



## Facturae a Salesforce

**Nombre Estudiante:** Roberto Rodríguez Vispo

**Programa:** Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones (MISTIC)

**Nombre Consultor:** Jordi Castellà Roca

**Centro:** UOC

**Fecha entrega:** 7 de enero del 2020



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	Facturae a Salesforce
<b>Nombre del autor:</b>	Roberto Rodríguez Vispo
<b>Nombre del consultor:</b>	Jordi Castellà Roca
<b>Fecha de entrega (mm/aaaa):</b>	01/2020
<b>Área del Trabajo Final:</b>	Seguridad en aplicaciones web
<b>Titulación:</b>	Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones (MISTIC)
<b>Resumen del Trabajo (máximo 250 palabras):</b>	
<p>Las empresas y administraciones han adoptado la digitalización por su impacto en la competitividad y el medioambiente. Un ejemplo significativo son las facturas electrónicas, donde las administraciones han desplegado su normativa legal y herramientas. Al igual que las facturas en papel, las facturas electrónicas deben proporcionar legibilidad, autenticidad e integridad. En el caso de España el formato definido es el Facturae.</p> <p>La factura electrónica debe incorporarse a los sistemas de información de las empresas, y uno de los más importantes son los Customer Relationship Management (CRM). La plataforma de CRM dominante es Salesforce.</p> <p>Sin embargo, no existe ninguna aplicación que permita incorporar ni categorizar las facturas en la plataforma Salesforce. Este proceso debe ser seguro y garantizar la confidencialidad de la información almacenada en el CRM.</p> <p>El objetivo de este trabajo ha sido desarrollar un sistema que permite validar la autenticidad e integridad de las facturas electrónicas (Facturae), y a continuación incorporarlas automáticamente de forma segura a Salesforce. Los campos sensibles de la factura son cifrados en la plataforma.</p> <p>El resultado final de este trabajo es una aplicación que dota a los usuarios de un repositorio centralizado seguro para almacenar sus facturas, además de habilitar la categorización de las mismas de manera flexible. La aplicación realiza el almacenado de las facturas de manera estructurada en el CRM permitiendo obtener informes y estadísticas sobre las mismas.</p> <p>Este proyecto abre un nuevo abanico de funcionalidades a sus usuarios, facilitando que puedan crear campañas de marketing o recibir notificaciones automáticas sobre sus gastos.</p>	

**Abstract (in English, 250 words or less):**

Public administrations and companies are leaning towards the digital transformation for the impact it has on boosting their competitiveness and for environmental reasons. A clear example is electronic invoices, where the public administration deployed tools and normative legal around it. In the same fashion as classic invoicing, their digital equivalent must provide legibility, authenticity and integrity. In Spain, the standard format is Facturae.

Companies must introduce electronic invoices in their information systems, especially in the critical ones, such as the Customer Relationship Management (CRM) systems (CRM). The world's number one CRM platform is Salesforce.

Nevertheless, there is not any application that allows importing or categorising electronic invoices in the Salesforce platform. This process must be secure and guarantee the confidentiality of the data stored in the CRM.

The objective of this work was to develop a system that allows validation of authenticity and integrity of the electronic invoices (Facturae). To (after validation is successful) proceed to automatically import them, in a secure way, to Salesforce, making sure the sensitive fields are encrypted on the platform.

The result of this work is an application that provides users with a secure repository to store their invoices, and enable their categorization in a flexible way. The application stores the invoices in a structured way in the CRM, allowing users to report and gather analytics on them.

This project offers a wide range of new functionality to their users: facilitating the creation of marketing campaigns, automatic alerts to control their spending, etc.

**Palabras clave (entre 4 y 8):**

Facturae, Salesforce, Seguridad, Cifrado, Facturas, Cloud, CRM

# Índice

1. Introducción.....	1
1.1 Objetivos del Trabajo.....	2
1.2 Enfoque y método seguido.....	2
1.3 Planificación del Trabajo .....	3
1.3.1 Recursos.....	3
1.3.2 Planificación temporal.....	3
1.4 Breve resumen de productos obtenidos .....	6
1.5 Breve descripción de los otros capítulos de la memoria.....	6
2. Conceptos básicos .....	7
2.1 Factura electrónica .....	7
2.1.1 Campos Obligatorios .....	7
2.1.2 Formatos (Facturae) .....	8
2.2 Salesforce .....	8
2.2.1 Seguridad.....	9
3. Diseño .....	10
3.1 Requisitos funcionales y de seguridad .....	10
3.2 Tipo de arquitectura de la aplicación .....	10
3.3 Casos de uso .....	10
3.4 Validación de facturas .....	11
3.4.1 Validación del formato.....	12
3.4.2 Validación de la firma electrónica .....	12
3.5 Configuración del Cloud .....	13
3.5.1 Usuario administrador.....	14
3.6 Modelado de los datos .....	14
3.6.1 Factura.....	15
3.6.2 Categoría .....	17
3.6.3 Contacto.....	17
3.6.4 Operación .....	18
3.6.5 Ficheros Adjuntos .....	18
3.7 APIs Salesforce.....	18
3.8 Seguridad .....	19
3.8.1 Cloud .....	19
3.8.1.1 Claves, “Tenant Secrets”, y “Master Secrets” .....	20
3.8.2 TLS .....	20
3.8.3 Campos Sensibles .....	20
3.9 Diseño de la interfaz.....	21
3.9.1 Aplicación web .....	21
3.9.2 Salesforce .....	22
3.10 Informes y estadísticas.....	26
3.10.1 Totales por categoría .....	26
3.10.2 Totales por emisor .....	27
4. Implementación .....	28
4.1 Requisitos previos .....	28
4.1.1 Creación de cuenta Salesforce .....	28
4.1.2 Connected App .....	29

4.2 Herramientas de desarrollo .....	30
4.2.1 Eclipse .....	30
4.2.2 Maven .....	31
4.3 Validación de facturas .....	32
4.3.1 Validación del formato.....	32
4.3.2 Validación de la firma.....	33
4.4 Integración con Salesforce .....	33
4.4 Cifrado de campos sensibles .....	34
5. Pruebas .....	38
5.1 Creación de facturas .....	38
5.2 Plan de pruebas .....	38
5.2.1 Pruebas de la validación del formato Facturae .....	38
5.2.2 Pruebas de la validación de la firma electrónica .....	40
5.2.3 Pruebas de la categorización de las facturas.....	42
5.2.4 Pruebas del almacenamiento del contenido de las facturas .....	42
5.2.5 Pruebas del almacenamiento del fichero con la factura .....	44
5.2.6 Prueba del correcto cifrado de los campos.....	45
6. Conclusiones.....	48
6.1 Trabajo futuro .....	48
7. Bibliografía .....	49

## Lista de Ilustraciones

Ilustración 1: Diagrama de Gantt	5
Ilustración 2: Diagrama de Casos de Uso de la Aplicación	11
Ilustración 3: Diagrama de Casos de Uso de la Aplicación Cloud	11
Ilustración 4: Diagrama del Modelo de Datos en Salesforce	15
Ilustración 5: Interfaz de la Aplicación	22
Ilustración 6: Mensajes de Resultados en la Interfaz	22
Ilustración 7: Interfaz de la Aplicación en Salesforce	23
Ilustración 8: Pestaña Facturas en Salesforce	23
Ilustración 9: Interfaz del Objeto Factura en Salesforce	24
Ilustración 10: Interfaz de la Pestaña Related	25
Ilustración 11: Informe Totales por Categoría	26
Ilustración 12: Informe Facturas por Emisor	27
Ilustración 13: Formulario de Solicitud de una Developer Org de Salesforce	29
Ilustración 14: Creación de una Connected App	30
Ilustración 15: Permisos Necesarios para Shield	34
Ilustración 16: Generación del Tenant Secret	35
Ilustración 17: Guardar Tenant Secret	35
Ilustración 18: Cifrado de Campos Sensibles	36
Ilustración 19: Cifrado de Ficheros Adjuntos	36
Ilustración 20: Confirmación del Correcto Cifrado de un Campo	37
Ilustración 21: Generación de un Nuevo Tenant Secret	45
Ilustración 22: Confirmación de Destrucción de un Tenant Secret	46
Ilustración 23: Campo de Tipo Fecha al Destruir el Tenant Secret	46
Ilustración 24: Campo de Tipo Texto al Destruir el Tenant Secret	47
Ilustración 25: Error al Mostrar Factura al Destruir el Tenat Secret	47

## Lista de Tablas

Tabla 1: Planificación temporal	4
Tabla 2: Objeto Factura	16
Tabla 3: Objeto Categoría	17
Tabla 4: Campos nuevos objeto Contacto	18
Tabla 5: Objeto Operación	18
Tabla 6: Pruebas de la validación del formato Facturae	40
Tabla 7: Pruebas de la validación de la firma electrónica	41
Tabla 8: Pruebas de la categorización de las facturas	42
Tabla 9: Pruebas del almacenamiento del contenido de las facturas	44
Tabla 10: Pruebas del almacenamiento del fichero con la factura	45

# 1. Introducción

Hoy en día estamos viviendo la era de la transformación digital (Salesforce.com, 2019), donde se observa una clara tendencia a la digitalización de todo tipo de documentación, contratos, entradas para eventos e incluso recibos y facturas. Este proceso de digitalización nos permite tener toda nuestra información alojada en un ordenador (o en la nube), evitando así tener que ocupar varios cajones en nuestras casas o decenas de archivadores en nuestras oficinas. La digitalización ayuda positivamente en nuestro camino hacia una sociedad más sostenible, siendo un proceso ecológico que no solo nos permite eliminar los materiales y costes de impresión, sino que también supone un impacto en el medioambiente al acabar por fin con el derroche de papel.

Un ejemplo significativo de la tendencia hacia la digitalización son las facturas, donde podemos observar como la administración desde el 18 de abril de 2019 ha impuesto a los Estados Miembros de la Unión Europea el uso obligatorio de las mismas en todos los para la en todas las transacciones que realicen con las Administraciones Públicas (AAPP). O por ejemplo en España, donde ya desde el año 2015 todas las AAPP deben recibir y enviar facturas solo en formato electrónico (PHC Software, 2019).

Con la digitalización de las facturas el manejo de las mismas ha cambiado drásticamente, pero los riesgos para la seguridad de los datos que estas contienen, aunque de manera diferente, siguen aún presentes. Las facturas en formato XML, por ejemplo, se pueden modificar fácilmente con un simple editor de texto, suponiendo esto un riesgo de alteración o falsificación de los datos. La normativa de facturación en España establece la necesidad de garantizar la autenticidad del origen y la integridad del contenido de las facturas, de manera que el estándar Facturae asegura que no se puedan modificar ni emitir facturas falsas.

Las facturas contienen datos sensibles y su incorrecta manipulación puede suponer tal problema de seguridad que incluso puede llevarnos a cometer delitos de manera inconsciente compartiendo, por ejemplo, información personal de terceros sin su permiso. En la actualidad existen diversos mecanismos de cifrado que nos permiten transmitir datos de forma que no sean interceptables por cibercriminales e incluso disponemos de diversos servicios de hosting que nos permiten el almacenamiento de datos ofreciendo diversos niveles de cifrado. Si optamos por una de estas soluciones, al cifrar la factura entera perderemos operatividad sobre su contenido. La plataforma Salesforce provee de protección segura y transparente para los usuarios, permitiendo el almacenamiento de los datos en el Cloud de manera que

nos permite realizar procesamiento adicional con sus capacidades de CRM realizando cifrado sobre los campos sensibles.

Como en la actualidad no existe ninguna aplicación que nos permita categorizar las facturas digitales en la plataforma Salesforce preservando su confidencialidad, el objetivo de este trabajo es implementar un sistema que automatice esta labor de manera segura.

## 1.1 Objetivos del Trabajo

El objetivo de este trabajo es desarrollar un sistema que permita validar e incorporar de forma segura las facturas electrónicas (Facturae) en la plataforma Salesforce de manera que se preserve su confidencialidad una vez almacenadas en ella.

En el proceso se van a realizar las tareas siguientes:

- T1. Validación de la estructura y campos de la factura contra el esquema definido por Facturae.
- T2. Validación de la integridad y autenticidad de la factura mediante la firma electrónica.
- T3. Carga segura de la factura electrónica a la plataforma Salesforce (autenticación y comunicación segura).
- T4. Custodia segura de la factura en la plataforma Salesforce.
- T5. Categorización y clasificación de las facturas electrónica.
- T6. Protección de los campos sensibles de las facturas.
- T7. Obtención de información estadística de las facturas cargadas.
- T8. Desarrollar una aplicación amigable y de fácil uso.

## 1.2 Enfoque y método seguido

Este proyecto responde a la necesidad de realizar la gestión de las facturas electrónicas de manera segura y confidencial. Se pretende proporcionar al usuario una manera fácil de almacenar sus facturas en el Cloud, automatizando el proceso íntegramente (desde la validación del formato hasta el almacenamiento de las mismas) y permitiendo obtener información estadística de las facturas procesadas.

La estrategia escogida es el desarrollo de una aplicación nueva que hará uso de librerías de libre distribución para realizar las validaciones necesarias sobre las facturas y emplear la tecnología que ofrece Salesforce para dotar el sistema no solo de almacenamiento en el Cloud, sino que también de los niveles de seguridad requeridos y la capacidad de generación de informes y estadísticas.

Bajo este contexto, teniendo en cuenta la complejidad del proyecto, para su gestión hacemos uso de la metodología ágil Scrum (Schwaber and Sutherland, 2019), que ofrece un marco de trabajo por medio del cual se pueden abordar proyectos complejos y adaptativos a la vez que se entregan de manera iterativa los productos de forma progresiva y

ordenada. Este método nos da la flexibilidad necesaria a la hora de afrontar posibles imprevistos que puedan surgir durante el desarrollo de este trabajo (Kneafsey, 2019).

### 1.3 Planificación del Trabajo

En esta sección se describen los recursos utilizados y la planificación del proyecto.

#### 1.3.1 Recursos

Para la realización de este proyecto se utilizarán los siguientes recursos:

##### Hardware

- Dell Latitude 7390 Core i7 8650U 8GB 256GB SSD 13.3 Inch Laptop

##### Software

- Ubuntu 18.04.2 LTS
- Eclipse Oxygen
- Java 1.8
- La plataforma Salesforce como servidor de Cloud mediante una cuenta Developer Edition gratuita
  - La aplicación hará uso también de las diversas API de integración con Salesforce
  - Cifrado de plataforma Shield
- Google Docs y Microsoft Office: Editores de texto elegidos para la elaboración de la memoria y documentación del proyecto.

#### 1.3.2 Planificación temporal

El desarrollo de este proyecto sigue la planificación marcada por las fechas claves de las entregas parciales y tendrá una duración total que va desde el 30 de septiembre de 2019 al 7 de enero del 2020, siendo esta la fecha cuando se realizará la entrega final.

A continuación, se especifican las tareas y tiempo estimado que se destinará a cada una de ellas:

<b>Descripción</b>	<b>Inicio</b>	<b>Fin</b>	<b>Dedicación (Horas)</b>
<b>Fase Inicial</b>	<b>30/09/2019</b>	<b>20/10/2019</b>	<b>30</b>
Investigación	30/09/2019	09/10/2019	10
Planificación de Trabajo	10/10/2019	10/10/2019	2
Diseño	11/10/2019	17/10/2019	10
Juego de Pruebas Facturae	18/10/2019	19/10/2019	2
Documentación	19/10/2019	20/10/2019	6
<b>Entrega 1</b>	<b>21/10/2019</b>	<b>11/11/2019</b>	<b>66</b>
Validación Formato	21/10/2019	23/10/2019	10
Autenticación Salesforce	24/10/2019	24/10/2019	5
Modelado Datos	25/10/2019	26/10/2019	8
Integración Salesforce	27/10/2019	08/11/2019	25
Documentación	08/11/2019	12/11/2019	18
<b>Entrega 2</b>	<b>12/11/2019</b>	<b>07/12/2019</b>	<b>44</b>
Validación Firma/Integridad	12/11/2019	20/11/2019	12
Cifrado de campos	21/11/2019	26/11/2019	10
Gestión de Claves	27/11/2019	01/12/2019	2
Documentación	02/12/2019	07/12/2019	20
<b>Entrega 3</b>	<b>25/11/2019</b>	<b>23/12/2019</b>	<b>35</b>
Interfaz Gráfica	07/12/2019	15/12/2019	15
Creación de Informes	15/12/2019	17/12/2019	6
Trabajo Futuro	17/12/2019	17/12/2019	4
Documentación	18/12/2019	23/12/2019	10
<b>Entrega Final</b>	<b>02/01/2020</b>	<b>06/01/2020</b>	<b>24</b>
Fase Pruebas Final	23/12/2019	03/01/2020	8
Documentación	03/01/2020	04/01/2020	10
Presentación y Video	05/01/2020	06/01/2020	6

Tabla 1: Planificación temporal

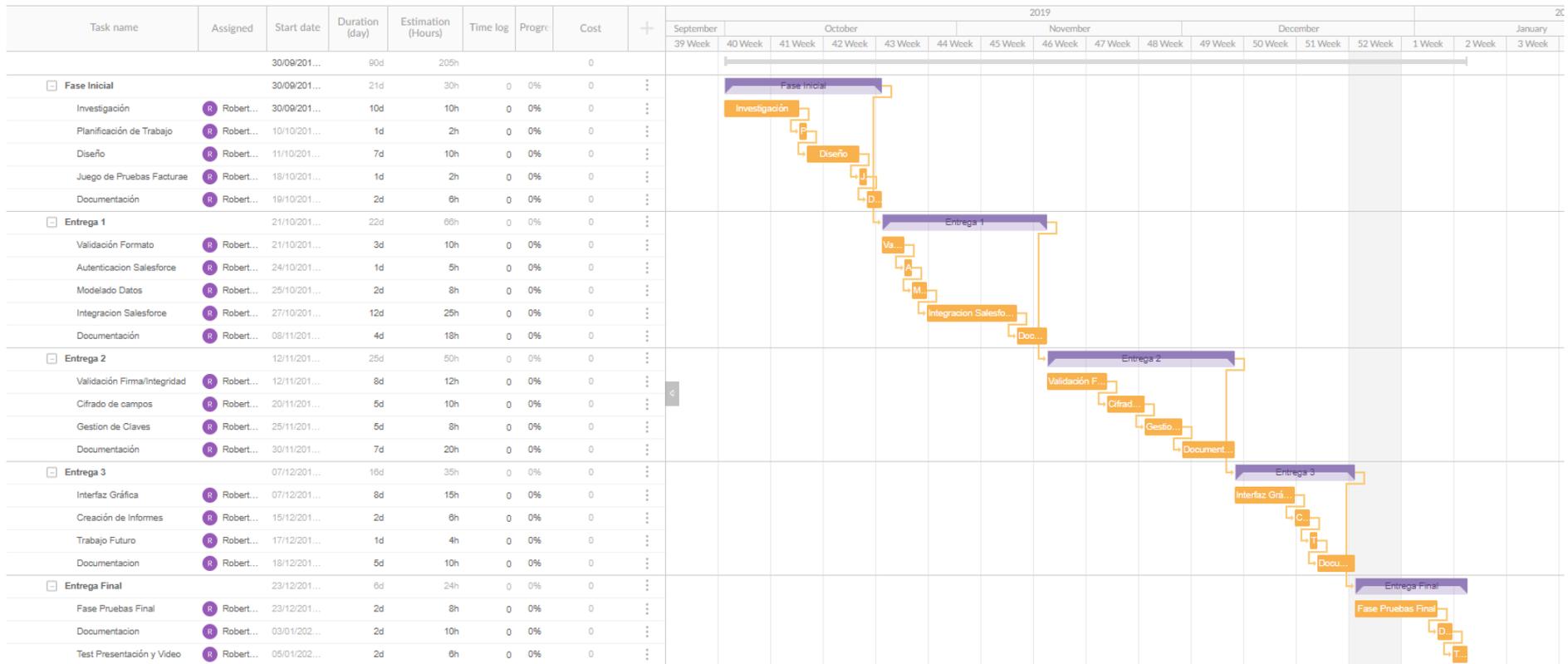


Ilustración 1: Diagrama de Gantt

#### 1.4 Breve resumen de productos obtenidos

Como resultado de este proyecto obtendremos los siguientes productos:

- Aplicación “Facturae a Salesforce”: aplicación compilada (archivo WAR), así como el código fuente utilizado para la implementación
- Presentación del proyecto en formato PowerPoint y vídeo explicativo
- Memoria del Trabajo Final de Máster.

#### 1.5 Breve descripción de los otros capítulos de la memoria

En la Sección 2 se introducen las tecnologías y herramientas necesarias para el desarrollo de este trabajo de fin de máster. Los aspectos de la arquitectura y diseño de la solución se describen en la Sección 3.

La Sección 4 señala los aspectos más relevantes de la fase de desarrollo de la aplicación, para a continuación, en la Sección 5, mostrar el proceso seguido para conseguir el juego de pruebas y la ejecución del mismo.

Finalmente, las conclusiones y líneas de trabajo futuro se presentan en la Sección 6, concluyendo la memoria con la Sección 7, donde se incluye la bibliografía con las referencias utilizadas en este proyecto.

## 2. Conceptos básicos

A continuación, se introducen brevemente las tecnologías y herramientas necesarias para el desarrollo de este trabajo de fin de master. El objetivo de este punto es dar una visión general sobre las mismas para entender la solución propuesta.

### 2.1 Factura electrónica

Una factura electrónica es un documento digital de índole fiscal, que tiene su origen en las legislaciones Latinoamericanas que surgieron entre los años 2000 a 2005 como esfuerzo global a la transparencia fiscal y al soporte a la lucha contra la evasión de impuestos (Barreix and Zambrano, 2018). Actualmente la factura electrónica es empleada de forma mandatoria u optativa en distintos países alrededor del mundo.

La factura electrónica cuenta con al menos dos elementos básicos:

- Mensaje de datos basado en estándar universal, abierto, no propietario: XML, (eXtensible Markup Language).
- Uso de firmas electrónicas basadas en Infraestructura de Clave Pública.

En los sistemas fiscales digitales más maduros, un tercer elemento caracteriza también a la factura electrónica:

- Certificación: Consistente en la validación de la sintaxis y el certificado digital del emisor realizado por la administración tributaria o un Tercero en Confianza, para garantizar su coherencia con el estándar definido por la autoridad fiscal correspondiente y la validez de la firma electrónica del emisor. Cuando ambas validaciones son exitosas, se adiciona al documento un sello digital que certifica la validez de dicha factura y otorga efectos fiscales a la misma a partir de ese momento.

#### 2.1.1 Campos Obligatorios

Tal y como recoge el artículo 6 del RD 1496/2003 que regula el contenido de una factura, para que una factura electrónica tenga la misma validez legal que una emitida en papel, la factura electrónica deberá contener los siguientes campos obligatorios:

- Número de factura, y si es necesario, serie.
- Fecha de expedición.
- Nombre y apellidos, o razón social, del emisor y del receptor.
- NIF del emisor y del receptor.
- Domicilio fiscal de los dos tributadores: del emisor y del receptor.
- Descripción de las operaciones (base imponible).

- Fecha de la entrega del bien o de la prestación del servicio (si es distinta a la de expedición).
- Referencia, de manera inequívoca, al albarán, si lo hay.
- Tipo impositivo aplicable a la operación.
- Cuota tributaria o repercutida.

El documento electrónico que la representa, además de contener estos campos, deberá estar firmada mediante una firma electrónica avanzada basada en certificado reconocido, y ser transmitida de un ordenador a otro, recogiendo el consentimiento de ambas partes.

### 2.1.2 Formatos (Facturae)

No existen requisitos formales respecto a la forma en que se debe proceder a la codificación de la factura, pero las modalidades más habituales son las siguientes:

- PDF: Cuando el destinatario es un particular, un profesional o una PYME cuyo único interés sea guardar electrónicamente la factura, pero no evitar volver a teclear los datos ya que con este formato no se facilita el ingreso de los datos de la factura en el ordenador de destino.
- XML: Cuando el envío es de ordenador a ordenador, puede también utilizarse este tipo de sintaxis. Existen diversas variantes cuya convergencia se espera en el marco de las Naciones Unidas. Las más importantes son UBL respaldado por OASIS y GS1 respaldado por la organización del mismo nombre.

En España la variante Facturae (procedente de CCI-AEAT), respaldada por el Centro de Cooperación Interbancaria, la Agencia Tributaria y el Ministerio de Industria, Turismo y Comercio es la más difundida, y cuenta con sistemas de traducción a y desde UBL.

### 2.2 Salesforce

Salesforce es una plataforma de gestión de las relaciones con los clientes (CRM) basada en la nube que proporciona a todos los departamentos de su organización, incluidos los de marketing, ventas, servicio al cliente y comercio electrónico, una visión unificada de sus clientes en una plataforma integrada.

El CRM o la gestión de las relaciones con los clientes es una estrategia para gestionar las relaciones e interacciones de una organización con los clientes y clientes potenciales. Un sistema de CRM ayuda a las empresas a permanecer conectadas con sus clientes, agilizar los procesos y mejorar la rentabilidad.

Salesforce ha puesto mucho esfuerzo en construir aplicaciones que pueden ser configuradas al estilo point-and-click, es decir, sin necesidad de escribir código. Sin embargo, las empresas casi siempre necesitan personalizar la plataforma para ajustarla a los requisitos específicos de cada una y es ahí donde los programadores entran en escena.

De toda la variedad de productos y servicios que ofrece la compañía, el que nos interesa en este trabajo es la plataforma Lightning y sus herramientas destinadas a la seguridad de los datos.

### 2.2.1 Seguridad

Las funciones de seguridad que ofrece Salesforce ayudan a dotar a sus usuarios con las herramientas necesarias para que realicen su trabajo de forma segura y efectiva. Salesforce limita la exposición de los datos a los usuarios que realizan acciones sobre ellos, permitiendo implementar los controles de seguridad que se consideren oportunos para asegurar la confidencialidad de los datos.

Salesforce utiliza parte de la tecnología más avanzada de seguridad de Internet disponible hoy día, desde el simple acceso a su interfaz web, donde la tecnología Transport Layer Security (TLS) protege la información que se intercambia con el servicio, ofreciendo también autenticación y cifrado clásico, garantizando que sus datos están seguros y únicamente disponible para usuarios registrados en su organización.

Este proveedor Cloud nos ofrece herramientas avanzadas como Salesforce Shield, un trío de herramientas de seguridad que los administradores y desarrolladores pueden utilizar para crear un nuevo nivel de confianza, transparencia, cumplimiento normativo y leyes en aplicaciones importantes para los negocios. Incluye Cifrado de plataforma, Supervisión de eventos y Seguimiento de auditoría de campos.

El cifrado de plataforma permite cifrar los datos más confidenciales de forma nativa durante el periodo de inactividad en todas sus aplicaciones de Salesforce. Esto ayuda a proteger información personal identificable (PII), datos confidenciales o patentados y cumplir las políticas de datos internos y externos mientras mantiene funciones importantes, como búsqueda, flujo de trabajo y reglas de validación. Mantiene un control completo sobre claves de cifrado y puede establecer permisos de datos cifrados para proteger datos confidenciales de usuarios no autorizados.

Toda esta funcionalidad convierte a la plataforma de Salesforce en una de las soluciones Cloud más completas del mercado.

## 3. Diseño

En esta sección se describen los aspectos relevantes al diseño y la arquitectura de la solución propuesta.

### 3.1 Requisitos funcionales y de seguridad

- La aplicación validará el esquema XML de la factura.
- La aplicación validará la firma electrónica de la factura.
- El usuario podrá categorizar las facturas.
- El usuario podrá definir las categorías de las facturas
- La aplicación permitirá configurar las credenciales de autenticación necesarios para el procesamiento en el Cloud
- La aplicación almacenará la factura en el Cloud
- Sólo se almacenarán facturas correctas
- El usuario podrá obtener informes y estadísticas de los totales de las facturas procesadas

### 3.2 Tipo de arquitectura de la aplicación

El proyecto se divide en dos partes claramente definidas. La primera es una aplicación web encargada del procesado de las facturas y la segunda es la parte Cloud encargada de almacenar los datos de las facturas, así como los relacionados con su categorización, a los cuales el usuario tendrá acceso cuando los necesite.

Para la primera parte, como consecuencia de las necesidades de seguridad de este proyecto, utilizaremos Java como lenguaje de programación. Java es hoy por hoy uno de los lenguajes más usados por la banca (Addicott, 2019) para el desarrollo de aplicaciones seguras debido a su robustez en lo que rodea a la funcionalidad que ofrece, como por ejemplo cifrado, autenticación avanzada y control de acceso para dotar de los requerimientos de seguridad necesarios hoy en día en cualquier aplicación web segura. Java proporciona también la funcionalidad necesaria para realizar la integración con entornos Cloud como Salesforce, el escogido para la segunda parte de este proyecto.

Las conexiones entre cada una de las partes de la aplicación se realizarán mediante las peticiones que ofrece la tecnología REST (Representational State Transfer), que posibilita el paso de cada dato del usuario entre la aplicación de escritorio y el Cloud sin problemas.

### 3.3 Casos de uso

A continuación, se muestran los casos de uso de la aplicación encargada del procesado de las facturas. Estos casos de uso se describen en los siguientes apartados.

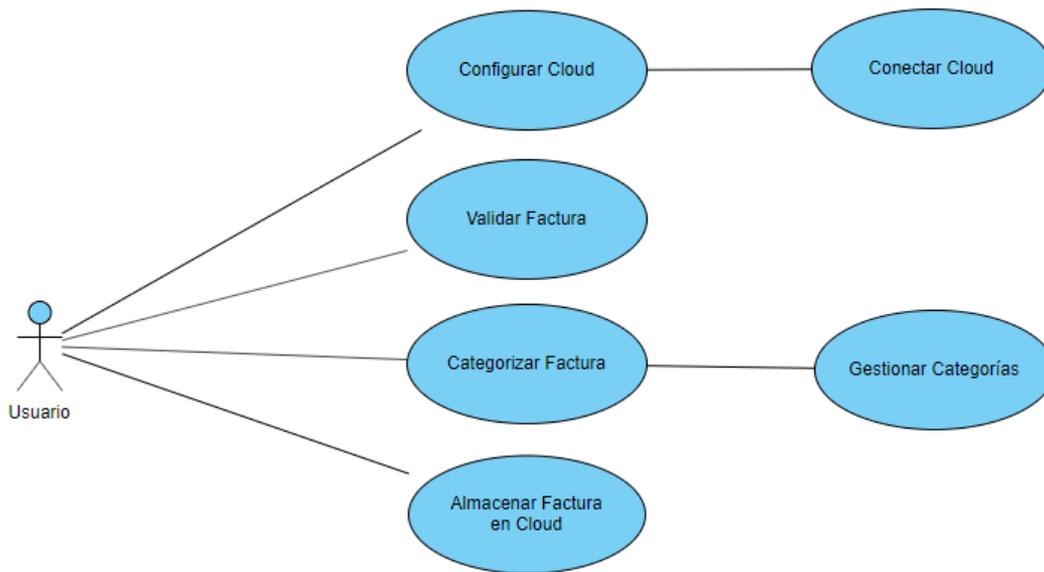


Ilustración 2: Diagrama de Casos de Uso de la Aplicación

En la Ilustración 3 se muestran los casos de uso de la aplicación Cloud.

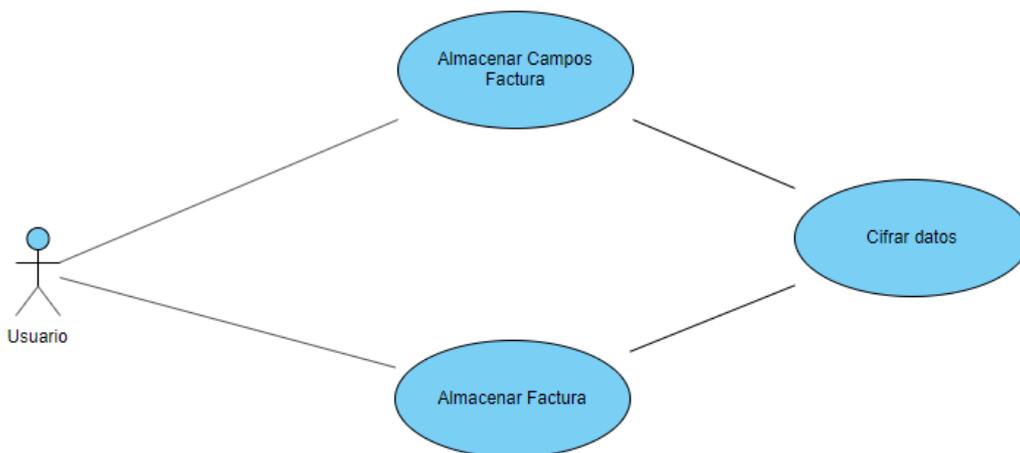


Ilustración 3: Diagrama de Casos de Uso de la Aplicación Cloud

### 3.4 Validación de facturas

Para la realización de este proyecto nos enfocaremos en los dos primeros elementos mencionados en anterioridad (capítulo 2.1), realizando la validación del documento XML comprobando que se ajuste al esquema XML definido para la versión 3.2.2 y validando la firma electrónica.

Para el desarrollo de la fase de validación será necesaria la creación de un conjunto de facturas electrónicas. Para ello es necesario descargar e instalar la aplicación Facturae (Facturae.gob.es, 2019). Con esta aplicación se realizará la creación de las facturas necesarias para la

ejecución de los casos de prueba necesarios para implementar la validación.

### 3.4.1 Validación del formato

La validación del formato de las facturas electrónicas consiste en comprobar que la estructura XML se ajusta al esquema definido para las facturas con el formato Facturae 3.2.1. Para ello una vez que se especifique el fichero con la factura a procesar se comprobará uno a uno cada uno de los elementos presentes, validándolo contra el esquema XSD definido por el standard de Facturae (Facturae.gob.es, 2019).

Durante la fase de validación, en el supuesto de que un campo no cumpla el formato indicado por el estándar o de que falte un campo obligatorio, la ejecución del procesado se detendrá mostrando el error de validación específico.

### 3.4.2 Validación de la firma electrónica

El formato de firma que define Facturae es XMLDSig ENVELOPED con extensiones XADES-EPES. Se trata de una firma XML avanzada que incluye datos sobre la política a la que debe ajustarse y que define las características específicas que deben cumplir. La política de la Facturae, para las versiones 3.1 y 3.2, se puede consultar en la web de Facturae (Facturae.gob.es, 2019).

El proceso de validación de la firma electrónica consta de las siguientes fases:

1. Localizar el elemento "Signature" donde se contienen no solo la firma si no los parámetros que la definen. De no existir este nodo se entiende que la firma no está firmada.

```
...  
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="Signature-998668816">
```

2. La firma es correcta, es decir, no se han modificado los datos firmados con posterioridad a la firma.

```
<ds:SignedInfo>  
<ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>  
<ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>  
<ds:Reference Id="Ref1" URI="">  
<ds:Transforms>
```

```

<ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature"/>
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>rKxz...</ds:DigestValue>
</ds:Reference>
<ds:Reference URI="#Certificatel">
<ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>pLvS...</ds:DigestValue>
</ds:Reference>
<ds:Reference Type="http://uri.etsi.org/01903#SignedProperties"
URI="#SignedProperties-292532795">
<ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>6tQA...</ds:DigestValue>
</ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>PVoD...</ds:SignatureValue>

```

3. La firma se ajusta a la política de Facturae, es decir que se trata de una firma XADES-EPES ENVELOPED, y se ajusta a una de las siguientes versiones del esquema XML de las firmas XADES: 1.2.2 o 1.3.2. El nodo KeyInfo contiene, al menos, el certificado empleado en la firma codificado en base64 como se puede ver en el código de ejemplo de continuación:

```

<ds:KeyInfo Id="Certificatel">
<ds:KeyValue>
<ds:RSAKeyValue>
<ds:Modulus>We0Z...</ds:Modulus>
<ds:Exponent>AQAB</ds:Exponent>
</ds:RSAKeyValue>
</ds:KeyValue>
<ds:X509Data>
<ds:X509Certificate>MIIC...</ds:X509Certificate>
</ds:X509Data>
</ds:KeyInfo>

```

Si tanto en la validación de la firma como en la comprobación de integridad se produce algún error, supondrá la interrupción de la ejecución mostrando el error correspondiente por pantalla.

### 3.5 Configuración del Cloud

Este módulo permitirá al usuario proporcionar las credenciales necesarias para realizar la autenticación necesaria con la cuenta de Salesforce. Por comodidad y facilidad de uso estos credenciales y parámetros de autenticación estarán codificados en el módulo de manera que solo se tengan que proporcionar una vez para poder realizar consecuentes procesados de las facturas.

### 3.5.1 Usuario administrador

Salesforce proporciona al administrador de la cuenta la posibilidad de crear usuarios con diversas jerarquías de roles para establecer, por ejemplo, diferentes niveles de acceso.

Puesto que el objetivo de este trabajo no se centra en la administración de usuarios, la aplicación encargada del procesado de las facturas no dispondrá de ningún tipo de autenticación, dejando este aspecto a la definición correcta de credenciales correctos con nivel de administrador para la cuenta definida por el usuario.

### 3.6 Modelado de los datos

A la hora de procesar los campos presentes en las facturas electrónicas, el alcance de este proyecto se limitará a los campos obligatorios definidos por el formato Facturae. No se realizará el procesado de campos opcionales al no suponer un reto tecnológico mayor que la dedicación de tiempo adicional al correcto mapeado de los mismos, por lo tanto, estos campos presentes en las facturas se ignorarán. Tampoco se realizará el procesado de múltiples facturas en lote, por lo tanto, si al procesar el XML hay más de una factura, se procesará tan solo la primera factura especificada.

Para la integración de la factura en una plataforma en la nube se ha escogido Salesforce debido a su posición dominante en el mercado (Salesforce.com, 2019) no sólo en el ámbito del CRM si no en el de las aplicaciones Enterprise de Cloud. Prueba de ello son las noticias de crecimiento, donde se prevé que el ecosistema creado alrededor de la empresa generará 48.000 nuevos empleos directos y 60.900 puestos indirectos en España entre 2019 y 2024, con un impacto económico de 17.000 millones (Expansión.com, 2019).

La plataforma Salesforce nos permite representar y almacenar nuestra información mediante una serie de objetos estándar (Cuenta, Contacto, Caso, etc.) y también nos da la flexibilidad de crear nuestros objetos personalizados. Para el objetivo de la aplicación a desarrollar hacemos uso de ambos para realizar la integración de los campos Facturae.

En la Ilustración 4 se muestra el modelo de datos que se definirá en Salesforce.

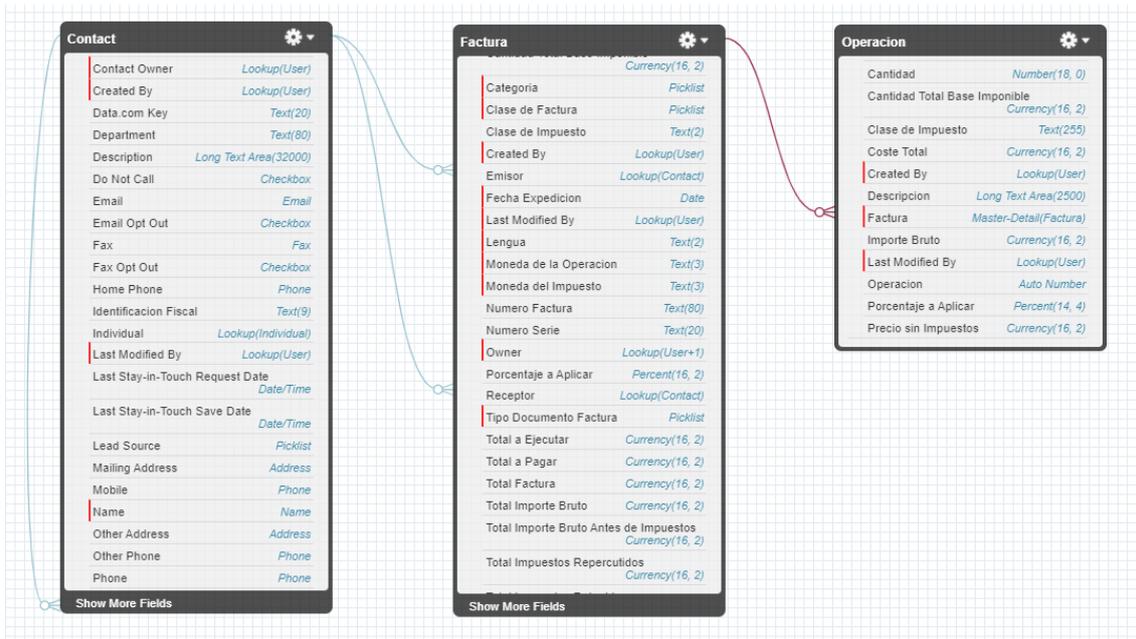


Ilustración 4: Diagrama del Modelo de Datos en Salesforce

A continuación, se muestra en detalle cada una de las entidades involucradas en el modelo.

### 3.6.1 Factura

Puesto que la plataforma no dispone de ningún objeto estándar con características similares a las necesarias para su representación, procederemos a la creación de un objeto personalizado llamado “Factura” con las siguientes características:

<b>Campo en Facturae</b>	<b>Nombre en Salesforce</b>	<b>Tipo</b>
-	Categoría	Picklist
TaxableTotalAmount	Cantidad Total Base Imponible	Currency(16, 2)
InvoiceClassType	Clase de Factura	Picklist
TaxTypeCode	Clase de Impuesto	Text(2)
SellerParty	Emisor	Lookup(Contact)
IssueDate	Fecha Expedición	Date
LanguageName	Lengua	Text(2)
InvoiceCurrencyCode	Moneda de la Operación	Text(3)
TaxCurrencyCode	Moneda del Impuesto	Text(3)
InvoiceNumber	Numero Factura	Text(80)
InvoiceSeriesCode	Numero Serie	Text(20)
TaxRate	Porcentaje a Aplicar	Percent(16, 2)
BuyerParty	Receptor	Lookup(Contact)
InvoiceDocumentTypeType	Tipo Documento Factura	Picklist
TotalExecutableAmount	Total a Ejecutar	Currency(16, 2)
TotalOutstandingAmount	Total a Pagar	Currency(16, 2)
InvoiceTotal	Total Factura	Currency(16, 2)
TotalGrossAmount	Total Importe Bruto	Currency(16, 2)
TotalTaxOutputs	Total Impuestos Repercutidos	Currency(16, 2)
TotalTaxesWithheld	Total Impuestos Retenidos	Currency(16, 2)
TotalGrossAmountBefore Taxes	Total Importe Bruto Antes de Impuestos	Currency(16, 2)

*Tabla 2: Objeto Factura*

### 3.6.2 Categoría

Para la representación de cada una de las categorías necesarias para la clasificación de las facturas nos decantamos por crear un objeto personalizado con los siguientes campos:

Campo en Facture	Nombre en Salesforce	Tipo
-	Categoría Name	Text(80)

*Tabla 3: Objeto Categoría*

### 3.6.3 Contacto

Para representar los datos del Emisor (vendedor) y Receptor (comprador) tenemos dos opciones, o bien representarlos como campos adicionales del objeto factura o como objetos distintos. La primera opción simplifica el diseño del modelado, pero no es óptima, debido a que al fin y al cabo los campos son los mismos para ambos sujetos, por lo tanto, optamos por representar a ambos como una entidad diferente.

Para representar ambas entidades hacemos uso del objeto estándar Contacto puesto que ya contiene la mayoría de los campos que necesitamos almacenar y además contiene otros campos que nos podrían servir para trabajo futuro si necesitáramos almacenar campos adicionales. Para las necesidades de este proyecto, creamos los siguientes campos en el objeto Contacto:

Campo en Facturae	Nombre en Salesforce	Tipo
TaxIdentificationNumber	Identificacion Fiscal	Text(9)
PersonTypeCodeType	Tipo de Persona	Picklist
ResidenceTypeCode	Tipo de Residencia	Picklist
Name / CorporateName	Name	Name
FirstSurname / TradeName		
Address	Direccion	Address
PostCodeAndTown / PostCode		
Province		
CountryCode		

Tabla 4: Campos nuevos objeto Contacto

### 3.6.4 Operación

Para la representación de cada una de las operaciones (bien entregado o servicio prestado) registradas en la factura podríamos optar por hacer uso del objeto Producto. Sin embargo, debido a que los campos que este objeto estándar contiene no se adaptan a los campos necesarios en el modelo de datos (y que, además, conceptualmente, una operación no solo se puede tratar de un bien, sino de un servicio) nos decantamos por crear un objeto personalizado con los siguientes campos:

Campo en Facture	Nombre en Salesforce	Tipo
Quantity	Cantidad	Number(18, 0)
TotalAmount	Cantidad Total Base Imponible	Currency(16, 2)
TaxTypeCode	Clase de Impuesto	Text(255)
TotalCost	Coste Total	Currency(16, 2)
ItemDescription	Descripcion	Long Text Area(2500)
-	Factura	Master-Detail(Factura)
GrossAmount	Importe Bruto	Currency(16, 2)
TaxRate	Porcentaje a Aplicar	Percent(14, 4)
UnitPriceWithoutTax	Precio sin Impuestos	Currency(16, 2)

Tabla 5: Objeto Operación

### 3.6.5 Ficheros Adjuntos

Tanto los objetos estándar como los objetos personalizados disponen de una relación con el objeto Adjunto que permite almacenar ficheros y asociarlos a estos. Hacemos uso de esta funcionalidad activándola en el objeto personalizado Factura, de forma que nos permitirá almacenar la factura íntegra después de procesar sus campos.

### 3.7 APIs Salesforce

Las APIs que ofrece Salesforce para la manipulación de datos son las siguientes: la API de REST, la API de SOAP, la API masiva y la API de transmisión. Su finalidad es permitirle manipular sus datos de Salesforce, mientras que otras API le permiten realizar acciones, como personalizar formatos de página o crear herramientas de desarrollo personalizadas. Además, puede usar otras API de Salesforce para manipular subconjuntos de sus datos de Salesforce. Por ejemplo, la API

de REST de Analytics se centra en operaciones relevantes para Analytics. No obstante, estas cuatro API se pueden usar para la amplia variedad de datos relevantes de Salesforce.

La API de REST proporciona una interfaz de servicios web basada en REST eficiente, práctica y sencilla para interactuar con Salesforce. Algunas de sus ventajas son la facilidad de la integración y el desarrollo, y además es una excelente elección de tecnología para su uso con aplicaciones móviles y proyectos web. Para ciertos proyectos, es posible que desee utilizar la API de REST con otras API de REST de Salesforce.

Nos decantamos por la API de REST para la autenticación, creación y manipulación de nuestro modelado de datos en Salesforce.

### 3.8 Seguridad

La seguridad es un aspecto fundamental de esta aplicación, por lo tanto, se tendrá en cuenta desde la fase de diseño, analizando las posibles vulnerabilidades a las que se puede ver afectada una aplicación con estas características.

#### 3.8.1 Cloud

Salesforce ofrece dos formas de cifrar los datos almacenados. El cifrado Classic (incluido en el precio base de la licencia) permite definir un campo como un tipo texto especial para aquellos que queramos cifrar. Este campo customizado se protege con claves del estándar 128-bit AES (Advanced Encryption Standard).

El cifrado de plataforma Shield (disponible de manera gratuita en las Developer Edition) supone un coste adicional al requerir de una licencia específica. Este mecanismo nos permite cifrar todo tipo de datos confidenciales y sensibles “At rest” en la plataforma Salesforce. “At rest” significa cualquier dato inactivo o almacenado en archivos, hojas de cálculo, campos estándar o personalizados, e incluso bases de datos data warehouses. Con Shield, el cifrado de los datos es más robusto con una clave 256-bit AES, permitiendo a sus usuarios la gestión a un mayor rango de claves y permisos y la búsqueda de datos cifrados en la base de datos.

Nos decantamos por el uso del cifrado de plataforma Shield puesto que provee a los usuarios de una ventaja al permitir demostrar la conformidad con las regulaciones y requerimientos de la industria al mostrar que mediante su uso se cumplen obligaciones contractuales de almacenamiento de datos privados en el Cloud (Aepd.es, 2019).

### 3.8.1.1 Claves, “Tenant Secrets”, y “Master Secrets”

Salesforce genera un nuevo Master Secret tres veces al año (con cada actualización de la aplicación). Este Master Secret se crea usando un procesador con el nombre “Hardware Security Module” (HSM) especialmente creado para realizar esta tarea y crear claves de cifrado robustas y seguras. El HSM, que se encuentra en una memoria USB y se encuentra almacenado en una caja de seguridad en un banco. Durante el lanzamiento de una nueva versión, solo oficiales de la seguridad designados por Salesforce pueden acceder al HSM almacenado en la caja de seguridad del banco. Este oficial usa el HSM para generar un nuevo Master Secret, que se almacena en un área segura de los servidores de Salesforce.

Cuando un usuario crea un Tenant Secret bajo demanda en Salesforce, este se almacenará de manera segura en una base de datos de Salesforce. El conjunto del Tenant Secret con el Master Secret es lo que se llama proceso de derivación de clave para crear las claves que cifran y descifran los datos almacenados en la plataforma. El Master Secret solo se actualiza una vez con cada actualización de la aplicación, pero el Tenant Secret se puede actualizar tan a menudo como se quiera.

### 3.8.2 TLS

La aplicación hará uso del Cloud para el almacenamiento de los datos de las facturas y las facturas en sí, por lo tanto, el método de conexión debe implementar un mecanismo seguro de modo que no se puedan interceptar el contenido de las comunicaciones.

Uno de los estándares de seguridad de comunicaciones de este tipo es el protocolo TLS, que se distingue por la seguridad con la interoperabilidad (las transmisiones de datos encriptados de diferentes aplicaciones como HTTP, que pasa a ser HTTPS). La comunicación con este protocolo se lleva a cabo mediante un canal seguro y cifrado entre cliente y servidor en donde se negocia la criptografía del mensaje, se autentican las claves del cifrado y se realiza una transmisión segura.

Para ello hacemos uso del protocolo TLS en su versión 1.2, puesto que versiones anteriores se encuentran en proceso de decomisión por la mayoría de dominios y proveedores de servicios ya que las versiones anteriores no son consideradas seguras (The Daily Swig | Cybersecurity news and views, 2019).

### 3.8.3 Campos Sensibles

Puesto que no todos los datos presentes en las facturas electrónicas son sensibles, nos centraremos en la información que realmente requiera de

cifrado para cumplir requisitos de privacidad. El cifrado innecesario de datos puede afectar el rendimiento del sistema al suponer un procesado adicional.

A continuación, se listan los campos que contienen información sensible, y que por lo tanto requerirán de un tratamiento especial puesto que será necesario su cifrado a la hora de almacenarlos:

Objeto Factura:

- Emisor
- Fecha Expedición
- Receptor

Objeto Contact:

- Identificación Fiscal
- Name
- Address

Objeto Operación:

- Descripción

Ficheros Adjuntos:

- El contenido íntegro las facturas

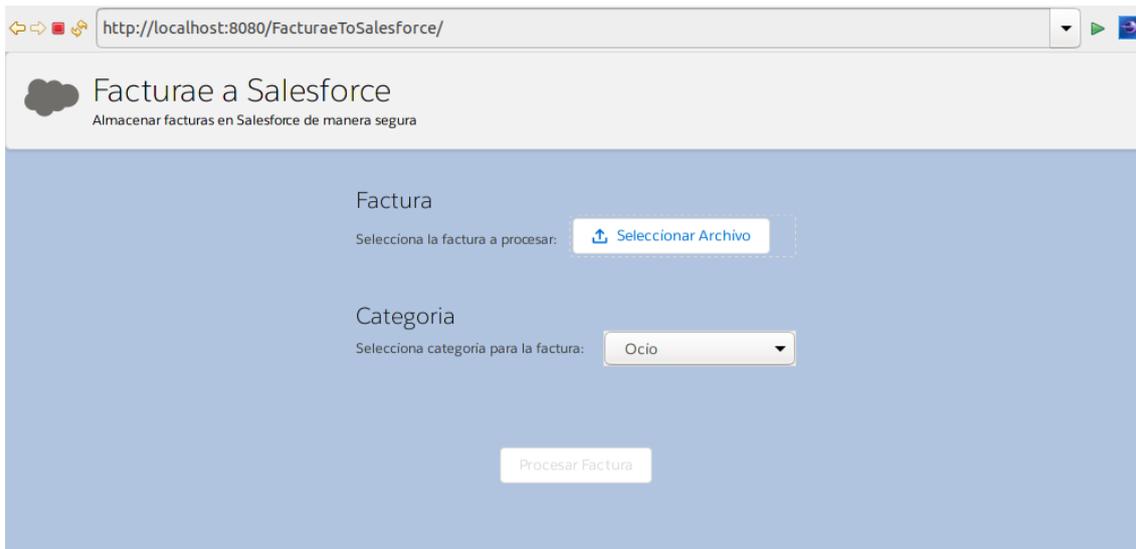
### 3.9 Diseño de la interfaz

En las siguientes secciones se describe el diseño utilizado tanto para la interfaz de la aplicación web como la interfaz del lado de Salesforce.

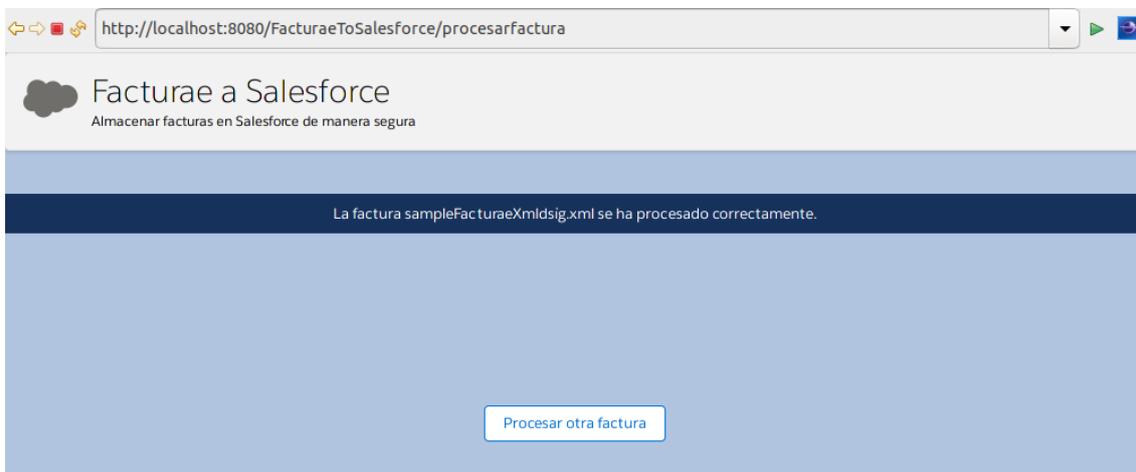
#### 3.9.1 Aplicación web

La interfaz de la aplicación web no requiere nada más que la selección de la factura a procesar y de una categoría, por lo tanto, se ha optado por la realización de una interfaz simple que permita al usuario la ejecución de estas selecciones de manera intuitiva y que muestre el resultado de la ejecución de manera sencilla y amigable.

Para la parte de estilos, fuentes e iconos utilizamos el Lightning Design System, puesto que nos permite construir experiencias de usuario ricas proporcionando patrones y estándares de diseño nativos de Salesforce, permitiendo que el diseño muy parecido a la interfaz de Salesforce (Lightningdesignsystem.com, 2019).



*Ilustración 5: Interfaz de la Aplicación*

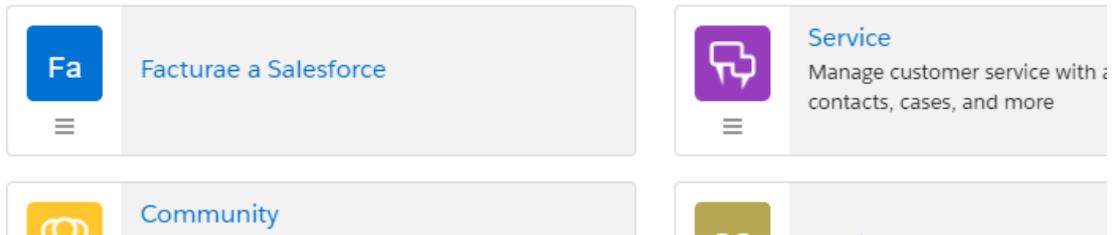
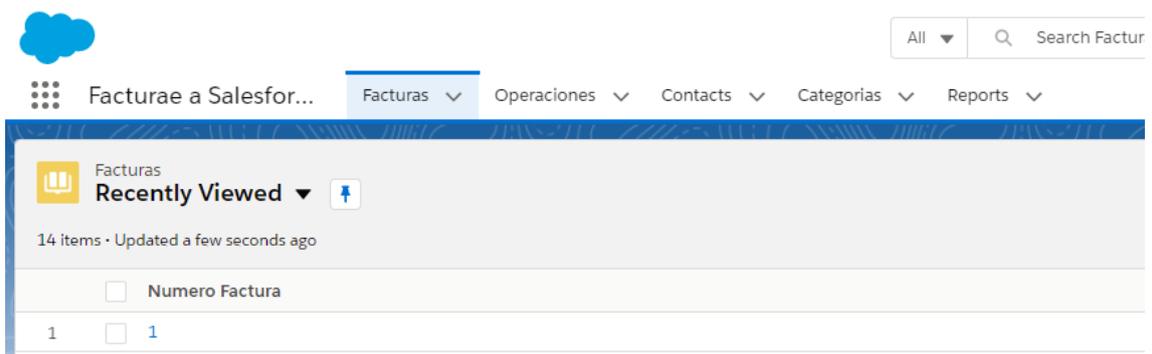


*Ilustración 6: Mensajes de Resultados en la Interfaz*

### 3.9.2 Salesforce

Para la parte de interfaz de Salesforce creamos una aplicación para agrupar los objetos mencionados en la sección de modelado de datos. Esto nos permitirá acceder directamente y de manera más cómoda a los datos creados por las facturas procesadas mediante la solución propuesta:

▼ All Apps

*Ilustración 7: Interfaz de la Aplicación en Salesforce**Ilustración 8: Pestaña Facturas en Salesforce*

Para el diseño del objeto Factura agrupamos los campos en 3 secciones diferentes. La primera con los campos identificativos tales como Número Factura, Clase de Factura, Categoría y Fecha de Expedición. La segunda sección contiene los sujetos involucrados en la transacción. La tercera categoría agrupa todos los campos relacionados con los impuestos, y la cuarta y última contendrá los campos relacionados con los totales de la factura.

U

Factura

## 18

Related
Details

Numero Factura <b>18</b>	Fecha Expedición <b>10/03/2010</b>
Numero Serie	Moneda de la Operación <b>EUR</b>
Tipo Documento Factura <b>FC</b>	Moneda del Impuesto <b>EUR</b>
Clase de Factura <b>OO</b>	Lengua <b>es</b>
Categoría <a href="#">Seguros y Finanzas</a>	

▼ **Sujetos**

Emisor <a href="#">Company Comp SA Comp</a>	Receptor <a href="#">Juana Maurino</a>
--	---

▼ **Impuestos Repercutidos**

Clase de Impuesto <b>01</b>	
Porcentaje a Aplicar <b>16.00%</b>	
Cantidad Total Base Imponible <b>€26.00</b>	

▼ **Totales**

Total Importe Bruto <b>€59.75</b>	Total Factura <b>€63.13</b>
Total Importe Bruto Antes de Impuestos <b>€59.75</b>	Total a Pagar <b>€63.13</b>
Total Impuestos Repercutidos <b>€4.42</b>	Total a Ejecutar <b>€63.13</b>
Total Impuestos Retenidos <b>€1.04</b>	

Ilustración 9: Interfaz del Objeto Factura en Salesforce

En la pestaña Related se encuentra el listado de operaciones de la factura y la factura íntegra en la sección Notes & Attachments tal y como se muestra en la Ilustración 10.

The screenshot shows the 'Related' tab of a Salesforce record for 'Factura 18'. The interface includes a navigation bar with 'Facturas', 'Operaciones', 'Contacts', 'Categorías', and 'Reports'. Below the header, there are two main sections: 'Operaciones (2)' and 'Notes & Attachments (1)'. The 'Operaciones' section contains a table with columns for 'DESCRIPCION', 'PORCENTAJE A APLIC...', 'CANTIDAD TOTAL BA...', and 'COSTE TOTAL'. The 'Notes & Attachments' section shows a file named 'ejemplo2.xml' uploaded on 22-Dec-2019.

DESCRIPCION	PORCENTAJE A APLIC...	CANTIDAD TOTAL BA...	COSTE TOTAL
Flores	0.0000%	€0.00	€25.00
Mate	16.0000%	€26.00	€26.00

Ilustración 10: Interfaz de la Pestaña Related

### 3.10 Informes y estadísticas

Con esta solución se pretende que el usuario disponga de una serie de informes y estadísticas que le permita hacer un seguimiento de los totales de las facturas procesadas por categoría o emisor.

#### 3.10.1 Totales por categoría

Para la obtención de los totales de las facturas procesadas por categoría creamos un informe (Report) del tipo "Facturas with Categoría". La unión de ambos objetos nos permite obtener los resultados clasificados por categorías de manera que es posible generar gráficos mostrando totales y porcentajes totales tal y como se muestra en la Ilustración 11.

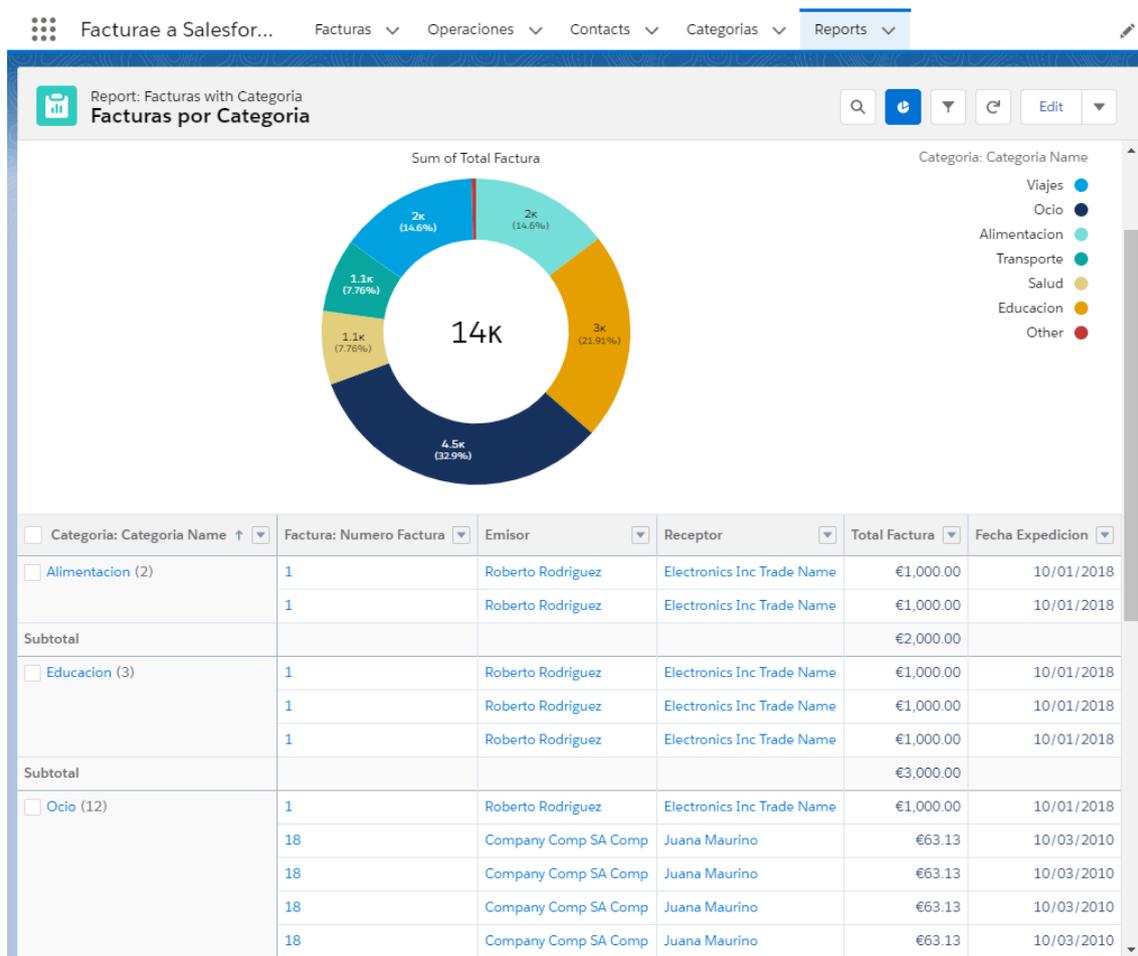


Ilustración 11: Informe Totales por Categoría

### 3.10.2 Totales por emisor

Para la obtención de los totales de las facturas procesadas por emisor creamos otro informe del tipo “Facturas with Categoría”. Este tipo de informe incorpora los datos necesarios de los emisores, por lo tanto, al realizar una agrupación adicional por el campo Emisor nos permite obtener los totales que buscamos tal y como se muestra en la Ilustración 12.

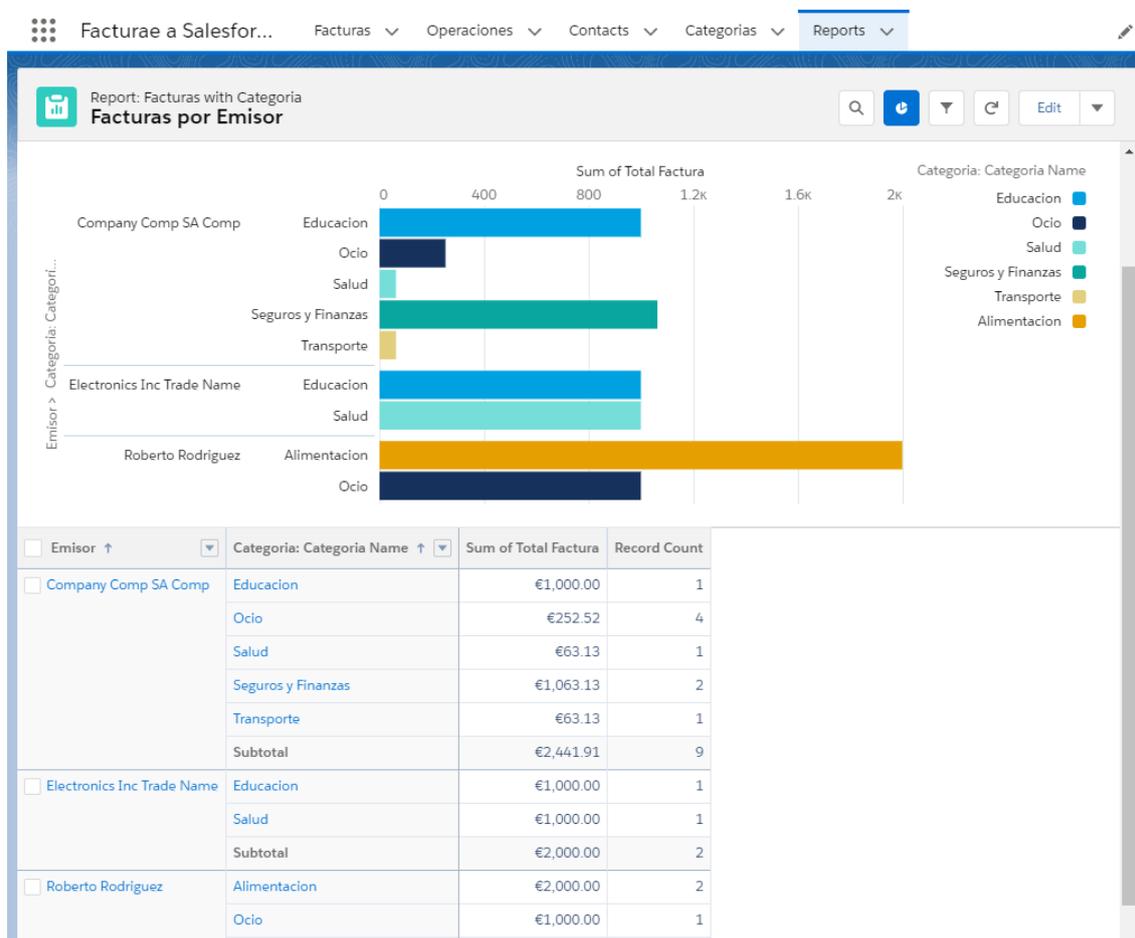


Ilustración 12: Informe Facturas por Emisor

## 4. Implementación

En este apartado, se detalla el proceso de implementación llevado a cabo para construir la aplicación, los requisitos previos y las decisiones tomadas más relevantes.

### 4.1 Requisitos previos

Para la implementación de este proyecto, existen una serie de requisitos previos para poder empezar a trabajar.

#### 4.1.1 Creación de cuenta Salesforce

Dado que la aplicación está exclusivamente diseñada para trabajar con Salesforce, lo primero que necesitamos es una cuenta Developer Edition gratuita para tener un entorno Cloud necesario para dar soporte a la funcionalidad de esta solución propuesta.

Esta cuenta gratuita nos permitirá definir los metadatos necesarios para dar soporte al modelo de datos, hacer llamadas a la API y la generación de informes y estadísticas.

Para crear una Developer org seguiremos los siguientes pasos:

1. Acceder a <https://developer.salesforce.com/signup>.
2. Introducir la información de contacto.
3. Introducir un nombre de usuario único.
4. Enviar el formulario, y esperar a recibir el email de bienvenida.
5. En el email de bienvenida, hacer clic en el link para establecer una contraseña.

The image shows a registration form for Salesforce Lightning Platform Developer Edition. At the top, the Salesforce logo and 'lightning platform' text are displayed. Below this, the heading 'Get your very own Developer Edition' is followed by the subtext 'A full-featured copy of Lightning Platform, for FREE.' The form itself is a white box with a blue border, containing several input fields: 'Name' (split into 'First' and 'Last'), 'Email' (with placeholder 'Your email address'), 'Role' (a dropdown menu with 'Your job role'), 'Company' (with placeholder 'Company Name'), 'Country' (a dropdown menu with 'United States'), 'Postal Code', and 'Username' (with placeholder 'Ex: name@yourcompany.com'). Below the fields is a checkbox for terms and conditions, and a blue 'Sign me up >' button. At the bottom of the form area, there is a link for 'Log In' for those who already have a Salesforce Developer Environment.

Ilustración 13: Formulario de Solicitud de una Developer Org de Salesforce

#### 4.1.2 Connected App

Una vez obtenida la cuenta de Salesforce, como primer paso para poder realizar una integración desde cualquier aplicación tenemos que registrar una Connected App. Una Connected app es un marco de desarrollo que permite a una aplicación externa integrar con Salesforce mediante APIs y protocolos estándar como SAML, OAuth, y OpenID Connect. Las Connected apps hacen uso de estos protocolos para autenticar, autorizar y proveer de single sign-on (SSO) para aplicaciones externas.

Para registrar una Connected app será necesario especificar varios detalles como el nombre, descripción, logo, protocolos de autorización, el tipo de datos a los que podrá acceder, entre otros.

The screenshot shows the Salesforce Setup interface for creating a new Connected App. The page title is 'New Connected App'. At the top, there are 'Save' and 'Cancel' buttons. Below them, a note states: 'To publish an app, you need to be using a Developer Edition organization with a namespace prefix chosen.' The form is divided into several sections: 'Basic Information' (with a 'Required Information' indicator), 'API (Enable OAuth Settings)', 'Web App Settings', and 'Custom Connected App Handler'. The 'Basic Information' section includes fields for Connected App Name, API Name, Contact Email, Contact Phone, Logo Image URL (with a link to 'Upload logo image or Choose one of our sample logos'), Icon URL (with a link to 'Choose one of our sample logos'), Info URL, and Description. The 'API (Enable OAuth Settings)' section has a checkbox for 'Enable OAuth Settings'. The 'Web App Settings' section has a text field for 'Start URL' and a checkbox for 'Enable SAML'. The 'Custom Connected App Handler' section has a text field for 'Apex Plugin Class'.

Ilustración 14: Creación de una Connected App

Una vez registrada, se nos proporciona una *Consumer Key*, que hay que incluir en el código fuente de la aplicación. La cuenta utilizada para registrar la aplicación es indiferente; una vez registrada, la aplicación se puede utilizar para acceder a cualquier cuenta de Salesforce.

## 4.2 Herramientas de desarrollo

En esta sección se describen brevemente las herramientas principales utilizadas en el desarrollo del proyecto.

### 4.2.1 Eclipse

Eclipse es un editor de código abierto y de uso gratuito creado originalmente como un entorno de desarrollo exclusivo de Java. Hoy en día, su funcionalidad se ha incrementado, dando soporte también al desarrollo en otros lenguajes con C, C++, PHP, Python, Ruby o Perl entre otros.

Existen otros entornos de desarrollo para Java, entre los que destacan Netbeans e IntelliJ IDEA. Netbeans, es el IDE de Oracle, que es el principal distribuidor de JDK. Es gratuito y muy usado en el ámbito

académico, pero no tanto en el profesional. IntelliJ IDEA, por el contrario, es recomendado por muchos profesionales, pero no es gratuito y no es tan usado (Edu4java.com, 2019).

Una de las ventajas de Eclipse frente a otros entornos de desarrollo, es importante destacar el alto grado de extensibilidad que ofrece mediante la instalación de plugins disponibles gracias a la comunidad activa de desarrolladores. Teniendo en cuenta esta robustez y que para nuestro proyecto una de las claves es la extensibilidad (puesto que requerimos de diversas tecnologías) hacen de Eclipse la opción más apropiada.

#### 4.2.2 Maven

Maven es una herramienta de software para la gestión y construcción de proyectos Java creada por Jason van Zyl, de Sonatype, en 2002. Es similar en funcionalidad a Apache Ant (y en menor medida a PEAR de PHP y CPAN de Perl), pero tiene un modelo de configuración de construcción más simple, basado en un formato XML. Estuvo integrado inicialmente dentro del proyecto Jakarta, pero ahora ya es un proyecto de nivel superior de la Apache Software Foundation (Porter, Zyl and Lamy, 2019).

Una característica clave de Maven es que está listo para usar en red. El motor incluido en su núcleo puede dinámicamente descargar plugins de un repositorio, el mismo repositorio que provee acceso a muchas versiones de diferentes proyectos Open Source en Java, de Apache y otras organizaciones y desarrolladores. Este repositorio y su sucesor reorganizado, el repositorio Maven, pugnan por ser el mecanismo de facto de distribución de aplicaciones en Java, pero su adopción ha sido muy lenta. Maven provee soporte no solo para obtener archivos de su repositorio, sino también para subir artefactos al repositorio al final de la construcción de la aplicación, para que puedan acceder todos los usuarios. Una caché local de artefactos actúa como la primera fuente para sincronizar la salida de los proyectos a un sistema local.

Para este proyecto hemos utilizado Maven para producir los ficheros JAR de la aplicación encargada de procesar las facturas. Hemos optado por esta herramienta para la construcción de nuestra aplicación porque al definir en el fichero POM.xml de configuración de Maven propio de nuestro proyecto las dependencias de librerías que necesitamos, procede a su descarga y compilación automáticamente al generar el JAR. Además, su buena integración con el IDE Eclipse, hace que vaya comprobando o nos sugiera en tiempo real aquellas librerías de las que hacemos o deseemos hacer uso:

### 4.3 Validación de facturas

En la validación de las facturas se verifica su formato y la firma digital que garantiza su autenticidad e integridad.

#### 4.3.1 Validación del formato

Para implementar la validación del formato de las facturas, puesto que disponemos de los esquemas XML a los que Facturae se debe ceñir, hacemos uso de las librerías Apache XMLBeans. XMLBeans es una herramienta que nos permite hacer uso de XML en Java de manera amigable, haciendo eso de las propiedades de XML y de esquemas XML para realizar un mapeado de manera natural en Java.

XMLBeans hace uso de esquemas XML para compilar interfaces y clases Java que pueden ser usadas para acceder y modificar ficheros XML. El uso de XMLBeans es similar al uso de cualquier interfaz o clase de Java, con métodos tipo get y set, tal y como se trabaja habitualmente con Java.

Para hacer uso de esta librería modificaremos el POM de la siguiente forma:

```
<dependency>
  <groupId>org.apache.xmlbeans</groupId>
  <artifactId>xmlbeans</artifactId>
  <version>2.6.0</version>
  <type>jar</type>
  <scope>compile</scope>
</dependency>
```

También será necesario incorporar los archivos XSD necesarios en el proyecto en Eclipse y definir la configuración del plugin de la siguiente forma:

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>xmlbeans-maven-plugin</artifactId>
  <version>2.3.3</version>
  <inherited>>true</inherited>
  <configuration>
    <download>>true</download>
    <schemaDirectory>main/resources</schemaDirectory>
    <sourceGenerationDirectory>src</sourceGenerationDirectory>
    <classGenerationDirectory>target/classes</classGenerationDirectory>
  </configuration>
</plugin>
```

### 4.3.2 Validación de la firma

Para la parte de la validación de la firma digital hacemos uso del API XML Digital Signature. Estas librerías basadas en la Java Cryptography Service Provider Architecture proporcionan los métodos necesarios para generar y validar firmas en formato XML, permitiendo autenticar y proteger la integridad de los datos en XML (Docs.oracle.com, 2019).

Para implementar el proceso de validación de la firma utilizamos el DocumentBuilderFactory para analizar el formato del documento XML contenedor de la firma (elemento Signature):

```
DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
```

Mediante una llamada al método DOM `getElementsByTagNameNS` obtenemos el elemento contenedor de la firma:

```
DocumentBuilder builder = dbf.newDocumentBuilder();  
Document doc = builder.parse(facturae);  
  
doc.getElementsByTagNameNS(XMLSignature.XMLNS, "Signature")
```

Si no existe el elemento Signature damos por hecho que la factura no está firmada. Si está presente, invocamos el método `unmarshalXMLSignature` para a continuación pasarlo al contexto validador que creamos en anterioridad:

```
XMLSignature signature = factory.unmarshalXMLSignature(valContext);
```

Ahora que tenemos la firma, solo es necesario invocar al método `validate`, que devolverá `true` si la firma es correcta y `false` de lo contrario:

```
boolean coreValidity = signature.validate(valContext);
```

### 4.4 Integración con Salesforce

Para hacer uso de la API REST que ofrece Salesforce tanto para la autenticación como para acceder a los datos almacenados hacemos uso de las librerías `HttpClient` y `JSON`. `HttpClient` nos permite acceder a los recursos HTTP para crear las peticiones de comunicación con la API y las librerías de `JSON` nos proporcionan las herramientas necesarias para definir el formato de las consultas y procesar las respuestas obtenidas de la plataforma Salesforce.

Para hacer uso de la librería `HttpClient` modificamos el POM de Maven para incluir lo siguiente:

```
<dependency>
  <groupId>org.apache.httpcomponents</groupId>
  <artifactId>httpclient</artifactId>
  <version>4.5.10</version>
</dependency>
```

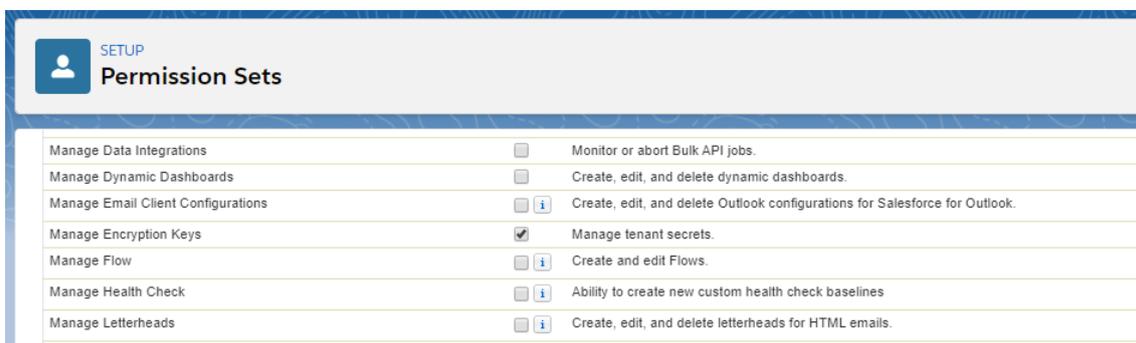
Para incluir la librería JSON es necesario incorporar al POM lo siguiente:

```
<dependency>
  <groupId>org.json</groupId>
  <artifactId>json</artifactId>
  <version>20190722</version>
</dependency>
```

#### 4.4 Cifrado de campos sensibles

Para implementar el cifrado sobre los campos sensibles es necesario configurar el cifrado de plataforma Shield. Para ello seguiremos los siguientes pasos:

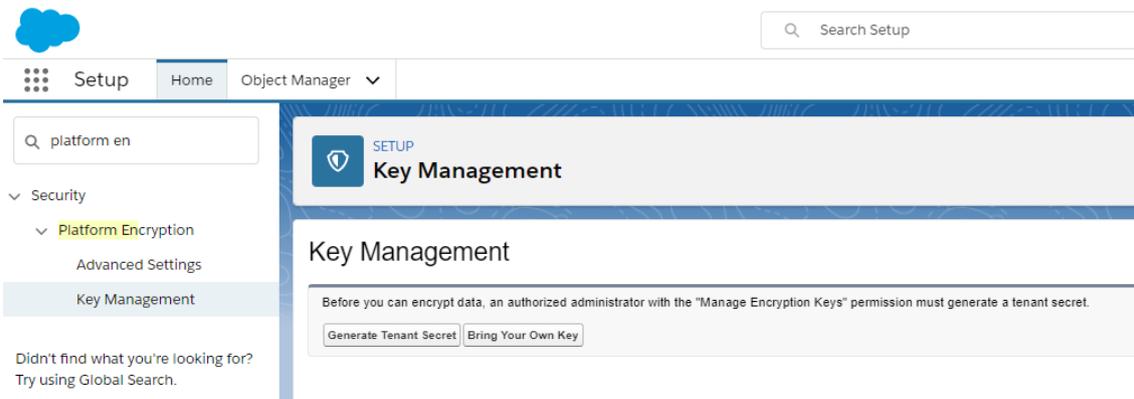
1. Adquirir una licencia de Shield, en nuestro caso no es necesario porque las Developer Edition la incorporan desde la versión Summer 2015.
2. Asignar los permisos necesarios al usuario encargado de gestionar las claves. Para ello, este usuario necesita los permisos “Customize Application” y “Manage Encryption Keys”. Los podremos añadir fácilmente mediante un set de permisos que incorpore ambos.



*Ilustración 15: Permisos Necesarios para Shield*

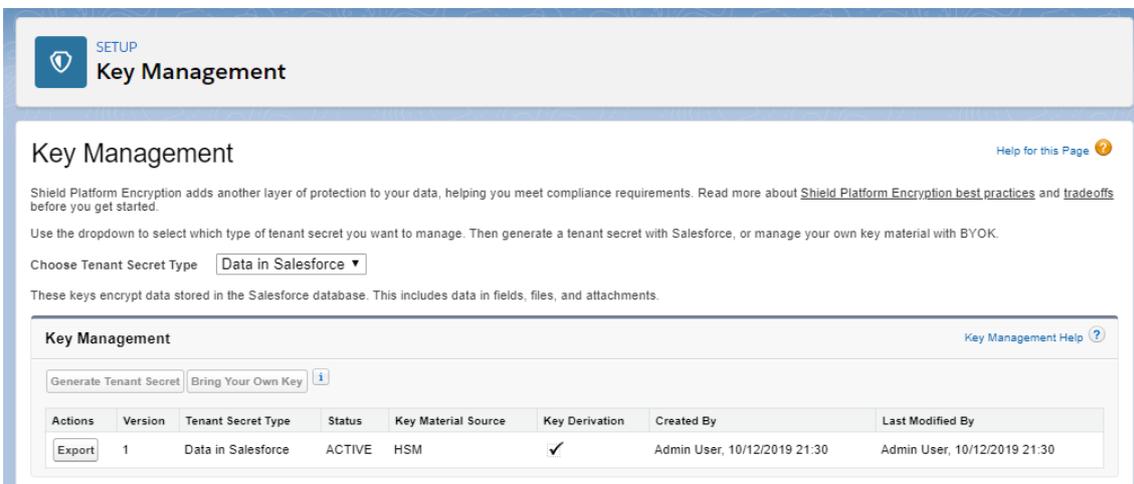
Es importante auditar correctamente qué usuario/s tendrá acceso a esta funcionalidad puesto que es importante limitarla a usuarios que realmente lo necesiten con esta finalidad.

3. Activar el cifrado mediante la generación de un Tenant Secret.



*Ilustración 16: Generación del Tenant Secret*

Es aconsejable guardar el Tenant Secret, en un lugar seguro como, por ejemplo, una llave USB mediante la opción Export que se muestra en la Ilustración 17.



*Ilustración 17: Guardar Tenant Secret*

4. Una vez creado el Tenant Secret definimos los campos que queremos cifrar seleccionando la opción Encrypt para cada uno de ellos.

Edit Contact Custom Field  
Identificación Fiscal

The screenshot shows the 'Custom Field Definition Edit' window for a field named 'Identificación Fiscal'. The 'Field Information' section includes fields for 'Field Label' (Identificación Fiscal), 'Field Name' (Identificación\_Fiscal), 'Description', 'Help Text', 'Data Owner' (User), 'Field Usage' (--None--), and 'Data Sensitivity Level' (--None--). The 'Compliance Categorization' section features two lists: 'Available' (PII, HIPAA, GDPR, PCI) and 'Chosen'. The 'General Options' section contains several checkboxes: 'Required' (unchecked), 'Unique' (unchecked), 'External ID' (unchecked), and 'Encrypted' (checked). The 'Default Value' is set to 'Show Formula Editor'.

Ilustración 18: Cifrado de Campos Sensibles

Puesto que queremos cifrar la factura íntegra (que almacenamos como fichero adjunto) es necesario acceder a Encryption Policy y seleccionar “Encrypt Files and Attachments”.

The screenshot shows the 'Encryption Policy' configuration page. The page title is 'Encryption Policy' with a 'SETUP' icon. Below the title, there is a brief introduction: 'Platform Encryption helps you protect your data and meet compliance requirements. To apply an encryption policy to your data, select the items you want to encrypt. Choose Encrypt Fields to manage field-level encryption settings. Read more about Shield Platform Encryption best practices and tradeoffs before you get started.' The 'File and Field Encryption' section is active, showing a list of 'Encrypt Fields' options: 'Encrypt Files and Attachments' (checked), 'Encrypt Chatter', 'Encrypt Search Indexes', and 'Encrypt and deliver Change Data Capture events'. A 'Save' button is located at the bottom left of this section.

Ilustración 19: Cifrado de Ficheros Adjuntos

Una vez activado el cifrado se haya activado para los campos recibiremos un correo confirmando la acción tal y como se muestra en la Ilustración 20.

## Sandbox: Salesforce Encryption Enabled ▶ Inbox x



**support@salesforce.com** <support@salesforce.com>  
to me ▾

Hello rvispo@production.com.tfm2,

Encryption is now enabled on one or more fields.

These fields use probabilistic encryption:

Contact: Identificacion Fiscal

Salesforce Shield Platform Encryption

*Ilustración 20: Confirmación del Correcto Cifrado de un Campo*

## 5. Pruebas

En este apartado se describe las diferentes pruebas realizadas para comprobar el correcto funcionamiento de la aplicación.

### 5.1 Creación de facturas

El juego de facturas que se crearán (usando la aplicación Facturae tal y como se explica en la sección 3.4) contendrán diversos receptores y emisores ficticios, con diversas operaciones describiendo bienes y servicios. Para realizar las firmas de las facturas se utilizarán diversos certificados electrónicos, haciendo uso de los emitidos por autoridades de certificación reconocidas como la FNMT o la Dirección General de la Policía (DNle).

Se creará también un juego de pruebas con facturas incorrectas con elementos de la estructura no presentes, con valores inapropiados, facturas sin firmar, con errores en la firma o facturas cuyo contenido ha sido modificado.

### 5.2 Plan de pruebas

El plan de pruebas se ha dividido en las siguientes categorías:

- Pruebas de la validación del formato Facturae
- Pruebas de la validación de la firma electrónica
- Pruebas de la categorización de las facturas
- Pruebas del almacenamiento del contenido de las facturas
- Pruebas del almacenamiento del fichero con la factura
- Prueba del correcto cifrado de los campos

#### 5.2.1 Pruebas de la validación del formato Facturae

El objetivo de estas pruebas es obtener la confirmación de que los archivos XML contenedores de las facturas se adaptan al formato Facturae, incluyendo los campos mandatorios en el formato (nombres de campos y tipos) correctos. Para ello se realizan las pruebas tal y como se indican en la Tabla 6.

Prueba	Descripción	Resultado esperado	Resultado obtenido
Se procesa una factura con formato correcto.	Se selecciona una de las facturas correctas, se especifica cualquier categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce a las facturas una nueva entrada se habrá creado con los valores de la factura.	OK
Se procesa una factura firmada pero con un campo adicional no requerido por el formato Facturae.	Se selecciona una factura donde uno de los campos no requeridos está presente, se selecciona una categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce a las facturas una nueva entrada se habrá creado con los valores de la factura.	OK
Se procesa una factura firmada pero con un campo con tipo incorrecto.	Se selecciona una factura donde uno de los campos requeridos muestra un valor no permitido o de un tipo diferente, se selecciona una categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un error indicando el nombre del campo cuyo valor no es correcto. Al acceder a Salesforce no se habrá creado ninguna entrada nueva para esa factura.	OK
Se procesa una factura firmada	Se selecciona una factura	La interfaz mostrará un error	OK

pero sin un campo.	donde uno de los campos requeridos no está presente, se selecciona una categoría y se hace clic en el botón Procesar factura.	indicando el campo requerido que falta. Al acceder a Salesforce no se habrá creado ninguna entrada nueva.	
--------------------	---	---	--

*Tabla 6: Pruebas de la validación del formato Facturae*

### 5.2.2 Pruebas de la validación de la firma electrónica

El objetivo de estas pruebas es obtener la confirmación que las facturas a almacenar están correctamente firmadas digitalmente y no se hayan modificado después de firmadas. Para realizar esta comprobación se realizan las pruebas tal y como se muestran en la Tabla 7.

Prueba	Descripción	Resultado esperado	Resultado obtenido
Se procesa una factura firmada correctamente.	Se selecciona una de las facturas correctas, se especifica cualquier categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce a las facturas una nueva entrada se habrá creado con los valores de la factura.	OK
Se procesa una factura que no está firmada.	Se selecciona una de las facturas que no está firmada digitalmente, se especifica cualquier categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje de error indicando que la factura no está firmada. Al acceder en Salesforce a las facturas una nueva entrada no se habrá creado una entrada nueva.	OK
Se procesa una firma firmada cuyo campo "TaxIdentification Number" se ha modificado después de firmada.	Se selecciona la factura firmada digitalmente cuyo campo "TaxIdentification Number" se ha modificado después de ser firmada, se especifica cualquier categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje de error indicando que la factura se ha modificado después de ser firmada. Al acceder en Salesforce a las facturas no se habrá creado una entrada nueva.	OK

Tabla 7: Pruebas de la validación de la firma electrónica

### 5.2.3 Pruebas de la categorización de las facturas

El objetivo de estas pruebas es obtener la confirmación del correcto funcionamiento de la categorización de las facturas procesadas por la aplicación. Para realizar esta comprobación se realizan las pruebas tal y como indica la Tabla 8.

Prueba	Descripción	Resultado esperado	Resultado obtenido
Se procesa una factura firmada correctamente con la categoría "Ocio".	Se selecciona una de las facturas correctas, se especifica la categoría "Ocio" y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce se habrá creado una nueva entrada cuyo campo Categoría mostrará el valor "Ocio".	OK
Se procesa una factura firmada correctamente con la categoría "Viajes".	Se selecciona una de las facturas correctas, se especifica la categoría "Viajes" y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce se habrá creado una nueva entrada cuyo campo Categoría mostrará el valor "Viajes".	OK

*Tabla 8: Pruebas de la categorización de las facturas*

### 5.2.4 Pruebas del almacenamiento del contenido de las facturas

El objetivo de estas pruebas es obtener la confirmación de que, para las facturas procesadas correctamente, los campos obligatorios son

almacenados en Salesforce. Para realizar esta comprobación se realizan las pruebas tal y como indica la Tabla 9.

Prueba	Descripción	Resultado esperado	Resultado obtenido
Se procesa una factura correcta y su contenido se almacena.	Se selecciona una de las facturas correctas, se especifica una categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce se habrá creado una nueva entrada con todos valores tal y como especifica la factura electrónica.	OK
Se procesa una factura incorrecta y su contenido no se almacena.	Se selecciona una de las facturas incorrectas, se especifica una categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje de error indicando que la factura no se ha procesado correctamente. Al acceder en Salesforce no se habrá creado una nueva entrada para esa factura.	OK
Se procesa una factura correcta con dos operaciones y su contenido se almacena.	Se selecciona una de las facturas correctas que contiene dos operaciones, se especifica una categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce se habrá creado una nueva entrada con todos valores tal y como	OK

		<p>especifica la factura electrónica. Al acceder en la pestaña Related, se observa como las dos operaciones especificadas en la factura se han creado correctamente.</p>	
--	--	--	--

Tabla 9: Pruebas del almacenamiento del contenido de las facturas

### 5.2.5 Pruebas del almacenamiento del fichero con la factura

El objetivo de estas pruebas es obtener la confirmación de que, para las facturas procesadas correctamente, el archivo XML se almacenará íntegramente en Salesforce. Para realizar esta comprobación se realizan las pruebas tal y como indica la Tabla 10.

Prueba	Descripción	Resultado esperado	Resultado obtenido
Se procesa una factura correcta y la factura se almacena.	Se selecciona una de las facturas correctas, se especifica una categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje indicando que la factura se ha procesado correctamente. Al acceder en Salesforce se habrá creado una nueva entrada y al acceder a la pestaña Related en la sección attachments se encuentra la factura íntegra. Al descargarla y compararla con la original, los ficheros son	OK

		idénticos.	
Se procesa una factura incorrecta y la factura no se almacena.	Se selecciona una de las facturas incorrectas, se especifica una categoría y se hace clic en el botón Procesar factura.	La interfaz mostrará un mensaje de error indicando que la factura no se ha procesado correctamente. Al acceder en Salesforce no se habrá creado una nueva entrada para esa factura y consecuentemente no se habrá almacenado íntegramente.	OK

Tabla 10: Pruebas del almacenamiento del fichero con la factura

### 5.2.6 Prueba del correcto cifrado de los campos

El objetivo de estas pruebas es obtener la confirmación que el cifrado de los campos sensibles se realiza correctamente. Para probar el cifrado correcto de los campos será necesario archivar el Tenant Secret, asegurándonos que previamente lo hemos almacenado en un lugar seguro. Para archivarlo bastará con generar uno nuevo.

The screenshot shows the 'Key Management' interface in Salesforce. At the top, there is a 'SETUP' header and a 'Key Management' title. Below this, there is a 'Key Management' section with a 'Help for this Page' link. The main content area includes a description of Shield Platform Encryption and a dropdown menu for 'Choose Tenant Secret Type' set to 'Data in Salesforce'. Below this, there is a table of tenant secrets with columns for Actions, Version, Tenant Secret Type, Status, Key Material Source, Key Derivation, Created By, and Last Modified By.

Actions	Version	Tenant Secret Type	Status	Key Material Source	Key Derivation	Created By	Last Modified By
<a href="#">Export</a>	2	Data in Salesforce	ACTIVE	HSM	✓	Admin User, 27/12/2019 18:08	Admin User, 27/12/2019 18:08
<a href="#">Destroy</a> <a href="#">Export</a>	1	Data in Salesforce	ARCHIVED	HSM	✓	Admin User, 23/12/2019 18:34	Admin User, 27/12/2019 18:08

Ilustración 21: Generación de un Nuevo Tenant Secret

Una vez que se disponga de un Tenant Secret nuevo, se podrá destruir el anterior mediante la opción “Destroy”. Una vez seleccionada esta opción, se mostrará un mensaje de advertencia tal y como muestra la Ilustración 22.

Do you want to destroy version 1 of this Data in Salesforce key material?

Destroy Cancel

When you destroy a tenant secret, you can no longer access the data encrypted with that secret. We recommend exporting your secret before you destroy it, so that you can import it back into Salesforce if you need to decrypt any data that's encrypted with it.

To confirm that you want to destroy your key material, type the following text in the text box.

I want to destroy this key material.

I want to destroy this key material

I understand that destroying this key material means I will lose access to any data encrypted with this key material.

Destroy Cancel

Ilustración 22: Confirmación de Destrucción de un Tenant Secret

Una vez que se destruya, bastará con ir a comprobar cualquiera de los campos cifrados con anterioridad usando ese Tenant Secret y veremos cómo los campos dejan de mostrar los valores para pasar a mostrar “08/08/1888” en caso de fechas o “?????” para campos de tipo texto.

Factura 1

Related Details

Numero Factura	1	Fecha Expedicion	08/08/1888
Numero Serie		Moneda de la Operacion	EUR

Ilustración 23: Campo de Tipo Fecha al Destruir el Tenant Secret

Factura 1

**Related** Details

**Operaciones (1)** New

DESCRIPCION	PORCENTAJE A AP...	CANTIDAD TOTAL ...	COSTE TOTAL
?????	0.0000%	€1,000.00	€1,000.00

Ilustración 24: Campo de Tipo Texto al Destruir el Tenant Secret

Si intentamos descargar las facturas no nos lo permitirá mostrando un error de que la clave no está disponible tal y como muestra la Ilustración 25.

Groups Files Leads Accounts Contacts Opportunities Reports

**Key is unavailable.**

Contact your administrator for more information.

Ilustración 25: Error al Mostrar Factura al Destruir el Tenat Secret

Importando de nuevo el Tenant Secret podemos comprobar cómo efectivamente los campos se muestran de nuevo correctamente.

## 6. Conclusiones

En este proyecto se han puesto en práctica muchos de los conceptos, metodologías, herramientas y conocimientos relacionados con la seguridad adquiridos a lo largo del máster en conjunción con la experiencia previa en Cloud Computing para desarrollar una aplicación completamente funcional.

Se ha creado una aplicación desde cero, pasando por todas las fases de un proyecto, consiguiendo un producto que cumple con los objetivos iniciales fijados y, además, se han añadido sencillez y automatización a un proceso tedioso como es la gestión manual de facturas.

Esta aplicación dota principalmente a los usuarios de un repositorio centralizado seguro para almacenar sus facturas electrónicas. Además, habilita a sus usuarios no solo de una categorización de las mismas de manera flexible, sino que introduce los datos de las facturas en un CRM de manera que permite obtener informes y estadísticas sobre las mismas.

Al introducir la información de las facturas de manera estructurada en un CRM, este proyecto abre un nuevo abanico de funcionalidad y opciones a sus usuarios. Permitiendo, por ejemplo, el uso de esta información para la creación de campañas de marketing (enfocándose en los productos o servicios registrados) o hacer uso de notificaciones automáticas para tener un mayor control sobre los gastos registrados por categoría.

Con respecto a la planificación inicial es importante mencionar la dificultad añadida de su realización al tener que compaginarlo con un trabajo a tiempo completo. Esto ha supuesto en varias ocasiones de una drástica modificación de la planificación al tener que priorizar de acuerdo con la carga laboral.

### 6.1 Trabajo futuro

Sería interesante evolucionar la aplicación para proporcionar a los usuarios la siguiente funcionalidad adicional:

- Una factura electrónica Facturae puede contener más de una factura, sería interesante dotar la aplicación de esta funcionalidad para permitir el procesado en lote de facturas adicionales.
- La aplicación está limitada al procesado de una versión específica de Facturae, sería interesante introducir la posibilidad de importar facturas en formatos anteriores.
- Convertir la aplicación web en una aplicación en el Cloud, haciendo uso de Heroku (por ejemplo) y exponer la configuración de la cuenta de Salesforce, de forma que cualquier usuario pueda hacer uso esta funcionalidad simplemente accediendo a su URL.

## 7. Bibliografía

Salesforce.com. (2019). [online] Disponible en: <https://www.salesforce.com/products/platform/what-is-digital-transformation/> [Accedido 23 Nov. 2019].

PHC Software. (2019). Facturación electrónica: obligatoria en el 2019 en la contratación pública | PHC Software. [online] Disponible en: <https://www.phcsoftware.com/es/business-at-speed-es/facturacion-electronica/> [Accedido 23 Nov. 2019].

Schwaber, K. and Sutherland, J. (2019). [online] Scrumguides.org. Disponible en: <https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf> [Accedido 23 Nov. 2019].

Kneafsey, S. (2019). The Benefits of Scrum & Agile - TheScrumMaster.co.uk. [online] TheScrumMaster.co.uk. Disponible en: <https://www.thescrummaster.co.uk/scrum/benefits-scrum-agile/> [Accedido 23 Nov. 2019].

Barreix, A. and Zambrano, R. (2018). Factura electrónica en América Latina.

Salesforce.com. (2019). [online] Disponible en: <https://www.salesforce.com/ap/campaign/worlds-number-one-CRM/> [Accedido 23 Nov. 2019].

Expansión.com. (2019). El ecosistema de Salesforce generará 48.000 empleos en España. [online] Disponible en: <https://www.expansion.com/economia-digital/companias/2019/11/14/5dcc47af468aeb7228b4624.html> [Accedido 23 Nov. 2019].

The Daily Swig | Cybersecurity news and views. (2019). The end is nigh: Browser-makers ditch support for aging TLS 1.0, 1.1 protocols. [online] Disponible en: <https://portswigger.net/daily-swig/the-end-is-nigh-browser-makers-ditch-support-for-aging-tls-1-0-1-1-protocols> [Accedido 8 Dec. 2019].

Facturae.gob.es. (2019). Factura Electrónica - Descargar Aplicaciones Gestión de Facturación Electrónica 3.4. [online] Disponible en: <https://www.facturae.gob.es/formato/Paginas/descarga-aplicaciones.aspx> [Accedido 8 Dec. 2019].

Facturae.gob.es. (2019). [online] Disponible en: [https://www.facturae.gob.es/formato/Versiones/Facturaev3\\_2\\_1.xml](https://www.facturae.gob.es/formato/Versiones/Facturaev3_2_1.xml) [Accedido 8 Dec. 2019].

Facturae.gob.es. (2019). Factura Electrónica - Políticas de firma electrónica. [online] Disponible en:

<https://www.facturae.gob.es/formato/Paginas/politicas-firma-electronica.aspx> [Accedido 8 Dec. 2019].

Porter, B., Zyl, J. and Lamy, O. (2019). Maven – Welcome to Apache Maven. [online] Maven.apache.org. Disponible en: <https://maven.apache.org/> [Accedido 8 Dec. 2019].

Facturae.gob.es. (2019). Factura Electrónica - Descargar Aplicaciones Gestión de Facturación Electrónica 3.4. [online] Disponible en: <https://www.facturae.gob.es/formato/Paginas/descarga-aplicaciones.aspx> [Accedido 8 Dec. 2019].

SecureBlackbox. (2019). Signing invoices for the Spanish government in the factura format. [online] Disponible en: <https://www.secureblackbox.com/kb/articles/SigningDocuments.rst> [Accedido 9 Dec. 2019].

Oracle.com. (2019). Programming With the Java XML Digital Signature API. [online] Disponible en: <https://www.oracle.com/technical-resources/articles/javase/dig-signature-api.html> [Accedido 10 Dec. 2019].

Docs.oracle.com. (2019). Java XML Digital Signature API. [online] Disponible en: <https://docs.oracle.com/javase/8/docs/technotes/guides/security/xmlldsig/XMLDigitalSignature.html> [Accedido 10 Dec. 2019].

Aepd.es. (2019). [online] Disponible en: <https://www.aepd.es/sites/default/files/2019-09/guia-rgpd-para-responsables-de-tratamiento.pdf> [Accedido 23 Dec. 2019].

Lightningdesignsystem.com. (2019). Lightning Design System. [online] Disponible en: <https://www.lightningdesignsystem.com/> [Accedido 23 Dec. 2019].