Practica 1 ITI: ODOO

Alejandro Gallego López

Edmundo Gascón Berenguer



2017

UMH

12/05/2017

**ÍNDICE**

**1.INTRODUCCIÓN2**

1.1 ¿Por qué escoger ODOO como sistema ERP? 2

1.2 Principales Características Versiones3

1.3. Versiones3

1.4. ODOO como TPV3

1.5. El TPV es personalizable3

**2.Descripción del sistema4**

2.1 Parte Hardware4

2.2 Parte Software5

**3.Instalación5**

**4.Dependencias6**

4.1. Configuración terminal punto de venta 6

**5.Explicación de las categorías de los productos** **7**

5.1. Diseño del piso: 7

**6.Modificación del código interno** **8**

**7.Pantalla Login(código)** **24**

**8.Powered by odoo** **24**

**9. Creación de widgets personalizados para crear poups personalizados24**

**10.Código CSS** **24**

**11. Video explicativo funcionamiento TPV-ADMIN:24**

# ITRODUCCIÓN:

Odoo, nace en Bélgica en 2004 (hasta 2013 funcionó bajo el nombre de OpenERP), y cuenta actualmente con más de 2 millones de usuarios, siendo el software para negocios más instalado en el mundo. Su principal mercado son las pequeñas y medianas empresas, aunque eso no le exime de contar, entre sus usuarios, con más de 300.000 grandes empresas de la talla de **Danone**, Auchan o Le Poste.

**1.1 ¿Por qué escoger ODOO como sistema ERP?**

Odoo es un sistema ERP integrado y de software libre. ¿Esto qué quiere decir?, pues que permite la integración con otras herramientas y tiene cero costes de licencia. Podrás instalarlo en tantas terminales como quieras durante un periodo indefinido.

**1.2 Principales Características**

Es un sistema de código abierto: esto significa que cualquier módulo puede ser modificado o diseñado desde cero, adaptándolo a las necesidades de tu empresa.

**Multiplataforma**: independientemente del sistema operativo que utilices, a través de un navegador web podrás acceder a su interfaz.

**Su interfaz es 100% web**, por lo que podrá usarlo desde cualquier dispositivo (ordenador, tableta o móvil) y desde cualquier sistema operativo (Windows, MacOS, Linux, Android). Además resulta especialmente intuitivo y sencillo de usar, con lo que reducirá en gran medida los costes de formación.

Está pensado para cubrir las necesidades de las empresas del siglo XXI: **en más de 100 idiomas**, disponible en la nube o en local, muy flexible,...

**Fácil** **manejo**: no son necesarios grandes conocimientos de informática para poder utilizar su sencilla interfaz.

Posee una importante comunidad de desarrolladores, con más de 2000 módulos liberados disponibles que pueden combinarse entre sí.

Integración con otras aplicaciones: visualización de PDF, importación/exportación de documentos de Microsoft Office u Open Office, compatibilidad con Google Maps…

Puedes contratar únicamente lo que necesites. Lo más habitual es contratar el proceso de implantación a una empresa especializada y dado su carácter modular, podrás instalar solamente aquellas partes que realmente necesites.

**1.3. Versiones**



**1.4. ODOO COMO TPV**

Requiere para funcionar un navegador web, y la propia instalación de odoo como servidor, **funciona tanto conectado en línea con el servidor OpenERP** (online) **como sin dicha conexión (offline)**, manteniendo toda su funcionalidad durante los periodos de desconexión, volverán a sincronizarse los datos cuando recupere la conexión, siempre y cuando en dicho equipo esté implantado ODOO como servidor.

**Integrado con el Inventario, Almacén y la Contabilidad:** Todos los cambios que se realicen en el TPV y punto de venta aparecerá al momento en la gestión de Inventario y Cuentas.

Lo que permitirá disponer en tiempo real del inventario actualizado, estadísticas de venta o la facturación realizada y su correspondiente asiento contable, utilizando por tanto OpenERP como administrador. Todo ello facilita enormemente la gestión de sus tiendas.

**1.5. EL TPV es personalizable**

El acceso al TPV se puede restringir mediante permisos y a diferencia de los pedidos de venta tradicionales, se relaciona con la apertura y cierre de caja, permitiendo y controlando el registro de las mismas.

Posibilidad de identificación de los clientes, podrás realizar un seguimiento de sus hábitos de consumo, para después ajustar las ofertas, promociones y descuentos a sus necesidades, todo ello a través del programa de fidelización.

1. **Descripción del Sistema:**

Se trata de un sistema de gestión de comandas para bares y restaurantes.

Para ello elegimos el módulo de Odoo-TPV con varias modificaciones de código interno para que el estilo, la apariencia y la funcionalidad fuese lo más vistosa y completa posible para el cliente y sus necesidades previamente acordadas.

Para realizar el proyecto hemos tenido en consideración las siguientes dos partes:

**2.1. Parte Hardware:**

Esta parte consta de los dispositivos electrónicos y otros elementos físicos que debemos aportar:

* **2 Impresoras TPV:** para la impresión de tickets, una para impresión de productos de barra como por ejemplo (coca colas, cerveza, café, tapas…), y otra impresora que estará ubicada en la cocina para primeros platos o comidas más elaboradas.
* **Pc-sobremesa:** en este caso hemos elegido un “mini-pc” para la instalación de nuestro “Ubuntu server” sin interfaz gráfica, dado que no requiere un ordenador potente para su ejecución y funcionamiento, en concreto elegimos Cubi N-021BEU Celeron N3050/4GB/500GB.
* **2 tablets:** Elegimos estos dos dispositivos para los camareros, desde ellos a través de un “Frame” previamente configurado a través de cualquier navegador en nuestro caso google Chrome, el camarero podrá realizar las comandas, y estas según el producto se imprimirán en barra o en cocina según corresponda.
* **RASPBERRY PI 1 MODEL B+:** Para la configuración y funcionamiento de las impresoras debemos obtener este mismo modelo de RaspBerry “PI 1 Model B+”, la conectaremos por usb previamente al ordenador, instalaremos Ubuntu server y un script que contendrá los drivers de la impresora. Al conectar las impresoras se ejecutará dicho script y le asignará una IP fija a cada impresa, más tarde en el apartado de configuración TPV de odoo le asignaremos dicha IP a productos de barra y de cocina según corresponda.
* **Cable de red 3m:** Irá conectado a la RaspBerry directamente del router.

**2.2. Parte Software:**

* Script de auto instalación de Odoo 10.
* Ubuntu Server.
* Módulos:
  + Point of sale (tpv)
  + ESC/POS Hardware Driver
  + Restaurante
  + Pos\_cache
  + Pos\_box\_software upgrader
  + Point of sale receipt reprinting
  + Point of sale extra Access right (restringir acceso al tpv )
  + Mercury payments (pagos con tarjeta)
  + Página de inicio de pos-box.
* Modificación código interno.
  + Módulo pos\_restaurant
  + Código CSS

1. **Instalación:**
2. Introducir el script de auto instalación de Odoo 10 en el servidor (Ubuntu server) ejecutarlo

Sudo sh.” nombre del archivo ”.--> esperar a que termine (establecer una ip estatica al servidor de Ubuntu).

1. Crear permisos para acceder a ftp de odoo y permisos para la base de datos:

El usuario de postgres es “postgres” debemos darle una contraseña para poder acceder a la base de datos con programas externos

* 1. **Sudo –u postgres psql**
  2. **Alter user postgres password “password”**
  3. **Sudo service postgresql restart (**reiniciamos el servicio **)**

Una vez instalado debemos modificar el archivo de configuración de postgres para que nuestro servidor sea accesible en la mima red.

* + 1. Vamos a la ruta 🡪/**etc/postgresql/9.3/main/pg\_hba.conf**
    2. Editamos el archivo añadiendo debajo del host en la línea 92 introduciendo lo siguiente:

**Host all all 127.0.0.1/24 md5**

* + 1. Editamos el archivo de configuración de postgres 🡪 **\*misma dirección** que la anterior**/postgresql.conf**
    2. En la línea 59 donde podemos leer “listen address” ponemos **“\*”.**

1. **Dependencias:**

Si falta alguna dependencia instalarla, es posible que se deba descargar el archivo y descomprimirlo en la capeta adecuada de python para aplicar la instalación de la dependencia. ( Página de dependencias en marcadores🡪”comandas” ).

**Instalado Odoo 10**

Módulos a instalar:

1. Point of sale (tpv)
2. ESC/POS Hardware Driver
3. Restaurante
4. Pos\_cache
5. Pos\_box\_software upgrader
6. Point of sale receipt reprinting
7. Point of sale extra Access right (restringir acceso al tpv )
8. Mercury payments (pagos con tarjeta)
9. Página de inicio de pos-box.

Creación de usuarios:

Creamos un usuario administrador y otro usuario “camarero” (a este acceden todos los camarero) mientras que al usuario administrador sólo el administrador, es decir, la maquina principal en barra.

**4.1. Configuración terminal punto de venta:**

*Configuración punto de venta Administrador:*

🡪Imágenes de configuración

🡪Ip de las diferentes impresoras de pedido y de tiquek final(pos-box)

🡪Configuración de mercury payments.

🡪Configuración general -🡪

-Restaurante

-Descuentos globales

-Tarjeta de crédito

-Reimprimir recibos.

**5. Explicación de las categorías de los productos:**

1. Creamos las categorías generales primero, como pueden ser :

Bocadillos, hamburguesas etc.

1. Creamos las subcategorías que van a englobar las categorías principales que hemos creado. Por ejemplo, si tenemos la categoría principal “bocadillos” y queremos incluir los bocadillos que tengan entre sus ingredientes “carne” debemos hacer una subcategoría que los incluya. A esta subcategoría la llamaremos “Bocadillos/carne” para ello creamos una categoría “carne” cuya categoría padre sea “Bocadillos”. Ahora si queremos hacer lo mismo con los bocadillos de origen vegetal, el procedimiento es el mismo, solo que ahora añadiremos una categoría llamada “vegetal” cuya categoría padre sea también “Bocadillos”.
2. Una vez hecho esto, en la configuración de cada producto indicaremos a que categoría pertenece cada producto. Por ejemplo, el bocadillo de lomo pertenecerá a la categoría de “bocadillos/carne”.

Nota: para que las categorías de los productos se muestren como imagen en el tpv esta opción debe estar marcada en la configuración del tpv.

**5.1. Diseño del piso:**

En la configuración debemos diseñar al menos una mesa para que aparezca el diseño del piso y poder modificarlo. También debemos crear el número de pisos que deseemos.

# MODIFICACION DEL CÓDIGO INTERNO

Módulo pos\_restaurant:

***Pos.xml 🡪 línea 137🡪***

<!-- si el usuario no es administrador el boton de pago cambia

-->

<t t-if="widget.pos.user.name !='Administrator'">

<button class='pagoCamarero'>

<div class='pay-circle'>

<i class='fa fa-chevron-right' />

</div>

PAGAR

</button>

</t>

***Pos.xml 🡪 línea 155🡪***

<!-- **si el usuario es administrador muestra todo el teclado** -->

<t t-if="widget.pos.user.name =='Administrator' ">

“codigo del teclado”

</t>

<!-- **si el usuario no es administrador solo muestra dos botones** -->

<t t-if="widget.pos.user.name !='Administrator'">

<div class="numpad">

<button class="anular number-char">ANULAR</button>

<button id="botonBorrar" class="borrar">BORRAR</button>

</div>

</t>

***Pos.xml 🡪 línea 767🡪***

<!-- si existen datos del cliente, es decir, si hemos seleccionado

un cliente, mostramos en el tiquet final, los datos del mismo a modo

de factura -->

<t t-if="widget.pos.get\_client()">

--------Datos del cliente--------<br/>

Empresa: <t t-esc="widget.pos.get\_client().name" /><br/>

Direccion: <t t-esc="widget.pos.get\_client().street"/><br/>

CIF/NIF:<t t-esc="widget.pos.get\_client().barcode"/><br/>

Tlf:<t t-esc="widget.pos.get\_client().phone"/><br/>

<br />

----------------------------------<br/>

</t>

</div>

<br /><br />

***Screen.js 🡪linea370🡪***

/\*

Obtenemos las clases de los botones del teclado, tambien las

de los botones que aparencen cuando no se es administrador.

Al hacer click en alguno de los botones con la funcion .click\_.bind

llamamos a la funcion determinada.

La accion del boton de borrar esta dentro de un bucle para que se

aplique dos veces y así poder borrar el producto por completo.

\*/

var NumpadWidget = PosBaseWidget.extend({

template:'NumpadWidget',

init: function(parent) {

this.\_super(parent);

this.state = new models.NumpadState();

},

start: function() {

this.state.bind('change:mode', this.changedMode, this);

this.changedMode();

this.$el.find('.numpad-backspace').click(\_.bind(this.clickDeleteLastChar, this));

this.$el.find('.numpad-minus').click(\_.bind(this.clickSwitchSign, this));

this.$el.find('.number-char').click(\_.bind(this.clickAppendNewChar, this));

this.$el.find('.mode-button').click(\_.bind(this.clickChangeMode, this));

this.$el.find('.anular').click(\_.bind(this.clickAppendNewChar, this));

for(var i =0; i<2; i++){

this.$el.find('.borrar').click(\_.bind(this.clickDeleteLastChar, this));

}

},

/\*

Funcion que borra el producto seleccionado, a la que hemos llamado en el boton "Borrar"

\*/

clickDeleteLastChar: function() {

/\*

Obtenemos by comprobamos si el usuario conectado es o no administrador

\*/

var users = this.pos.user.name;

if(users != "Administrator"){

/\*

Si el usuario es administrador, obtenemos la clase del boton y deshabilitamos

el boton de borrrar en el caso de que el producto este de color negro(que se ha enviado a cocina).

\*/

var spanses = document.getElementsByTagName('li');

var bottonBorrar = document.getElementById('botonBorrar');

for (var x = 0; x < spanses.length; x++) {

if (spanses[x].className == 'orderline selected'){

bottonBorrar.disabled = true;

}

}

}

return this.state.deleteLastChar();

},

***Screen.js 🡪linea503🡪***

/\*

Condicion para mostrar la pantalla de pagos -->(que no haya ningún pedido a medias).

obtenemos todos los spans si la clase es order-submit, es decir, está todo envido podemos pasar

a la pantalla de pagos.

}

\*/

var spans = document.getElementsByTagName("span");

for (var i = 0; i < spans.length; i++) {

if (spans[i].className == 'control-button order-submit') {

self.gui.show\_screen('payment');

}

}

});

/\*

boton de pago para el usuario camarero

hacemo lo mismo que el boton anterior(solo para administrador)

solo que en el boton de pagos de la pantalla para el camarero.

\*/

this.$('.pagoCamarero').click(function(){

/\*

Condicion para mostrar la pantalla de pagos -->(que no haya ningún pedido a medias).

\*/

var spans = document.getElementsByTagName("span");

for (var i = 0; i < spans.length; i++) {

if (spans[i].className == 'control-button order-submit') {

self.gui.show\_screen('payment');

}

}

});

***Screen.js 🡪linea597🡪***

/\*

Por defecto dehabilitamos el boton de borrar si el usuario no es Administrador.

\*/

var users = this.pos.user.name;

if(users != "Administrator"){

var bottonBorrar = document.getElementById('botonBorrar');

botonBorrar.disabled = true;

}

***Screen.js 🡪linea624🡪***

var users = this.pos.user.name;

if(users != "Administrator"){

/\*

obtiene la categoria del producto de la linea seleccionada

descativa el boton "borrar" si el producto que pincha no esta en verde

\*/

var orden = this.pos.get\_order();

var spans = document.getElementsByTagName("li");

var button = document.getElementsByTagName("button");

var botonBorrar = document.getElementById("botonBorrar");

var precio = orden.get\_selected\_orderline().product.price;

botonBorrar.disabled = true;

for (var i = 0; i < spans.length; i++) {

if (spans[i].className == 'orderline selected dirty'

&& $('.orderline selected dirty')) {

for(var j = 0; j< button.length; j++){

button[j].disabled = false;

}

}

}

}

var order = this.pos.get\_order();

var nombre = order.get\_selected\_orderline().product.pos\_categ\_id[1];

var categoria = nombre.split("/");

/\*

si la categoria es carne muestra el panel de seleccion de las carnes

\*/

if(categoria[1] == " Carne" ){

var line = this.pos.get\_order().get\_selected\_orderline();

this.gui.show\_popup('selection',{

title: "TIPO DE CARNE",

list: [

{ label: 'Muy Hecho', item: 'Muy Hecho' },

{ label: 'Poco Hecho', item: 'Poco Hecho' },

{ label: 'Al Punto', item: 'Al Punto' },

],

confirm: function(item) {

if (line) {

line.set\_note(item);

}

},

cancel: function(){

// user chose nothing

}

});

}

},

***Screen.js 🡪linea1222🡪***

click\_product: function(product) {

if(product.to\_weight && this.pos.config.iface\_electronic\_scale){

this.gui.show\_screen('scale',{product: product});

}else{

/\*

Obtenemos la categoria de los productos, si no contiene la palabra Extras

agregamos el producto.

\*/

var categoria = product.pos\_categ\_id[1];

var palabra = categoria.split("/");

if(categoria != "EXTRAS"){

this.pos.get\_order().add\_product(product);

}

/\*

Obtenemos la categoria de los productos, si contiene la palabra Extras

añadimos una nota con el nombre del Extra en el producto seleccionado(el actual).

\*/

var categoria = product.pos\_categ\_id[1];

var palabra = categoria.split("/");

console.log(palabra[1]);

if(categoria == "EXTRAS"){

var lineaActual = this.pos.get\_order().get\_selected\_orderline();

var nota = lineaActual.get\_note();

lineaActual.set\_note(nota + "/" + product.display\_name);

}

/\*

Si el extra contiene la palabra "carne" al pinchar en el producto

aparece el popup de la seleccion del tipo de carne por si queremos

modificarlo.

\*/

if(palabra[1] == " Carne" ){

var line = this.pos.get\_order().get\_selected\_orderline();

this.gui.show\_popup('selection',{

title: "TIPO DE CARNE",

list: [

{ label: 'Muy Hecho', item: 'Muy Hecho' },

{ label: 'Poco Hecho', item: 'Poco Hecho' },

{ label: 'Al Punto', item: 'Al Punto' },

],

confirm: function(item) {

/\*

agrega una nota al producto con el tipo de

carne seleccionado.

\*/

if (line) {

line.set\_note(item);

}

},

cancel: function(){

// user chose nothing

}

});

}

}

},

1. **Código pantalla de lógin**

Configuración 🡪 interfaz de usuario 🡪 vistas 🡪 “buscar login”🡪 login layout (web.login).

1. **POWERED BY: ODOO**

Configuración 🡪 interfaz de usuario 🡪 vistas 🡪” buscar web.menu\_secondary ”.

1. **Creación de widgets personalizados para crear poups personalizados**

Pos.xml 🡪 crear selectionpopupwidget con otro nombre

Popups.js🡪crear un selecctionpopupwidget con otro nombre y llamarlo desde pos.xml

Screen.js 🡪llamar a selectionwidget creado

1. **CODIGO CSS**

“Los archivos del código CSS están comentados” (ejemplo línea 672 del archivo pos.css)

1. **Video explicativo funcionamiento TPV-ADMIN:**

[**https://www.youtube.com/watch?v=tgc273by2Sk&ab\_channel=NeriTechnology**](https://www.youtube.com/watch?v=tgc273by2Sk&ab_channel=NeriTechnology)