



Universidad de Valladolid

ESCUELA DE INGENIERÍA INFORMÁTICA (SG)

Grado en Ingeniería Informática de Servicios y Aplicaciones

MINERÍA DE PROCESOS

Alumno: Ángel Martín Marinas

Tutores: José Ignacio Farrán e Ignacio Rodríguez Coteló

Índice de contenidos

MINERÍA DE PROCESOS.....	1
Alumno: Ángel Martín Marinas.....	1
Tutores: José Ignacio Farrán e Ignacio Rodríguez Cotelo.....	1
Índice de figuras:.....	5
Índice de tablas:.....	7
Resumen.....	8
Abstract.....	9
1. Introducción:.....	10
1.1. Motivación:.....	10
1.2. Objetivos.....	11
1.3. Estructura.....	11
2. Planificación.....	13
2.1. Planificación temporal.....	13
2.2. Presupuesto económico.....	15
2.2.1. Hardware y Software:.....	15
2.2.2. Recursos humanos.....	16
2.2.3. Presupuesto total.....	17
3. Estado del arte.....	18
3.1. Software relacionado.....	19
3.1.1. Celonis:.....	19
3.1.2. UiPath Process Mining.....	20
3.1.3. Aris.....	20
3.1.4. Disco.....	20
3.1.5. ProM.....	21
4. Gestión de procesos.....	22
5. Minería de procesos.....	24
5.1. Tipos de minerías de procesos.....	27
5.1.1. Descubrimiento.....	28
5.1.2. Conformidad.....	28
5.1.3. Mejora.....	28
5.2. Tipos de perspectivas en la minería de procesos.....	29

5.2.1. Perspectiva de control de flujo.....	29
5.2.2. Perspectiva organizativa.....	29
5.2.3. Perspectiva de caso.....	29
5.2.4. Perspectiva temporal.....	29
5.3. Registros de eventos.....	30
5.4. Modelado de procesos.....	32
5.4.1. Redes de Petri.....	32
4.4.2 BPMN.....	35
6. Caso de Uso: Proceso O2C (Order to Cash).....	38
6.1. Introducción.....	38
6.2. Objetivos.....	39
6.3. Herramientas.....	39
6.3.1. Celonis.....	39
6.4. Datos.....	40
6.4.1. Clientes:.....	41
6.4.2. Productos:.....	41
6.4.3. Usuarios:.....	41
6.4.4. Pedidos:.....	42
6.4.5. Bloqueos:.....	43
6.4.6. Devoluciones:.....	43
6.4.7. Historico_Cambios:.....	44
6.4.8. Fichero Python para generar los datos:.....	44
6.5. ETL (Extracción, Transformación y Carga).....	48
6.5.1. Conexión:.....	48
6.5.2. Extracción:.....	53
6.5.3. Transformación:.....	59
6.5.4. Carga:.....	69
6.6. Creación de dashboards.....	76
6.6.1. Executive Dashboard.....	80
6.6.2. Plazos de entrega:.....	86
6.6.3. Devoluciones:.....	90
6.6.4. Facturación tardía:.....	94

6.6.5. Bloqueos:	100
6.6.6. Cambios:	104
6.6.7. Automatización:	108
6.6.8. Benchmark:	111
6.6.9. Social:	112
6.6.10. Conformance:	113
6.6.11. Explorador de variantes:	114
6.6.12. Explorador de Procesos:	114
6.6.13. Explorador de Casos:	115
6.7. Análisis de datos - Insights detectados:	116
6.7.1. Plazos de entrega:	117
6.7.2. Devoluciones:	118
6.7.3. Facturación tardía:	119
6.7.4. Bloqueos:	120
6.7.5. Cambios:	121
6.7.6. Automatización:	121
6.7.7. Benchmark:	122
7. Conclusiones:	123
7.1. Consecución de los objetivos:	123
7.2. Líneas de trabajo futuras:	123
Bibliografía:	125

Índice de figuras:

2.1 Planificación en diagrama de Gantt.....	8
3.1 Logo de Celonis.....	12
3.2 Logo UiPath Porcess Mining.....	13
3.3 Logo Aris.....	13
3.4 Logo Disco.....	14
3.5 Logo ProM.....	14
5.1 Ciclo de vida BPM.....	17
5.2 Relación entre datos y modelos	19
5.3 Tipos de minería de procesos.....	20
5.4 Ejemplo de red de Petri.....	26
5.5 Lógica en redes de Petri.....	27
5.6 Ejecución de una secuencia sobre una red de Petri.....	28
5.7 Notación BPMN.....	29
5.8 Ejemplo de modelo usando BPMN	30
6.1 Esquema de funcionamiento de Celonis.....	33
6.2 Celonis – Menú lateral izquierdo.....	42
6.3 Celonis – Data Pools.....	42
6.4 Celonis – Nuevo data pool.....	43
6.5 Celonis – Connect to data source.....	44
6.6 Celonis – Tipos de conexión.....	45
6.7 Celonis – Upload files.....	47
6.8 Celonis – Ficheros cargados.....	48
6.9 Celonis – Ajustes sobre ficheros.....	48
6.10 Celonis – Fichero Pedidos.csv.....	49
6.11 Celonis – Tipo de datos fichero Pedidos.csv.....	50
6.12 Celonis – Formato DATETIME.....	51
6.13 Celonis – Ficheros cargados y formateados.....	52
6.14 Celonis – Esquema de modelo de datos.....	53
6.15 Celonis – Data Integration.....	54
6.16 Celonis – Data Jobs.....	55
6.17 Celonis – Nuevo data job.....	55
6.18 Celonis – Crear data model.....	63
6.19 Celonis – Nuevo data model.....	64
6.20 Celonis – Selección de tablas.....	65
6.21 Celonis – Tabla de actividades.....	65
6.22 Celonis – Selección del campo CASE_KEY.....	66

6.23 Celonis – Relación de tablas en el data model.....	67
6.24 Celonis – Data Model.....	68
6.25 Celonis – Carga del dara model.....	69
6.26 Celonis – Apartado Studio.....	70
6.27 Celonis – Crear paquete.....	71
6.28 Celonis – Crear análisis.....	72
6.29 Executive Dashboard.....	73
6.30 Dashboard Plazos de entrega.....	80
6.31 Dashboard Devoluciones.....	84
6.32 Dashboard facturación tardía.....	88
6.33 Dashboard Bloqueos.....	94
6.34 Dashboard Cambios.....	98
6.35 Dashboard Automatización.....	102
6.36 Dashboard Benchmark.....	104
6.37 Dashboard Social.....	105
6.38 Dashboard Conformance.....	106
6.39 Dashboard Explorador de variantes.....	107
6.40 Dashboard Explorador de procesos.....	108
6.41 Dashboard Explorador de Casos.....	109
6.39 Dashboard Explorador de variantes.....	107

Índice de tablas:

2.1 Resumen del presupuesto total.....	17
5.1 Ejemplo de registro de eventos.....	32

Resumen

El proyecto consiste en un estudio teórico de las técnicas de minería de procesos, una rama de la minería de datos y big data centrada en los distintos procesos empresariales existentes. Para ello, se combinan técnicas relativas al procesado de datos y posterior análisis de estos. Se incluye un caso de uso de ejemplo en el cual se aplican todas aquellas técnicas estudiadas de forma teórica y se lleva a cabo el proceso de desarrollo completo, haciendo uso de una de las herramientas de minería de procesos más destacadas del mercado. También, comprende el trabajo de análisis y obtención de resultados y conclusiones a partir de los datos analizando proponiendo mejoras en el proceso empresarial del ejemplo.

Palabras clave: Minería de procesos, minería de datos, procesos, Celonis, big data, análisis de datos, dashboarding, inteligencia empresarial.

Abstract

The project consists of a theoretical study of process mining techniques, a branch of data mining and big data focused on the different existing business processes. For this purpose, techniques related to data processing and subsequent data analysis are combined. An example use case is included in which all the techniques studied theoretically are applied and the complete development process is carried out, making use of one of the most prominent process mining tools on the market. It also includes the work of analysis and obtaining results and conclusions from the data analysed, proposing improvements in the business process of the example.

Keywords: Process mining, data mining, processes, Celonis, big data, data analysis, dashboarding, business intelligence.

1. Introducción:

1.1. Motivación:

En la actualidad, debido a la transformación digital de los procesos y la masificación de los dispositivos, se ha generado una enorme cantidad de información que fluye a través de internet y se almacena en diversas bases de datos. En cualquier actividad que llevemos a cabo, se acumulan grandes cantidades de datos que a menudo resultan valiosos por diversas razones. Entre ellas, podemos utilizar estos datos para analizar y comprender el mundo que nos rodea, como pueden ser los distintos procesos que llevamos a cabo en nuestro día a día o los procesos empresariales que se siguen en los distintos sectores.

Una de las principales fuentes de información la proporciona el internet de las cosas, *Internet of Things (IOT)*, así como las redes sociales, la automatización de las fábricas y todos aquellos datos que viajan por internet. Al disponer de esta gran cantidad de datos, ha producido que en la actualidad tenga una gran importancia disciplinas como el Big Data, la cual se encarga de analizar estos datos con el fin de obtener patrones o tendencias que nos puedan proporcionar un conocimiento.

El marco de trabajo en el que vamos a trabajar es la ciencia de datos, en concreto la minería de datos, en la cual, por medio de técnicas computacionales, inteligencia artificial y machine learning podemos extraer información de los datos que obtenemos de los distintos sistemas.

En concreto, en este trabajo vamos a desarrollar la disciplina de la minería de procesos, una variante de la minería de datos. La minería de procesos se basa en la ciencia de procesos, la cual permite estudiar los distintos procesos empresariales que se llevan a cabo y la forma de optimizarlos, automatizarlos, etc con la intención de aumentar la productividad y reducir los costes de los procesos. Otro de los pilares en los que se basa la minería de procesos es la ciencia de datos, en concreto, la minería de datos. Aplicando la técnica de minería de datos sobre los distintos procesos empresariales y apoyándonos de los conocimientos que nos proporciona la ciencia de procesos obtenemos la minería de procesos.

La motivación detrás de este trabajo de investigación radica en el potencial que ofrece la Minería de Procesos para optimizar la eficiencia, identificar cuellos de botella y mejorar la toma de decisiones en una amplia gama de sectores industriales y empresariales. A medida que las empresas y las instituciones acumulan enormes volúmenes de datos sobre sus operaciones, surge la necesidad de desarrollar métodos efectivos para analizar y comprender estos procesos

en profundidad. Al explorar las técnicas de Minería de Procesos, se busca revelar patrones ocultos, tendencias significativas y oportunidades de mejora, lo que a su vez puede catalizar la innovación y el crecimiento.

1.2. Objetivos

El principal objetivo del proyecto es realizar un trabajo de investigación sobre la minería de procesos, dando a conocer en qué consiste esta novedosa técnica y cuáles son los beneficios que proporciona.

Se llevará a cabo un estudio acerca de la minería de procesos, donde se realizará una explicación teórica de en qué consiste la técnica, así como los conceptos que tienen que ver con ella. También, se expondrá de forma breve cómo ha sido la evolución histórica desde los inicios hasta la actualidad. Analizaremos las distintas soluciones software que existen actualmente en el mercado. Por último, para mostrar el potencial que presenta esta técnica haremos un ejemplo con un caso de uso real, basándonos en una simulación de una base de datos haciendo uso del software Celonis.

Cabe destacar, que debido a que el proyecto ha sido realizado en colaboración a la empresa en la que actualmente estoy trabajando el software Celonis se corresponde con un software propietario, y para su uso es necesario disponer de una licencia. Por este motivo no es posible incluir el software en un repositorio al cual se tenga acceso, puesto que sería imprescindible de disponer licencias de uso de la herramienta. El software se encuentra en un servidor web que pertenece a la herramienta Celonis.

A continuación, haremos una enumeración de los principales objetivos que tratarán de cumplir desarrollando dicho trabajo:

- Explicación de en qué consiste la gestión de procesos, en concreto, la minería de procesos.
- Establecer cuáles son los usos de la minería de procesos.
- Identificar la evolución de la minería de procesos.
- Analizar las soluciones software que existen.
- Mostrar un ejemplo de caso de uso real.
- Mostrar las técnicas empleadas para identificar cuellos de botella, oportunidades de mejora.
- Ilustrar cuáles son las transformaciones que se deben de llevar a cabo sobre los datos para aplicar las técnicas de minería de procesos.
- Establecer los distintos pasos que se deben de llevar a cabo para aplicar las técnicas.

1.3. Estructura

La memoria está estructurada principalmente en 7 capítulos, los cuales se detallarán a continuación. Al final de la memoria se encuentran las referencias bibliográficas.

Este primer capítulo en el que nos encontramos se trata de un capítulo introductorio en el que se pone en contexto el tema principal del proyecto. En el segundo de los capítulos, se trata la planificación y organización llevada a cabo para abordar el proyecto, con la planificación temporal y presupuestos previstos.

En el tercer y cuarto capítulo se lleva a cabo un estudio del arte de las soluciones existentes en la actualidad, así como una introducción teórica a que consiste la gestión de procesos y porqué son necesarias las técnicas de minería de datos aplicadas a esta rama.

En el quinto capítulo se hace un estudio teórico en profundidad de los distintos tipos de minería de procesos, las distintas técnicas que existen, las técnicas de modelado que se emplean, así como las perspectivas que pueden adquirirse para tratar la minería de procesos.

En el sexto capítulo entramos en detalle y se muestra con un ejemplo ficticio en qué consiste la minería de procesos aplicándolo a un caso de uso concreto sobre el proceso de gestión de pedidos. Para ello, se detallan cada uno de los pasos que se deben de seguir para conseguir dashboards que permitan realizar un ejercicio de análisis apoyándose de la herramienta Celonis. Por último, en este mismo capítulo se incluye un informe con el análisis realizado.

El séptimo y último capítulo consiste en analizar con retrospectiva el trabajo realizado así como definir posibles líneas de trabajo futuras, dando un cierre al proyecto.

2. Planificación

En este apartado se detalla la planificación llevada a cabo para el desarrollo del proyecto. Se detallan las actividades que se tienen que realizar así como el tiempo que conlleva la realización de cada una de ellas con el fin de cubrir los objetivos marcados en el proyecto. En base a dicha planificación temporal se incluye un presupuesto ficticio basado en las necesidades identificadas y los recursos necesarios para completar el proyecto. Dicho apartado marca la hoja de ruta a seguir para el desarrollo correcto del proyecto.

2.1. Planificación temporal

El trabajo de fin de grado se engloba en una asignatura de 12 créditos (ECTS). Estos créditos representan un esfuerzo de 300 horas de trabajo, por lo tanto, se establece como base de que se disponen de 300 horas de esfuerzo que se pueden dedicar a la realización del proyecto. Por tanto, se ha realizado una distribución de dichas horas entre las distintas tareas que se deben de llevar a cabo teniendo en cuenta el grado de esfuerzo que conlleva cada tarea.

Puesto que el proyecto se va a realizar compaginando una jornada completa de trabajo, se dispone de unas 3 horas diarias que se pueden dedicar al desarrollo de este. Descontando los fines de semana, esto hace que por cada semana se disponga de una dedicación de unas 15 horas de trabajo, al mes serían en torno a unas 60 horas. Se dispone de los meses de enero, febrero, marzo, abril, mayo y junio. Por tanto hacen un total de 360 horas, a las que se descuenta las dos primeras semanas de enero (vacaciones de navidad) y la última semana de marzo y la primera de abril (vacaciones de semana santa). Esto hace que se disponga de las 300 horas que se parten como base.

De acuerdo a cubrir los objetivos definidos en el apartado anterior, se han planificado una serie de tareas que permitan ser estimables de una manera más precisa y así llevar a cabo la distribución del tiempo disponible para poder llevarlas a cabo en el orden correcto y reducir los bloqueos y retrabajos. Entre estas tareas, se deben incluir como es lógico la redacción de esta memoria.

Análisis y Definición de Objetivos

- **Duración:** 5 horas
- **Descripción:** Revisión de los requisitos del TFG, reunión con el tutor, definición del problema y objetivos específicos del proyecto.

Estudio Teórico Acerca de la Minería de Procesos

- **Duración:** 25 horas
- **Descripción:** Investigación y revisión de literatura relevante, estudio de conceptos teóricos y metodologías.

Evaluación y Selección de Herramientas

- **Duración:** 10 horas
- **Descripción:** Revisión de herramientas existentes, selección de la herramienta Celonis para el proyecto.

Formación Técnica en el Uso de la Herramienta Celonis

- **Duración:** 60 horas
- **Descripción:** Estudio de la documentación de Celonis y realización de cursos en Celonis Academy.
 - **Estudio de la Documentación de Celonis:** 30 horas
 - **Realización de Cursos en Celonis Academy:** 30 horas

Implementación del Caso de Uso de Ejemplo

- **Duración:** 90 horas
- **Descripción:** Desarrollo completo del caso de uso, incluyendo la generación de datos, proceso de ETL y desarrollo de dashboards.
 - **Generar Estructura de Datos:** 10 horas
 - **Generar Datos Aleatorios:** 20 horas
 - **Proceso de ETL:** 30 horas
 - **Desarrollo de los Dashboards:** 30 horas

Pruebas y Ajustes

- **Duración:** 10 horas
- **Descripción:** Realización de pruebas unitarias e integrales, verificación de la funcionalidad del sistema y ajustes necesarios.

Análisis de los Datos y Generación del Informe de Insights

- **Duración:** 20 horas
- **Descripción:** Análisis de los resultados obtenidos y generación de un informe detallado con los insights encontrados.

Redacción de la Memoria

- **Duración:** 80 horas
- **Descripción:** Documentación del proyecto, redacción del informe técnico del TFG y preparación de la presentación final.

La metodología seguida para el desarrollo del proyecto ha sido una metodología en cascada, puesto que debido a la naturaleza del proyecto es la que mejor se adapta. El proyecto es de carácter teórico, con un ejemplo de caso de uso, por tanto los requisitos son precisos desde las primeras fases del proyecto, lo que hace que sea previsible que no haya grandes cambios y en consecuencia no es necesario adoptar una metodología ágil. Además, cada una de las fases dependen de que se hayan completado las anteriores, lo que hace que no se pueden paralelizar las tareas y la metodología en cascada se adapta perfectamente para dicho proyecto.

A continuación, se incluye la planificación del proyecto por días de acuerdo a la estimación de duración de cada una de las fases del proyecto y a la metodología en cascada que se ha decidido seguir. Para ello se ha hecho uso de un diagrama de Gantt, Figura 2.1, el cual nos permite visualizar en un primer vistazo la duración de cada una de las fases y los días o semanas que abarcan. Se ha hecho uso del software de Jira para llevar a cabo dicha planificación temporal.

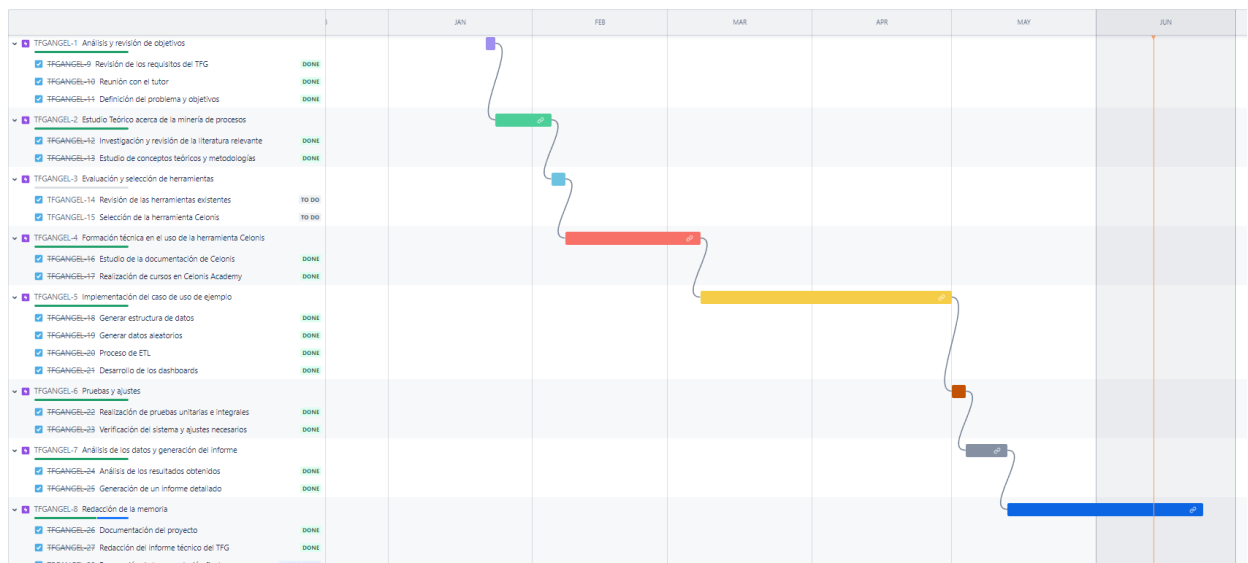


Figura 2.1: Planificación en diagrama de Gantt

2.2. Presupuesto económico

2.2.1. Hardware y Software:

En esta sección se evalúan los costes relativos al uso de recursos de hardware y software utilizados en el desarrollo del proyecto, así como los relativos a la literatura utilizada para disponer de los conocimientos necesarios sobre esta ciencia. Para ello, se considerarán los costos asociados a la adquisición y uso de estos recursos.

- **Hardware:** El recurso principal relativo al hardware para el desarrollo del proyecto es el uso de mi ordenador portátil personal, dicho ordenador tuvo un coste de unos 800€. Se estima que la vida útil del ordenador es de unos 4 años. Para este proyecto se estima que se va a hacer un uso de él de unos 6 meses, por lo tanto un 12.5% de su vida útil. De esta forma, se estima que el coste del hardware para dicho proyecto es de $800€ \times 0.125 = 100€$.
- **Software y herramientas empleadas:** En cuanto a lo relativo al software en todo momento se han utilizado programas de código abierto para la redacción de la memoria. Para la implementación del caso de uso de ejemplo, la herramienta Celonis tiene un coste de licencia, pero se ha utilizado un entorno de prueba que es gratuito con una licencia de tipo académica. Por tanto, no se ha incurrido en ningún coste para este proyecto en cuanto a lo relativo a software. A continuación, se detallan las herramientas empleadas en el proyecto.
 - Celonis: Herramienta principal sobre la que se ha realizado el caso de uso de ejemplo. Es una herramienta dedicada a la minería de procesos y al dashboarding para la generación de paneles y análisis. Es una herramienta de software propietario, pero para la realización del proyecto se ha utilizado un entorno de prueba, el cual no permite compartir el software a otros usuarios, pero permite su desarrollo y visualización con una licencia única de forma gratuita.
 - Jira: Herramienta de gestión de proyectos, con ella he podido realizar la gestión integral de proyecto. Ha sido utilizada para la planificación temporal del proyecto, así como la identificación de bloqueos en el proceso de desarrollo. Permite la gestión de proyectos usando tableros Kanban y Scrum. Para el desarrollo del proyecto fin de grado he utilizado el plan gratuito, el cual incluye la funcionalidad necesaria.
 - LibreOffice: Suite de ofimática de código abierto, ha sido empleada para la redacción de esta memoria, así como para su edición.
 - Python – Google Colab: Python es un lenguaje de alto nivel, interpretado y de código abierto. En concreto en el proyecto he utilizado el entorno de Google

Colab, el cual es un entorno gratuito basado en Jupyter Notebook que permite escribir y ejecutar código directamente en el navegador. Se caracteriza por su ejecución en la nube, y con una cuenta de google se dispone de un entorno de prueba gratuito. En concreto, python y google colab ha sido empleado para la generación de los ficheros .csv que componen la base de datos del ejemplo de caso de uso. Con ello se ha implementado código que genera de forma aleatoria registros de eventos.

- **Literatura y formación:** Debido a la poca documentación teórica existente acerca de esta nueva rama que es la minería de procesos he adquirido el libro *Process Mining: Data Science in Action* del autor *Wil van der Aalst [8]* a través del cual he adquirido los conocimientos necesarios. Dicho libro tiene un coste de 65€.

2.2.2. Recursos humanos

Para hacer el cálculo que supone el trabajo realizado y las horas dedicadas, se han basado los cálculos en el sueldo medio en España del rol requerido o similar para dicho proyecto.

El puesto de trabajo que mejor se ajusta al desarrollo del proyecto es el de un analista de datos, que disponga de conocimientos de consultoría de negocios, así como un manejo a la hora de limpiar, modelar, interpretar, analizar y presentar datos. El salario medio en España de un analista de datos es de unos 12€ la hora. Por la estimación realizada en cuanto a los 12 créditos ECTS que debe suponer el proyecto y en consecuencia las 300 horas estimadas, se concluye que el presupuesto final relativo a recursos humanos para el proyecto es de $12€ \times 300 \text{ h} = 3600€$.

2.2.3. Presupuesto total

En la tabla 2.1 se incluye un resumen de los aspectos económicos del proyecto, comparando las estimaciones iniciales y los costos reales.

Concepto	Estimación Inicial	Costo Real
RRHH	3600 €	3600 €
Hardware	100 €	100 €
Software	0 €	0 €
Literatura/Formación	65 €	65 €
Fondo de contingencia	188 € (5 % de lo presupuestado)	0 €
TOTAL	3953 €	3765 €

Tabla 2.1: Resumen del presupuesto total

Sobre el presupuesto, se ha añadido un concepto relativo al fondo de contingencia, utilizable en caso de que se produzca algún imprevisto en el proyecto. El fondo de contingencia supone un 5% de lo presupuestado. Sin embargo, como se puede apreciar no ha sido necesario utilizar dicho fondo de contingencia y es lo que supone los 188€ de diferencia entre el presupuesto y el costo real del proyecto.

3. Estado del arte

En este apartado, vamos a tratar cual es la situación actual de la minería de procesos y cuál ha sido su evolución a lo largo de los años. También vamos a mencionar algunos profesionales del área que han ayudado en su evolución.

Cabe destacar, que la minería de procesos tiene sus orígenes en la ciencia de los procesos, en la cual se estudia el modelado de los distintos procesos empresariales, con el fin de optimizarlos. Al principio, esta labor era realizada de forma totalmente manual y servía para disponer de forma teórica cómo se realizaban los procesos empresariales, los cuales en una gran mayoría de casos no se corresponden con la realidad.

A medida que la tecnología ha avanzado, ha hecho que este proceso manual se haya dejado de lado, dando lugar a la minería de procesos. Existen dos factores clave que han permitido que esto sea posible. El primero de ellos, se corresponde con la posibilidad de disponer de un gran número de estados por los que puede pasar un proceso y poder recolectar estos datos gracias a los sistemas de información que tenemos hoy en día y a la digitalización de los procesos. Por otro lado, se encuentra la posibilidad de disponer de las herramientas necesarias para manejar estos grandes volúmenes de datos en cuanto a capacidad de cálculo y almacenamiento se refiere.

Junto a las evoluciones que se han comentado anteriormente, fue relevante en torno a la década de 1990 las aportaciones que tuvo el científico holandés Wil van der Aalst, el cual buscó una alternativa a las formas manuales de modelado de procesos. Se dio cuenta de que los datos presentes en los sistemas informáticos podían utilizarse para optimizar los modelos de procesos. La técnica de la minería de procesos fue en gran parte un ejercicio académico hasta el 2011, cuando el Manifiesto de la Minería de Procesos fue publicado por la IEEE. Dos años antes de publicar dicho manifiesto, se fundó el IEEE Task Force en 2009, con la intención de incrementar la visibilidad de la ciencia de la gestión de los procesos.

Hoy en día, disponemos de una gran variedad de software específico, los cuales nos van a permitir realizar nuestros estudios de una forma mucho más rápida y sencilla. Entre dichos programas existen distintas alternativas, existiendo software libre y programas de pago. Más adelante haremos una pequeña introducción a los más importantes.

Por otro lado, a continuación, listamos una serie de profesionales que han tenido relevancia en todos estos años atrás:

- **Dr. Wil M.P. van der Aalst:** Profesor en el departamento de matemáticas y ciencia de computadores en la universidad tecnológica de Eindhoven. Dirige el grupo de ciencia de procesos y datos. Sus intereses en investigación y docencia incluyen sistemas de

información, gestión de flujos de trabajo, redes de Petri, minería de procesos, lenguajes de especificación y simulación.

- **Dr. Marlon Gerardo Dumas:** Profesor en la Universidad de Tartu (Estonia), los últimos años se ha especializado en el campo de la gestión de procesos de negocio.
- **Dr. Anne Rozinat :** Comenzó como estudiante de doctorado en la Universidad de Eindhoven bajo la supervisión de Wil van der Aalst y a día de hoy es una de las mejores profesionales en el campo de la minería de procesos. Entre sus logros destacan el diseño de la herramienta ProM y la confección íntegra del programa Disco.
- **Dr. Marcello La Rosa:** Profesor titular en la Universidad tecnológica de Queensland (Australia); Se le considera uno de los padres de los procesos de negocio y también creador del manifiesto de la minería de procesos.

3.1. Software relacionado

En este apartado de la memoria, vamos a exponer las soluciones en cuanto software se refiere que se encuentran disponibles hoy en día en el mercado y que nos permiten llevar a cabo un proyecto de minería de procesos. Cada una de ellas, dispone de una serie de funcionalidades específicas y que dependiendo de la finalidad de nuestro proyecto se van a adecuar en una mayor o menor medida a nuestras necesidades.

3.1.1. Celonis:

Hoy en día, es el software líder en el mercado de la minería de procesos, (véase su logo en la figura 3.1). El sistema de gestión de ejecución de Celonis proporciona un conjunto de instrumentos y aplicaciones, con un estudio de desarrollo y capacidades de plataforma para ejecutivos y gerentes de negocios.

Las ofertas de Celonis EMS ayudan a las empresas a gestionar todas las facetas de la gestión de la ejecución, desde el análisis, la estrategia y la planificación, la gestión, las acciones y las automatizaciones.

Para poder utilizar Celonis es necesario adquirir licencias, por tanto no es de código abierto.



Figura 3.1: Logo de Celonis

3.1.2. UiPath Process Mining

Herramienta que permite una visualización digital de los procesos, mostrando dónde se encuentran las desviaciones y cuáles son las causas de estas. Es la solución de descubrimiento de procesos comerciales (ABPD) más automatizada disponible. Ofrece información valiosa en tiempo real para mejorar los procesos comerciales a lo largo del tiempo. Véase su logo en la figura 3.2.

Permite que todos los niveles de una organización comprendan los procesos comerciales e identifiquen nuevas soluciones para impulsar la mejora de los procesos comerciales.

También es necesario adquirir licencias para su uso, pero dispone de pruebas de uso gratuitas.



Figura 3.2: Logo UiPath Process Mining

3.1.3. Aris

ARIS es una suite de software desarrollada por la empresa alemana Software AG que ofrece una plataforma integral para modelar, analizar y optimizar procesos empresariales, incluida la minería de procesos. ARIS no es exclusivamente una herramienta de minería de procesos, pero ofrece funcionalidades sólidas en este campo. Proporciona herramientas para descubrir patrones a partir de datos de eventos, comparar modelos con la realidad y mejorar la eficiencia

operativa. ARIS no es de código abierto y es necesario adquirir licencias para su uso. Véase su logo en la figura 3.3.

ARIS

Process Mining

Figura 3.3: Logo Aris

3.1.4. Disco

Desarrollada por la empresa Fluxicon. Son un grupo de estudiantes que realizaron su tesis doctoral en la universidad Tecnológica de Eindhoven con el profesor Wil van der Aalst. La herramienta a nivel de diseño es una de las más intuitivas y fáciles de usar. Está diseñada para ser amigable para usuarios no técnicos y ofrece características como descubrimiento de procesos, conformance checking y análisis de variantes. Es de código abierto. Véase su logo en la figura 3.4.



Figura 3.4: Logo Disco

3.1.5. ProM

Ofrece una amplia gama de herramientas y algoritmos para el descubrimiento, el análisis y la mejora de procesos. ProM es ampliamente utilizado en la comunidad de minería de procesos y permite a los usuarios personalizar sus análisis según sus necesidades. Es de código abierto. Es una herramienta más potente que Disco, permite realizar modificaciones sobre los datos. Requiere mayor experiencia, permite probar algoritmos y métodos propios. Dispone de una gran cantidad de plug-ins creados por la comunidad. Véase su logo en la figura 3.5.



Figura 3.5: Logo ProM

Por último, existen otras herramientas de minería de datos que no están especializadas en minería de procesos, que utilizadas de forma concreta pueden ser útiles para visualizar los procesos y mejorar los modelos. Entre ellas destacan RapidMiner, Kmine o R.

4. Gestión de procesos.

Como hemos comentado anteriormente, la minería de procesos se basa en la gestión de los procesos empresariales. Se considera la gestión de procesos como el origen de la minería de procesos, ya que fue la primera de las técnicas que hacía uso de un modelado de los procesos. Para ello, en este apartado trataremos de introducir en qué consiste la gestión de procesos para poder asentar las bases de la minería de procesos y poder desarrollarla en profundidad.

La **gestión de procesos**, también conocida por gestión de procesos empresariales. (**BPM**, por sus siglas en inglés), comprende la disciplina que se encarga de analizar, modelar, diseñar, implementar, medir, controlar y mejorar los procesos de una organización con el objetivo de lograr una mayor eficiencia, calidad, agilidad en las operaciones.

Los **procesos empresariales** son una secuencia de actividades interrelacionadas que transforman una serie de entradas o inputs en una serie de salidas u outputs. Los procesos empresariales, tienen cabida en cualquier organización, independientemente del tamaño o sector, en toda organización se sigue un proceso.

Los enfoques previos de la gestión de procesos trataban de manera separada a los procesos, las personas y la tecnología, produciendo una brecha a veces insalvable entre las tecnologías y el negocio o giro comercial de la organización. Esta brecha al mediano/largo plazo resulta en pérdidas (millonarias en muchos casos) que las empresas deben afrontar. Si una empresa conoce sus procesos, podrá entonces modelarlos, estudiarlos, medirlos y optimizarlos para satisfacer los objetivos del negocio, y eventualmente cambiarlos según evolucione la actividad, cuando aparezcan nuevas oportunidades, cuando determinada tecnología se quede obsoleta, etc.

BPM considera que una labor fundamental es la supervisión del proceso, para determinar si el proceso está generando los resultados esperados en función de los objetivos del negocio. Para ello, es necesario disponer de una serie de métricas y KPIs (Key Performance Indicator) los cuales resultan claves para realizar un control detallado del proceso.

A continuación, enumeraremos una serie de aspectos clave en la gestión de procesos:

- **Análisis y modelado:** La primera de las fases de la gestión de procesos consiste en analizar y comprender los procesos existentes en la organización. Para ello, debemos de mapear los pasos del proceso, así como los distintos flujos que este puede seguir. Para poder modelar los procesos se hace uso de diagramas visuales, como pueden ser los diagramas de flujo, redes Petri o notación BPMN (Notación de Modelado de Procesos de Negocio), los cuales se explicarán mas adelante en la memoria. Al disponer de un

modelo del proceso, podremos anticipar cuáles van a ser los flujos que va a seguir el proceso, y así también planificar los recursos necesarios en cada etapa. Así como disponer de una visión de la etapa en que nos encontramos en cada momento.

- **Diseño y rediseño:** Una vez que disponemos de una comprensión del proceso, podemos diseñarlos o rediseñarlos con el objetivo de mejorar la eficiencia, calidad y alineación con los objetivos de la organización. Para ello, se podrían eliminar ciertas tareas que se puedan considerar innecesarias, dedicar más recursos a una cierta actividad para reducir los cuellos de botella, automatizar partes del proceso y llevar a cabo una optimización general del flujo de trabajo.
- **Implementación y automatización:** Una vez diseñados los procesos, el siguiente paso consiste en implementarlos para pasar del modelo teórico a la funcionalidad real.
- **Medición y control:** Como ya hemos comentado, es necesario medir constantemente el rendimiento de los procesos, para ello se establecen y definen indicadores clave de rendimiento (KPIs), para evaluar cómo se está cumpliendo con los objetivos en cuanto a tiempo, calidad y recursos. En caso de que los KPIs no cumplan con los objetivos propuestos, se llevan a cabo las correcciones oportunas.
- **Optimización continua:** Mientras que los procesos están funcionando, se debe de buscar oportunidades de mejora de forma continua, como puede ser la identificación de cuellos de botella, adoptar mejores prácticas, etc.

A continuación, vamos a definir que es un **KPI** (Key Performance Indicator en inglés), que se traduce por Indicador Clave de Rendimiento. Es una métrica o medida utilizada para evaluar el rendimiento y el éxito de un proceso en relación con los objetivos estratégicos y operativos establecidos. Son herramientas esenciales para medir, monitorear y gestionar el progreso de los objetivos. Permiten identificar tendencias, oportunidades y tomar decisiones para mejorar la eficiencia y eficacia. Las características que debe cumplir un KPI son las siguientes:

- **Relevancia:** Deben estar alineados con los objetivos de la organización.
- **Medibles:** Deben poderse cuantificar y medir de manera propia.
- **Específicos:** Deben ser claros y centrados en un aspecto concreto del rendimiento.
- **Comparables y analizables:** Deben de poder ser comparados con estándares, objetivos anteriores, etc.

5. Minería de procesos

Una vez que disponemos del conocimiento que tiene que ver con las bases de la minería de procesos, estamos en condiciones de desarrollar en profundidad el tema principal del trabajo.

En primer lugar, para poder posicionar la minería de procesos, vamos a describir en que consiste el ciclo de vida BPM. El ciclo de vida se encarga de describir e ilustrar las diferentes fases que tienen lugar en la gestión de un proceso. Véase la figura 5.1.

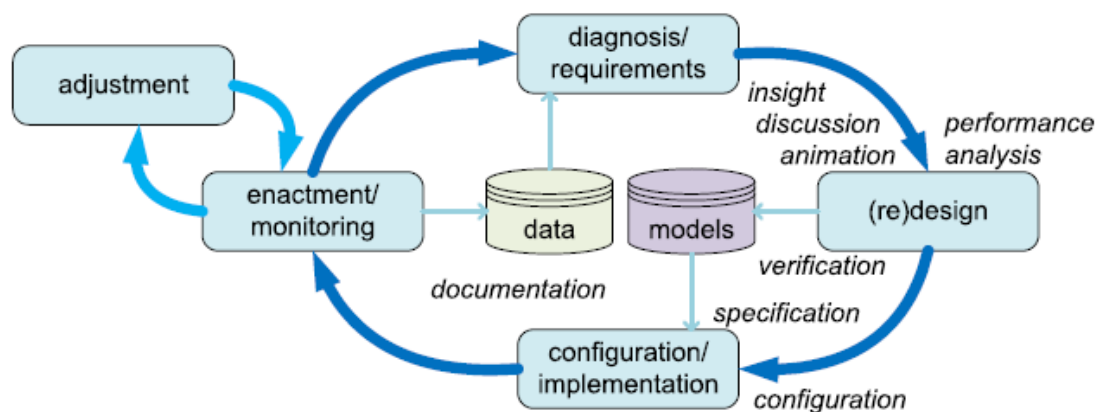


Figura 5.1: Ciclo de vida BPM

En la fase de diseño, el proceso es diseñado. Para que el proceso pueda estar ejecutándose y funcionando es necesario que pase por la fase de configuración o implementación. En esta fase, si el proceso que estamos considerando ya se encuentra funcionando puede ser una fase muy rápida, en caso contrario puede tomar bastante tiempo hasta que el proceso pueda estar ejecutándose completamente. Desde que el proceso empieza a ejecutarse, el proceso pasa por la fase de monitorización en la cual se monitoriza el proceso con el fin de encontrar posibles cambios que se puedan llevar a cabo. En caso de que se detecte un posible cambio, hace que el proceso pase por la fase de ajuste. Es importante mencionar, que en la fase de ajuste el proceso no es rediseñado, ni se implementa nada nuevo, tan solo se llevan a cabo los controles predefinidos para adaptar o reconfigurar el proceso. En la fase de diagnosis se evalúa el proceso y se revisan los nuevos requisitos que pueda tener el proceso por cambios en su entorno, como pueden ser nuevas políticas, competidores, o nuevos objetivos empresariales.

En caso de detectar que el proceso no tiene el rendimiento deseado o se tengan nuevas demandas impuestas por el entorno del proceso puede hacer que se lleve a cabo una nueva iteración en el ciclo de vida del proceso, empezando de nuevo por la fase del rediseño.

Como podemos ver en la figura 5.1. del ciclo de vida del proceso, tienen una gran importancia el modelo y los datos. Por la parte de los modelos, tienen una gran importancia en las fases de diseño (o rediseño) y la de configuración o implementación. Los modelos sirven para verificar que se cumple con lo diseñado en la fase de diseño y establecen una serie de requisitos que se deben de cumplir y seguir en la fase de configuración o implementación.

Por la otra parte, los datos juegan un rol importante en las fases de monitorización y de diagnóstico. A partir de la monitorización documentamos una serie de datos los cuales podemos analizar y por medio de la diagnosis apoyándonos en dichos datos podemos establecer nuevos requisitos sobre los procesos, así como disponer de una mejor visión del proceso actual. Se pueden analizar las desviaciones y la calidad de los modelos puede ser mejorada.

Una vez que conocemos el ciclo de vida de los procesos, podemos situar donde se encuentra la minería de procesos y de ello derivar la importancia que tiene en todo el ciclo de vida. La minería de procesos se encuentra entre los datos y los modelos. Como hemos dicho anteriormente, es una disciplina que se sitúa entre el aprendizaje automático y la minería de datos, por un lado, y el modelado y análisis de procesos, por otro. La principal idea de la minería de procesos consiste en descubrir, monitorizar y mejorar los procesos reales (que en numerosas ocasiones no se corresponde con los procesos que asumimos) a partir del conocimiento que extraemos de los registros de eventos que disponemos en los sistemas actuales.

En la figura 5.2. podremos ver con mayor detalle cual es la conexión entre los procesos y los datos, así como el papel que juega la minería de procesos con los tres tipos de minería de procesos que existen, los cuales son descubrimiento, conformidad y mejora. Se explicarán mas adelante en la memoria.

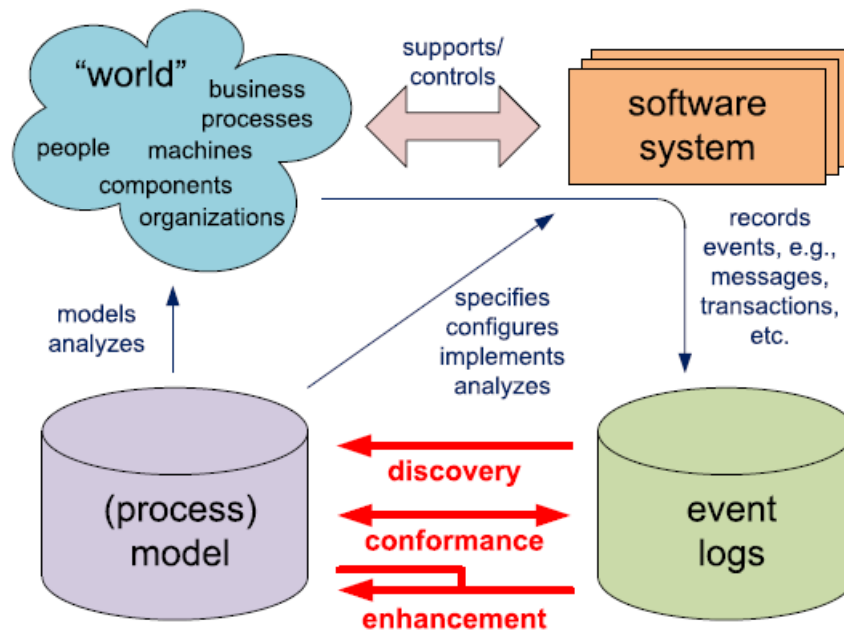


Figura 5.2: Relación entre datos y modelos

Como ya hemos comentado, en la actualidad existen un elevado número de sistemas de información, los cuales registran un gran número de eventos. Entre dichos sistemas destacan los sistemas ERP (SAP, Oracle E-Business Suite, Microsoft Dynamics NAV), sistemas CRM (Microsoft Dynamics CRM, Salesforce), sistemas de información de hospitales, etc. Los cuales, proporcionan información detallada de las actividades que se han ejecutado en cada uno de los procesos que gestionan. A esta información se la conoce como event logs o registros de eventos y todos estos sistemas anteriormente mencionados son los encargados de producir dichos event logs.

Los registros de eventos, en numerosas ocasiones no se encuentran de una forma estructurada, a veces se encuentran dispersos en varias tablas distintas o se encuentran en subsistemas. Para ello, es necesario llevar a cabo un proceso de extracción y transformación de los datos, con la finalidad de obtener los registros de eventos de la forma más clara posible. Los procesos de extracción y transformación son dos fases que se deben de llevar a cabo en la minería de procesos y que se explicarán en detalle más adelante.

Asumiremos que estamos en el caso en el que podemos registrar eventos de forma secuencial, de forma que cada evento se corresponde con una actividad (es decir, un paso que esté bien definido dentro del proceso). Cada uno de los eventos, a parte de corresponderse con una actividad, debe corresponderse con un caso, el cual se corresponde con una instancia del proceso. Fijada una instancia de un proceso, es decir, un caso, se tienen una serie de

actividades sobre dicho caso. Dicha colección de actividades (eventos sobre un mismo caso) de forma secuencial definen una traza del caso.

Las actividades están compuestas por un nombre de actividad y sirve para identificar cada uno de los tipos de eventos, dándoles un nombre reconocible. En ocasiones, los event logs también almacenan información adicional, que en la minería de datos resulta de gran utilidad para realizar análisis más detallados. Esta información adicional que se almacena en los event logs son los recursos o usuarios que realizan la actividad (pueden ser personas o dispositivos), el tiempo exacto del evento, denominado por timestamp o elementos de datos relacionados con el caso y el evento en concreto.

5.1. Tipos de minerías de procesos

Como podemos ver en la figura 5.3 existen tres tipos de minería de procesos los cuales se basan en los registros de eventos (event logs) y modelos. Cada uno de ellos busca mejorar algún aspecto de un proceso o todos ellos pueden llevarse a cabo sobre un mismo proceso. A continuación, explicaremos en que consiste cada una de ellas.

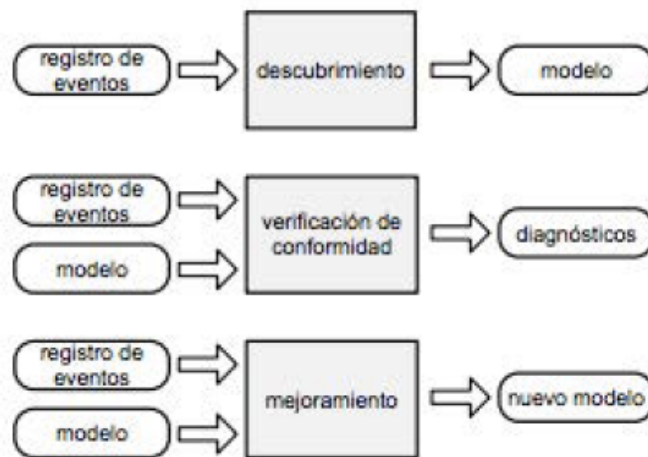


Figura 5.3: Tipos de minería de procesos

Como podemos ver de forma esquemática en la figura 5.3. de cada tipo de minería necesitamos una serie de elementos distintos como entradas o requisitos y se producen una serie de elementos distintos como salidas.

5.1.1. Descubrimiento

Los requisitos necesarios para este tipo de minería son básicamente los registros de eventos del proceso. A partir de dichas entradas vamos a obtener como salida un modelo que describe cómo se está llevando a cabo el proceso realmente. A partir del registro de eventos se puede construir una red de Petri, sin utilizar ningún otro conocimiento previo acerca del proceso, que explica el comportamiento del proceso de acuerdo con los registros de eventos. Más adelante explicaremos en qué consisten las redes de Petri.

Si los registros de eventos, además de la información acerca del nombre de la actividad y del timestamp de esta también contiene información complementaria como pueden ser los recursos o los usuarios que realizan la actividad, podemos también descubrir cómo los trabajadores trabajan juntos y cooperan entre ellos dentro de la organización.

5.1.2. Conformidad

Para este tipo de minería se parte del registro de eventos y de un modelo previo del mismo proceso a estudiar. Como salida obtendremos una serie de diagnósticos que nos indicarán si realmente los registros de eventos se adecúan con el modelo que habíamos definido previamente o no lo hacen. De esta forma, por medio de este tipo de minería podemos detectar, localizar y explicar las desviaciones que tienen lugar en los procesos y que no se adaptan al modelo predefinido, así como medir el impacto de dichas desviaciones sobre el proceso pudiendo tomar las correcciones oportunas.

5.1.3. Mejora

En este tipo de minería, partimos de un modelo y del registro de eventos. La principal idea consiste en extender o mejorar el modelo existente sobre el proceso a partir del proceso actual por medio del registro de eventos. Mientras que en la conformidad vemos como se adecuó el modelo con el proceso real, aquí tratamos de cambiar o ampliar el modelo que disponemos a priori.

Una posible acción a tomar consiste en modificar el modelo, para que este refleje de una mejor forma la realidad del proceso que modela. Como puede ser sustituir el hecho de que dos actividades estuvieran modeladas de forma que tenían que realizarse de forma secuencial, pero

en realidad gracias a los registros de eventos hemos visto que estas pueden ocurrir en cualquier orden, por tanto, debemos de modificar el modelo para que refleje esta posibilidad.

Otra posibilidad, es que podamos encontrarnos con cuellos de botella o bucles que reducen la eficiencia del modelo. Por tanto, el nuevo modelo que se va a generar debe contemplar estos casos y debe ayudar a que se mejore el flujo de ejecución de las actividades.

5.2. Tipos de perspectivas en la minería de procesos

Después de estudiar los distintos tipos de minerías de procesos existentes, es interesante estudiar cuales son las diferentes perspectivas que existen a la hora de desarrollar cada una de las minerías, las cuales nos van a proporcionar una visión distinta de cada uno de los aspectos relacionados con los procesos dentro de las organizaciones. Cada una de las perspectivas que vamos a mencionar pueden ser solapadas de forma parcial y no son exhaustivas. Estas perspectivas nos muestran los aspectos que pretende analizar la minería de procesos.

5.2.1. Perspectiva de control de flujo

Con este tipo de perspectiva tratamos de hacer un análisis centrándonos en el flujo de las distintas actividades del proceso, así como el orden de las actividades. La principal meta de esta perspectiva consiste en encontrar una buena caracterización de todos los caminos posibles dentro del flujo del proceso, para ello se suele utilizar distintas notaciones como las redes de Petri, que veremos más adelante en la memoria.

5.2.2. Perspectiva organizativa

En este tipo de perspectiva se centra en la información de todos aquellos recursos ocultos en los registros de datos. En concreto, se centra en los actores (personas, sistemas, roles, departamentos) y cómo se relacionan entre ellos. El objetivo principal es determinar cual es la estructura de la organización, de forma que podamos clasificar a las personas que forman la organización por roles o departamentos, para poder visualizar cual es la red social de la organización.

5.2.3. Perspectiva de caso

Esta perspectiva se centra en el propio caso que se está estudiando en el proceso. Un caso puede caracterizarse por la trayectoria que sigue en el proceso de estudio, pero también por todas aquellas personas que trabajan en él. En numerosas ocasiones, cada uno de los casos dispone de diferente información relevante con el propio dato y esta información hace que se pueda hacer un estudio sobre esta información extra con el fin de encontrar distintos patrones en relación con dicha información. Por ejemplo, si un caso representa un pedido de aprovisionamiento, es interesante conocer el proveedor o el número de productos pedidos con tal de poder analizar los datos en función de estas características de los pedidos.

5.2.4. Perspectiva temporal

Este tipo de perspectiva se centra en el tiempo y la frecuencia de las distintas actividades. El principal objetivo es detectar por medio de los timestamps los distintos cuellos de botella que se producen en el proceso, así como supervisar la utilización de los recursos y predecir el tiempo de procesamiento de los casos futuros, así como aquellos que se estén ejecutando. Otro de los objetivos clave es analizar el retrabajo, es decir, aquellas actividades que se repiten en un elevado número de veces dentro de un mismo caso y analizar el impacto que estas tienen en cuanto a tiempo de procesamiento se refiere.

Podemos pensar que la minería de procesos se lleva a cabo de forma “off-line”, es decir, la minería de procesos se limita a analizar datos, haciendo solamente una lectura de estos datos. El fin de llevar a cabo dicho análisis es poder mejorar o entender de una forma más correcta los procesos. Sin embargo, en la actualidad cada vez es mayor el número de técnicas de minería de procesos que se llevan a cabo de forma “on-line”, también conocido como soporte operacional. Por ejemplo, podemos detectar en tiempo real cuando un caso se está desviando del modelo, con el fin de tomar acciones en ese mismo momento antes de que dicha desviación conlleve consecuencias peores. Otro ejemplo puede ser predecir cual es el tiempo restante de procesamiento para los casos que se encuentran en un punto intermedio del proceso, para ello, se basa en el histórico de casos similares.

Con todo esto, podemos ver como el alcance de la minería de datos es muy amplio y no se limitan tan solo al descubrimiento de procesos. En la actualidad, la minería de procesos proporciona soporte para todo el ciclo de vida BPM de la figura 5.1. Sirve de soporte no solo

para las fases de diseño y diagnóstico, sino para las fases de implementación y seguimiento, así como la supervisión y ajuste.

5.3. Registros de eventos

Una vez que disponemos los conocimientos previos acerca de la gestión de procesos, así como una introducción de en qué consiste la minería de procesos y cuáles son los objetivos buscados con ella, podemos pasar a estudiar conceptos más técnicos que tienen que ver con la minería de procesos y los cuales nos sirven como base para poder desarrollar toda la teoría que existe por detrás.

La principal herramienta y la base de la minería de datos son los registros de eventos o event logs. Estos registros de eventos, como ya hemos comentado en ocasiones anteriores son producidos por los diferentes sistemas que gestiona el proceso que se va a estudiar. A partir de ellos vamos a poder obtener información de lo que ocurre en el proceso.

Un caso dentro de un proceso está formado por varias actividades, y se corresponde con una instancia de ejecución del proceso. Por otra parte, una actividad es una tarea concreta que se lleva a cabo sobre un caso dentro del proceso. Cada una de las actividades que se llevan a cabo se registran a nivel de dato por medio de la estructura de registro de eventos.

Un registro de eventos está compuesto por tres campos clave y obligatorios. Los cuales son los siguientes:

- **Id del caso:** Identificador único del caso. Es decir, se corresponde con el identificador que nos sirve para identificar una estancia u objeto de estudio sobre el que está involucrado el proceso. Por ejemplo, para el caso del proceso de aprovisionamiento, el caso se corresponde con un pedido (o línea de pedido) y el Id del caso se corresponde con el identificador del pedido (concatenado con la línea del pedido en caso de que el caso sea línea de pedido).
- **Actividad:** Nombre de la actividad que representa ese registro de evento. Debe ser un nombre general de la actividad en cuestión y no debe depender del caso sobre el que se ha realizado la actividad, para ello ya se dispone del campo Id del caso.
- **Fecha o Eventime:** Se corresponde con un timestamp, el cual registra la fecha y hora del evento que representa ese registro. En ocasiones no es posible disponer de la hora en la que ha ocurrido el evento, por lo que se suele fijar una hora por defecto en esa actividad.

Los campos que hemos mencionado anteriormente se corresponden con aquellos campos obligatorios en la estructura de registro de eventos o de actividades. Pero además de estos, es

usual que se incluyan más campos extra, los cuales ayudan a disponer de información contextual de las actividades que se llevan a cabo y permiten posteriormente llevar a cabo un análisis más fructífero. Entre estos campos, se encuentra el usuario, máquina o servicio que lleva a cabo la actividad, el coste que supone la actividad, la fecha de inicio de la actividad (puesto que la fecha del registro o Eventtime siempre almacena la fecha de fin de la actividad), o incluso un campo numérico conocido como sorting el cual sirve para definir el orden de las actividades en caso de que coincidan en fecha y hora. En ocasiones, también se incluye un identificador de evento, el cual permite identificar cada uno de los registros, esto es útil en el caso de que las actividades puedan repetirse dentro de un mismo caso, para poder distinguir cada una de ellas.

A continuación, mostraremos en un formato de tabla un conjunto de registros que forman los registros de eventos o actividades. Véase Tabla 5.1.

Id del caso	Actividad	Fecha/Eventtime
1	A	21/08/2023-10:25
2	C	23/08/2023-12:42
1	B	22/08/2023-08:35
3	D	24/08/2023-07:15
2	D	16/07/2023-18:52
2	A	14/07/2023-19:33
3	B	25/08/2023-22:48

Tabla 5.1: Ejemplo de registro de eventos

En este ejemplo, podemos ver que existen tres casos, con identificadores 1,2 y 3. Sobre cada uno de ellos se han realizado una serie de actividades, por ejemplo, fijémonos en el caso 2, podemos obtener por inspección cuál es su traza. Para el caso 2, se ha realizado el día 14/07/2023 a las 19:33 la actividad A, el día 16/07/2023 a las 18:52 la actividad D y el día 23/08/2023 a las 12:42 la actividad C.

Distintos algoritmos de minería de procesos, como el algoritmo Alpha, permite obtener la red de Petri que describe el proceso a partir de los datos en forma tabular como en la Tabla 1.

5.4. Modelado de procesos

Como hemos hablado en los capítulos previos, el disponer de un modelo de nuestro proceso es una de las tareas principales de la minería de procesos. El medio por el cual a partir de los registros de eventos obtenemos el modelo del proceso es realizado por medio de algoritmos como puede ser el algoritmo Alpha.

Una vez que disponemos de nuestro modelo de procesos, es importante conocer la notación que nos permita entenderle e interpretarle. Para ello, existen distintas notaciones que veremos a continuación y haremos una explicación de cada una de ellas. Cabe destacar, que el pasar de una notación a otra es una tarea sencilla una vez que se entiende cada una de las notaciones.

5.4.1. Redes de Petri

Las redes de Petri son el tipo de notación más antiguo y que más se ha investigado a la hora de modelar procesos. Este tipo de notación permite la concurrencia de actividades dentro de un proceso. La notación que utiliza las redes de Petri es simple e intuitiva, lo que hace que las redes de Petri sea una de las principales herramientas de modelado.

En primer lugar, mostremos por medio de la Figura 5.4. un ejemplo de una red de Petri, el cual utilizaremos como base para explicar en qué consiste la notación empleada.

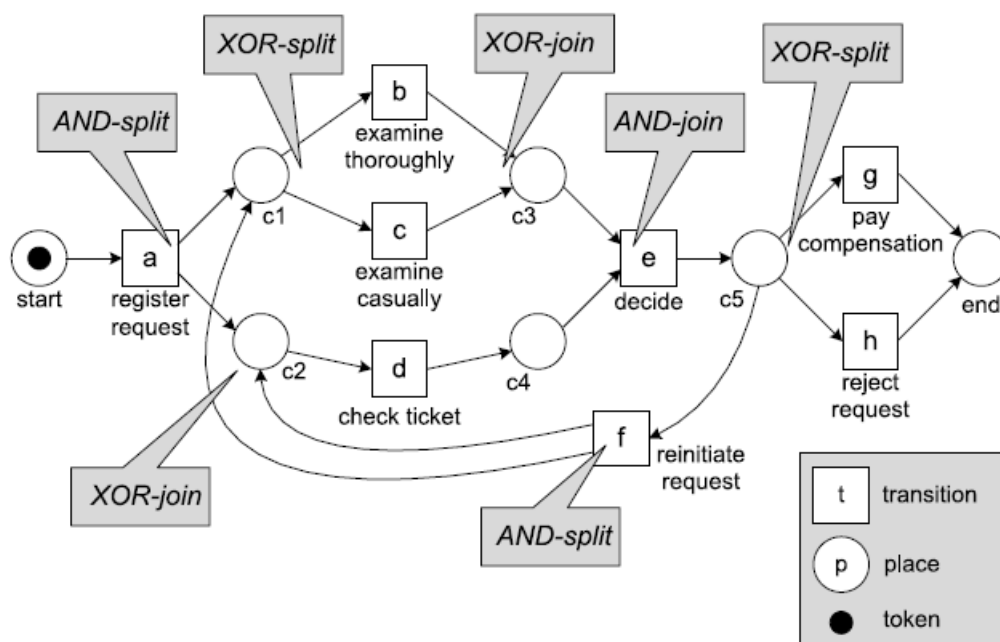


Figura 5.4: Ejemplo de red de Petri

En primer lugar, una red de Petri es un grafo bipartito el cual tiene distintos puestos o places (círculos blancos con letra en el interior que identifica el puesto) y distintas transiciones o transition (cuadrado blanco con letra en el interior que identifica la transición).

La estructura del grafo o red es estática y está gobernada por la “regla del disparo”, existen unos determinados tokens los cuales van moviéndose a través de la red. El estado de la red está determinado por la distribución de los distintos tokens a través de los distintos puestos de la red. Cuando en un puesto existe un token, se dice que dicho puesto está marcado. En la figura 5.4. existe un único token (círculo negro), y el puesto ‘start’ es el único que está marcado.

Las redes de Petri tienen un aspecto dinámico y muestran como se puede ir moviendo el proceso entre los distintos puestos que los componen. Para poder modelar este comportamiento, se basan en la llamada “regla del disparo”. Se dice que una transacción se encuentra activa, si cada uno de los puestos entrantes a la transacción contienen un token, es decir, si se encuentran marcados cada uno de los puestos de entrada de la transición. Una transición que se encuentra activa puede “dispararse” (ejecutarse) consumiendo un token de cada uno de los puestos entrantes a la transacción, y generando un token por cada uno de los puestos salientes de la transición.

Veamos por medio del ejemplo de la figura 5.4. cómo funciona la “regla del disparo”. En un primer lugar, al haber un token en el puesto [start] (el puesto [start] está marcado), y ser el único puesto de entrada de la transición a , la transición a está activa, por lo tanto “disparando” la transición a se producen dos tokens sobre los puestos [c1,c2] provocando su marcado. La transición a no va a poder activarse debido a que no es posible volver a ella a por medio de los caminos del grafo.

En este momento, debido al marcado de los puestos [c1,c2], las transiciones b , c , d se encuentran activas. Del marcado de [c1,c2] y “disparar” b produce que se marquen [c2,c3]. Lo cual provoca que el puesto d esté activo y no pueda volver a estarlo los puestos b y c .

Cabe destacar que la transición f provoca un bucle, mediante el cual hay infinitas secuencias distintas de “disparos” entre los distintos puestos [start] y [end].

Las redes de Petri, permiten distintas lógicas entre los distintos eventos que forman la red, como ejemplo se muestran las siguientes imágenes con las lógicas posibles. Véase la figura 5.5.

Como ya hemos comentado, las redes de Petri son dinámicas y permiten que con una misma red se puedan modelar distintas secuencias de actividades como veremos en el ejemplo de la figura 5.6. En dicha figura, se ejecuta la secuencia [a,b,c,d]. Sin embargo, con la misma red de Petri se podrían haber ejecutado las secuencias [a,c,b,d], [a,e,d] y [a,b,c,d].

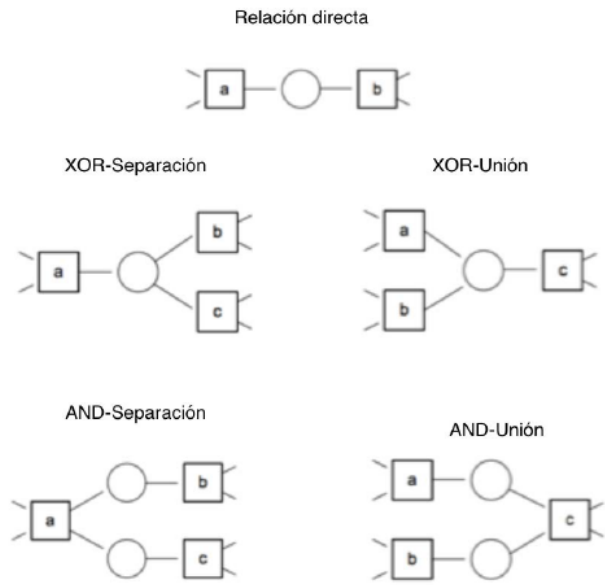


Figura 5.5: Lógica en redes de Petri

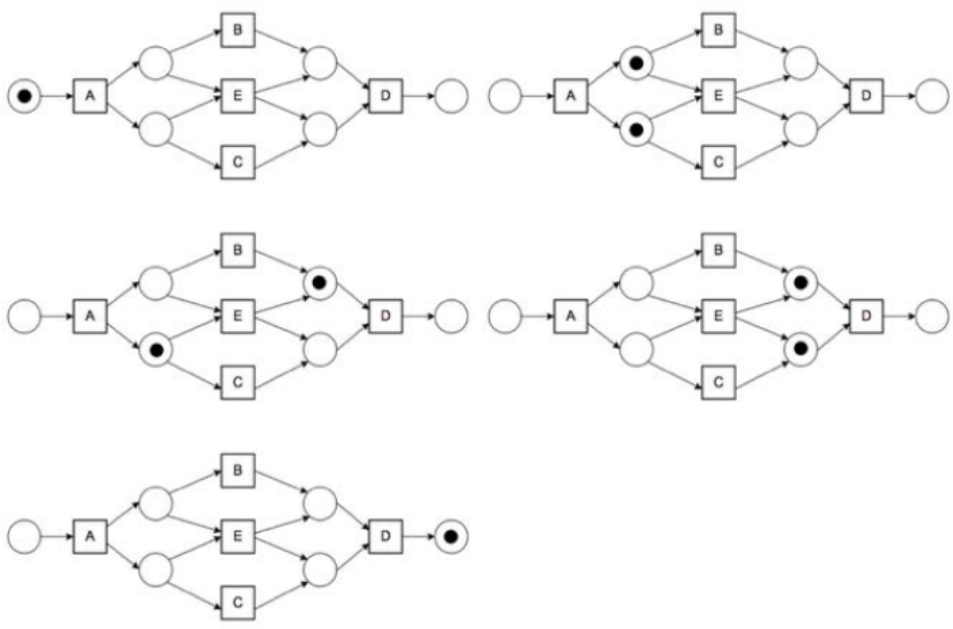


Figura 5.6: Ejecución de una secuencia sobre una red de Petri

4.4.2 BPMN

El BPMN (Business Process Modeling Notation) es otro de los lenguajes utilizados para modelar procesos. En la actualidad, es uno de los más utilizados y se utiliza en numerosas herramientas de modelado. En este lenguaje, se utilizan puertas lógicas en lugar de places para modelar la lógica del control del flujo del proceso.

Los rombos con una 'X' denotan la puerta lógica de XOR y son puertas lógicas de división-uniión. Los rombos con '+' denotan las puertas lógicas de tipo AND.

En la figura 5.7. muestran algunos de los elementos de notación más utilizados. Las actividades atómicas se las denomina tareas y como es usual, estas pueden ser anidadas.

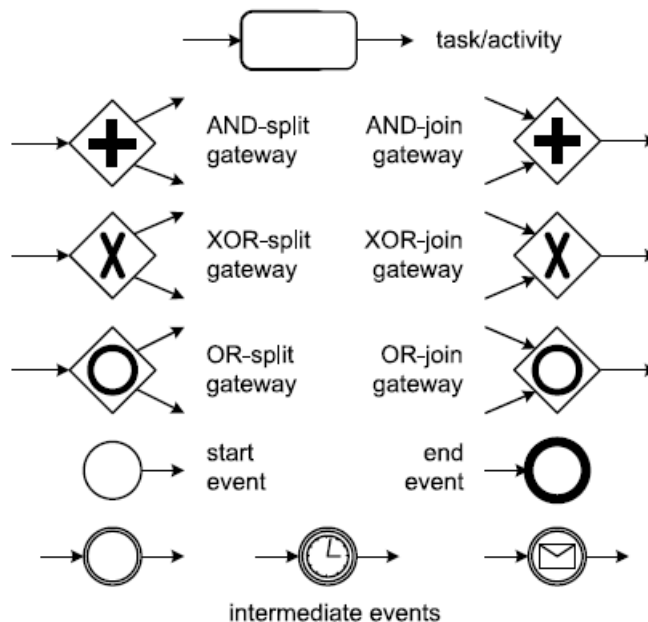


Figura 5.7: Notación BPMN

Las diferentes rutas que puede tomar un proceso son definidas por las distintas puertas lógicas que unen las diferentes tareas. Existen diferentes puertas lógicas de división y unión. Como son las puertas AND, XOR y OR. Los eventos son comparables con los places de las redes de Petri. Sin embargo, la semántica de los places de las redes de Petri y los eventos en BPMN son diferentes. En BPMN no es necesario insertar eventos entre actividades y eventos, tampoco es posible disponer de múltiples arcos de entrada o salida. Los eventos iniciales solo tienen un

único arco saliente, los eventos intermedios tienen un arco entrante y otro saliente y los eventos finales tan solo tienen un arco entrante.

A diferencia de las redes de Petri, en BPMN no se puede tener eventos con múltiples arcos de entrada o de salida, para ello es necesario hacer uso de las puertas lógicas haciendo uso de divisiones y uniones.

A continuación, se explica como podría ser la secuencia de eventos del modelo BPMN de la figura 5.8.

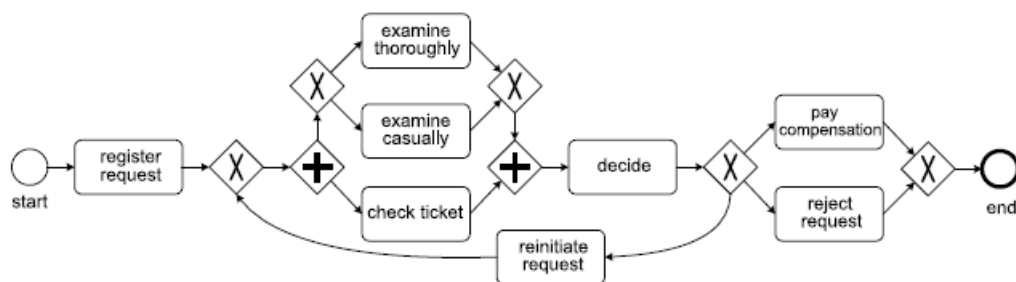


Figura 5.8: Ejemplo de modelo usando BPMN.

Para este modelo, el evento inicial sería la actividad 'register request' a continuación nos encontramos con una puerta lógica XOR de unión (la cual no va a tener mucho protagonismo en este momento, tan solo sirve como unión con la puerta lógica de unión que hay después de la actividad 'decide'), seguidamente nos encontramos con una puerta lógica de tipo AND de división la cual indica que se deben de recorrer ambos caminos, es decir, por un lado se debe de realizar la actividad 'check ticket' y por otro camino nos encontraríamos con la puerta lógica XOR en la cual se puede o bien realizar la actividad 'examine thoroughly' o bien 'examine casually'. Una vez recorridos ambos caminos, nos encontramos con la actividad 'decide' y seguidamente la puerta XOR de división nos lleva a que podemos llevar a cabo la actividad de 'reinitiate request' y volver a empezar (bucle) o bien tomar el camino que nos lleva a realizar o bien la actividad 'pay compensation' o bien 'reject request' (puerta XOR) terminando el proceso.

6. Caso de Uso: Proceso O2C (Order to Cash)

6.1. Introducción

Una vez que hemos estudiado de forma teórica en qué consiste la minería de procesos, vamos a pasar a la parte práctica. Como venimos comentando desde el principio de la memoria la minería de procesos tiene una infinidad de aplicaciones y se puede emplear en estudiar una amplia variedad de procesos empresariales. Desde el proceso productivo de una fábrica, como el proceso de aprovisionamiento de materias primas o incluso el proceso de ventas a los clientes. En general, la minería de procesos se puede aplicar a todo aquello que siga una metodología y unas reglas de trabajo.

Para centrarnos en un proceso concreto y que nos sirva de ejemplo para mostrar cómo se lleva a cabo un proyecto de minería de procesos, vamos a escoger el proceso de recibir y procesar los pedidos de los clientes y su reconocimiento de ingresos (usualmente reconocido por sus siglas en inglés O2C, “Order to Cash”).

Para ello, vamos a suponer que la empresa ficticia llamada “Recambios Car-litos” encargada de la venta de recambios de automóviles nos ha encargado un proyecto de minería de procesos para estudiar su proceso desde que reciben un pedido de un cliente hasta que reciben el ingreso de dicho pedido, es decir del proceso de Order to Cash.

Con el proyecto que nos han encargado se desea tener una visibilidad completa de todo el proceso así como ver si realmente se están ajustando al proceso comercial que tienen marcado. Por otra parte, también se quiere estudiar los plazos de entrega que tienen con los clientes, así como las devoluciones de los artículos que pueden estar defectuosos o los clientes se han equivocado.

Al tratarse de un proyecto de ejemplo y de una empresa ficticia, los datos de los cuales partiremos para nuestros análisis han sido generados de forma aleatoria y son totalmente ficticios, por tanto todas aquellas conclusiones que se puedan obtener dependen de la aleatoriedad de dichos datos. Por tanto, partiendo de dichos datos ficticios, vamos a aplicar técnicas de minería de procesos para obtener una serie de análisis y conclusiones sobre dichos análisis. A partir de dichas conclusiones, se pueden proponer mejoras las cuales hagan más eficiente el proceso de estudio y lograr una mayor satisfacción del cliente.

Para desarrollar el proyecto nos vamos a apoyar en el software de minería de procesos Celonis, el cual detallaremos más adelante de la memoria. Se busca que con este caso de uso a parte de

cumplir con los objetivos del cliente también el lector sea capaz de utilizar la herramienta Celonis, para poder sacar sus propias conclusiones así como indagar de forma autónoma en el proceso.

6.2. Objetivos

El objetivo principal del proyecto consiste en disponer de una visibilidad completa del proceso de O2C que lleva a cabo la empresa “Recambios Car-litos”. También, se desea conocer realmente cómo se están ajustando al proceso teórico que tienen marcado en la empresa a la hora de tener que gestionar el proceso de gestión de pedidos y el recibo de los pagos de esta, para ver si es necesario hacer ajustes sobre dicho proceso teórico o analizar en qué puntos están ocurriendo las desviaciones así como analizar las posibles causas raíz de estas.

Por otra parte, existen una serie de objetivos específicos propios del proceso a estudiar, como son el estudio de los plazos de entrega para mejorar la satisfacción del cliente y lograr aumentar las ventas. Otro de los objetivos es disponer de un estudio de las devoluciones de los pedidos, para poder conocer qué clientes son más propensos a realizar devoluciones, así como el motivo de estas. También es importante conocer cuántos días de media se realiza la devolución después de haber sido entregado el pedido al cliente.

Otro de los puntos de estudio de interés es la facturación tardía, es decir el tiempo que pasa desde que se despacha el producto al cliente hasta que se le genera la factura correspondiente, cuanto menor sea dicho tiempo el cliente debe de pagar antes y por tanto podemos aumentar el flujo de caja.

También, se desea estudiar las actividades indeseadas de bloqueos y cambios en los pedidos, las cuales producen que se aumente el tiempo de proceso y llevan a cabo una mayor implicación de los trabajadores de “Recambios Car-litos” lo que supone costes adicionales. Se desea analizar las causas raíz así como el impacto que tienen.

Por otra parte, otro de los objetivos es analizar la automatización del proceso con el fin de lograr aumentarla y disponer de un proceso lo más automatizado posible, relacionado con esto se desea hacer un análisis del desempeño de los empleados.

Con todo esto, podemos enumerar los distintos objetivos marcados en el análisis del proceso de O2C del ejemplo:

- Visibilidad completa del proceso de O2C.
- Conformidad del proceso con el modelo teórico.

- Estudio de los plazos de entrega y retrasos en las entregas.
- Estudio de las devoluciones de pedidos.
- Análisis de la facturación tardía a clientes.
- Estudio de bloqueos.
- Estudio de cambios.
- Análisis de la automatización del proceso

6.3. Herramientas

Para realizar el proyecto, hemos utilizado la herramienta de software propietario Celonis. Esta herramienta es una de las varias existentes en el mercado para realizar minería de procesos. A continuación, haremos una pequeña explicación de la herramienta y más adelante a medida que se vayan describiendo los pasos que se han llevado a cabo en el proyecto se irá describiendo su funcionalidad de una forma más detallada.

6.3.1. Celonis

Celonis es una herramienta de software propietario fundada en 2011 por Alex Rinke, Bastian Nominacher y Martin Klenk cuyo objetivo principal es permitir realizar un Process Mining de una manera sencilla y muy gráfica.

Celonis está impulsado por su núcleo de minería de procesos líder en el mercado, **Celonis Execution Management System** proporciona un conjunto de aplicaciones y capacidades de plataforma y estudio de desarrolladores para ejecutivos de negocios y usuarios para eliminar miles de millones en ineficiencias corporativas. Véase esquema de funcionamiento en la figura 6.1.

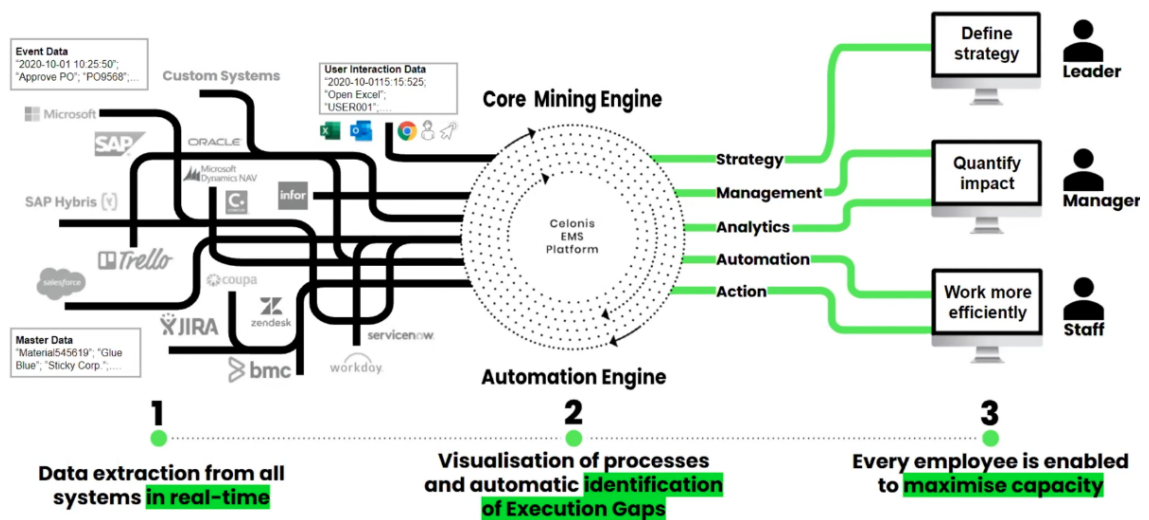


Figura 6.1: Esquema de funcionamiento Celonis

La herramienta de Celonis, permite aplicar inteligencia empresarial en tiempo real en los sistemas de datos de las empresas. Al disponer como base la tecnología de minería de procesos, da soporte a la recopilación de datos, la exploración, el análisis y la concepción de soluciones.

Dependiendo de la licencia de Celonis contratada se tiene acceso a una funcionalidad u otra. Para el caso del proyecto que he desarrollado se ha utilizado una licencia de prueba gratuita con la cual se tiene acceso a un entorno de pruebas con fines académicos.

6.4. Datos

Como ya hemos comentado anteriormente, el proyecto ha sido realizado a partir de una empresa de venta de recambios de automóviles ficticia, por tanto todos los datos que se han empleado han sido generados de forma aleatoria y no se corresponden con datos reales. Los nombres, direcciones y resto de datos acerca de las entidades como clientes, usuarios y pedidos son inventados y cualquier coincidencia con algún dato real será casualidad y carece de validez.

Para hacer un estudio del proceso de O2C, hemos supuesto que los sistemas de información de la empresa “Recambios Car-litos” dispone de una base de datos en la cual cuenta con distintas entidades que permiten almacenar toda la información relativa al proceso. El listado de dichas entidades, las cuales detallaremos a continuación es el siguiente:

- Pedidos
- Productos
- Bloqueos
- Clientes
- Devoluciones
- Historico_Cambios
- Usuarios

Vamos a detallar, en qué consiste cada una de las entidades, reflejando que es lo que modela y el significado de cada uno de los campos, el tipo de dato de cada campo, así como cuando estas han sido generadas de forma aleatoria haciendo uso de Python cual ha sido el código empleado.

6.4.1. Clientes:

Esta entidad refleja un cliente de la empresa “Recambios Car-litos”. En el proyecto, he creado un fichero CSV con 95 registros de clientes distintos, el fichero CSV se llama Clientes.csv

Los campos de dicha entidad son los siguientes:

- Id: (VARCHAR) Identificador único del cliente
- Nombre: (VARCHAR) Nombre del cliente
- Dirección: (VARCHAR) Dirección del cliente
- País: (VARCHAR) País del cliente
- Codigo_iso: (VARCHAR) Código ISO del país del cliente
- Email: (VARCHAR) Dirección de correo del cliente.
- Teléfono: (VARCHAR) Número de teléfono del cliente

6.4.2. Productos:

Esta entidad modela cada uno de los productos que puede vender la empresa “Recambios Car-litos” en general se van a corresponder con recambios de automóviles. Para el proyecto, he creado un fichero CSV con 45 registros de productos distintos, el nombre del fichero CSV es Productos.csv

Los campos de la entidad Productos son:

- Id: (VARCHAR) Identificador único del producto
- Nombre: (VARCHAR) Nombre del producto
- Descripción: (VARCHAR) Descripción del producto
- Categoría: (VARCHAR) Categoría en la que se incluye el producto

6.4.3. Usuarios:

Con esta entidad, se modela cada uno de los usuarios que pueden hacer acciones en el sistema de la empresa de “Recambios Car-litos”. En general, se corresponde con los trabajadores de la empresa, pero también puede incluir algún usuario de tipo automático como pueden ser máquinas o PC’s. Para el proyecto, he creado un fichero CSV con 30 registros de distintos usuarios del sistema, el fichero se llama Usuarios.csv

Los campos de la entidad Usuarios son:

- Id: (VARCHAR) Identificador único del usuario
- Nombre: (VARCHAR) Nombre del usuario
- Automático: (BOOLEANO) Booleano que indica si el usuario es automático o si no lo es

6.4.4. Pedidos:

Entidad que representa cada uno de los pedidos que se han realizado a la empresa “Recambios Car-litos”, esta entidad va a ser la entidad caso sobre la que se van a llevar a cabo los estudios, puesto que es la base del proceso. Para el proyecto, se ha generado de forma aleatoria un fichero CSV por medio de Python en el cual se rellenan de forma aleatoria los datos de los campos de la entidad. En el mismo fichero de Python, se genera de forma aleatoria también los ficheros CSV que modelan las entidades de Bloqueos, Devoluciones, Historico_Cambios puesto que son entidades que están directamente relacionadas con los pedidos y es necesario para mantener la consistencia de los datos. Al final de la descripción de las entidades, se incluirá el código del fichero de generación de pedidos llamado Generacion_Pedidos.ipynb

Para este proyecto el fichero CSV que se ha generado tiene 5000 registros de distintos pedidos realizados entre los años 2022 y 2023. El fichero CSV se llama Pedidos.csv

Los campos de la entidad pedidos son:

- Id: (VARCHAR) Identificador único del pedido
- Id_Cliente: (VARCHAR) Identificador único del cliente que ha realizado el pedido
- Fecha_Pedido_Recibido: (DATETIME) Fecha en la que se ha realizado el pedido por el cliente

- Fecha_Validacion_Pedido: (DATETIME) Fecha en la que se ha validado el pedido por la empresa “Recambios Car-litos”
- Usuario_Valida: (VARCHAR) Identificador único del usuario que valida el pedido
- Fecha_Aprobacion: (DATETIME) Fecha en la que se aprueba el pedido por un trabajador de “Recambios Car-litos”
- Usuario_Aprueba: (VARCHAR) Identificador único del usuario que aprueba el pedido
- Fecha_Comprobacion_Credito: (DATETIME) Fecha en la que se comprueba que exista crédito para el cliente del pedido
- Usuario_Comprueba: (VARCHAR) Identificador único del usuario que comprueba el crédito del cliente
- Fecha_Envio_Confirmacion: (DATETIME) Fecha en la que se envía la confirmación del pedido al cliente
- Usuario_Confirma: (VARCHAR) Identificador único del usuario que envía la confirmación del pedido al cliente
- Fecha_Inicio_Preparacion: (DATETIME) Fecha en la que se empieza a preparar el pedido para enviárselo al cliente
- Usuario_Prepara: (VARCHAR) Identificador único del usuario que prepara el pedido
- Fecha_Envio: (DATETIME) Fecha en la que se ha terminado de preparar el pedido y se envía al cliente
- Usuario_Envio: (VARCHAR) Identificador único del usuario que envía el pedido
- Fecha_Confirmacion_Entrega: (DATETIME) Fecha en la que el cliente confirma que ha recibido el pedido
- Fecha_Limite_Entrega: (DATETIME) Fecha límite acordada en la que el pedido debe ser entregado al cliente
- Fecha_Factura: (DATETIME) Fecha en la que se genera la factura del pedido y se envía al cliente
- Usuario_Factura: (VARCHAR) Identificador único del usuario que genera la factura del cliente y se la envía
- Fecha_Pago: (DATETIME) Fecha en la que el cliente realiza el pago del pedido
- Id_Factura: (VARCHAR) Identificador único de la factura asociada al pedido
- Id_Producto: (VARCHAR) Identificador único del producto que contiene el pedido
- Condicion_Pago: (VARCHAR) Cadena de texto que identifica el tipo de condición de pago del cliente para dicho pedido
- Cantidad: (INTEGER) Cantidad del producto que se ha pedido
- Precio_Total: (FLOAT) Precio total del pedido en €

6.4.5. Bloqueos:

Esta entidad representa un bloqueo que se ha producido sobre un pedido, los tipos de bloqueos son distintos, pueden ser por crédito o por falta de stock. Para el proyecto, se ha generado un fichero CSV en la que los datos de los campos han sido introducidos de forma aleatoria. El fichero CSV se genera también dentro del fichero de Python `Generacion_Pedidos.ipynb` y se llama `Bloqueos.csv`

Los campos de dicha entidad son:

- `Id_Pedido`: (VARCHAR) Identificador único del pedido que ha sufrido el bloqueo
- `Tipo_Bloqueo`: (VARCHAR) Tipo de bloqueo que ha sufrido el pedido
- `Fecha_Bloqueo`: (DATETIME) Fecha en la que se ha producido el bloqueo en el pedido
- `Fecha_Desbloqueo`: (DATETIME) Fecha en la que ha finalizado el bloqueo del pedido y ha quedado desbloqueado
- `Usuario_Bloqueo`: (VARCHAR) Usuario que ha generado el bloqueo y lo ha resuelto

6.4.6. Devoluciones:

Entidad que representa una devolución de un pedido, estas devoluciones pueden ser por varios motivos, como puede ser equivocación, producto roto, desperfectos en el envío, descripción del pedido errónea

Para el proyecto, se ha generado de forma aleatoria el fichero CSV al igual que en el caso de los bloqueos a partir del fichero Python llamado `Generacion_Pedidos.ipynb`, el fichero CSV se llama `Devoluciones.csv`

Los campos de dicha entidad son:

- `Id_Pedido`: (VARCHAR) Identificador único del pedido que ha sufrido el bloqueo
- `Fecha_Devolución`: (DATETIME) Fecha en la cual se ha producido la devolución
- `Motivo`: (VARCHAR) Motivo de la devolución

6.4.7. Historico_Cambios:

Entidad que refleja un histórico de cambios que se han realizado sobre los distintos pedidos, los tipos de cambios que pueden tener los pedidos son: cambios de condición de pago y cambios de cantidad.

Al igual que las últimas entidades se ha generado un fichero CSV de forma aleatoria a través del fichero de Python `Generacion_Pedidos.ipynb`, el nombre del fichero CSV es `Historico_Cambios.csv`

Los campos de la entidad son:

- `Id_Pedido`: (VARCHAR) Identificador único del pedido que ha sufrido un cambio
- `Tipo_Cambio`: (VARCHAR) Tipo del cambio que ha sufrido el pedido en formato de texto
- `Fecha_Cambio`: (DATETIME) Fecha en la que ha ocurrido al cambio en el pedido
- `Old_Value`: (VARCHAR) Valor del campo antes del cambio
- `New_Value`: (VARCHAR) Valor del campo después del cambio
- `Usuario_Cambio`: (VARCHAR) Usuario que ha realizado el cambio en el pedido

6.4.8. Fichero Python para generar los datos:

A continuación, se incluye el código del fichero de Python `Generacion_Pedidos.ipynb` el cual ha sido utilizado para generar las entidades `Pedidos`, `Bloqueos`, `Devoluciones` e `Historico_Cambios`. El resultado obtenido al ejecutar el fichero es de 4 ficheros CSV, uno por cada entidad generada. Con los nombres `Pedidos.csv`, `Bloqueos.csv`, `Devoluciones.csv` e `Historico_Cambios.csv`

```

1. import csv
2. import random
3. from datetime import datetime, timedelta
4.
5. # Función para generar fechas aleatorias dentro del rango de fechas
   indicadas, donde start es la fecha inicial y end la fecha final
6. def random_date(start, end):
7.     return start + timedelta(
8.         seconds=random.randint(0, int((end - start).total_seconds()))
9.     )
10.
11. # Fecha inicial para generación de casos
12. start_date = datetime(2022, 1, 1, 00, 00, 00)
13.
14. # Configuración de rangos
15. num_clientes = 95
16. num_usuarios = 30
17. num_productos = 45
18. num_facturas = 5000
19.
20. #Motivos de devoluciones
21.

```



```

22. motivos_devolución = ['Roto', 'Equivocación', 'Descripción del producto
errónea', 'Defectos en el envío']
23. tipos_cambios = ['Cantidad', 'Condición Pago']
24. condiciones_pago = ['Por adelantado', 'Plazos', 'Al contado', 'Factura
mensual', 'Contrareembolso']
25. tipos_bloqueo = ['Sin stock', 'Crédito']
26.
27. # Incluir las cabeceras de cada uno de los ficheros CSV a generar con
los nombres de los campos
28. with open('Devoluciones.csv', mode='w', newline='') as file2:
29.     writer2 = csv.writer(file2)
30.     writer2.writerow([
31.         'ID_Pedido', 'Fecha_Devolucion', 'Motivo'
32.     ])
33.
34. with open('Historico_Cambios.csv', mode='w', newline='') as file3:
35.     writer3 = csv.writer(file3)
36.     writer3.writerow([
37.
38.         'ID_Pedido',
39.         'Tipo_Cambio', 'Fecha_Cambio', 'Old_Value', 'New_Value', 'Usuario_Cambio'
40.     ])
41.
42. with open('Bloqueos.csv', mode='w', newline='') as file4:
43.     writer4 = csv.writer(file4)
44.     writer4.writerow([
45.
46.         'ID_Pedido',
47.         'Tipo_Bloqueo', 'Fecha_Bloqueo', 'Fecha_Desbloqueo', 'Usuario_Bloqueo'
48.     ])
49.
50. with open('Pedidos.csv', mode='w', newline='') as file:
51.     writer = csv.writer(file)
52.     writer.writerow([
53.
54.         'ID', 'ID_Cliente', 'Fecha_Pedido_Recibido',
55.         'Fecha_Validación_Pedido', 'Usuario_Valida',
56.         'Fecha_Aprobación', 'Usuario_Aprueba',
57.         'Fecha_Comprobacion_Credito', 'Usuario_Comprueba',
58.         'Fecha_Envio_Confirmacion', 'Usuario_Confirma',
59.         'Fecha_Inicio_Preparación', 'Usuario_Prepara',
60.         'Fecha_Envio', 'Usuario_Envio', 'Fecha_Confirmación_Entrega',
61.         'Fecha_Limite_Entrega', 'Fecha_Factura', 'Usuario_Factura', 'Fecha_Pago',
62.         'ID_Factura', 'ID_Producto', 'Condicion_Pago', 'Cantidad',
63.         'Precio_Total'
64.     ])
65.
66.
67.
68. # Generar 5000 pedidos ficticios
69. for pedido_id in range(1, 5001):
70.     id_cliente = random.randint(1, num_clientes)
71.     fecha_pedido_recibido = random_date(start_date, datetime.now())
72.     fecha_validacion_pedido = fecha_pedido_recibido +
timedelta(days=random.randint(0, 5), minutes=random.randint(0, 60),
hours=random.randint(0, 24))
73.     usuario_valida = f'{random.randint(1, num_usuarios)}'

```

```

65.         fecha_aprobacion = fecha_validacion_pedido +
timedelta(days=random.randint(-2, 2),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
66.         usuario_aprueba = f'{random.randint(1, num_usuarios)}'
67.         fecha_comprobacion_credito = fecha_aprobacion +
timedelta(days=random.randint(-2, 2),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
68.         usuario_comprueba = f'{random.randint(1, num_usuarios)}'
69.         fecha_envio_confirmacion = fecha_comprobacion_credito +
timedelta(days=random.randint(0, 4),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
70.         usuario_confirma = f'{random.randint(1, num_usuarios)}'
71.         fecha_inicio_preparacion = fecha_envio_confirmacion +
timedelta(days=random.randint(-2, 10),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
72.         usuario_prepara = f'{random.randint(1, num_usuarios)}'
73.         fecha_envio = fecha_inicio_preparacion +
timedelta(days=random.randint(0, 10),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
74.         usuario_envio = f'{random.randint(1, num_usuarios)}'
75.         fecha_confirmacion_entrega = fecha_envio +
timedelta(days=random.randint(2, 7),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
76.         fecha_limite_entrega = fecha_pedido_recibido +
timedelta(days=random.randint(15, 30),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
77.         fecha_factura = random_date(fecha_envio,
fecha_confirmacion_entrega)
78.         usuario_factura = f'{random.randint(1, num_usuarios)}'
79.         fecha_pago = random_date(fecha_envio, fecha_confirmacion_entrega)
80.         id_factura = pedido_id
81.         id_producto = random.randint(1, num_productos)
82.         condicion_pago = condiciones_pago[random.randint(0,
len(condiciones_pago)-1)]
83.         cantidad = random.randint(1, 10)
84.         precio_total = round(random.uniform(20, 200), 2)
85.
86.         if (33 == random.randint(1,50 )): # Generar una devolución del
pedido de forma aleatoria
87.             with open('Devoluciones.csv', mode='a', newline='') as file2:
88.                 writer2 = csv.writer(file2)
89.                 fecha_devolucion = fecha_confirmacion_entrega +
timedelta(days=random.randint(5, 30),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
90.                 writer2
.writerow([pedido_id,fecha_devolucion,motivos_devolucion[random.randint(0,
len(motivos_devolucion)-1)]]])
91.             if (14 == random.randint(1,20)): # Generar un cambio en el pedido
de forma aleatoria
92.                 with open('Historico_Cambios.csv', mode='a', newline='') as
file3:
93.                     writer3 = csv.writer(file3)
94.                     fecha_cambio = fecha_pedido_recibido +
timedelta(days=random.randint(0, 3),minutes=random.randint(0, 60),
hours=random.randint(0, 24))

```

```

95.         tipo_cambio = random.randint(0, len(tipos_cambios)-1)
96.         if tipo_cambio == 0: #Cantidad
97.             old_value = random.randint(1, 10)
98.             new_value = cantidad
99.         elif tipo_cambio == 1: #Condición de pago
100.            old_value = condiciones_pago[random.randint(0,
len(condiciones_pago)-1)]
101.            new_value = condicion_pago
102.
writer3
.writerow([pedido_id,tipos_cambios[tipo_cambio],fecha_cambio,old_value,new_
value,random.randint(1, num_usuarios)])
103.         if (33 == random.randint(1,50 )): # Generar un bloqueo en el
pedido de forma aleatoria
104.             with open('Bloqueos.csv', mode='a', newline='') as file4:
105.                 writer4 = csv.writer(file4)
106.                 tipo_bloqueo = random.randint(0, len(tipos_bloqueo)-1)
107.                 if tipo_bloqueo == 0: # Sin stock
108.                     fecha_bloqueo = fecha_validacion_pedido +
timedelta(days=random.randint(0, 2),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
109.                     fecha_desbloqueo = fecha_bloqueo +
timedelta(days=random.randint(0, 2),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
110.                 elif tipo_bloqueo == 1: #Bloqueo de crédito
111.                     fecha_bloqueo = fecha_comprobacion_credito +
timedelta(days=random.randint(0, 2),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
112.                     fecha_desbloqueo = fecha_bloqueo +
timedelta(days=random.randint(0, 2),minutes=random.randint(0, 60),
hours=random.randint(0, 24))
113.
writer4
.writerow([pedido_id,tipos_bloqueo[tipo_bloqueo],fecha_bloqueo,fecha_desblo
queo,random.randint(1, num_usuarios)])
114.
115.         # Escribir los datos del pedido en el CSV
116.         with open('Pedidos.csv', mode='a', newline='') as file:
117.             writer = csv.writer(file)
118.             writer.writerow([
119.                 pedido_id, id_cliente, fecha_pedido_recibido,
fecha_validacion_pedido, usuario_valida,
120.                 fecha_aprobacion, usuario_aprueba,
fecha_comprobacion_credito, usuario_comprueba,
121.                 fecha_envio_confirmacion, usuario_confirma,
fecha_inicio_preparacion, usuario_prepara,
122.                 fecha_envio, usuario_envio, fecha_confirmacion_entrega,
fecha_limite_entrega,fecha_factura,usuario_factura,fecha_pago,
123.                 id_factura, id_producto, condicion_pago,cantidad,
precio_total
124.             ])

```

Como se puede comprobar en el código del fichero, es necesario fijar el número de usuarios, clientes, productos y facturas y que los identificadores de estos son enteros en el rango de registros que existan en cada entidad.

Por otra parte, las fechas de los distintos eventos realizados sobre cada uno de los pedidos, no pueden ser aleatorias dentro de los dos años posibles de 2022 y 2023, puesto que la duración total del proceso sería muy larga, y no reflejaba la realidad. Además para disponer de un flujo de proceso lógico es necesario considerar que ciertas fechas de alguna actividades deben ser anteriores a otras, por ejemplo no podemos considerar la fecha de pago previa a la realización del pedido por el cliente. Por ello, para cada una de las fechas de los eventos se ha considerado una fecha aleatoria dentro de un margen lógico y partiendo de la fecha de alguna de las actividades previas para que el flujo sea coherente dentro de la aleatoriedad.

De igual forma, con las actividades de cambios para mantener la consistencia de los datos se ha considerado que el nuevo valor que tiene el campo después de haber sufrido un cambio debe corresponderse con el valor actual que tiene el campo en el registro del pedido.

6.5. ETL (Extracción, Transformación y Carga)

Una vez que disponemos de los datos con los que vamos a llevar a cabo el análisis, el siguiente paso consiste en el proceso de ETL, conocido por sus siglas en inglés de Extraction, Transformation and Load (Extracción, Transformación y Carga de los datos). En este paso, se hace un procesado de los datos para que luego sea más sencillo el análisis de estos.

Siguiendo con el caso de uso, vamos a explicar cómo se lleva a cabo cada uno de los pasos en el proceso de ETL en la herramienta Celonis, explicando cuales son los fundamentos de cada uno de ellos.

6.5.1. Conexión:

En este paso, hacemos la conexión con los distintos sistemas de información que contengan los datos con los que vamos a trabajar en el proyecto. Existen distintos tipos de conexión, los cuales vamos a detallar directamente en las opciones que nos marca la herramienta de Celonis. Para ello, el primero de los pasos es registrarnos en el entorno de trabajo con un usuario que disponga de permisos de analista. Para el caso del proyecto, me he dado de alta en la herramienta y directamente se nos proporciona un entorno de pruebas en las que tenemos un entorno gratuito con una capacidad de 1GB de datos. Una vez que disponemos de nuestro entorno de pruebas, nos logueamos y tendremos acceso con permisos de analista.

Para poder solicitar un entorno de pruebas, así como darse de alta con un usuario visite el siguiente enlace: <https://signup.celonis.com/ui/sign-up/get-started>

Una vez que tenemos acceso al entorno de pruebas de Celonis, disponemos de un menú lateral con varios apartados, el que ahora nos interesa es el apartado de Data, y dentro de Data, el apartado de Data Integration. Véase figura 6.2.

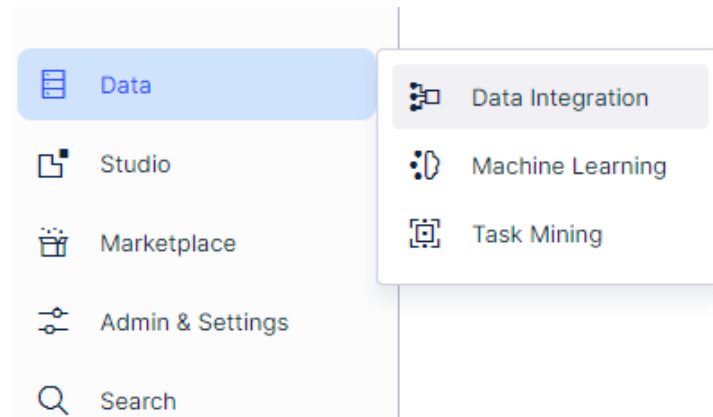


Figura 6.2: Celonis - Menú lateral izquierdo

Cuando hemos accedido al apartado de Data Integration, debemos de crear un Data Pool, el cual sirve para organizar dentro de nuestro entorno los distintos proyectos que podemos crear, cada Data Pool se corresponde con un proyecto. Véase figura 6.3.

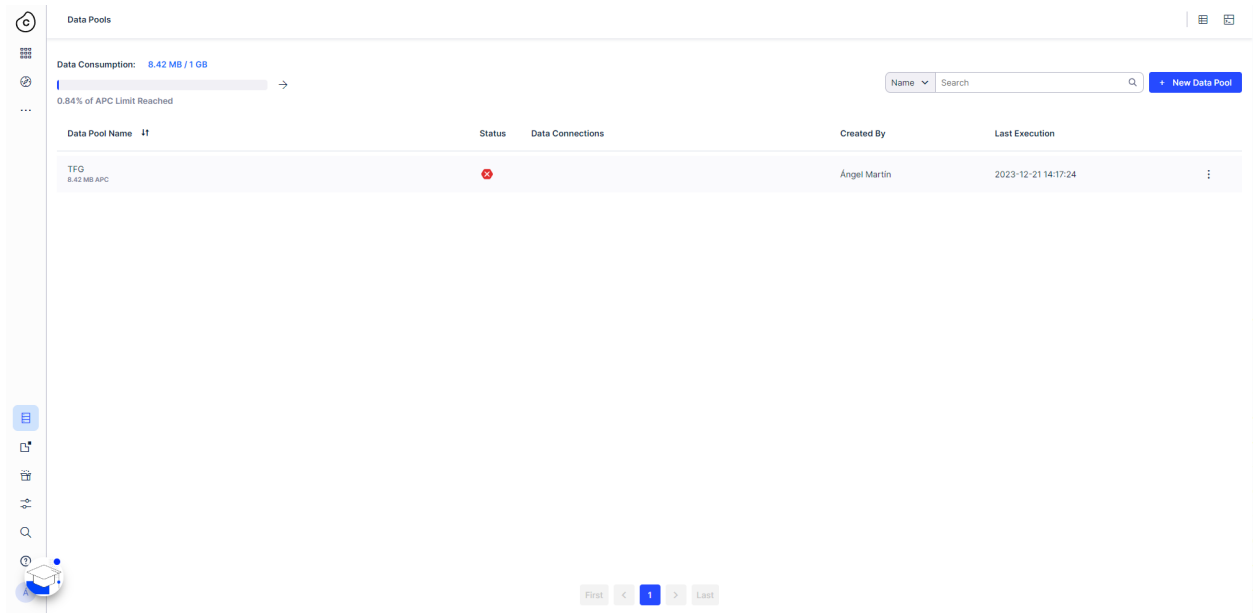


Figura 6.3: Celonis - Data Pools

Clicando en el botón azul en la parte superior derecha con nombre 'New Data Pool', se nos abrirá el siguiente cuadro en el cual daremos un nombre al Data Pool. Véase Figura 6.4.

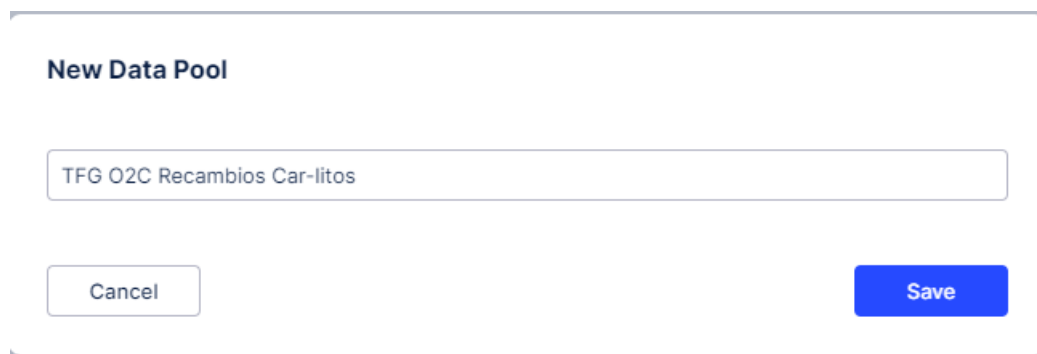


Figura 6.4: Celonis - Nuevo data pool

En el caso del proyecto le hemos dado el nombre de "TFG O2C Recambios Car-litos", clicando en 'Save', creamos el Data Pool y accedemos a él. Lo que visualizamos es por medio de un esquema el cual sirve para entender cuál es el proceso que debemos seguir para realizar el proyecto los distintos apartados en los que vamos a realizar distintas operaciones con los datos.

De izquierda a derecha, se encuentran los sistemas de información de donde vamos a obtener los datos para el proyecto (símbolo de la nube en azul), a su derecha nos encontramos con el propio Data Pool dentro del entorno donde vamos a procesar los datos. Para ello disponemos de tres apartados, el input de los datos (botón azul con el nombre de ‘Connect to Data Source’) clicando en él accederemos a la configuración de la conexión con los sistemas de información.

A su derecha se encuentra el procesamiento de los datos, para ello tendremos que crear distintos Data Jobs en los que llevaremos a cabo el proceso de ETL, creando Data Jobs de extracción, transformación y carga. Como se puede ver, por encima de ellos se encuentra el apartado de Schedules, donde podemos programar la ejecución de esto Data Jobs con el fin de cargar los datos en el sistema de forma periódica. La última parte se corresponde con la carga y generación del Data Model, en cual definiremos las relaciones entre las tablas del proyecto. La parte de la derecha del esquema, se corresponde con el apartado del Studio el cual se explicará con mayor detalle más adelante de la memoria. En el Studio se crearán los distintos dashboards y a ellos tendrán acceso los usuarios finales de la aplicación creada con el proyecto.

A continuación, procederemos a configurar la conexión con los sistemas de información. Para ello, clicando en el botón azul ‘Connect to Data Source’ (Véase figura 6.5) nos aparece el siguiente cuadro (véase figura 6.6) con las distintas posibilidades de conexión y acceso a los datos.

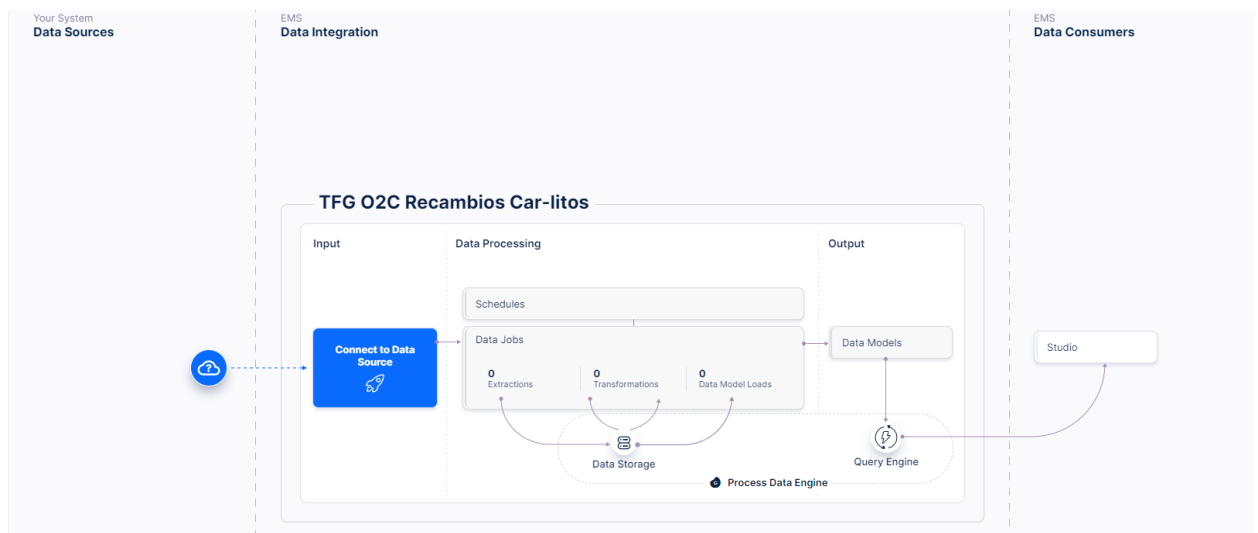


Figura 6.5: Celonis - Connect to data source

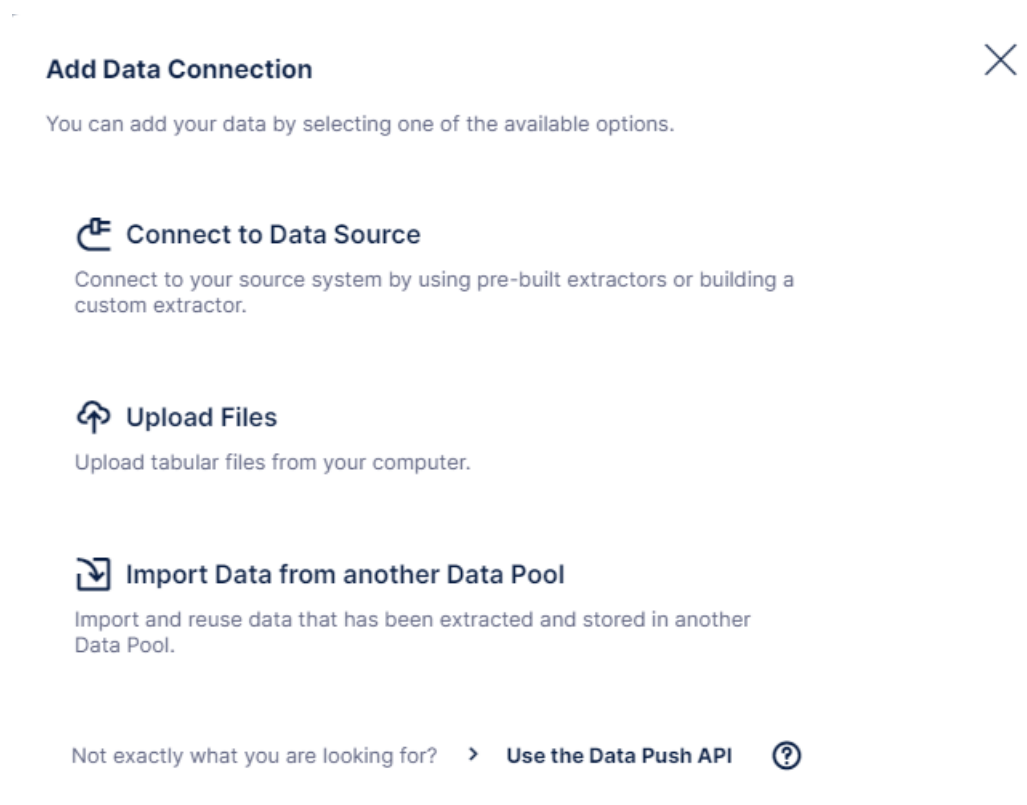


Figura 6.6: Celonis - Tipos de conexión

A continuación, vamos a detallar en qué consiste cada una de ellas y cuál es la que debemos elegir por las características del proyecto.

- **Connect to Data Source:** Esta opción hay que seleccionarla en caso de disponer de un sistema del cual vamos a extraer los datos. Dentro de esta opción se nos mostrarán los distintos sistemas con los cuales podemos hacer una conexión directa. Estos se encuentran divididos en On-Premise (sistemas con la información en servidores físicos), Cloud (sistemas con la información en la nube) o Custom (opción seleccionada en caso de que Celonis no permita conectarse por defecto al sistema). Podemos conectarnos a sistemas CRM, ERP, etc como Salesforce, SAP o también a distintas bases de datos por medio de conexión JDBC o conectores estándar para bases de datos usuales.

Otra de las opciones consiste en crear un conector *custom* por medio de llamadas a API. Para ello, se debe configurar el método de autenticación, así como los distintos *endpoints* para acceder a las distintas entidades que queramos acceder con las llamadas a API.

- **Import Data from another Data Pool:** Esta opción permite importar directamente un Data Pool desde otro Data Pool que hayamos exportado y descargado previamente.

- **Upload Files:** Esta opción permite importar los datos con los que vamos a hacer el estudio directamente de ficheros con formato de datos como pueden ser ficheros .JSON, .CSV, .PARQ, .ZIP, .XLS,....

Esta opción es la que se debe utilizar cuando no se dispone de una conexión directa a las bases de datos que contienen la información del proceso a estudiar, pero si disponemos de unos ficheros con la exportación de cada una de las tablas necesarias dentro de las bases de datos.

Para el caso de uso de ejemplo que estamos tratando al no disponer de una base de datos real y haber generado de forma ficticia unos ficheros que contienen la información de las entidades ficticias que componen los sistemas de información de la empresa 'Recambios Car-litos' es la opción que debemos de escoger como método de conexión a los sistemas de información. (Podríamos decir que cada uno de estos ficheros se corresponden con una exportación de datos de cada una de las entidades consideradas para realizar el estudio).

6.5.2. Extracción:

El proceso de extracción consiste en obtener las tablas de los sistemas de información que nos dan acceso a los datos del proyecto. Es decir, consiste en la descarga de los datos que nos van a permitir llevar a cabo estudios en el sistema. Para ello, dependiendo de la conexión con los sistemas de información se realizará de una forma u otra.

- **Conexión directa con la fuente de datos:** Si disponemos de una conexión directa con un sistema de información como puede ser una base de datos en este proceso debemos de seleccionar aquellas tablas y los campos que disponen de los datos necesarios para llevar a cabo el estudio y que deben de haber sido identificadas en un proceso previo. El primer paso consiste en seleccionar los campos a extraer así como el tipo de datos de cada uno de ellos. Otro de los pasos a dar en la extracción se corresponde con un filtrado por cierto campo dentro de la tabla a extraer para limitar la descarga de todos los registro de dicha tabla y así solo extraer los registros que nos van a resultar útiles para el posterior estudio. Un paso importante se corresponde con el filtrado por fecha, para así solo extraer aquellos registros de la tabla que se correspondan con una fecha en adelante puesto que a lo mejor en nuestro estudio tan solo tienen sentido datos a partir de una fecha.
- **Ficheros de datos:** Cuando disponemos de la información en forma de ficheros de datos, el proceso de conexión realmente no es necesario y se sustituye por la carga de ficheros en la herramienta Celonis y la parte de la extracción se corresponde con la

configuración de la lectura de dichos ficheros y su posterior lectura. A continuación se va a detallar cómo se lleva a cabo dicho proceso para el caso de uso de ejemplo de la empresa ‘Recambios Car-litos’

Ahora, vamos a mostrar como se ha realizado la carga de los ficheros .CSV dentro de la herramienta Celonis para posteriormente llevar a cabo el procesado de dichos datos y poder hacer estudios con ellos. Partiendo de la imagen anterior, en el cual se indican los distintos métodos de conexión explicados anteriormente, se debe de seleccionar como ya hemos dicho la opción de “Upload Files” y se nos mostrará la siguiente ventana (véase figura 6.7):

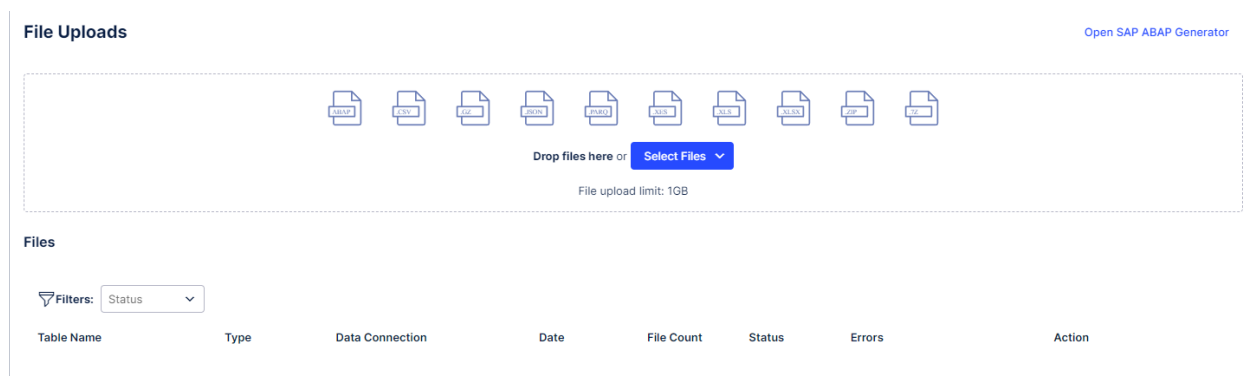


Figura 6.7: Celonis - Upload files

En ella, seleccionaremos o arrastraremos los distintos ficheros de datos que queramos cargar en el sistema. En nuestro ejemplo de caso de uso, cargaremos los ficheros .CSV generados y que ya hemos mencionado en el apartado 5.4 Datos. Dichos ficheros se corresponden con Pedidos.csv, Bloqueos.csv, Historico_Cambios.csv, Usuarios.csv, Productos.csv, Devoluciones.csv y Clientes.csv . Una vez cargados en el sistema, se mostrarán de la siguiente forma en la herramienta (véase figura 6.8):

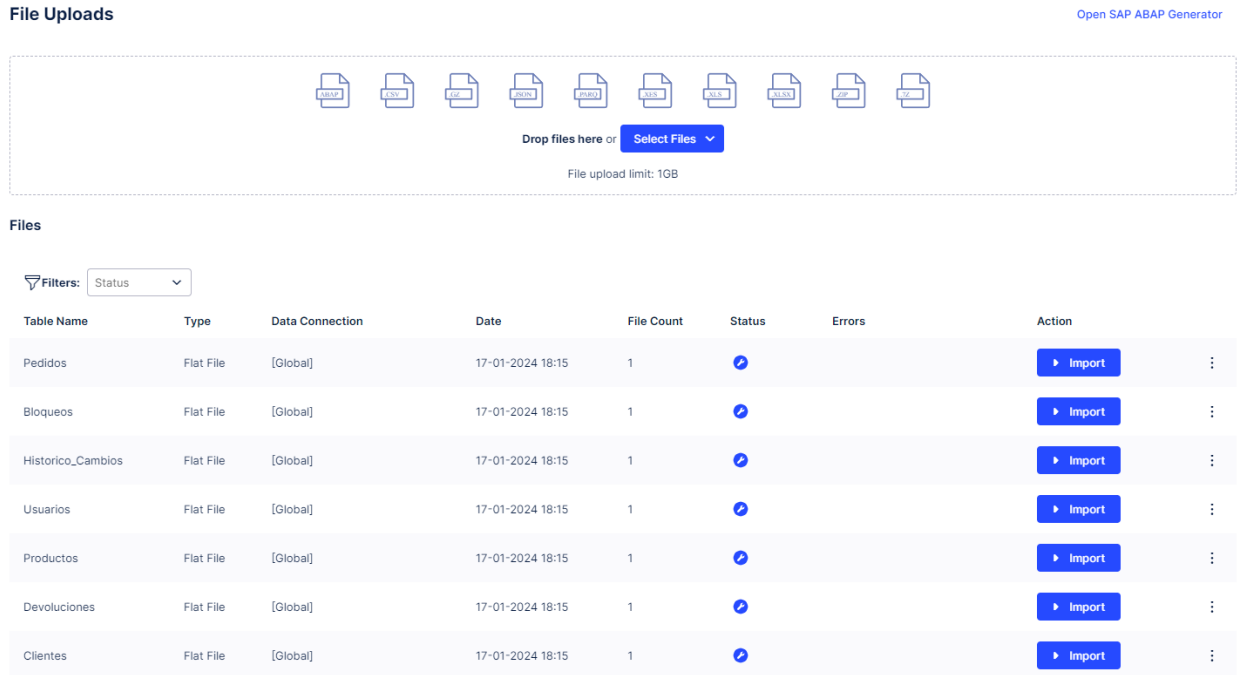


Figura 6.8: Celonis - Ficheros cargados

Llegados a este punto, debemos de clicar en cada uno de los ficheros .CSV en los tres puntos al lado derecho de la columna de “Action”

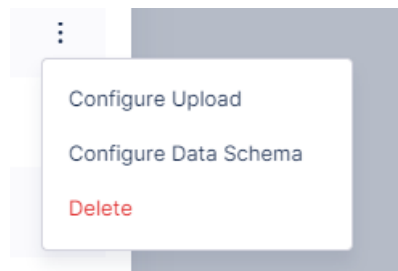


Figura 6.9: Celonis - Ajustes sobre ficheros

Y seleccionar la opción de Configure Data Schema (véase figura 6.9). De esta forma, podremos determinar cual es la forma en la que se tienen que hacer la lectura de cada uno de los ficheros .CSV, así como el tipo de datos de cada una de los campos que componen dichos

ficheros de datos. La ventana que nos aparece para por ejemplo el fichero Pedidos.csv es la siguiente (véase figura 6.10):

×

Pedidos

File encoding Field separator Quote character Escape sequence Line ending

Sheet has header row

ID	ID_CLIENTE	FECHA_PEDIDO_RECIBIDO	FECHA_VALIDACIÓN_PEDI...	USUARIO_VALIDA	FECHA_APROBACIÓN	USUARIO_APRUEBA	FECHA_COMPROE
1	45	2022-09-24 16:34:43	2022-09-26 20:58:43	14	2022-09-24 22:49:43	14	2022-09-27 1
2	16	2022-11-03 12:50:26	2022-11-09 01:52:26	13	2022-11-11 04:23:26	27	2022-11-13 22
3	54	2022-04-18 16:11:25	2022-04-19 01:17:25	13	2022-04-18 11:03:25	24	2022-04-16 1
4	25	2022-10-27 00:57:16	2022-10-30 09:31:16	22	2022-10-29 22:17:16	15	2022-10-31 10
5	36	2023-01-24 22:36:17	2023-01-30 03:55:17	22	2023-02-02 02:23:17	4	2023-02-04 0
6	29	2022-08-26 06:01:10	2022-08-27 11:02:10	18	2022-08-28 15:47:10	1	2022-08-29 1
7	77	2023-08-21 13:33:46	2023-08-24 08:14:46	19	2023-08-22 23:23:46	15	2023-08-25 1
8	57	2023-05-18 11:58:50	2023-05-23 23:09:50	25	2023-05-25 11:01:50	8	2023-05-26 0
9	20	2023-02-27 16:07:19	2023-03-03 14:08:19	25	2023-03-03 14:52:19	20	2023-03-04 2
10	80	2023-05-18 12:40:37	2023-05-22 04:30:37	23	2023-05-21 09:54:37	12	2023-05-22 1
11	23	2022-07-02 21:18:26	2022-07-06 16:18:26	20	2022-07-06 09:58:26	6	2022-07-08 1
12	92	2022-08-28 21:19:25	2022-09-03 01:07:25	10	2022-09-05 20:48:25	22	2022-09-06 0
13	6	2022-06-15 11:12:01	2022-06-20 16:27:01	23	2022-06-22 13:25:01	13	2022-06-23 1
14	10	2023-01-30 12:42:28	2023-02-03 01:58:28	17	2023-02-05 00:23:28	26	2023-02-04 1
15	72	2023-01-24 06:20:53	2023-01-25 05:19:53	12	2023-01-27 11:55:53	10	2023-01-26 1
16	48	2022-09-13 01:41:36	2022-09-14 17:26:36	7	2022-09-16 10:55:36	18	2022-09-14 1
17	12	2023-03-13 12:42:28	2023-03-16 21:05:28	8	2023-03-16 14:04:28	13	2023-03-14 2
18	90	2023-04-07 10:44:35	2023-04-13 08:58:35	17	2023-04-14 05:59:35	17	2023-04-17 0
19	54	2022-02-21 10:59:32	2022-02-21 14:51:32	11	2022-02-20 16:59:32	27	2022-02-21 0
20	77	2022-07-09 03:46:39	2022-07-09 03:48:39	5	2022-07-07 16:05:39	29	2022-07-07 0

Figura 6.10: Celonis - Fichero Pedidos.csv

En ella, en la parte superior, se debe de indicar cual es la codificación del fichero (en este caso UTF-8), cual es el carácter que hace de separador de los campos de cada una de las líneas (en este caso el carácter ‘,’), cual es el carácter de comillas (en este caso no indicamos ninguno al no ser necesario). También se indica cual es el carácter que sirve para escapar caracteres especiales (en este caso es ‘\’). Por último, se indica cual es el carácter que determina cual es el cambio de línea y por tanto de registro (en este caso es ‘\r\n’). Cabe destacar, que en la mayoría de los casos la detección de los caracteres se hace de forma automática, pero siempre es necesario revisar que sean correctos para que la lectura de los datos sea correcta.

Una vez que hemos configurado la forma en la que se tiene que hacer la lectura de los ficheros .CSV debemos de dar en el botón azul “Next” de la parte inferior derecha y pasaremos a la nueva ventana en la cual debemos de configurar el tipo de datos de cada uno de los

campos que componen los ficheros de datos. Para el mismo fichero Pedidos.csv se muestra lo siguiente (véase figura 6.11):

The screenshot shows a configuration window for a CSV file named 'Pedidos'. At the top, there are options for 'Convert all columns to string', 'Thousand separator' (set to ','), and 'Decimal separator' (set to '.'). Below this is a table with 8 columns. Each column has a dropdown for its data type and a text input for its value. The data types are: ID (STRING), ID_CLIENTE (STRING), FECHA_PEDIDO_RECIBIDO (DATETIME), FECHA_VALIDACION_RECIBIDA (DATETIME), USUARIO_VALIDA (STRING), FECHA_APROBACION (DATETIME), USUARIO_APRUEBA (STRING), and FECHA_COMPRA (DATETIME). The values for each column are: 80, 80, 'yyyy-MM-dd HH:mm', 'yyyy-MM-dd HH:mm', 80, 'yyyy-MM-dd HH:mm', 80, and 'yyyy-MM-dd H'. Below the table is a list of 19 rows of data. At the bottom right, there are 'Previous' and 'Save' buttons.

ID	ID_CLIENTE	FECHA_PEDIDO_RECIBIDO	FECHA_VALIDACION_RECIBIDA	USUARIO_VALIDA	FECHA_APROBACION	USUARIO_APRUEBA	FECHA_COMPRA
3	54	2022-04-18 16:11:25	2022-04-19 01:17:25	13	2022-04-18 11:03:25	24	2022-04-16 12:00:00
4	25	2022-10-27 00:57:16	2022-10-30 09:31:16	22	2022-10-29 22:17:16	15	2022-10-31 13:00:00
5	36	2023-01-24 22:36:17	2023-01-30 03:55:17	22	2023-02-02 02:23:17	4	2023-02-04 07:00:00
6	29	2022-08-26 06:01:10	2022-08-27 11:02:10	18	2022-08-28 15:47:10	1	2022-08-29 16:00:00
7	77	2023-08-21 13:33:46	2023-08-24 08:14:46	19	2023-08-22 23:23:46	15	2023-08-25 12:00:00
8	57	2023-05-18 11:58:50	2023-05-23 23:09:50	25	2023-05-25 11:01:50	8	2023-05-26 07:00:00
9	20	2023-02-27 16:07:19	2023-03-03 14:08:19	25	2023-03-03 14:52:19	20	2023-03-04 21:00:00
10	80	2023-05-18 12:40:37	2023-05-22 04:30:37	23	2023-05-21 09:54:37	12	2023-05-22 11:00:00
11	23	2022-07-02 21:18:26	2022-07-06 16:18:26	20	2022-07-06 09:58:26	6	2022-07-08 19:00:00
12	92	2022-08-28 21:19:25	2022-09-03 01:07:25	10	2022-09-05 20:48:25	22	2022-09-06 07:00:00
13	6	2022-06-15 11:12:01	2022-06-20 16:27:01	23	2022-06-22 13:25:01	13	2022-06-23 11:00:00
14	10	2023-01-30 12:42:28	2023-02-03 01:58:28	17	2023-02-05 00:23:28	26	2023-02-04 14:00:00
15	72	2023-01-24 06:20:53	2023-01-25 05:19:53	12	2023-01-27 11:55:53	10	2023-01-26 17:00:00
16	48	2022-09-13 01:41:36	2022-09-14 17:26:36	7	2022-09-16 10:55:36	18	2022-09-14 17:00:00
17	12	2023-03-13 12:42:28	2023-03-16 21:05:28	8	2023-03-16 14:04:28	13	2023-03-14 20:00:00
18	90	2023-04-07 10:44:35	2023-04-13 08:58:35	17	2023-04-14 05:59:35	17	2023-04-17 00:00:00
19	54	2022-02-21 10:59:32	2022-02-21 14:51:32	11	2022-02-20 16:59:32	27	2022-02-21 05:00:00

Figura 6.11: Celonis - Tipo de datos fichero Pedidos.csv

En esta ventana, en la parte superior debemos de configurar para los tipos de datos numéricos el carácter que se corresponde con el separador de millares (en este caso se corresponde con ‘,’) y el separador de decimales (en este caso se corresponde con ‘.’).

Por último, para cada uno de los campos del fichero debemos de indicar el tipo de datos de los posibles que existen (INTEGER, DATE, DATETIME, FLOAT, BOOLEAN y STRING). Para el caso del tipo de dato DATE y DATETIME debemos de indicar cual es el formato de la fecha. Por ejemplo, para el campo FECHA_PEDIDO_RECIBIDO del fichero Pedidos.csv el tipo de datos es DATETIME (pues guardamos un registro del día y la hora-minutos-segundos en el cual se ha recibido el pedido, en caso de no guardar la hora-minutos-segundos el tipo de datos sería DATE) y el formato es “yyyy-MM-dd HH:mm:ss”. Donde yyyy indica el año, MM el mes, dd el día, HH las horas, mm los minutos y ss los segundos. Véase figura 6.12.

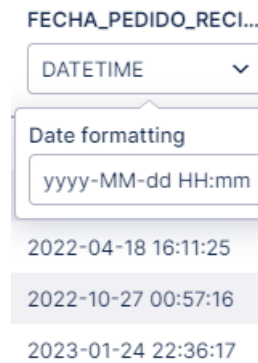


Figura 6.12:
Celonis - Formato
DATETIME

Una vez que hemos indicado el tipo de datos de cada uno de los campos del fichero debemos de dar en el botón azul “Save” de la parte inferior derecha. Y volveremos a la ventana en la cual disponemos el listado con todos los ficheros .CSV que se han cargado en el sistema.

El proceso anterior se debe de repetir con cada uno de los ficheros .CSV que se han cargado. Por último, una vez que se han configurado cada uno de los ficheros debemos de Importarlos en el entorno y hacer la lectura de ellos. Para ello, debemos de seleccionar en el botón azul “Import” debajo de la columna “Action” y se empezará a hacer la importación de cada uno de ellos. El proceso habrá terminado cuando el campo status disponga de un check verde en todos los ficheros cargados. De esta forma dispondremos en el sistema de una tabla de datos con los campos correspondientes a cada uno de los ficheros .CSV. y habremos terminado el proceso de extracción de datos. Véase figura 6.13.

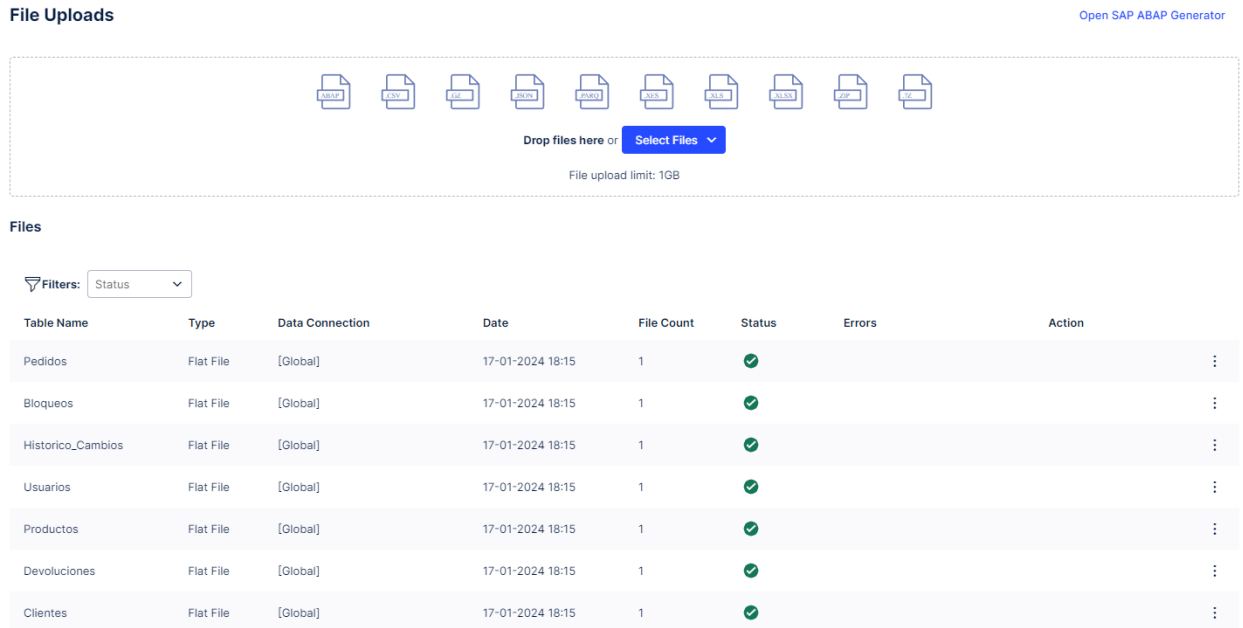


Figura 6.13: Celonis - Ficheros cargados y formateados

6.5.3. Transformación:

El siguiente paso es común independientemente del tipo de conexión que se disponga con los sistemas de información y es necesario haber realizado previamente la extracción de los datos y disponer de dichos datos en la herramienta Celonis. En el proceso de transformación, como su propio nombre indica se llevan a cabo las transformaciones necesarias sobre los datos en crudo extraídos de los sistemas de información para adaptarlos y que nos permitan llevar a cabo análisis con ellos. En este caso, puesto que queremos llevar a cabo un proceso de minería de procesos y como se ha comentado en la introducción teórica debemos de crear las tablas que modelen las entidades básicas con las cuales vamos a trabajar y estas son la tabla de casos y la tabla de actividades.

Aparte de las tablas propias necesarias para los proyectos de minería de procesos (casos y actividades) también se deben de hacer las transformaciones oportunas sobre las tablas que modelan las entidades que entran en juego dentro del proceso que se desea estudiar. Dichas tablas, componen lo que se conoce como metadata del proceso y forman las tablas de masterdata. Véase figura 6.14.

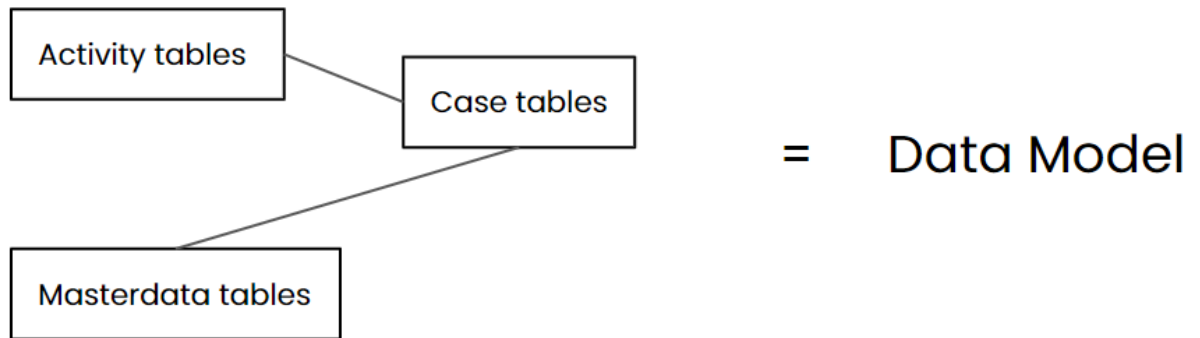


Figura 6.14: Esquema modelo de datos

Como hemos hecho anteriormente, vamos a basarnos en el caso de uso de ejemplo de la empresa 'Recambios Car-litos'. En primer lugar, partiremos del punto anterior, es decir, partimos de que hemos realizado la extracción de los datos y disponemos de las tablas con los datos de los distintos sistemas de información que almacenan la información del proceso que vamos a estudiar, en este caso el proceso de O2C. Puesto que en el ejemplo que estamos tratando hemos realizado una carga de las entidades como ficheros .CSV, se habrá generado en el sistema una tabla que modele cada uno de los ficheros .CSV cargados en el sistema y tendrán el mismo nombre.

Para crear las transformaciones necesarias en la herramienta Celonis debemos de partir de la vista de Data Pool y en este caso debemos ir al apartado de Data Jobs (véase figura 6.15). Dentro de dicho apartado nos encontramos con tres subapartados, los cuales son:

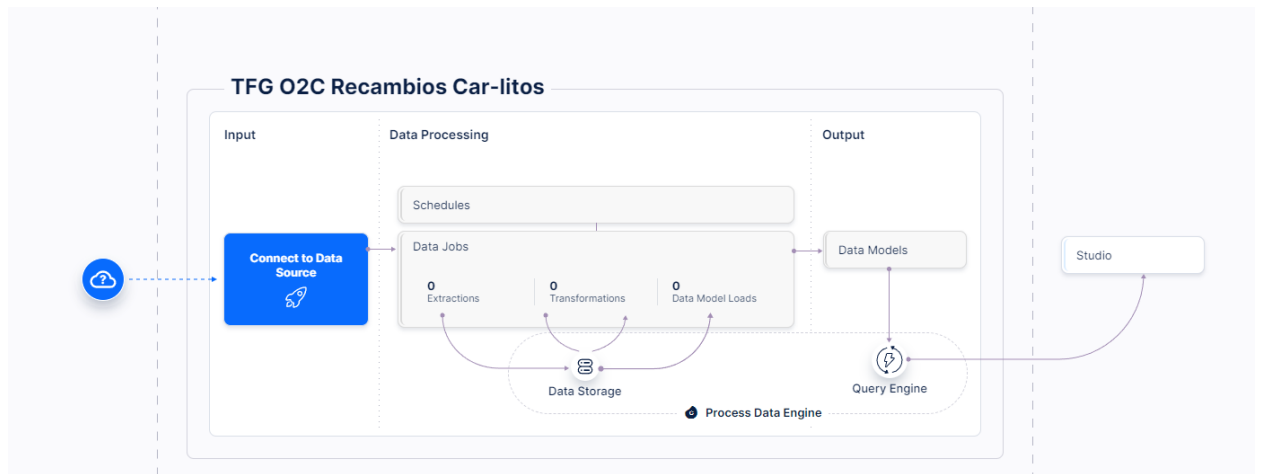


Figura 6.15: Celonis – Data Integration

- **Extractions:** Trabajos de datos que se encargan de configurar las extracciones de los datos de los distintos sistemas de información. Se configuran las tablas que se deben de extraer los distintos filtros temporales y sobre los campos que se deben de tener en cuenta a la hora de extraer los registros de dichas tablas. En este subapartado, se encontraría el trabajo que se realiza en el proceso de extracción dentro de la ETL.
- **Transformations:** Trabajos de datos en los cuales se definen por medio de consultas en SQL de las transformaciones que se deben de realizar sobre los datos en crudo. Creación de la tabla de casos, tablas de metadata del proceso y que aportan información contextual del proceso, tabla de actividades y creación de cada uno de los registros de eventos de las actividades. En este subapartado, es donde se trabaja en el proceso de Transformación dentro de la ETL, es decir, en el punto en el cual nos encontramos actualmente.
- **Data Model Loads:** Trabajos de datos en los cuales se crea el data model que modela las relaciones entre las distintas tablas consideradas. Y se configura como debe de llevarse a cabo la carga de dicho modelo de datos en el sistemas para poder ser procesadas por el motor de procesamiento de datos dentro de Celonis EMS. En este subapartado se trabaja con la parte de la carga dentro de la ETL. Se explicará con mayor detalle en el próximo punto de la memoria.

Volviendo a la herramienta, después de clicar en el apartado de Data Jobs, nos aparecerá la siguiente ventana (véase figura 6.16) en la cual debemos de clicar en el botón azul de la parte superior derecha “+ Add Data Job”

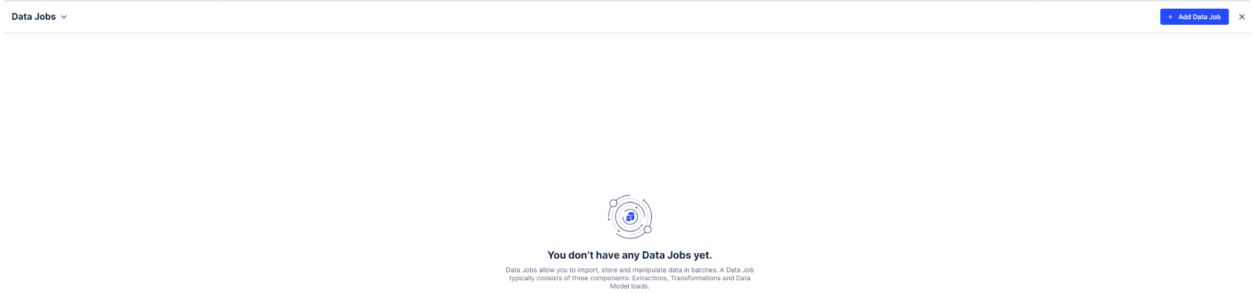


Figura 6.16: Celonis - Data Jobs

Y nos aparecerá la siguiente ventana (véase figura 6.17), en la cual debemos de dar un nombre al trabajo de datos que queremos crear, así como seleccionar la conexión a tener en cuenta para dicho trabajo de datos. En caso de disponer de varios sistemas de información seleccionaremos la conexión con aquel sistema de información al cual queremos tener visibilidad desde el trabajo de datos. Otra opción se corresponde con la conexión de datos global, desde la cual tendremos visibilidad de todos los sistemas que estén conectados (es necesario a la hora de trabajar en el trabajo de datos la conexión y la tabla que nos referenciamos con dicha conexión).

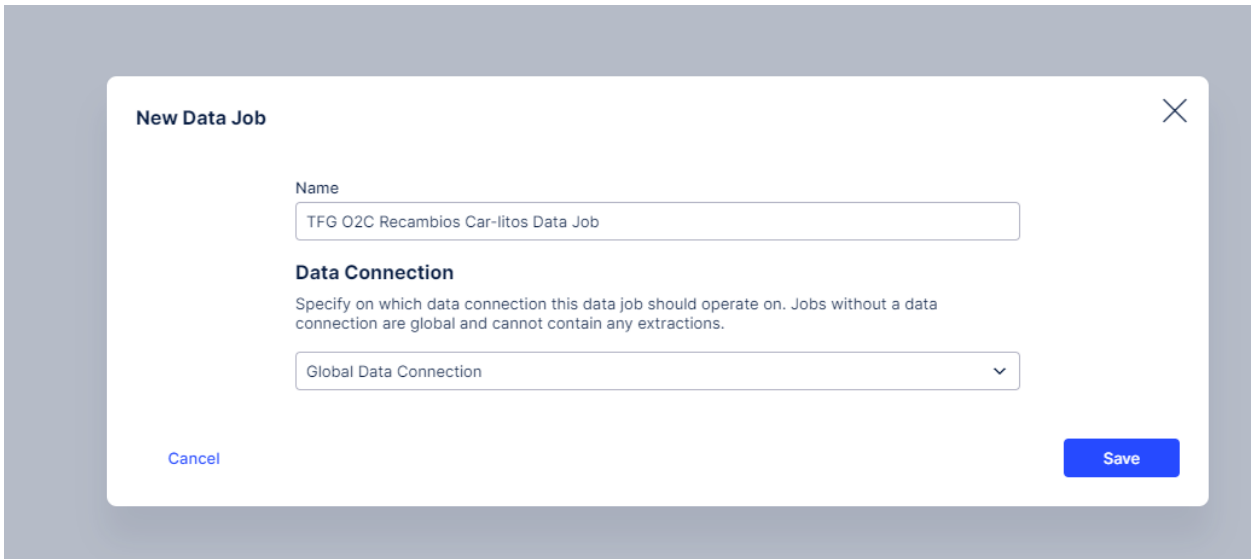


Figura 6.17: Celonis - Nuevo data job

Para nuestro ejemplo, vamos a utilizar la conexión de datos global puesto que no tenemos ninguna conexión directa con sistemas de información y tan solo accedemos a los datos a partir

de los ficheros CSV. El siguiente paso sería dar en el botón azul de “Save” y se creará el Data Job. Es posible tener dentro de un mismo Data Pool varios Data Jobs para disponer de una mejor organización y en caso de existir varios sistemas de información con distintas conexiones en cada Data Job tener una conexión única y visibilidad a tan solo dicha conexión así como agrupar los trabajos de datos de dicho sistema de información en concreto. Para el ejemplo que estamos tratando, tan solo vamos a trabajar con un Data Job y la conexión global.

El siguiente paso, sería crear dentro del Data Job las transformaciones con las cuales podremos modificar, limpiar y transformar los datos en crudo del sistema en actividades o registros de eventos. En este punto, existen cuatro grupos de transformaciones base y que son necesarias que se lleven a cabo.

- **Creación de la tabla de casos:** Por medio de esta transformación se crea una tabla en el sistema la cual cada uno de los registros se corresponde con un caso de estudio del proceso. En el caso del ejemplo que estamos tratando, se corresponde con la tabla de Pedidos.

En dicha transformación se crea a partir de las tablas de datos en crudo de los sistemas de información una tabla con los campos necesarios para el estudio así como si es necesario llevar a cabo joins con otras tablas para aumentar la información de los casos de estudio. Cada registro de dicha tabla va a tener una serie de registros de eventos o actividades las cuales deben ser referenciadas a dicho caso, para ello, un campo esencial en la tabla se corresponde con el denominado CASE_KEY, dicho campo identifica de forma única cada uno de los casos de estudio, es decir, cada registro de la tabla de casos. En general, el CASE_KEY se corresponde con la clave primaria de la tabla de casos. Cada actividad se va a referenciar con el caso que afecta por medio del campo CASE_KEY.

Para el ejemplo que estamos tratando, la tabla de casos como ya hemos mencionado se corresponde con la tabla de Pedidos, cargada directamente del fichero Pedidos.csv. Una buena práctica consiste en no trabajar directamente con las tablas de los datos en crudo del sistema, sino crear tablas propias con los datos “depurados” en los que tan solo tengamos en cuenta los campos necesarios, así como hacer las joins necesarias para solo disponer de los registros necesarios y no sobrecargar el sistema. La nomenclatura que se recomienda seguir es preceder el nombre de la tabla-entidad que modela con la abreviatura del proceso que se está estudiando. De esta forma, vamos a crear la tabla de casos a partir de la tabla Pedidos y la vamos a llamar O2C_PEDIDOS, a la hora de crear la tabla de casos también se va a crear el CASE_KEY, para el ejemplo el CASE_KEY se corresponde con el ID del pedido. Puesto que la tabla Pedidos, ha sido generada por nosotros y ha sido creada de forma que se adapte al caso de estudio que estamos tratando, los campos que consideramos

en la tabla O2C_PEDIDOS son los mismos que la de la tabla Pedidos, pero podríamos quedarnos con los que tan solo nos sean necesarios. La consulta que crea la tabla O2C_PEDIDOS sería la siguiente:

```

1. DROP TABLE IF EXISTS O2C_PEDIDOS;
2.
3. CREATE TABLE IF NOT EXISTS O2C_PEDIDOS AS (
4.     SELECT
5.     P.ID AS _CASE_KEY
6.     , P.*
7.     FROM "Pedidos" AS P
8. );

```

- **Creación de la tabla de Actividades:** Con esta transformación, se crea una tabla en el sistema en la cual cada registro se corresponde con un evento o actividad que ha ocurrido sobre un caso de la tabla de casos. La tabla de actividades se compone de varios campos, pero entre ellos existen 4 que se consideran obligatorios y que ya se han comentado en la parte de la introducción teórica. Dichos campos son:
 - CASE_KEY: Campo que hace referencia al caso que ha sufrido el evento o actividad en cuestión. Este campo sirve como clave foránea para relacionar los casos y las actividades. Debe coincidir con el CASE_KEY de la tabla de casos y en el ejemplo que estamos tratando se registrará el ID del pedido que ha sufrido la actividad.
 - ACTIVITY_ES: Campo que hace referencia al nombre de la actividad que está modelando. El sufijo ES indica que el nombre de la actividad está en castellano, se pueden incluir los campos ACTIVITY_EN y ACTIVITY_DE que almacenarán el nombre de la actividad en inglés y alemán respectivamente, o considerar los idiomas necesarios dependiendo de los usuarios que vayan a hacer uso del estudio.
 - EVENTTIME: Campo que indica la fecha y hora en la cual ha ocurrido la actividad. En ocasiones no se dispone de la hora y tan solo se almacena la fecha en la que ocurre la actividad. En caso de que la actividad en cuestión se componga de una fecha de inicio y otra de fin, por convenio, el EVENTTIME se corresponde con la fecha de fin de la actividad. La fecha de inicio de la actividad se puede modelar incluyendo otra actividad independiente en la cual se indique por medio del campo del nombre de la actividad que es el inicio o incluyendo en la tabla de actividades un campo adicional, llamado START_TIME en el cual se indica la fecha y hora que se ha empezado a realizar la actividad.

- SORTING: Dicho campo es numérico y sirve para ordenar las actividades con respecto al tiempo cuando para un mismo caso existen dos actividades que tengan el mismo EVENTTIME. En dicho supuesto, se considera que la primera actividad que ha tenido lugar es aquella que tenga el menor valor en el campo SORTING.

Aparte de estos campos, la tabla de actividades puede contener una serie de campos extra los cuales sirven para aportar información contextual de la actividad en cuestión. Un ejemplo de dichos campos son: ACTIVITY_DETAIL_ES el cual aporta un detalle de la actividad, DIF_TIME calcula el tiempo que ha durado la actividad en caso de que tenga una fecha de inicio y de fin, es decir, calcula la diferencia entre START_TIME y EVENTTIME. USER_NAME que indica el nombre del usuario que ha realizado la actividad. USER_AUTOMATICO que indica si el usuario que ha realizado la actividad es automático (una máquina) o no. CHANGED_TABLE en caso de que modele una actividad de cambio se indica la tabla que ha cambiado y CHANGED_FIELD el campo que ha cambiado, así como el valor del campo anterior al cambio CHANGED_FROM y el posterior al cambio CHANGED_TO.

Para el ejemplo del caso de uso de 'Recambios Car-litos' el nombre que hemos dado a la tabla de actividades es CEL_O2C_ACTIVITIES, donde por convenio el prefijo CEL indica que es una tabla propia de la herramienta Celonis y al igual que antes el prefijo O2C indica que se corresponde con el proceso O2C. La consulta que define y crea la tabla de casos es la siguiente, en ella se declara cada campo con su tipo de datos correspondiente.

```

1. DROP TABLE IF EXISTS CEL_O2C_ACTIVITIES;
2.
3. CREATE TABLE IF NOT EXISTS CEL_O2C_ACTIVITIES (
4.     "CASE_KEY" VARCHAR(4)
5.     , "ID_PEDIDO" VARCHAR(4)
6.     , "ACTIVITY_DE" VARCHAR(300)
7.     , "ACTIVITY_DETAIL_DE" VARCHAR(300)
8.     , "ACTIVITY_EN" VARCHAR(200)
9.     , "ACTIVITY_DETAIL_EN" VARCHAR(300)
10.    , "ACTIVITY_ES" VARCHAR(200)
11.    , "ACTIVITY_DETAIL_ES" VARCHAR(300)
12.    , "START_TIME" DATETIME
13.    , "EVENTTIME" DATETIME
14.    , "DIF_TIME" INTEGER
15.    , "SORTING" VARCHAR(10)
16.    , "USER_NAME" VARCHAR(50)
17.    , "USER_AUTOMATICO" VARCHAR(10)
18.    , "CHANGED_TABLE" VARCHAR(20)
19.    , "CHANGED_FIELD" VARCHAR(20)
20.    , "CHANGED_FROM" VARCHAR(50)
21.    , "CHANGED_TO" VARCHAR(50)

```

```

22.         , "CHANGED_FROM_FLOAT" FLOAT
23.         , "CHANGED_TO_FLOAT" FLOAT
24.     );

```

- **Creación de las tablas de entidades (Metadata):** Con estas transformaciones (que son varias) se crean las tablas que modelan las entidades que juegan un papel en el proceso que se desea estudiar y que aportan información contextual de la cual podemos derivar conclusiones en el análisis de los datos dimensionando y agrupando los datos por alguna característica en concreto.

En estas transformaciones, se hace la limpieza de los datos en crudo de los sistemas de información y tan solo nos quedamos con la que nos resulte útil para el estudio en cuestión para así no sobrecargar el sistema. Para ello, se suele seleccionar aquellos campos de las tablas que resulten útiles así como aplicar los filtros necesarios sobre dichas tablas.

La finalidad de dichas transformaciones, es crear una tabla con la información en crudo procesada y que se pueda incluir en el Data Model como un diagrama entidad relación y pueda ser relacionada con la tabla de casos. La finalidad de dichas transformaciones es similar a la creación de la tabla de casos.

Para el ejemplo que estamos tratando, hemos creado una transformación que crea una tabla de entidad para cada uno de los ficheros .CSV que hemos cargado en el sistema. Cabe destacar, que como en el ejemplo los ficheros .CSV han sido generados de forma que se adapten al ejemplo, vamos a seleccionar todos los campos del fichero, pero para otros casos de uso podrían seleccionarse sólo un subconjunto de estos. Por convenio, hemos dado el mismo nombre a las tablas de entidades que el fichero .CSV el cual define la entidad a modelar, salvo que se ha añadido el prefijo O2C del proceso que se está estudiando.

Cada tabla de entidad creada se ha incluido en una transformación independiente. Teniendo las siguientes tablas de entidades: O2C_PRODUCTOS, O2C_CLIENTES, O2C_USUARIOS, O2C_DEVOLUCIONES, O2C_BLOQUEOS. Y la consulta SQL que crea cada una de ellas son:

```

1. DROP TABLE IF EXISTS O2C_PRODUCTOS;
2. CREATE TABLE IF NOT EXISTS O2C_PRODUCTOS AS (
3.     SELECT *
4.     FROM "Productos"
5. );
6.
7. DROP TABLE IF EXISTS O2C_CLIENTES;
8.

```

```
9. CREATE TABLE IF NOT EXISTS O2C_CLIENTES AS (  
10.     SELECT *  
11.     FROM "Clientes"  
12. );  
13.  
14. DROP TABLE IF EXISTS O2C_USUARIOS;  
15.  
16. CREATE TABLE IF NOT EXISTS O2C_USUARIOS AS (  
17.     SELECT *  
18.     FROM "Usuarios"  
19. );  
20.  
21. DROP TABLE IF EXISTS O2C_DEVOLUCIONES;  
22.  
23. CREATE TABLE IF NOT EXISTS O2C_DEVOLUCIONES AS (  
24.     SELECT *  
25.     FROM "Devoluciones"  
26. );  
27.  
28. DROP TABLE IF EXISTS O2C_BLOQUEOS;  
29.  
30. CREATE TABLE IF NOT EXISTS O2C_BLOQUEOS AS (  
31.     SELECT *  
32.     FROM "Bloqueos"  
33. );
```

- **Añadir registros de actividades en la tabla de actividades:** Con este tipo de transformaciones, vamos a insertar registros en la tabla de actividades que previamente hemos creado y que la hemos dado el nombre de "CEL_O2C_ACTIVITIES". Es decir, vamos a generar los registros que modelan que ha ocurrido un evento en concreto con un caso de estudio. Cada uno de estos registros se denomina actividad o evento, y como ya se ha comentado anteriormente es necesario que se rellenen los campos CASE_KEY, ACTIVITY_ES, EVENTTIME y SORTING de la tabla de actividades.

Por organización del código, se ha creado una transformación por cada tipo de actividad que se considera en el análisis. De forma que, con cada transformación se va a realizar la inserción de los registros en la tabla de actividades que modelan una cierta actividad en concreto. Para poder modelar una actividad, es usual que sea necesario hacer joins entre varias tablas del sistema y para ello es necesario hacer previamente un análisis de la información que se almacena en el sistema, así como las tablas y las relaciones que existen entre ellas.

Aparte de rellenar los campos obligatorios que hemos comentado, también es de utilidad rellenar el resto de campos de la tabla de actividades para disponer de más información contextual sobre la actividad.

A continuación, a modo de ejemplo se incluye el código correspondiente a la inserción en la tabla de actividades de las actividades de “Aprobación Pedido” y “Cambios”

```

1. INSERT INTO "CEL_O2C_ACTIVITIES" (
2.   "_CASE_KEY"
3.   , "ID_PEDIDO"
4.   , "ACTIVITY_DE"
5.   -- , "ACTIVITY_DETAIL_DE"
6.   , "ACTIVITY_EN"
7.   -- , "ACTIVITY_DETAIL_EN"
8.   , "ACTIVITY_ES"
9.   -- , "ACTIVITY_DETAIL_ES"
10.  -- , "START_TIME"
11.  -- , "EVENTTIME"
12.  -- , "DIF TIME"
13.  -- , "SORTING"
14.  -- , "USER_NAME"
15.  -- , "USER_AUTOMATICO"
16.  -- , "CHANGED_TABLE"
17.  -- , "CHANGED_FIELD"
18.  -- , "CHANGED_FROM"
19.  -- , "CHANGED_TO"
20.  -- , "CHANGED_FROM_FLOAT"
21.  -- , "CHANGED_TO_FLOAT"
22. )
23.
24. SELECT
25. P."_CASE_KEY"
26. , P."ID"
27. , 'Genehmigung der Bestellung' AS "ACTIVITY_DE"
28. , 'Order approval' AS "ACTIVITY_EN"
29. , 'Aprobación Pedido' AS "ACTIVITY_ES"
30. , P."FECHA APROBACIÓN" AS "EVENTTIME"
31. , 3 AS "SORTING"
32. , U."NOMBRE" AS "USER_NAME"
33. , U."AUTOMÁTICO" AS "USER_AUTOMATICO"
34. FROM "O2C_PEDIDOS" P
35. JOIN "O2C_USUARIOS" U ON 1=1
36.   AND U."ID" = P."USUARIO_APRUEBA"
37.
38. INSERT INTO "CEL_O2C_ACTIVITIES" (

```



```

39.      , "ID_PEDIDO"
40.      , "ACTIVITY_DE"
41.      -- , "ACTIVITY_DETAIL_DE"
42.      , "ACTIVITY_EN"
43.      -- , "ACTIVITY_DETAIL_EN"
44.      , "ACTIVITY_ES"
45.      -- , "ACTIVITY_DETAIL_ES"
46.      -- , "START TIME"
47.      , "EVENTTIME"
48.      -- , "DIF TIME"
49.      , " SORTING"
50.      , "USER_NAME"
51.      , "USER_AUTOMATICO"
52.      , "CHANGED_TABLE"
53.      , "CHANGED_FIELD"
54.      , "CHANGED_FROM"
55.      , "CHANGED_TO"
56.      -- , "CHANGED_FROM_FLOAT"
57.      -- , "CHANGED_TO_FLOAT"
58.  )
59.
60.  SELECT
61.  P." CASE_KEY"
62.  , P."ID"
63.  , CASE
64.    WHEN TIPO_CAMBIO = 'Condición Pago'
65.    THEN 'Bedingung Zahlung'
66.    WHEN TIPO_CAMBIO = 'Cantidad'
67.    THEN 'Menge'
68.    ELSE ''
69.  END AS "ACTIVITY_DE"
70.  , 'Change of: ' ||
71.  CASE
72.    WHEN TIPO_CAMBIO = 'Condición Pago'
73.    THEN 'Payment Condition'
74.    WHEN TIPO_CAMBIO = 'Cantidad'
75.    THEN 'Quantity'
76.    ELSE ''
77.  END AS "ACTIVITY_EN"
78.  , 'Cambio de: ' || C.TIPO_CAMBIO AS "ACTIVITY_ES"
79.  , C."FECHA CAMBIO" AS "EVENTTIME"
80.  , 14 AS " SORTING"
81.  , U."NOMBRE" AS "USER_NAME"
82.  , U."AUTOMÁTICO" AS "USER AUTOMATICO"
83.  , 'O2C_PEDIDOS' AS "CHANGED_TABLE"
84.  , CASE
85.    WHEN TIPO_CAMBIO = 'Condición Pago'
86.    THEN 'CONDICION_PAGO'
87.    WHEN TIPO_CAMBIO = 'Cantidad'
88.    THEN 'CANTIDAD'
89.    ELSE NULL
90.  END AS "CHANGED_FIELD"
91.  , C.OLD_VALUE AS "CHANGED_FROM"
92.  , C.NEW_VALUE AS "CHANGED_TO"
93.  FROM "O2C_PEDIDOS" P

```

```
94. JOIN "Historico_Cambios" C ON 1=1
95. AND C."ID_PEDIDO" = P."ID"
96. LEFT JOIN "O2C_USUARIOS" U ON 1=1
97. AND C."USUARIO_CAMBIO" = U."ID";
```

6.5.4. Carga:

La carga se corresponde con el siguiente paso a las transformaciones, y el último del proceso de ETL. Para poder realizar la carga, es necesario haber realizado previamente las transformaciones oportunas sobre los datos en crudo de los sistemas de información, así como disponer la información modelada de acuerdo a disponer de una tabla de casos, una tabla de actividades con su correspondiente registro de actividades así como las tablas de las entidades que van a proporcionar la metadata del proceso.

El proceso de carga consiste en generar un “Data Model” o modelo de datos, que se va a corresponder con un modelo entidad-relación en el cual queda constancia de las relaciones existentes entre las distintas entidades que entran en juego en el proceso y en el análisis. A parte de estas entidades, se deben de relacionar la tabla de casos con la tabla de actividades. Por último, una vez que disponemos del Data Model, se carga toda la información de dichas entidades con sus relaciones en el sistema, en concreto en el EMS de Celonis.

El EMS (siglas en inglés de Execution Management System) de Celonis es el motor de consultas a datos de Celonis y el espacio en el cual podemos generar análisis y paneles o dashboards que permitan visualizar la información de una forma más gráfica. Este apartado le trataremos con mayor detalle en las próximas páginas de la memoria.

Volviendo al caso de uso de ejemplo que estamos tratando, el lugar donde gestionamos la carga de los Data Models en la herramienta Celonis es en la parte derecha de la imagen, clicando sobre el botón azul con el nombre “Create Data Model” (véase figura 6.18).

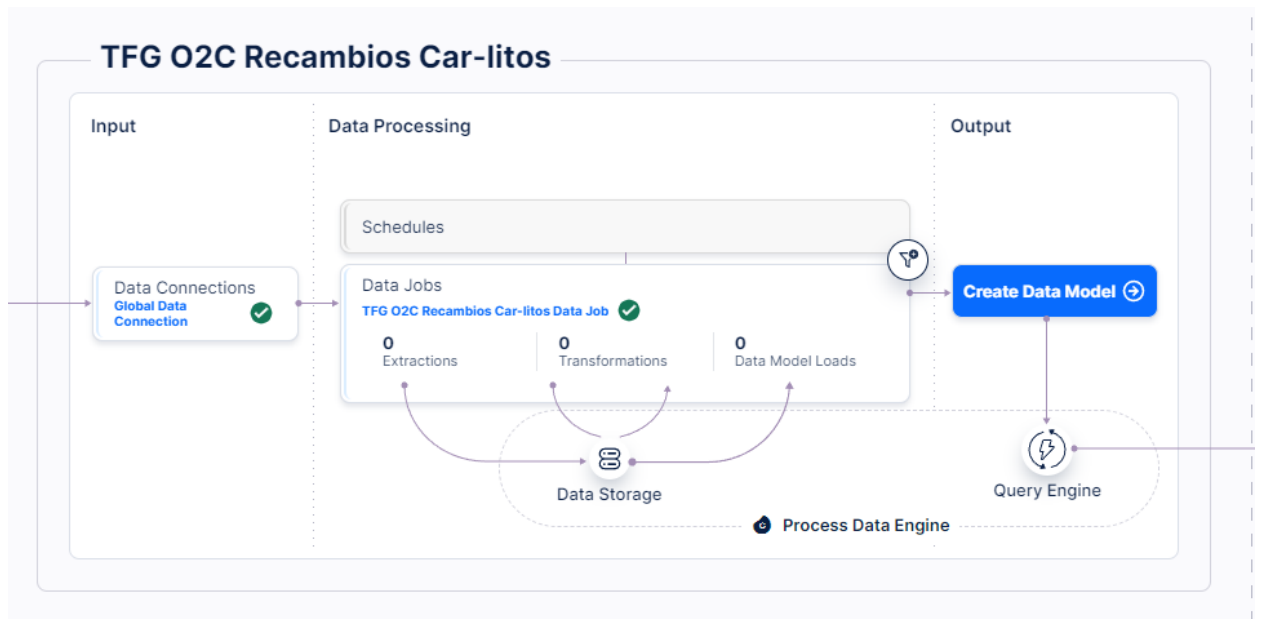


Figura 6.18: Celonis - Crear data model

Una vez que hayamos dado al botón, nos aparecerá la siguiente ventana (véase figura 6.19) en la cual debemos de dar un nombre al Data Model y damos al botón azul “Save” para guardar los cambios.

La imagen muestra una ventana de diálogo titulada "New Data Model". Dentro de la ventana, hay un campo de entrada de texto que contiene el texto "Data Model". Debajo del campo de texto, hay dos botones: "Cancel" a la izquierda y "Save" a la derecha. El botón "Save" es de color azul.

Figura 6.19: Celonis - Nuevo data model

La siguiente pantalla que nos aparecerá será la siguiente (véase figura 6.20), en la cual se deben seleccionar todas aquellas tablas que se desean incluir en el Data Model. En este punto

debemos de escoger todas las tablas de entidades que van a formar la metadata del proceso, así como la tabla de casos y la tabla de actividades. Dando sobre el botón de '+' al lado del nombre de cada una de las tablas las añadiremos. En nuestro caso vamos a seleccionar las tablas: O2C_CLIENTES, O2C_PRODUCTOS, O2C_BLOQUEOS, O2C_DEVOLUCIONES, O2C_PEDIDOS (Tabla de casos) y CEL_O2C_ACTIVITIES.

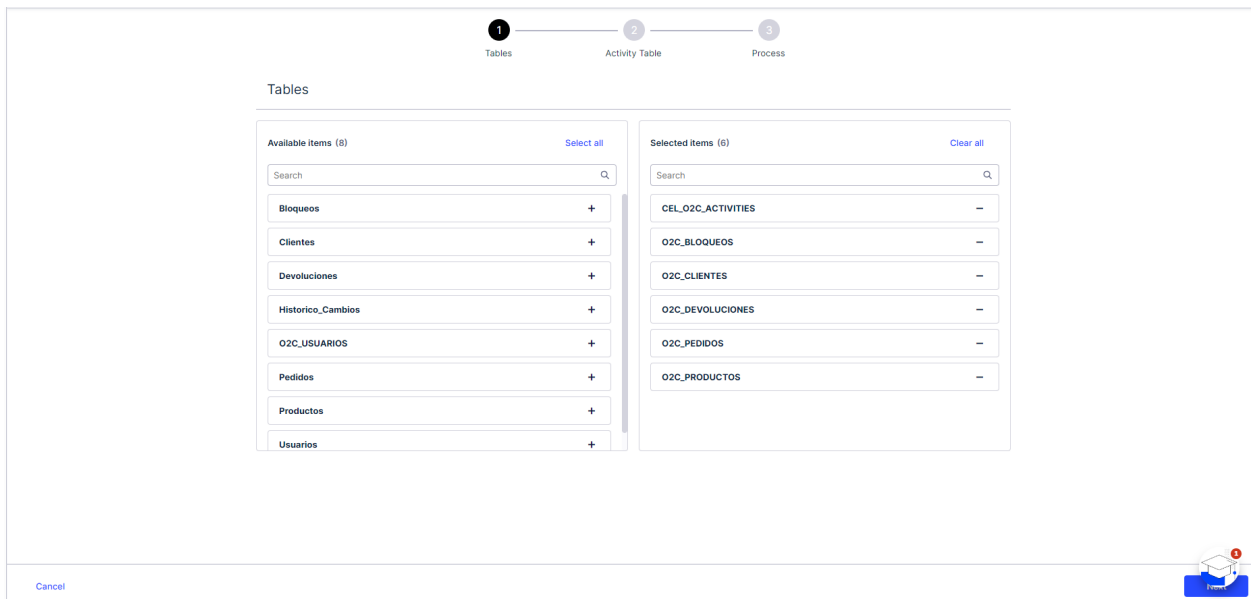
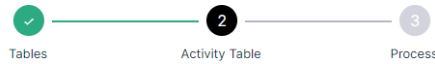


Figura 6.20: Celonis - Selección de tablas

Guardamos los cambios en el botón azul inferior derecho con nombre "Next" y aparecerá la siguiente pantalla (véase figura 6.21):



Activity Table

This table contains all information about the activities of your event log (activity name column, case ID column, timestamp column and optionally a sorting and end timestamp column).

TABLE NAME	Search <input type="text" value=""/>
CEL_O2C_ACTIVITIES	
O2C_BLOQUEOS	
O2C_CLIENTES	
O2C_DEVOLUCIONES	
O2C_PEDIDOS	
O2C_PRODUCTOS	

Figura 6.21: Celonis - Tabla de actividades

En dicha pantalla (figura 6.21), debemos de clicar sobre el nombre de la tabla que se corresponde con la tabla de actividades, en este caso CEL_O2C_ACTIVITIES y seguidamente se pasará a la siguiente pantalla (véase figura 6.22).

Process

Step 1/4: Select Case ID column

Click on the column that contains your case IDs. The case ID usually is a numeric value that is unique and identifies each case in your process.

ABC	_CASE_KEY	ABC	ID_PEDIDO	ABC	ACTIVITY_DE	ABC	ACTIVITY_DETAIL...	ABC	ACTIVITY_EN	ABC	ACTIVITY_DETAIL...	ABC	ACTIVITY_ES	ABC	ACTIVITY_DETAIL...	DATE	START_TIME	DATE	EVENTIME
1		1		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-09-24 18:34:43
1000		1000		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-06-27 00:43:55
1021		1021		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-03-09 11:12:58
1032		1032		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2023-02-10 13:12:27
105		105		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2023-03-12 11:29:32
1058		1058		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2023-10-20 06:46:25
1064	Case ID	1064	Case ID	Best	Case ID	Order	Case ID	Ped	Case ID	<Em	Case ID	<Em	Case ID	<Em	Case ID	<Em			2022-06-04 22:18:40
1066		1066		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-07-29 03:45:15
1094		1094		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-01-23 09:59:15
1101		1101		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-10-08 13:06:16
1107		1107		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2023-09-05 20:52:56
1113		1113		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2023-11-21 12:35:24
1117		1117		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2023-02-17 15:44:50
1134		1134		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-09-01 09:35:10
1137		1137		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-11-11 12:58:01
1138		1138		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2023-04-09 17:23:57
1147		1147		Bestellung erhalten	<Empty>	Order received	<Empty>	Pedido Recibido	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>	<Empty>				2022-07-27 23:30:25

Figura 6.22: Celonis - Selección del campo CASE_KEY

En la cual, debemos de seleccionar sobre la tabla de actividades que acabamos de definir, los campos obligatorios que identifiquen el caso sobre el afecta la actividad, en este ejemplo se corresponde con el campo _CASE_KEY, el campo que identifique el nombre de la actividad, en

este ejemplo `ACTIVITY_ES`, el campo que indica cuándo se ha llevado a cabo la actividad, el campo `EVENTTIME`, y por último el campo que sirve de sorting, campo `SORTING`.

Por último, accedemos a la ventana en la cual creamos el modelo entidad-relación con las tablas que habíamos seleccionado. Es necesario añadir las relaciones entre dichas tablas, indicando los campos que van a actuar como clave foránea. También, es necesario indicar la cardinalidad de la relación y el lado de la relación que se corresponde con cada tabla. Por la estructura de datos y el funcionamiento del EMS, no se permiten relaciones circulares en los Data Models.

Cuando seleccionemos una tabla para añadirla en el Data Model, aparecerá la siguiente ventana (véase figura 6.23), en la cual indicamos el lado de cada tabla en la relación (En este ejemplo la tabla `O2C_PEDIDOS` se corresponde con el lado del 1 y `CEL_O2C_ACTIVITIES` con el lado de la N) y debemos de indicar los campos que actúan como clave foránea (en este ejemplo el campo `_CASE_KEY`)

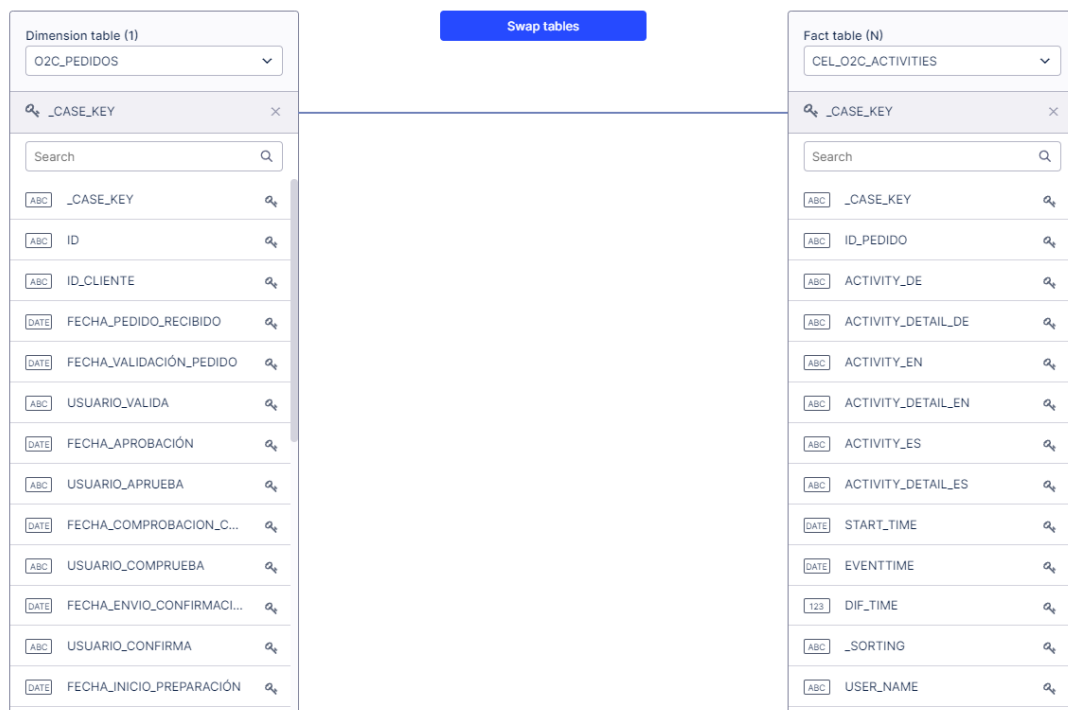


Figura 6.23: Celonis - Relación de tablas en el data model

Realizando este proceso con todas las tablas que sea necesario incluir en el Data Model vamos a ir creando el Data Model con la información del proceso. Para el caso de uso de ejemplo, el Data Model sería el siguiente (véase figura 6.24).

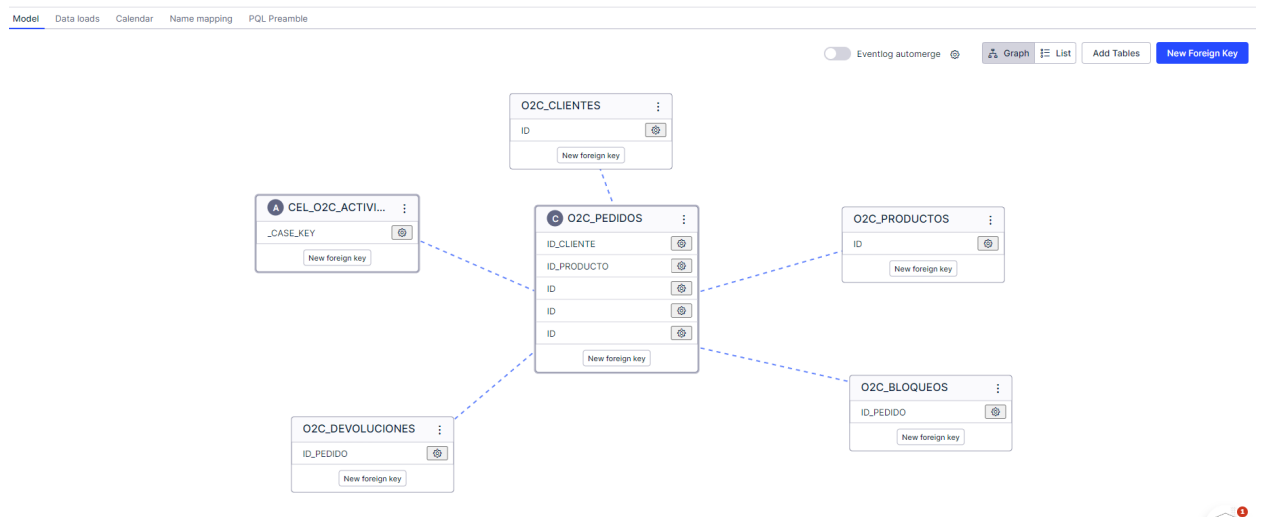


Figura 6.24: Celonis - Data Model

El último paso que debemos de dar es la carga del Data Model, para ello debemos de clicar sobre el botón superior a la izquierda con nombre “Data Loads” e iremos a la ventana de carga en la cual se muestran los logs de las distintas cargas así como el botón azul de la parte superior derecha “Load Data Model” el cual clicando nos permite cargar el Data Model (véase figura 6.25).

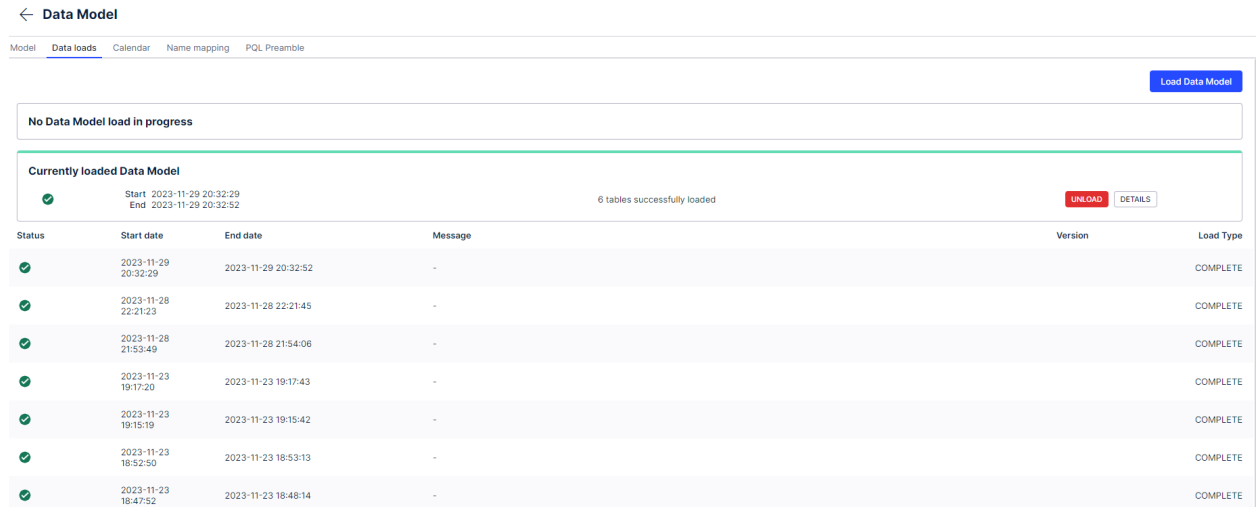


Figura 6.25: Celonis - Carga del data model

6.6. Creación de dashboards

Llegados a este punto, es porque disponemos de un Data Model cargado en el sistema y se han realizado las transformaciones necesarias sobre los datos en crudo de los sistemas de información que almacenan los datos del proceso en cuestión. Es decir, para poder empezar a desarrollar los dashboards o paneles es necesario que previamente se haya llevado a cabo el proceso de ETL sobre los datos.

En este momento es cuando se lleva a cabo la parte gráfica y con la cual podemos llevar a cabo análisis de una forma más sencilla sobre los datos. A partir de dichos análisis que generemos en este momento en un futuro se llevará a cabo un proceso de búsqueda de ineficiencias y causas raíz de los posibles problemas que se puedan descubrir del proceso que se esté estudiando.

Para realizar dichos análisis, se crean distintos paneles de datos o dashboards con información relativa al proceso y al negocio que se esté estudiando. Dichos dashboards deben de incluir la información necesaria de acuerdo a los objetivos que fijó el cliente y deben de permitir el identificar causas raíz e ineficiencias.

Entre los distintos componentes que pueden incluir los dashboards para mostrar datos se encuentran: Tablas, gráficos circulares, histogramas, KPI numéricos, botones y selectores para crear distintos filtros, etc.

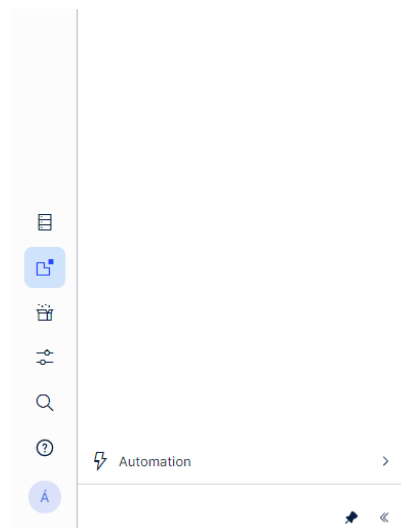
A la hora de crear los distintos dashboards es importante tener presentes los objetivos que se han fijado previamente al estudio para poder centrar en mostrar la información de acorde a dichos objetivos. Lo ideal, es crear un dashboard por cada objetivo que se desee cubrir, a parte

también se deben crear algunos dashboards generales que nos permitan disponer de una visión global del proceso para poder identificar cuellos de botella, ineficiencias y causas raíz que no habían sido identificadas previamente.

Volviendo al caso de uso de ejemplo que estamos tratando sobre la empresa 'Recambios Carlitos' y al uso de la herramienta Celonis vamos a ver cómo se desarrollan los distintos dashboards, para ello, previamente debemos comentar cual es el lenguaje que nos permite trabajar con los datos una vez que están cargados en Celonis EMS en forma de Data Model.

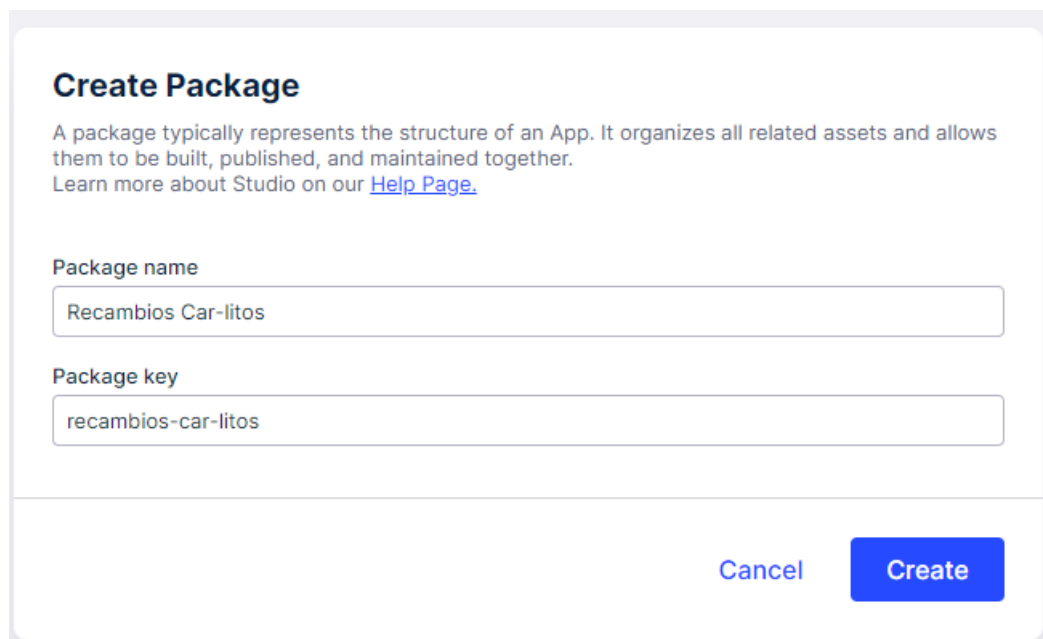
El lenguaje con el cual trabajamos a la hora de realizar los dashboards en Celonis se llama PQL (Process Query Language) (Lenguaje de consulta de procesos). PQL es un lenguaje de dominio específico adaptado a un modelo de datos de procesos especial. PQL permite realizar consultas sobre procesos y son ejecutadas por un motor de consultas especializado. PQL abarca un amplio conjunto de operadores, desde funciones específicas de procesos hasta agregaciones y operadores matemáticos. Su sintaxis se inspira en SQL, pero está especializada en consultas relacionadas con procesos.

A continuación, se muestra como se llevan a cabo los dashboards en Celonis, el primer paso que debemos dar es cambiar en el menú lateral izquierdo el apartado en el que estamos trabajando, anteriormente estábamos trabajando en el apartado Data, ahora debemos de ir al apartado Studio. (Véase figura 6.26)



*Figura 6.26: Celonis -
Apartado Studio*

En el apartado Studio, es donde podremos crear los distintos dashboards y análisis, incorporando los distintos componentes que incluye Celonis, como tablas, gráficos, KPI numéricos, etc. El siguiente paso, es crear un espacio y un paquete (véase figura 6.27) , en el cual se agruparán los distintos análisis que se vayan a realizar en caso de que haya múltiples. Actuarán de directorios. Para ello se dará un nombre a cada uno de ellos. Y posteriormente se creará el análisis, el cual incluirá los distintos dashboards.



Create Package

A package typically represents the structure of an App. It organizes all related assets and allows them to be built, published, and maintained together.
Learn more about Studio on our [Help Page](#).

Package name
Recambios Car-litos

Package key
recambios-car-litos

Cancel Create

Figura 6.27: Celonis - Crear paquete

A la hora de crear el análisis debemos de darle un nombre y además seleccionar cual es el Data Model que se debe de considerar a la hora de realizar las distintas consultas que se llevarán a cabo en el análisis (véase figura 6.28).

NAME ⓘ

Recambios Car-litos analysis

KEY ⓘ

recambios-car-litos-analysis

SELECT DATA MODEL / KNOWLEDGE MODEL

DATA MODEL VARIABLE ⓘ

data_model

KNOWLEDGE MODEL ⓘ

Figura 6.28: Celonis - Crear análisis

Llegados a este punto, es donde se empiezan a crear los dashboards, y es donde entra en juego la imaginación del analista de datos para poder extraer información de interés que se amolden a los objetivos fijados en el proyecto así como identificar nuevas necesidades en caso de inspeccionar los datos y detectar nuevos objetivos. Otro aspecto clave en este punto, es mostrar la información de una forma clara para que pueda entenderse por los usuarios de los dashboards, para ello es necesario mostrar la información de forma simple, los colores también ayudan a centrar la vista en los puntos clave, utilizar gráficos y no solo mostrar la información en tablas. El uso de KPIs numéricos que de un primer vistazo nos permitan medir los objetivos del cliente también ayudan.

Otro aspecto clave que se debe definir son los distintos filtros que se deben de utilizar y que resulten de interés para los análisis. Estos filtros dependen del caso que se esté tomando en el análisis y de la metadata que rodea el proceso en cuestión. En la mayoría de ocasiones dichos filtros nos van a permitir dimensionar la información y agruparla por dichas características a la hora de poder identificar patrones de comportamiento dependiendo de cierta característica.

Centrándonos en el caso de estudio que estamos tratando de 'Recambios Car-litos' se han desarrollado 13 dashboards a través de los cuales podemos tener una visión del proceso que estamos estudiando. Algunos de estos dashboards son generales y nos permiten tener una visión global del proceso, así como conocer información a alto nivel. Sin embargo, otros dashboards se corresponden con paneles específicos en los que se hacen estudios detallados

de acuerdo a algún objetivo, o problemática encontrada en el proceso con el fin de encontrar las causas raíz de dichas problemáticas.

A continuación, se muestra en detalle cada uno de estos dashboards. Para ello, se hará una pequeña descripción del dashboards en el que se indicará el objetivo del dashboard, así como una descripción de la información que muestra cada uno de los KPIs, tablas, gráficos que componen el dashboard.

6.6.1. Executive Dashboard

- **Definición:** Dashboard principal que proporciona una visión a alto nivel de los principales KPIs del proceso que sigue la empresa 'Recambios Car-litos'. Con este dashboard se pretende que los altos cargos de la empresa tengan en un primer vistazo toda la información relevante en su día a día para poder detectar problemas que están ocurriendo sin necesidad de tener que ir al detalle de cada una de las áreas de estudio. El dashboard se divide en tres áreas las cuales más adelante se estudian con dashboards específicos. El primero de los áreas (zona izquierda de la pantalla), se corresponde con los pedidos, con información relevante a ellos. En el área central se muestra información relativa a las devoluciones de pedidos y por último, el área de la derecha se corresponde con el área de la facturación de la empresa, y la automatización del proceso.

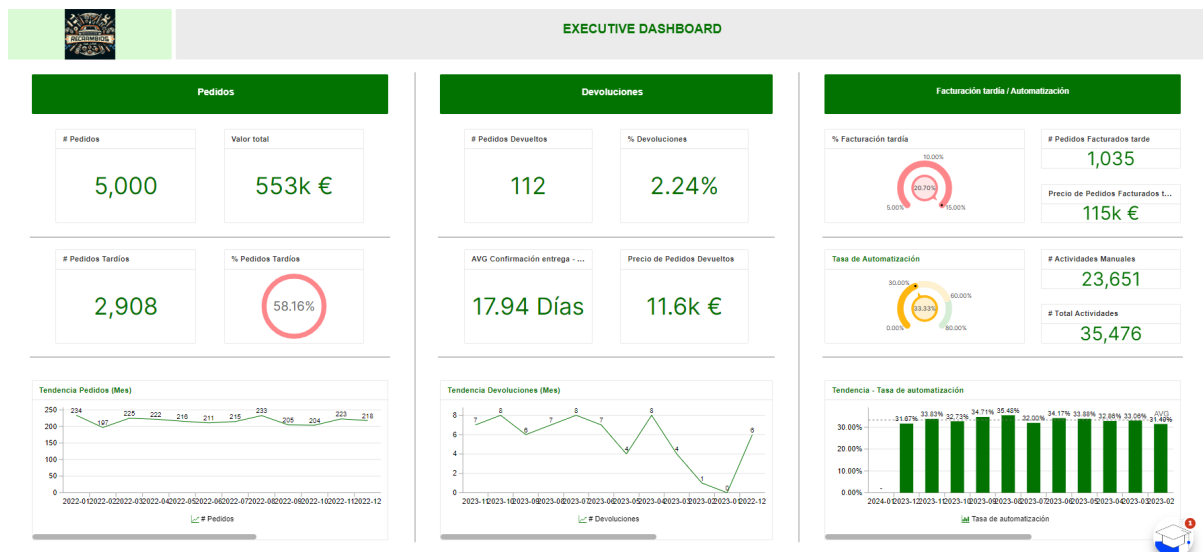


Figura 6.29: Executive Dashboard

- **KPIs y Gráficos Pedidos:** En el área de la izquierda (pedidos), de arriba a abajo, los distintos KPIs que se muestran son:
 - # Pedidos: Número total de pedidos sobre los que se están llevando a cabo el estudio.
 - Fórmula PQL: COUNT_TABLE("O2C_PEDIDOS")
 - Valor total: Valor total de los pedidos sobre los que se está llevando a cabo el estudio.
 - Fórmula PQL: SUM("O2C_PEDIDOS"."PRECIO_TOTAL")
 - # Pedidos tardíos: Número total de pedidos tardíos (pedidos en los que la actividad de 'Confirmación de entrega' es posterior a 'Fecha límite de entrega')
 - Fórmula PQL: SUM(CASE

WHEN DATEDIFF (

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega')) > 0

THEN 1

ELSE 0

END)

- % Pedidos tardíos: Porcentaje de pedidos que son tardíos con respecto al total de los pedidos.

- Fórmula PQL: SUM(CASE

```

WHEN          DATEDIFF          (          DD,
PU_FIRST("O2C_PEDIDOS", "CEL_O2C_ACTIVITIES"."EVENTTIME", "CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),

```

```

PU_FIRST("O2C_PEDIDOS", "CEL_O2C_ACTIVITIES"."EVENTTIME", "CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega')) > 0

```

```

THEN 1

```

```

ELSE 0

```

```

END)

```

```

/ COUNT_TABLE("O2C_PEDIDOS")

```

- Gráfica Tendencia pedidos (Mes): Gráfica que muestra por los distintos meses desde que se ha realizado el estudio de los pedidos el número de pedidos que se han recibido.

- Fórmula PQL:

- EjeX:

```

ROUND_MONTH("O2C_PEDIDOS"."FECHA_PEDIDO_RECIBIDO")

```

- Eje Y: COUNT_TABLE("O2C_PEDIDOS")

- **KPIs y Gráficos Devoluciones:** En el área central (devoluciones), de arriba a abajo, los distintos KPIs que se muestran son:

- # Pedidos Devueltos: Número total de pedidos sobre los que se han realizado devoluciones

- Fórmula PQL: SUM(CASE

```

WHEN

```

```

PU_COUNT("O2C_PEDIDOS", "CEL_O2C_ACTIVITIES"."ACTIVITY_ES", "CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0

```

```

THEN 1

```

```

ELSE 0

```

```

END)

```

- % Devoluciones: Porcentaje de pedidos con devoluciones con respecto al total de los pedidos

- Fórmula PQL:

```
SUM(CASE
  WHEN
  PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0
  THEN 1
  ELSE 0
  END)
/COUNT_TABLE("O2C_PEDIDOS")
```
 - AVG Confirmación entrega - Devolución : Número de días que pasan de media desde que se entrega el pedido hasta que es devuelto por el cliente.
 - Fórmula PQL:

```
AVG(
  DATEDIFF ( DD,
  PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),
  PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución'))
)
```
 - Precio de pedidos devueltos: Importe total de los pedidos que han sido devueltos.
 - Fórmula PQL:

```
SUM(CASE
  WHEN
  PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0
  THEN "O2C_PEDIDOS"."PRECIO_TOTAL"
  ELSE 0
  END)
```
 - Gráfica Tendencia Devoluciones (Mes): Gráfica que muestra cuál ha sido la evolución de los pedidos a lo largo de los distintos meses desde que se ha llevado a cabo el análisis.
 - Fórmula PQL:

- Eje X:
ROUND_MONTH("O2C_PEDIDOS"."FECHA_PEDIDO_RECIBIDO")

- Eje Y: SUM(CASE
 WHEN
 PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución')
 > 0
 THEN 1
 ELSE 0
 END)

- **KPIs y Gráficos Facturación tardía / Automatización:** En el área derecho (facturación tardía y automatización), de arriba a abajo, los distintos KPIs que se muestran son:

- % Facturación tardía: Número total de pedidos sobre los cual se ha facturado tarde (Han pasado más de 5 días desde se lleva a cabo la actividad 'Envío del pedido' y 'Envío de la factura')

- Fórmula PQL: SUM(CASE

- WHEN DATEDIFF (DD,

- PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envío del pedido'),

- PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envío de la factura')) > <%= Facturacion_tardia %>

- THEN 1

- ELSE 0

- END)

- /

- COUNT_TABLE("O2C_PEDIDOS")

- # Pedidos facturados tarde: Número de pedidos en los cuales se ha realizado la facturación tarde, es decir, han pasado más de 5 días desde se lleva a cabo la actividad 'Envío del pedido' y 'Envío de la factura'.

- Fórmula PQL: SUM(CASE

- WHEN DATEDIFF (DD,


```

    PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
    C_ACTIVITIES"."ACTIVITY_ES" = 'Envío del pedido'),

    PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
    C_ACTIVITIES"."ACTIVITY_ES" = 'Envío de la factura')) > <%=
    Facturacion_tardia %>

    THEN 1

    ELSE 0

    END)

```

- Precio de pedidos facturados tarde : Importe de los pedidos que han sido facturados tarde

- Fórmula PQL: SUM(CASE

```

    WHEN DATEDIFF ( DD,

```

```

    PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
    C_ACTIVITIES"."ACTIVITY_ES" = 'Envío del pedido'),

```

```

    PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
    C_ACTIVITIES"."ACTIVITY_ES" = 'Envío de la factura')) > <%=
    Facturacion_tardia %>

```

```

    THEN "O2C_PEDIDOS"."PRECIO_TOTAL"

```

```

    ELSE 0

```

```

    END)

```

- Tasa de automatización: Porcentaje de las actividades del proceso global que se llevan a cabo de forma automática. Indica cómo de automatizado es el proceso.

- Fórmula PQL:SUM(

```

    CASE

```

```

    WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" = 'Sí'

```

```

    THEN 1

```

```

    ELSE 0

```

```

    END

```

```

)

```

```

/

```

```

SUM(

```

```

    CASE

```

```

    WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" IN ('Sí','No')

```

```

    THEN 1

```

```

    ELSE 0

```

```

        END
    )

```

-
- # Actividades manuales: Número de actividades del proceso que se llevan a cabo de forma manual por personas humanas.

- Fórmula PQL:SUM(

```

        CASE

```

```

            WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" = 'No'

```

```

            THEN 1

```

```

            ELSE 0

```

```

        END

```

```

    )

```

- # Total Actividades: Número de actividades totales que se llevan a cabo en el proceso.

Fórmula PQL:SUM(

```

CASE

```

```

    WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" IN ('Sí','No')

```

```

    THEN 1

```

```

    ELSE 0

```

```

END

```

```

)

```

- Gráfica Tendencia Tasa de Automatización: Gráfica que muestra cómo ha sido la evolución de la automatización del proceso a lo largo de los meses, es decir, la evolución del porcentaje de actividades automáticas a lo largo de los meses.

- Fórmula PQL:

- Eje X: ROUND_MONTH("CEL_O2C_ACTIVITIES"."EVENTTIME")

- Eje Y: SUM(

```

CASE

```

```

    WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" = 'Sí'

```

```

    THEN 1

```

```

    ELSE 0

```

```
        END
    )
/
SUM(
CASE
    WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" IN ('Sí','No')
    THEN 1
    ELSE 0
END
)
```

6.6.2. Plazos de entrega:

- **Definición:** Dashboard con el cual se puede realizar un análisis de los plazos de entrega de los pedidos. Permite visualizar si la entrega de los pedidos tiene lugar a tiempo o no cuando se ha comprometido con el cliente. Dicho dashboard, incluye un menú lateral a la izquierda con el cual se puede realizar filtros por distintas características. También, es posible estudiar el retraso de los envíos y determinar las causas raíz de estos. Con dicho dashboard, se cubre el objetivo de estudiar los plazos de entrega con la finalidad de mejorar la satisfacción del cliente y lograr aumentar las ventas.

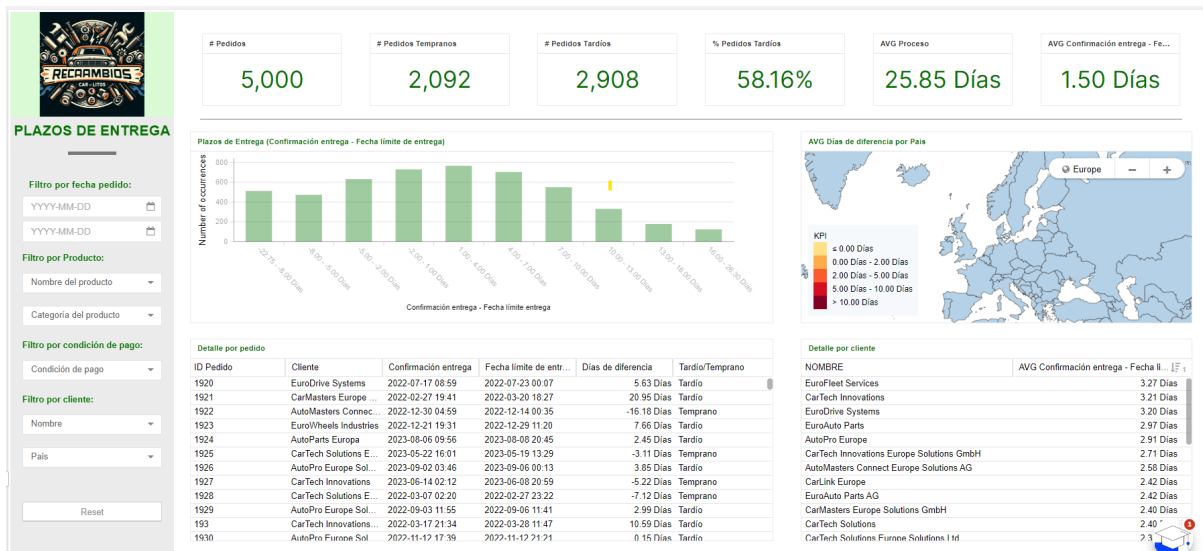


Figura 6.30: Dashboard Plazos de entrega

- **KPIs y Gráficos:** Definición de los KPIs y gráficos incluidos en el dashboard, así como la fórmula empleada para cada uno de ellos. Se especifican de arriba a abajo y de izquierda a derecha por su disposición en el dashboard.
 - # Pedidos: Número de pedidos sobre los que se está llevando a cabo el análisis.
 - Fórmula PQL: COUNT_TABLE("O2C_PEDIDOS")
 - # Pedidos tempranos: Número de pedidos que han sido entregados en tiempo con respecto a la fecha límite de entrega comprometida con el cliente. Es decir, número de pedidos en los que la actividad de 'Confirmación de entrega' es anterior a 'Fecha límite de entrega'
 - Fórmula PQL: SUM(CASE

```

WHEN          DATEDIFF          (          DD,
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega')) <= 0
THEN 1
ELSE 0
END)
    
```

- # Pedidos tardíos: Número de pedidos en los que la actividad ‘Confirmación de entrega’ es posterior a ‘Fecha límite de entrega’. Es decir, el número de pedidos que se han entregado después de la fecha acordada con el cliente.

- Fórmula PQL: SUM(CASE

```

WHEN DATEDIFF ( DD,
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega')) > 0
THEN 1
ELSE 0
END)
    
```

- % Pedidos tardíos: Porcentaje de pedidos que han sido entregados tarde con respecto al total.

- Fórmula PQL:SUM(CASE

```

WHEN DATEDIFF ( DD,
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega')) > 0
THEN 1
ELSE 0
END)
    
```

/ COUNT_TABLE("O2C_PEDIDOS")

- AVG Proceso: Tiempo medio que dura el proceso desde que se tiene la primera actividad de un pedido hasta la última, con independencia de cuáles son estas.

- Fórmula PQL: AVG(

CALC_THROUGHPUT (CASE_START TO CASE_END, REMAP_TIMESTAMPS
("CEL_O2C_ACTIVITIES"."EVENTTIME" , DAYS)))

- AVG Confirmación entrega - Fecha límite de entrega: Tiempo medio que transcurre entre las actividades de 'Confirmación de entrega' y 'Fecha límite de entrega'. Es decir, con dicho KPI podemos calcular el tiempo medio de retraso-adelanto en la entrega de los pedidos con respecto a la fecha fijada con el cliente.

- Fórmula PQL: AVG(

DATEDIFF (DD,
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega'))

- Gráfica Plazos de Entrega (Confirmación entrega - Fecha límite de entrega): Gráfica que muestra la distribución de los días que pasan entre ambas actividades, cuando el número de días es negativo quiere decir que la entrega ha sido anterior a la fecha límite y cuando sea positivo, la entrega es posterior a la fecha límite de entrega. Con esta gráfica podemos ver cual es la distribución de dicha diferencia, para así poder fijar la atención en aquellos en los que la entrega ha sido muy posterior o muy anterior a la fijada para poder estudiar dichos pedidos en detalle.

- Fórmula PQL:

- Eje X: DATEDIFF (DD,

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
"CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
"CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega'))

- Eje Y: COUNT_TABLE("O2C_PEDIDOS")

- Gráfico AVG Días de diferencia por País: Gráfico que permite visualizar por colores, aquellos países que tienen de media un mayor número de días de

retraso. Para ello, se tiene en cuenta el código ISO del país del cliente que ha realizado el pedido, es decir, el campo "O2C_CLIENTES"."CODIGO_ISO".

■ Fórmula PQL: DATEDIFF (DD,

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Fecha límite de entrega'))

- Tabla detalle por pedido: Tabla que muestra en detalle información relativa a cada uno de los pedidos de forma individual. Dispone de varios campos por orden de izquierda a derecha se encuentran el ID del pedido, el cliente del pedido, la fecha de confirmación de entrega, la fecha límite de entrega, los días de diferencia entre ambas fechas y si la entrega ha sido tardía o temprana con respecto a la fecha límite de entrega.
- Tabla detalle por cliente: Tabla que muestra por cliente la media de tiempo que existe en la entrega de los pedidos y la fecha límite. Con esta tabla podemos determinar si existen retrasos de forma continuada con algún cliente en específico, para poder tomar acciones y subsanar dichos retrasos con la intención de mejorar la satisfacción del cliente. Los campos son el nombre del cliente y la media de tiempo entre ambas fechas para todos sus pedidos.

6.6.3. Devoluciones:

- **Definición:** Dashboard que permite analizar cómo se llevan a cabo las devoluciones de los pedidos, para intentar determinar las causas raíz de estas, poder detectar ciertos patrones que hacen que tengan lugar dichas devoluciones con la finalidad de aplicar cambios en el proceso para tratar de subsanar dichas devoluciones y en consecuencia aumentar las ganancias y mejorar la satisfacción del cliente. Se analiza el tiempo medio que se tardan en devolver los productos. Con este dashboard se trata de cubrir el objetivo de disponer de un estudio de las devoluciones de los pedidos, para poder conocer qué clientes son más propensos a realizar devoluciones, así como el motivo de estas. Así como conocer cuántos días de media se realiza la devolución después de haber sido entregado el pedido al cliente.

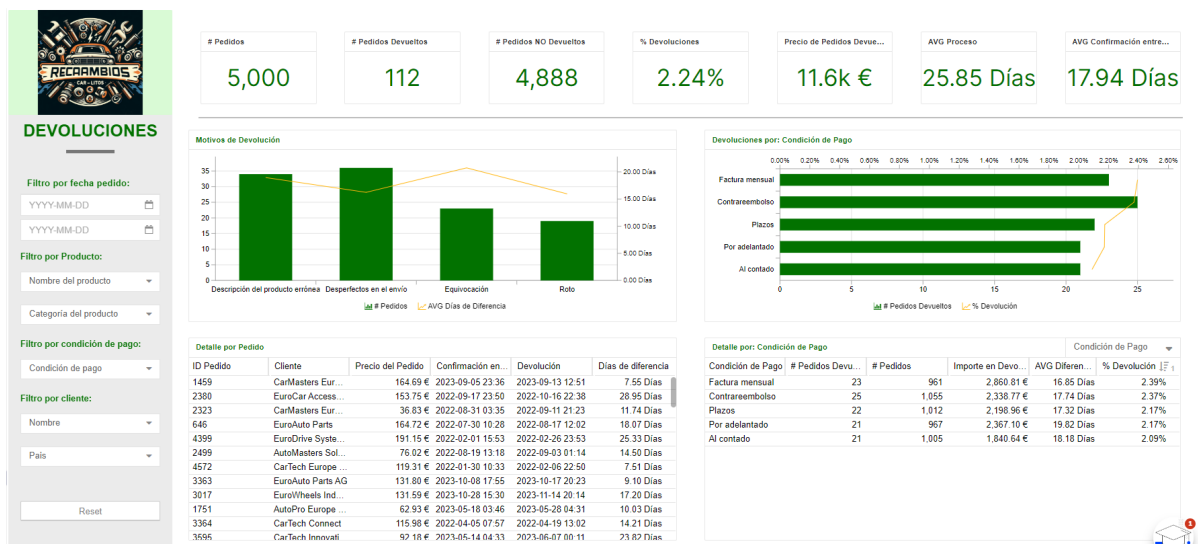


Figura 6.31: Dashboard Devoluciones

- **KPIs y Gráficos:** Definición de los KPIs y gráficos incluidos en el dashboard, así como la fórmula empleada para cada uno de ellos. Se especifican de arriba a abajo y de izquierda a derecha por su disposición en el dashboard.
 - # Pedidos devueltos: Número de pedidos sobre los que se ha llevado a cabo una devolución de los productos que incluía.
 - Fórmula PQL: SUM(CASE


```

WHEN
  PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0

  THEN 1

  ELSE 0

END)

```

- # Pedidos NO devueltos: Número de pedidos que no han tenido devoluciones.

- Fórmula PQL: SUM(CASE

```

WHEN
  PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0

  THEN 0

  ELSE 1

END)

```

- % Devoluciones: Porcentaje de pedidos que han tenido devoluciones con respecto del total de los pedidos.

- Fórmula PQL: SUM(CASE

```

WHEN
  PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0

  THEN 1

  ELSE 0

END)

/ COUNT_TABLE("O2C_PEDIDOS")

```

- Precio de pedidos devueltos: Importe de todos aquellos pedidos que han sido devueltos.

- Fórmula PQL: SUM(CASE

```

WHEN
  PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0

  THEN "O2C_PEDIDOS"."PRECIO_TOTAL"

```

ELSE 0

END)

- AVG Proceso: Tiempo medio que dura el proceso desde que se tiene la primera actividad de un pedido hasta la última, con independencia de cuáles son estas.
 - Fórmula PQL: AVG(

 CALC_THROUGHPUT (CASE_START TO CASE_END, REMAP_TIMESTAMPS

 ("CEL_O2C_ACTIVITIES"."EVENTTIME" , DAYS)))
- AVG Confirmación entrega - Devolución: Media de días que transcurren entre la entrega de los pedidos y su devolución por parte del cliente. Es decir, la media de tiempo que transcurre entre las actividades de 'Confirmación de entrega' y 'Devolución'
 - Fórmula PQL: AVG(

 DATEDIFF (DD,

 PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),

 PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución'))
- Gráfica motivos de devolución: Gráfica que permite analizar cuáles son aquellos motivos más frecuentes por los cuales se realizan devoluciones de los pedidos, indicando por cada motivo de devolución que está tabulado, el número de pedidos que han sido devueltos y la media de tiempo que ha transcurrido desde la entrega del pedido y su devolución.
 - Fórmula PQL:
 - Eje X : "O2C_DEVOLUCIONES"."MOTIVO"
 - Eje Y.1: COUNT_TABLE("O2C_PEDIDOS")
 - Eje Y.2: AVG(DATEDIFF (DD,

 PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Confirmación de entrega'),

 PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución'))
- Devoluciones por X: Gráfica que muestra el número de pedidos devueltos y el % de devolución que se corresponde, agrupada por distintas dimensiones que son dinámicas. En función de la dimensión que se elija en el botón desplegable, la dimensión sobre la cual se agruparán los datos cambiará y en consecuencia se

van a recalcular los datos. Las posibles dimensiones son: producto, categoría del producto, condición de pago, cliente y país.

■ Fórmula PQL:

- Eje X: Campo que depende del botón desplegable escogido.
- Eje Y.1: SUM(CASE

```

    WHEN
    PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVIT
    Y_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') >
    0

    THEN 1

    ELSE 0

    END)
    
```

- Eje Y.2: SUM(CASE

```

    WHEN
    PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVIT
    Y_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') >
    0

    THEN 1

    ELSE 0

    END)

    / COUNT_TABLE("O2C_PEDIDOS")
    
```

- Tabla detalle por pedido: Tabla que muestra un detalle por cada pedido individual sobre los que se han realizado devoluciones. Para ello la tabla está previamente filtrada para que solo aparezcan aquellos pedidos con devoluciones. Los campos de la tabla de izquierda a derecha son: ID del pedido, cliente, precio del pedido, fecha de la confirmación de entrega, fecha de la devolución y días de diferencia entre ambas fechas.

■ Filtro:

FILTER

```

    PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
    O2C_ACTIVITIES"."ACTIVITY_ES" = 'Devolución') > 0;
    
```

- Tabla detalle por X: Tabla con el detalle agrupada por distintas dimensiones que se actualizan de forma dinámica al igual que en el caso de la gráfica de devoluciones por X, en base a lo que se seleccione en el botón desplegable. Los campos de la tabla de izquierda a derecha son: La dimensión sobre la que se

agrupa la información en base a lo que se seleccione en el botón desplegable, el número de pedidos devueltos, el número total de pedidos, el importe en devoluciones, la media de tiempo en días entre la entrega del pedido y su devolución y el % de devolución de pedidos.

6.6.4. Facturación tardía:

- Definición:** Dashboard que permite analizar la facturación de los pedidos y si se ha llevado a cabo a tiempo o no. La facturación comprende el tiempo que transcurre desde el envío del pedido hasta el envío de la factura al cliente. Con este análisis podremos analizar los pedidos y los motivos por los cuales se ha realizado la facturación tarde al cliente y las consecuencias económicas que esto conlleva reduciendo el flujo de caja de la empresa. Por tanto, con dicho dashboard podemos cubrir el objetivo de estudio de la facturación tardía para detectar aquellos puntos que puedan tener ineficiencias y reducir los cuellos de botella, proponiendo acciones que conlleven a reducir la facturación tardía y como consecuencia aumentar el flujo de caja de la empresa. Cabe destacar, que depende de las políticas internas de las empresas se considera que una facturación es tardía una vez que han pasado un cierto número de días que pueden variar de unas empresas a otras, para ello en el dashboard se ha incluido un selector de días en el cual se debe de incluir cuál es el margen en el cual se considera que la facturación se ha realizado de forma correcta. Para el ejemplo, hemos considerado que el margen permitido es de 5 días, una vez que hayan pasado esos 5 días desde que se envía el pedido hasta que se envía la factura se considera facturación tardía.

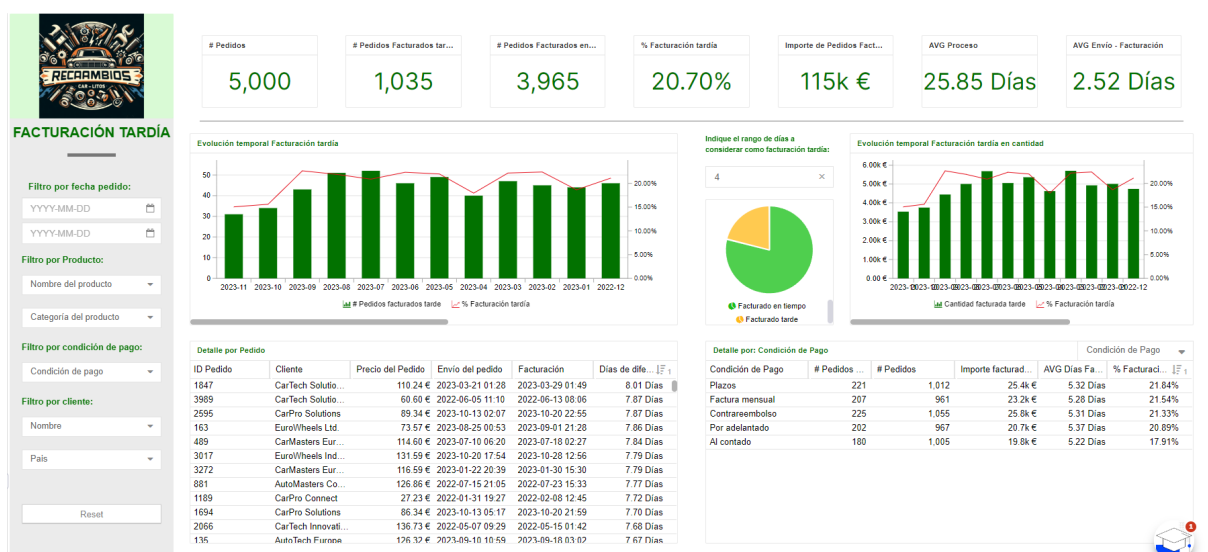


Figura 6.32: Dashboard Facturación tardía

- **KPIs y Gráficos:** Definición de los KPIs y gráficos incluidos en el dashboard, así como la fórmula empleada para cada uno de ellos. Se especifican de arriba a abajo y de izquierda a derecha por su disposición en el dashboard.
 - # Pedidos facturados tarde: Número de pedidos facturados fuera de plazo. Es decir, número de pedidos en los que han transcurrido más de el número de días que se considera como correcta la facturación (en nuestro ejemplo 5 días, pero la variable que almacena el número de días se llama <%=Facturacion_tardia%>) entre las actividades de 'Envío del pedido' y 'Envío de la factura'.
 - Fórmula PQL: SUM(CASE


```

              WHEN DATEDIFF ( DD,
              PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
              C_ACTIVITIES"."ACTIVITY_ES" = 'Envío del pedido'),
              PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
              C_ACTIVITIES"."ACTIVITY_ES" = 'Envío de la factura')) > <%=
              Facturacion_tardia %>
              THEN 1
              ELSE 0
              END)
              
```
 - # Pedidos Facturados en tiempo: Número de pedidos que han sido facturados en tiempo, es decir, el número de días entre las actividades de 'Envío del pedido' y 'Envío de la factura' se encuentra dentro del margen.
 - Fórmula PQL: SUM(CASE


```

              WHEN DATEDIFF ( DD,
              PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
              C_ACTIVITIES"."ACTIVITY_ES" = 'Envío del pedido'),
              PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
              C_ACTIVITIES"."ACTIVITY_ES" = 'Envío de la factura')) > <%=
              Facturacion_tardia %>
              THEN 0
              ELSE 1
              END)
              
```
 - % facturación tardía: Porcentaje de pedidos que han sido facturados tarde con respecto del total.
 - Fórmula PQL: SUM(CASE

```

WHEN DATEDIFF ( DD,
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Envio del pedido'),
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Envio de la factura')) > <%=
Facturacion_tardia %>

THEN 1

ELSE 0

END)

/ COUNT_TABLE("O2C_PEDIDOS")
    
```

- Precio de Pedidos Facturados tarde: Importe total de todos aquellos pedidos que han sido facturados tarde.

- Fórmula PQL: SUM(CASE

```

WHEN DATEDIFF ( DD,
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Envio del pedido'),
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2
C_ACTIVITIES"."ACTIVITY_ES" = 'Envio de la factura')) > <%=
Facturacion_tardia %>

THEN "O2C_PEDIDOS"."PRECIO_TOTAL"

ELSE 0

END)
    
```

- AVG Proceso: Tiempo medio que dura el proceso desde que se tiene la primera actividad de un pedido hasta la última, con independencia de cuáles son estas.

- Fórmula PQL: AVG(

```

CALC_THROUGHPUT ( CASE_START TO CASE_END, REMAP_TIMESTAMPS
("CEL_O2C_ACTIVITIES"."EVENTTIME" , DAYS ) )
    
```

- AVG Envío - Facturación: Medía de tiempo en días en la facturación, es decir, media de tiempo en días desde que se envía el pedido hasta que se realiza la factura.

- Fórmula PQL: AVG(

```

DATEDIFF ( DD,
    
```

```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio del pedido'),
```

```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio de la factura'))))
```

- Gráfica evolución temporal facturación tardía: Gráfica que muestra el número de pedidos que han sido facturados tarde en cada uno de los meses y el porcentaje de facturación tardía correspondiente a cada mes.

- Fórmula PQL:

- EjeX:

```
ROUND_MONTH("O2C_PEDIDOS"."FECHA_PEDIDO_RECIBIDO")
```

- Eje Y.1: SUM(CASE

```
WHEN DATEDIFF ( DD,
```

```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
  CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio del pedido'),
  PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
  CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio de la
  factura')) > <%= Facturacion_tardia %>
```

```
THEN 1
```

```
ELSE 0
```

```
END)
```

- Eje Y.2: SUM(CASE

```
WHEN DATEDIFF ( DD,
```

```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
  CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio del pedido'),
  PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
  CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio de la
  factura')) > <%= Facturacion_tardia %>
```

```
THEN 1
```

```
ELSE 0
```

```
END)
```

```
/ COUNT_TABLE("O2C_PEDIDOS")
```

- Gráfica evolución temporal facturación tardía en cantidad: Gráfica que muestra el importe de los pedidos pedidos que han sido facturados tarde en cada uno de los meses y el porcentaje de facturación tardía correspondiente a cada mes.

- Fórmula PQL:

- Eje X:

```
ROUND_MONTH("O2C_PEDIDOS"."FECHA_PEDIDO_RECIBIDO")
```

- Eje Y.1: SUM(CASE

```
WHEN DATEDIFF ( DD,
```

```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
  CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio del pedido'),
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIM
E","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio de la
factura')) > <%= Facturacion_tardia %>
```

```
THEN "O2C_PEDIDOS"."PRECIO_TOTAL"
```

```
ELSE 0
```

```
END)
```

- Eje Y.2: SUM(CASE

```
WHEN DATEDIFF ( DD,
```

```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME",
  CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio del pedido'),
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIM
E","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envio de la
factura')) > <%= Facturacion_tardia %>
```

```
THEN 1
```

```
ELSE 0
```

```
END)
```

```
/ COUNT_TABLE("O2C_PEDIDOS")
```

- Tabla detalle por pedido: Tabla que muestra el detalle por cada uno de los pedidos que han sido facturados tarde, la tabla se encuentra filtrada para que solo se muestran los datos de todos aquellos pedidos que han sido facturados tarde. De izquierda a derecha los campos son: ID del pedido, cliente del pedido, precio del pedido, fecha del envío del pedido, fecha de facturación del pedido, días de diferencia entre ambas fechas.

- Filtro: FILTER DATEDIFF (DD,


```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envío del pedido'),
```

```
PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Envío de la factura')) > <%= Facturacion_tardia %>
```

- Tabla de detalle por X: Tabla que muestra información agrupada por una cierta dimensión que es posible seleccionar por medio de un botón desplegable y en base a dicha dimensión agrupada se realizan los cálculos. Las posibles dimensiones seleccionables son: producto, categoría del producto, condición de pago, cliente y país. Las columnas de izquierda a derecha son: la dimensión X por la cual se agrupan los datos, el número de pedidos facturados tarde, el número total de pedidos, el importe facturado tarde, la media de días facturado tarde en aquellos casos que se ha facturado tarde y el porcentaje de facturación tardía.

6.6.5. Bloqueos:

- **Definición:** Dashboard que permite analizar el impacto que tienen los distintos bloqueos dentro del proceso de O2C, y el tiempo que conllevan hasta que se desbloquean. Para este ejemplo, existen dos tipos de bloqueos, bloqueo de stock y bloqueo de crédito. Se pueden analizar en dicho dashboard de forma conjunta o de forma individual cada uno de ellos. Con dicho dashboard se trata de analizar las causas raíz y el impacto que estos tienen en el proceso. Con este dashboard se aborda el objetivo de estudiar las actividades indeseadas de bloqueos, las cuales producen que se aumente el tiempo del proceso y llevan a cabo una mayor implicación de los trabajadores, lo que supone costes adicionales.

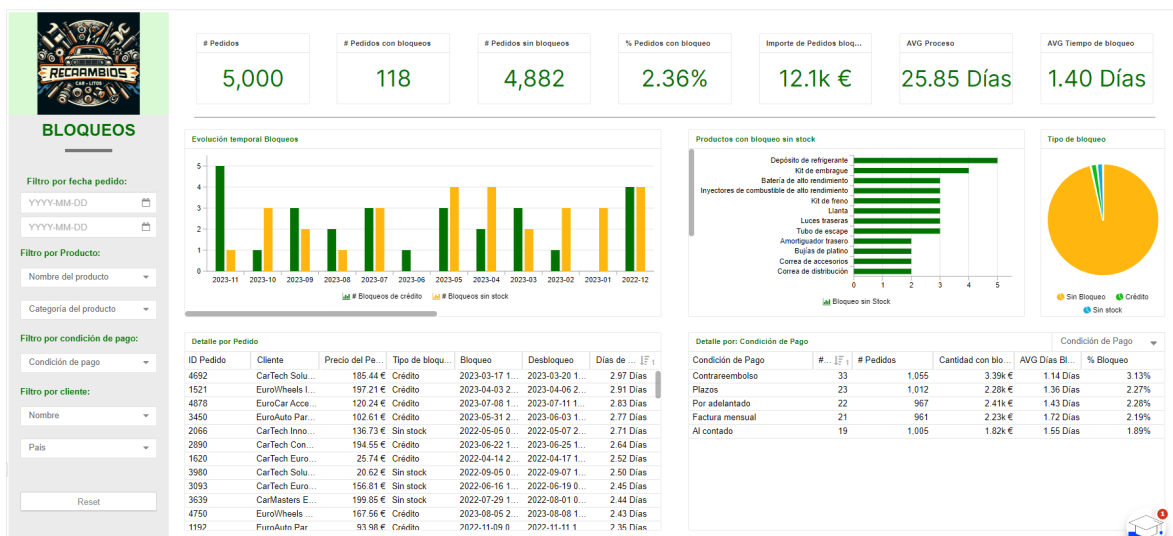


Figura 6.33: Dashboard Bloqueos

- **KPIs y Gráficos:** Definición de los KPIs y gráficos incluidos en el dashboard, así como la fórmula empleada para cada uno de ellos. Se especifican de arriba a abajo y de izquierda a derecha por su disposición en el dashboard.
 - # Pedidos con bloqueos: Número de pedidos que han tenido un bloqueo de cualquiera de los dos tipos a lo largo del proceso.
 - Fórmula PQL: SUM(CASE
WHEN
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo') > 0
THEN 1
ELSE 0
END)
 - # Pedidos sin bloqueos: Número de pedidos que no han tenido bloqueos de ningún tipo en el proceso.
 - Fórmula PQL:SUM(CASE
WHEN
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo') > 0
THEN 0
ELSE 1
END)
 - % Pedidos con bloqueo: Porcentaje que suponen los pedidos que han tenido bloqueos con respecto del total de pedidos.
 - Fórmula PQL: SUM(CASE
WHEN
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo') > 0
THEN 1
ELSE 0
END)
/ COUNT_TABLE("O2C_PEDIDOS")
 - Importe de pedidos bloqueados: Importe en € que suponen aquellos pedidos que han sido bloqueados.
 - Fórmula PQL: SUM(CASE
WHEN
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo') > 0

```

THEN "O2C_PEDIDOS"."PRECIO_TOTAL"
ELSE 0
END)

```

- AVG Tiempo con bloqueo: Media de tiempo en días que transcurre entre que se bloquea y se desbloquea un pedido.

- Fórmula PQL:AVG(

```

DATEDIFF ( DD,

```

```

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O
2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo'),

```

```

PU_FIRST("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."EVENTTIME","CEL_O
2C_ACTIVITIES"."ACTIVITY_ES" = 'Desbloqueo'))

```

- Gráfica evolución temporal bloqueos: Gráfica que muestra el número de bloqueos que existen de cada uno de los dos tipos (crédito y sin stock) a lo largo de cada uno de los meses de estudio, lo que permite ver una evolución temporal o posible picos en los que existan mayor número de bloqueos.

- Fórmula PQL:

- EjeX:

```

ROUND_MONTH("O2C_PEDIDOS"."FECHA_PEDIDO_RECIBIDO")

```

- Eje Y.1: SUM(CASE

```

WHEN

```

```

PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_E
S","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo' AND
"CEL_O2C_ACTIVITIES"."ACTIVITY_DETAIL_ES" = 'Crédito') > 0

```

```

THEN 1

```

```

ELSE 0

```

```

END)

```

- Eje Y.2: SUM(CASE

```

WHEN

```

```

PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_E
S","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo' AND
"CEL_O2C_ACTIVITIES"."ACTIVITY_DETAIL_ES" = 'Sin stock') > 0

```

```

THEN 1

```

```

ELSE 0

```

```

END)

```

- Gráfica productos con bloqueo sin stock: Gráfica que muestra por cada tipo de producto el número de bloqueos de tipo stock ha tenido de forma histórica, lo que permite identificar aquellos productos que es más normal que se queden sin stock y en consecuencia tomar acciones para evitar que este tipo de bloqueo ocurra.
 - Fórmula PQL:
 - Eje X: "O2C_PRODUCTOS"."NOMBRE"
 - Eje Y: SUM(CASE
 - WHEN
 - PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo' AND "CEL_O2C_ACTIVITIES"."ACTIVITY_DETAIL_ES" = 'Sin stock') > 0
 - THEN 1
 - ELSE 0
 - END)
- Tabla detalle por pedido: Tabla que muestra el detalle por cada uno de los pedidos que han sufrido un tipo de bloqueo a lo largo del proceso, la tabla se encuentra filtrada para que solo se muestran los datos de todos aquellos pedidos que han tenido un bloqueo. De izquierda a derecha los campos son: ID del pedido, cliente del pedido, precio del pedido, fecha del bloqueo, fecha de debloqueo, días de diferencia entre ambas fechas.
 - Filtro:FILTER
 - PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_O2C_ACTIVITIES"."ACTIVITY_ES" = 'Bloqueo') > 0
- Tabla de detalle por X: Tabla que muestra información agrupada por una cierta dimensión que es posible seleccionar por medio de un botón desplegable y en base a dicha dimensión agrupada se realizan los cálculos. Las posibles dimensiones seleccionables son: producto, categoría del producto, condición de pago, cliente y país. Las columnas de izquierda a derecha son: la dimensión X por la cual se agrupan los datos, el número de pedidos con bloqueos, el número total de pedidos, el importe con bloqueo, la media de días que dura el bloqueo en los pedidos y el porcentaje de pedidos con bloqueos.

6.6.6. Cambios:

- **Definición:** Dashboard que permite analizar el impacto que tienen los distintos cambios dentro del proceso de O2C, y el tiempo que conllevan el efectuar dichos cambios en cuanto a tiempo en el proceso. Para este ejemplo, existen dos tipos de cambios sobre el pedido, cambios de cantidad y de condición de pago. Se pueden analizar en dicho dashboard de forma conjunta o de forma individual cada uno de ellos. Con dicho dashboard, se trata de analizar las causas raíz y el impacto que estos tienen en el proceso. Con este dashboard, se aborda el objetivo de estudiar las actividades de cambios sobre los pedidos, las cuales producen que se aumente el tiempo del proceso y llevan a cabo una mayor implicación de los trabajadores, lo que supone costes adicionales.

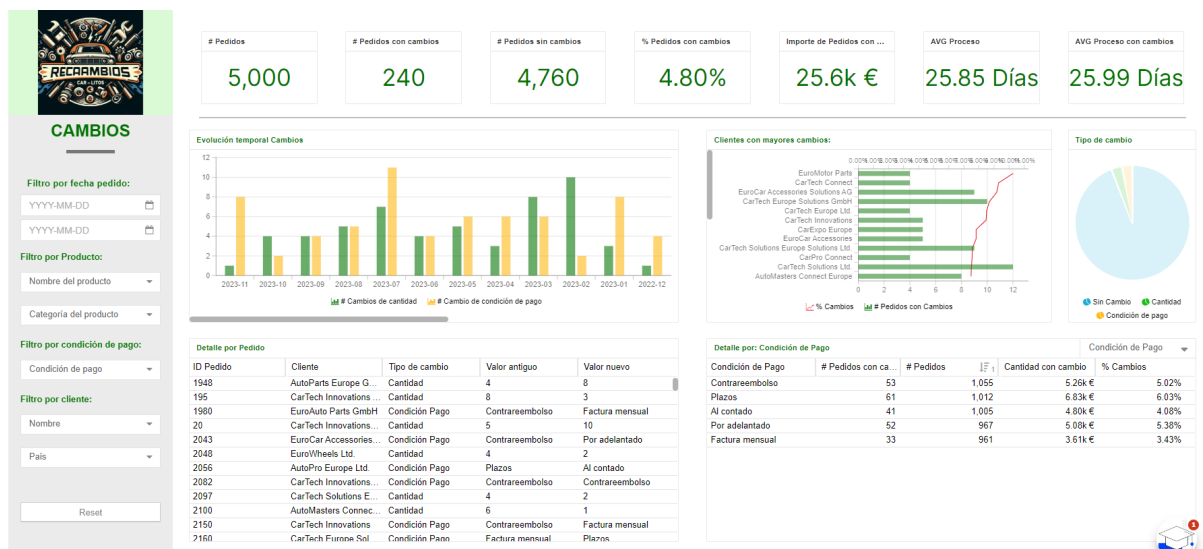


Figura 6.34: Dashboard Cambios

- **KPIs y Gráficos:** Definición de los KPIs y gráficos incluidos en el dashboard, así como la fórmula empleada para cada uno de ellos. Se especifican de arriba a abajo y de izquierda a derecha por su disposición en el dashboard.
 - # Pedidos con cambios: Número de pedidos que tienen un tipo de cambio, independientemente de cual sea:
 - Fórmula PQL: `SUM(CASE WHEN PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_`

- ```
O2C_ACTIVITIES"."ACTIVITY_ES" LIKE 'Cambio de:%') > 0
THEN 1
ELSE 0
END)
```
- # Pedidos sin cambios: Número de pedidos que no tienen cambios en el proceso completo del pedido.
    - Fórmula PQL: SUM(CASE
 

```
WHEN
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" LIKE 'Cambio de:%') > 0
THEN 0
ELSE 1
END)
```
  - % Pedidos con cambios: Porcentaje de pedidos que tienen cambios con respecto del total de pedidos.
    - Fórmula PQL:SUM(CASE
 

```
WHEN
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" LIKE 'Cambio de:%') > 0
THEN 1
ELSE 0
END)
/ COUNT_TABLE("O2C_PEDIDOS")
```
  - Importe de pedidos con cambios: Importe de los pedidos que han tenido algún tipo de cambio.
    - Fórmula PQL:SUM(CASE
 

```
WHEN
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" LIKE 'Cambio de:%') > 0
THEN "O2C_PEDIDOS"."PRECIO_TOTAL"
ELSE 0
END)
```
  - AVG Proceso con cambios: Media de días que dura el proceso en aquellos pedidos que tienen cambios.
    - Fórmula PQL: AVG(CALC\_THROUGHPUT ( CASE\_START TO CASE\_END,
 REMAP\_TIMESTAMPS ( "CEL\_O2C\_ACTIVITIES"."EVENTTIME" , DAYS ) ) )
    - Filtro: FILTER

```
PU_COUNT("O2C_PEDIDOS","CEL_O2C_ACTIVITIES"."ACTIVITY_ES","CEL_
O2C_ACTIVITIES"."ACTIVITY_ES" LIKE 'Cambio de:%') > 0
```

- Gráfica evolución temporal cambios: Gráfica que muestra el número de cambios que existen de cada uno de los dos tipos (cantidad y condición de pago) a lo largo de cada uno de los meses de estudio, lo que permite ver una evolución temporal o posible picos en los que existan mayor número de cambios.

- Fórmula PQL:

- EjeX:

- ROUND\_MONTH("O2C\_PEDIDOS"."FECHA\_PEDIDO\_RECIBIDO")

- Eje Y.1: SUM(CASE

- WHEN

- PU\_COUNT("O2C\_PEDIDOS","CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES",
        - "CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES" = 'Cambio de: Cantidad')
        - > 0

- THEN 1

- ELSE 0

- END)

- Eje Y.2: SUM(CASE

- WHEN

- PU\_COUNT("O2C\_PEDIDOS","CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES",
        - "CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES" = 'Cambio de: Condición
        - Pago') > 0

- THEN 1

- ELSE 0

- END)

- Gráfica clientes con mayores cambios: Gráfica que muestra aquellos clientes que han realizado mayor número de cambios sobre sus pedidos, se muestra de forma absoluta el número de cambios que han realizado sobre sus pedidos, así de forma porcentual.

- Fórmula PQL:

- Eje Y : "O2C\_CLIENTES"."NOMBRE"

- Eje X.1: SUM(CASE

- WHEN

- PU\_COUNT("O2C\_PEDIDOS","CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES",
        - "CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES" LIKE 'Cambio de:%') > 0

- THEN 1

- ELSE 0

- END)

- Eje X.2: SUM(CASE

- WHEN

- PU\_COUNT("O2C\_PEDIDOS","CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES",
        - "CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES" LIKE 'Cambio de:%') > 0

```

THEN 1
ELSE 0
END)
/ COUNT_TABLE("O2C_PEDIDOS")

```

- Tabla detalle por pedido: Tabla que muestra el detalle por cada uno de los pedidos que han sufrido un tipo de cambio a lo largo del proceso, la tabla se encuentra filtrada para que solo se muestran los datos de todos aquellos pedidos que han tenido cambios. De izquierda a derecha los campos son: ID del pedido, cliente del pedido, tipo de cambio, valor antes del cambio, valor después del cambio.
  - Filtro: FILTER  
 PU\_COUNT("O2C\_PEDIDOS", "CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES", "CEL\_O2C\_ACTIVITIES"."ACTIVITY\_ES" LIKE 'Cambio de:%') > 0
- Tabla de detalle por X: Tabla que muestra información agrupada por una cierta dimensión que es posible seleccionar por medio de un botón desplegable y en base a dicha dimensión agrupada se realizan los cálculos. Las posibles dimensiones seleccionables son: producto, categoría del producto, condición de pago, cliente y país. Las columnas de izquierda a derecha son: la dimensión X por la cual se agrupan los datos, el número de pedidos con cambios, el número total de pedidos, el importe con cambios y el porcentaje de pedidos con cambios.

#### 6.6.7. Automatización:

- **Definición:** Dashboard que permite analizar el grado de automatización que existe en el proceso de O2C. Es decir, permite analizar cuantas actividades se realizan de forma automática por medio de procesos automáticos y cuantas es necesario involucrar a una persona en el proceso.

La automatización de los procesos permiten asumir un mayor volumen de pedidos, así como una reducción de costes. Existen actividades en las cuales es más sencillo el automatizarlas y otras en las que es necesario que se realicen de forma manual, por ello, se dispone de un detalle por actividades de su tasa de automatización.



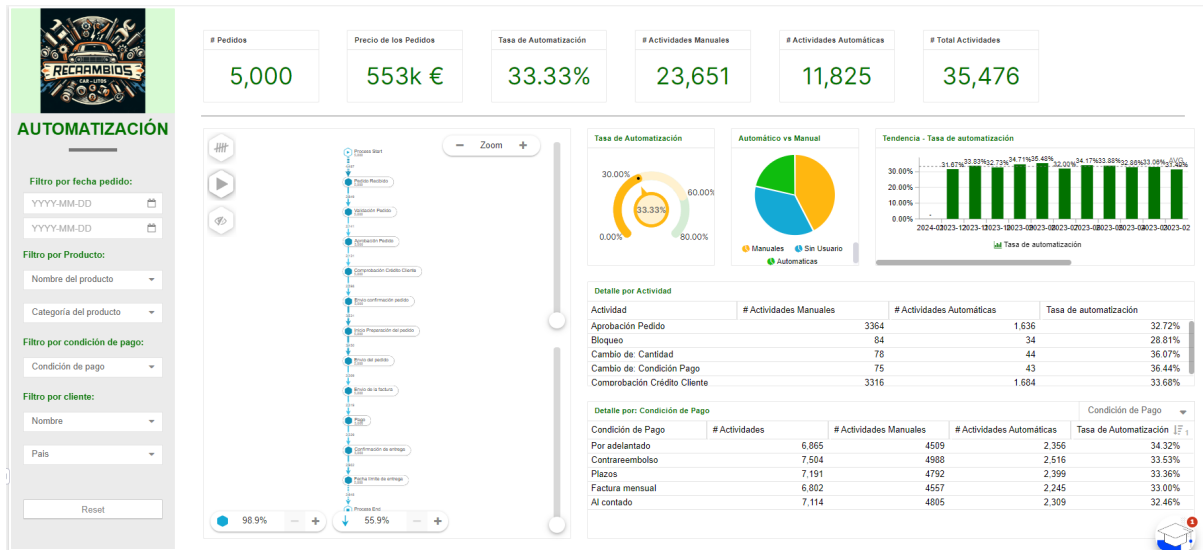


Figura 6.35: Dashboard Automatización

- **KPIs y Gráficos:** Definición de los KPIs y gráficos incluidos en el dashboard, así como la fórmula empleada para cada uno de ellos. Se especifican de arriba a abajo y de izquierda a derecha por su disposición en el dashboard.
  - Tasa de automatización: Porcentaje de automatización que tiene el proceso, es decir, el número de actividades que se llevan a cabo de forma automática con respecto al total de actividades.
    - Fórmula PQL:
 

```
SUM(
CASE
WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" = 'Sí'
THEN 1
ELSE 0
END
)/
SUM(
CASE
WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" IN ('Sí','No')
THEN 1
ELSE 0
END)
```
  - # Actividades manuales: Número de actividades que se llevan a cabo de forma manual dentro del proceso.
    - Fórmula PQL:
 

```
SUM(
CASE
WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" = 'No'
```

- ```

        THEN 1
        ELSE 0
        END)
    
```
- # Actividades automáticas: Número de actividades que se llevan a cabo de forma automática dentro del proceso.
 - Fórmula PQL:SUM(


```

                            CASE
                            WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" = Sí
                            THEN 1
                            ELSE 0
                            END)
                    
```
 - # Total Actividades: Número total de actividades registradas en el proceso.
 - Fórmula PQL:SUM(


```

                            CASE
                            WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" IN ('Sí','No')
                            THEN 1
                            ELSE 0
                            END)
                    
```
 - Gráfica tendencia tasa de automatización: Gráfica que muestra cuál es la tasa de automatización a lo largo de cada uno de los meses de estudio, lo cual permite ver si en algún momento se ha realizado alguna automatización en algún punto del proceso.
 - Fórmula PQL:
 - Eje X: ROUND_MONTH("CEL_O2C_ACTIVITIES"."EVENTTIME")
 - Eje Y: SUM(


```

                                    CASE
                                    WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" = 'Sí'
                                    THEN 1
                                    ELSE 0
                                    END
                                    )/
                            SUM(
                                    CASE
                                    WHEN "CEL_O2C_ACTIVITIES"."USER_AUTOMATICO" IN ('Sí','No')
                                    THEN 1
                                    ELSE 0
                                    END)
                            
```
 - Tabla detalle por actividad: Tabla que muestra la tasa de automatización en detalle por cada una de las actividades que tienen lugar en el proceso. De

izquierda a derecha los campos de la tabla son: Actividad, # actividades manuales, # actividades automáticas, tasa de automatización.

- Tabla de detalle por X: Tabla que muestra información agrupada por una cierta dimensión que es posible seleccionar por medio de un botón desplegable y en base a dicha dimensión agrupada se realizan los cálculos. Las posibles dimensiones seleccionables son: producto, categoría del producto, condición de pago, cliente y país. Las columnas de izquierda a derecha son: la dimensión X por la cual se agrupan los datos, el número total de actividades, el número de actividades manuales, el número de actividades automáticas, la tasa de automatización.

6.6.8. Benchmark:

- **Definición:** Dashboard que permite hacer una comparativa del proceso así como de los KPIs generales ya analizados a través de distintas dimensiones de los pedidos. Este dashboard permite identificar distintos patrones de comportamiento que puede afectar a una cierta característica de los pedidos, así como pasos distintos en el proceso. En el lado izquierdo del dashboard se muestran los KPIs y el gráfico que muestra el proceso para una cierta condición de filtro sobre una dimensión específica, y en el lado derecho se muestran para otra condición de filtrado en paralelo, lo que permite visualizar de forma sencilla las diferencias. Las posibles dimensiones de filtrado son: Producto del pedido, Categoría del producto, Condición de pago, Cliente y País.

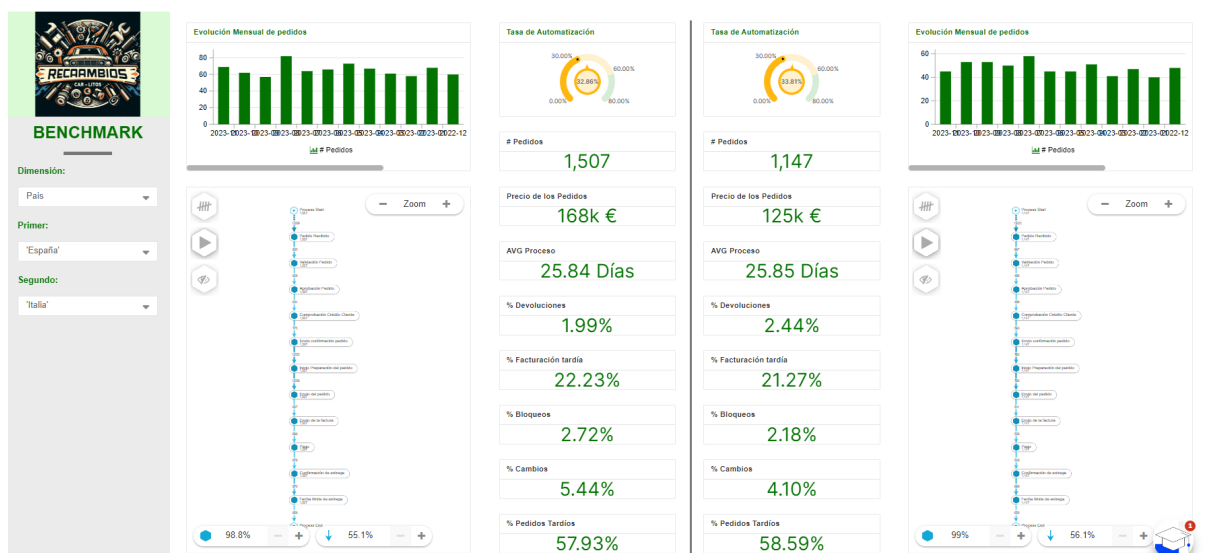


Figura 6.35: Dashboard Benchmark

- **KPIs y Gráficos:** Los KPIs y gráficos que se incluyen en este dashboard se corresponden con los principales analizados en cada uno de los dashboards anteriores, por lo que se hará una enumeración de ellos. De arriba a abajo en el dashboards se incluyen la gráfica de evolución mensual de pedidos, tasa de automatización, # Pedidos, Importe de los pedidos, AVG días del proceso, % devoluciones, % facturación tardía, % bloqueos, % cambios, % pedidos tardíos.

6.6.9. Social:

- **Definición:** Dashboard preconfigurado de la herramienta de Celonis, en la que a partir del campo de la actividad que identifica al usuario que ha realizado dicha actividad permite hacer un análisis de cada uno de los usuarios del sistema. En este dashboard, es posible hacer un análisis de los usuarios que mayor número de actividades realiza, así como cuál es su evolución mensual, conocer el número de actividades que realiza cada usuario por día, así como el número de usuarios que está involucrado por cada pedido. También, es posible hacer un estudio filtrado por usuario, el cual indica las horas en las que el usuario realiza un mayor número de actividades.

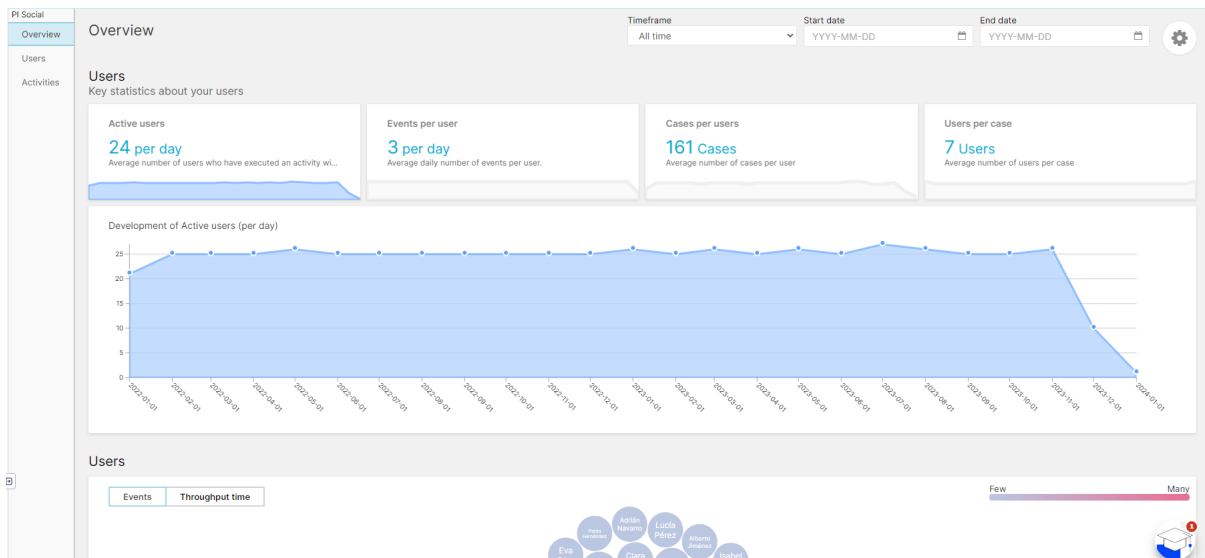


Figura 6.36: Dashboard Social

- **KPIs y Gráficos:** Los KPIs y gráficos son preconfigurados, se hace un listado de cada uno de ellos: Número de usuarios activos en el sistema, número de eventos medios por

usuario y día, número de pedidos afectados por usuario de media, número medio de usuarios involucrados en cada pedido, evolución mensual de usuarios activos en el sistema. Gráfica con usuarios con mayor número de actividades.

6.6.10. Conformance:

- **Definición:** Dashboard preconfigurado de la herramienta de Celonis, que permite analizar cómo es la realidad del proceso que se estudia en comparativa con el modelo de proceso ideal que se modela por medio de BPMN. Para ello, es necesario configurar el dashboard con el modelo BPMN que modela el proceso teórico que se debe de llevar a cabo. De esta forma, se compara caso por caso, es decir pedido a pedido si se sigue el modelo teórico y en caso de que no sea exactamente el flujo se identifica el punto en el cual se desvía del modelo. Por medio de la metadata del pedido permite analizar causas raíz de dichas desviaciones para una cierta dimensión del pedido.

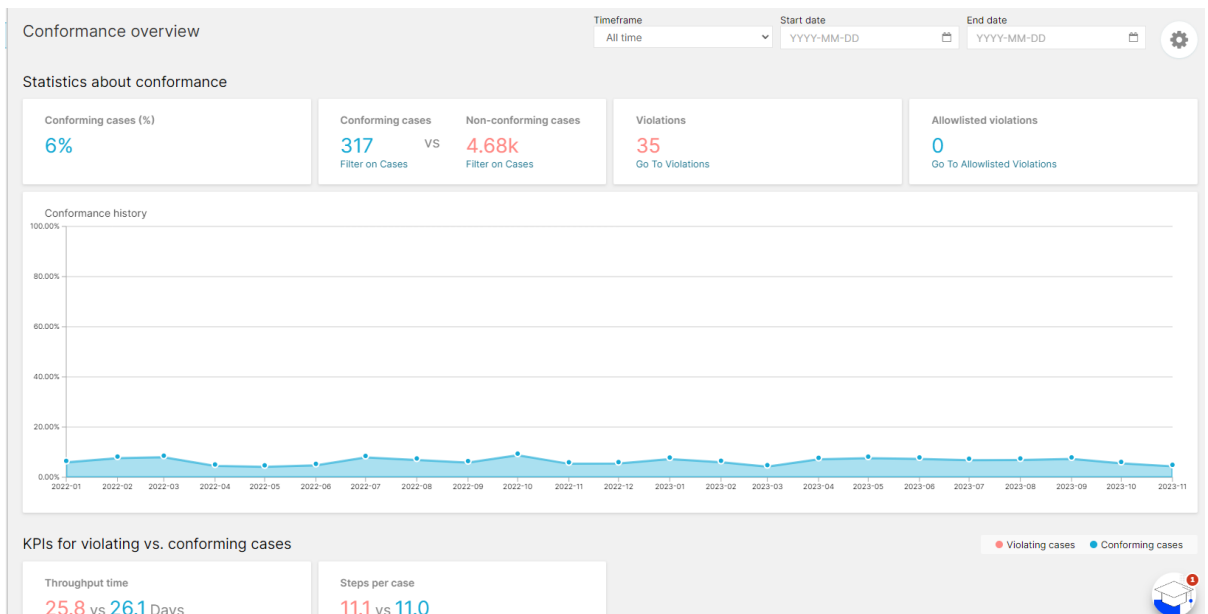


Figura 6.37: Dashboard Conformance

- **KPIs y Gráficos:** Los KPIs y gráficos son preconfigurados, se hace un listado de cada uno de ellos: % de pedidos conformes al modelo teórico, número de pedidos conformes al modelo, número de pedidos disconformes, evolución mensual de conformidades de los pedidos con el proceso teórico, tiempo medio del proceso cuando los pedidos son conformes al proceso vs cuando no lo son. Número medio de actividades en los pedidos que son conformes al proceso vs cuando no lo son.

6.6.11.Explorador de variantes:

- **Definición:** Dashboard preconfigurado de la herramienta Celonis que permite visualizar de forma gráfica cuales son las distintas variantes del proceso que existen. Se muestran ordenadas de la variante más frecuente a la menos frecuente los flujos completos para los casos de estudio.

La forma de visualizar el proceso, es por medio de un grafo en el que cada uno de los nodos se corresponde con una actividad del proceso. Las aristas se corresponden con los distintos “caminos” por los que se puede ir a la hora de pasar de una actividad a otra. El proceso comienza con un nodo inicial y con un nodo final en los cuales empiezan y acaban respectivamente cada una de las variantes. Cada una de las aristas que unen dos nodos se encuentran ponderadas con información relativa al número de casos que tiene dicha variante, así como los tiempos medios entre las actividades.

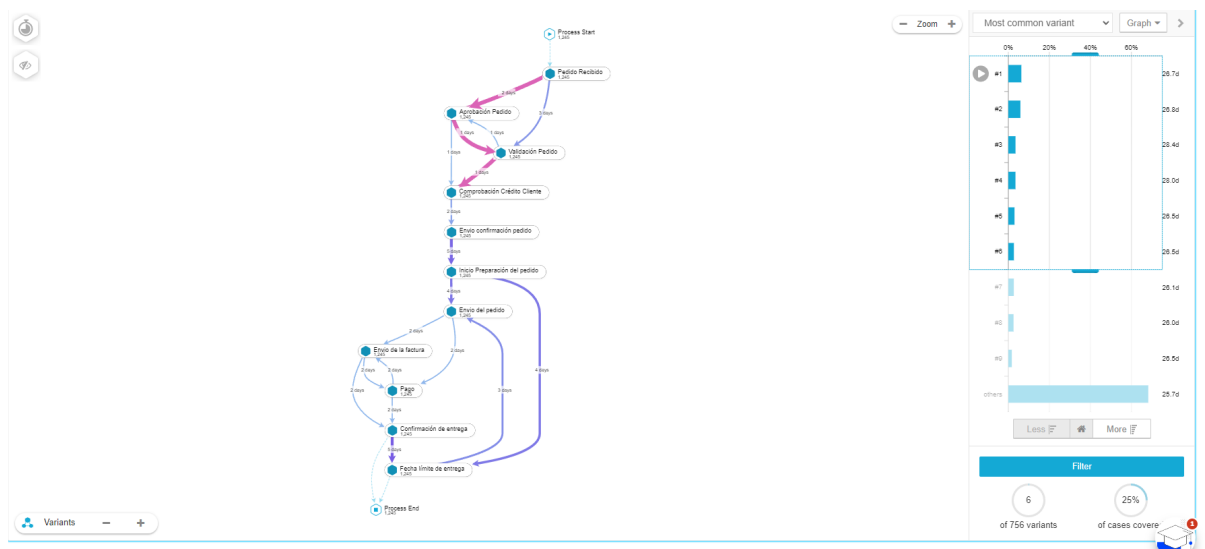


Figura 6.38: Dashboard Explorador de variantes

- **KPIs y Gráficos:** El principal gráfico es el ya comentado, en el cual podemos analizar cuales son las variantes más repetidas del proceso y los puntos que no son comunes a ellos. También, se permiten seleccionar distintas variantes para disponer del % de casos que cubren dichas variantes, permite conocer el tiempo medio entre actividades, así como el número de casos que pasan por dicha actividad o incluso la frecuencia de la actividad en caso de que pueda ser repetida varias veces. Por otro lado, es posible conocer los tiempos del proceso completo para cada una de las variantes.

6.6.12.Explorador de Procesos:

- **Definición:** Dashboard preconfigurado de la herramienta Celonis, que permite visualizar de forma gráfica cuál es el proceso de estudio. En este caso, el gráfico y la interpretación es idéntica al dashboard Explorador de variantes, pero dispone de un selector con el cual es posible añadir o quitar de forma gradual actividades y conexiones entre ellas al gráfico que permite visualizar el proceso.

Este dashboard, permite explorar rápidamente cuál es el proceso que se sigue así como detectar cuellos de botella o actividades no deseadas o en un orden no lógico dentro del proceso.

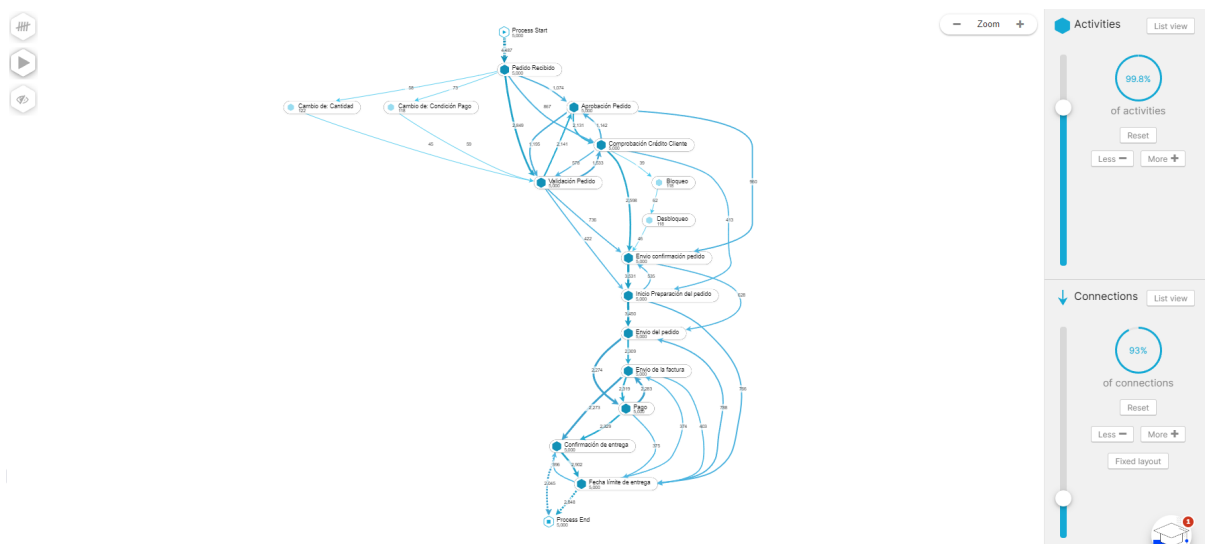


Figura 6.39: Dashboard Explorador de procesos

6.6.13.Explorador de Casos:

- **Definición:** Dashboard preconfigurado de la herramienta Celonis, que permite disponer una visión a alto nivel de cada uno de los casos de estudio, en el ejemplo de cada uno de los pedidos. La información se muestra en forma tabular donde cada uno de los campos se corresponden con los campos del caso de estudio.

Una vez que seleccionamos un caso concreto, se muestra el detalle de las distintas actividades que han tenido lugar para dicho caso seleccionado. Se muestra la fecha y hora de cada actividad así como información contextual relativa a la actividad (como el

usuario que la ha realizado, si es automática o no), permitiendo calcular de forma dinámica la diferencia de días existente entre cada una de las actividades previas o posteriores a la seleccionada.

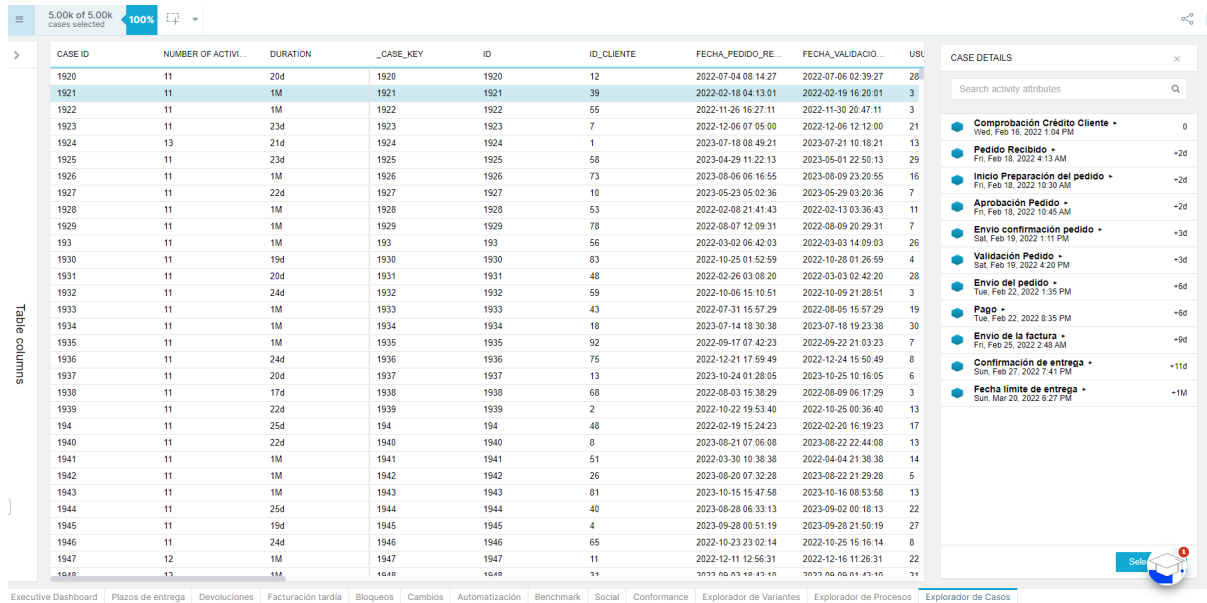


Figura 6.40: Dashboard Explorador de casos

6.7. Análisis de datos - Insights detectados

Una vez que disponemos de los distintos dashboards que incluyen todos los datos de origen de los distintos sistemas de información y las transformaciones oportunas que nos permiten hacer un procesado de todos ellos, estamos en condiciones de poder realizar un análisis de ellos de una forma sencilla. El hecho de disponer de distintas gráficas, KPIs globales, tablas y demás componentes en los dashboards, permiten que el analista de datos pueda realizar dicho análisis de una forma sencilla y apoyándose en información agrupada en distintos niveles de detalle. El proceso de análisis permite identificar distintas casuísticas, anomalías en el proceso o ineficiencias e indagar en las causas raíz de estas, así como medir el impacto que tienen. Una vez identificadas dichas casuísticas, es posible plantear distintos planes de acción los cuales permitan mitigarlas o reducirlas en la medida de lo posible para lograr una estandarización del proceso.

Para el ejemplo del caso de uso de estudio se incluye un análisis de los datos con una serie de insights detectados basándonos en los dashboards explicados anteriormente. Para ello, se mostrarán los resultados de los análisis centrándonos en los objetivos marcados. A parte de

mostrar los resultados obtenidos, se plantean posibles planes de acción para cada uno de ellos que permitan solventar las posibles problemáticas detectadas.

Antes de centrarnos en cada uno de los objetivos que son objeto de estudio, ponemos en situación en que consisten los casos de estudio. El proceso a analizar se corresponde con el proceso de Order to Cash, es decir todo el proceso que comprende desde que se crea un pedido hasta que se recibe el pago por dicho pedido. En este caso, la empresa sobre la cual vamos a realizar el estudio es una empresa de venta de piezas de automóviles, la cual recibe pedidos de distintas piezas por parte de sus clientes y se encargan de preparar el pedido, enviárselo al cliente, y que este lo reciba en la dirección de destino. Entre medias de todo este proceso existen distintas actividades como validaciones del pedido, bloqueos o desbloques de este por distintos motivos, envío de la factura y su correspondiente pago y/o incluso devoluciones o cambios sobre el pedido.

Cada uno de los casos de estudio se corresponde con un pedido que ha recibido la empresa "Recambios Car-litos" por parte de alguno de sus clientes. Los pedidos se corresponden con 5000 pedidos que se han realizado entre el 1 de enero de 2022 y el 31 de diciembre de 2023, es decir los pedidos correspondientes a dos años naturales.

A continuación se desarrollarán los resultados obtenidos del análisis para cada uno de los objetivos de estudio.

6.7.1. Plazos de entrega

De los 5000 pedidos de estudio 2908 son pedidos tardíos, es decir el 58.16% de los pedidos son entregados más tarde de la fecha límite de entrega acordada con el cliente. Dicho volumen de pedidos tardíos puede desembocar en una insatisfacción del cliente y su consecuencia pérdida de ventas por elegir a otro competidor en el mercado. De media, el tiempo de retraso es de 1.5 días, es decir, la entrega de los pedidos se hace de media 1.5 días más tarde de la fecha acordada con el cliente.

La distribución de los pedidos agrupa que la mayoría de ellos el retraso se encuentra entre 1 y 4 días, esto es en un total de 766 pedidos. Sin embargo, también existen pedidos en los cuales son entregados antes de la fecha límite de entrega, llegando a ocasiones en los que la entrega puede ser hasta 10 días antes de la fecha límite de entrega. Por otro lado, existen 124 pedidos en los que el retraso en la entrega se encuentra entre los 16 y 26 días.

Haciendo el estudio por los países de destino de los pedidos, se identifica que los países de destino que mayor retraso provoca en la entrega son Francia y Portugal con un retraso de media de 1.75 y 1.74 días respectivamente.

Además, existen una serie de clientes en los que es más frecuente que existan retrasos así como la media de días de retraso es mayor. Esto puede ser debido a la dirección de destino de dichos clientes y las comunicaciones que existen. Se han identificado que los pedidos relativos a los clientes EuroFleet Services, CarTech Innovations, EuroDrive Systems y EuroAuto Parts tienen un retraso de media en sus pedidos de 3.27,3.21,3.20 y 2.97 días respectivamente.

A partir de las evidencias mostradas, se plantean distintos planes de acción que permitiría reducir este retraso con respecto a los envíos. Para ello, de acuerdo al análisis realizado para el histórico de pedidos, se pueden acordar unas fechas de entrega límite en función del país de destino del cliente, así como del cliente en sí. De esta forma, las fecha límite de entrega no serán comunes a todos los pedidos sino que se adaptarán a cada cliente y esto producir que no se creen falsas expectativas al cliente y que no se puedan llegar a cumplir provocando su insatisfacción. Por otro lado, es posible hacer una planificación a la hora de gestionar y preparar los pedidos en base a cada cliente y/o su país de destino para agilizar el proceso de preparación de dicho pedido y poder ser enviado cuanto antes debido a que el cuello de botella del proceso se encuentra en el transporte de dicho pedido.

6.7.2. Devoluciones

Existen 112 pedidos que han sido devueltos por el cliente, esto supone que el 2.24% de los pedidos sufren devoluciones. El hecho de devolver un producto conlleva costes adicionales además de la pérdida de una venta y su reembolso al cliente, así como la insatisfacción del cliente. Por este motivo, es necesario hacer un análisis detallado de las posibles causas de devoluciones para intentar mitigarlas y reducir el número de devoluciones en el proceso.

Entre los 112 pedidos devueltos supone una pérdida de ingresos de 11600€ y que se deben de reembolsar a los clientes. Además, las devoluciones tienen lugar de media 18 días después de que le llega el producto al cliente, lo que provoca que durante ese tiempo dicho producto tampoco se encuentre disponible en stock para poder vendérselo a otros clientes.

Los principales motivos de devolución de los pedidos son desperfectos en el envío, el cual afecta a 36 pedidos y una descripción del producto errónea, afectando a 34 pedidos. Lo cual, puede derivar como plan de acción para reducir dichas devoluciones el cambiar de empresa de transporte para tratar de reducir los desperfectos causados en el envío o actualizar las descripciones de los productos del catálogo por una descripción más precisa para evitar confusiones.

El análisis profundo permite identificar que la mayor tasa de devoluciones las tienen aquellos pedidos relativos a la categoría de “Carrocería” con una tasa de devolución del 4% por encima de la media. Además, para dicha categoría de producto el principal motivo de devolución es la

descripción errónea del producto, lo que indica que como plan de acción es necesario ajustar las descripciones del catálogo para los productos de la categoría de “Carrocería”. Al igual que hemos hecho la categoría de “Carrocería” es posible hacerlo con otras categorías de productos o incluso con los productos concretos identificando cuáles son los productos que mayor porcentaje de devolución tienen así como cuáles son los principales motivos de devolución de estos para aplicar las medidas correctoras oportunas en función del producto.

Por otro lado, a partir del análisis se puede concluir que el país de destino de los pedidos con mayor ratio de devoluciones es Portugal con un total de 15 pedidos devueltos y un ratio de 2.76% de los pedidos, además el principal motivo de devolución es por desperfectos en el envío, afectando a 7 pedidos. Por tanto una posible acción es cambiar el transportista para los pedidos con destino Portugal.

6.7.3. Facturación tardía

El caso de estudio se centra en determinar todos aquellos pedidos en los que se ha llevado a cabo una facturación tardía, es decir, han pasado más días de los permitidos en las políticas de la empresa desde que se envía el pedido al cliente hasta que se le envía la factura para su posterior pago. Como ejemplo, vamos a suponer que para la empresa “Recambios Car-litos” las políticas acordadas de forma interna dicen que se debe de enviar la factura del pedido como máximo en 4 días naturales desde que se envía el pedido al cliente, es decir, todo pedido en el que la factura se haya enviado al cliente 5 o más días después del envío del pedido, se considera que ha sido facturado tarde.

El hecho de facturar tarde un envío supone que recibas más tarde el dinero por parte del cliente, puesto que la fecha base sobre la cual se aplican los días de pago acordados con el cliente en los términos de pago de la factura se corresponde con la fecha en la que se envía la factura al cliente. Esto supone que el recibir más tarde el pago de los pedidos disminuye el flujo de caja de la empresa y en consecuencia se disponga de menor capital circulante para afrontar distintas oportunidades o se tenga que recurrir a créditos bancarios con las comisiones que ello conlleva y las pérdidas económicas que supone para la empresa.

A partir del dashboard desarrollado, podemos analizar que de los 5000 pedidos que son objeto de estudio en 1035 se ha llevado a cabo una facturación tardía, es decir el 20.7% de los pedidos han sido facturados tarde. El importe afectado por facturación tardía es de 115k € lo cual supone flujo de caja. Además, entre todos aquellos pedidos facturados tarde la media de días que transcurren entre el envío y la facturación es de 5.3 días, puesto que lo permitido es como máximo 4 días, se factura en media tarde 1.3 días.

Un análisis profundo, nos permite concluir que en algunos meses se han tenido picos en cuanto al ratio de facturación tardía, como pueden ser los meses de Octubre de 2022 o Septiembre de 2023. Con un ratio de facturación tardía del 27.94% y 22.63% respectivamente. Esto puede ser por un aumento en los pedidos durante estos meses después del verano y que se produzca una congestión en la facturación por no disponer de los recursos suficientes. A partir de este conocimiento del histórico, se puede ser capaz de anticiparse a lo que pueda ocurrir los siguientes años y tomar acciones para reducir la facturación tardía, como puede ser el contratar en los meses de después de verano a otro empleado más que se encargue de la facturación del pedido y liberar el cuello de botella que se produce en esa actividad durante dichos meses.

Existen pedidos en los que han llegado a transcurrir en 8 días desde que se envía el pedido y se lleva a cabo la facturación de este, como los pedidos con ID 1847,3989 y 2595.

Además, existen clientes en los cuales existe un elevado ratio de facturación tardía, como puede ser CarTech Solutions, AutoMasters Connect Europe y CarTech Innovations GmbH con unos ratios de facturación tardía de 37.5%, 31.78% y 30.91% respectivamente y bastante superiores a la media.

6.7.4. Bloqueos

En este caso vamos a analizar los bloqueos que tienen lugar en el proceso O2C, en este caso para la empresa “Recambios Car-litos” existen dos tipos de bloqueo, bloqueos de crédito y bloqueos de stock. El hecho de que se produzca un bloqueo supone que hasta que se lleve a cabo el desbloqueo de este el pedido queda parado y supone que se aumente el tiempo global del proceso y puede ocasionar retrasos en la entrega del pedido y la insatisfacción de los clientes.

El porcentaje de bloqueos en los pedidos que son objeto de estudio es del 2.36%, es decir, existen 118 pedidos de los 5000 en los que han tenido lugar al menos uno de los dos tipos de bloqueo. Además, de media el tiempo que han estado bloqueados es de 1.4 días que engrosan el tiempo global del proceso. Existen pedidos en los que el bloqueo ha llegado a suponer 3 días, como son los pedidos con ID 4692,1521 y 4878. En todos ellos el bloqueo ha sido de crédito.

Existen meses en los que existen mayores bloqueos de crédito o de stock, como por ejemplo, el mes de enero de 2022 se llegó a tener 5 bloqueos de stock. Lo cual, permite identificar aquellos meses en los que existe una mayor demanda de pedidos y adelantarse para evitar quedarse sin stock y que se produzcan bloqueos sobre los pedidos.

Por otro lado, se puede analizar que los productos que mayores bloqueos por falta de stock son los depósitos de refrigerante, el kit de embrague y batería de alto rendimiento con 5.4 y 3

bloqueos de stock respectivamente. Esto permite definir un plan de acción que haga que la revisión de stock se lleve a cabo más frecuentemente para este tipo de productos. Además, es posible analizar si para un cierto tipo de producto existen meses con picos en cuanto a bloqueos de stock para anticiparse a ellos.

Otro de las conclusiones que se llega por medio del análisis es que clientes como EuroWheels Ltd. y CarTech Solutions Europe son clientes con un alto número de bloqueos de crédito con 5 y 4 respectivamente. Esto desencadena en tomar las acciones necesarias para reducir dichos bloqueos con estos clientes.

6.7.5. Cambios

El caso que se analiza ahora consiste en los cambios que tienen que ver en los pedidos, existen dos tipos de cambios que pueden afectar a los pedidos durante su preparación. El primero de ellos son cambios de cantidad en el pedido y el segundo de ellos cambios de la condición de pago de estos. El hecho de que se produzcan cambios en los pedidos supone retrasos a la hora de entregar los pedidos, así como incurrir en retrabajos.

El porcentaje de cambios en pedidos es del 4.8%, es decir, existen 240 pedidos de 5000 que tienen cambios. Existen 122 cambios de cantidad y 118 cambios de condición de pago, por tanto, es ligeramente más frecuente que se hagan cambios de cantidad.

Entre los clientes que mayor ratio de cambios en los pedidos se encuentran EuroMotor Parts, CarTech Connect y EuroCar Accessories Solutions AG con un ratio de cambios del 10.26%, 9.3% y 9.18% bastante superior a la media. Lo cual hace que como plan de acción se tengan que acordar ciertas políticas con dichos clientes a la hora de que se les permita hacer cambios sobre los pedidos.

Por ejemplo, para el caso del cliente CarTech Innovations Europe Ltd. son frecuentes los cambios en la condición de pago. En 9 de los 14 cambios que ha realizado son cambios en la condición de pago. Esto plantea planes de acción que permitan actualizar los maestros con las condiciones de pago de dicho cliente debido a que en los últimos acuerdos comerciales se hayan fijado otras condiciones distintas. De esta forma no se producirían tantos cambios en las condiciones de pago y supondría un ahorro de tiempo y retrabajos.

Además, la condición de pago que más veces sufre cambios es la de contrareembolso con 36 cambios, seguida de pago a plazos con 26 cambios.

6.7.6. Automatización

En este caso vamos a analizar el grado de automatización que tiene el proceso completo. El hecho de analizar un proceso conlleva la reducción de tiempos en el proceso, así como la reducción de costes en cuanto a realizar actividades repetidas dentro del proceso y que permiten una automatización.

La tasa de automatización global del proceso es del 33.33% es decir, 1 de cada 3 actividades dentro del proceso se lleva a cabo de forma automática. De 35476 actividades 11825 se realizan de forma automática.

A partir del dashboard, se puede analizar aquellas actividades que mayor o menor ratio de automatización tienen en el proceso. Otra de las gráficas, permite identificar aquellos meses a partir de los cuales se empieza a aumentar la tasa de automatización del proceso, así como si existen meses con picos de automatización. Esto puede ser útil para visualizar cómo tienen efecto aquellas automatizaciones que se pueden implementar en el proceso.

Aquellas actividades que tienen una mayor tasa de automatización son los cambios de condición de pago y los cambios de cantidad con una tasa del 36.44% y del 36.07%. Lo cual es debido a que los cambios se pueden realizar directamente desde la página web de la empresa y se llevan a cabo de forma automática en los sistemas de información de la empresa. Sin embargo, las actividades de bloqueo y desbloqueo son actividades que ambas tienen una tasa de automatización del 28.81% debido a su necesidad en la mayoría de las ocasiones de que se lleven a cabo por humanos.

Las tasas de automatización también varían de acuerdo al tipo de producto, de esta forma los pedidos del producto de "Sistema de escape completo" tienen la mayor tasa de automatización con un 35.33%. Sin embargo, los pedidos del producto "Kit de freno de alto rendimiento" tienen la menor tasa de automatización del 28.7%.

Los planes de acción que se pueden plantear es el hecho de aumentar la automatización de aquellas actividades que tienen una baja tasa de automatización o incluso automatizar en mayor medida los pedidos que afectan a ciertos tipos de productos.

6.7.7. Benchmark

Por medio del benchmark podemos hacer comparaciones directas sobre los pedidos a partir de alguna característica de estos. Esto permite visualizar cómo puede haber cambios en el proceso dependiendo de alguna característica concreta.

Por ejemplo, comparando los pedidos en los que la categoría del producto es “Sistema de encendido” con aquellos en los que la categoría es “Carrocería”, podemos ver como el porcentaje de devoluciones en los productos de categoría “Carrocería” es superior a la de “Sistema de encendido” siendo del 4% y 1.77% respectivamente. Algo similar ocurre con el % de bloqueos siendo del 3% y 1.59%. El % de cambios también es superior en el caso de la carrocería, siendo del 7% frente al 4.95%. Para el caso del ratio de pedidos tardíos en los de “Carrocería” es del 58.5% y en los de “Sistema de encendido” es del 54.77%.

Este análisis, permite identificar que aquellos pedidos de categoría “Carrocería” suelen tener mayor problemáticas, al ser devueltos en mayor ocasiones, producen bastantes bloqueos, cambios y además se entregan tarde a los clientes. Esto puede desencadenar planes de acción específicos sobre cada tipo de producto.

7. Conclusiones

En este capítulo se discutirá si los objetivos marcados al inicio del proyecto han sido conseguidos y en qué grado, así como comentar posibles impedimentos encontrados en el proceso y proponer líneas de trabajo futuras.

7.1. Consecución de los objetivos

En general, los objetivos marcados antes de abordar el proyecto han sido alcanzados. Antes de empezar la elaboración del proyecto tenía un ligero conocimiento de en qué consistía la técnica de minería de procesos, debido a que había realizado las prácticas curriculares del grado en una empresa en la cual se dedicaba a ello. Sobre todo el poco conocimiento que tenía era un conocimiento meramente técnico sobre cómo utilizar una herramienta de visualización de datos. Sin embargo, el realizar este proyecto me ha permitido descubrir desde el punto de vista teórico en qué consiste la minería de datos, las distintas aplicaciones que tiene a cualquier proceso empresarial así como los distintos enfoques que se le puede dar.

Uno de los aspectos que más importante ha sido para mí, es el ver como una ingesta cantidad de datos sin ningún tipo de estructura y que por sí solos carecen de información por medio de técnicas matemáticas, estadísticas, de análisis y visualización permiten obtener conclusiones acerca de ellos así como obtener patrones de comportamiento que permiten eficientar los procesos y la traducción económica que esto conlleva. También, me ha servido para entender la necesidad que existe en la actualidad de disponer de grandes cantidades de datos y obtener un conocimiento de ellos.

De acuerdo a los objetivos fijados al principio del proyecto, ha sido cubierto el entendimiento de en qué consiste la gestión de procesos y en concreto la minería de procesos. También, se han desarrollado los distintos usos que se le puede dar a la minería de procesos, así como identificar la evolución que ha tenido la minería de procesos y las distintas soluciones software existentes.

Por otro lado, se ha mostrado un ejemplo de principio a fin de en qué consiste un proyecto de minería de procesos en un caso de uso, incluyendo las técnicas utilizadas así como las técnicas empleadas para analizar datos y detectar patrones de comportamiento y cuellos de botella.

7.2. Líneas de trabajo futuras

El proyecto ha terminado con el ejemplo de caso de uso en el cual se muestran las técnicas de minería de procesos. En la actualidad, la minería de procesos se encuentra en auge y esto conlleva a que se están llevando a cabo numerosas investigaciones sobre técnicas que mejoren la minería de procesos así como la aplicación de algoritmos de machine learning para poder realizar predicciones en base a los datos recopilados. Por tanto, una línea de trabajo futuro podría ser el incluir algoritmos de predicción a los análisis para poder anticiparse a aquellos problemas y tomar una posición proactiva.

Otra línea de trabajo futuro y que no ha sido abordado en el proyecto debido a la limitación en cuanto a carga de horas dedicables, es el hecho de adquirir dicha posición de proactividad implementando automatizaciones las cuales sean capaces de notificar en algún momento a las personas responsables de que hay que tomar alguna acción en cuanto se detecta algún indicador que desencadena problemas en el proceso. La herramienta Celonis, dispone de un módulo de automatizaciones que permite implementar distintos planes de acción como envíos de correos automáticos, notificaciones en Slack, Teams o incluso envío de Excels con la información tabulada y filtrada.

Bibliografía

- [1] *Celonis Academy* (2024). URL: <https://academy.celonis.com/>
- [2] *Celonis Product Documentation* (2024). URL: <https://docs.celonis.com/?lang=en>
- [3] *Celonis Registration* (2024). URL: <https://signup.celonis.com/ui/sign-up/get-started>
- [4] *Guía de modelado, reglas y símbolos de BPMN 2.0* (2024). URL: <https://www.gbtec.com/es/recursos/bpmn/>
- [5] *Jira Documentation* (2024). URL: <https://confluence.atlassian.com/jira>
- [6] *Process Mining Software Comparison* (2024). URL: <https://www.processmining-software.com/>
- [7] *Prom Tools* (2024). URL: <https://promtools.org/>
- [8] *UI Path Process Mining* (2024). URL: <https://www.uipath.com/product/process-mining>
- [9] WIL VAN DER AALST (2016), *Process Mining. Data Science in Action. Second Edition*. Springer
- [10] WIL VAN DER AALST (2011), *Process Mining. Discovery, Conformance and Enhancement of Business Processes. First Edition*. Springer