



***Facultad
de
Ciencias***

Desarrollo de herramienta software para la generación y procesamiento automático de los intercambios de información entre comercializadora y distribuidora eléctrica.

Software tool development for the automatic generation and processing of the information exchanges between electricity retailers and distributors.

**Trabajo de Fin de Grado para acceder al
GRADO EN INGENIERÍA INFORMÁTICA**

Autor: Álvaro Alcántara Ruiz de Garibay
Director: Iván Pedraja Lombilla
Co-Director: Diego García Saiz
Julio - 2023

Resumen corto:

Todas las modificaciones que un cliente realiza en su contrato de electricidad como por ejemplo cambios de compañía, altas, bajas, etc. deben de ser informadas por su compañía comercializadora a la distribuidora correspondiente mediante archivos XML que siguen un esquema concreto regido por la CNMC (Comisión Nacional de los Mercados y la Competencia), con el objetivo de que la distribuidora pueda llevar a cabo los cambios necesarios en la red e informar de posibles incidencias. Este proceso de intercambio de información es gestionado por los operadores de la comercializadora de forma manual, tomando un tiempo considerable de su jornada e induciendo a posibles errores en su transcripción, por lo que se antoja necesario el desarrollo de una herramienta que automatice este proceso, generando de manera automática los archivos correspondientes a las solicitudes de intercambio por parte de los clientes y procesando las respuestas que dan las distribuidoras a estas solicitudes, permitiendo así a los operadores centrarse en otras tareas menos tediosas y a la compañía incrementar su velocidad de respuesta ante estas solicitudes

Palabras clave: XML, automática, herramienta, desarrollo.

Abstract:

All the modifications that a customer makes to their electricity contract, such as company changes, new registrations or cancellations, etc., must be notified by their energy supplier to the corresponding distribution company through XML files that must follow a specific schema regulated by the CNMC (spanish acronym for National Commission of Markets and Competition). The aim is to enable the distribution company to implement the necessary changes in the electricity network and report any potential incidents. Currently, this information exchange process is manually managed by operators at the energy supplier, consuming a significant portion of their workday schedule and can potentially lead to transcription errors. Thus, it becomes necessary to develop a tool that automates this process, generating the corresponding files for customer exchange requests automatically and processing the responses provided by the distribution companies. This would allow operators to focus on other less tedious tasks, while enabling the company to enhance its responsiveness to these requests.

Keywords: XML, automatically, tool, develop.

Índice

1. Introducción.....	5
2. Entendiendo cómo funcionan los intercambios de información.....	6
2.1 Procesos con tres agentes implicados (Archivos C1 y C2):.....	8
2.2 Procesos con dos agentes implicados (A3, B1, D1, E1, F1, M1, R1, W1, Q1).....	9
3. ¿Cómo gestiona la comercializadora los intercambios? - Estado del arte.....	11
4. Tecnologías utilizadas.....	14
5. Cronograma y metodología.....	16
5.1 Cronograma.....	16
5.2 Metodología.....	17
6. Captura de requisitos.....	18
6.1 Fuentes de requisitos.....	18
6.2 Actividades de captura de requisitos.....	19
6.2 Objetivos.....	22
7. Casos de uso.....	22
8. Requisitos no funcionales.....	26
9. Arquitectura y diseño de la aplicación.....	27
9.1 Diseño de componentes.....	27
9.2 Diagrama de clases e implementación.....	28
9.3 Secuencia de generación de los archivos XML.....	30
9.4 Secuencia de lectura de los archivos XML.....	31
9.5 Secuencia de creación automática de nuevos registros SWD.....	32
9.6 Automatización del proceso.....	33
10. Pruebas software.....	34
11. Conclusiones y desarrollo futuro.....	35
11.1 Conclusiones.....	35
11.2 Desarrollo futuro.....	36
12. Referencias.....	36

1. Introducción

Durante los últimos años el mercado eléctrico ha estado en boca de todos, es raro abrir cualquier periódico digital y tras pasar unos segundos navegando, no encontrarse noticias sobre subidas de precios, cambios de regulación, energías renovables y una larga retahíla de artículos, estudios y estadísticas sobre cuál es la mejor hora para poner la lavadora o usar nuestro calentador eléctrico.

Ante esta coyuntura, muchos consumidores han optado por modificar su contrato, bajando la potencia con el objetivo de reducir su factura o buscando cambiarse al mercado libre, otros quizá hayan contemplado cambiarse de compañía eléctrica en busca de las mejores condiciones posibles o incluso directamente han dado de baja el suministro en residencias vacías.

Todos estos cambios están regulados por la CNMC (Comisión Nacional de los Mercados y la Competencia), que, para hacerlos efectivos, obliga a las comercializadoras a informar de estos a las empresas distribuidoras, junto con todos los datos técnicos necesarios para que la distribuidora haga los ajustes pertinentes en la red, asimismo, las distribuidoras han de confirmar a la comercializadora cuando estos cambios hayan sido aceptados y posteriormente activados, además deberán informar de cualquier incidencia que surja en el proceso [1].

Este flujo de información se lleva a cabo mediante el envío archivos XML como el de la figura 1, que han de seguir un esquema concreto establecido por la CNMC en función del tipo de modificación que se solicite, además, cada proceso de intercambio sigue una serie de pasos correspondientes a diferentes fases como el envío de la solicitud, aceptación de esta, activación, notificación de incidencias, entre otros, a cada uno de estos pasos le corresponde un formato concreto de archivo XML.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MensajeCambioComercializadorSinCambios xmlns="http://localhost/elegibilidad">
  <Cabecera>
    <CodigoREEEmpresaEmisora>1497</CodigoREEEmpresaEmisora>
    <CodigoREEEmpresaDestino>0021</CodigoREEEmpresaDestino>
    <CodigoDelProceso>C1</CodigoDelProceso>
    <CodigoDePaso>01</CodigoDePaso>
    <CodigoDeSolicitud>000000000041</CodigoDeSolicitud>
    <SecuencialDeSolicitud>01</SecuencialDeSolicitud>
    <FechaSolicitud>2022-02-21T07:31:00</FechaSolicitud>
    <CUPS>ES0021000003291273VQ0F</CUPS>
  </Cabecera>
  <CambioComercializadorSinCambios>
    <DatosSolicitud>
      <IndActivacion>A</IndActivacion>
      <ContratacionIncondicionalPS>N</ContratacionIncondicionalPS>
      <ContratacionIncondicionalBS>N</ContratacionIncondicionalBS>
    </DatosSolicitud>
  </CambioComercializadorSinCambios>
</MensajeCambioComercializadorSinCambios>
```

Figura 1- Archivo XML de intercambio de información entre comercializadora y distribuidora eléctrica.

En Ekiluz, comercializadora eléctrica de reciente creación impulsada por Repsol y grupo Mondragón [2], este proceso se realiza de forma manual. Los operadores de *Backoffice* de la compañía cumplimentan directamente los datos que leen en el Salesforce de la empresa en unos archivos XML con los elementos vacíos que sirven como plantilla, y actualizan a mano cada solicitud en el CRM una vez reciben las respuestas de la distribuidora.

Estos servicios de gestión y de consultoría técnica están externalizados y son llevados a cabo por la consultora tecnológica NTT Data, dentro de su oficina de Santander, en la que actualmente me encuentro trabajando y para la cual se realizará el proyecto.

El proceso de gestión manual que se lleva a cabo actualmente es altamente costoso en tiempo y tiene una alta tasa de error, lo cual en periodos de alta actividad puede generar atrasos a la hora de atender las peticiones de los clientes, en vista de todo esto, se diseñará e implementará una herramienta software que automatice en la medida de lo posible el proceso de intercambio de información, tanto la generación de las nuevas solicitudes como el procesamiento de las respuestas.

Este sistema tendrá como objetivo mejorar la velocidad de respuesta de la compañía ante las solicitudes de los clientes, para ello, se deberá implementar los tipos de intercambio que atiende la compañía con mayor frecuencia, siendo capaces de generar archivos y procesar respuestas en bloque, haciendo un software escalable y mantenible, pudiéndose moldear fácilmente en caso de cambios de regulación de la CNMC o migraciones en el CRM del cliente.

2. Entendiendo cómo funcionan los intercambios de información

Con la ley actual, la CNMC contempla diez tipos de intercambio de información relativos al mercado eléctrico [3], cada uno con su esquema correspondiente para los archivos XML, los diferentes tipos de intercambio se detallan en la tabla 1:

Código del archivo	Proceso	Esquema xsd
A3	Alta de un punto de suministro	Alta.xsd
B1	Baja o Suspensión del suministro	BajaSuspension.xsd
C1	Cambio de comercializador sin modificaciones en el contrato de acceso	CambiodeComercializadorSinCambios.xsd
C2	Cambio de comercializador con modificaciones en el contrato de acceso	CambiodeComercializadorConCambios.xsd
D1	Notificación de cambios en el punto de suministro	NotificacionCambiosATRDesdeDistribuidor.xsd
F1	Facturación de los peajes de acceso y otros conceptos regulados	Facturacion.xsd
M1	Modificación en el contrato de acceso	ModificacionDeATR.xsd
Q1	Saldos y lecturas de consumidores con peaje directo	SaldoLecturasFacturacion.xsd

R1	Reclamaciones	ReclamacionPeticion.xsd
W1	Autolecturas	SolicitudAportacionLectura.xsd

Tabla 1- Resumen de las diferentes tipologías de intercambio de información

La CNMC proporciona además archivos Excel con información sobre los formatos que deben seguir los archivos XML que manejan el intercambio de información como los que podemos ver en la figura 2 [4].

Estos archivos informan del nombre del campo, que corresponde a un atributo XML, su obligatoriedad, formato y una columna "Tabla" que indica en qué tabla de códigos de la CNMC consultar los códigos identificativos en caso de que se trate de algún campo codificado. Por ejemplo, el código de identificación de la distribuidora eléctrica, que debe ser expresado con su número asignado por la CNMC.

TABLA PROCESO C1 - CambiodeComercializadorSinCambios.xsd						
Elemento XSD	Descripción	Condición XSD	Condición distribuidor	Obligatorio S/N	Formato	Tabla
Cabecera						
CodigoREEEmpresaEmisora	Código REE Empresa Emisora			S	X(4)	1
CodigoREEEmpresaDestino	Código REE Empresa Destino			S	X(4)	1
CodigoDelProceso	Código de Proceso			S	X(2)	2
CodigoDePaso	Código del Paso			S	X(2)	3
CodigoDeSolicitud	Este número será obligatorio, único e incremental por empresa solicitante o iniciadora de un proceso.	Sólo caracteres numéricos permitidos.		S	X(12)	
SecuencialDeSolicitud	Secuencial de solicitud	Sólo caracteres numéricos permitidos.		S	X(2)	
FechaSolicitud	Fecha de construcción del paso por el agente emisor			S	AAAA-MM-DD THH:MM:SS	
CUPS	Código Universal del Punto de Suministro			S	X(22)	
CambiodeComercializadorSinCambios						
DatosSolicitud						
IndActivacion	Indicativo del tipo de activación solicitada. Este campo, manda sobre la "FechaPrevistaAccion".			S	X(1)	8
FechaPrevistaAccion	Solo se informará cuando se solicite un cambio a fecha fija. Esta fecha fija no podrá ser superior a 30 días a contar desde la fecha de solicitud. Fecha (incluida) prevista del cambio		Se debe cumplir que: FechaDeSolicitud (del paso 01)+30días ≥FechaPrevistaAccion Si IndActivacion=F, este campo se debe informar, de lo contrario, se rechazará con "Falta campo obligatorio" Si IndActivacion=F y FechaPrevistaAccion<FechaSolicitud(Cabecera), se rechazará la solicitud con el motivo "Fecha de la operación solicitada con carácter retroactivo" Si IndActivacion=F y FechaSolicitud(Cabecera)<FechaPrevistaAccion<FechaCarga en el portal del distribuidor, se activará con la opción "cuanto antes".	N	AAAA-MM-DD	
ContratacionIncondicionalPS	Indicativo S/N Si "S", el distribuidor no rechaza el cambio de comercializador si el suministro está cortado por impago (o en curso) o en baja por impago en curso. Si "N", el distribuidor rechaza el cambio de comercializador para informar de que el suministro está cortado por impago (o en curso) o en baja por impago en curso.		Solo aplica para los proceso B1 motivos 03 y 04. No aplica para los procesos de corte iniciados por el distribuidor.	S	X(1)	26
ContratacionIncondicionalPS	Indicativo S/N Si "S", el distribuidor no rechaza el cambio de comercializador si el suministro es un bono			S	X(1)	26

Figura 2- Ejemplo del esquema proporcionado por la CNMC correspondiente a uno de los archivos del proceso C1

Además de los campos y condiciones de formato que vemos en la figura 2, cada uno de estos esquemas establece una serie de pasos, que equivalen a cada una de las fases del intercambio de información, se generará un archivo XML por cada uno. Dependiendo del paso, el archivo podría ser generado por comercializadora y procesado por distribuidora, o viceversa.

En este caso, podemos dividir cada uno de los tipos de intercambio en dos subgrupos según la cantidad de agentes implicados en el intercambio, en cada subgrupo, cada paso es semánticamente similar, aunque los formatos de archivo son distintos:

2.1 Procesos con tres agentes implicados (Archivos C1 y C2):

En este caso, al tratarse ambos procesos de un cambio de compañía, podríamos asimilarlo a una portabilidad de compañía telefónica.

Imaginemos que tenemos la electricidad contratada con Endesa e Iberdrola nos hace una oferta para cambiar de compañía, aceptamos la oferta e Iberdrola (comercializadora entrante) comienza el proceso de intercambio C1 (mantenemos la potencia contratada, tarifa y titular):

- En primer lugar Iberdrola enviará un archivo XML de paso 01, donde indicará la información del punto de suministro y la información del cliente correspondiente.
- La distribuidora correspondiente a ese punto de suministro procesa la solicitud y responderá con un archivo de paso 02, este puede ser de aceptación, donde indicará la información relativa a ese punto de suministro así como la fecha estimada de activación, o de rechazo, donde indicará la razón del rechazo (normalmente en este paso se debe a un defecto de forma)
- Si es aceptado, la distribuidora enviará un archivo de paso 11 a la comercializadora saliente (Endesa) informando de que se ha aceptado un cambio de comercializador para uno de sus puntos de suministro, si se rechaza, la comercializadora entrante podrá reenviar el paso 01

Llegados a este punto pueden ocurrir dos cosas:

- La distribuidora activa el cambio, así que envía un archivo de paso 05 a la comercializadora entrante y uno de paso 06 a la comercializadora saliente, notificando de la activación, que será efectiva el próximo día hábil desde la notificación.
- Debido a problemas técnicos de la red o en el punto de suministro el suministro eléctrico está suspendido:
 - Si consigue arreglar la incidencia se vuelve al punto anterior.
 - Si no fuera posible arreglar el problema, la distribuidora envía un archivo de paso 04 a la comercializadora entrante para notificar de que es imposible realizar el cambio, y notifica a la comercializadora saliente con un archivo de paso 12, avisando de que el suministro sigue estando a su cargo.

Antes de la activación definitiva del cambio, la comercializadora entrante podría querer anularlo, aunque esta casuística es extremadamente rara, la CNMC también la contempla, la distribuidora podrá comenzar el siguiente proceso siempre y cuando no haya pasado un día hábil desde la notificación mediante un paso 05:

- La comercializadora entrante envía un paso 08 a la distribuidora, indicando el por qué de la solicitud.
- Si ha llegado a la distribuidora en tiempo y forma, la distribuidora envía un archivo de paso 09 de aceptación y uno paso 10 a la comercializadora saliente para notificar de que el cambio se ha interrumpido.
- Si no ha llegado en tiempo informa la distribuidora envía a la comercializadora entrante un archivo de paso 09 de rechazo, esta se deberá hacer cargo del suministro.

La CNMC nos ofrece también un esquema del funcionamiento de este proceso [5], que podemos ver en la figura 3.

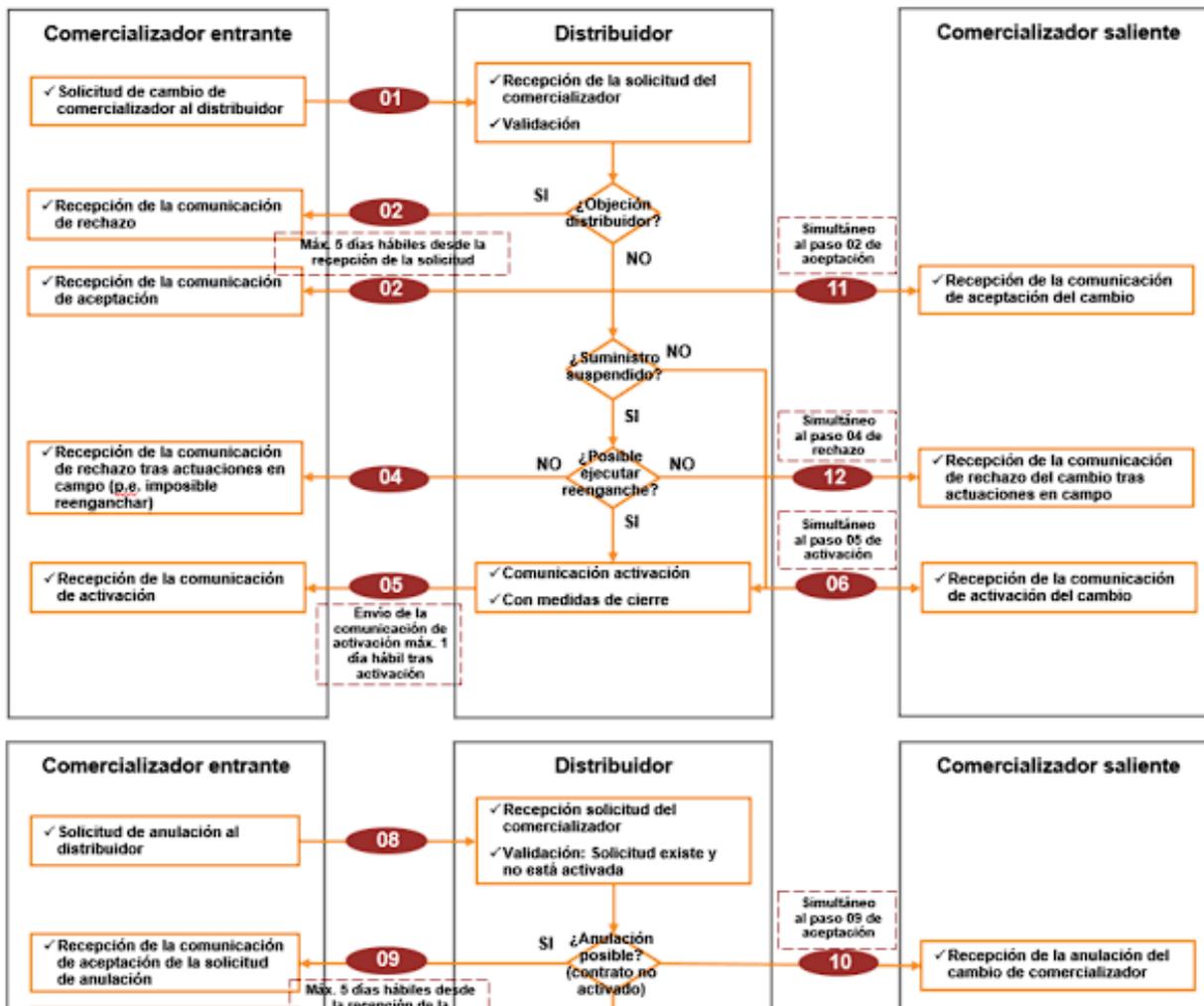


Figura 3- Flujograma que resume el proceso previamente descrito

2.2 Procesos con dos agentes implicados (A3, B1, D1, E1, F1, M1, R1, W1, Q1)

Debido a la semántica de estos procesos, el intercambio de información se produce de forma bilateral entre la comercializadora y la distribuidora.

Cabe aclarar que a efectos de este trabajo, sólo se tienen en cuenta archivos de tipo A3, B1 y M1, esto es debido a que el proceso F1 se gestiona ya en el sistema que se encarga de la facturación de la compañía y el resto de tipologías son de extrema rareza, hasta el punto de que en los dos años de vida de Ekiluz, jamás se han gestionado procesos de este tipo.

Imaginemos que hemos comprado un piso de nueva construcción y tenemos que dar de alta el suministro eléctrico, por lo que elegimos Viesgo pues nos ofrece las mejores condiciones:

- Viesgo enviará a la distribuidora un archivo A3 de paso 01, en el que indicará toda la información del punto de suministro, la tarifa y potencia contratadas, la información del cliente y la fecha en la que solicita la activación del suministro.
- El distribuidor enviará un archivo de paso 02 de aceptación o de rechazo tras validar la solicitud, indicando la fecha estimada de activación si es aceptada o si es rechazada, los motivos del rechazo.
- Si se produce alguna incidencia, el distribuidor envía un archivo de paso 03, informando a la comercializadora, que una vez las solucione, enviará un paso 13 para notificar a la distribuidora

- Si al momento de hacer los trabajos en campo se da algún problema que imposibilite hacer la activación, la distribuidora enviará un paso 04 para notificar de que no es posible hacer la activación.
- Si no hay ningún problema a la hora de acometer los trabajos de campo, la distribuidora enviará un paso 05, notificando a la comercializadora de la activación, que se hará efectiva al siguiente día hábil.

De nuevo, la CNMC contempla la posibilidad de que la comercializadora quiera anular el proceso antes de ser activado de forma definitiva dando de nuevo como plazo hasta un día hábil tras el envío del archivo de paso 05 por parte de la distribuidora, si esto ocurriera:

- La comercializadora enviará un archivo de paso 06 a la comercializadora, indicando los motivos de la anulación del proceso.
- Si ha llegado en tiempo y forma, la distribuidora enviará a la comercializadora un archivo de paso 07 de aceptación. En caso contrario, enviará un paso 07 de rechazo.

En la figura 4 podemos ver un esquema que resume el proceso descrito [6]:

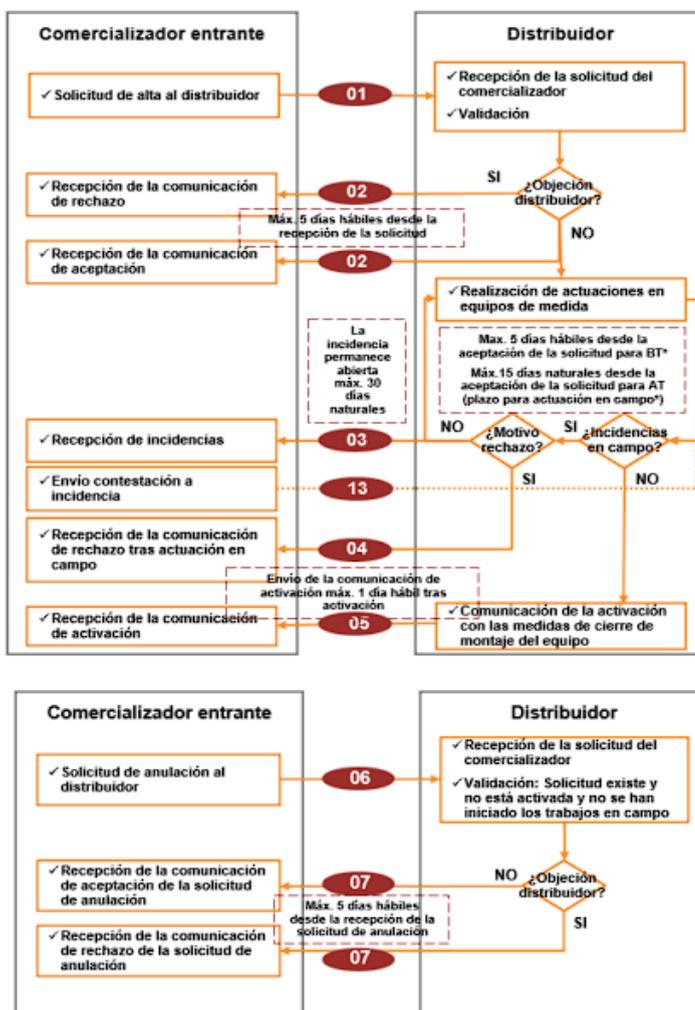


Figura 4- Flujograma que resume el proceso previamente descrito

3. ¿Cómo gestiona la comercializadora los intercambios? - Estado del arte ¹

A continuación se describe el proceso de creación y gestión de las solicitudes de forma manual, esta información ha sido extraída mediante un proceso de entrevistas con la coordinadora del BackOffice de Ekiluz, el cual será descrito más en profundidad en el apartado de captura y especificación de requisitos.

Los operadores de Backoffice de la compañía siguen un proceso de creación y gestión de distintos objetos de Salesforce CRM con el objetivo de llevar la trazabilidad completa del proceso de intercambio de información, desde la posible venta en la que hacen una oferta telefónica o por email a un posible cliente interesado, hasta la activación del suministro eléctrico.

Para entender este proceso, debemos conocer los siguientes objetos:

- **Objeto Oferta:** A la hora de intentar captar un nuevo cliente o una posible renovación de contrato para la compañía, el operador de Backoffice registrará en el CRM un objeto de tipo *Oferta* a través de la interfaz de la figura 5, este objeto recogerá las condiciones que se ofrecerán al cliente así como los datos que se hayan obtenido de este.

Oferta
MQ-ES0021000005092433ZT

Número de Oferta: 00001125
Nombre de la oportunidad: OP-0000000393
Nombre de la cuenta: Krean
Estado: Completada

Completada

✓ Marcar Estado como completado(a)

Detalles Relacionado

Número de Oferta: 00001125
Nombre de Oferta: MQ-ES0021000005092433ZT
Tipo de registro de Oferta

Última modificación por: Usuario BackOffice BackOffice, 16/1/2023, 9:36
Creado por: Usuario BackOffice BackOffice, 16/1/2023, 9:36
Nombre de la cuenta

Demo generar contrato

Figura 5- Visualización de un objeto Oferta en Salesforce CRM

- **Objeto Oportunidad [7]:** Este objeto es creado para hacer seguimiento de las ofertas realizadas a un cliente, en este objeto, se indicarán los datos del cliente conocidos hasta el momento, las ofertas que se le hayan hecho al cliente, el estado actual de la oportunidad, la fecha de vencimiento a partir de la que no se le harán nuevas ofertas al cliente y la probabilidad de cerrar la venta, que dependerá de la cercanía a la fecha límite y de la cantidad de ofertas realizadas. El operador puede gestionar oportunidades en una interfaz como la de la figura 6.

¹ En virtud de la protección de los datos de los clientes de la compañía, todos los datos que se muestran a continuación corresponden a clientes y puntos de suministro ficticios, extraídos del Salesforce de preproducción de la compañía.

Oportunidad
OP-0000000393

+ Seguir Modificar Cambiar propietario Multi-Site Quote

Nombre de la cuenta: [Krean](#) Etapa: En tratamiento Fecha de cierre: 19/1/2023

En tratamiento Pendiente validación En validación Pendiente BO Pendiente Agencia Cerrada ✓ Marcar Etapa como completado(a)

Detalles Relacionado

Nombre de la oportunidad: OP-0000000393 Fecha de cierre: 19/1/2023

Nombre de la cuenta: [Krean](#) Propietario de oportunidad: [Usuario BackOffice BackOffice](#)

Acciones disponibles

Ofrecer Producto Gestionar Oferta

Figura 6- Visualización de un objeto Oportunidad en Salesforce CRM

- **Objeto Pedido [8]:** Si se consigue cerrar la venta o nos llega una solicitud por parte de un cliente como una modificación del contrato o una baja. El operador registrará un pedido en Salesforce usando el formulario que se puede observar en la figura 7, el pedido guardará el tipo de modificación, el cliente, la fecha de activación de ese pedido si la hubiera, el punto de suministro perteneciente al cliente sobre el que se ejecutará la solicitud y el estado del pedido.

Pedido
00000397

Modificar Duplicar Cambiar propietario

Nombre de la cuenta: [Rodrigo](#) Número de contrato: [00000258](#) Estado: Aceptada Oferta

✓ Aceptada Pendiente cliente Rechazada ✓ Marcar Estado como completado(a)

Detalles Relacionado

Número de pedido: 00000397 Fecha activación servicio

Estado: Aceptada Fecha inactivación servicio: 23/12/2021

Fecha inicial del pedido: 5/1/2022 Oferta

Acciones disponibles

SWD (3+) SWD_BAJA

Figura 7- Visualización de un objeto Pedido en Salesforce CRM

- **Objeto SWD (SWitch Document):** Este es el objeto clave para este trabajo, es el objeto que consultaremos y modificaremos a la hora de manejar las solicitudes de intercambio.

Para cada orden, el operador deberá crear un objeto SWD, en base a los datos introducidos en pedido, donde irá actualizando toda la información referente al estado del intercambio de información según las respuestas que reciba de la distribuidora, en base a la información de cada uno de estos objetos, cuya visualización en Salesforce aparece en la figura 8, se generarán el archivos XML que se envía a la distribuidora, y se realizarán actualizaciones en base a las respuestas recibidas.

SWD
SWD_C1_3103

+ Seguir Modificar Duplicar Versión para imprimir

Código de solicitud	Estado de Solicitud	Código Proceso ATR	Cliente	Caso Relacionado
000000000126	Aceptada	C1	Diego	

Detalles Relacionado

SWD SWD_C1_3103	Propietario Usuario BackOffice BackOffice
Código de solicitud 000000000126	Fecha Solicitud 7/4/2023, 12:00
Cliente Diego	Fecha Última Lectura 2022-11-18

History Notes

Figura 8- Visualización de un objeto SWD en Salesforce CRM

Una vez creado el SWD, el operador genera a mano el XML de acuerdo al esquema correspondiente al tipo de solicitud recibida, para posteriormente subir el archivo al portal de la distribuidora correspondiente² a los 6 primeros caracteres del código único del punto de suministro.

Una vez subido el archivo, deberá comprobar periódicamente si la distribuidora ha respondido a las solicitudes y deberá actualizar el objeto SWD correspondiente en Salesforce de acuerdo a la información que contenga la respuesta recibida.

En la figura 9 podemos observar un diagrama ER que resume las relaciones entre los distintos objetos

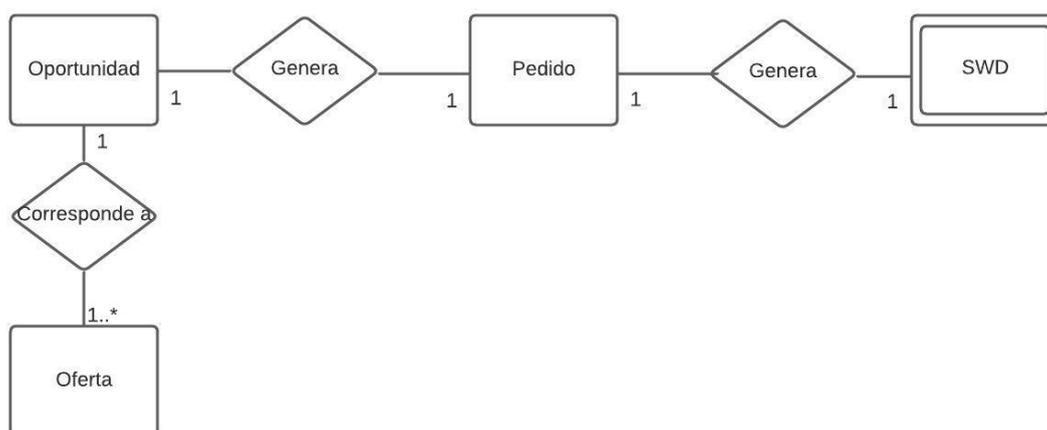


Figura 9- Diagrama ER correspondiente a los objetos de Salesforce CRM implicados en el proceso de intercambio de información

² Por motivos de confidencialidad, no se puede mostrar ninguna figura para ejemplificar el funcionamiento de estos portales

4. Tecnologías utilizadas

En este capítulo vamos a hablar sobre el stack de tecnologías que se utilizarán para implementar la herramienta, y el por qué se ha tomado la decisión de incorporarlas:

Python: Es el lenguaje en el que vamos a desarrollar la plataforma, se ha elegido sobre otros lenguajes como Java o C# por las siguientes razones:

- Es el lenguaje más extendido en el contexto de la empresa, lo que facilita la asimilación de nuevos desarrolladores en el proyecto si se requiriera implementar nuevas funcionalidades o hacer mantenimiento.
- Está altamente integrado mediante librerías con otras tecnologías utilizadas en el proyecto, esto nos permite acceder a los mecanismos de persistencia que vamos a utilizar (CRM y contenedor de archivos) de forma casi transparente usando librerías como:
 - Simple Salesforce: que permite hacer peticiones a la API de Salesforce de forma casi transparente para el usuario, de forma muy similar a como lo haríamos con una BBDD relacional al uso.
 - Librería de Azure para Python: Que de nuevo, permite enviar y descargar archivos del contenedor de forma sencilla.
 - ETree: Para la serialización/deserialización de los datos a XML, que permitirá generar los archivos XML de intercambio que enviamos a distribuidora, y leer los datos que llegan en las respuestas.
 - Pandas: Es una de las librerías más extendidas en el mundo de la ciencia de datos [9], nos permite leer y manipular datos en una estructura de datos en formato tabular llamada *DataFrame*, cuya estructura se puede ver en la figura 10, esto es especialmente útil para leer las consultas que hacemos sobre Salesforce y poder generar los archivos simplemente accediendo a la tabla, y leer los datos que nos llegan en los XML de respuesta.
 - Tkinter: Es la librería de python más utilizada para generar clientes de escritorio, nos ofrece la posibilidad de desarrollar frontend de cierta complejidad con una barrera de entrada bastante baja, además de contar con bastante documentación e incluso proyectos de integración con algunas plataformas de modelado como Figma.
- Orientación a objetos: Nos permitirá hacer arquitecturas de n capas e implementar patrones de diseño de forma similar a como lo haríamos en otros lenguajes.

The diagram shows a Pandas DataFrame with 10 rows (index 0-9) and 9 columns. The columns are labeled as 'Mountain', 'Height (m)', 'Range', 'Coordinates', 'Parent mountain', 'First ascent', 'Ascents bef. 2004', and 'Failed attempts bef. 2004'. The data is annotated with orange boxes and lines: 'index labels' points to the row indices, 'column names' points to the column headers, and 'data' points to the cell contents. Specific cells highlighted include 'Baltoro Karakoram' in the Range column of row 1, 'Kangchenjunga' in the Mountain column of row 2, and '79' in the Ascents bef. 2004 column of row 5.

	Mountain	Height (m)	Range	Coordinates	Parent mountain	First ascent	Ascents bef. 2004	Failed attempts bef. 2004
0	Mount Everest / Sagarmatha / Chomolungma	8848	Mahalangur Himalaya	27°59'17"N 86°55'31"E	NaN	1953	>>145	121.0
1	K2 / Qogir / Godwin Austen	8611	Baltoro Karakoram	35°52'53"N 76°30'48"E	Mount Everest	1954	45	44.0
2	Kangchenjunga	8586	Kangchenjunga Himalaya	27°42'12"N 88°08'51"E	Mount Everest	1955	38	24.0
3	Lhotse	8516	Mahalangur Himalaya	27°57'42"N 86°55'59"E	Mount Everest	1956	26	26.0
4	Makalu	8485	Mahalangur Himalaya	27°53'23"N 87°05'20"E	Mount Everest	1955	45	52.0
5	Cho Oyu	8188	Mahalangur Himalaya	28°05'39"N 86°39'39"E	Mount Everest	1954	79	28.0
6	Dhaulagiri I	8167	Dhaulagiri Himalaya	28°41'48"N 83°29'35"E	K2	1960	51	39.0
7	Manaslu	8163	Manaslu Himalaya	28°33'00"N 84°33'35"E	Cho Oyu	1956	49	45.0
8	Nanga Parbat	8126	Nanga Parbat Himalaya	35°14'14"N 74°35'21"E	Dhaulagiri	1953	52	67.0
9	Annapurna I	8091	Annapurna Himalaya	28°35'44"N 83°49'13"E	Cho Oyu	1950	36	47.0

Figura 10- Estructura de un *Dataframe* de la librería Pandas

Salesforce: Es el CRM que utiliza Ekiluz, lo usaremos como mecanismo de persistencia debido a su facilidad de acceso mediante API y a que proporciona un lenguaje de dominio bastante similar a SQL, llamado SOQL [10], y cuya sintaxis básica podemos ver en la figura 11. Puede ser utilizado de forma similar a una base de datos objeto-relacional, gracias a estas características podremos consultar las nuevas solicitudes de intercambio de forma sencilla y actualizarlas en base a las respuestas que recibamos utilizando diccionarios (Si se accede mediante librería) o en formato JSON (si accedemos mediante API).

```
SELECT Id, Email, Name, Languages__c FROM Contact
```

Query Results - Total Rows: 20

Id	Email	Name	Languages__c
0033h000005n0L...	rose@edge.com	Rose Gonzalez	English
0033h000005n0Ln...	sean@edge.com	Sean Forbes	English
0033h000005n0Lo...	jrogers@burlington.com	Jack Rogers	
0033h000005n0Lp...	pat@pyramid.net	Pat Stumuller	French, English
0033h000005n0Lq...	a_young@dickenson.com	Andy Young	English
0033h000005n0Lr...	barr_tim@grandhotels.com	Tim Barr	English
0033h000005n0Ls...	bond_john@grandhotels.com	John Bond	English
0033h000005n0Lt...	spavlova@uog.com	Stella Pavlova	English
0033h000005n0Lu...	lboyle@uog.com	Lauren Boyle	English
0033h000005n0Lv...	b.levy@expressl&t.net	Babara Levy	English
0033h000005n0Lx...	j.davis@expressl&t.net	Josh Davis	English
0033h000005n0Lx...	jane_gray@uoa.edu	Jane Grey	English
0033h000005n0Ly...	asong@uog.com	Arthur Song	English
0033h000005n0Lz...	ajames@uog.com	Ashley James	English
0033h000005n0M...	triplej@uog.com	Tom Ripley	English

Figura 11- Estructura de una consulta sencilla utilizando SOQL

Azure Blob Storage: Contenedor de Blobs alojado en la nube de Azure, la principal ventaja que ofrece sobre otras tecnologías similares es que permite autorización SAS (Shared Access

Signature), que proporciona acceso a los usuarios a un contenedor sin necesidad de login, bastando con introducir la URI del contenedor, que tiene un token embebido en el cual podremos configurar expiración, limitaciones sobre ciertas carpetas o recursos, etc.

Esto tiene como principal ventaja la posibilidad de poder acceder al contenedor de forma programática utilizando solamente la URI con el token SAS, lo cual nos quitará la complejidad de tener que manejar también logins de usuario para el contenedor [11].

En este contenedor se volcarán los archivos XML generados por el programa y el operador deberá también cargar los archivos de respuesta que extraiga de los portales web de las distribuidoras.

Prefect: Es una tecnología de orquestado de flujos de trabajo para Python, la cual se encarga de coordinar, desplegar y ejecutar código de forma automática y programada, basta con abrir un servidor, donde desplegamos un script Python con las funciones que queremos ejecutar, permite ejecución tanto en local como en su entorno cloud dedicado, aunque en este caso se hará de forma local ya que las tareas que se ejecutan son bastante ligeras en cómputo.

Además, como podemos ver en la figura 12, tiene un cliente web donde podremos llevar registro de los resultados de las ejecuciones que se realizan y poder detectar posibles errores.

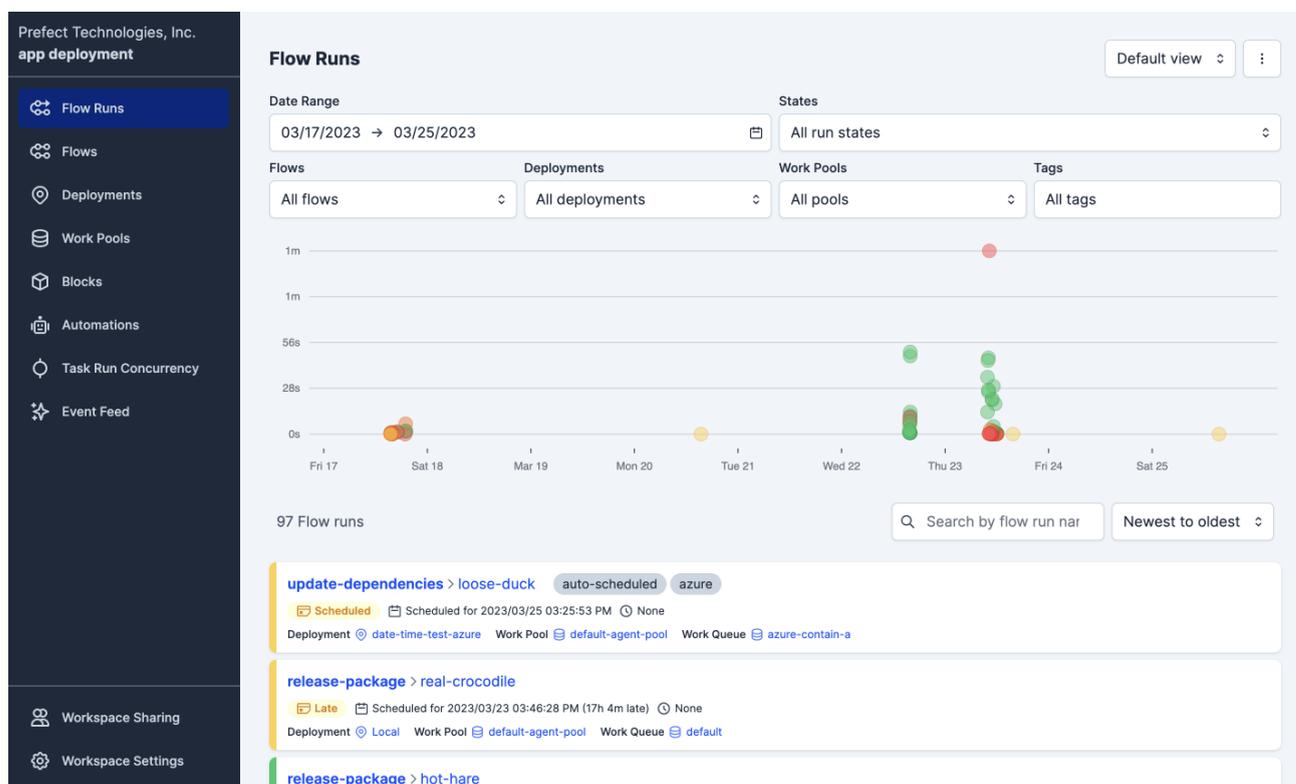


Figura 12- Aplicación web de prefect donde podremos consultar el estado de los diferentes flujos de trabajo

5. Cronograma y metodología

5.1 Cronograma

Se han seguido los tiempos mostrados en la figura 13 para la realización del proyecto, cubriendo las 300 horas requeridas para la obtención de los créditos correspondientes al trabajo de fin de grado.

Como vemos en la figura 13, el proceso de captura de requisitos ha sido más largo de lo que cabría esperar debido a que se ha necesitado un gran trabajo de investigación y entendimiento del contexto funcional explicado en el capítulo 2, pero tras haber contextualizado correctamente la información funcional, el proceso se ha agilizado considerablemente debido a la facilidad de contacto con las fuentes de captura implicadas.

Con el objetivo de optimizar el desarrollo lo máximo posible, se ha llevado a cabo el desarrollo en paralelo tanto de la herramienta como de las pruebas, procurando además hacer pruebas de aceptación con negocio de manera periódica, sobretodo para la validación de los formatos para los archivos generados y de los datos que se actualizan en el CRM, y para validar cambios en la capa de presentación.

La capa de negocio (donde más adelante veremos que están los generadores y lectores de los archivos) ha llevado comparativamente mucho más tiempo que las demás capas, esto es debido a que la capa de persistencia se ha desarrollado de una forma muy ágil gracias a que se han utilizado las tecnologías y librerías descritas anteriormente y a que la capa de presentación para herramientas de este tipo está estandarizada a nivel del equipo donde he estado encuadrado, por lo que solo ha sido necesaria una adaptación para cubrir las necesidades concretas de la herramienta.

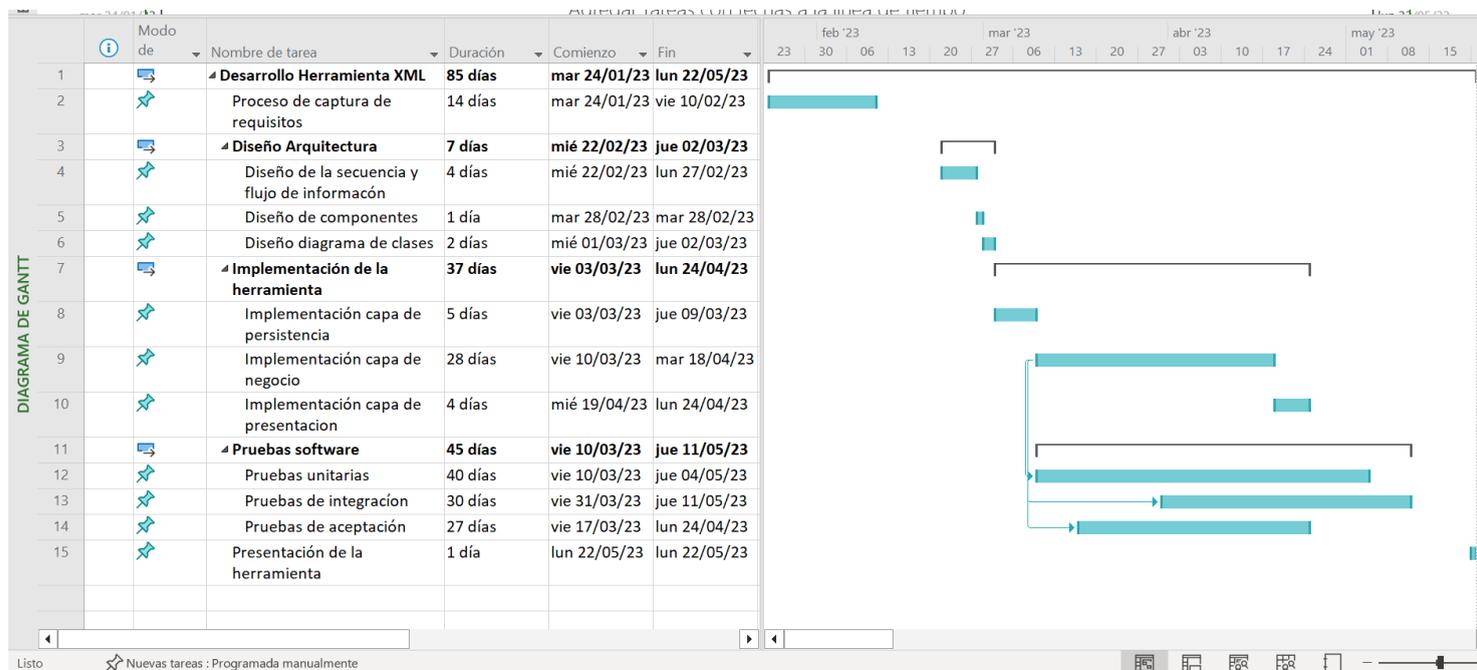


Figura 13- Diagrama de Gantt y cronología del proyecto

5.2 Metodología

En cuanto a la metodología seguida durante la realización del proyecto, no se podría catalogar al 100% como una metodología ágil, ya que los tipos de archivo que se deben de tratar y generar son limitados y con un formato cerrado por una entidad externa a la compañía como es la CNMC, por lo que en ese sentido, se conocen desde el inicio del proyecto los objetivos que se pretende satisfacer y cómo deben ser los archivos generados en cuanto a formato, sin lugar a modificaciones demandadas por el cliente, además, dado que los casos a implementar eran conocidos desde el comienzo, y el tiempo para implementarlos limitado, el trabajo se ha distribuido de forma lineal durante toda su duración, sin organizarse en sprints.

Sin embargo, durante el día a día del desarrollo del proyecto, si se establecen mecanismos y rutinas propias de metodologías ágiles como SCRUM, como podrían ser reuniones diarias para conocer el estado del desarrollo o validaciones periódicas con negocio para comprobar que los distintos pasos que se dan son correctos e incorporar posibles mejoras.

6. Captura de requisitos

Una vez explicado todo el contexto que envuelve a la realización de este trabajo, vamos a comenzar detallando el proceso de captura de requisitos realizado para llevar a cabo su especificación.

6.1 Fuentes de requisitos

Para realizar el proceso de captura de requisitos, se ha recurrido a las fuentes detalladas entre las tablas 2 y 4, se han considerado las mejores fuentes de requisitos debido a su disponibilidad, conocimiento sobre el proceso a automatizar y su conocimiento y capacidad de transmitirnos el contexto funcional:

Id	Stk-01
Fuente	Project Manager Ekiluz (Stakeholder)
Información	<p>Del jefe del proyecto se pretende extraer la siguiente información:</p> <ul style="list-style-type: none"> - Funcionamiento de aquellos procesos involucrados en los intercambios de información. - Cómo se integra el intercambio de información dentro del mercado eléctrico. - Agentes implicados en los distintos pasos del intercambio. - Regulaciones sobre el sector. - Organización del CRM de la compañía. -
Prioridad	Alta
Contacto	Iván Pedraja Lombilla, Project Leader en NTT Data Santander

Tabla 2- Descripción de la fuente de captura de requisitos Stk-01

Id	Stk-02
Fuente	Coordinadora del BackOffice de Ekiluz (Stakeholder)
Información	<p>De la coordinadora de BackOffice de la compañía cliente se pretende extraer la siguiente información:</p> <ul style="list-style-type: none"> - Proceso de creación manual de los archivos XML. - Proceso de procesamiento manual de las respuestas de la distribuidora. - Procesos de creación de los objetos implicados dentro del CRM de la compañía. - Estadísticas sobre aquellos procesos involucrados en la gestión de los intercambios más costosos en tiempo.
Prioridad	Alta
Contacto	María Herrero Mazón, Coordinadora BackOffice Ekiluz

Tabla 3- Descripción de la fuente de captura de requisitos Stk-02

Id	Doc-01
Fuente	Documentación de la CNMC referente al proceso de intercambio de información

Información	Mediante el estudio de estos documentos, se espera obtener un conocimiento detallado de las diferencias entre el formato de los distintos archivos a la hora de su generación y procesamiento, así como entender las diferencias semánticas entre los distintos tipos de intercambio.
Prioridad	Alta
Forma de Acceso	Portal web de la CNMC

Tabla 4- Descripción de la fuente de captura de requisitos Doc-01

6.2 Actividades de captura de requisitos

Con el objetivo de realizar la captura de requisitos propiamente dicha, se han llevado a cabo las actividades detalladas entre las tablas 5 y 7:

Id	ACR-000							
Técnica	Entrevista							
Participantes	Project Manager Ekiluz							
Objetivo	El objetivo de la entrevista es reunirse con el stakeholder involucrado en el proceso de la organización y dirección técnica del negocio, de forma que se pueda obtener una visión del funcionamiento de dicho negocio y poder obtener una idea inicial de sus necesidades							
Información a obtener	<ul style="list-style-type: none"> (1) Una descripción teórica del proceso de intercambio de información (2) Una descripción de los distintos objetos implicados en el funcionamiento del intercambio dentro del CRM de la compañía (3) Una idea general de cuáles son las necesidades del negocio en lo que refiere a los intercambios de información (4) Las normativa y leyes que intervienen en el proceso 							
Duración	Aproximadamente 30 minutos por entrevista ¹							
Lugar	La actividad se llevará a cabo en las oficinas de NTT Data, en una franja horaria establecida por ambas partes. De esta forma, obtendremos más disponibilidad y un trato más inmediato, además de poder contar con recursos como pizarras y pantallas de conferencia que nos proporciona la empresa dentro de la oficina. En el caso de que no pueda ser así, se realizará de forma telemática mediante Microsoft Teams.							
Recursos Necesarios	Ingeniero de Requisitos Junior (para el rol de entrevistador y de secretario). Sala adecuada para la entrevista. Sistema de grabación. ²							
Coste y Tiempo Estimado	<table border="1"> <thead> <tr> <th>Concepto</th> <th>Importe</th> </tr> </thead> <tbody> <tr> <td>Desplazamiento</td> <td>1'30€</td> </tr> <tr> <td>Hora Ing. Junior (5€/hora)</td> <td>30€</td> </tr> </tbody> </table> <p>*El cálculo se ha hecho suponiendo que la actividad se desarrolla en el lugar en las oficinas de NTTData y que el entrevistador vive en la misma ciudad, por lo que el desplazamiento se realizará mediante autobús. En las horas del ingeniero se incluyen:</p> <ul style="list-style-type: none"> - Costes de preparación (1 hora) - Procesamiento de la actividad (4 horas) - Desplazamiento (1 hora). 		Concepto	Importe	Desplazamiento	1'30€	Hora Ing. Junior (5€/hora)	30€
Concepto	Importe							
Desplazamiento	1'30€							
Hora Ing. Junior (5€/hora)	30€							

Comentarios adicionales	<p>¹ Se realizarán varias entrevistas, debido a la facilidad que entrevistador y entrevistado tienen para reunirse ya que trabajan para la misma empresa, además es probable que esto facilite el tiempo de procesamiento de la entrevista al entrevistador, debido a lo complicado que puede llegar a ser en ocasiones el entendimiento del mercado eléctrico.</p> <p>² El sistema de grabación se utilizará siempre que el entrevistado esté cómodo con él y nos dé su consentimiento. Puede ser grabación de audio y video o solo de audio, siempre a elección del entrevistado.</p>
-------------------------	---

Tabla 5- Descripción de la actividad de captura de requisitos ACR-000

Id	ACR-001							
Técnica	Entrevista							
Participantes	Coordinadora de BackOffice Ekiluz							
Objetivo	El objetivo de la entrevista es reunirse con el stakeholder involucrado en el trabajo completo de gestión de los intercambios de información desde la captación de clientes/procesamiento de sus solicitudes, hasta el procesamiento de los archivos de respuesta por parte de la distribuidora							
Información a obtener	<p>(1) Una descripción del flujo seguido por los operadores para la gestión de los intercambios.</p> <p>(2) Una lista con aquellos procesos más costosos en tiempo.</p> <p>(3) Una lista con criterios de uso del CRM (Campos obligatorios, nombres, etc).</p> <p>(4) Una descripción de cómo funcionan las campañas de contratación de la compañía.</p>							
Duración	Aproximadamente 30 minutos por entrevista ¹							
Lugar	La actividad se llevará a cabo de manera telemática por Microsoft Teams, en una franja horaria establecida por ambas partes. se ha decidido así ya que se pedirán en numerosas ocasiones muestras de pantalla para mostrar de forma más realista los procesos que se pretenden entender							
Recursos Necesarios	Ingeniero de Requisitos Junior (para el rol de entrevistador y de secretario). Sala adecuada para la entrevista. Sistema de grabación. ²							
Coste y Tiempo Estimado	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Concepto</th> <th style="text-align: left;">Importe</th> </tr> </thead> <tbody> <tr> <td>Desplazamiento</td> <td>1'30€</td> </tr> <tr> <td>Hora Ing. Junior (5€/hora)</td> <td>30€</td> </tr> </tbody> </table> <p>*El cálculo se ha hecho suponiendo que el entrevistador desarrollará la actividad en las oficinas de NTTData y que el entrevistador vive en la misma ciudad, por lo que el desplazamiento se realizará mediante autobús. En las horas del ingeniero se incluyen:</p> <ul style="list-style-type: none"> - Costes de preparación (1 hora) - Procesamiento de la actividad (4 horas) - Desplazamiento (1 hora). 		Concepto	Importe	Desplazamiento	1'30€	Hora Ing. Junior (5€/hora)	30€
Concepto	Importe							
Desplazamiento	1'30€							
Hora Ing. Junior (5€/hora)	30€							

Comentarios adicionales	<p>¹ Se realizarán varias entrevistas, debido a la alta disponibilidad de la entrevistada y con el objetivo de hacer validaciones y cubrir posibles cambios.</p> <p>² El sistema de grabación se utilizará siempre que el entrevistado esté cómodo con él y nos dé su consentimiento. Puede ser grabación de audio y video o solo de audio, siempre a elección del entrevistado.</p>
-------------------------	--

Tabla 6- Descripción de la actividad de captura de requisitos ACR-001

Id	ACR-002					
Técnica	Observación					
Objetos a observar	Documentación de la CNMC referente al proceso de intercambio de información.					
Objetivo	Conocer y entender de forma concreta cómo se regula el proceso de intercambio y cómo deben ser los archivos XML implicados.					
Información a obtener	<p>(1) Un documento con la semántica de cada tipo de archivo de intercambio y los distintos pasos que lo componen.</p> <p>(2) Un documento con un resumen de cómo deben ser los formatos de cada archivo</p>					
Duración	5 horas.					
Lugar	Puesto de trabajo del ingeniero Junior. ¹					
Recursos Necesarios	<ul style="list-style-type: none"> • 1 Ingeniero de Requisitos Junior 					
Coste y Tiempo Estimado	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Concepto</th> <th style="text-align: left;">Importe</th> </tr> </thead> <tbody> <tr> <td>Horas Ingeniero Junior(5€/hora)</td> <td>25€</td> </tr> </tbody> </table> <p>En los costes del Ingeniero Senior se estima lo siguiente:</p> <ul style="list-style-type: none"> • 3 horas de estudio de la documentación • 2 horas de elaboración de los documentos resumen 		Concepto	Importe	Horas Ingeniero Junior(5€/hora)	25€
Concepto	Importe					
Horas Ingeniero Junior(5€/hora)	25€					
Comentarios adicionales	¹ Podrá ser tanto dentro de la oficina de NTTData como en su casa mediante teletrabajo					

Tabla 7- Descripción de la actividad de captura de requisitos ACR-002

Con estas actividades de captura se espera obtener la suficiente información como para entender las necesidades a cubrir mediante la realización de este trabajo, así como conocer con la suficiente profundidad los distintos tipos de archivo, como se integran tanto en el CRM del cliente como en la parte de distribuidora y además entender cómo son los formatos de cada archivo, lo cual será imprescindible para su serialización/deserialización.

6.2 Objetivos

Durante la realización de las actividades de captura, se han inferido los siguientes objetivos a satisfacer para la realización de la herramienta:

- **Obj-000:** Reducir el tiempo dedicado por Backoffice a estas tareas: Para ello es necesario automatizar la generación de los archivos de paso 1 correspondientes a las tipologías más comunes de entre las descritas así como el procesamiento de las respuestas de la distribuidora, en este caso paso 2, paso 4 y paso 5.
Se debe dar soporte a las siguientes tipologías:
 - C1 - Cambio de compañía sin modificación en el contrato.
 - C2 - Cambio de compañía con modificación en el contrato.
 - A3 - Alta de nuevos puntos de suministro.
 - B1 - Baja de un punto de suministro.
 - M1 - Modificación de datos del contrato.
- **Obj-001:** Minimizar la tasa de errores: Para esto se deberá dar soporte a la creación automática de los objetos SWD de Salesforce descritos en el capítulo 3, tratando de reducir el número de registros que se crean manualmente.
- **Obj-002:** Permitir cambios de CRM o de mecanismo de persistencia de las solicitudes: Se deberá crear una herramienta que sea modificable de manera sencilla en caso de que sea necesario migrar el CRM de la compañía o se decida cambiar la forma actual de cambiar los datos de las solicitudes.

7. Casos de uso

En base a los resultados obtenidos en las actividades de captura de requisitos previamente descritas, se ha obtenido el diagrama de casos de uso de la figura 14:

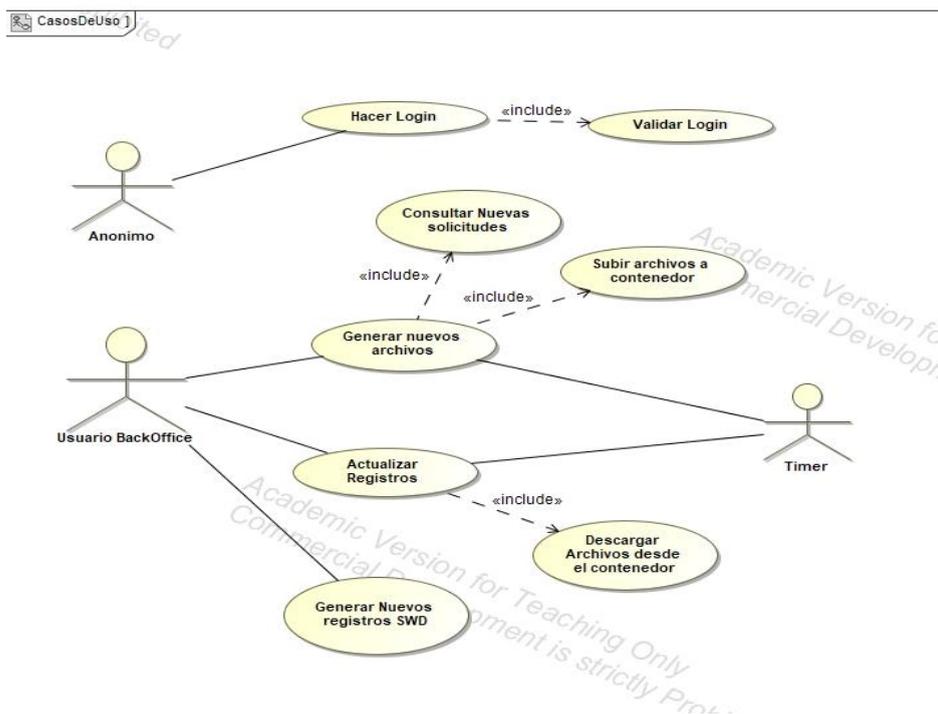


Figura 14- Diagrama de casos de uso de la aplicación

Como podemos ver, se ha establecido un caso de uso para materializar cada uno de los objetivos extraídos, además del caso de uso *Hacer Login*, que corresponde al requisito no funcional RNF003 descrito en la tabla 11, el cual veremos más adelante.

El actor Timer correspondería al flujo de trabajo automatizado con Prefect.

A continuación, se detallan los casos de uso extraídos en las plantillas entre las tablas 8 y 11:

Id + Nombre	CU000 - Hacer Login
Actor Principal	Anónimo
Actores Secundarios	API del CRM a especificar por el usuario
Descripción	Con el objetivo de poder acceder a los datos alojados en el CRM, el usuario deberá hacer login en el sistema a través de la aplicación.
Evento de Activación	El usuario presiona el botón "Configuración" en el menú principal
Precondición	- El usuario debe tener acceso a las credenciales para acceder al sistema
Garantías Si Éxito	El sistema guarda las credenciales y podrá acceder a los datos guardados en el CRM
Garantías Mínimas	- El login se podrá reintentar
Escenario Principal	<ul style="list-style-type: none"> - El usuario rellena un formulario indicando el CRM al que acceder, la URL de este, su usuario, su contraseña, y su Token de seguridad y presiona el botón, modificar configuración. - El sistema guarda las credenciales introducidas y manda una petición a la API del CRM correspondiente para validar la información, después, muestra una ventana de confirmación indicando que se ha guardado la información de inicio de sesión correctamente
Extensiones	<p>1a- El usuario introduce credenciales inválidas:</p> <ul style="list-style-type: none"> - El usuario rellena un formulario indicando el CRM al que acceder, la URL de este, su usuario, su contraseña, y su Token de seguridad y presiona el botón,"Modificar configuración". - El sistema guarda las credenciales introducidas y manda una petición a la API del CRM correspondiente para validar la información, pero recibe una respuesta que indica que la información es incorrecta. El sistema muestra una ventana de error indicando que no se ha podido iniciar sesión. <p>1b Error de conexión:</p> <ul style="list-style-type: none"> - El usuario rellena un formulario indicando el CRM al que acceder, la URL de este, su usuario, su contraseña, y su Token de seguridad y presiona el botón,"Modificar configuración". - El sistema guarda las credenciales introducidas y manda una petición a la API del CRM correspondiente para validar la información, pero no recibe una respuesta. El sistema muestra una ventana de error indicando que no se ha podido iniciar sesión.

Tabla 8 - Descripción del caso de uso CU000

Id + Nombre	CU - 001 Generar Archivos XML
Actor Principal	Usuario Backoffice, Timer
Actores Secundarios	API del CRM especificado Contenedor de Blobs de Azure
Objetivo	Obj-000
Descripción	El sistema deberá generar los archivos XML correspondientes a todas aquellas solicitudes de intercambio que no hayan sido previamente procesadas

Evento de Activación	Usuario pulsa el botón “Generar nuevos archivos” El timer se dispara por que llega la hora indicada
Precondición	El usuario deberá estar logueado en el sistema.
Garantías Si Éxito	Se subirán todos los archivos generados al contenedor de blobs de Azure.
Garantías Mínimas	Los archivos que se hayan podido generar se guardarán en una carpeta local. El usuario es notificado de los resultados del programa
Escenario Principal	<ul style="list-style-type: none"> - El sistema lanza una consulta sobre el CRM para extraer la información sobre las solicitudes etiquetadas como Nuevas - El sistema genera los archivos con la información correspondiente y los guarda en una carpeta local. - El sistema inicia la subida masiva de los archivos generados al contenedor de blobs de Azure. - El sistema borra los archivos guardados en local. - El sistema muestra al usuario una ventana indicando que los archivos han sido correctamente generados.
Extensiones	<p>1a - Error debido a campos obligatorios vacíos:</p> <ul style="list-style-type: none"> - El sistema lanza una consulta sobre el CRM para extraer la información sobre las solicitudes etiquetadas como Nuevas - El sistema genera los archivos con la información correspondiente y los guarda en una carpeta local. - El sistema se encuentra con que algún campo obligatorio no se ha encontrado, por lo que recolecta el Id de la solicitud y manda una notificación por microsoft teams al usuario antes de pasar a la siguiente solicitud. - El sistema inicia la subida masiva de los archivos generados al contenedor de blobs de Azure. - El sistema borra los archivos guardados en local. - El sistema muestra al usuario una ventana de error indicando el número de solicitudes erróneas que se han encontrado <p>1b- Error en el contenedor de blobs de Azure:</p> <ul style="list-style-type: none"> - El sistema lanza una consulta sobre el CRM para extraer la información sobre las solicitudes etiquetadas como Nuevas - El sistema genera los archivos con la información correspondiente y los guarda en una carpeta local. - Debido a algún error a la hora de conectarse al contenedor de blobs de Azure, el sistema no puede iniciar la subida de archivos, el sistema lanzará un diálogo de error al usuario informando del error e indicando que cuenta con las copias locales de los archivos <p>1c - Error de conexión con el CRM:</p> <ul style="list-style-type: none"> - El sistema no se puede conectar al CRM debido a algún tipo de error de conexión - El sistema muestra al usuario un diálogo de error indicando la incidencia

Tabla 9- Descripción del caso de uso CU001

Id + Nombre	CU - 002 Procesar Respuestas
Actor Principal	Usuario Backoffice, Timer
Actores Secundarios	API del CRM especificado Contenedor de Blobs de Azure
Objetivo	Obj-000
Descripción	El sistema deberá procesar los archivos XML de respuesta que envía la distribuidora correspondientes a todas aquellas solicitudes que se hayan iniciado previamente

Evento de Activación	Usuario pulsa el botón "Procesar respuestas" El timer se dispara por que llega la hora indicada
Precondición	El usuario deberá estar logueado en el sistema. El usuario deberá subir al contenedor los archivos de respuesta obtenidos
Garantías de Éxito	Si Se actualizará en el CRM el estado de todas las solicitudes respondidas
Garantías Mínimas	Los archivos que se hayan procesar serán actualizados en el CRM El usuario es notificado de los resultados del programa
Escenario Principal	<ul style="list-style-type: none"> - El sistema busca en el contenedor de blobs de Azure los archivos con las respuestas y los guarda en una carpeta local. - El sistema inicia la lectura masiva de los archivos descargados y actualiza los registros en el CRM en base a la información guardada. - El sistema borra los archivos guardados en local y mueve a una carpeta de historial en el contenedor los archivos ya procesados. - El sistema muestra al usuario una ventana indicando que los archivos han sido correctamente generados.
Extensiones	<p>1a - Error debido a campos obligatorios vacíos:</p> <ul style="list-style-type: none"> - El sistema busca en el contenedor de blobs de Azure los archivos con las respuestas y los guarda en una carpeta local. - El sistema inicia la lectura masiva de los archivos descargados y actualiza los registros en el CRM en base a la información guardada. - El sistema se encuentra con que algún campo obligatorio no se ha encontrado en el archivo de respuesta, por lo que recolecta el Id de la solicitud y manda una notificación por microsoft teams al usuario antes de pasar a la siguiente solicitud. - El sistema borra los archivos guardados en local y mueve a una carpeta de historial en el contenedor los archivos ya procesados. - El sistema muestra al usuario una ventana de error indicando el número de solicitudes erróneas que se han encontrado <p>1b- Error en el contenedor de blobs de Azure:</p> <ul style="list-style-type: none"> - Debido a algún error a la hora de conectarse al contenedor de blobs de Azure, el sistema no puede iniciar la subida de archivos, el sistema lanzará un diálogo de error al usuario informando del error e indicando que cuenta con las copias locales de los archivos <p>1c - Error de conexión con el CRM:</p> <ul style="list-style-type: none"> - El sistema busca en el contenedor de blobs de Azure los archivos con las respuestas y los guarda en una carpeta local. - El sistema inicia la lectura masiva de los archivos descargados y trata de actualizar los registros en el CRM en base a la información guardada. - El sistema no se puede conectar al CRM debido a algún tipo de error de conexión. - El sistema muestra al usuario un diálogo de error indicando la incidencia. <p>1d- Solicitud no encontrada en el CRM:</p> <ul style="list-style-type: none"> - El sistema busca en el contenedor de blobs de Azure los archivos con las respuestas y los guarda en una carpeta local. - El sistema inicia la lectura masiva de los archivos descargados y actualiza los registros en el CRM en base a la información guardada. - Durante la subida, el sistema detecta que no existe una solicitud en el CRM que corresponda a la indicada en el archivo XML. - por lo que recolecta el Id de la solicitud y manda una notificación por microsoft teams al usuario antes de pasar a la siguiente solicitud. - El sistema borra los archivos guardados en local y mueve a una carpeta de historial en el contenedor los archivos ya procesados. - El sistema muestra al usuario una ventana de error indicando el número de solicitudes erróneas que se han encontrado

--	--

Tabla 10- Descripción del caso de uso CU002

Id + Nombre	CU - 003 Generar nuevas solicitudes de intercambio
Actor Principal	Usuario Backoffice, Timer
Actores Secundarios	API del CRM especificado
Objetivo	Obj-001
Descripción	El sistema deberá generar en el CRM de forma masiva todas las solicitudes de intercambio correspondientes a los pedidos que no tengan una solicitud ya asignada
Evento de Activación	Usuario pulsa el botón "Generar nuevas solicitudes SWD"
Precondición	El usuario deberá estar logueado en el sistema.
Garantías Si Éxito	Se generarán todas las solicitudes correspondientes a los pedidos que no hayan tenido solicitud previamente
Garantías Mínimas	Se generarán todas aquellas solicitudes que no tengan errores de forma El usuario es notificado de los resultados del programa
Escenario Principal	<ul style="list-style-type: none"> - El sistema lanza una consulta sobre el CRM para extraer la información sobre los pedidos que no tengan solicitudes de intercambio. - El sistema inserta en el CRM las solicitudes con la información de los pedidos correspondientes. - El sistema muestra al usuario una ventana indicando que las solicitudes han sido correctamente generadas.
Extensiones	<p>1a - Error debido a campos obligatorios vacíos:</p> <ul style="list-style-type: none"> - El sistema lanza una consulta sobre el CRM para extraer la información sobre los pedidos que no tengan solicitudes de intercambio. - El sistema inserta en el CRM las solicitudes con la información de los pedidos correspondientes. - El sistema se encuentra con que algún campo obligatorio no se ha encontrado, por lo que recolecta el Id de la solicitud y manda una notificación por microsoft teams al usuario antes de pasar a la siguiente solicitud. - El sistema muestra al usuario una ventana de error indicando el número de solicitudes erróneas que se han encontrado <p>1c - Error de conexión con el CRM:</p> <ul style="list-style-type: none"> - El sistema no se puede conectar al CRM debido a algún tipo de error de conexión - El sistema muestra al usuario un diálogo de error indicando la incidencia

Tabla 11- Descripción del caso de uso CU003

8. Requisitos no funcionales

Durante las actividades de captura también se extrajeron los requisitos no funcionales para la aplicación mencionados entre las tablas 12 y 15:

Id + Nombre	RNF000 - Permitir ampliaciones a nuevas tipologías de archivo
Descripción	Para extender la vida útil de la herramienta, se debe dar soporte a la incorporación de nuevos tipos de archivo que la CNMC pueda incorporar tras nuevos cambios de regulación.

Tabla 12- Descripción del requisito no funcional RNF000

Id + Nombre	RNF001 - Permitir cambios en la forma de persistencia de datos
Descripción	Para extender la vida útil de la herramienta, se debe dar soporte a posibles cambios que se den en la forma que tiene la compañía de persistir los datos de las solicitudes.

Tabla 13- Descripción del requisito no funcional RNF001

Id + Nombre	RNF002 - Hacer lecturas y generaciones de archivos de forma masiva
Descripción	Para hacer la herramienta lo más eficiente posible, es necesario que las cargas y modificaciones de datos se hagan de forma masiva, pudiendo procesar todas las solicitudes que sean necesarias con una sola ejecución

Tabla 14- Descripción del requisito no funcional RNF002

Id + Nombre	RNF003 - Limitar el acceso a los usuarios no registrados en el CRM de la compañía
Descripción	Para hacer la herramienta más segura y proteger los datos de los clientes que se revelan en los archivos XML, se ha de limitar el acceso a la herramienta a aquellos usuarios que no estén registrados en el CRM que esté usando la compañía en ese momento

Tabla 15- Descripción del requisito no funcional RNF003

A todos estos requisitos no funcionales extraídos se les ha dado una prioridad alta, debido a que todos resultan claves para realizar una herramienta completa y que cubra las necesidades del negocio de la forma más efectiva posible, obteniendo un rendimiento adecuado y cubriendo la herramienta ante posibles cambios de CRM o cambios regulatorios por parte de la CNMC.

9. Arquitectura y diseño de la aplicación

En este capítulo vamos a describir como se ha llevado a cabo el diseño arquitectural de la herramienta, así como varios aspectos relevantes de su implementación.

9.1 Diseño de componentes

Como podemos ver en la figura 15, en la aplicación contamos con 5 componentes, los cuales se distribuyen en 3 capas según sus funciones dentro de la aplicación de la siguiente manera:

- Capa de persistencia: Aquí convivirán el componente CRMStrategies, que dará soporte a la consulta y carga de información en el CRM de la compañía, desde la capa de negocio se invocarán las distintas consultas necesarias sobre el CRM para extraer los datos necesarios para generar los archivos XML y también se cargarán los datos que se extraigan de las respuestas.

También nos encontramos el componente BlobStorageAccess, que servirá tanto para subir al contenedor de archivos de Azure los XML generados por la capa de negocio como para realizar la descarga de los archivos de respuesta que generan los archivos.

- Capa de negocio: Aquí nos encontramos también dos componentes, el primero es el módulo Management que sirve como controlador de la vista y que se comunica con persistencia para lanzar las consultas sobre el CRM y enviar después esos datos al generador de archivos e invocar también al manejador del contenedor para subir y descargar los archivos XML.

También tenemos el módulo generador de archivos, este módulo recibirá los datos del CRM desde el módulo de gestión y se encargará tanto de generar los archivos XML de paso 1 como de leer y extraer la información que traen los archivos de respuesta para actualizar los registros en CRM, con el que se comunicará mediante una clase DataLoader que servirá como puente entre este módulo y el de persistencia.

- Capa de presentación: Es aquí donde se encuentra la interfaz gráfica de la herramienta para que pueda ser controlada de forma manual por el operador de BackOffice, será controlada por el módulo de gestión dentro de la capa de negocio.

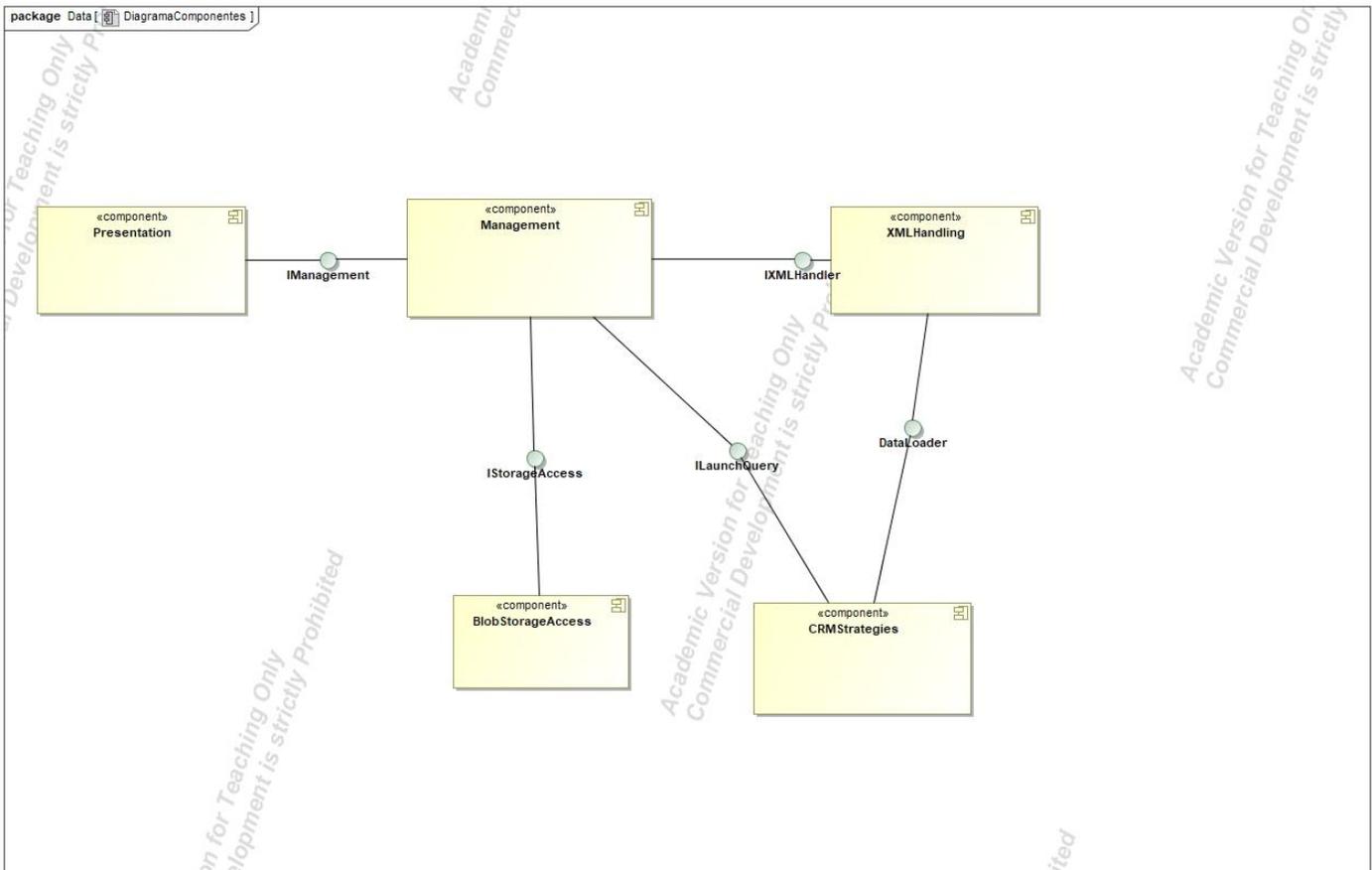


Figura 15- Diagrama de componentes de la aplicación

9.2 Diagrama de clases e implementación

Debido a que en realidad la herramienta trata de realizar un proceso de Extracción, transformación y carga de datos, y no de implementar un sistema de información al uso, no se ha seguido un enfoque de diseño domain driven, por lo que realmente no contamos con un modelo de dominio claro que tener en cuenta para el diseño de la aplicación.

No obstante, para hacer el sistema lo más flexible y escalable posible, se ha seguido un paradigma 100% orientado a objetos que sigue el diagrama de clases expuesto en la figura 16:

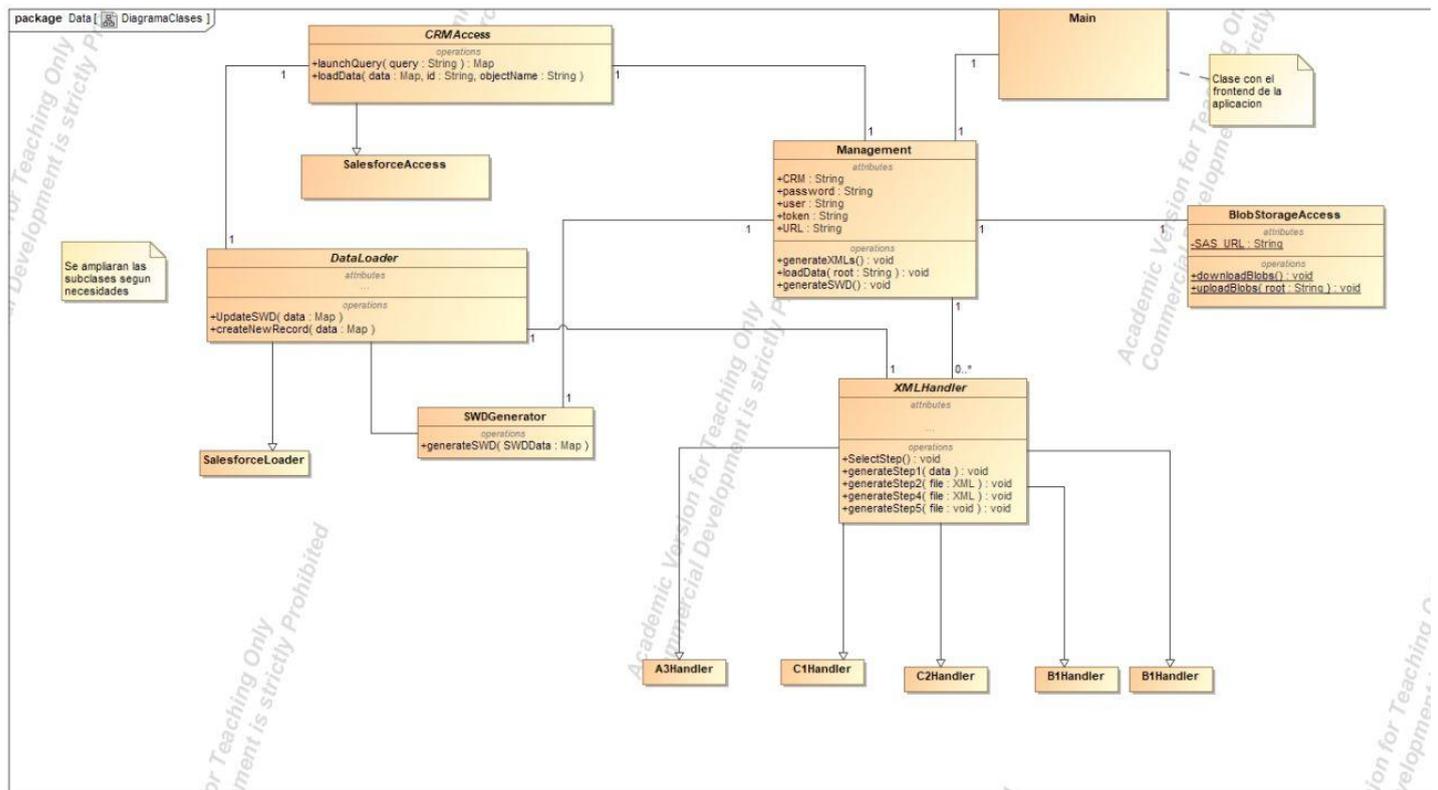


Figura 16- Diagrama de clases de la aplicación

Podemos dividir las clases según la funcionalidad que cumplen en la herramienta:

- Clase abstracta CRMAccess y subclase SalesforceAccess: Se trata de la clase de acceso a la información alojada en un CRM, se puede asimilar a una clase DAO más tradicional, donde hacer operaciones CRUD en los registros, con la única diferencia que en vez de usar algún framework de mapeo objeto-relacional, accederemos mediante la API que el CRM nos proporcione, y que al no tener modelo de dominio serializaremos las respuestas y las peticiones en un diccionario o con un dataframe de Pandas ya que Python nos da mucha flexibilidad para ello. Las claves o nombres de columnas se normalizarán para poder enviar la información sea procesada por las clases generadoras de archivos de forma independiente, pues es lógico pensar que las convenciones de nombres de algunos campos cambien según el CRM al que se acceda.

En este caso, se ha decidido hacer con una clase abstracta y una estrategia concreta para implementar el acceso a Salesforce, esto nos permite no solo poder adaptar la herramienta de forma sencilla a cambios de CRM, si no que el operador pudiera seleccionar a qué CRM acceder en tiempo de ejecución.

- Clase abstracta XMLHandler y subclases C1Handler, C2Handler, A3Handler, B1Handler y M1Handler: Son las clases generadoras y procesadoras de los archivos XML, se ha aplicado el patrón de diseño *strategy* para poder cambiar el tipo de archivo que generamos en tiempo de ejecución a medida que leamos los archivos.

Cada una de las funciones corresponde a un paso en el proceso de intercambio de información, se ha decidido tomar este enfoque en vez de el uso de clases de dominio debido a que la CNMC contempla campos condicionales que podrían no aparecer, y algunos campos que dependen de otros (Por ejemplo, si una persona es física, se escribe un campo *DNI*, y si es jurídica se escribe un campo *NIF*), esta circunstancia es más fácil de

gestionar con este enfoque ya que podemos escribir funciones complejas para escribir los archivos de paso 1 o leer los archivos de respuesta.

- Clase abstracta DataLoader y subclase SalesforceLoader: Esta clase hace de puente entre las clases generadoras y el acceso al CRM, este intermediario es necesario debido a que las clases generadoras son independientes del CRM que se utiliza, por lo que recibe de la capa de persistencia la información con los nombres de los campos normalizados, y es necesario desnormalizarlos para insertar datos en el CRM concreto que se utilice, SalesforceLoader es la estrategia concreta para poder alterar los nombres de los campos a como aparecen en los objetos de salesforce.
- BlobStorageAccess: Es la clase de acceso al contenedor de archivos de azure, proporciona la posibilidad de descargar los archivos a procesar y subir los archivos que se generan de forma programática.
- Clase Management: Es la clase de gestión de la herramienta, controla la vista de forma similar como haríamos con un *presenter* en una arquitectura MVP, se encarga de definir las estrategias según el CRM seleccionado, y el tipo de archivo que haya que generar/procesar, realiza las consultas sobre el CRM y sobre el contenedor de archivos y le pasa la información extraída de estos procesos a la clase generadora correspondiente según el tipo de archivo.
- Clase Main: Contiene la interfaz gráfica de la aplicación, es controlada por la clase Management.

9.3 Secuencia de generación de los archivos XML

Como vemos en la figura 17, para generar los archivos XML, la clase Management lanzará una consulta sobre el CRM para obtener los datos de aquellas solicitudes que aún no estén procesadas (son marcadas con estado "Nuevo") una vez obtiene la tabla con Pandas, lee el tipo de solicitud e invoca la función de generación del archivo al objeto Handler correspondiente a este, una vez generado, llama al objeto DataLoader para actualizar el estado de la solicitud y escribe en disco el archivo.

Una vez generados todos los archivos, se subirán al contenedor de azure y se borrarán del disco duro local una vez completada la operación.

Si se detecta algún error durante la generación de los archivos, el proceso no para, si no que se captura la excepción en el Objeto Management que llevará la cuenta de los errores y enviará mediante webhook una notificación por microsoft teams al equipo de BackOffice para que revisen la solicitud fallida, además, se notificará por pantalla del número de errores ocurridos durante el proceso una vez termine.

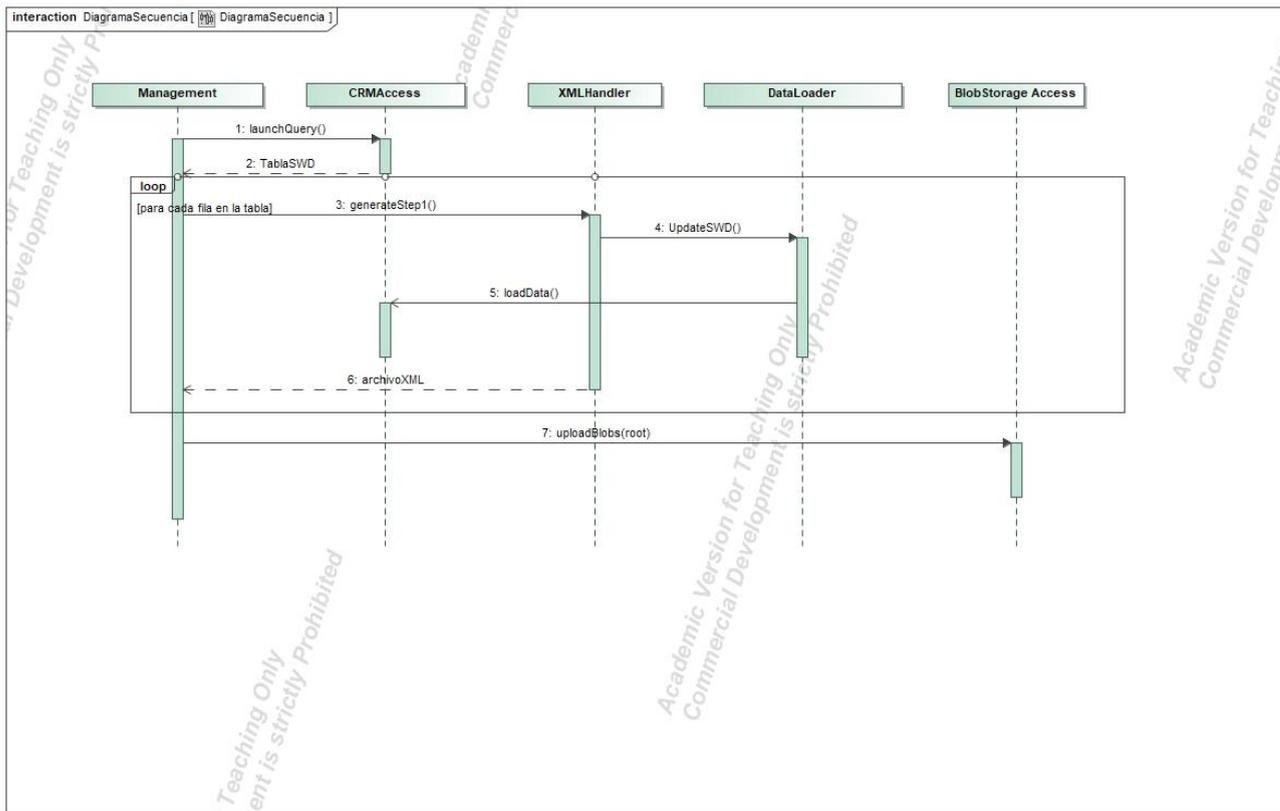


Figura 17- Diagrama de secuencia de la generación de archivos XML

9.4 Secuencia de lectura de los archivos XML

Como se observa en la figura 18, a la hora de procesar la información, la clase Management descarga todos los archivos de respuesta pendientes de procesar al disco duro local, una vez descargados, se consigue la información de este utilizando Pandas y se envía la información a la clase Handler correspondiente, que reconocerá el paso correspondiente al archivo para invocar la función pertinente, tras esto, se generará un diccionario con la información contenida en el archivo y se le pasa al DataLoader que desnormaliza los nombres de las columnas según el CRM y actualiza en este la información obtenida invocando a la clase de acceso al CRM.

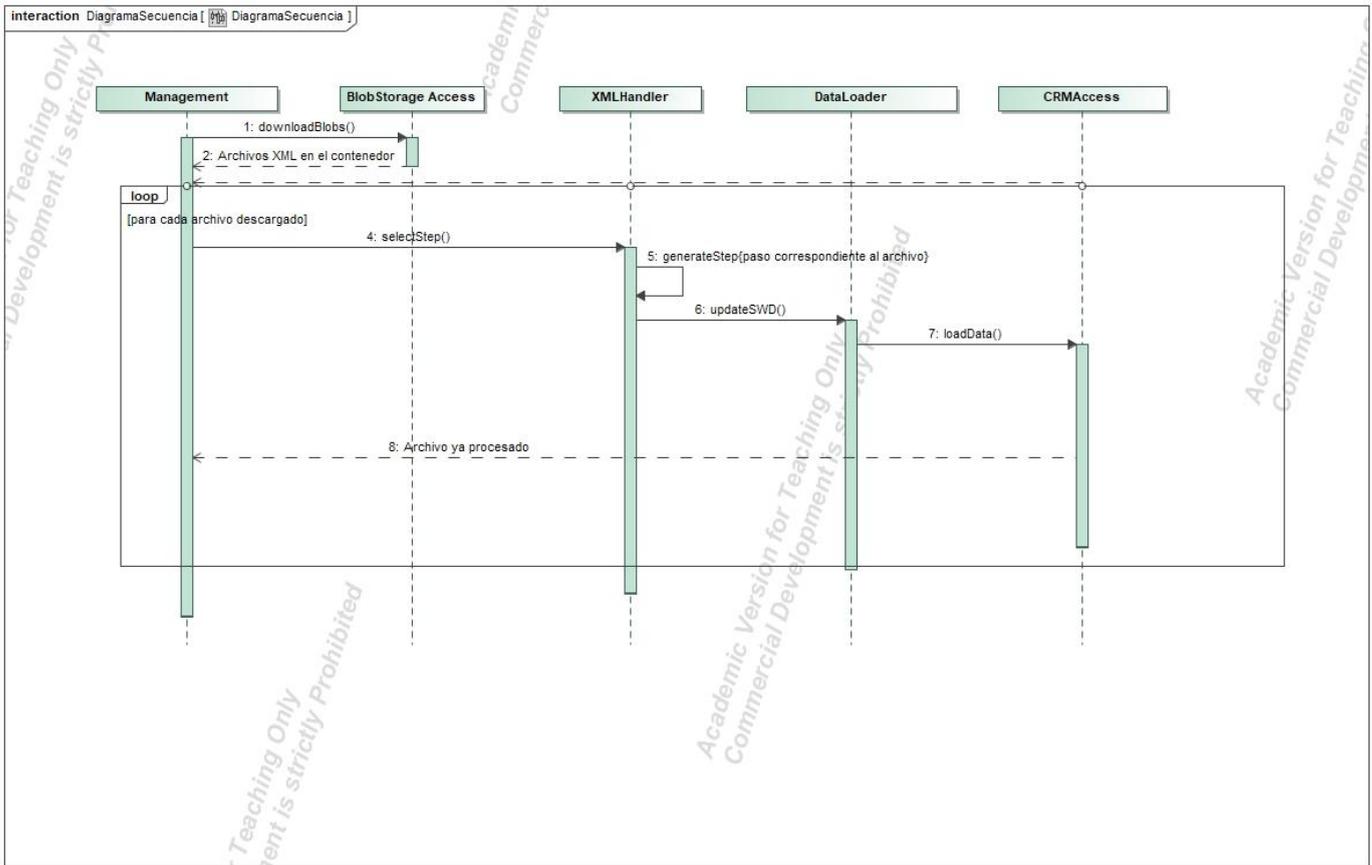


Figura 18- Diagrama de secuencia de procesamiento de archivos XML

9.5 Secuencia de creación automática de nuevos registros SWD

Como se describe en la figura 19, para introducir nuevas solicitudes en el CRM, la case Management lanza una query sobre la tabla orden para extraer los pedidos que aún no tienen solicitud creada, esta tabla se lee utilizando Pandas y se envía cada fila al generador de solicitudes, que filtra la información necesaria para la creación de la solicitud y la envía al DataLoader, que desnormaliza los nombres de las columnas para el CRM correspondiente y crea los nuevos registros.

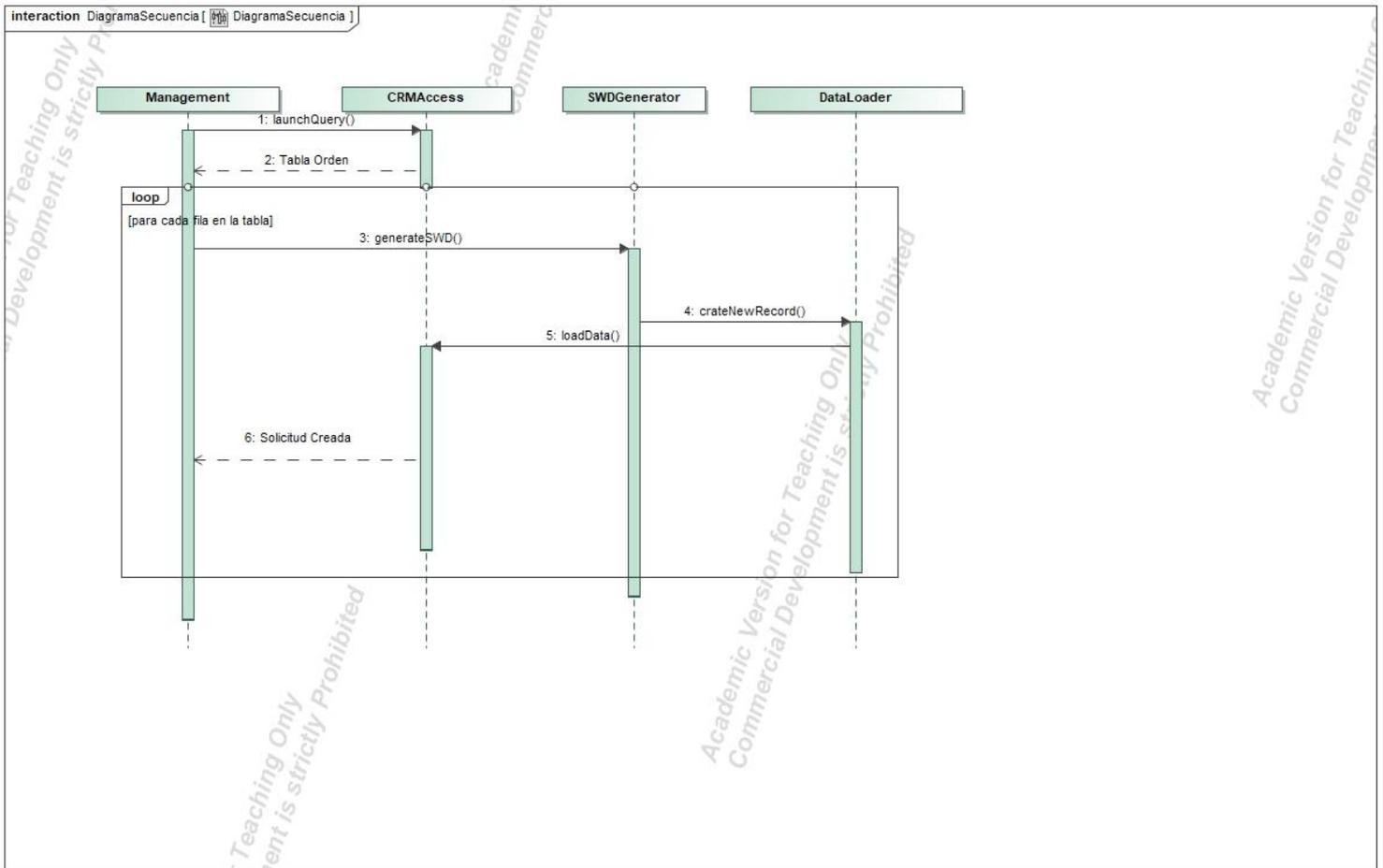


Figura 19- Diagrama de secuencia de creación de nuevas solicitudes

9.6 Automatización del proceso

Como se mencionó en la especificación de casos de uso del capítulo 7, los procesos definidos en las figuras 16 y 17 se podrán accionar de forma manual a través de la interfaz gráfica, pero también se ha creído beneficioso realizar la ejecución diaria de este proceso.

Como comentamos en el capítulo 4, esta característica se implementará usando Prefect, que proporciona un framework completo para orquestar y automatizar la ejecución de código de forma sencilla.

Prefect cuenta con varios objetos que se coordinan para hacer esta tarea:

- **Tasks** [12]: Es la unidad más básica, corresponde a un fragmento de código que se ejecuta de forma independiente y sin interactuar con otros objetos del framework, en este caso, se han definido dos *tasks* correspondientes a dos funciones que invocan a la clase Management para realizar cada uno de los procesos.

Para crear una *task* en python basta con anotar una función con el decorador `@task`

- **Flows** [13]: Corresponde a un conjunto de *tasks* que se ejecutan de forma ordenada o paralela, permite el paso de mensajes entre funciones y puede recibir parámetros desde el exterior, para esta automatización se ha hecho un *flow* sencillo que ejecute las dos *tasks* creadas previamente de forma secuencial.

Para crear un *flow* en python basta con anotar una función con el decorador `@flow`

Cada *flow* o conjunto de estos que creamos debe ser encapsulado en un archivo de despliegue para poder ser subido a un servidor local o en cloud y ser programado, ejecutado y registrado por el cliente de Prefect instalado en nuestra máquina [14].

Para crear un despliegue bastará con ejecutar el comando “prefect deployment build {ruta del *flow* que deseamos ejecutar}” esto dará como resultado un descriptor de extensión .yaml que deberemos editar para configurar diferentes opciones, en este caso, se ha establecido un crontab para que se ejecute de lunes a viernes a las 9:00 de la mañana hora española.

Una vez creado y configurado el descriptor de despliegue, se ha abierto un servidor local de Prefect con el comando “prefect server start” y se ha añadido el despliegue con el comando “prefect deployment apply {ruta del descriptor}” una vez completado este paso, ya tendremos nuestro flujo de trabajo funcionando.

10. Pruebas software

Para testar la herramienta se han establecido 3 niveles de prueba:

- **Pruebas unitarias:** En este caso, se han hecho pruebas unitarias de las clase generadoras de los archivos XML para comprobar que generan los archivos adecuados y que contienen los datos recibidos desde la capa de persistencia, también se han probado las funciones de lectura de archivos, asegurando que generan los diccionarios con los datos acordes a los incluidos en los archivos de prueba antes de ser pasado a persistencia.

Se han probado también los métodos de acceso al CRM, aprovechando los entornos de prueba del CRM para poder comprobar que las consultas lanzadas dan los resultados esperados.

Además se ha probado también la clase de acceso al contenedor de azure para comprobar que la descarga de archivos es correcta y que la subida surte el efecto deseado

Debido a la falta de tiempo y a que se ha considerado innecesario dado que realmente estas clases sólo hacen de puente entre los datos y las clases generadoras, no se han hecho pruebas unitarias ni de la clase Management ni de los DataLoader, decidiendo probarse directamente a nivel de integración.

Los casos de prueba se han diseñado teniendo en cuenta el código implementado, por lo que se consideran de caja blanca, se ha tomado esta decisión para poder controlar que aquellos campos condicionales o con dependencias tengan el comportamiento esperado.

- **Pruebas de integración:** Una vez comprobado que la capa de acceso da los datos esperados en base a las consultas proporcionadas y que la generación y procesamiento de los archivos XML era correcta, se han hecho pruebas de integración para asegurar que la clase SalesforceLoader realmente conseguía desnormalizar las columnas y actualizar el estado de las solicitudes en el CRM de forma correcta, y que la clase Management lanzaba las consultas y pasaba los datos a las clases generadoras correctamente, además de comunicarse correctamente con el contenedor para subir y descargar los archivos.
- **Pruebas de aceptación:** Para validar con negocio formatos de archivos, adecuación de la interfaz y asegurar que los procesos automatizados son realmente sustitutivos de las tareas que el BackOffice de la compañía realiza de forma manual.

Se ha decidido prescindir de pruebas de interfaz para este proyecto, debido a la simplicidad de esta y que realmente no muestra información de ningún tipo sobre los archivos generados, simplemente informa de que el proceso se ha terminado correctamente o da un aviso si se ha producido alguna excepción durante el transcurso de este, la adecuación de la interfaz ha sido validada con negocio en la fase de pruebas de aceptación.

Para implementar las pruebas se ha usado la librería unittest de Python, que proporciona un framework completo para el desarrollo de pruebas software de forma similar a como lo haríamos en JUnit y además da soporte para mocks de forma nativa.

11. Conclusiones y desarrollo futuro

11.1 Conclusiones

En esta memoria se ha recogido todo el proceso seguido a la hora de desarrollar un software de automatización de una tarea compleja y tediosa de gestionar de manera manual como es la gestión de los intercambios de información entre comercializadores y distribuidores eléctricos, tratando de explicar de una forma resumida su contexto funcional, las particularidades de la tarea y el por qué de las decisiones tomadas a la hora de diseñar y modelar la aplicación con el objetivo de hacerla lo más escalable y mantenible posible.

El mayor reto de este proceso ha sido sin duda el tratar sumar a los conocimientos técnicos adquiridos como estudiante de ingeniería informática una gran cantidad de conceptos de funcionamiento del mercado eléctrico, con el objetivo de modelar una arquitectura orientada a obtener la máxima efectividad posible para desarrollar el proceso y hacerla lo más flexible y adaptable a cambios posible, en el marco de un sector altamente regulado como es el de la electricidad, que sin embargo está sujeto a cambios de manera habitual debido al contexto de crisis energética mundial que estamos viviendo, en el que los gobiernos tratan de introducir decretos y cambios normativos de forma constante con el objetivo de paliar sus efectos.

Además de esto, el hecho de tener que automatizar un proceso previamente gestionado por humanos remarca la importancia de la comunicación constante con los implicados y de la ingeniería de requisitos, ante problemáticas como la tratada en este trabajo, es imprescindible capturar de una forma correcta y con el mayor nivel de detalle posible el funcionamiento del proceso que queremos automatizar, ya que de lo contrario se puede llegar a hacer muy cuesta arriba desarrollar una solución que satisfaga el problema de forma correcta.

Utilizar nuevas tecnologías menos habituales en el contexto del grado, pero bastante extendidas en el ámbito laboral como Pandas, Prefect o Salesforce han supuesto un aprendizaje y un reto extra, pero que ha dado un resultado realmente positivo y sin duda han aportado a hacer el mejor desarrollo posible.

Como futuro ingeniero informático, hacer un trabajo de estas características en el entorno empresarial ha sido una experiencia que sin duda marcará los primeros años de mi carrera profesional, pues ha supuesto un cambio total tanto de métodos de trabajo como de forma de modelar soluciones y comunicarme con mi entorno respecto a lo experimentado en el ámbito universitario, lo que sin duda me ha hecho un desarrollador mucho más completo que al momento de empezar.

A fecha de entrega de este trabajo, la herramienta está puesta en producción y ha sido presentada con éxito a negocio, habiendo sido ya utilizada para la tramitación de alguna solicitud aislada que ha surgido durante las últimas semanas, en las que el cliente no ha tenido mucha actividad comercial.

Actualmente el cliente está preparando el lanzamiento al mercado de nuevas promociones comerciales para la temporada de verano y se espera que la herramienta ayude a reducir la carga de trabajo derivada de este lanzamiento a los operadores de backoffice, permitiendo que la carga de este departamento se distribuya hacia otras tareas.

11.2 Desarrollo futuro

A pesar de la puesta en producción y los resultados descritos, la herramienta puede seguir siendo desarrollada para mejorar su usabilidad y ampliar aún más la variedad de casos que maneja, algunos posibles desarrollos futuros serían:

- Despliegue de la automatización en algún entorno cloud: Gracias a la facilidad de integración de los servidores de prefect en contenedores docker de forma nativa, podríamos aprovechar para desplegar el flujo de trabajo ya automatizado en algún contenedor cloud, por ejemplo podríamos aprovechar la integración nativa de contenedores docker en la nube de azure mediante el servicio ACI (Azure Container Instances).
- Ampliar el rango de tipos de intercambio: Aunque los tipos de intercambio que se han dejado fuera de la herramienta son muy poco comunes, su implementación sería interesante para evitar posibles picos de carga de trabajo desde backoffice debido a su gestión manual.
- Automatizar la comunicación con las distribuidoras: Este paso sería el último eslabón que necesita la herramienta para hacer el proceso completamente automático y de hecho se hubiera incluido en este trabajo si no fuera por que las distribuidoras actualmente no disponen de ningún tipo de servicio para enviar solicitudes o descargar repuestas de forma programática, no obstante, debería ser implementado si estos servicios fueran publicados por parte de las distribuidoras, hecho plausible debido a que ya existen servicios web habilitados y estandarizados para otros procesos.

12. Referencias

[1] Gobierno de España. Anuncio de la Comisión Nacional de los Mercados y la Competencia por el que se hace pública la Resolución de 21 de julio de 2022 por la que se modifican las resoluciones de 20 de diciembre de 2016 y de 17 de diciembre de 2019 sobre formatos de los ficheros de intercambio. *BOE-B-2022-23914*. 2022, pages 36797 - 36798

[2] Redacción Interempresas. Repsol lanza Ekiluz para promover cooperativas ciudadanas de generación renovable. *Interempresas*. 2021.
<https://www.interempresas.net/Estaciones-servicio/Articulos/350933-Repsol-lanza-Ekiluz-para-promover-cooperativas-ciudadanas-de-generacion-renovable.html>

[3] Comisión Nacional de los Mercados y la Competencia. Resolución por la que se modifican las resoluciones de 20 de diciembre de 2016 y de 17 de diciembre de 2019 sobre formatos de los ficheros de intercambio. 2022, pages 3 - 7.

[4] Comisión Nacional de los Mercados y la Competencia. Formato Fichero C1. 2022, page 12

- [5] Comisión Nacional de los Mercados y la Competencia. Formato Fichero C1. 2022, page 13
- [6] Comisión Nacional de los Mercados y la Competencia. Formato Fichero A3. 2022, page 10
- [7] Salesforce.com. Opportunity.
https://developer.salesforce.com/docs/atlas.en-us.object_reference.meta/object_reference/sforce_api_objects_opportunity.htm
- [8] Salesforce.com. Order.
https://developer.salesforce.com/docs/atlas.en-us.object_reference.meta/object_reference/sforce_api_objects_order.htm
- [9] Statista.com. Frameworks used in Python for data science in 2021.
<https://www.statista.com/statistics/1338424/python-use-frameworks-data-science/#:~:text=In%202021%2C%20Numpy%20and%20Pandas,and%2055%20percent%20share%20respectively.>
- [10] Salesforce.com Trailhead. Escribir consultas SOQL.
https://trailhead.salesforce.com/es/content/learn/modules/apex_database/apex_database_soql
- [11] Learn.microsoft.com. Grant limited access to Azure Storage resources using shared access signatures (SAS). <https://learn.microsoft.com/en-us/azure/storage/common/storage-sas-overview>
- [12] Prefect Docs. Tasks. <https://docs.prefect.io/2.10.13/concepts/tasks/>
- [13] Prefect Docs. Flows. <https://docs.prefect.io/2.10.13/concepts/flows/>
- [14] Prefect Docs. Deployments. <https://docs.prefect.io/2.10.13/concepts/deployments/>