

Taller Flexygo

Ejercicios Lowcode Parte 2

0. Video presentación: https://youtu.be/BbV_AJ4Vksl

Ejercicio 1 Dependencias entre propiedades

1. Añadir nuevos campos al formulario de edición del cliente

- FechaBloqueo: DateTime
- MotivoBloqueo: text 1000

Step 2 - Object Properties

Save >

Relationship of dependencies Add fields

Clientes

Code #

Nombre: * T

Razón social: * T

Dirección: T

Código postal: T

Teléfono: T

Email: T

NIF: T

Situación

Estado: 0

Empleado asignado: T

Permitir enviar información: 1

Fecha de alta: dd/mm/aaaa {{currentDate}}

Antigüedad: dd/mm/aaaa

Add properties

☒ ☐

☐ FechaBloqueo

Add detached property

Property Name Property Label +

Create new table fields

FechaBloqueo DateTime Clientes +

Añadimos un nuevo campo con el asistente de creación de formularios con los datos indicados en el enunciado en el objeto de clientes.

Clientes

Code #

Nombre: * T

Razón social: * T

Dirección: T

Código postal: T

Teléfono: T

Email: T

NIF: T

Situación

Estado: 0

Empleado asignado: T

Permitir enviar información: 1

Fecha de alta: dd/mm/aaaa {{currentDate}}

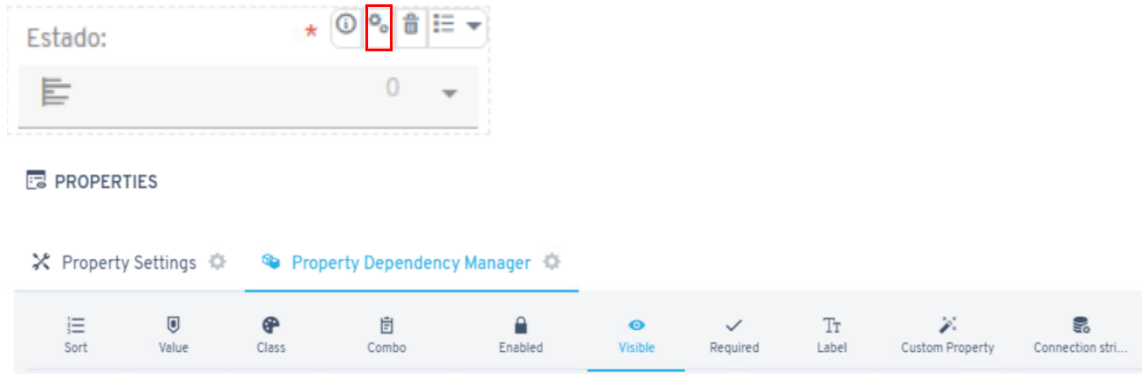
Antigüedad: dd/mm/aaaa

Fecha de bloqueo: dd/mm/aaaa --:--

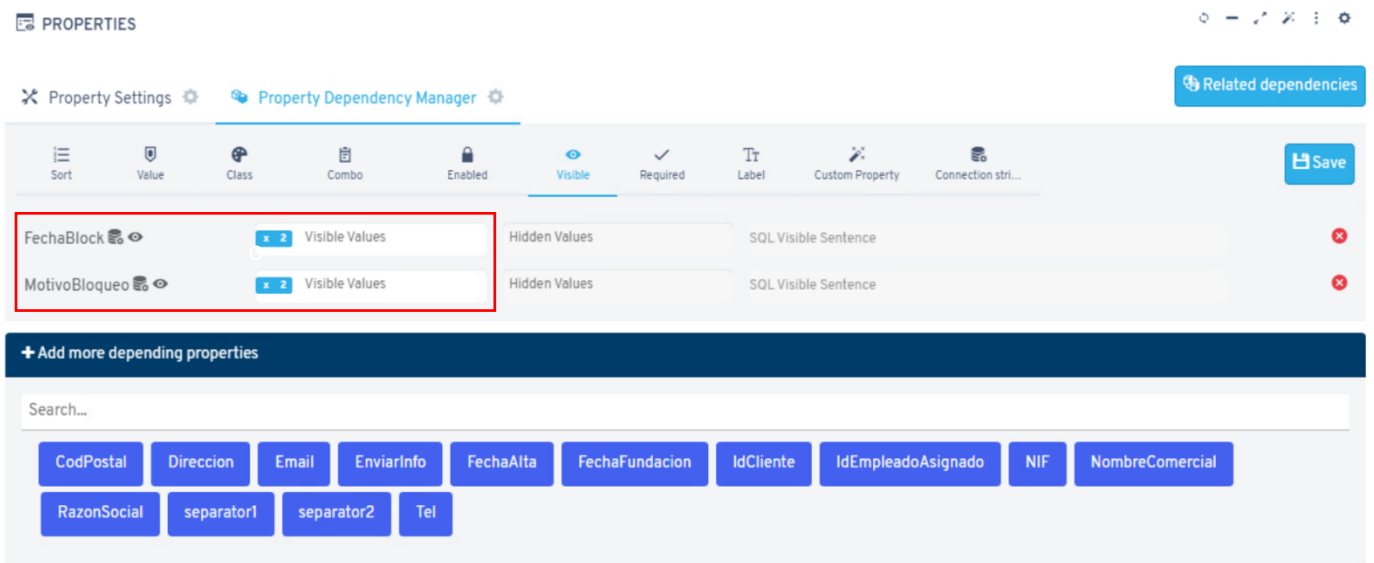
Motivo de bloqueo: T

2. Hacer que los campos motivo de bloqueo y fecha de bloqueo sean obligatorios y visibles únicamente cuando el valor del campo Estado sea 2.

Para mostrar los nuevos campos únicamente cuando el estado sea 2, entramos en las opciones de este campo y clicamos sobre la pestaña de “Property Dependency Manager” y sobre la opción “Visible”



Una vez aquí dentro, clicamos sobre los campos que queramos mostrar, indicando el parámetro que se tendrá que cumplir en la opción “Visible Values” para que se muestren dichos campos.

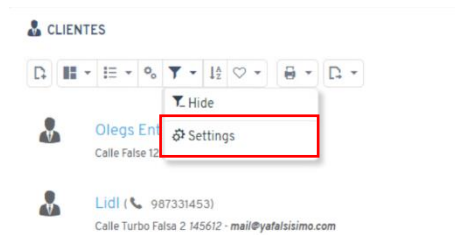


Ejercicio 2 Filtros

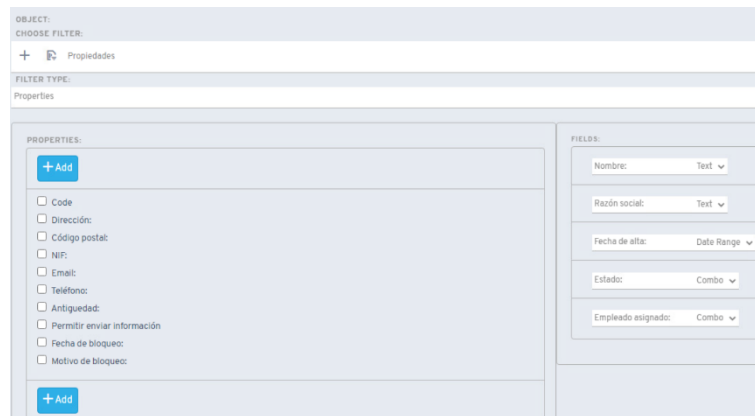
1. Crear filtros en listado de clientes.

- Configurar un filtro llamado “Propiedades”, de tipo properties, que contemple los campos Nombre comercial, Razón social, Empleado asignado, fecha de alta y estado.

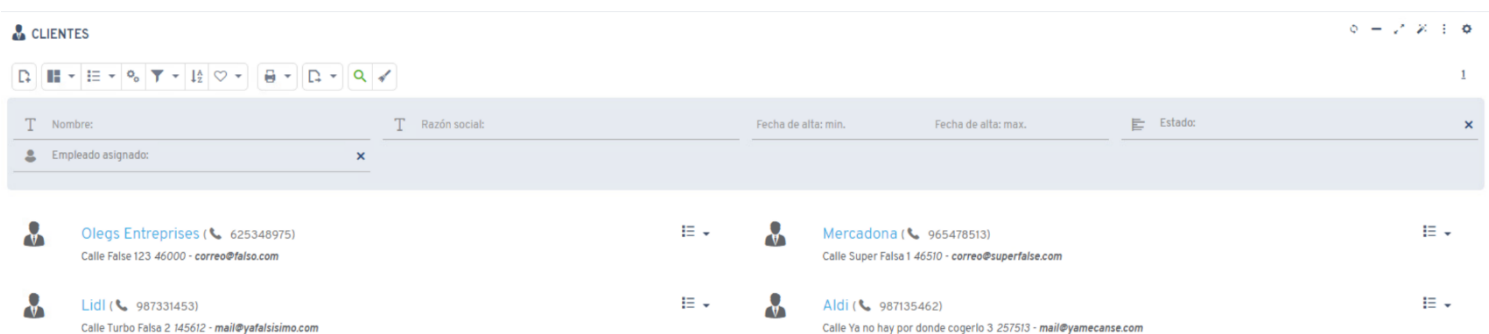
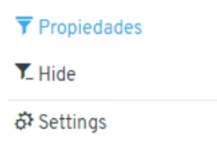
Desde el objeto de clientes, clicamos sobre el botón de filtro y en la opción Settings para abrir el panel de configuración.



En la ventana que se nos abre, clicamos añadimos el nombre que tendrá el filtro indicando que es de tipo “Properties” y asignando los campos por los cuales se va a filtrar.

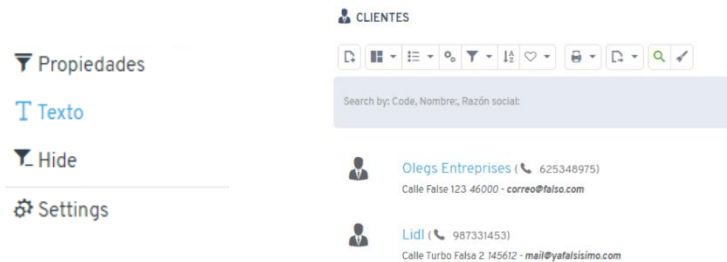


Si volvemos al listado, veremos que nos aparece una nueva opción para filtrar y podemos clicar en la misma para que nos aparezcan las nuevas opciones de filtrado.



- Configurar un filtro llamado “Texto”, de tipo Text, que contemple los campos Nombre comercial, Razón social y código.

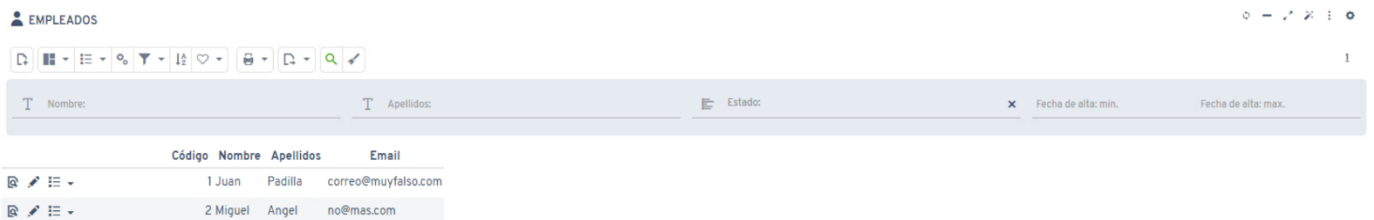
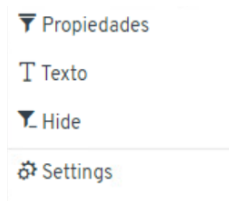
Realizamos los mismos pasos que antes para poder mostrar el filtro solicitado.v



2. Crear filtros en listado de empleados.

- Configurar un filtro llamado “Propiedades”, de tipo properties, que contenga los campos Nombre, apellidos, estado y fecha de alta.
- Configurar un filtro llamado “Texto”, de tipo Text, que contenga los campos Nombre, apellidos y código.

Creamos los dos filtros igual que lo hemos realizado en el apartado anterior.



3. Crear filtro genérico para buscar clientes y empleados.

- Búsqueda genérica de clientes por nombre comercial, código y razón social
- Búsqueda genérica de empleados por nombre y apellidos.

Para crear un filtro genérico, indicamos los campos que queremos utilizar con el tipo texto.

Propiedades

- T Texto
- T Genérico
- Hide
- Settings

CLIENTES

Search by: Code, Nombre, Razón social:

EMPLEADOS

Search by: Nombre, Apellidos:

Código	Nombre	Apellidos	Email
1	Juan	Padilla	correo@muyfalso.com
2	Miguel	Angel	no@mas.com

4. Definir list presets o filtros predefinidos para los clientes.

- Clientes activos (clientes.IdEstado=0)
- Clientes bloqueados (clientes.IdEstado=1)
- Clientes cerrados (clientes.IdEstado=2)

Para definir cada uno de los estados, tenemos que dirigirnos a la sección 5 del configurador del objeto del cliente y crear una nueva lista predefinida. En este caso, para cada uno de los tipos de listado de cliente, en el apartado “SQL Where Sentence” indicamos la sentencia SQL indicada en el enunciado para que nos devuelva la lista de clientes indicada.

Step 5 - List Template (optional)

Custom Property Settings

Template Id: Clientes_Activos
Description: Clientes Activos
Type: List
Is Default: No
Active: Yes

Object Name: sysObjectTemplate
View Name: Clientes Activos
SQL Where Sentence: WHERE IdEstado=0

Template

Header:

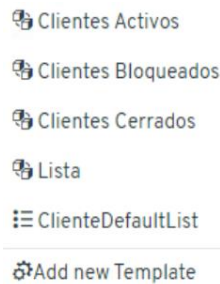
Body:

```

1 <div class="col-6 col-l-6 col-m-12 padding-l">
2 <div class="col-1 col-l-1 col-m-1 col-s-1">
3 <i class="flx-icon icon-client icon-2x icon-margin-right" title="Cliente: {{NombreComercial|isnull:Sin
4 nombre}}"></i>
5 </div>
6 <div class="col-10 col-l-10 col-m-10 col-s-10">

```

Si nos dirigimos al listado y sobre el botón de Templates veremos que se han creado las nuevas vistas.

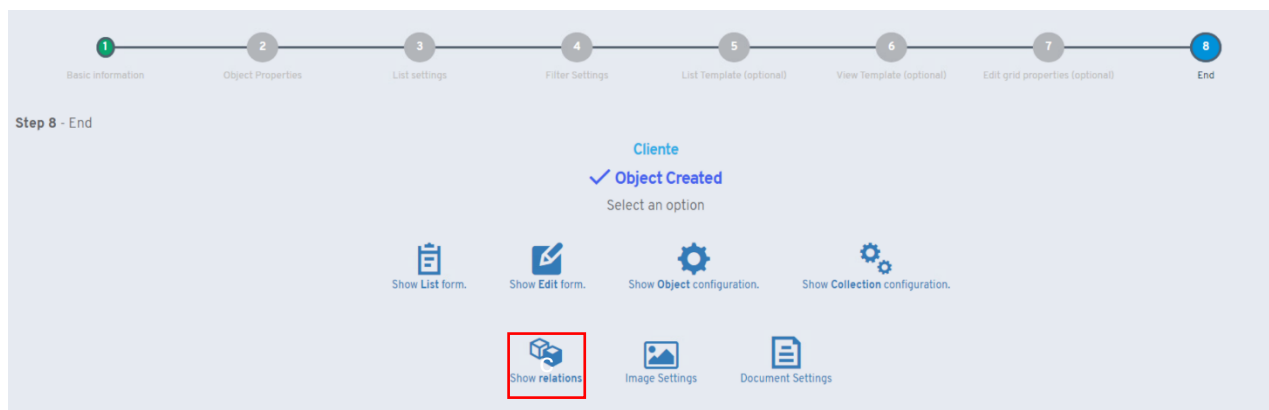


Ejercicio 3 Relaciones entre objetos

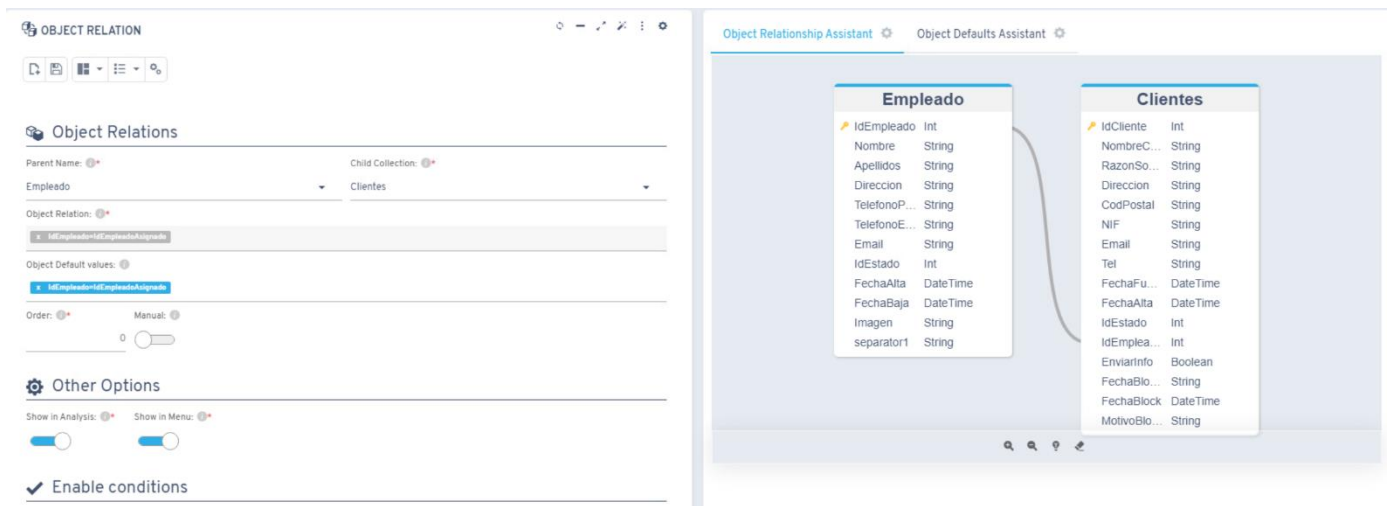
1. Definir relación empleado – clientes

- Relación = IdEmpleado = IdEmpleadoAsignado
- Defaults = IdEmpleado = IdEmpleadoAsignado

Para poder crear las relaciones entre los objetos, nos dirigimos al objeto de clientes y en el asistente de configuración clicamos sobre la opción “Show relations”



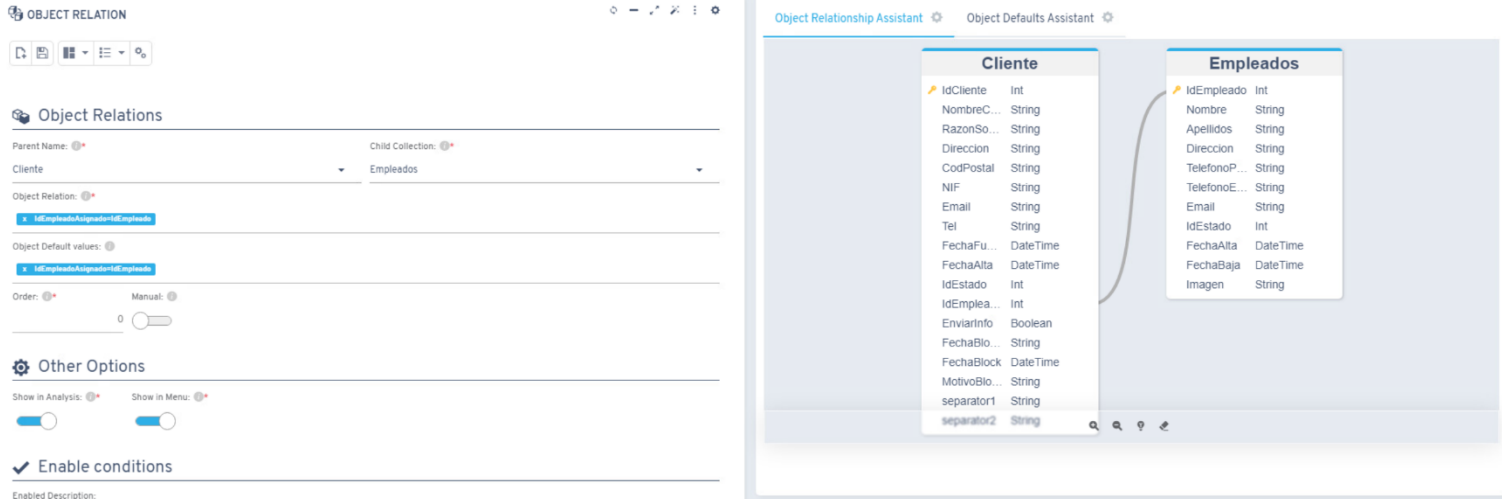
En la ventana nueva que nos aparece, indicamos el objeto padre y el hijo y relacionamos las tablas con el idEmpleado de Empleados a idEmpleado de Clientes.



2. Definir relación cliente – empleados

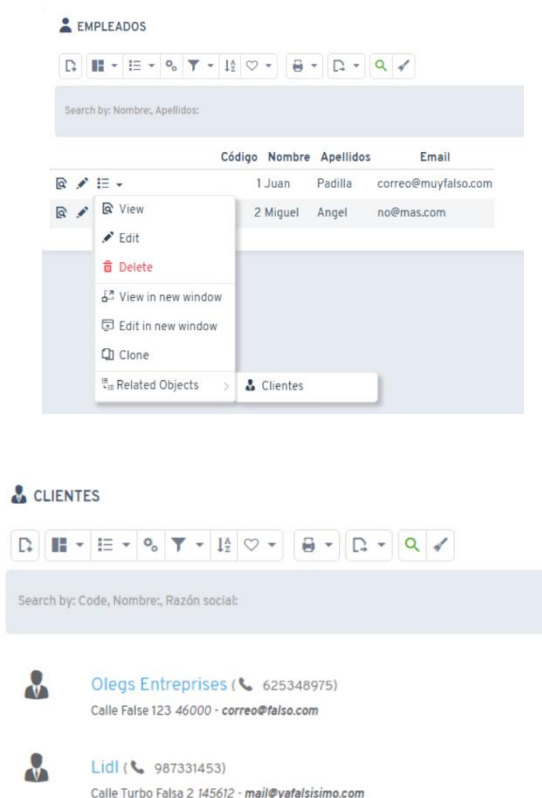
- Relación = IdEmpleadoAsignado=IdEmpleado
- Defaults = IdEmpleadoAsignado=IdEmpleado

Realizamos la relación igual que antes, pero esta vez desde empleado a cliente.



3. Probar que desde un empleado se pueda navegar a sus clientes y que al crear un cliente nuevo herede el empleado del cual partimos.

Si clicamos sobre el menú contextual del empleado, en la opción de “Related Objects”, veremos que nos aparece la relación a los clientes del empleado, y si clicamos esta nos redirige los clientes asociados a ese cliente.



Ejercicio 4 Templates y módulos

1. Crear un template para la lista de clientes.

- Acceder a la lista, en el toolbar: Menú templates – Add new template
- Probar que funcione el template

Añadimos el template tal y como se indica en el enunciado.

Nombre:	Razón Social:	Email:	Teléfono:
Legs Entreprises	B12345678	correo@falso.com	625348975
Mercadona	B12345676	correo@superfalse.com	965478513
Lidl	B25412412	mail@yafalsisimo.com	987331453
Aldi	B123451241	mail@yamecanse.com	987135462

Aquí le indicamos el Template Id, el Type como List y en las secciones de los templates de Body añadimos el código HTML indicado en el enunciado.

Custom Property Settings

Template Id: ListaOlegs	Description: Lista	Type: List
Object Name: Cliente	View Name: Cliente ClienteDefaultList	SQL Where Sentence:

Template

Header:

Body:

```

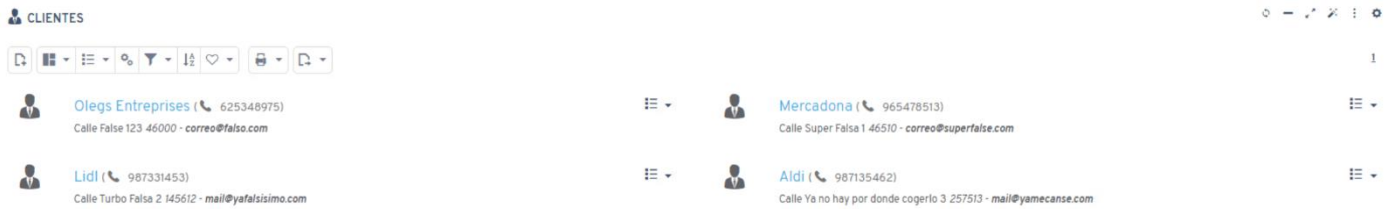
1 <div class="col-6 col-l-6 col-m-12 padding-l">
2 <div class="col-1 col-l-1 col-m-1 col-s-1">
3 <i class="flx-icon icon-client icon-2x icon-margin-right" title="Cliente: {{NombreComercial|isnull:Sin
4 nombre}}"></i>
5 </div>
6 <div class="col-10 col-l-10 col-m-10 col-s-10">
7 <h4>
8 </h4>

```

Si guardamos y refrescamos, vemos que nos aparece una nueva lista personalizada.

Nombre:	Razón Social:	Email:	Teléfono:
Legs Entreprises	B12345678	correo@falso.com	625348975
Mercadona	B12345676	correo@superfalse.com	965478513
Lidl	B25412412	mail@yafalsisimo.com	987331453
Aldi	B123451241	mail@yamecanse.com	987135462

Entramos a la lista personalizada y comprobamos que efectivamente se visualiza con los componentes indicados en el HTML



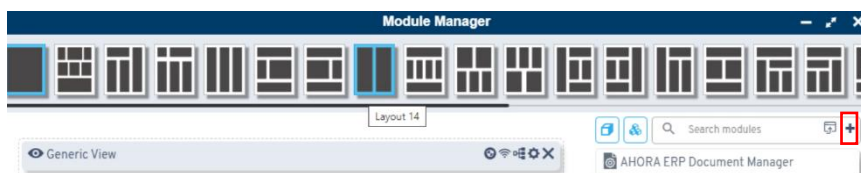
2. Visualizar en el formulario de visualización de un empleado, sus clientes asignados.

- Entrar en la ficha de visualización de un empleado
- Menú contextual – Modules – Module manager
- Crear un módulo de tipo object list:
 - Id: ClientesDeEmpleado
 - Description Clientes de empleado
 - Type: Object List
 - Title: Sus clientes
 - Clasification: Project
 - Container: Default
 - Toolbar: List Toolbar
 - Object Name: Clientes
 - Object Filter: (Clientes.IdEmpleadoAsignado={{IdEmpleado}})

Para poder crear un módulo nuevo, nos dirigimos a cualquier ficha de visualización de empleado y desde el menú contextual que nos indica el enunciado, clicamos en la opción “Module Manager”



En la nueva ventana, clicamos sobre la opción de más para poder crear un nuevo módulo.



En la nueva ventana, rellenamos los datos con los indicados en el enunciado.

Module

Definition

Id: ClientesDeEmpleado Description: ClientesDeEmpleado Icon: noicon

Type: Object List Title: Sus clientes Classification: Project Module

Container

Container: Default Toolbar: List toolbar Full Screen Button: ☐ Collapsible Button: ☐

Header Class: Module Class: Refresh Button: ☐ Search Button: ☐

Refers to object

Object Name: Clientes Object Filter: IdEmpleadoAsignado=(IdEmpleado) Show When New: ☐ Manual Init: ☐

Si volvemos a la anterior ventana, veremos que se nos ha creado un nuevo módulo, el cual podemos arrastrar para que se pueda visualizar con la vista de por defecto.



Volviendo a la vista de la ficha del empleado, vemos como ahora nos aparecen los clientes asociados al mismo.

Empleado

Nombre: Juan Apellidos: Padilla

Tel personal: 645987132 Tel empresa: 987632145

Email: correo@muysalao.com Fecha de alta: 08/12/2024

Estado: Inactivo Fecha de baja: 08/12/2024

Dirección: Calle Aun mas falso 123

SUS CLIENTES

Oleg's Enterprises (625348975)
Calle Falso 123 46000 - correo@falso.com

Mercadona (965478513)
Calle Super Falso 1 46510 - correo@superfalso.com

Lidl (98731453)
Calle Turbo Falso 2 145612 - mail@yalafalsissimo.com

Aldi (987135462)
Calle Ya no hay por donde cogerlo 3 257513 - mail@yamecanse.com

3. Modificar template:

- Añadir el literal Estado al template del listado de clientes
- Modificar el template para que, en vez de navegar a la ventana de visualización del cliente, lo haga a la ventana de edición.

Para modificar el template, vamos a la lista que hemos creado previamente y abrimos su configuración. Para abrir la ventana de edición, modificamos la fila donde se hace el click con el siguiente código:

```
<a href="#"
onclick="flexygo.nav.openPage('edit','Cliente','(IdCliente={{IdCliente}})',null,'current',false,$(this))">
<span class="{{IdEstado|switch:[2:txt-danger,else:txt-outstanding]}}">{{NombreComercial|isnull:Sin
nombre}}</span>
```

Y añadimos el estado a la visualización de la lista.

```
<small><i class="">{{IdEstado|isnull:Sin estado}}</i></small>
```

Si volvemos a la lista, vemos como ahora nos aparece el estado de los mismos, y si hacemos clic en alguno de ellos, nos manda a su ventana de edición.

The screenshot displays the FlexyGO application interface. On the left, the 'CLIENTES' list shows two entries: 'Olegs Entreprises' and 'Lidl'. The main area shows the 'Cliente' edit form for 'Olegs Entreprises'. The form includes fields for 'Nombre', 'Razón social', 'Dirección', 'Teléfono', 'Email', 'NIF', 'Estado', 'Empleado asignado', 'Fecha de alta', and 'Antigüedad'. The 'Estado' field is set to 'Activo'. The 'Empleado asignado' field is set to 'Juan'. The 'Fecha de alta' field is set to '08/12/2024'. The 'Antigüedad' field is set to '07/01/1999'. The 'Permitir enviar información' toggle is turned on. A 'Save' button is at the bottom.