**SISTEMAS DE GESTIÓN EMPRESARIAL**

**TEMA 5, Tarea**

**Desarrollo de componentes**

**Contenido**

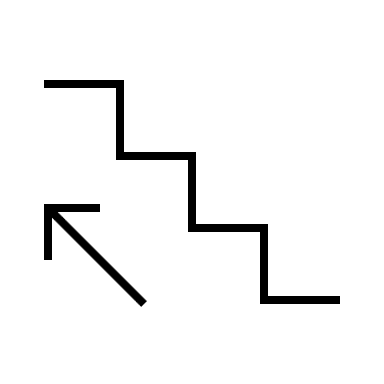
* [Enunciado](#enunciado)
* [Previo](#previo)
* [Ejercicios](#ejercicio1)
  + [Ejercicio 1](#ejercicio1)
  + [Ejercicio 2](#ejercicio2)
    - [Importar CSV Clientes](#importarClientes)
    - [Importar CSV Productos](#importarProductos)
  + [Ejercicio 3](#ejercicio3)
  + [Ejercicio 4](#ejercicio4)
* [Anexo](#anexo)
  + [Archivo CSV productos](#csvProductos)
  + [Cliente/Proveedor](#anexocliente)
  + [Crear campos](#anexoCrearCampos)
  + [Error de acceso por permisos (odoo.exceptions.AccessError)](#anexoErrorPermisos)

**Notas:**

Los ejercicios presentan algunas dificultades debidas a la desactualización.  
El ejercicio lo hago con Odoo 14 (2020) (actualmente está en uso Odoo 15, 2021) y la última versión “compatible” con el ejercicio es Odoo 12 (2018).

Problemas:

* Ejercicio 2:
  + Cambios en la estructura del objeto res\_partner:
    - Cliente /proveedor: no existe tal diferencia entre los contactos. Se eliminó del formulario de contactos la posibilidad de marcarlo como cliente o proveedor.
      * Solución: usar los campos supplier\_rank y customer\_rank.
    - Fax: no existe el campo para el fax. Entiendo que se considera tecnología obsoleta.
      * Solución: no se integra el campo “Fax”.
    - Tipo de compañía: Cualquier contacto es de tipo “Contacto”. Al importar los clientes es posible que estos no sean una compañía.
      * Solución: incorporar el campo “is\_company”.

**Enunciado [](#contenido)**

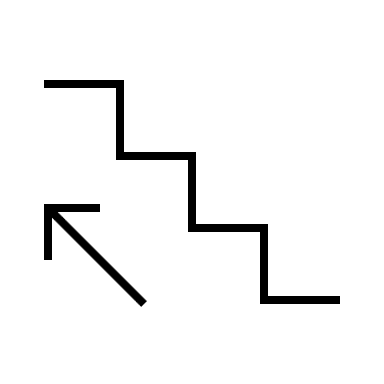
La empresa **Datalab** se ha puesto en contacto con **BK Programación** para solicitarles un programa de gestión empresarial. La situación actual es que disponen de una aplicación principal para todo lo relativo a la gestión financiera de su empresa, almacén y nóminas. El área Comercial trabaja con hojas de cálculo y sistemas gestores de bases de datos no conectados con la aplicación principal. Además, la aplicación principal es un software privativo que no puede modificarse para adaptarse a ciertas necesidades que tiene la empresa. Por todo ello, en **Datalab** han tomado la decisión de implantar un ERP.

Imagina que formas parte del equipo de proyecto de **BK Programación** y tienes que realizar la implantación de un ERP en **Datalab**.

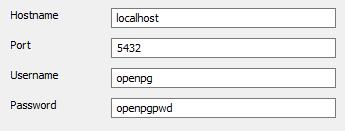
Ten en cuenta que en el nombre de los objetos **AA** y **BB** corresponden a las dos primeras letras de tu primer y segundo apellido, respectivamente.

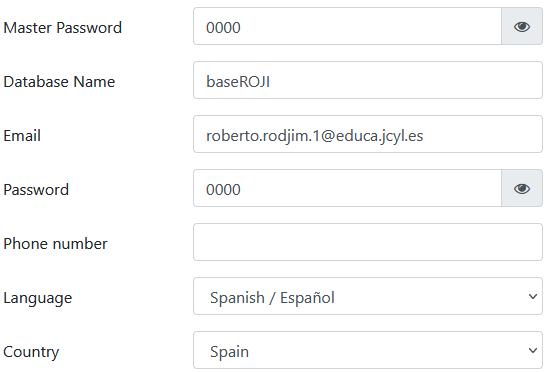
Realiza las operaciones necesarias para:

1. Crear una aplicación ERP llamada **baseAABB**, que disponga de todos los módulos necesarios para operar en una pequeña y mediana empresa española.
2. Realizar la migración de datos de clientes del antiguo sistema al nuevo ERP. Nos han pedido que hagamos las primeras pruebas de importación de clientes y productos, para ello tenemos que:
   * Crear un archivo CSV con los datos de tres clientes arbitrarios, e importar esos registros a la nueva aplicación ERP. Los datos a traspasar de cada cliente son:
     + Código del cliente.
     + Nombre del cliente.
     + Nombre comercial.
     + Calle.
     + Ciudad.
     + Provincia.
     + Código postal.
     + Teléfono.
     + Fax.
     + Email.
     + País.
     + Si es cliente o proveedor.
     + Web.
     + Categoría de la empresa.
   * Dado el siguiente archivo CSV con datos de productos, importar esos registros a la nueva aplicación ERP, haciendo modificaciones en el archivo si fuera necesario para que la importación se pueda realizar.   
     [Archivo CSV de productos.](https://fpdistancia.educa.jcyl.es/pluginfile.php/240529/mod_assign/intro/SGE04/SGE04_RecursosTarea/SGE04_AUX_SOLUCIONTAREA_25_datosproductos.csv) (0.01 MB)
3. Nos hemos dado cuenta de que existen incongruencias en los datos importados. Un compañero experto en bases de datos va a actualizar masivamente los datos incorrectos, para ello nos ha pedido que creemos un procedimiento almacenado que muestre el nombre del cliente, la provincia y el país al que pertenece dicho cliente.
4. Crear un informe personalizado llamado **InformeAABB** con dos vistas (árbol y formulario) que muestre los siguientes datos:
   * Nombre de la empresa.
   * Provincia.
   * País.

**Previo [](#contenido)**

Instalación de Odoo 14 en Windows 10 y Ubuntu 18.04 (Virtual Box) sin datos de ejemplo.





Aplicación: baseROJI

Empresa: Datalab, una empresa pequeña.

Aplicaciones que ya tienen:

.- Gestión financiera, almacén y nóminas.  
.- Hojas de cálculo.  
.- Bases de datos sin interconexión.  
.- El software es privativo.

**Selección del ERP**

**Tipos de empresa y necesidades de ERP**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CRM  PROVEEDORES | COMPRAS |  | VENTAS  ALMACÉN | PRODUCTOS | PROYECTOS |  | TPV  CONTABILIDAD |  | RR.HH. |  |
| PYME | **O** | **O** | **O** | **O** | **O** | **O** |  |  |  |  |
| Servicios |  |  |  |  |  |  | O |  |  |  |
| Comercio |  |  |  |  |  |  |  | O |  |  |
| Restaurantes |  |  |  |  |  |  |  | O |  |  |
| Administración | O |  | O |  | O |  | O |  | O | O |
| Televenta | O |  |  |  |  |  |  |  |  |  |

**Módulos necesarios**

|  |  |  |  |
| --- | --- | --- | --- |
| **ÁREA** | **DATOS** | **LIBRE** | **MÓDULO** |
| Gestión de los clientes | Son necesarios los datos de contacto para la comunicación. | SÍ | CRM |
| Productos | Controlan el stock y se relacionan con la cadena de suministro.  Se relaciona con Contabilidad y envíos. | SÍ | Inventario |
| Finanzas | Incluye facturación, pero es de pago. | NO | Contabilidad |
| Finanzas | Se relaciona con clientes, ventas, contabilidad, … | SÍ | Facturación |
| Nóminas | Necesitamos los datos de los empleados. Módulo para descargar vía web. <https://apps.odoo.com/apps/6.1/l10n_es_hr_nominas/> | SÍ | Nóminas |
| Empleados | Es necesario para las nóminas | SÍ | Empleados |
| Ventas | Se relaciona con CRM, contabilidad y almacén | SÍ | Ventas |
| Compras | Tiene relación con contabilidad, inventario y facturación. | SÍ | Compra |

**Informes necesarios**

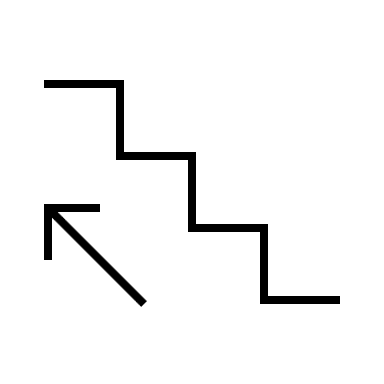
|  |  |  |
| --- | --- | --- |
| **ÁREA** | **INFORMACIÓN** | **VISTA** |
| Proveedores | - Nombre de la empresa. - Provincia - País | - Árbol - Formulario |

ERP seleccionado:

El software seleccionado para instalar es **Odoo**, por los siguientes motivos:

* Software gratuito en gran parte de sus módulos.
* Fácilmente modificable.
* Código abierto que permite las modificaciones.
* Dispone de todos los módulos necesarios.

Se instalará como un sistema *monopuesto*.

**Ejercicio 1 [](#contenido)**

*Crear una aplicación ERP llamada* ***baseAABB****, que disponga de todos los módulos necesarios para operar en una pequeña y mediana empresa española.*

Esta parte corresponde a la fase se selección del sistema y los módulos que debemos usar.  
Instalamos los módulos necesarios y realizamos las adaptaciones necesarias.

Después de haber seleccionado Odoo 14 como el ERP que vamos a utilizar, pasamos a instalar los módulos que se consideran mínimos para una PYME:

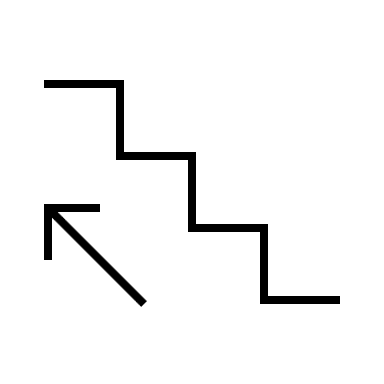
.- CRM   
.- Proveedores (Compras)  
.- Compras   
.- Ventas   
.- Almacén (Inventario)  
.- Productos (Inventario)

Además, para poder trabajar con los datos que Datalab ya manejaba, debemos instalar:

.- Facturación / Contabilidad  
.- Nóminas (Empleados)  
.- Empleados

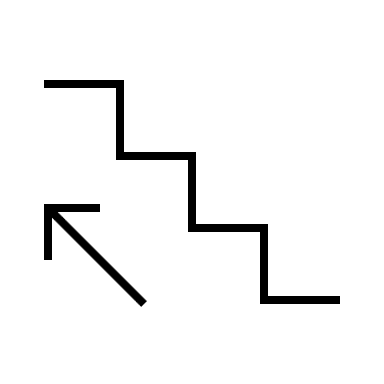
Todos los módulos instalados están precargados en la versión Odoo 14.  
El módulo de Facturación / Contabilidad es un módulo base.  
El módulo de nóminas seleccionado en el análisis previo es para la versión Odoo 6.1, para la actual (14), las nóminas se gestionan desde el módulo Empleados.   
Existen módulos de gestión de nóminas independientes, pero de pago, pero Datalab exige software gratuito.

Referencias:  
<https://ateneolab.com/blog/odoo-1/novedades-de-odoo-14-1>  
<https://www.youtube.com/watch?v=019m06Upw-0>

**Ejercicio 2 [](#contenido)**

Realizar la migración de datos de clientes del antiguo sistema al nuevo ERP. Nos han pedido que hagamos las primeras pruebas de importación de clientes y productos, para ello tenemos que:

1. Crear un archivo CSV con los datos de tres clientes arbitrarios, e importar esos registros a la nueva aplicación ERP. Los datos a traspasar de cada cliente son:   
   * Código del cliente.
   * Nombre del cliente.
   * Nombre comercial.
   * Calle.
   * Ciudad.
   * Provincia.
   * Código postal.
   * Teléfono.
   * Fax.
   * Email.
   * País.
   * Si es cliente o proveedor.
   * Web.
   * Categoría de la empresa
2. Dado el siguiente archivo CSV con datos de productos, importar esos registros a la nueva aplicación ERP, haciendo modificaciones en el archivo si fuera necesario para que la importación se pueda realizar.

Crear CSV de clientes **[](#contenido)**

Debemos comprobar los datos y sus tipos en el CRM.

El modelos usado es res.partner, puede verse en los parámetros del formulario

**Datos necesarios y sus tipos**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tipo** | **Nombre del campo** | **Dependencia** |
| Código de cliente | integer | user\_id | Se supone que es el id propio del cliente. |
| Nombre del cliente | character | name |  |
| Nombre comercial | character | comercial\_company\_name | res.company |
| Calle | character | street |  |
| Ciudad | character | city |  |
| Provincia | integer | state\_id | res.country.state |
| Código postal | character | zip |  |
| Teléfono | character | phone |  |
| Fax | --- | --- | El campo fax no existe en Odoo 14, es tecnología obsoleta. |
| Email | character | email |  |
| País | integer | country\_id | res.country |
| Cliente | integer | customer\_rank | [Ver anexo Cliente/Proveedor](#anexocliente) |
| Proveedor | integer | supplier\_rank | [Ver anexo Cliente/Proveedor](#anexocliente) |
| Web | character | website |  |
| Es compañía | boolean | is\_company |  |
| Categoría de la empresa | integer | industry\_id | res.partner.industry |

Tabla 1 Campos implicados en la tabla res.partner

Creamos el archivo CSV:

ID externo,Nombre,Nombre de la compañía,Calle,Ciudad,Estado/ID,C.P.,Teléfono,correo electrónico,País/ID,Rango del Cliente,Rango de proveedor,Enlace a página web,Es una compañia,Sector / Id. de la BD

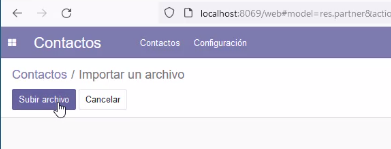
1247,Creaciones del Norte,Creaciones del Norte S.A.,Calle primera s/n,Villanueva,433,65676,787232323,info@creanorte.esd,68,1,0,www.creanorte.esd,VERDADERO,10

1347,Criações do Norte,Criações do Norte,Rua Bacalao a bras nº 345,Coimbra,253,76543,525525525,info@nortecreations.ptd,183,1,0,www.nortcreations.ptd,VERDADERO,13

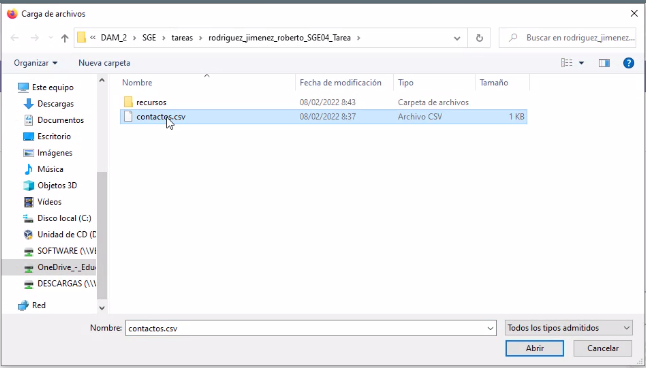
1447,Piezas los Cuñaos,Piezas los Cuñaos C.B.,Paseo al sol nº 7,Aldeavieja,451,98984,999777666,info@piezascunao.esd,68,0,1,www.piezascunaos.esd,VERDADERO,3



**1** Desde la vista de contactos nos vamos a Favoritos 🡪 Importar registros

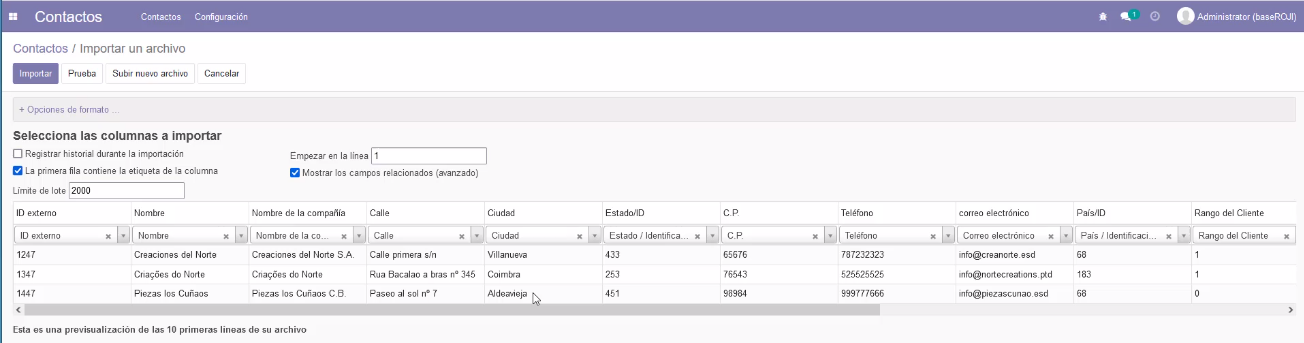


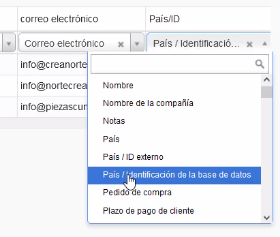
**2** En la ventana abierta, pulsamos en “Subir archivo”.



**3** Abrimos el archivo csv.

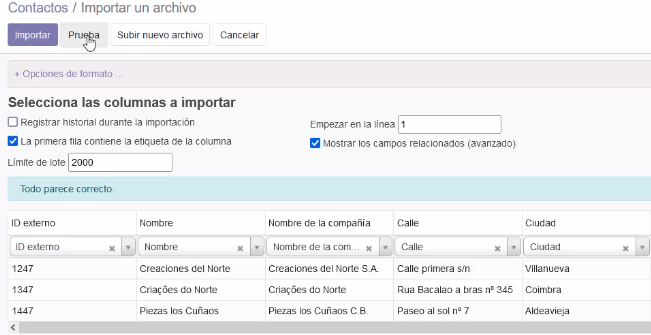
Ahora queda cargado el archivo en la aplicación, pero aún hay que realizar algunos retoques.



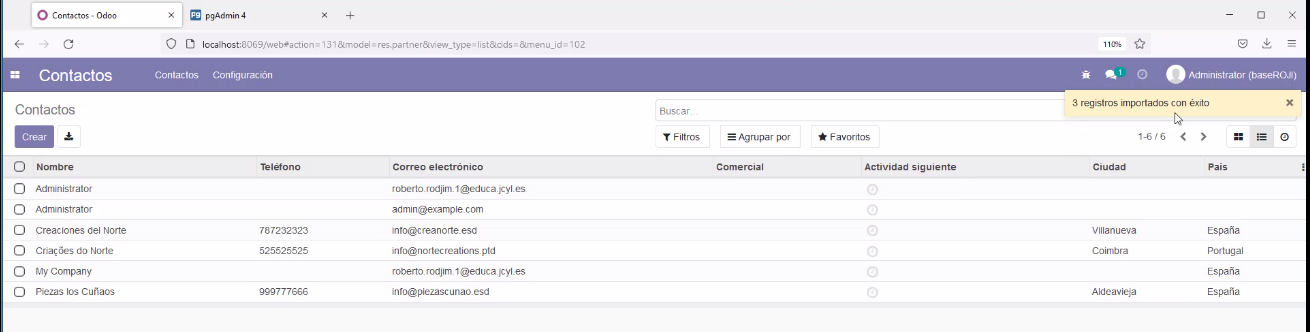
****Por ejemplo:

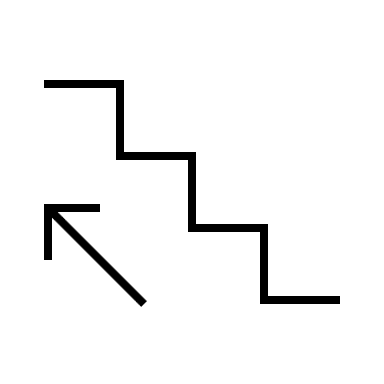
No concuerda el nombre del campo País con el que le hemos dado, ya que en los campos relacionados se admiten tres opciones: el nombre del país directamente, el id de una tabla externa, o el id de la tabla de países de Odoo, que es el que hemos dado.

Tras ajustar los nombres de los campos, ejecutamos una prueba previa a la importación definitiva.



Si el resultado es correcto, podemos importar.



Importar CSV de Productos **[](#contenido)**

Dado el siguiente archivo CSV con datos de productos, importar esos registros a la nueva aplicación ERP, haciendo modificaciones en el archivo si fuera necesario para que la importación se pueda realizar.

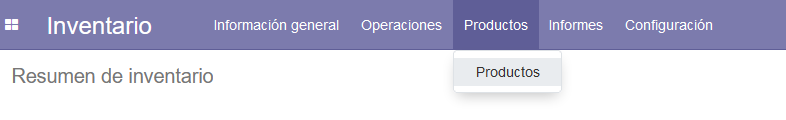
"Código";"Nombre";"Categoría";"Descripción";"Método abastecimiento";"Método coste";"Método suministro";"Precio coste";"Precio de venta";"Puede ser comprado";"Puede ser vendido";"Tipo de producto";"Empresas/Empresa";"Impuestos cliente";"Impuestos proveedor";"UdM de compra";"UdM por defecto"

"cod1";"";"cosmetica";"descripcion producto1";"make\_to\_order";"standard";"buy";0;0;"True";"True";"product";"Empresa demo";"IVA 7%";"7% IVA Soportado (operaciones corrientes)";"Kg.";"Kg."

"cod2";"nombreproducto2";"cosmetica";"descripcion producto2";"make\_to\_stock";"standard";"buy";0;0;"True";"True";"product";"Empresa demo";"IVA 7%";"7% IVA Soportado (operaciones corrientes)";"Kg.";"Kg."

"cod3";"nombreproducto3";"cosmetica";"descripcion producto3";"make\_to\_order";"standard";"buy";0;0;"True";"True";"product";"Empresa demo";"IVA 7%";"7% IVA Soportado (operaciones corrientes)";"Kg.";"Kg."

Comprobamos el modelo que se usa para los productos desde Inventario 🡪 Productos 🡪Productos



Probamos con product.template, que son parte de product.product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Tipo** | **Nombre del campo** | **Dependencia** | **Observaciones.** |
| Código | character | default\_code *Referencia interna* |  |  |
| Nombre | character | name *Nombre* |  |  |
| Categoría | integer | categ\_id *Categoría de producto (id)* | product.category |  |
| Descripción | text | description *Descripción* |  | Es “notas internas” en la vista. |
| M. abastecimiento | text | x\_product\_sourcing\_method *Método de abastecimiento* |  | Campo personalizado de solo lectura. |
| Método coste | select | cost\_method  *Método de coste* | category\_id | Se define en “Categoría del producto” |
| Método suministro | text | x\_product\_supply\_method *Método de suministro* |  | Campo personalizado de solo lectura. |
| Precio coste | float | price *Precio* |  |  |
| Precio venta | float | list\_price *Precio de venta* |  |  |
| Puede ser compr. | boolean | purchase\_ok *Puede ser comprado* |  |  |
| Puede ser vendido | boolean | sale\_ok *Puede ser vendido* |  |  |
| Tipo de producto | select | type *Tipo de producto* |  | ‘consu’,’service’,’product’ |
| Empresa | integer | company\_id *Compañía* | res.company |  |
| Impuestos cliente | integer | taxes\_id *Impuestos del cliente* | account.tax |  |
| Impuestos prov. | integer | supplier\_taxes\_id *Impuestos del proveedor* | account.tax |  |
| UdM de compra | integer | uom\_po\_id *Unidad de medida compra* | uom.uom |  |
| UdM de venta | integer | uom\_id *Unidad de medida* | uom.uom |  |

Tabla 2 Campos implicados en el modelo product.template

“Métodos de coste” es de sólo lectura y esta propiedad debe cambiarse desde el código Python al tratarse de un campo base.  
Estos valores están relacionados con *product\_category*.

Referencia <https://www.youtube.com/watch?v=m9jUeUi2OcI>

No se va a importar este campo.

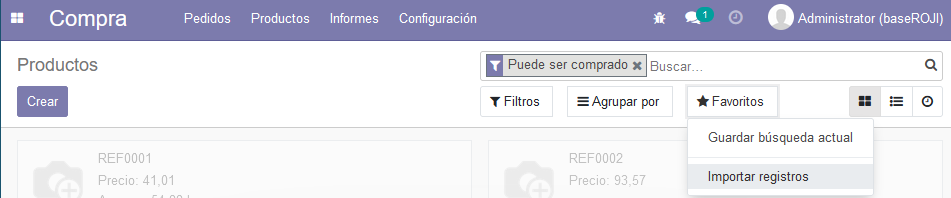
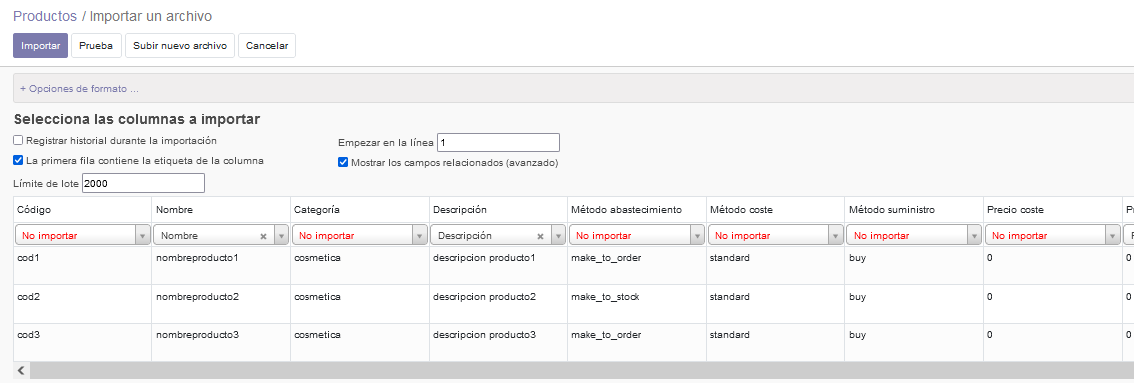
Existen dos datos para los que no hay campo (Odoo 14):

* Método de abastecimiento.
* Método de suministro.

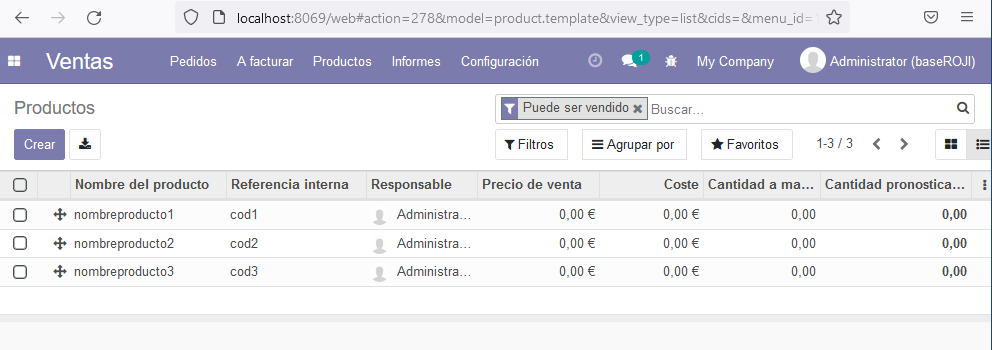
Para estos datos vamos a crear los campos necesarios en el modelo product\_template, con los nombres que pueden verse en la [tabla 2](#tabla2) (ver [Crear Campos](#anexoCrearCampos)).

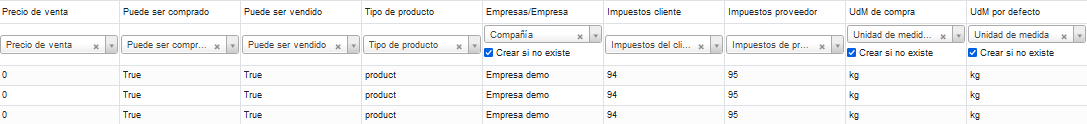
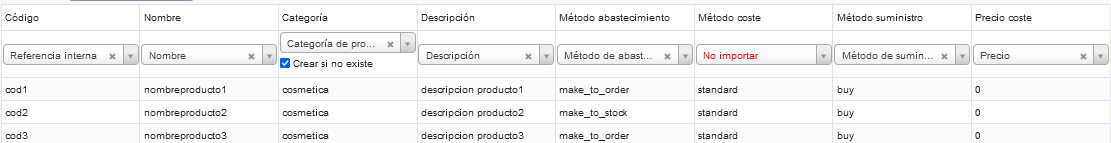
Una vez que tenemos todos los campos del modelo producto.producto, podemos cargar el archivo CSV.

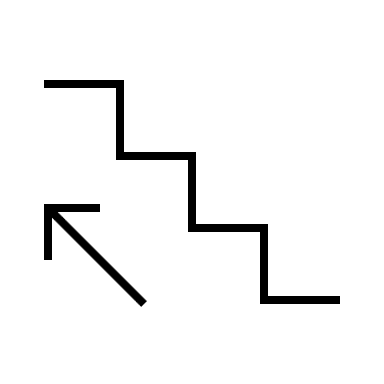
La importación del archivo csv se hace desde *Compra🡪Productos🡪Productos* y desde *Favoritos* seleccionamos *Importar registros*.



Ahora debemos relacionar los nombres de los campos teniendo en cuanta que algunos deben apuntar a los id de otras tablas y otros deben crearse en el momento de ser importados.



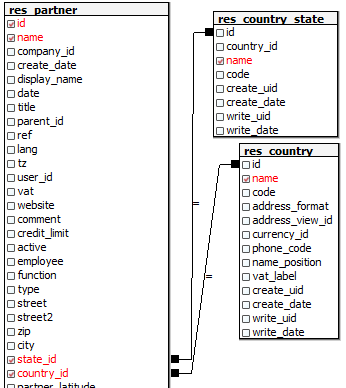


**Ejercicio 3 [](#contenido)**

*Nos hemos dado cuenta de que existen incongruencias en los datos importados. Un compañero experto en bases de datos va a actualizar masivamente los datos incorrectos, para ello nos ha pedido que creemos un procedimiento almacenado que muestre el nombre del cliente, la provincia y el país al que pertenece dicho cliente.*

Para obtener los nombres de la provincia y del país necesitamos los id.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **TIPO** | **CAMPO** | **CAMPO RELACIONADO** |
| Provincia | integer | state\_id | res\_country.id |
| País | integer | country\_id | res\_country\_state.id |

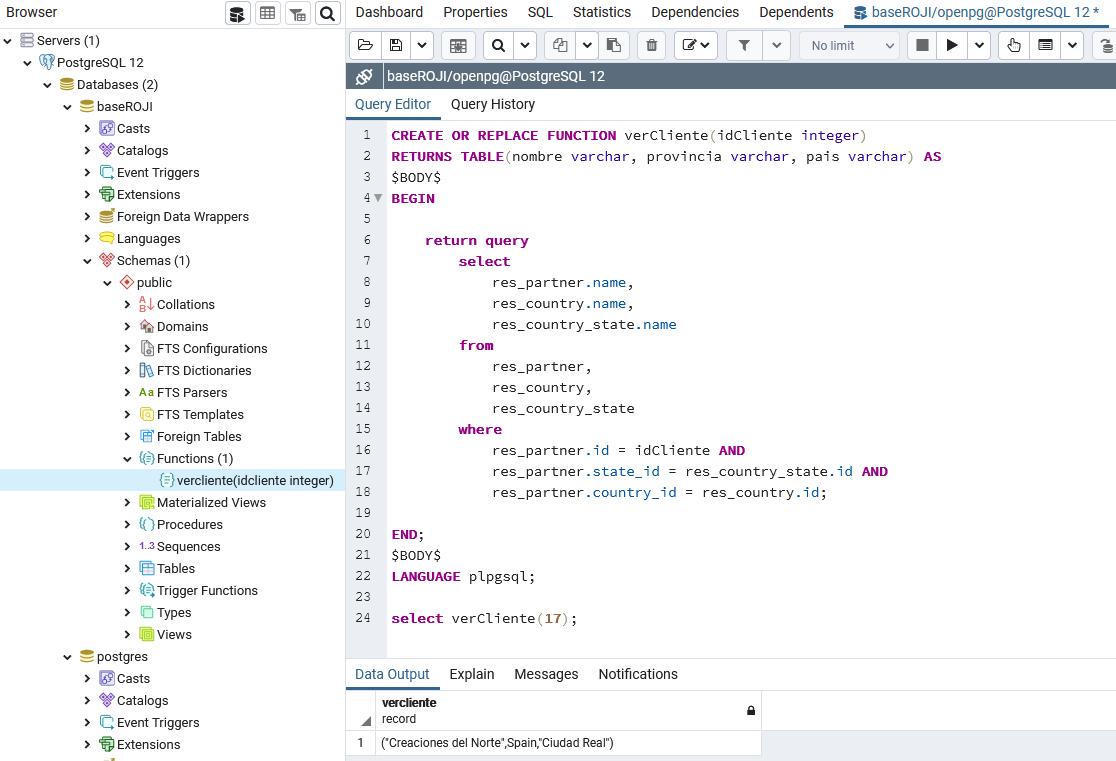
****Consulta

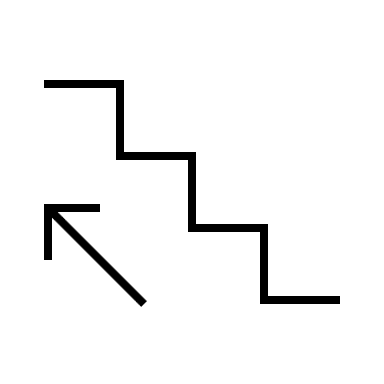
SELECT   
 res\_partner.name,   
 res\_country.name,  
 res\_country\_state.name  
FROM   
 public.res\_partner,   
 public.res\_country,   
 public.res\_country\_state  
WHERE   
 res\_partner.id = idCliente AND  
 res\_partner.state\_id = res\_country\_state.id AND  
 res\_partner.country\_id = res\_country.id;

Procedimiento

CREATE OR REPLACE FUNCTION verClienteplpgsql(idCliente integer)  
RETURNS TABLE(nombre varchar, provincia varchar, pais varchar) AS  
$BODY$  
BEGIN  
 return query  
 select   
 res\_partner.name,   
 res\_country.name,   
 res\_country\_state.name  
 from   
 res\_partner,  
 res\_country,  
 res\_country\_state  
 where   
 res\_partner.id = idCliente AND  
 res\_partner.state\_id = res\_country\_state.id AND  
 res\_partner.country\_id = res\_country.id;

END;  
$BODY$  
LANGUAGE plpgsql;

****

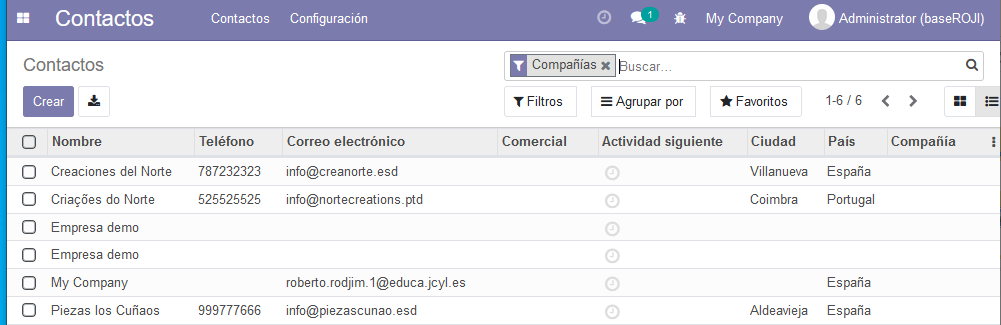
**Ejercicio 4 [](#contenido)**

*Crear un informe personalizado llamado* ***InformeAABB*** *con dos vistas (árbol y formulario) que muestre los siguientes datos:*

* + *Nombre de la empresa.*
  + *Provincia.*
  + *País.*

Tenemos que mostrar un informe de las empresas registradas, lo que significa que debemos ver los contactos filtrados por *compañía*.

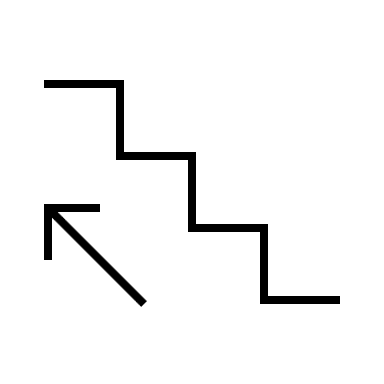
El modelo sobre el que se trabaja es *res.partner*.

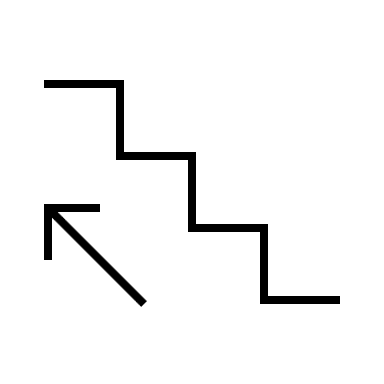


También habrá que definir una plantilla.

No hay forma: todos los tutoriales se basan en la creación de aplicaciones.  
El módulo base\_report\_designer se usa hasta Odoo 7.

Probamos con JasperReport y JasperSoft Studio 6.13  
Lo dejo por imposible, no se encuentra información suficiente.  
  
La única opción para la que encuentro suficiente información es usando wkhtmltopdf, pero esta herramienta lo que hace es convertir el HTML en Pdf, no genera informes, por lo que no creo que sea válido.

**Anexo [](#contenido)**

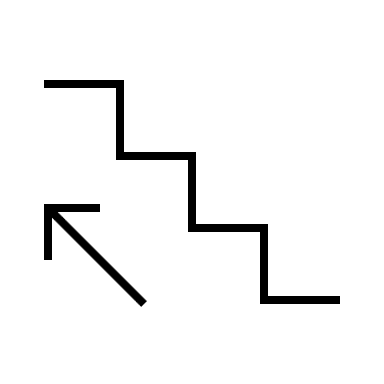
**Archivo** **CSV de productos [](#contenido)**

"Código";"Nombre";"Categoría";"Descripción";"Método abastecimiento";"Método coste";"Método suministro";"Precio coste";"Precio de venta";"Puede ser comprado";"Puede ser vendido";"Tipo de producto";"Empresas/Empresa";"Impuestos cliente";"Impuestos proveedor";"UdM de compra";"UdM por defecto"

"cod1";"nombreproducto1";"cosmetica";"descripcion producto1";"make\_to\_order";"standard";"buy";0;0;"True";"True";"product";"Empresa demo";"IVA 7%";"7% IVA Soportado (operaciones corrientes)";"Kg.";"Kg."

"cod2";"nombreproducto2";"cosmetica";"descripcion producto2";"make\_to\_stock";"standard";"buy";0;0;"True";"True";"product";"Empresa demo";"IVA 7%";"7% IVA Soportado (operaciones corrientes)";"Kg.";"Kg."

"cod3";"nombreproducto3";"cosmetica";"descripcion producto3";"make\_to\_order";"standard";"buy";0;0;"True";"True";"product";"Empresa demo";"IVA 7%";"7% IVA Soportado (operaciones corrientes)";"Kg.";"Kg."

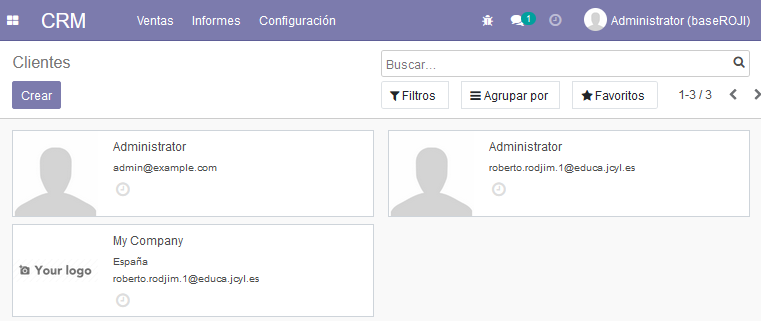
**Cliente / proveedor [](#contenido)**

La opción de añadir la distinción entre cliente y proveedor se quitó de Odoo en la versión 13.

Aparentemente no hay forma de saber si un contacto o una empresa es un cliente o un proveedor.

En los modelos res.partner y res.partner.company no figura ningún campo que guarde la información de si es cliente o proveedor.

Cualquier registro guardado en res.partner se muestra como “contacto”, no importa si es cliente o proveedor, de hecho, puede ser ambos, o ninguno, simultáneamente.



La forma de crear clientes y proveedores de forma directa es a través de los módulos

* *Ventas* 🡪 *Pedidos* 🡪 *Clientes* 🡪 *Crear*
* *Compras* 🡪 *Pedidos* 🡪 *Clientes* 🡪 *Crear*

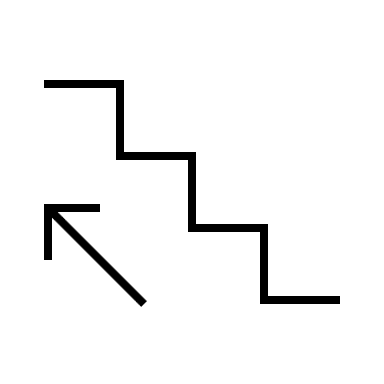


¿Y cómo se identifican? Odoo marca los campos res\_partner.suppler\_rank y res\_partner.customer\_rank con “1” o un número mayor si se corresponde con proveedor o cliente respectivamente.

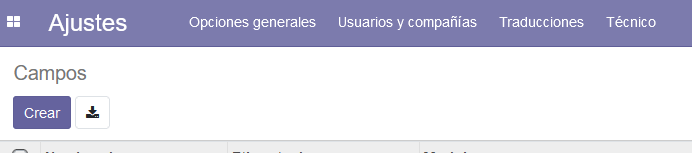
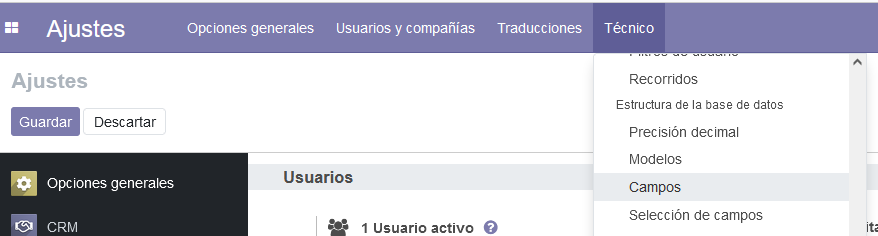
Si ambos campos tienen un valor mayor que 0, pues el contacto es tanto cliente como proveedor.

Si los dos campos son 0, pues no es ni cliente ni proveedor.

Estos valores se actualizan automáticamente cuando el contacto realiza alguna transacción, pasando a ser cliente o proveedor automáticamente.

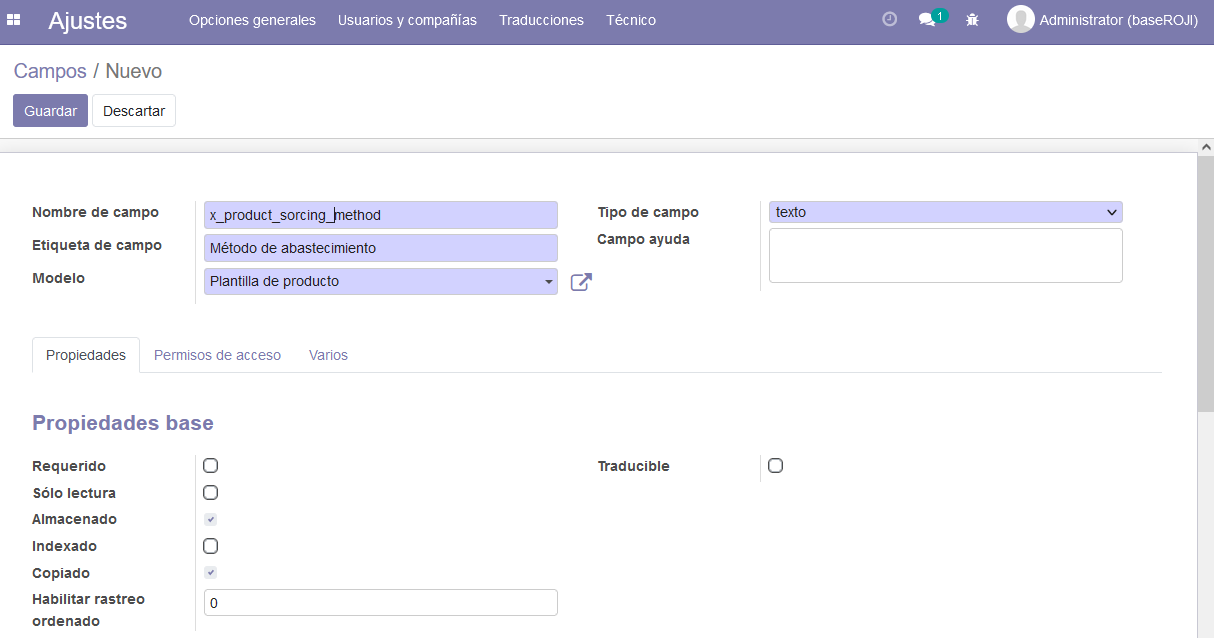
**Crear campos [](#contenido)**

La forma de los campos es idéntica para los dos tipos, por lo que sólo se muestra uno de ellos.



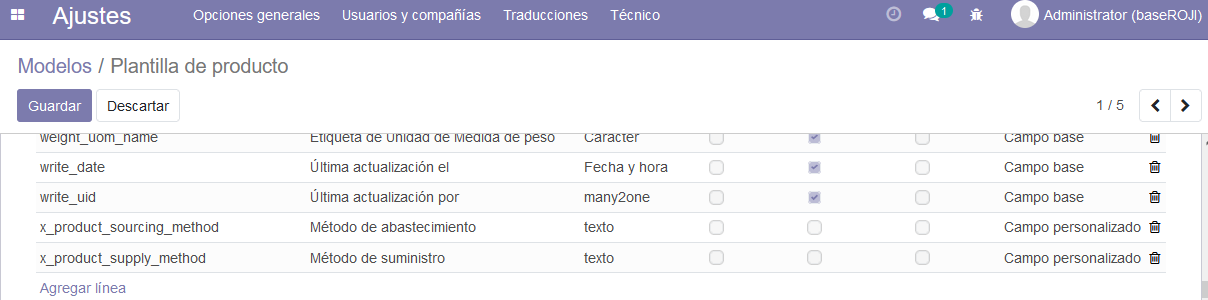
**2** Crear.

**1** Desde Ajustes🡪Técnico🡪Campos se accede al formulario para crear los campos.

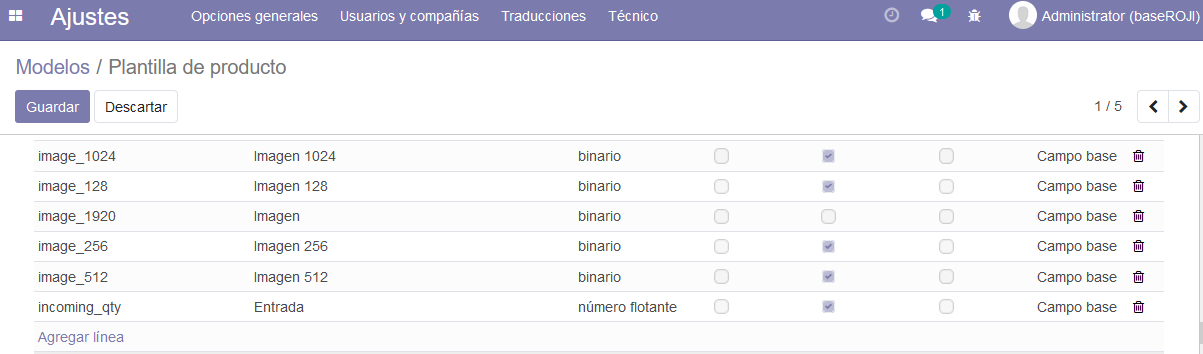


Rellenamos los campos necesarios.

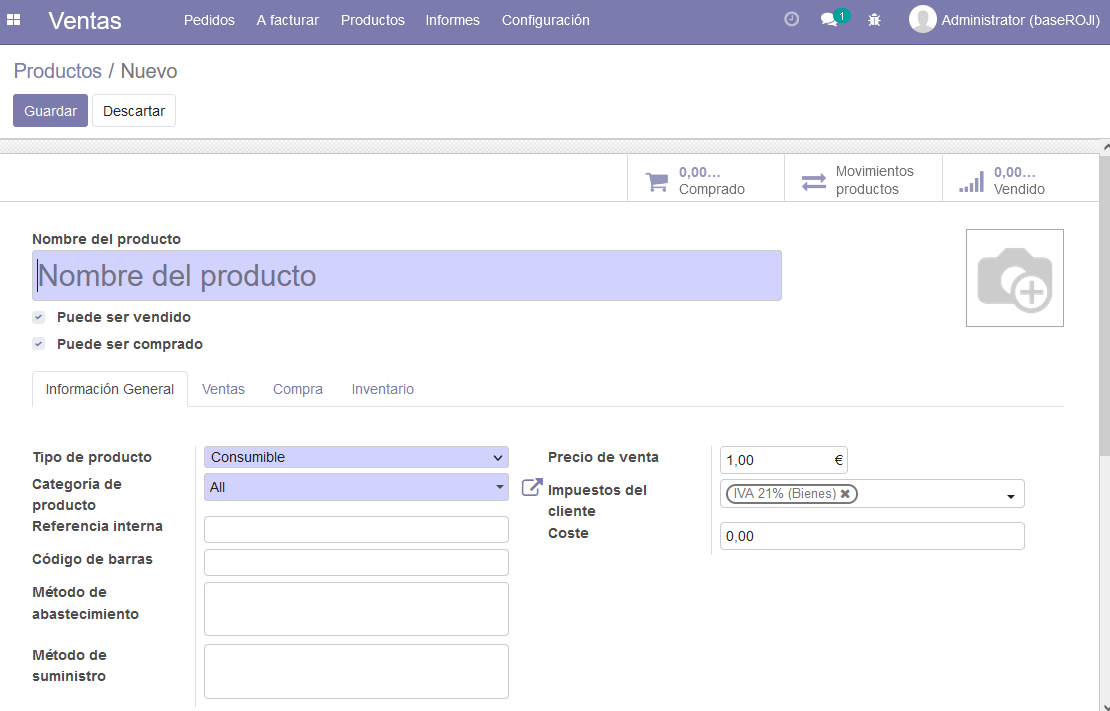
El nombre del modelo es product\_template, pero se muestra como “Plantilla de producto”.

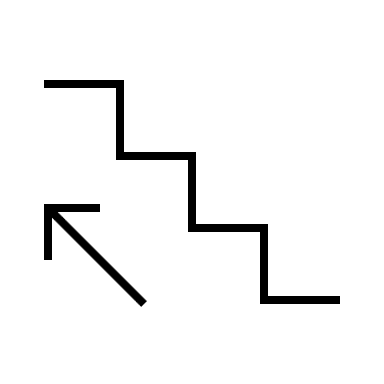
Vemos los campos creados  
  


Otra opción para crear los campos es desde la edición del modelo y en la pestaña “Campos” ir a “Agregar línea”, pero esta opción no me ha funcionado.  
  
  
  
  
  
  
  
  
  
  
  
Introducimos los campos en la vista product.template.product.form.

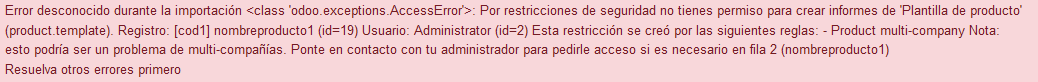


Vista final



**Error de acceso por permisos (odoo.exceptions.AccessError) [](#contenido)**

Al ejecutar la importación del archivo csv, aparece el siguiente mensaje de error:



Parece ser que no cumplimos las condiciones de permiso para la regla creada.  
Accedemos al modelo *product\_template* para comprobar cómo es esta regla y damos permisos a todas aplicaciones en las que intervenga el administrador.

