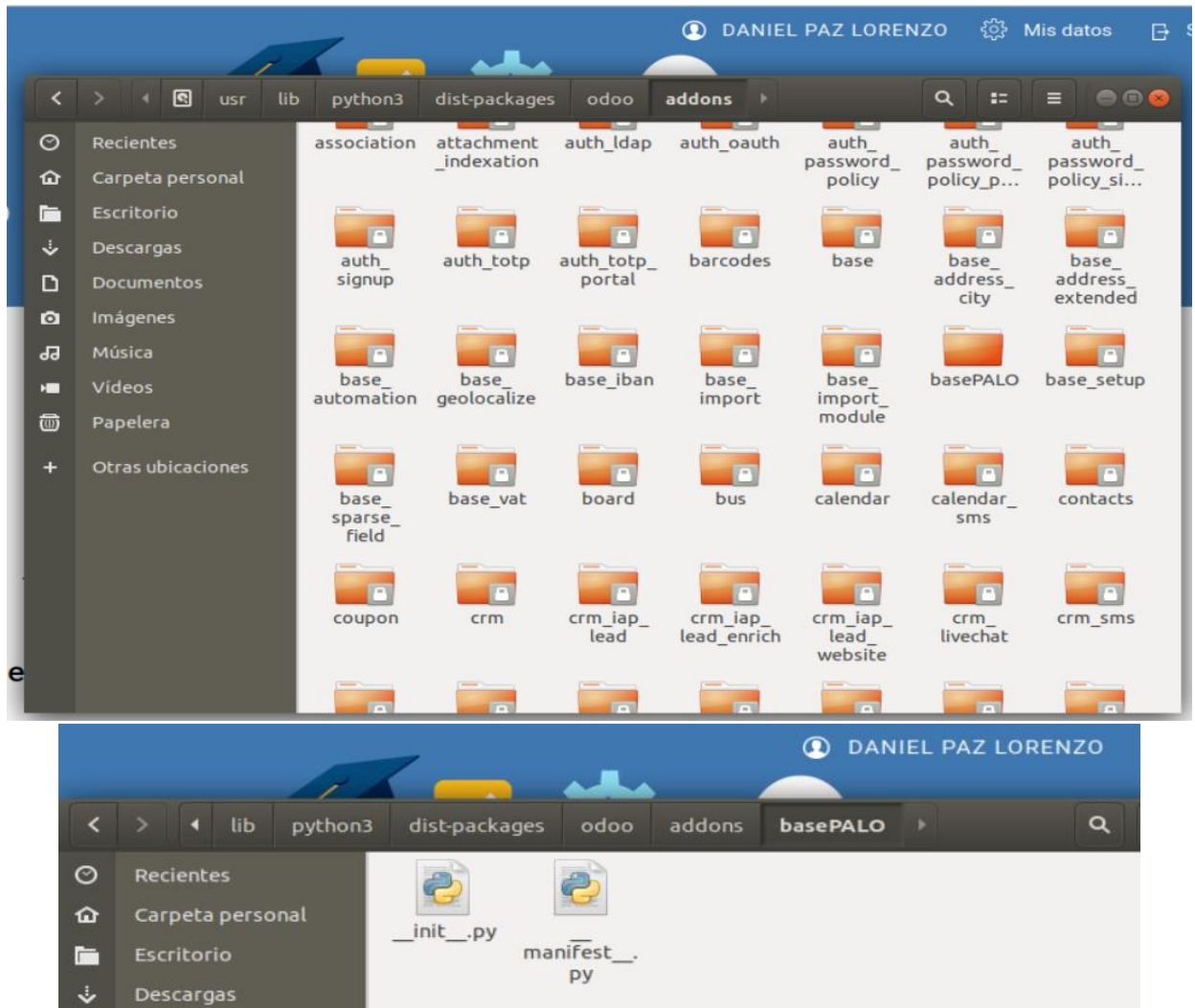


El objetivo de esta tarea es el de hacer una simulación de implantación de un sistema ERP en la empresa *Datalab*, para ello tenemos que seguir los siguientes pasos:

1. CREAR UNA APLICACIÓN

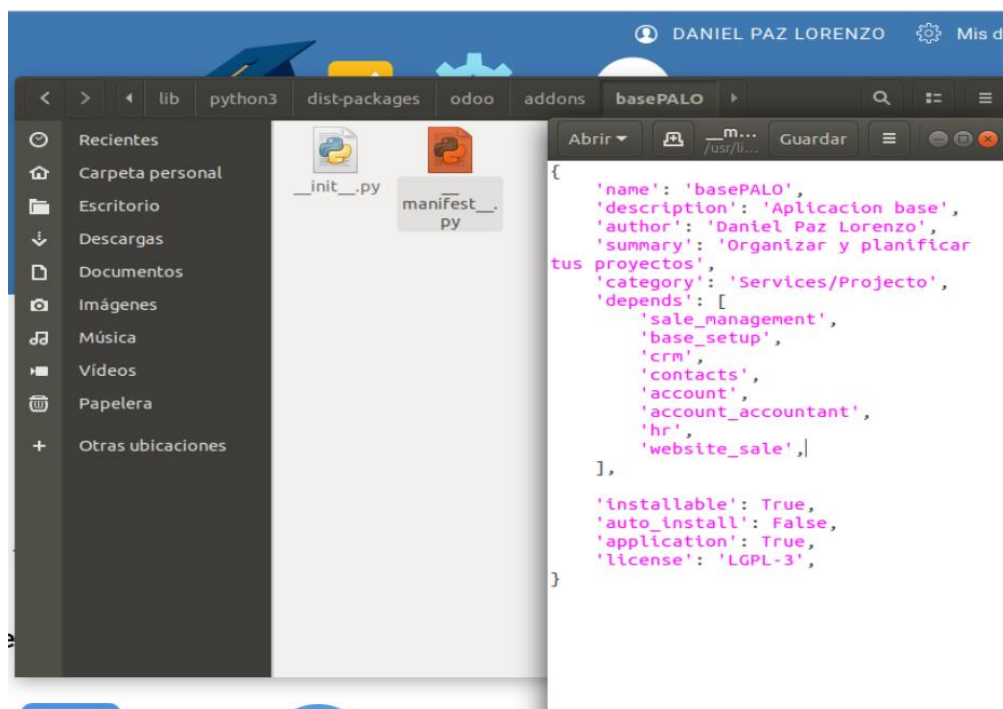
Lo primero que haremos será crear una nueva carpeta con el nombre de “basePALO” dentro de la carpeta addons donde tenemos instalado Odoo:



Dentro de la carpeta creamos los 2 ficheros Python (init y manifest) necesarios para la creación de la aplicación

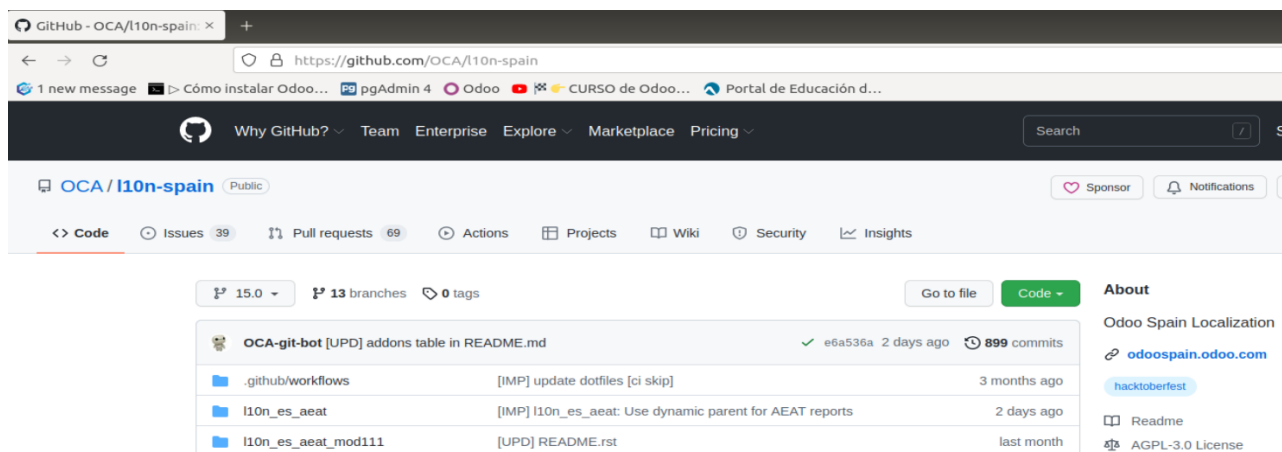
Ya solo nos queda editar estos ficheros para que Odoo la reconozca como una aplicación, le pondremos nombre, una descripción y le diremos los módulos que queremos instalar. Como es para una pyme he seleccionado los siguientes que serían los básicos para empezar:

- **Odoo CRM y Ventas.**
- **Odoo Facturación.**
- **Odoo Contabilidad.**
- **Odoo Recursos Humanos.**
- **Odoo Comercio Electrónico.**
- **Odoo Servicio de Asistencia.**

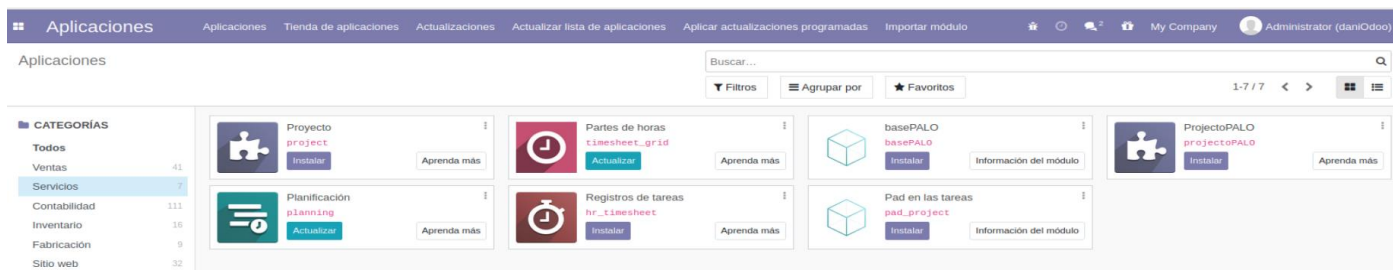


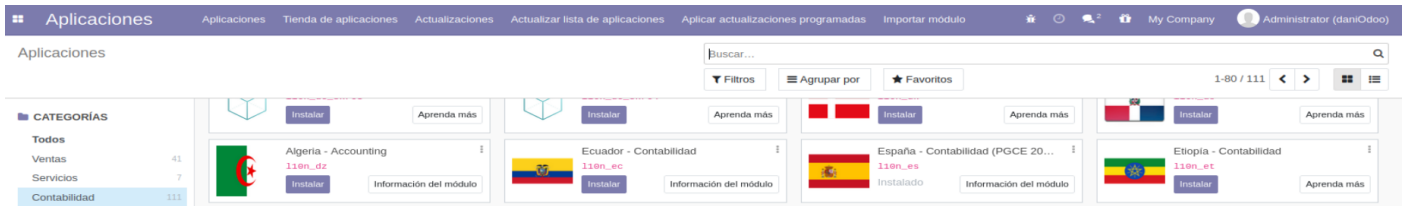
Así quedaría el fichero “manifest__.py”

Como la implantación la vamos a hacer para una empresa española vamos a descargar el módulo de localización española desde el repositorio de github ya que la versión odoo que tengo instalada es comunitaria y la empresa solo da soporte de este módulo pagando:



Ya tenemos todo preparado para instalar así que abrimos Odoo y vamos al apartado de aplicaciones, buscamos los módulos y los instalamos:





Módulo de contabilidad en español



Ya tenemos instalados todos los módulos que necesitamos para la pyme

2. REALIZAR LA MIGRACIÓN DE DATOS DEL ANTIGUO AL NUEVO CLIENTE ERP

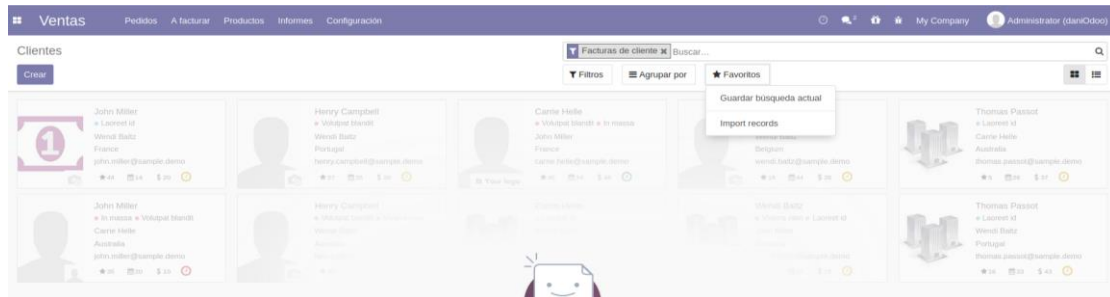
a. Crear e importar archivo .csv a la nueva aplicación ERP

Con el bloc de notas creamos un fichero .csv con los datos inventados de 3 clientes haciendo que los campos coincidan con atributos del objeto res.partner (contactos):

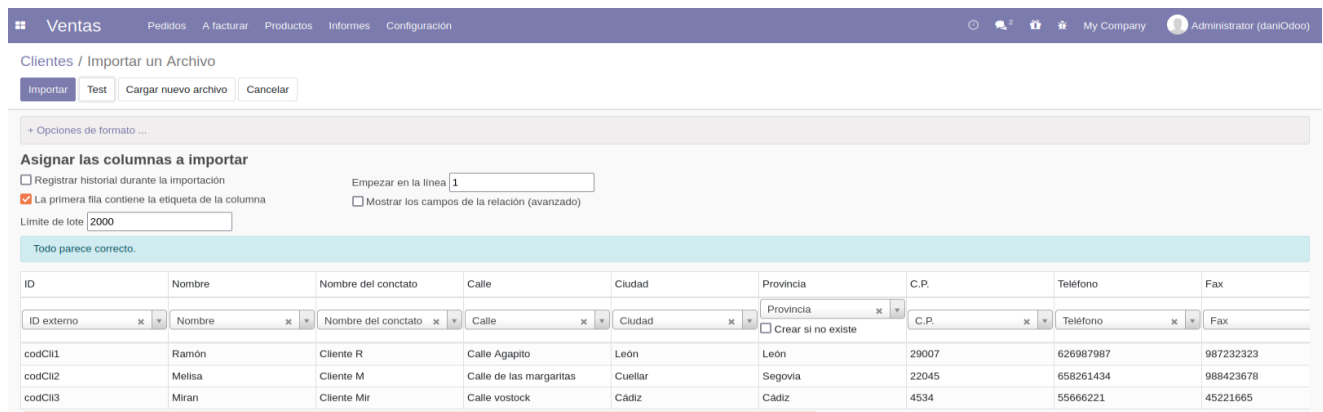


El campo cliente/proveedor no le he exportado porque ya no existe desde la versión 13 de Odoo

Una vez tenemos ya guardado el fichero clientes.csv pasamos a importarlo desde la vista ventas/pedidos/clientes, desde allí pulsamos favoritos/import records y nos llevará a la vista de importación del archivo:



Cargamos el archivo .csv que hemos creado, comprobamos que nos ha cogido los datos correctamente y pulsamos importar:

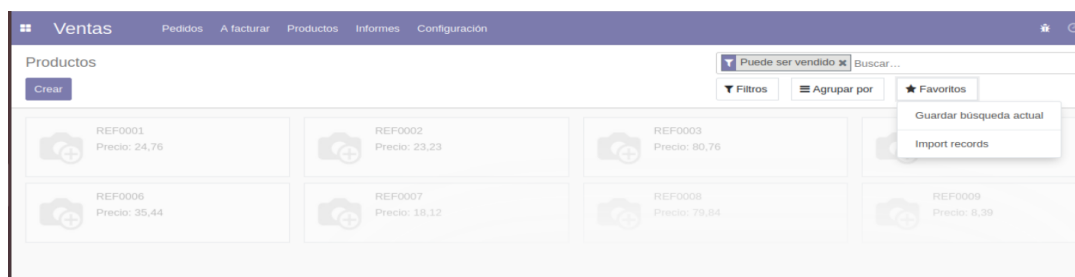


Podemos ver en la vista clientes como se han importado correctamente los nuevos clientes:



b. Importar archivo .csv con datos de producto

Vamos al apartado de ventas y en la etiqueta de productos seleccionamos la opción de importar records:



Desde aquí abrimos el archivo .csv que viene en la tarea y lo abrimos:

Ventas Pedidos A facturar Productos Informes Configuración

Productos / Importar un Archivo

Importar Test Cargar nuevo archivo Cancelar

+ Opciones de formato ...

Codificación: utf-8 Formato de la fecha: Separador de miles: Comma

Separador: Punto y coma Formato de fecha y hora: Separador de decimales: Punto

Delimitador de texto: "

Asignar las columnas a importar

☐ Registrar historial durante la importación Empezar en la línea 1

☒ La primera fila contiene la etiqueta de la columna ☒ Mostrar los campos de la relación (avanzado)

Límite de lote 2000

Código	Nombre	Categoría	Descripción	Método abastecimiento	Método coste	Método suministro	Precio coste	Precio de venta
No importar	Nombre	No importar	Descripción	No importar	No importar	No importar	No importar	Precio de venta
cod1	nombreproducto1	cosmetica	descripcion producto1	make_to_order	standard	buy	0	0
cod2	nombreproducto2	cosmetica	descripcion producto2	make_to_stock	standard	buy	0	0
cod3	nombreproducto3	cosmetica	descripcion producto3	make_to_order	standard	buy	0	0

Podemos ver que hay ciertos campos que no reconoce por el nombre que viene en el archivo, lo solucionamos seleccionando manualmente a que campo serán importados los valores que den error:

Productos / Importar un Archivo

Importar Test Cargar nuevo archivo Cancelar

+ Opciones de formato ...

Asignar las columnas a importar

☐ Registrar historial durante la importación Empezar en la línea 1

☒ La primera fila contiene la etiqueta de la columna ☒ Mostrar los campos de la relación (avanzado)

Límite de lote 2000

Todo parece correcto.

Código	Nombre	Categoría	Descripción	Método abastecimiento	Método coste	Método suministro	Precio coste	Precio de venta
ID externo	Nombre	Categoría de produ...	Descripción	Descripción de Ven...	Descripción de co...	Descripción	Coste	Precio de venta
cod1	nombreproducto1	cosmetica	descripcion producto1	make_to_order	standard	buy	0	0
cod2	nombreproducto2	cosmetica	descripcion producto2	make_to_stock	standard	buy	0	0
cod3	nombreproducto3	cosmetica	descripcion producto3	make_to_order	standard	buy	0	0

This is a preview of the first 10 rows of your file

He modificado en algunos casos el fichero, en otros el nombre del campo y en otros el valor del propio campo

Este sería el resultado que nos presenta Odoo después de ser importados los datos:

Ventas Pedidos A facturar Productos Informes Configuración

Productos


Crear

Puede ser vendido x Buscar...

Filtros Agrupar por



nombreproducto1
Precio: 0,00 €



nombreproducto2
Precio: 0,00 €



nombreproducto3
Precio: 0,00 €

3. SOLUCIONAR LA INCONGRUENCIA EN LOS DATOS IMPORTADOS

Se nos pide crear un procedimiento almacenado dentro de la base de datos de PostgreSQL para solucionar el problema de los datos importados.

Para ello abrimos el administrador Pg4Admin y creamos una consulta que será la base del procedimiento que queremos crear:

The screenshot shows the PgAdmin interface with the 'Query Editor' tab active. The query is as follows:

```
1 SELECT DISTINCT res_partner.name AS Nombre_Cliente , res_country.name AS País, res_country_state.name AS Provincia
2 FROM public.res_partner, public.res_country, public.res_country_state
3 WHERE res_partner.country_id=res_country.id AND res_partner.state_id=res_country_state.id;
```

The 'Salida de Datos' (Data Output) tab shows the results of the query:

	nombre_cliente character varying	país character varying	provincia character varying
1	Baterías Loper	Portugal	Lisboa
2	Bolquetes Peto	Spain	Girona (Gerona)
3	COMPANIA_DANIEL S.L.	Spain	Valladolid
4	Fran	Spain	Valladolid
5	Joao	Portugal	Lisboa
6	Melisa	Spain	Segovia
7	Miran	Spain	Cádiz
8	Pimardicop	Spain	Valladolid
9	Ramón	Spain	León
10	Tere	Spain	Girona (Gerona)

Ejecutamos el script y comprobamos que se ejecuta correctamente y muestra los datos deseados

A continuación ya podemos crear el procedimiento dentro del esquema de la base de datos en el apartado de procedimientos:

The screenshot shows the PgAdmin interface with the 'Query Editor' tab active. The query is as follows:

```
1 CREATE PROCEDURE public.Procedimiento_Tarea_4()
2 LANGUAGE 'sql'
3 AS $BODY$
4 SELECT DISTINCT res_partner.name AS Nombre_Cliente , res_country.name AS País, res_country.
5 FROM public.res_partner, public.res_country, public.res_country_state
6 WHERE res_partner.country_id=res_country.id AND res_partner.state_id=res_country_state.
7 $BODY$;
```

The 'Mensajes' (Messages) tab shows the output of the query:

```
CREATE PROCEDURE
Consulta retornó exitosamente en 125 msec.
```

Ejecutamos el procedimiento creado mediante script y comprobamos que el resultado es el que nos piden:

	nombre_cliente character varying	país character varying	provincia character varying
1	Baterías Loper	Portugal	Lisboa
2	Bolquetes Peto	Spain	Girona (Gerona)
3	COMPANIA_DANIEL S.L.	Spain	Valladolid
4	Fran	Spain	Valladolid
5	Joao	Portugal	Lisboa
6	Melisa	Spain	Segovia
7	Miran	Spain	Cádiz
8	Pimardicop	Spain	Valladolid
9	Ramón	Spain	León
10	Tere	Spain	Girona (Gerona)

4. CREAR UN INFORME PERSONALIZADO


Desde que OpenERP cambió y se convirtió en Odoo ha cambiado la forma de hacer los informes, así que para realizar este punto de la tarea he utilizado la documentación oficial de Odoo para la versión 14 que tengo instalada en mi máquina virtual de Linux:

<https://www.odoo.com/documentation/14.0/developer/reference/addons/reports.html>

Lo primero que haremos será abrir el archivo de Odoo en el IDE de Python “PyCharm” que he instalado en mi máquina virtual, aunque no es necesario para realizar la tarea es cómodo para navegar por los archivos de la aplicación y crear nuevos archivos:

Desde Pycharm abro el fichero donde tengo instalado Odoo

Ahora siguiendo los pasos de la documentación debemos crear un documento “*.xml” que contenga la definición del action.report y la plantilla del informe o también llamada template:

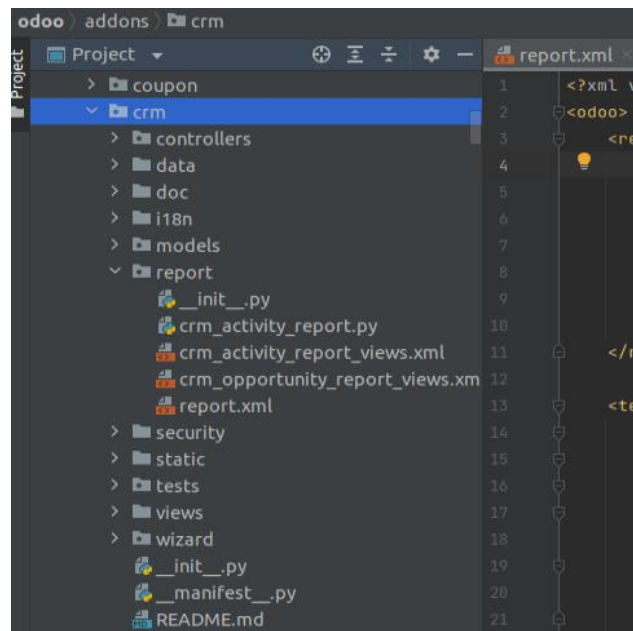


```

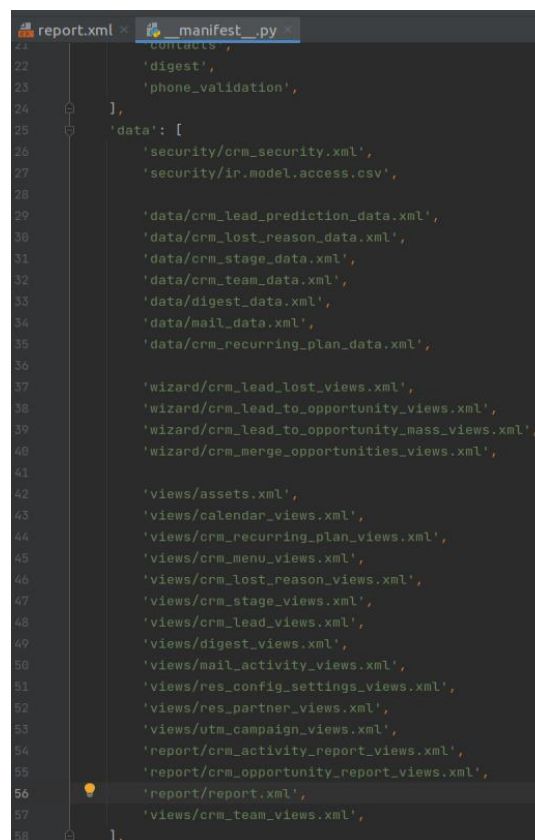
1  <?xml version="1.0" encoding="utf-8"?>
2  <odoo>
3      <record id="crm.empresa_report" model="ir.actions.report">
4          <field name="name">InformePAL0</field>
5          <field name="model">res.partner</field>
6          <field name="report_type">qweb-pdf</field>
7          <field name="report_name">crm.report_empresas</field>
8          <field name="report_file">crm.report_empresas</field>
9          <field name="binding_model_id" ref="model_res_partner"/>
10         <field name="binding_type">report</field>
11     </record>
12
13     <template id="report_empresas">
14         <t t-call="web.html_container">
15             <t t-foreach="docs" t-as="o">
16                 <t t-call="web.external_layout">
17                     <div class="page">
18                         <h2>Informe de la tarea 4</h2>
19                         <p>Nombre de la compañía
20                             <span t-field="o.name"/>
21                         </p>
22                         <p>País de la compañía
23                             <span t-field="o.country_id"/>
24                         </p>
25                         <p>Provincia de la compañía
26                             <span t-field="o.state_id"/>
27                         </p>
28                     </div>
29                 </t>
30             </t>
31         </t>
32     </template>
33 </odoo>
  
```

Creamos un nuevo documento al que llamaremos “report.xml” que contendrá la definición del action.report como queremos que actúe y el formato de la plantilla que va a tener el informe

Este archivo lo vamos a guardar dentro de /addons/crm/report porque CRM es el módulo que vamos a utilizar para crear nuestro informe ya que contiene el modelo “res.partner” que es el que nos interesa para poder extraer la información del nombre, país y provincia de la empresa:



El siguiente paso es indicar dentro del archivo “_manifest__.py” del módulo CRM, cual es la ruta de este nuevo fichero para que el interprete de Python lo tenga en cuenta:



Incluimos la ruta dentro del apartado Data

Para que estos cambios tengan efecto dentro de la aplicación tenemos que abrir Odoo, ir a aplicaciones y actualizar el módulo de CRM que es el que hemos modificado:



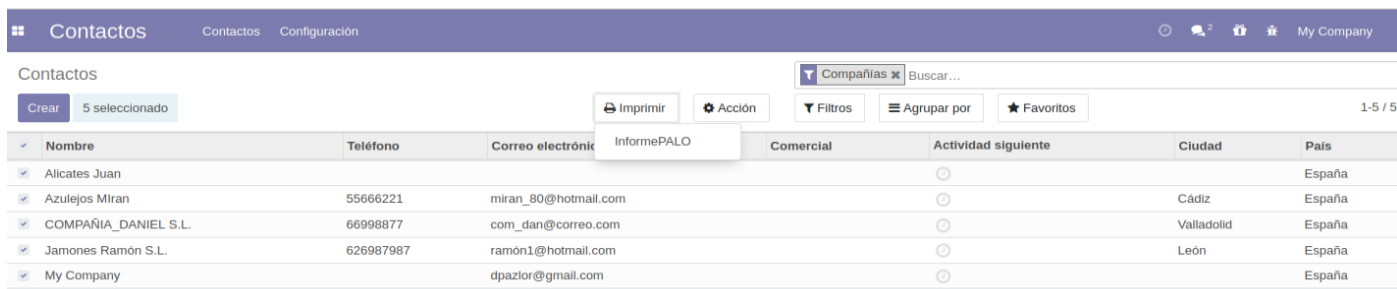
Pulsamos en actualizar y transcurrido un tiempo se va a reiniciar Odoo ya con los nuevos cambios

Para comprobar que la aplicación ha recogido esta nueva acción de reporte nos vamos a ajustes/técnico/informes:

Ajustes					
Opciones Generales Usuarios y compañías Traducciones Técnico					
Informes					
Crear 1 seleccionado		<div> <div>Acción</div> <div>Filtros</div> <div>Agrupar por</div> <div>Favori</div> </div>			
<input type="checkbox"/>	Nombre	Nombre del modelo	Tipo de acción	Nombre de plantilla	Tipo de informe
<input type="checkbox"/>	Auditoria de Libros	account.common.journal.report	ir.actions.report	account.report_journal	PDF
<input type="checkbox"/>	Código de Barras del Producto (PDF)	product.template	ir.actions.report	product.report_producttemplatebarcode	PDF
<input type="checkbox"/>	Código de Barras del Producto (PDF)	product.product	ir.actions.report	product.report_productbarcode	PDF
<input type="checkbox"/>	Embalaje del Producto (PDF)	product.packaging	ir.actions.report	product.report_packagingbarcode	PDF
<input type="checkbox"/>	Etiqueta del Producto (PDF)	product.product	ir.actions.report	product.report_productlabel	PDF
<input type="checkbox"/>	Etiqueta del Producto (PDF)	product.template	ir.actions.report	product.report_producttemplatelabel	PDF
<input type="checkbox"/>	Extracto	account.bank.statement	ir.actions.report	account.report_statement	PDF
<input type="checkbox"/>	Factura PRO-FORMA	sale.order	ir.actions.report	sale.report_saleorder_pro_forma	PDF
<input type="checkbox"/>	Facturas	account.move	ir.actions.report	account.report_invoice_with_payments	PDF
<input type="checkbox"/>	Facturas originales	account.move	ir.actions.report	account.report_original_vendor_bill	PDF
<input type="checkbox"/>	Facturas sin pago	account.move	ir.actions.report	account.report_invoice	PDF
<input type="checkbox"/>	Guía técnica	ir.module.module	ir.actions.report	base.report_irmodulereference	PDF
<input type="checkbox"/>	Imprimir Identificación	hr.employee	ir.actions.report	hr.print_employee_badge	PDF
<input type="checkbox"/>	Información general del modelo	ir.model	ir.actions.report	base.report_irmodeloverview	PDF
<input type="checkbox"/>	Informe de empresas	contacts.base	ir.actions.report	contacts.report_empresas_view	PDF
<input checked="" type="checkbox"/>	InformePALO	res.partner	ir.actions.report	crm.report_empresas	PDF
<input type="checkbox"/>	Presupuesto / Pedido	sale.order	ir.actions.report	sale.report_saleorder	PDF
<input type="checkbox"/>	Recibo de Pago	account.payment	ir.actions.report	account.report_payment_receipt	PDF
<input type="checkbox"/>	Resultado de integridad de hash PDF	res.company	ir.actions.report	account.report_hash_integrity	PDF
<input type="checkbox"/>	Tarifa	product.product	ir.actions.report	product.report_pricelist	PDF
<input type="checkbox"/>	Vista previa del Reporte Externo	res.company	ir.actions.report	web.preview_externalreport	PDF
<input type="checkbox"/>	Vista previa del Reporte Interno	res.company	ir.actions.report	web.preview_internalreport	PDF

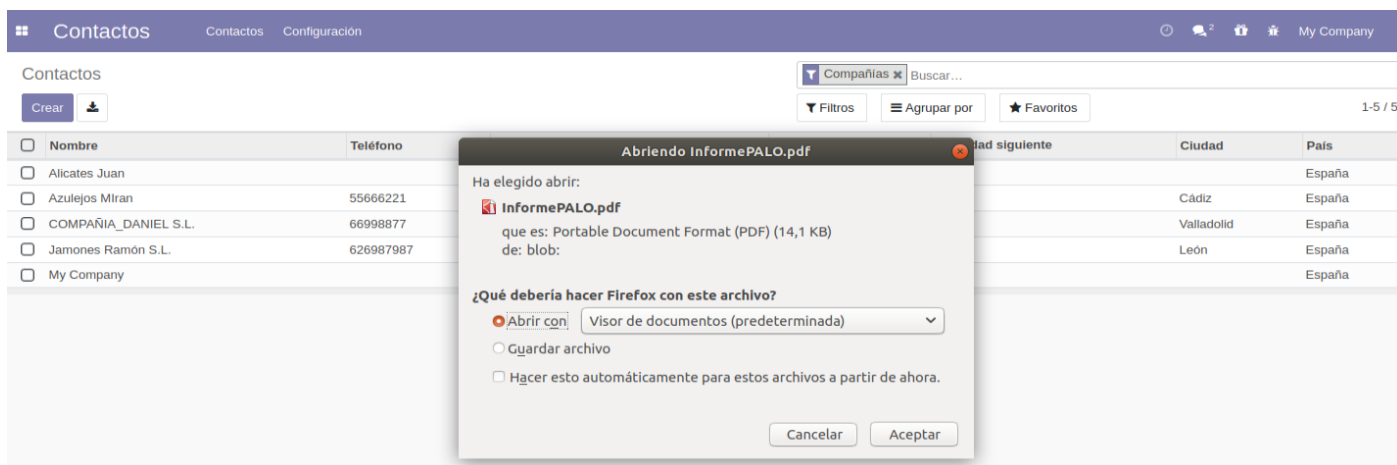
Podemos ver remarcado que ya se ha incluido la nueva acción

Ya solo nos queda ver que realmente se hace el informe de la manera y con los datos que nosotros queremos. Para ello nos vamos al menú de contactos, filtramos por empresas y seleccionamos la vista de “lista”:

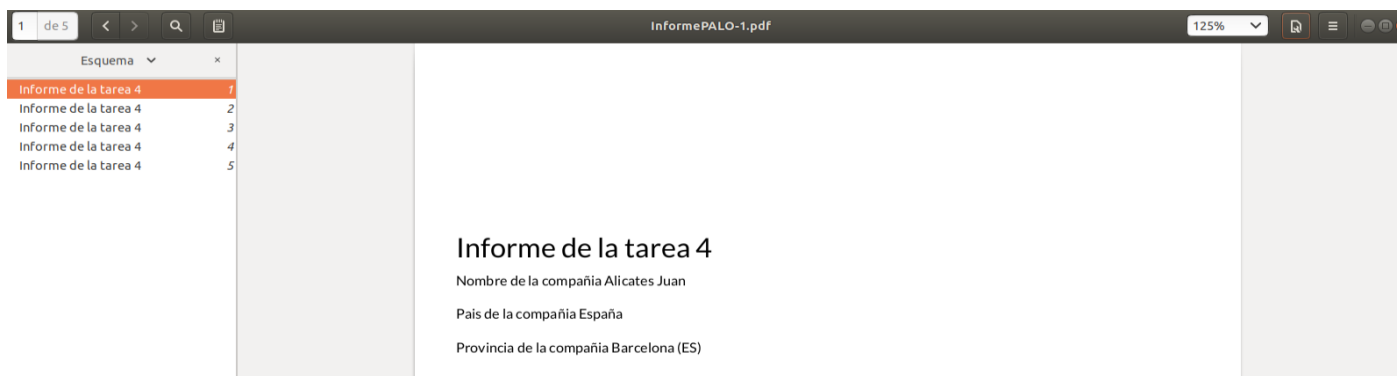


Al seleccionar las líneas podemos ver que nos ha aparecido un nuevo menú llamado “imprimir”

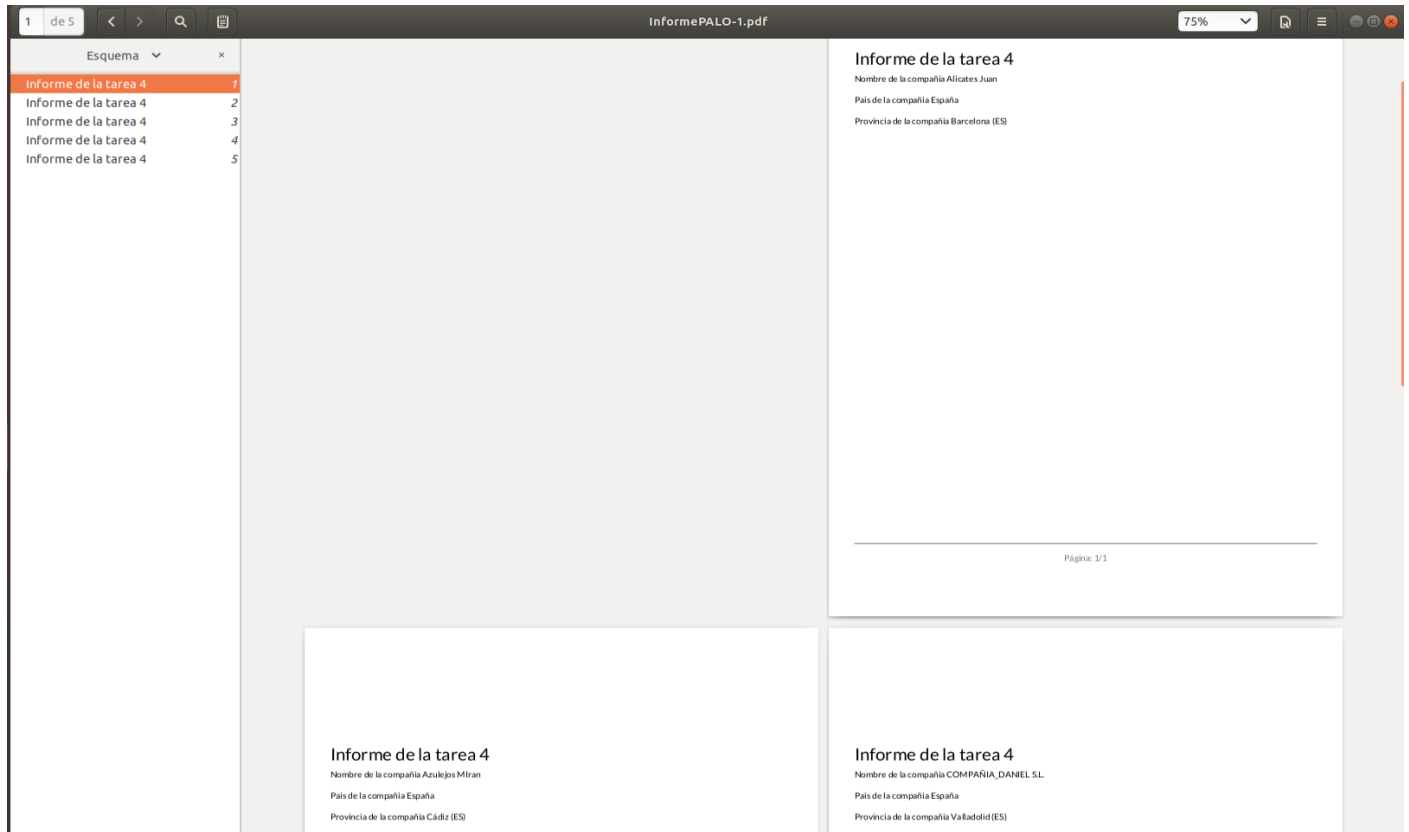
Dentro de imprimir y teniendo ya las compañías de las que queremos informe seleccionadas, pulsamos en el submenú “informePALO”:



Nos pregunta si queremos guardar o abrir el informe, seleccionamos la opción abrir porque solo queremos ver que se ha generado bien el informe:



Vemos como quedaría la presentación junto con los datos que nos interesa mostrar



Comprobamos que se han generado 5 hojas de informe, una para cada una de las 5 empresas que habíamos seleccionado