto en Power BI, esta expresión mantendrá el filtrado específico en la columna o medida que se exprese.*
 - ALL ([<table> | <column>[, <column>]]) → *Elimina de una tabla completa o de algún conjunto de columnas los filtros de contexto que son aplicados.*
 - FILTER (<table>, <filter>) → *Devuelve tabla con aquellas filas que cumplen la condición del filtrado.*
- **Funciones de información.** Son utilizadas para conocer con resultado booleano alguna situación. Por ejemplo el tipo de valor: ISNUMBER, ISODD, ISLOGICAL. También dan información sobre el dominio o las credenciales: USERNAME ()

- **Funciones lógicas.** Son las necesarias para realizar diferentes flujos dependiendo de sentencias condiciones y también para controlar sus posibles errores. Las expresiones lógicas que se evalúan pueden contener: AND, OR, NOT o IN. DAX cuenta con IF y SWITCH:
 - IF(<logical_test>,<value_if_true>[, <value_if_false>])
 - SWITCH(<expression>, <value>, <result>[, <value>, <result>]...[, <else>])
- **Funciones matemáticas y trigonométricas.** Con nombres intuitivos realizan operaciones básicas de matemática, ejemplos son:
 - Trigonómicas: COS, SIN, ATAN
 - Aproximación: CEILING, ROUND
 - Cálculo: SUM, MOD, PRODUCT
- **Funciones estadísticas.** Con nombres intuitivos realizan acciones de interés para la estadística, ejemplos son:
 - MAX y MIN (<column>)
 - PERCENTILE.EXC(<column>, <k>)
 - COUNT(<column>)
 - RANKX (<table>, <expression>[, <value>[, <order>[, <ties>]]]) → *Solo son necesarios el primero y el segundo argumento. Con esta función se realiza un orden para cada fila de la tabla seleccionada utilizando la expresión del segundo argumento.*
- **Funciones de texto.**
 - LEFT / RIGHT (<text>, <num_chars>) → *Esta función permite extraer del texto que se le pasa como primer argumento tantos caracteres como los indicados por el segundo atributo.*
 - CONCATENATE (<text1>, <text2>)
 - SUBSTITUTE(<text>, <old_text>, <new_text>, <instance_num>)
- **Funciones padre-hijo.** No se han llegado a utilizar. Son funcionales que permiten realizar jerarquía entre los datos de una tabla. Puede ser útil en una tabla de Empleado con una clave primaria para cada empleado y una clave ajena a esta misma tabla que indique la clave del jefe (que a su vez es empleado). A través de estas funciones podemos determinar un camino desde un empleado a su jefe, y a su vez, el jefe de éste.

Ejemplos de aplicación de funciones

Se presentan tres ejemplos de operaciones con DAX para conocer mejor el funcionamiento de este lenguaje. Para cada ejemplo se presenta primero el problema, después se descompone en subproblemas y su solución, para terminar con la solución final y una imagen con el resultado.

Ejemplo 1: Margen total de ventas

Queremos conocer el porcentaje sobre el total de ventas de un subconjunto de la tabla *financials*, presentada en el punto “Columnas calculadas, tablas calculadas y medidas”. Al utilizar la palabra subconjunto se hace referencia a un conjunto de los datos de la tabla donde aparecen cierta cantidad de columnas, no necesariamente todas.

Al utilizar solo una parte de las columnas en POWER BI, estas son agrupadas. De tal manera que si tenemos 200 filas en nuestra tabla pero en la columna producto solo 7 tipos de producto: si solo representamos los productos saldrán solo 7 filas y serán sobre ellas donde se realizarán los cálculos del tipo fila a fila. De la misma manera, solo se actualizarán los valores numéricos por agrupación.

Para obtener el margen de una fila hemos de conocer a cuánto asciende el total de ventas y, después, dividirlo por la cantidad de ventas de una fila específica. A través de la función `SUM (financials [Sales])` obtenemos el total de ventas de toda la tabla

Para realizar el cálculo a nivel de fila necesitamos el agregador de SUM, es decir SUMX: `SUMX (ALL (financials); financials [Sales])`. Con esta última sentencia se obtiene la venta parcial, la específica para cada fila. Véase que SUMX requiere de primero la tabla sobre la que se realizará la suma fila a fila, en este caso se obtiene del *financials* al que no se aplicarán filtros en ningún caso por utilizar la función ALL. La función quedará del siguiente modo:

```
MargenTotalVentas = SUM (financials [Sales]) /  
SUMX (ALL (financials); financials [Sales])
```

El resultado y cómo transformar este valor a porcentaje puede verse en la imagen X, del punto “Columnas calculadas, tablas calculadas y medidas”.

Ejemplo 2: Ranking

Queremos conocer el ranking de más ventas para cada conjunto diferente de producto – país. Es decir, si seleccionáramos también la columna segmento, queremos que en cada campo del ranking siga manteniendo el valor solo para cada producto – valor.

Para extraer un ranking utilizamos la función RANKX que requiere como primer parámetro el subconjunto de una tabla sobre el que se evalúa la posición y en el segundo el que es utilizado para establecer el valor por el que se realizará el ranking.

Como queremos que se evalúen todos los valores independientemente de los filtros existentes utilizaremos ALL. Sin embargo, si el problema es diferente se puede utilizar ALLSELECTED, en este ejemplo podemos ver la sutil diferencia entre ambos. El resultado puede observarse en la figura 20b.

```
Ranking = RANKX( ALL(financials[Product]; financials[Country]);
                CALCULATE(SUM(financials[ Sales])))
```

Product	Sales	Profit	Units Sold	Sale Price	Ranking	Country
Paseo	7.627.731,39	928.651,39	63.282,00	4963	1	Mexico
Paseo	7.611.520,99	1.265.017,99	78.191,50	4780	2	Canada
Paseo	6.944.325,77	1.020.603,27	69.466,50	4141	3	United States of America
Paseo	5.597.751,06	838.748,56	71.606,00	3691	4	France
VTT	5.465.592,64	647.896,64	34.500,00	3948	5	United States of America
Paseo	5.229.814,74	744.416,74	55.693,50	4277	6	Germany
VTT	4.768.754,31	488.808,81	41.248,50	3148	7	Canada
Velo	4.392.907,00	788.789,00	31.050,00	2557	8	Germany
Velo	4.298.828,00	265.401,00	35.761,00	2818	9	United States of America
Amarilla	4.016.427,13	667.867,63	31.603,00	2034	10	France
Velo	3.978.096,24	707.930,24	36.609,50	3143	11	France
Amarilla	3.960.250,26	612.137,26	30.614,50	2270	12	Germany
Amarilla	3.855.765,88	646.861,38	29.232,00	3058	13	Canada
VTT	3.811.193,59	716.371,09	35.774,50	3062	14	France
Montana	3.644.126,80	434.521,80	31.612,50	1915	15	United States of America
Montana	3.566.044,37	559.438,37	28.061,00	2665	16	Germany
Montana	3.527.382,37	461.238,37	31.282,00	2605	17	France
Carretera	3.423.321,90	388.864,90	34.056,00	2045	18	France
Velo	3.329.490,34	370.568,34	32.464,00	2172	19	Canada
VTT	3.293.983,77	605.932,77	31.131,00	2376	20	Germany
VTT	2.172.206,71	575.508,71	26.120,00	2572	21	Mexico
Total	118.726.350,26	16.893.702,26	1.125.806,00	82900	1	

Figura 20b: DAX – Salida de fórmula de Ranking

Ejemplo 3: Marca de tipo de segmento

Queremos identificar cada segmento del tipo ventas para un país y producto, con “Parte superior” si se encuentra por encima de la media de los elementos y “Parte inferior” en el caso contrario. Es decir, queremos saber si cierto elemento de una fila se encuentra por encima de.

Para realizar un cálculo sobre agrupados deberemos crear una nueva tabla. Para esto, se utiliza la función SUMMARIZE que nos permitirá agrupar por una tabla (primer parámetro) y realizar un cálculo específico con esta agrupación (tercer parámetro) dándole un nombre (segundo parámetro) Para nuestro caso, hacemos así:

```
PAIS_VENTAS =
SUMMARIZE(financials; financials[Country];
          "TotalComprado"; SUM(financials[ Sales]))
```

La anterior tabla extrae cada uno de los países existentes y realiza el cálculo de su total de ventas. Ahora, queremos conocer la media aplicada a todos los países, recordemos que no debe estar afectada por estado actual de la aplicación u otros filtros (utilizaremos CALCULATE con el segundo atributo ALL). Para ello hacemos:

```
ValThreshold =
    CALCULATE (AVERAGE (PAIS_VENTAS [TotalComprado]));
    ALL (PAIS_VENTAS [Country]))
```

La última parte es mostrar la etiqueta “Parte superior” si la suma del país que se está consultado está por encima de la media o “Parte inferior” si está por debajo de ésta. Para ello utilizaremos la condición IF. Se utiliza para resolver este ejemplo el formato de variables de DAX, tomando la anterior medida como variable quedaría así:

```
Threshold_ =
    VAR ValThreshold =
    CALCULATE (AVERAGE (PAIS_VENTAS [TotalComprado]);
    ALL (PAIS_VENTAS [Country]))
    VAR Period =
    IF (SUMX (PAIS_VENTAS; PAIS_VENTAS [TotalComprado]) >=
    ValThreshold;
    "Parte superior"; "Parte inferior")
    RETURN Period
```

El resultado puede observarse en la figura 20c.

Country	Threshold_
Canada	Parte superior
France	Parte superior
Germany	Parte inferior
Mexico	Parte inferior
United States of America	Parte superior
Total	Parte superior

Figura 20c: DAX – Salida de fórmula de Marca Segmento

3.3 Paquete de objetos visuales pbiviz

Power BI presenta una amplia variedad de archivos visuales entre aquéllos que vienen predeterminados y aquéllos que pueden descargarse desde su store. Además, también permite a los desarrolladores generar sus propios gráficos mediante una librería específica: **pbiviz**. El acceso a esta librería se consigue mediante `node.js`.

Para realizar este apartado se ha utilizado la versión 1.3.12 (La más actual a 22/04/20). Para comenzar un nuevo proyecto es necesario descargar las herramientas de `powerbi-visuals` así como crear un nuevo esqueleto con la orden `pbiviz new myOwnVisual`.

La forma de trabajar al desarrollar con este paquete es: realizar modificaciones en el código y visualizar cómo se muestra a través de Power BI web. Esto se debe a que cuando el código se compila (a través de la orden `pbiviz start`) se muestra el contenido del visual a través de `localhost`; este contenido es recogido por el Power BI web.

Cualquier modificación en el código en relación con el formato del visual (`visual.ts`) no requiere de la recompilación, sino que se actualiza directamente, pudiendo ver las modificaciones en Power BI web de inmediato. Podemos ver en la figura 21 el elemento gráfico que permite ver el visual que estamos construyendo, la figura 22 el resultado de poner este en el informe y la figura 23 una vez se da a la opción de play, estando utilizando el puerto nuestro visual. Para la figura 23 se muestra el gráfico `ciclecard`, utilizado para introducirse a esta biblioteca.



Figura 21: DAX – Cambio de contexto en medida

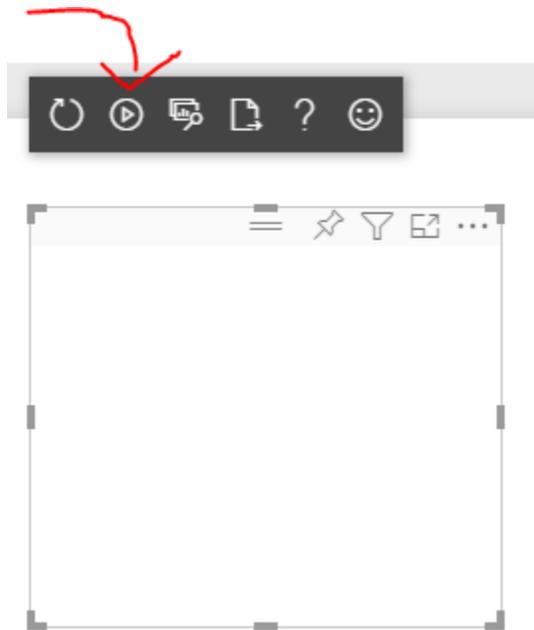


Figura 22: DAX – Cambio de contexto en medida



Figura 23: DAX – Cambio de contexto en medida

Cuando se encuentre terminado el gráficos será el momento en el que podremos ejecutar la orden *pbviz package* que tendrá como resultado la salida de un fichero .pbviz. Este archivo se trata realmente de un .zip con toda la información necesaria para que el gráfico se transforme en un archivo entendible por Power BI Desktop.

Esqueleto de un proyecto pbiviz

Al utilizar la orden: “pbiviz new myOwnVisual” se crea un esqueleto compartido por todos los objetos visuales pbiviz. A continuación se muestra cada uno de los directorios y archivos que existen en este junto a su función en él.

assets/

Se compilan todos los iconos e imágenes que requiere el objeto visual.

dist/

Al utilizar el comando pbiviz package se genera aquí el objeto visual en formato “.pbiviz”. Éste ya es funcional para ser utilizado por Power BI Desktop.

src/

El código typescript del visual

visual.ts

Tiene el código fuente del objeto visual

settings.ts

Define las propiedades con las que cuenta el archivo

style/

Estilos de los elementos del visual definidos con archivos less

visual.less

Archivo de estilo del visual

.gitignore

Especifica a Git qué archivos debe ignorar cuando lleve el código de local a remoto.

capabilities.json

Es utilizado para definir qué capacidades presenta el visual. Esto es: proporcionan la información al servidor sobre el objeto visual creado, aquéllas que están en el modelo como opciones. Por una parte, de qué tipo se permitirán los datos (dataRoles). Por otra, modificaciones de estilo que podrán llevarse a cabo dentro del gráfico, ejemplos son el color de visual o el tamaño de sus líneas (objects)

package.json

Utilizado por npm para administrar los módulos de los que depende el proyecto

pbiviz.json

Metadatos del archivo

tsconfig.json

Archivo de configuración para TypeScript

Visuals

Uno de los ejemplos recomendados para comenzar a conocer cómo funciona esta librería es el ciclec card ^[22]. Además de éste también se consiguieron montar algunos más ^[23] e ir mejorando la comprensión de estos archivos. Tras haber probado diferentes configuraciones se recoge lo siguiente:

- Se utilizan elementos del **lenguaje SVG**. Este es el lenguaje encargado de mostrar gráficos basados en vectores escalables. Tienen la característica de nunca perder la resolución. Es necesario conocer este lenguaje para poder dibujar las líneas de los objetos visuales.
- Se utiliza la librería d3 (**Data-Driven Documents**). Ésta permite hacer dinámico el comportamiento de los gráficos, en nuestro caso ante: los cambios de contexto de Power BI o al añadir valores a alguno de los campos definidos en el capabilities

4. Análisis del sistema

En este apartado se realiza una descripción del sistema a realizar. Con este fin, se compila y refina la información obtenida en las entrevistas (necesidades del usuario) y se descompone el sistema para conocer mejor cada componente y cómo se espera que éstos interactúen con el resto. La información que se muestra hace referencia a qué debe hacer la aplicación, mientras que el cómo conseguirá esto se desarrollará en el apartado de Diseño.

En la primera parte se recoge cómo se han realizado las entrevistas y la documentación extraída de éstas. Esto es: las metas, perspectivas y requisitos. En la segunda, se compilan las necesidades que debe cubrir el sistema y, también, las mejoras que se pueden hacer sobre el producto mínimo viable. En la tercera, se realiza una modulación del sistema, transformando éste en un conjunto de problemas pequeños que irán resolviéndose en el diseño ^{[[19]}.

4.1 Entrevistas a usuario

Para realizar este proyecto se han realizado diferentes entrevistas que han permitido perfilar el producto. Las entrevistas a usuario se trata de un ítem de la ingeniería de software que se realiza en las etapas tempranas de éste. Forma parte de las técnicas de Elicitación. En ellas se realizan los siguientes pasos: identificar interesados, preparación de la entrevista, entrevista en sí y documentación de los resultados ^[24]. A través de ellas se obtiene información del dominio del problema y se identifican los objetivos así como los requisitos de datos y de funcionalidad.

A lo largo del proyecto a través de éstas se han actualizado y mejorado los requisitos del sistema, así como las diferentes etapas del diseño. En este apartado se hace una descripción de cómo se han realizado estas entrevistas y qué temas se han abordado en cada una de ellas.

Las entrevistas pueden dividirse en tres partes bien diferenciadas: Preparación, realización y análisis (realizado posteriormente). Al desarrollar la preparación de la entrevistas se debe determinar el objetivo y también una idea del contenido que se desarrollará, contando con una buena batería de preguntas. Del mismo modo debe conocerse de antemano el momento y el entorno donde se desarrollará.

Se ha de iniciar la entrevista expresando qué se espera conseguir, la razón o la mecánica de preguntas. Además, la aplicación deberá estar abierta mientras se realiza la entrevista. Pese a que debe guiarse el formato del desarrollo de la entrevista, nunca debe influirse sobre la opinión del entrevistado. Por esta razón un buen desarrollo de la entrevista es aquél en el que el entrevistador ocupa un 20% de la conversación y el cliente un 80% ^[25].

Al terminar la entrevista es importante también recapitular sobre todo lo desarrollado en la misma y citar para la próxima reunión. Como paso posterior a la entrevista se debe recoger todo lo que se ha expresado en la propia tanto para entender mejor el producto como para mejorar como entrevistador. Este último punto se trata del análisis de cada entrevista y se presenta a continuación.

Entrevista 01 – Introducción al proyecto

Día: 11/03/2020 Duración: 45 minutos Lugar: Sala de Reuniones

Objetivo: Comprender mejor necesidades del sistema

Contenido y resultados:

En esta primera entrevista, se conocen a los clientes y también las bases sobre las que se realizará el producto. Se explican los principios básicos sobre los que trabaja el Power BI y los sistemas internos sobre los que deberá conectarse.

También se dan pinceladas sobre otros conceptos que estarán presentes durante la realización de proyecto, tales son: DAX, BW, conexión de Power BI con web utilizando indexmudi.com o inetivig.com, con query o mediante una tabla EXCEL

Entrevista 02 – Perfilando el proyecto

Día: 04/03/2020 Duración: 1 hora Lugar: Sala de IT

Objetivo: Conocer cuál es el proyecto a desarrollar

Contenido:

Se presenta un primer informe completo con parte de las necesidades del tablero final. Tras el primer contacto en la entrevista 01, se consideró importante como preparación de la entrevista 02 conocer la terminología base del producto así se recompilo ^[26]:

Concepto	Significado
Agregación	Reducción de las filas del origen para ajustarse a un modelo. Cuando se realiza una agregación sobre una fila específica es un proceso equivalente al del GROUP BY de SQL.
Dashboard	Objetos visuales donde se muestra el estado y las métricas.
Medida	Un campo cuantitativo (numérico) que se puede usar para realizar cálculos. Los cálculos comunes son los de suma, promedio y mínimo.
Seguridad de nivel de fila	RLS La característica de Power BI que permite a los administradores de bases de datos controlar el acceso a las filas de una tabla de base de datos según las características del usuario que ejecuta una consulta (por ejemplo, la pertenencia a grupos)

En la segunda entrevista se muestra un tablero con objetos visuales de la store y lógicas que podrían servir para el tablero final. Mientras se muestra se realizan preguntas sobre las bases del sistema a desarrollar.

Resultados de las preguntas se obtiene:

- Se cierra el significado de rotación de inventario
- Un ejemplo del comportamiento de datos a través de un fichero EXCEL, se entiende el proceso por el que tienen que pasar los datos temporales y de cantidades de salida para que pueda extraerse la unidad de rotación.
- Se aprende la diferencia entre el calendario fiscal y el natural
- Gestión de parte de las query de dónde provendrán los datos que se mostrarán a nivel interno de UBE.

Entrevista 03 – Cerrando lógica tablero y perfilando vista

Día: 25/03/2020 Duración: 40 minutos Lugar: Telemáticamente

Objetivo: Cerrar lógica del sistema principal

Contenido:

La tercera reunión se realiza de forma telemática y se muestran los avances que se han realizado así como una posible configuración de objeto visual. Se preguntan sobre los formatos que deben seguir los datos y las vistas para mejorar los presentados.

Resultados de las preguntas se obtiene:

- Lógica de presentación de unidad de rotación como una previsión. Se plantea como mejora que pueda hacerse en periodos diferentes para llevar a un gráfico una tendencia ajustada.
- Se comprende cómo funciona la lógica del lote. Solo participa en relación con formato uno a muchos con el material.
- Comienza el proceso de creación para un conjunto de queries específica que permitirá la obtención de todos los datos necesarios para el tablero.
- Comprensión de lógicas especiales de algunos materiales que modifica en qué pestaña deben mostrarse.

Entrevista 04 – Muestra del prototipo I & feedback

Día: 30/04/2020

Duración: 1 hora

Lugar: Telemáticamente

Objetivo: Obtener feedback de prototipo I y cerrar lógica de subsistema gráfico

Contenido:

En la cuarta entrevista se muestra el primer prototipo con la funcionalidad de datos demandada, incluyendo los cálculos (en Power BI, a través de medidas) y un primer diseño de cómo se verá cada información dentro del sistema. También se muestran el análisis realizado sobre el sistema, cómo se han agrupado los datos.

Se da feedback sobre cómo podría mejorarse la presentación y también se hace algún inciso sobre algún cálculo que puede realizarse para mejorar la funcionalidad de la aplicación.

Resultados de las preguntas se obtiene:

- Se cierra la lógica de subsistema gráfico
- Mejoras de estilo de presentación

Entrevista 05 – Muestra del prototipo II & feedback

Día: 22/05/2020

Duración: 1 hora

Lugar: Telemáticamente

Objetivo: Obtener feedback de prototipo II y cerrar producto

Contenido:

En la quinta entrevista se validan los ítems del feedback del prototipo II, se dan los últimos retoques de estilo junto al cliente. Se consigue de esta manera, obtener el producto final.

También se informa de que, tras un tiempo documentando la biblioteca pbiviz, el avance es más lento al estimado inicialmente. Se informa de que es posible que no se llegue a su implementación.

Resultado obtenido:

- Se cierra el producto principal

4.2 Requisitos del sistema principal

En este apartado se analizan los datos y lógica el Tablero de rotación, el sistema principal del proyecto. Primero se presentará, explicando su formato, los atributos que deben aparecer o que son entrada para realizar cálculos. Después se describirán cada uno de los atributos, incluyendo aquellos que son calculados (extraídos indirectamente del modelo). Por último, se realiza una agrupación que permitirá entender mejor en que espacio se encuentra cada uno de los atributos.

El tablero, construido sobre un archivo .pbix debe estar formado por tres páginas, todas ellas referentes a la rotación de inventario pero dirigidos a tres diferentes bloques de materias: Productos acabados, Embalajes y Materias primas y aditivos. Teniendo en cuenta que hay productos que podrán pertenecer al primero y al tercer grupo al mismo tiempo.

Para cada uno de las páginas se pide en cada material existente poder verlo con la información básica de éste y la información relacionada con el stock, esto es:

- Nº de material, se le denominará id material
- Nombre material
- Stock. Cantidad de existencias actuales en los diferentes almacenes. Éste debe permitir conocer en el/ los almacenes que están. Diferenciación entre interno y externo. Marcando qué es consigna.
- Ventas y consumos internos (solo consumos internos en el caso de Embalajes y Materias primas y aditivos) con dos valores fundamentales: cantidad y fecha. Debe permitirse discriminar qué es venta y qué consumo interno siguiendo las siguientes franjas:
 - A nivel anual, semestral y trimestral
 - A nivel mensual para cada uno de los periodos anteriores.
 - Rotación (será tiempo que tarda en salir una materia específica)
- Por lote
- BU
- Grupo de productos

El número y el nombre del material identifican de forma única el material, siendo en el segundo donde se encuentra también el formato del material: tipo de embalaje, forma de acabado... El stock contiene tanto lo que se encuentra físicamente en los almacenes como aquello que se encuentra en consigna.

Las ventas y consumos internos que forman parte de la salida de un material y el stock son los elementos clave para obtener el cálculo de la rotación. Son requisitos del sistema:

- Poder saber qué cantidad es salida. Discriminando por venta, consumo interno y otros tipos de salida.
- Conocer, dada una fecha (en un inicio se considera la fecha en la cual se revisa el tablero) qué cantidad ha salido a nivel de un año, seis meses y tres meses.
- Para cada uno de los bloques de tiempo anteriores una aproximación a nivel de mes
- La rotación propiamente dicha que es, en los márgenes de 12, 6 y 3 meses cuánto tiempo (en días) ha tardado en salir un producto determinado. Realizando cálculos

temporales, podremos calcular una tendencia y, con ella, al tener en cuenta el stock, una previsión.

Se hará alusión a producto de manera equivalente a material, cuando se especifique artículo significará un lote específico de un material. El lote de un material es un de un artículo específico que corresponde a un material realizado en cierta fecha y condiciones.

El BU hace referencia a la unidad de negocio (del inglés: business unit), se trata de un fragmento o elemento de una compañía (como contabilidad, ventas...) que identifica una tarea de negocio, en un lugar definido del organigrama y bajo la dirección de un gerente ^[27]. Tanto el BU como el grupo de producto son atributos asociados a cada material.

Con todo lo anterior, podemos asociar en cinco grupos la información que requiere ser mostrada en el tablero. Cada uno de los datos se extrae de un origen diferente, por lo que todos ellos necesitan una referencia otro objeto que los conecte con el resto de tablas. En el diseño se desarrollará la relación entre las tablas.

- **Identificación de material:** id de material, nombre de material, lote, Business Unit y grupo del producto. Todos los artículos tendrán los tres primeros y tendrán la posibilidad de ser filtrados por los últimos dos que serán opcionales (algunos productos tendrán otros no). Además, si filtramos por el nombre de material o el nº de material (es equivalente) se mostrarán los diferentes lotes existentes.
- **Identificadores de lote:** El lote funciona con una relación uno a muchos. Es decir, para cada materia habrá 0 o n lotes; mientras que un lote tiene uno y solo un material asociado a él.
- **Identificadores de salida:** id de material, cantidad, fecha de salida y tipo de salida (ventas, consumos internos y otros). Los tres primeros atributos se utilizarán para calcular la rotación. Los tipos de salida identifican cada uno de los formatos por los que sale un material de almacén y deben poder ser utilizados para filtrar.
- **Identificadores para cada stock:** id de material, stock, porción de consigna, almacén y tipo de almacén. Tipo de almacén depende por transición de almacén y será externo o interno. La consigna se representará como una cantidad específica del stock. El almacén identifica cada stock y debe poder ser utilizado para filtrarlos.
- **Calculadores de rotación:** Stock total, salidas (en cantidad y fecha) y fecha sobre la que se realiza cálculo. El stock total será la cantidad real de cierto producto en suma de todos los almacenes que lo almacenan. Las salidas son la suma de: ventas, consumos internos y otros formatos por los que un producto deja de estar en el almacén. El stock y las salidas junto a una fecha serán suficientes para obtener la rotación total de un producto en cierto margen de tiempo.

Mejoras del sistema

[1] Se pretende conseguir un Dashboard lo más funcional posible, esto significa: que siga un buen conjunto de métricas (conocidas como KPI), que se trate de un gráfico limpio que permita acceder de un vistazo a toda la información relevante para el usuario.

[2] Mejorar tiempos de obtención (no solo para el propio día) o la cantidad de meses implicados en hacer el cálculo de la rotación.

4.3 Requisitos del subsistema gráfico

Como objetivo secundario del proyecto, se marca la creación de gráficos. En especial, aquél que en el eje de las abscisas presenta conjuntos de clientes, agrupados: Grupo1, Grupo2... Y en el eje de las ordenadas representa el porcentaje de ventas acumulado, diferenciando el total realizado por el grupo de clientes. A este gráfico se le da el nombre de “Porcentaje de ventas por grupos de clientes”.

Para cada grupo de clientes se espera una curva desde el punto final del grupo anterior (en el grupo 1, el eje de coordenadas) hasta el punto que marca la suma de lo acumulado y lo que aporta el grupo específico. Se espera que quede marcado (con una línea de puntos por ejemplo) la diferencia entre aquello que es acumulado de los grupos anteriores y lo que es del propio grupo.

También se espera que en cada espacio del grupo se muestren la cantidad de clientes que conforman el grupo y si se clicara encima información relevante sobre los clientes específicos que lo forma. En la figura 24 se muestra un borrador donde pueden verse las ideas presentadas, siendo la viñeta de la derecha una aproximación de lo que muestra el Power BI sobre elementos que tienen información extra.



Figura 24: Boceto de Porcentaje de ventas por grupos de clientes

4.4 Modulaci3n del sistema

En este apartado se toma toda la informaci3n desarrollada en el resto de puntos del An3lisis, se sintetiza sobre los que se desarrollar3 el dise1o y este documento. El primer punto hace referencia al documento en s3, se trata de: Realizar una **documentaci3n** que sirva de gu3a para desarrollar un tablero de Power BI utilizando DAX, creando objetos visuales y siguiendo buenas m3tricas.

El resto de los elementos se trata de un desarrollo que se lleva a cabo en el Dise1o, la siguiente fase, estos puntos output del an3lisis son:

1. **Modelizaci3n de datos** y especificaci3n de su comportamiento
2. **DAX: C3lculo temporal**
3. **DAX: C3lculo de medida de rotaci3n**
 - a. Estimaci3n mensual de salida a 12, 6 y 3 meses.
 - b. Obtenci3n de rotaci3n por d3as para 3, 6 y 12.
4. Elecci3n de plan de **presentaci3n** del tablero
 - a. Elecci3n de gr3ficos y formato de filtrados
 - b. Disposici3n de gr3ficos e informaci3n
 - c. Selecci3n de estilo
5. Implementaci3n de **gr3fico** Porcentaje de ventas por grupos de clientes

Cada cierto periodo de tiempo se considera de vital importancia una **revisi3n con cliente** para asegurar que todo se est3 desarrollando seg3n lo planteado.

5. Diseño del sistema

El Diseño de Software es la actividad del ciclo de vida del software en la cual se utilizan los requisitos para producir una descripción de la estructura interna del software que sirva de base para su construcción ^[28].

Todos los datos mostrados en este documento corresponden a unos datos que emulan sobre los que se realiza realmente la aplicación. En el subapartado del Análisis: Requisitos del sistema se mostró qué datos aparecerán y cómo se relacionan entre ellos. En el siguiente apartado se muestra el archivo EXCEL que se ha utilizado para emular estos datos. Al igual que con otros almacenes de datos, Power BI consume las tablas de EXCEL y las muestra como colecciones dentro de él. Estas podrán relacionarse con datos de orígenes diferentes. El resto de los apartados son el resultado de la identificación de los módulos del proyecto y su resolución. Dejando para el siguiente apartado el resultado final.

5.1 Modelo de datos

Para realizar el Tablero es necesario conocer cómo pueden obtenerse los datos reales, en qué formato se encuentran y cómo relacionarlos. Una vez obtenidos los datos es necesario darles el formato adecuado.

Se presenta a continuación las tablas que emulan aquéllas que son proporcionadas por el almacén de datos de UBE. Estas tablas, realizadas mediante EXCEL muestran toda la funcionalidad que se ha desarrollado en el tablero. La figura 25 muestra los datos de identificación de material, la figura 26 los de lote, la figura 27 los de cada para stock y, por último, la figura 28 los identificadores de salida.

Id material	Nombre material	Grupo producto	BU	Bloque materia
1	caprolactama L 200	CPL	Lactama	ACABADO
2	caprolactama L 400	CPL	Lactama	ACABADO
3	caprolactama L 600	CPL	Lactama	ACABADO
4	caprolactama sobre	CPL	Lactama	ACABADO
5	copoliamidas 50kg	CPL	RA4	ACABADO
6	copoliamidas 100Kg	CPL	RA4	ACABADO
7	polibutadieno tanque	FCH	RA4 / 44	ACABADO
8	polibutadieno saco 25	FCH	RA4 / 44	ACABADO
9	polibutadieno saco 50	FCH	RA4 / 44	ACABADO
10	nylon 80% cubo	Nylon	RA3	ACABADO
11	nylon 90% cubo	Nylon	RA3	ACABADO
12	nylon 100% cubo	Nylon	RA4	ACABADO
13	nylon 80% saco	Nylon	RA3	ACABADO
14	nylon 90% saco	Nylon	RA3	ACABADO
15	nylon 100% saco	Nylon	RA4	ACABADO
16	fertilizantes A	Fertilizantes	Lactama	ACABADO

17	fertilizantes B	Fertilizantes	Lactama	ACABADO
18	fertilizantes C	Fertilizantes	Lactama	ACABADO
19	Hexanodiol Granel	CPL	RA3	Mat. Prima
20	caprolactama L 100	CPL	Lactama	Mat. Prima
21	caprolactama L 200	CPL	Lactama	Mat. Prima
22	caprolactama L 350	CPL	Lactama	Mat. Prima
23	Cajas empaquetado	Otros	Lactama	Embalaje
24	Palets	Otros	RA3	Embalaje
25	Tanque	Otros	RA4 / 44	Embalaje

Figura 25: Modelo – Identificación de material

Lote	Id material
CaP1	1
CaP2	1
CaP3	1
CaP4	1
CaP5	1
CaP6	1
CaPD5	2
CaPD6	2
CaPD7	2
CaPD8	2
Copo7	5
Copo8	5
Copo9	5
Copo10	5
Copo11	5
Copo12	5
Copo13	5
Copo14	5
HexGr1	19
HexGr2	19
HexGr3	19
HexGr4	19
HexGr5	19
HexGr6	19
Pal1	24
Pal2	24
Pal3	24
Pal4	24
Pal5	24

Pal6	24
Pal7	24
Pal8	24
Pal9	24
Pal10	24

Figura 26: Modelo – Identificación de lote

id material	stock	Consigna	almacen	tipo_Almacen
1	150	0	AI - 001	interno
1	350	0	AI - 002	interno
2	300	0	AI - 001	interno
2	140	14	AI - 002	interno
2	200	0	AI - 003	externo
3	600	30	AI - 002	interno
3	830	240	AI - 003	externo
4	1820	0	AI - 004	externo
5	520	0	AI - 003	externo
6	340	0	AI - 004	externo
8	310	0	AI - 003	externo
9	100	25	AI - 004	externo
9	1460	0	AI - 003	externo
10	170	30	AI - 002	interno
10	1520	0	AI - 005	interno
11	2190	750	AI - 002	interno
12	1250	0	AI - 002	interno
12	1180	0	AI - 005	interno
13	780	0	AI - 002	interno
14	1620	0	AI - 001	interno
15	1820	182	AI - 002	interno
15	120	0	AI - 005	interno
15	1720	172	AI - 001	interno
16	1400	70	AI - 002	interno
17	1770	0	AI - 003	externo
18	320	0	AI - 003	externo
19	560	0	AI - 003	externo
19	530	0	AI - 004	externo
20	1010	11	AI - 002	interno
20	1170	0	AI - 001	interno
21	3020	0	AI - 002	interno

22	870	0	AI - 002	interno
23	2360	0	AI - 003	externo
24	3170	0	AI - 003	externo
25	2120	0	AI - 003	externo
25	690	0	AI - 003	externo

Figura 27: Modelo – Identificación para cada stock

id material	cantidad	tipo_salida	fecha_salida
1	390	VENTA	01/05/2018
1	410	VENTA	01/06/2018
1	570	VENTA	01/07/2018
1	370	VENTA	01/08/2018
1	380	VENTA	01/09/2018
1	450	VENTA	01/10/2018
1	50	VENTA	01/11/2018
1	390	VENTA	01/12/2018
1	150	VENTA	01/01/2019
1	610	VENTA	01/02/2019
1	440	VENTA	01/03/2019
1	450	VENTA	01/04/2019
1	100	VENTA	01/05/2019
1	190	VENTA	01/06/2019
1	380	VENTA	01/07/2019
1	460	VENTA	01/08/2019
1	480	VENTA	01/09/2019
1	380	VENTA	01/10/2019
1	110	VENTA	01/11/2019
1	380	VENTA	01/12/2019
1	540	VENTA	01/01/2020
1	570	VENTA	01/02/2020
1	630	VENTA	01/03/2020
1	240	VENTA	01/04/2020
1	150	VENTA	01/05/2020
1	440	CONSUMO INT.	01/10/2018
1	230	CONSUMO INT.	01/01/2019
1	340	CONSUMO INT.	01/07/2019
2	430	VENTA	01/05/2018
2	500	VENTA	01/06/2018
2	420	VENTA	01/07/2018
2	180	VENTA	01/08/2018
2	260	VENTA	01/09/2018

2	240	VENTA	01/10/2018
2	500	VENTA	01/11/2018
2	490	VENTA	01/12/2018
2	420	VENTA	01/01/2019
2	300	VENTA	01/02/2019
2	270	VENTA	01/03/2019
2	190	VENTA	01/04/2019
2	320	VENTA	01/05/2019
2	240	VENTA	01/06/2019
2	500	VENTA	01/07/2019
2	250	VENTA	01/08/2019
2	390	VENTA	01/09/2019
2	250	VENTA	01/10/2019
2	420	VENTA	01/11/2019
2	350	VENTA	01/12/2019
2	150	VENTA	01/01/2020
2	420	VENTA	01/02/2020
2	190	VENTA	01/03/2020
2	440	VENTA	01/04/2020
2	250	VENTA	01/05/2020
5	330	VENTA	01/05/2018
5	460	VENTA	01/06/2018
5	150	VENTA	01/07/2018
5	260	VENTA	01/08/2018
5	400	VENTA	01/09/2018
5	190	VENTA	01/10/2018
5	410	VENTA	01/11/2018
5	220	VENTA	01/12/2018
5	410	VENTA	01/01/2019
5	340	VENTA	01/02/2019
5	230	VENTA	01/03/2019
5	250	VENTA	01/04/2019
5	500	VENTA	01/05/2019
5	310	VENTA	01/06/2019
5	490	VENTA	01/07/2019
5	390	VENTA	01/08/2019
5	180	VENTA	01/09/2019
5	170	VENTA	01/10/2019
5	220	VENTA	01/11/2019
5	500	VENTA	01/12/2019
5	160	VENTA	01/01/2020
5	290	VENTA	01/02/2020
5	230	VENTA	01/03/2020

5	300	VENTA	01/04/2020
5	470	VENTA	01/05/2020
5	380	OTROS	01/02/2020
9	1020	VENTA	01/04/2018
9	1500	VENTA	01/05/2018
9	240	VENTA	01/06/2018
9	240	VENTA	01/07/2018
9	1810	VENTA	01/08/2018
9	1440	VENTA	01/09/2018
9	940	VENTA	01/10/2018
9	420	VENTA	01/11/2018
9	800	VENTA	01/12/2018
9	1730	VENTA	01/01/2019
9	460	VENTA	01/02/2019
9	1100	VENTA	01/03/2019
9	1460	VENTA	01/04/2019
9	600	VENTA	01/05/2019
9	1540	VENTA	01/06/2019
9	430	VENTA	01/07/2019
9	530	VENTA	01/08/2019
9	1550	VENTA	01/09/2019
9	820	VENTA	01/10/2019
9	880	VENTA	01/11/2019
9	1780	VENTA	01/12/2019
9	1070	VENTA	01/01/2020
9	430	VENTA	01/02/2020
9	1030	VENTA	01/03/2020
9	1450	VENTA	01/04/2020
9	1250	CONSUMO INT.	01/08/2018
9	650	OTROS	01/01/2019
9	680	CONSUMO INT.	01/10/2019
19	2040	CONSUMO INT.	01/04/2018
19	860	CONSUMO INT.	01/05/2018
19	1520	CONSUMO INT.	01/06/2018
19	1330	CONSUMO INT.	01/07/2018
19	1770	CONSUMO INT.	01/08/2018
19	760	CONSUMO INT.	01/09/2018
19	700	CONSUMO INT.	01/10/2018
19	1720	CONSUMO INT.	01/11/2018
19	270	CONSUMO INT.	01/12/2018
19	1830	CONSUMO INT.	01/01/2019
19	260	CONSUMO INT.	01/02/2019
19	1550	CONSUMO INT.	01/03/2019

19	1850	CONSUMO INT.	01/04/2019
19	1460	CONSUMO INT.	01/05/2019
19	1740	CONSUMO INT.	01/06/2019
19	190	CONSUMO INT.	01/07/2019
19	1060	CONSUMO INT.	01/08/2019
19	1600	CONSUMO INT.	01/09/2019
19	1450	CONSUMO INT.	01/10/2019
19	2130	CONSUMO INT.	01/11/2019
19	1820	CONSUMO INT.	01/12/2019
19	1690	CONSUMO INT.	01/01/2020
19	380	CONSUMO INT.	01/02/2020
19	640	CONSUMO INT.	01/03/2020
19	1850	CONSUMO INT.	01/04/2020
19	680	OTROS	01/09/2019
24	1550	CONSUMO INT.	01/04/2018
24	3140	CONSUMO INT.	01/05/2018
24	2160	CONSUMO INT.	01/06/2018
24	2010	CONSUMO INT.	01/07/2018
24	1120	CONSUMO INT.	01/08/2018
24	1380	CONSUMO INT.	01/09/2018
24	430	CONSUMO INT.	01/10/2018
24	3080	CONSUMO INT.	01/11/2018
24	2640	CONSUMO INT.	01/12/2018
24	3110	CONSUMO INT.	01/01/2019
24	2500	CONSUMO INT.	01/02/2019
24	2510	CONSUMO INT.	01/03/2019
24	980	CONSUMO INT.	01/04/2019
24	850	CONSUMO INT.	01/05/2019
24	3120	CONSUMO INT.	01/06/2019
24	820	CONSUMO INT.	01/07/2019
24	3070	CONSUMO INT.	01/08/2019
24	500	CONSUMO INT.	01/09/2019
24	2760	CONSUMO INT.	01/10/2019
24	2110	CONSUMO INT.	01/11/2019
24	1340	CONSUMO INT.	01/12/2019
24	2840	CONSUMO INT.	01/01/2020
24	1180	CONSUMO INT.	01/02/2020
24	1450	CONSUMO INT.	01/03/2020
24	2960	CONSUMO INT.	01/04/2020
25	1240	CONSUMO INT.	01/04/2018
25	720	CONSUMO INT.	01/05/2018
25	1270	CONSUMO INT.	01/06/2018
25	730	CONSUMO INT.	01/07/2018

25	3080	CONSUMO INT.	01/08/2018
25	2640	CONSUMO INT.	01/09/2018
25	2670	CONSUMO INT.	01/10/2018
25	270	CONSUMO INT.	01/11/2018
25	1200	CONSUMO INT.	01/12/2018
25	1160	CONSUMO INT.	01/01/2019
25	1170	CONSUMO INT.	01/02/2019
25	1020	CONSUMO INT.	01/03/2019
25	1860	CONSUMO INT.	01/04/2019
25	880	CONSUMO INT.	01/05/2019
25	2210	CONSUMO INT.	01/06/2019
25	3090	CONSUMO INT.	01/07/2019
25	1750	CONSUMO INT.	01/08/2019
25	2720	CONSUMO INT.	01/09/2019
25	1230	CONSUMO INT.	01/10/2019
25	1150	CONSUMO INT.	01/11/2019
25	3110	CONSUMO INT.	01/12/2019
25	2700	CONSUMO INT.	01/01/2020
25	2720	CONSUMO INT.	01/02/2020
25	830	CONSUMO INT.	01/03/2020
25	710	CONSUMO INT.	01/04/2020
25	2310	OTROS	01/08/2019

Figura 28: Modelo – Identificación de salidas

5.2 Modulo DAX: Cálculo temporal

La primera parte para poder configurar aquello que se muestra que tiene configuración temporal es crear una tabla más allá de nuestro modelo donde se encuentren todas las fechas que nos son necesarias. Esto se hace a través de la opción “Crear nueva tabla” y la función `CALENDARAUTO()` de DAX. Quedando la sentencia de la siguiente forma:

```
Relative_Calendar = CALENDARAUTO()
```

Sin ninguna configuración adicional nos generará una tabla con su primer día el 1 de Enero de 2018. Siendo referencia temporal suficiente. Cuando tengamos esta tabla creada, deberemos relacionarla con la tabla que contiene información sobre fechas y con ello podremos realizar cálculos temporales. Además, también se ha de tener en cuenta el filtrado global que existe sobre las fechas: que no se muestren más allá de 12 meses desde la fecha en la que se consulta.

Parte de los datos de entrada del Power BI cuentan con un formato de fecha que no sigue el calendario natural, sino un calendario fiscal. Para que ambos calendarios puedan relacionarse y funcionar de manera correcta al realizar todos los cálculos, se ha de tener una función que permita pasar de un formato temporal al otro y, con esto, poder compararlos desde la fecha actual de la manera que se indicó en el párrafo anterior.

Para resolver este problema se utilizará una columna calculada sobre la tabla que utiliza el calendario fiscal. De esta manera podremos, manteniendo la integridad de todos los datos, poder interrelacionarla con el resto de medidas temporales. La columna calculada se obtendrá con:

```
Fiscal Month number = if([Month Num]>4;[Month Num]-4;[Month Num]+10)
```

5.3 Modulo DAX: Cálculo de rotación

La medida de rotación del tablero es cuánto tiempo tardará cierta materia en salir del almacén teniendo en cuenta su stock actual y la tendencia que ha llevado a 12, 6 y 3 desde el día que se realiza el cálculo.

Existen dos cálculos de valor señalados por el cliente, por una parte cuánto ha salido a nivel mensual según el pronóstico a 12, 6 y 3. Para realizar este cálculo se ha de crear una medida para cada uno de los pronósticos.

Con el modulo temporal obtenido del apartado anterior podremos configurar tiempos en nuestro tablero. Para ello, el filtro se establecerá con la función DATESINPERIOD con cuatro atributos: El primero que será el calendario que acabamos de crear, la fecha desde la que realizamos el cálculo, la cantidad de elementos del cuarto argumento en los que queremos estas fechas y, el cuarto, el formato de tiempo (DAY, MONTH, YEAR). Así nuestro filtro temporal quedaría como:

```
DATESINPERIOD(Relative_Calendar[Date]; TODAY();-6;MONTH)
```

Este filtro deberá realizarse sobre el sumatorio de la cantidad de salida. Esto significa utilizar la función SUM sobre cantidad y que esta sea primer argumento de CALCULATE que tendrá como segundo el filtro que acabamos de obtener. Con esto obtendríamos el total en seis meses y al dividirse por 6, una media de estos valores. Quedando la expresión final como:

```
MonthTrend_6M = CALCULATE( SUM('Id salida'[cantidad]);  
DATESINPERIOD(Relative_Calendar[Date]; TODAY();-6;MONTH)) / 6
```

Para el cálculo de la segunda parte, debemos incluir el stock en nuestro cálculo y pasar la medida temporal a días. Este cálculo se consigue con:

```
SUMX('Id stock';( 'Id stock'[stock] )) / [MonthTrend_6M] * 30
```

La suma de cada uno de los stocks de cierto material (la suma de sus diferentes almacenes) dividido por su tendencia mensual a 6 meses nos da un valor de cuánto tardará en salir el material en unidades mes. Este valor al multiplicarse por 30, nos dará el valor de rotación por días.

5.4 Diseño de la interfaz

En este apartado se muestra como se ha realizado el diseño de la interfaz del tablero. Primero se presentan dos tableros existentes de los que se extrajeron ideas, en la segunda parte se muestran los elementos gráficos utilizados y su justificación mediante los KPI (o indicadores).

Revisión de otras presentaciones de tablero

Para elegir cómo se presentará la información de nuestro tablero se toman de referencia tableros ya terminados para tomar ideas que puedas ser de utilidad al realizar el propio. Este primer tablero utiliza un filtrado lateral, seguido de la información clave en dos círculos y después de estos utiliza un objeto visual relacionado con esta información clave.

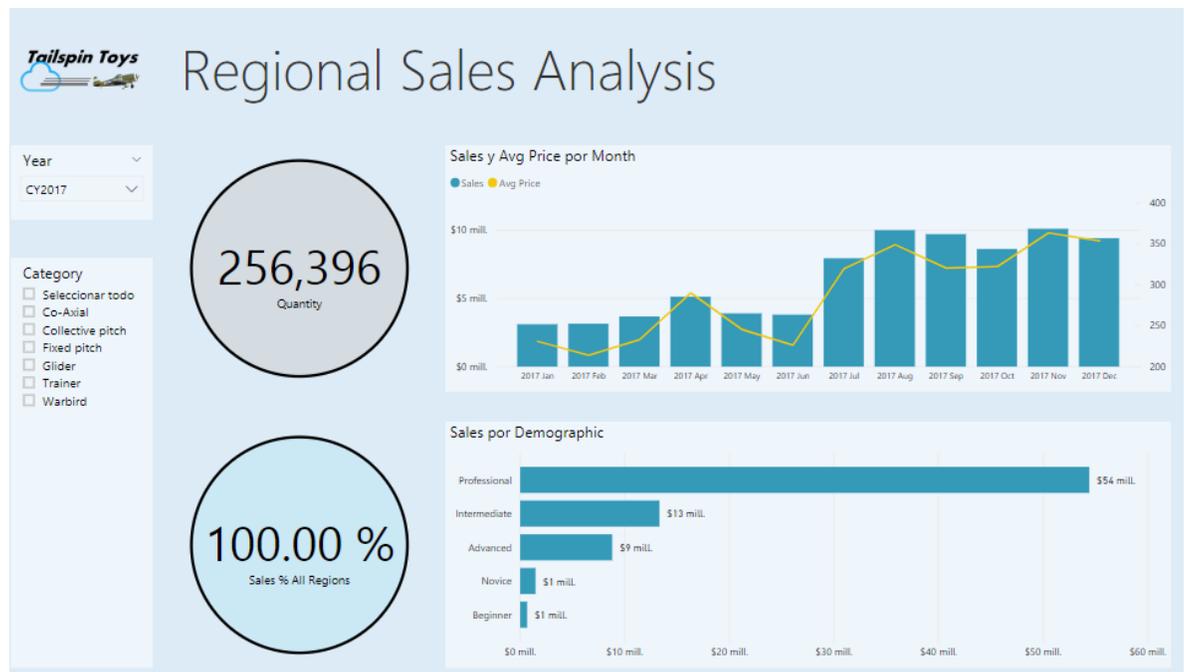


Figura 29: Interfaz – Ejemplo Regional Sales Analysis

El segundo objeto visual hace referencia a un formato de rotación de inventario. En este caso el filtrado es superior, la información clave queda a la izquierda y en la tabla se especifica para cada producto que se encuentra dentro del sistema la información más relevante sobre la rotación de inventario. En ésta también se utilizan barras laterales para mostrar información específica del precio que tiene cada uno de los stocks que todavía no han salido a mercado.

45 mil €

Consumo

35.931 €

Importe de Compras

-4.060 €

Variación de Existencias

34 mil €

Exceso de Stock

Artículo	Unidades consumidas en el Mes	Unidades consumidas por día	Stock Unidades en Almacén	Días de Auto suficiencia con ese stock	Días para entrega de pedidos	Días de Exceso con el Stock Actual	Valoración Económica del Exceso de Stock
BACON EXTRA LONCHEADO EL POZO (KG)	215,00	7,17	162,00	22	2	20	609 €
7UP BIB 10L (60L)	4,90	0,16	9,10	55	3	52	443 €
VOĐKA GENERICO (L)	66,20	2,21	101,60	46	2	44	437 €
CAFE LEOPILIZADO CAFFEPOT NATURAL 500GR	22,00	0,73	27,00	36	2	34	391 €
BRANDY GENERICO (L)	10,40	0,35	79,50	229	2	227	384 €
SALMON FILETE CONG (KG)	20,00	0,67	70,00	105	3	102	372 €
RECORTES DE SALMON AHUMADO	27,00	0,90	45,00	50	3	47	353 €
QUESO BARRA EDAM ALEMAN (KG)	130,93	4,36	113,22	25	2	23	336 €
KIT DENTAL (UND)	509,00	16,97	2.021,00	119	3	116	333 €
FOGONERO FILETE (KG)	25,00	0,83	59,00	70	3	67	318 €
PECHUGA POLLO FILETEADO CONG. (KG)	44,99	1,50	85,10	56	2	54	312 €
DRAMBUIE (L)	0,70	0,02	16,80	720	2	718	306 €
LECHE HOSTELERIA ASTURIANA 1.5L (UND)	609,00	20,30	338,00	16	2	14	276 €
CHURUMBITOS FANTASIA (GOLOSINAS) C/6 (UD	12,00	0,40	60,00	150	1	149	273 €
ZAPATILLAS DUCHA	35,00	1,17	591,00	506	3	503	273 €
LIMON BIB 10L (50L)	4,90	0,16	6,70	41	3	38	271 €
BLUE CURASAO (L)	0,11	0,00	40,84	11.138	2	11.136	265 €
ESTRELLA DAMM BARRIL	58,60	1,95	14,70	7	2	5	264 €
SALMON RODAJA EXTRA (KG)	63,00	2,10	49,00	23	3	20	262 €
GELATINA SABORES KG	38,00	1,27	49,00	38	1	37	246 €
PEPSI LIGHT BIB 10L (55L)	6,60	0,22	5,60	25	3	22	246 €
LICUAFIS (CALGONIT XTRA) AGUAS DURAS 20L	160,00	5,33	80,00	15	3	12	243 €
PERFOCUT (PAPEL FILM) 45 CM X 3 M	4,00	0,13	54,00	405	3	402	228 €
GLASSEX ULTRA BIDÓN 10 KG (KG)	200,00	6,67	100,00	15	3	12	216 €
PIERNA CORDERO CORTADA CONG (kg)	25,00	0,83	40,00	48	2	46	215 €

Figura 30: Interfaz – Ejemplo Rotación de inventario hotelcostcontrol [29]

Elementos y diseño del tablero

Para poder seleccionar los elementos gráficos del tablero de la mejor manera es necesario conocer la naturaleza de nuestros datos. Así, se divide la información en tres tipos de datos: los de filtrado, los de información y los de cálculo. Los de información son valores que se mostrarán directamente, mientras que los de cálculo son los que tiene sentido mostrarlos a través de un gráfico [30].

Otra consideración fundamental es que se han de evitar el uso de colores brillantes o fondos llamativos que desvíen la atención. Así, para este tablero se utilizará el tema “The Honey Bee” que utiliza una gama de grises de base y, para la comparación entre valores, el amarillo.

Para cada uno de los tres bloques del tablero se debe crear un panel, así, se podrá navegar entre los tres paneles para poder conocer los datos de rotación sobre: Productos acabados, Embalajes y Materias primas y aditivos. Para el primero se deberá poder distinguir que es venta y qué es consumo interno. Para los tres también existirá el formato “Otros” que identifica, si existen, formas de salida no contempladas ni en venta ni en consumo.

Así el primer dato de valor es un filtrado global para cada uno de los **bloques de materia**, hacemos que toda la información de cada panel solo haga referencia al tipo específico al que hace referencia.

Por otra parte, tenemos que permitir la **selección de cada materia** y se decide que se haga agrupada cada una de ellas por la supercategoría “Grupo de producto”. Este elemento debe tener formato de filtrado que permita seleccionar entre todos los materiales disponibles. Este elemento

se insertará en la parte izquierda de cada uno de los paneles. Para conseguir lo anterior, se elige el siguiente gráfico:

Nombre:	Hierarchical Filter - xViz
Id.:	HierarchicalFilterF39DAE8D57A743EF89F5C3809DEE2B67
Versión:	1.1.0
Origen:	AppSource.
Soporte técnico:	https://xviz.com/

Para darle la funcionalidad definida se marcarán los valores de *Grupo producto* y *nombre material* dentro de los campos de Id material.

El otro elemento que debe servir como filtrado de los datos es filtrar **cada una de las salidas** por Ventas, Consumos Internos u Otros. Puesto que existen materiales que pueden tener los tres tipos de salida. Este filtrado se posicionará en la parte superior y utilizará el gráfico con la siguiente referencia:

Nombre:	ChicletSlicer 1.6.3
Id.:	ChicletSlicer1448559807354
Versión:	1.6.3
Origen:	AppSource.
Soporte técnico:	https://community.powerbi.com

Los datos de información que se muestran en un tablero deben estar dentro un contexto que permita entender el significado de éstos. Es decir, un tablero no ha de requerir información adicional dentro de éste o en un documento de apoyo. Además, los que son número deben estar acotados y redondeados ^[31].

Por una parte para los datos de información, queremos que pueda mostrarse toda la **información** tanto del **material** específico como de los **lotes** que forman parte de éste. Para esto se utilizará el gráfico de base “Tarjeta de varias filas”. La información del material con cuatro valores (BU, id, grupo y nombre) estará a continuación del filtrado de material y a su derecha tendrá el listado de los lotes existentes para ese material específico.

Por otra parte dentro de los datos de información, queremos que se muestren los **cálculos realizados para la rotación de inventario**. Puesto que este dato es la medida más importante del tablero se considera que el usuario de este no requiere información de su sentido más allá de la diferencia entre: 3, 6 y 12 meses. Estos se acotan antes de utilizarse a dos decimales y se representan

a través del gráfico “Tarjeta”. Existiendo tres de ellos, uno para cada temporalidad existente. Este bloque se insertará a la derecha de los lotes.

Se deshecho que este elemento estuviera dentro de otro tipo de gráfico o tabla tras la muestra del primer prototipo. Donde se marcó que debía ser independiente del resto de visuales.

Los datos de cálculo se dividen en dos grupos: **la salida de material mes a mes** del último año y la cantidad de **stock existente en cada uno de los almacenes**, separando aquello que es consigna. Para el primero se utilizará un gráfico lineal que debe tener el filtrado sobre él de solo mostrar los últimos 12 meses desde hoy. Para el segundo se utilizará el gráfico “de columnas apiladas” que nos permitirá a fecha de hoy conocer cuánto material hay en cada almacén. Separando aquello que realmente tiene con aquello que estuvo en él y es consigna.

5.5 Diseño del gráfico

Para poder representar este gráfico se requiere de un conjunto donde se cada uno de sus elementos contenga: identificador de cliente y su cantidad de ventas (si estas deben acotarse en un tiempo se hará de forma externa). Con estos datos tendremos lo necesario para realizar el siguiente algoritmo:

1. Se crea un mapa donde la clave es cada identificador de cliente y el valor es la cantidad de ventas respecto al total. Se le nombra con `cliente_porcentaje`.
2. Se parametriza el porcentaje mínimo de ventas al que debe llegar un grupo para cerrarse. Es decir, cuando los clientes que forman este grupo sobrepasen este umbral se formará un nuevo grupo.
3. Se inicializa el conjunto: `grupo_actual` y el listado de conjuntos: `list_grupos`
4. Se comienza un bucle para cada uno de los elementos de `cliente_porcentaje`
 - a. Se añade el id al `grupo_actual`
 - b. ¿La suma de los porcentajes de `grupo_actual` es mayor a porcentaje mínimo de ventas?
 - i. Sí → Se añade a `list_grupos` y se vacía la variable `grupo_actual`
 - c. Se toma el siguiente elemento

Pese a que en las entrevistas solo se indicó que fuera de ventas, podría resolver cualquier valor que resumiera en formato de suma como: el de porcentaje de margen de beneficio, cantidad de pedidos...

6. Implementación y pruebas

6.1 Implementación del tablero

Se decide utilizar una temática sencilla, en la que:

- Las fuentes grandes son utilizados para mostrar los valores de la rotación, considerados como los que tienen más expresión en este tablero.
- Las fuentes medianas dan los valores específicos para los atributos de los productos.
- Las fuentes pequeñas se utilizan para los filtrados y las referencias de valores en cada elemento visual.

También se realizan algunas modificaciones de estilo sobre los datos de filtrados para que utilicen un formato más acorde con el resto de la plantilla. En la figura 31 se muestra el producto base, encontraremos la mayor funcionalidad al clicar en uno de los elementos de filtrado.



Figura 31: Interfaz Final – Entrada base para producto acabado

Al seleccionar o bien un artículo de la izquierda o bien uno de los formatos de salida, se producirá un cambio en el contexto del Tablero. En la figura 32 se mostrará una etiqueta de los valores más importantes del producto, seguido de un listado de los lotes que se tienen de éste. Más a la derecha de este se muestran las medidas de rotación. La parte inferior del tablero se utiliza para mostrar el gráfico de las salidas de los últimos 12 meses y el gráfico que muestra cuánto de la materia se encuentra en los diferentes almacenes. Marcando en amarillo aquello que es consigna.



Figura 32: Interfaz Final – Entrada con selección de producto (en pestaña producto acabado)

En la parte inferior del tablero podremos navegar entre los tres paneles existentes que dividen en los tres grandes bloques todos los materiales, puede verse en la figura 33.

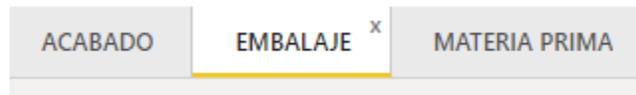


Figura 33: Interfaz Final – Entrada con selección de producto (en pestaña producto acabado)

6.2 Implementación del gráfico

Durante la el desarrollo del gráfico surgieron algunos problemas pero pudieron solucionarse con el tiempo. Al comenzar, justo después de la preparación del entorno surgió problema con la documentación base de Microsoft. Pues las modificaciones a las que esta página hace referencia justo después de crear el paquete pbviz se basan en una versión anterior (2.1) que ya no tiene compatibilidad con la actual (3.1). Una vez encontrado en la comunidad que se trataba de este problema, se encontraron vías alternativas para desarrollar el primer gráfico.

Un error recurrente al lanzar objetos visuales es que uno de los procesos que se lanzaron al lanzar el programa sigue utilizando el puerto abierto aun al cerrar el objeto visual. Esto hace que no se nos permita lanzarlo de nuevo puesto que el puerto está ocupado. La solución es conocer qué proceso sigue manteniendo ese puerto abierto y matarlo.

Pese a mejorar los conocimientos sobre estas herramientas no se ha podido llegar a implementar el gráfico.

7. Conclusión

Nivel profesional y tecnológico

Tras la experiencia de las prácticas se han mejorado diferentes competencias tales como: realizar una correcta entrevista con clientes, importancia de la planificación temporal, crear de documentación para una tecnología nueva o realizar un análisis detallado que permita un diseño coherente.

Las reuniones con clientes se trata de uno de los elementos clave de elicitación en el proceso de ingeniería del software. Se trata de un ítem con alto valor ya que el éxito de un producto es cumplir con las expectativas del cliente. Se ha mejorado esta competencia al entender cómo es su correcta organización y cómo debe guiarse.

La planificación es clave para asegurar invertir nuestro tiempo con eficacia y eficiencia. Al comenzar el proyecto no se conocía la previsión de los ratios de tarea – tiempo para muchos de los hilos de trabajo. Además de conocerlo ahora mejor, se revalora más la correcta planificación.

La documentación de una tecnología permite un punto de partida para comenzar a utilizarla. Pese a que con anterioridad se había creado alguna pequeña documentación, tras el trabajo se aprende la importancia de realizar apartados y qué resulta realmente útil para la documentación. Junto a este ítem, también se aprende cómo se explora una tecnología desconocida: tomar proyectos abiertos, revisar documentación...

El análisis de un sistema es una de las tareas que he podido desarrollar de manera paralela en otro proyecto (Majors a Casa). Aprendiendo a realizar un primer buen detalle de qué es lo que se espera, modelando los datos que deben y consiguiendo entender mejor cómo debe llevarse a cabo un refinamiento continuo. También se interioriza que: el qué corresponde al análisis y cómo (la implementación) al diseño.

Como competencia transversal se dan pasos en cómo sintetizar la información y discriminar aquello que es de interés. Esto es primordial para un buen modelado del sistema o un análisis profundo sobre las necesidades de éste; pero también a nivel personal tiene gran valor.

Nivel personal

Se trata de exponerse a aquello que quiere resolverse. Conocerse y encontrar una fórmula que nos invite a estar ante el problema, aumentando el tiempo en el que estamos frente él, poniendo a nuestro favor las probabilidades y, además, todas sus posibilidades.

Se trata de dejarse apasionar, aumentando las posibilidades de encontrar una solución al primero identificado y los que nacen por convergencia o divergencia. Dando valor a cada paso que da camino al aprendizaje tanto profesional como personal.

8. Bibliografía

- [1] Grupo UBE, "**Empresa – Grupo UBE** "
<https://www.ube.es/ES/index.php> [Consulta a 10 de Marzo del 2020]
- [2] 40defiebre, "**¿Qué es un Dashboard?**"
<https://www.40defiebre.com/que-es/dashboard> [Consulta a 10 de Marzo del 2020]
- [3] Universidad de la plata, Herramienta para Implementar LEL y Escenarios – "**Ingeniería de Requerimientos**" [archivo PDF]
http://sedici.unlp.edu.ar/bitstream/handle/10915/4057/2_-_Ingenier%C3%ADa_de_requerimientos.pdf?sequence=4 [Consulta a 17 de Marzo del 2020]
- [4] Martínez Robalino D.A., "**Metodología para el diseño de Dashboards orientado hacia el registro de evidencias en el proceso de evaluaciones institucionales**" [archivo PDF]
<https://reunir.unir.net/bitstream/handle/123456789/6171/MARTINEZ%20ROBALINO%2C%20DANIEL%20ANDRES.pdf?sequence=1&isAllowed=y> [Consulta: a 28 de Marzo del 2020]
- [5] shopify, "**Rotación de inventario**"
<https://es.shopify.com/enciclopedia/rotacion-de-inventario> [Consulta: a 3 de Abril del 2020]
- [6] Moreira da Cruz L., "**Los 100 riesgos en la gestión de proyectos**"
<https://www.linkedin.com/pulse/los-100-riesgos-en-la-gesti%C3%B3n-de-proyectos-luciano-moreira-da-cruz> [Consulta: a 28 de Marzo del 2020]
- [7] Dwivedi V., "**Top 10 Programming Languages of the World -- 2019 to begin with**"
<https://www.geeksforgeeks.org/top-10-programming-languages-of-the-world-2019-to-begin-with/> [Consulta: a 8 de Abril del 2020]
- [8] Abernethy M., "**Simplemente qué es Node.js**"
<https://www.ibm.com/developerworks/ssa/opensource/library/os-nodejs/index.html>
[Consulta: a 6 de Abril del 2020]
- [9] Less CSS, "**In-Depth Guide**"
<http://lesscss.org/features/> [Consulta a 21 de Marzo del 2020]
- [10] Apuntes asignatura EI1050 [Consulta a 12 de Marzo del 2020]
- [11] Scaricaciottoli R., "**7 tips to know when to use Scrum in your project**"
<https://www.hexacta.com/2018/09/10/7-tips-to-know-when-to-use-scrum-in-your-project/>

- [12] Drake, J.M., “**Análisis de requisitos y especificación de una aplicación**” [archivo PDF] https://www.ctr.unican.es/asignaturas/Ingenieria_Software_4_F/Doc/M3_08_Especificacion-2011.pdf [Consulta: a - de Abril del 2020]
- [13] Choudary A., “**Agile Scrum Tutorial: What is Scrum and how to master it?**” <https://www.edureka.co/blog/agile-scrum-tutorial/> [Consulta a 12 de Marzo del 2020]
- [14] Pesquera C., “**Metodologías ágiles y gestión del cambio**” <https://carlospesquera.com/metodologias-agiles-y-gestion-del-cambio-valor-y-estimacion-historias-usuario-2-de-3/> [Consulta: a 3 de Abril del 2020]
- [16] Wikimedia, “**List of countries by regional classification**” https://meta.wikimedia.org/wiki/List_of_countries_by_regional_classification [Consulta: a 5 de Mayo del 2020]
- [17] Aitana, “**Tipos gráficos Power BI**” <https://blog.aitana.es/2018/09/12/tipos-graficos-power-bi/> [Consulta: a 23 de Abril del 2020]
- [18] García E.I., “**¿Qué es la programación funcional?**” <https://codigofacilito.com/articulos/programacion-funcional> [Consulta: a 12 de Marzo del 2020]
- [19] Microsoft, “**Referencia DAX**” <https://docs.microsoft.com/es-es/dax/> [Consulta: a 25 de Marzo del 2020]
- [20] Russo M. & Ferrari A., “**Row Context and Filter Context in DAX**” <https://www.sqlbi.com/articles/row-context-and-filter-context-in-dax/> [Consulta: a 16 de Abril del 2020]
- [21] InteractiveChaos, “**Funciones DAX**” <https://www.interactivechaos.com/recursos-educativos/funciones-dax> [Consulta: a 27 de Marzo del 2020]
- [22] Microsoft, “**Visuals Circlecarrd**” <https://github.com/Microsoft/PowerBI-visuals-circlecarrd> [Consulta: a 18 de Mayo del 2020]
- [23] Microsoft, “**Linedotchart**” <https://github.com/microsoft/powerbi-visuals-linedotchart> [Consulta: a 20 de Mayo del 2020]
- [24] Grangel R. (2019). Apuntes de la asignatura “EI1030. Anàlisi de Sistemes d’Informació/ EI1032. Anàlisi de Programari” Reyes Grangel Seguer

- [25] Gimeno Artigas, S., “Entrevistas con usuarios en cuatro sencillos pasos”
<https://www.torresburriel.com/weblog/2017/10/16/entrevistas-usuarios-cuatro-sencillos-pasos/>
- [26] mihart, “Glossary for Power BI service”
<https://docs.microsoft.com/es-es/power-bi/consumer/end-user-glossary> [Consulta: a 13 de Marzo del 2020]
- [27] Business Dictionary, “Significado de business unit”
<http://www.businessdictionary.com/definition/business-unit.html> [Consulta: a 23 de Abril del 2020]
- [28] R. Francisco, Universidad Cantabria – “Diseño de Software” [archivo PDF]
<https://www.ctr.unican.es/asignaturas/is1/is1-t04-trans.pdf> [Consulta: a - de Abril del 2020]
- [29] Tapia A., “BI: Gestión de Almacenes”
<https://www.hotelcostcontrol.com/business-intelligence-gestion-almacenes/> [Consulta: a 18 de Mayo del 2020]
- [30] Logicalis, “Tableros de control: indicadores y buenas prácticas”
<https://blog.es.logicalis.com/analytics/tableros-de-control-indicadores-y-buenas-practicas#:~:text=Los%20tableros%20de%20control%20o,minimizando%20la%20probabilidad%20de%20error> [Consulta: a 29 de Mayo del 2020]
- [31] Madero M., “TABLEROS DE CONTROL DE KPIS PARA UNA TOMA DE DECISIONES EFECTIVA”
<https://www.grou.com.mx/blog/tableros-de-control-de-kpis-para-una-toma-de-decisiones-efectiva> [Consulta: a 29 de Mayo del 2020]