

# UD 01 - INTRODUCCIÓN A LOS SISTEMAS ERP-CRM

## 1. Introducción a la gestión empresarial

Una empresa:

- existe siempre y cuando obtenga beneficios porque gracias a ellos adquieren capacidad para crecer y desarrollarse
- tiene que gestionar perfectamente los recursos (eficientemente) para poder ser competitiva ante otras (privadas) o para dar un servicio (públicas)
- tiene entre sus principales objetivos el cliente (necesidades, cómo satisfacerlas)

La gestión empresarial engloba multitud de procesos y técnicas.

El desarrollo empresarial es complejo y necesita gran organización y planificación para llevar a cabo sus propósitos.

## El empresario

Para llevar a buen puerto la empresa:

- Debe fijarse un **proyecto empresarial realista** a medio y largo plazo, definiendo **estrategias** para llevarlo a cabo
- Mantener **tamaño de la empresa adecuado** para que no se **sobredimensione** y obtener así **aumento de rentabilidad**
- Disponer de **herramientas de análisis efectivas** para **anticiparse a los cambios del mercado**
- El **cliente** es el objeto principal de producción y todos los departamentos deben centrarse en él, orientado sus actuaciones a un proyecto que lo **satisfaga** lo mayor posible.

## Estrategia

Plan coherente que integra todas las áreas de la empresa y que sigue un mismo plan de acción basado en tres aspectos:

- Misión: Propósito para el que se ha formado la empresa
  - Visión: Imagen futura de la empresa, a dónde se quiere llegar
  - Valores: Reglas y fundamentos que regulan el comportamiento de las personas ante diferentes situaciones
- Esos conceptos forman la IDENTIDAD de la empresa.

## Áreas de la gestión empresarial

La Dirección coordina la interacción entre todas las áreas (Recursos Humanos, Contabilidad y Finanzas, Marketing y Ventas, Producción, Compras, Logística...). Las compañías tradicionalmente siguieron el concepto de silos o modelo funcional (Unidades funcionales independientes).

### Modelo funcional

- Es un **modelo muy vertical** que exige más personas (revisores, coordinadores, jefes) y que provoca un flujo lento de información de donde se genera a donde se necesita
- **Impide a las empresas reaccionar rápidamente** ante cambios (Este es el factor más importante para el éxito de una empresa)
- Las áreas funcionales son realmente **interdependientes**
- Una **mejor integración de las áreas funcionales** significa: Mejor comunicación, mejor flujo de información (workflow), optimización de costes, agilidad empresarial

### Problemas técnicos en la integración

Los sistemas de información de cada departamento tienen formatos propios para almacenar la información

El intercambio de información muchas veces es un proceso manual o periódico semiautomatizado... Son procesos costosos, provocan datos obsoletos o incoherentes.

Se desarrolla el estándar EDI (Electronic Data Interchange) para intercambio de documentos de negocio entre SI de diferentes compañías.

Se desarrollan productos específicos para integrar los sistemas de información (Middleware) pero estos aunque son una verdadera industria, son artesanales, parches que deben adaptarse, la capacidad de compartir información está limitada a lo que el middleware permite...

## Modelo de procesos de negocio

Es un cambio de paradigma.

Es el conjunto de actividades que dadas determinadas entradas generan una salida o producto valioso para el consumidor.

El consumidor puede ser el cliente o los trabajadores de la propia empresa.

Una visión de la empresa en términos de procesos de negocio y no de áreas funcionales permite ver la empresa en términos productivos.

## Ejemplo: Proceso de negocio. Venta de ordenador

Input	Functional area responsible for input	Process	Output
Request to purchase computer	Marketing and Sales	Sales order	Order is generated
Financial help for purchase	Accounting and Finance	Arranging financing in-house	Customer finances through the computer company
Technical support	Marketing and Sales	24-hour help line available	Customer's technical query is resolved
Fulfillment of order	Supply Chain Management	Shipping and delivery	Customer receives computer

## Requisitos

- Compartir datos entre áreas funcionales de forma eficiente y efectiva
- Información actualizada

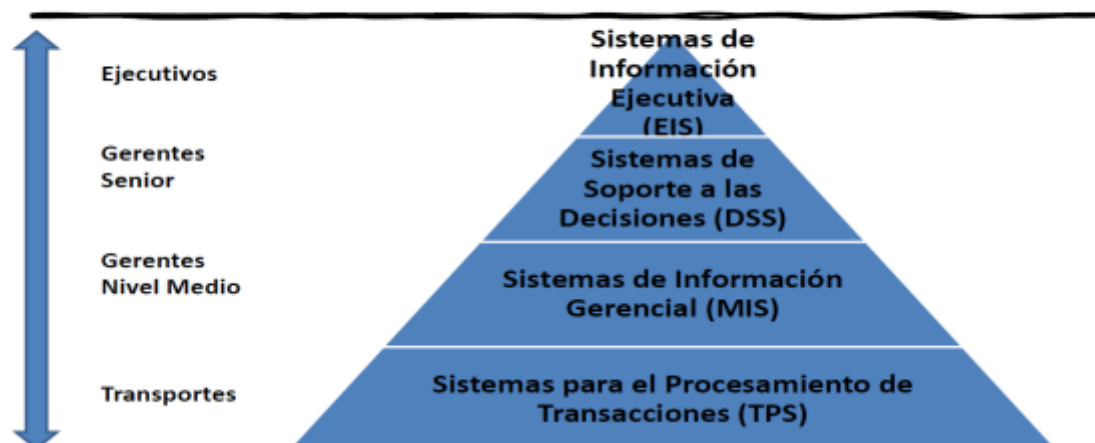
## Solución

- Entender qué procesos de negocio se realizan en cada área de la empresa
- Conocer qué datos necesitan compartir las áreas funcionales
- Desarrollar un único Sistema de Información Empresarial Integrado para toda la compañía.

## Los sistemas de información



## Tipos de Sistemas de Información



## La informática de gestión empresarial

Evolución de la informática de gestión (tratamiento automático de la información y comunicación de la misma):

- **Sistema de procedimiento de transacciones (TPS):** Almacena, modifica y recupera información referente a transacciones en una empresa
- **Sistema de automatización de oficinas:** Destinadas a ayudar en el trabajo diario del administrativo (Hoja de cálculo, editores de presentaciones, gestores de correo)
- **Sistemas de planificación de recursos:** Información y procesos de organización en un solo sistema. (ERP). Partes integradas en una única.
- **Sistema experto:** Capaces de solucionar un conjunto de problemas que exigen gran conocimiento sobre determinado tema. Imitan actividades de humano para resolver problemas emulando el comportamiento del experto en un dominio concreto (rama de la IA)

## Organización de una empresa y relaciones externas

- **Entorno próximo:** Factores de producción y distribución (mano de obra, entidades financieras, proveedores, organismos oficiales) que tienen que ver con su actividad.
- **Entorno general:** Efectos directos e indirectos en la gestión general de la empresa.

La empresa necesita sistema que dé, rápida y eficientemente, toda la información del entorno que le rodea. Varias aplicaciones suponen ineficacia y repetición de datos -> La solución es un ERP. Necesita saber cómo fluye la información dentro de la empresa, conocerse todos sus recursos empresariales y gestionarlos de forma eficaz para ser eficiente en los procesos productivos.

- Entre empleados de empresa
- Entre empleados y empresa
- Entre la empresa con clientes y proveedores

Flujos de información pueden ser:

- Informales y no estructurados
- Formales y estructurados que se centran en información acerca de procesos críticos de la empresa.

Un sistema de información es un conjunto organizado de elementos relacionados orientados al tratamiento y administración de información.

Facilita el conocimiento propio de la empresa para mejorar planificación, gestión y control.

De conocer y gestionar los flujos de información obtiene ventajas competitivas que mejoran su eficiencia, la calidad del producto, el servicio ofrecido y facilitan la captación.

Los ERP (Enterprise Resource Planning), sistemas de planificación de recursos empresariales, son sistemas de gestión de información que se caracterizan por ser una aplicación con varias partes integradas que se especializan en manejar todos los datos relevantes para la continuidad de la empresa.

Un CRM (Customer Relationship Management) es un ERP que se centra en la relación con los clientes -> Información de contacto orientada a ventas.

## 1. Objetivos de un ERP

- Optimizar procesos empresariales
- Acceder a información confiable
- Permitir compartir información entre los componentes de la organización
- Eliminar datos y operaciones innecesarias

UN ERP se distingue de otro software empresarial porque:

- Es **integral**: Integra todos los procesos de la empresa, considerándolos como áreas relacionadas entre sí. Reduce tiempo y costes. Los datos forman parte de una base de datos centralizada (se ingresan una sola vez) y facilitan el flujo de información entre los módulos.
- Es **modular**: Cada módulo del sistema es un área funcional de la empresa. Los módulos comparten información entre sí. Habitual que cada módulo use software específico para su funcionalidad.
- Es **adaptable**: Las anteriores facilitan la adaptabilidad a las necesidades de cada empresa; pero aun así, para abaratar costes, puede usarse solución más genérica y modificar algunos procesos para alinearlos al sistema ERP. (Total configuración. Una única herramienta ERP puede dar servicio a diferentes empresas, modificando los módulos activos y sus relaciones)
- Tiene **acceso a base de datos**
- Sus componentes **interactúan** entre sí
- Los datos son consistentes y completos

Son complejos y difíciles de implantar por necesitar un desarrollo personalizado para cada empresa a partir del paquete inicial. De estas adaptaciones se encargan las consultoras:

- **Consultoría de negocios**: Estudia procesos de negocio de la compañía y evalúa su correspondencia con los procesos ERP para personalizarlo y ajustarlo a las necesidades de la organización
- **Consultoría técnica**: Estudio de los recursos tecnológicos existentes, en ocasiones, programación del sistema, obtener determinados informes.

Se puede complementar con otros módulos como proyectos, planificación de ventas, mantenimiento, gestión de cambios de producto, configuración de productos a medida, gestión de relaciones con los clientes (CRM), controles de planta y almacenes, seguridad y contingencia, cuadros de mando, gestión de archivos, gestor comercial, proveedores...

Cada grupo de trabajadores tiene su **rol** supervisado por un administrador. Según el puesto a desempeñar se activarán o desactivarán sus módulos asociados.

Los **CRM** son una aplicación específica de los **ERP** para mantener crear, potencial relaciones de clientes con la empresa, apoyando sus políticas de marketing. Los sistemas de CRM se pueden dividir en

- aplicaciones electrónicas para distribución de la empresa
  - centros de atención telefónica
  - autoservicio a clientes
  - gestión de actividades que afectan a clientes y ventas.
- Facilitan tomar decisiones en tiempo real, incrementando la rentabilidad del cliente al obtener información útil a partir de datos complejos (clientes interesados; clientes no interesados...)

## 2. Ventajas ERP

- Aumento de la información que posee la empresa sobre los potenciales clientes. Si incluyen CRM aportan beneficios relacionados con la administración de los clientes de la empresa
- Aumento de las ventas
- Problemas derivados del tratamiento de información con sistemas anteriores
- Mayor eficiencia operativa
- Acceso de información y mejora en herramientas para su tratamiento
- Reducción de costes empresariales: Operaciones de TI y comunicación
- Mayor facilidad en configuración
- Mejora del entorno de integración
- Administrar interdependencias, mayor seguridad

## 3. Limitaciones ERP

- Debe ser utilizado y realizado por personal capacitado. Puede haber malas inversiones o falta de políticas corporativas que afectan al input de datos del ERP.
- Su instalación es costosa, puede no ser asequible a cualquier empresa
- Se ven como sistemas rígidos y difícil de adaptar al modo de trabajo de las empresas
- Sufren "cuellos de botella". Incidencia de un departamento afectan al resto
- Altos costes de modificar ERP una vez implantado

## 4. Tipos de licencias y revisión de ERP actuales

### Licencias

Licencia de software: Contrato entre el propietario y el usuario de una aplicación.

**Software bajo licencia GPL:** 'Permite redistribución y modificación de aplicaciones bajo aplicación de la misma licencia. Licencia usada por Open ERP. Creada por FSF promotora de GNU

**Software bajo licencia BSD:** Dar crédito a los autores. Permite libre distribución y modificación. Puede distribuirse como software no libre. Creada por Universidad de Berkeley.

**Software bajo licencia MPL:** Copia, modificación y distribución limitada. Pueden liberar el código manteniendo control sobre creaciones y modificaciones. Creada por Netscape. Openbravo ERP lo usa para su OBPL (adaptación de MPL bajo la que se rige el núcleo de su aplicación)

**Software semilibre:** Software no libre. autorización de uso, copia y distribución sin propósitos lucrativos pero con restricciones en uso.

**Software privativo:** Software no libre con uso, redistribución y modificación prohibidos si no es con autorización. Openbravo ERP usa esto para sus módulos comerciales (OBCL). También lo usa SAP (nombre de ERP y de empresa).

La versión Odoo 8.0 utiliza licencia AGPL (derivación de GPL para software en red de ordenadores) Odoo 9.0 LGPL que permite cerrar parte del código de forma que si se hace derivación de la obra original no se obliga a distribuir el código fuente.

También sería de comentar los contratos firmados por desarrollador o cliente, si está por encima de las licencias o si prevalecen las mismas.

Richard Stallman creó el concepto de software libre al trabajar en el proyecto GNU para crear un sistema completamente libre. Es la libertad de los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software (no necesariamente gratis).

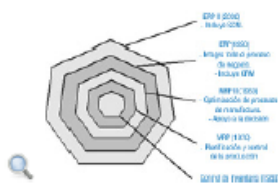
## Revisión ERP

Puede destacarse:

**Openbravo:** Aplicación de código abierto. Arquitectura cliente/servidor. Escrita en Java. Se ejecuta sobre servidor web y con soporte para bases de datos Oracle y PostgreSQL. Tiene dos versiones:

- Openbravo Community Edition: Libre y gratuita desde la que no se puede acceder a los módulos comerciales. Licencia OBPL.
  - Openbravo Network Edition: Sí se pueden acceder a módulos comerciales y actualizar código. Licencia OBCL.
- Está la:
- Suite de comercio Openbravo (minoristas)
  - Suite de negocio Openbravo. Global para empresa.

**Odoo (anteriormente OpenERP):** Resuelve problemas complejos usando soluciones sencillas. Escrito en Python y con BBDD PostgreSQL. OpenERP fue creado por informático belga en 2005.



Los antecedentes de los sistemas ERP se remontan a la Segunda Guerra Mundial, cuando el gobierno de Estados Unidos intentaba controlar la logística bélica con programas especializados. Así surgieron los primeros sistemas para la planificación de requerimiento de materiales (Material Requirements Planning Systems o MRP Systems).

Al final de los años 50 las compañías de Estados Unidos se dieron cuenta que estos sistemas MRP les permitían llevar un control de actividades como: facturación y pago, administración de nóminas y control de inventario, pero los ordenadores que utilizaban eran aún muy primitivos. Aún así durante los años 60 y 70, los sistemas MRP evolucionaron ayudando a reducir costes en lo referente al control de inventario.

Fue ya en los años 80 cuando estos sistemas pasaron a llamarse MRP II o planificación de los recursos de manufactura (Manufacturing Resource Planning) y su gestión iba mas allá del control de disponibilidad de materiales.

Así, a principios de los años 90, MRP-II evolucionó y abarcó áreas como Recursos Humanos, Finanzas, Ingeniería Gestión de Proyectos, etc, y fue esta ampliación lo que dio lugar al sistema ERP.

Actualmente, el mercado de los ERP está dominado por productos como SAP, Oracle y Microsoft, entre los que se reparten el mercado global. En España cabe destacar los ERP de Navision y Axapta entre otros, ya consolidados en nuestro país y desarrollados y diseñados en Dinamarca para un tejido empresarial muy similar al nuestro.

La mayoría de los proveedores optan por Windows como plataforma para desarrollar sus sistemas ERP. Sin embargo, Linux se está potenciando cada vez más, principalmente en las grandes empresas que son las que tienen un capital suficiente, como para poder permitirse el probar diferentes plataformas.

En este sentido, se puede hacer una primera clasificación de los sistemas ERP actuales, entre los que requieren un pago de una licencia para poder ser utilizados, es decir, propietarios, como pueden ser los productos de SAP, Microsoft, Solmicro u Openbravo entre otros, y los de software libre u Opensource como OpenERP, Odoo, Openxptiva, Tiny ERP o Abang entre otros.

Actualmente, existe una nueva tendencia de sistemas ERP denominada SaaS o software como servicio. Esta modalidad es compatible tanto con los ERP propietarios, como con los ERP Opensource. El SaaS consiste en proporcionar a los clientes el acceso al software adquirido a través de la red.

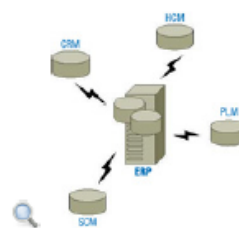
## Revisión CRM

En la actualidad los sistemas globales de CRM se pueden dividir en:

- ✓ **Aplicaciones electrónicas para los canales de distribución de la empresa:**  
Proporcionando a los canales de distribución las herramientas tecnológicas para mejorar y coordinar sus relaciones con los clientes.
- ✓ **Centros de atención telefónica (call centres):**  
Facilitan ayuda telefónica para resolver problemas y dudas a clientes.
- ✓ **Autoservicio hacia los clientes:**  
Proporciona una gestión directa de sus propios requerimientos.
- ✓ **Gestión electrónica de las actividades que afectan a clientes y ventas:**  
Proporciona información para conocer mejor las necesidades del cliente.

Además de las funciones ya vistas, los ERP pueden tener funcionalidades adicionales como:

- ✓ **CRM** (Customer Relationship Management) o administración basada en la relación con los clientes, como ya se ha visto con más detalle en apartados anteriores.
- ✓ **HCM** (Human Capital Management) o gestión del Capital Humano que tiene como objetivo llevar un mayor control de nuestros empleados conociendo su rendimiento, entre otros factores, permitiendo así automatizar al máximo el departamento de Recursos Humanos.
- ✓ **SCM** (Supply Chain Management) o administración de la cadena de suministro, controlando eficientemente las operaciones de la cadena de suministro.
- ✓ **PLM** (Product Lifecycle Management) o gestión de la vida del producto, intentando controlar todo el proceso de fabricación de productos desde la introducción al mercado hasta su lanzamiento y posterior evolución, poniendo en contacto las personas y documentos relacionadas con el desarrollo de un producto en concreto.



## 5. Instalación

Las tareas implicadas en la instalación e implantación de un ERP son:

- **Diseño de la instalación:** Estudiar necesidades de la empresa, definirlas, y ver cómo estas pueden ser resueltas por un ERP (y cuál en concreto). Tablas que es preciso adaptar, formularios e informes necesitados...
- **Instalación de equipos servidores y clientes:** Necesaria la instalación, revisión, actualización del hardware de la empresa
- **Instalación del software** Instalación de la aplicación ERP y del software que esta necesite para su correcto funcionamiento
- **Adaptación y configuración del programa:** Necesaria configuración de software y adaptación al cliente
- **Migración de datos:** Los datos son imprescindibles. A veces, si no se puede automatizar, será necesario hacerlo manualmente.
- **Realización de pruebas:** Periodo de transición en el que coexistan la solución ERP con la antigua. Pruebas en el nuevo aplicativo y migración de datos. Verificar que todo es como debe.
- **Documentación del sistema:** Documentos y manuales necesarios .Puesta a disposición de la organización.
- **Formación de usuarios:** Formación en el uso del ERP.

Mayoritariamente correrán sobre cliente-servidor. Aunque pueden estar integrados en un servidor web o usar tecnologías SaaS (Software as A Service).

Debe tenerse en cuenta:

- Disponer de máquina con prestaciones de servidor
- Instalar BBDD y conectarla
- Instalar módulos que se haya decidido adquirir
- Configurar los clientes para que accedan al servidor y puedan hacer peticiones al sistema ERP

El sistema se basa en una base de datos.

Lo habitual será incorporar el ERP en la intranet de la empresa. **En este caso es necesario un servidor web con soporte para base de datos**

Las fases de instalación ERP/CRM dependerán de la plataforma en la que se van a instalar los programas y del ERP con el que se trabajará. Los más habituales escenarios son:



- Instalación mediante máquina virtual: Máquina virtual lista para ejecutar. No apta para producción. Se usa para primera evaluación del producto.
- Instalación de paquetes bajo entorno gráfico: Aplicaciones instaladas por entorno gráfico del sistema operativo mediante asistentes. Puede usarse en producción pero los paquetes pueden ser que no estén actualizados a la última versión.
- Instalación personalizada: Descargarnos paquetes fuente desde la web e instalarlo mediante comandos. Mayor control sobre lo que se instala y sus dependencias, aunque es más complejo.
- No instalar y acceder a aplicación online: Acceder a demostraciones online del producto, conectándonos a servidor online que dispone de todos los datos y programas.

En Odoo hay diversos tipos de instalaciones. Desde la versión (8.0) donde pasa a llamarse Odoo no hay cliente de escritorio, solo se accede por el cliente web y la información se almacena en BDD. (Mayoría de ERP hacen esto)

Entre las formas de trabajo del ERP puede haber:

- **Monopuesto:** Base de datos, programas y ordenador de la aplicación cliente están en el mismo equipo. Se conecta mediante `localhost`
- **Cliente/Servidor:** Modelo cliente servidor cuando el ordenador en el que se ejecuta la aplicación cliente es distinto de donde se almacenan los datos y se ejecutan los programas. Debe indicarse la IP del servidor.

## Procesos de instalación del sistema

Si la empresa se decanta por usar SaaS no es necesario un proceso de instalación porque todo lo da el proveedor externo.

En caso contrario, sí que debe instalarse el ERP.

- Dependerá del sistema operativo. Recomendable coger la última versión. Recomendable coger la de mayor tiempo de soporte en producción.
- Asegurarse que estén activados todos los repositorios necesarios, para evitar dependencias entre paquetes sin resolver

La aplicación ERP se instalará en modo asistido, siendo necesario una intervención mínima en muchos casos.

El proceso de instalación conllevará:

- Instalación del ERP
- Instalación y configuración del servidor de base de datos que contendrá información de la empresa.
- Instalación de los servicios de acceso para los clientes

## Parámetros de configuración

Tras la instalación del ERP se lleva a cabo la configuración (establecimiento de parámetros para correcto funcionamiento).

Se podrían citar:

- Conexión con servidores: Configuración del cliente para conexión al servidor donde está la aplicación (mismo equipo, otro equipo, misma red, otra red)
- Acceso a base de datos: Acceso a los datos del sistema de gestión de bases de datos usado (local o remoto)
- Configuración del idioma: De menús y otros elementos de la aplicación.
- Localización del país: Adaptando el sistema a leyes y necesidades de cada país (Ej.: Plan General Contable español)

## Instalando Openbravo

ISO de Openbravo. Válida para producción. Instalación como si fuese servidor Ubuntu.

Virtualización VMWare. Para pruebas de evaluación. Disco virtualizado Openbravo y usar un software de virtualización (VMWare clickando sobre el .vmdk o Virtual Box con máquina virtual con indicaciones dadas por fabricante y añadir disco virtualizado).



No se necesita configuración tras la instalación. Puede accederse a la aplicación introduciendo `http://ip/openbravo` y usuario y contraseña (Openbravo, openbravo)  
Ejecutar el comentado Aplicaciones/Oficina/Openbravo ERP en el menú del sistema operativo.

## Instalando/Desinstalando Odoo

Diferentes tipos de instalaciones. En Ubuntu la forma más fácil es descargar del repositorio o paquete autoinstalable. `http://ip:8069`

Si se quiere desinstalar... `apt-get purge` en Ubuntu o en Windows por el Panel de control.  
Desinstalar también los programas asociados (que pueden llegar a impedir que se instale de nuevo Odoo.)

## Configurando servidor y base de datos Odoo

1. Archivo de configuración de la base de datos  
Lo encontramos en:

```
sudo vim /etc/postgresql/9.3/main/pg_hba.conf
```

2. Para cambios en configuración debe reiniciarse el servidor de la base de datos.  
El script de reinicio de PostgreSQL está en `/etc/init.d` y se llama `postgresql`.  
La última parte de la orden debe contener la palabra `restart`.  
En la actual Odoo no suele ser necesario meterse a cambiar el archivo de configuración manualmente, por lo que tampoco lo será reiniciar el servidor.  
Sí que si debe desinstalarse Odoo por tener una BBDD corrupta, también debe desinstalarse Postgres.
3. Manejo de la base de datos. Crear usuarios

```
sudo su postgres  
createuser userbd -pwprompt # Crear usuario userbd
```

En Windows será mediante asistente. No hará falta. Y casi que hoy día en Ubuntu tampo.

4. Modificar parámetros de conexión con la base de datos

```
sudo vim /etc/odoo.conf
```

Ahí están los parámetros: `db_name`, `db_user`, `db_password`, `log_file`  
Si se modifica debe reiniciar servidor.

```
# Cambiar contraseña de postgres  
sudo passwd postgres  
# Cambiar al usuario postgres para administrar BBDD  
su postgres  
# Crear sesion de terminal con Postgres  
psql  
# Ayuda en el terminal de postgres  
\h  
# Listar BBDD existentes  
\l  
# Salir  
\q
```

## Servicios de acceso

Servicios de acceso permiten la conexión de los usuarios al ERP.  
Suele accederse a través del navegador web del cliente.  
Debe conocerse:

- Servidor (IP)

- Puerto (8069)
- Protocolo de conexión. (XML-RPC)

### Acceso Odoo local: localhost:8069

Inicio de sesión.

- Nombre de base de datos (La primera vez se creará la BBDD)
- Email
- Contraseña
- Idioma
- Cargar datos de demostración (check)

### Acceso en otro equipo

ip\_servidor:8069

Puede hacerse ping.

Dentro de la máquina se ejecuta Odoo `opt/odoo/odoo` `./odoo-bin`

Login desde el equipo cliente

--

Al acceder a Odoo por primera vez se muestra la pantalla de Aplicaciones y se accede como usuario administrador. Desde ella se pueden instalar los módulos de los que se compone Odoo.

Se instala un módulo y aparece un nuevo menú con él relacionado.

En Open bravo se introduce `IP_servidor/openbravo`, se accede a Openbravo con Openbravo, openbravo. Para acceder al sistema operativo: openbravo openbravo.

### Actualización del sistema

Se produce normalmente de forma automática entre versiones secundarias o parches. (Misma versión base)

Si son cambios significativos (cambio de versión)

Las decisiones de cambios de versión son importantes porque afectan al funcionamiento de la empresa y a los datos. Puede requerir formación de los usuarios para adaptarse a las nuevas funcionalidades.

Analizar también si será posible traspaso de los datos del sistema antiguo al nuevo (quizás deban coexistir las dos versiones durante un tiempo)

### Asistencia técnica remota en ERP-CRM

El **soporte remoto** permite controlar un equipo físicamente como si estuviéramos sentados frente a él.

Existen programas de estructuras cliente/servidor que utilizan el VNC (VNC viewer/server).

Cliente es el equipo desde el que controlamos el servidor.

Servidor es el ordenador a controlar.

Al ejecutar el VNC en el equipo cliente se muestra el escritorio del servidor como si estuviésemos físicamente delante. Así puede revisarse aspectos del ERP, modificar o corregir lo que sea necesario, instalar nuevos programas, ejecutar los existentes.

- Windows: RealVNC, UltraVNC, TightVNC, Anydesk
- GNU/Linux: VINO(servidor), RealVNC, TightVNC
- Macintosh: VineSever(servidor), chicken of the VNC (cliente)

## Instalación configuración en Ubuntu

Ubuntu incorpora sistema de asistencia remota por VNC gracias al servidor VINO.  
Servidor remoto VINO se configura en Sistema / Preferencias / Escritorio remoto.

Una vez hecho esto, para acceder remotamente se usa el cliente VNC en otro equipo:

```
sudo apt-get install vncviewer  
vncviewer IP_servidor
```

En caso de ser un Windows, usar una aplicación cliente.

Si están en la misma red, usar la IP privada del servidor (ipconfig)

Si están en redes separadas por un router, usar la IP pública que asigna el operador de la línea. El router dirigirá la petición del cliente VNC al equipo donde se esté ejecutando el servidor VNC.

El ejecutar el cliente VNC, si se especificó en VINO, se pedirá contraseña para conectar.

## Instalación y configuración en Windows

UltraVNC/UltraVNC Server.

Contraseña VNCPassword de conexión.

La existencia de firewall podría bloquear estas conexiones.

Si hay equipos en diferentes redes deben abrirse los puertos 5900-5999 para permitir entrada VNC desde el exterior.

## Conexión remota por VNC inverso

En vez de solicitar conexión el cliente al servidor la solicita el servidor. El cliente está a la escucha y el servidor la establece.

Es más fácil para el cliente porque no tiene que preocuparse de a donde conectarse. Es alternativa cuando el VNC normal no permite la conexión o si hay un cortafuegos.

Se ejecuta el visor VNC en el cliente en modo escucha `vncviewer -liste`

En el servidor se ejecuta UltraVNC server. Desde él servidor se establece la conexión con el cliente. Add new client, Host Name Ip-servidor:0. Si los puertos en el router estan correctos aparece ventana en el cliente con el escritorio del ordenador remoto.

## Conexión por SSH

Administrar el equipo de forma remota es posible por SSH. Permite realizar conexiones en modo terminal.

La más importante es la desarrollada por OpenSSH.

```
sudo apt-get install openssh-server  
ssh usuario@servidorssh
```

## Herramientas para programar en sistemas ERP/CRM

Conocido el sistema funcional del ERP y realizada la instalación del mismo, prosigue la puesta en funcionamiento, adaptación al entorno productivo e incorporación de nuevas funcionalidades.

Esta adición se hace mediante programación de nuevos módulos. Interesante considerar cuál será el entorno de programación a emplear.

Netbeans, Eclipse, Dia (diagramas UML, se pueden crear módulos uml-dia en Python y ejecutarse como módulos de OpenERP), Gedit (ficheros de texto para programar, posibilidad de crear módulos con un plugin con facilidad), Open Office (crear y modificar informes, campos, expresiones), iReport (diseñador de informes para el ERP. Facilita creación de informes en JasperReport al ser visual, intuitiva y de fácil manejo)

## 6. Personalización del ERP

Personalización: Adecuarlo al entorno de la empresa que lo usa.

Meter el logo de la compañía, dar de alta a los usuarios con los permisos adecuados, configurar sistema de avisos, herramientas de correo...

Los sistemas ERP permiten la incorporación de módulos predefinidos. Se llama **módulo** al programa que se realiza para cubrir una función de la aplicación. Existen módulos básicos que se pueden cargar durante la instalación del sistema y otros posteriormente desde el programa o la web del ERP. Estos módulos pueden realizarse tanto en el momento inicial como en un proceso posterior. Los recursos son muy variados (creación de informes, comunicación con plataformas móviles, interconexión con paquete ofimático de la empresa). La mayoría tienen CRM integrado o puede incorporarse como módulo.

Módulos más importantes:

- **Gestión de las relaciones con el cliente (Ventas / Marketing):** Interfaz que interactúa con clientes, pedidos, estrategias de ventas, precios, promociones, publicidad... CRM (Customer Relationship Management) todo lo relativo a la relación comercial desde creación de ficha de cliente hasta determinación de productos que le interesen, pedidos, ventas, contactos...
- **Gestión contable y financiera (Finanzas):** Base del ERP. Almacena cada transacción facilitando auditorías. Debe estar integrado con compras y ventas para evitar duplicidad y tener información en tiempo real. Facturas, albaranes...
- **Compra, ventas y almacén (Inventario / Logística):** Stocks y flujos de entrada salida. Operaciones de solicitud de presupuestos, recepción de precios, creación de pedidos. En Almacén se pueden gestionar las existencias de productos.
- **Gestión de personal (Recursos humanos):** Personal, nóminas, productividad, incentivos, beneficios... Control de horarios. Altas. Bajas. Contratos.
- **Facturación (Producción):** Movimientos físicos de los artículos, planificación de materiales... Facturación de productos y servicios a los clientes. Facturas de ventas, albaranes, tarifas. Formas de cobro y pago del cliente...

## 7. Seguridad del ERP: Planificación, usuarios y roles

Para aumentar la seguridad deben analizarse los riesgos a los que se somete el sistema:

- Riesgos físicos: Fallos de componentes electrónicos por agresiones externas
- Riesgos lógicos: Falta de política adecuada en sistemas, accesos no autorizados, bugs, intrusiones externas...

Medidas de seguridad:

- Niveles de acceso configurables para los usuarios según su rol
- Autoría de cada transacción
- Soporte para conexión segura por HTTPS

Importante una buena asignación del rol a cada usuario del sistema.

Usuarios posibles:

- Administrador: Todos los privilegios y toda la información sobre el acceso y la gestión
- Usuario normal: Sin privilegios de administrador pero sí sobre la información
- Usuario de grupo: Correo entrante para distribución
- Usuario de portal: Acceder a portales creados en el entorno pero no a la aplicación

Hay aplicaciones que no pueden eliminarse directamente. Debe hacerse desde la BBDD, aunque siempre permiten desactivarlos.

También conocer los roles:

- Rol es grupo particular de privilegios
- El rol tiene validez al estar asignado a un usuario

- El usuario puede tener varios roles, prevaleciendo el más restrictivo
- Cambios en roles no son efectivos hasta no iniciar otra sesión
- Cuando un rol niega acceso a un módulo se pierde posibilidad de ver el subpanel del mismo.

# UD 02 - USO DEL SISTEMA ERP-CRM

## 1. Primeros pasos con el ERP

### 1.1. Acceso al ERP

Accedemos a Odoo en **IpServidor:8069**.

Debe estar arrancado `/opt/odoo/odoo` `./odoo-bin`

Se muestra la pantalla de Aplicaciones y se accede como Administrador.  
Se pueden instalar los módulos de los que se compone Odoo.

Al abrirlo por primera vez solicita:

- Nombre base dedatos
- Usuario o correo
- Idioma
- Contraseña de inicio de sesión

Se pulsa en "Crear base de datos"

Debe tenerse localizado el archivo de configuración de Odoo.

Reinicio:

En Windows se puede acceder a los servicios locales y allí buscar odoo-server y con eso detenerlo y reiniciar

### 1.2. Definición del módulo

Módulo: Programa que se realiza para cubrir una función de la aplicación.  
Toda la funcionalidad de un ERP está contenida en sus distintos módulos.

Los módulos se caracterizan por:

- Instalación - desinstalación mediante asistentes
- Configuración o parametrización de los módulos para su adaptación al entorno de producción
- Generación de informes por cada módulo
- Incorporación de niveles de seguridad (módulos solo accesibles por administrador)
- Interconexión entre los módulos, información compartida entre ellos
- Posibilidad de incluir textos y comentarios en las opciones del programa
- Adaptación de menús de los módulos de las necesidades de cada usuario

### 1.3. Entorno modular de un ERP

El diseño del ERP es modular. Hay un módulo o conjunto de módulos base necesarios para que funcione la aplicación, alrededor de los que se sitúan módulos adicionales que se van desarrollando según la necesidad concreta.

## 2. Tipos de módulos

La **funcionalidad o módulo base\*** normalmente contempla los siguientes aspectos:

- Configuración de la aplicación
- Gestión de los datos maestros Introducción de datos para que funcionen los procesos de gestión
- Establecimiento del idioma o importación de traducciones
- Seguridad: Gestión de usuarios y accesos

- Administración de módulos: Para instalar nuevos módulos.

Adicionalmente al módulo base destacan:

- Gestión contable y financiera
- Operaciones de compra: Compras y almacén
- Operaciones de venta
- Facturación, cobros y pagos
- Gestión comercial
- Gestión de personal
- Gestión de las relaciones con el cliente (CRM, Customer Relationship Management)
- Productos
- Inventario
- Proveedores
- Gestión de proyectos
- Gestión de informes y estadísticas

## Contabilidad y finanzas (Account)

Recoge y automatiza todas las operaciones contables de la compañía centralizándolas para su consulta publicación o control.

Cubre la contabilidad general, análisis contable y costes, gestión de impuestos, presupuestos, facturas, tanto de clientes como de proveedores.

**Debe estar integrado con los módulos de compras y ventas para evitar duplicidades.** Así también se pueden obtener facturas de clientes y proveedores desde pedidos de venta y compra o albaranes de salida y entrada, respectivamente.

También **debe estar integrado con el resto de módulos del ERP para realizar una gestión integral de la compañía.**

Las facturas tienen asientos contables. A través de los asientos contables se reflejan movimientos entre cuentas de una empresa. Movimiento de la cuenta bancaria a la cuenta de efectivo para tener más dinero en Caja.

<http://www.youtube.com/embed/01BUNIA5UQU>

<http://www.youtube.com/embed/Qu6R3yNKR60> (No precargado Odoo 12 Accounting report)

<http://www.youtube.com/embed/RS8zTRBy82U>

Las funcionalidades básicas de este módulo son:

- Contabilidad general.
- Contabilidad analítica / costes.
- Gestión de impuestos.
- Presupuestos.
- Facturas de clientes y proveedores.
- Extractos de cuentas bancarias.
- Informes contables.

Para la realización de tareas contables en el ERP será necesario:

- Completar la información de la empresa.
- Introducir las cuentas bancarias.
- Selección del Plan de Cuentas.
- Selección de los impuestos necesarios para la gestión.
- Datos de clientes, proveedores y productos.
- Saldos iniciales.
- Definición de usuarios para la contabilidad.

## Compras (Purchase Management)



Aprovisionamiento para poder vender. Operaciones de solicitudes de presupuesto a proveedor, recepción de precios, creación de pedidos de compra, con indicación de precios, plazos de entrega...

Este módulo automatizar el flujo de trabajo de las órdenes de compra, incluye la gestión de proveedores, y el control de productos y facturas, en lo relativo a la generación de facturas en base a órdenes de compra o productos recibidos, creación manual de facturas e importar líneas desde órdenes de compra, etc.

- Seguimiento de tarifas de sus proveedores.
- Conversión de tarifas en órdenes de compra.
- Gestionar entregas parciales del proveedor.
- Gestión de reclamaciones a proveedor.
- Generación automática de borradores de pedidos de compra.

En Odoo las órdenes de compra pueden crearse como tal o pueden provenir de Solicitudes de Cotización o de Licitaciones de Compra. Las órdenes de compra generarán una factura del proveedor. Mediante el menú Compra/Pedidos de Compras. Crearemos un nuevo presupuesto que se convertirá en un pedido de compras.

## Ventas (Sale)

Este módulo, eje principal del ERP, gestiona todas las operaciones relacionadas con la venta de los productos de la empresa, como la creación de órdenes de venta, facturas de venta configurables, facturas en reparto, gestión de stock disponible, etc.

Similar a compras pero para ventas.

- Creación de pedidos de venta.
- Revisión de los pedidos en sus distintos estados.
- Confirmación de envío.
- Definición de formas de pago por pedido y fecha de facturación.
- Gestión y cálculo de gastos de envío de un pedido.
- Albaranes automáticos desde pedido.
- Albaranes de envíos parciales.

## Facturación

Generación de datos que tengan que ver con facturación de productos y servicios a los clientes: facturas de venta, albaranes, tarifas. Contempladas diversas formas de cobro y pago.

- Configuración de formas de pago de Clientes o cobro de proveedores.
- Facturas automáticas desde pedido o albarán.
- Generación automática de efectos de cobro y pago.
- Gestión de recibos, órdenes de pago y transferencias.
- Importación de extractos bancarios.
- Envío telemático de remesas al banco.
- Gestión de bancos propios, bancos de Clientes y bancos de proveedores.

En la mayoría de las ocasiones las facturas son generadas automáticamente desde diferentes procesos del sistema, aunque también es posible generarlas manualmente. De esta forma, no son necesarias crearlas manualmente, sino que los diferentes procesos generan facturas en Borrador, y éstas deben ser aprobadas por el usuario de sistema que corresponda y enviadas al cliente.

Existen diferentes maneras de automatizar la creación de la factura del cliente en Odoo. Dependiendo de las características de la empresa que utilicemos, podemos optar por una de las siguientes maneras para crear facturas:

- **Orden de venta -> Factura.** La factura se crea basándose en una orden de venta.
- **Orden de venta -> Entrega -> Factura.** En lugar de facturar basándose en una orden de venta, se factura basándose en cantidades entregadas. De esta manera, se permiten órdenes de venta parciales, facturándose sólo lo que se ha entregado.

- **Suscripciones -> Facturas.** Para suscripciones, una factura se activa periódicamente, de forma automática. La frecuencia de la facturación y de los servicios/productos facturados están definidos en el contrato.
- **Orden de comercio electrónico -> Factura.** Cuando se trabaja con comercio electrónico la factura se activará una vez que se recibe el pago.
- **Creación manual de la factura.** También se pueden crear facturas manuales sin utilizar ninguno de los métodos anteriores. Por ejemplo, para reembolsos, para hacer descuentos o facturar algo no relacionado con el negocio principal de la empresa.

## Almacén / Inventario (Inventory Management)

Módulo para la gestión del almacén, nos proporciona diferentes métodos de inventario, gestiona el valor del stock, además podemos crear reglas de reordenación automáticas, gestionando la historia y la planificación

Permite gestionar las existencias de productos en almacén.

- Definición de múltiples almacenes.
- Gestión de la rotación de inventario y niveles de stock.
- Traspasos entre almacenes.
- Codificar y numerar productos de distinta forma.
- Definir compras de un producto a distintos proveedores.

## Gestión de personal

Planificación y realización de nóminas de los empleados, altas, bajas, contratos, control de horarios, datos de personal, sistema de remuneraciones, comisiones...

- Gestión de empleados y calendario de vacaciones.
- Gestión de contratos de empleados.
- Gestión de beneficios.
- Gestión de ausencias.
- Gestión de producción o rendimiento.
- Gestión de perfiles y responsabilidades.

Podemos incluir aquí un conjunto de módulos que permiten gestionar los recursos humanos de una empresa. Destacan el módulo de Directorio de Empleados (Employee Directory), Gestión de vacaciones y ausencias (Leave Management) o el de Productividad (Productivity) que proporciona herramientas para priorizar tareas, compartir documentos e incrementar la productividad del empleado.

A veces no tendrá módulo de RRHH y será llevado con conceptos contables relacionados y gestión de comisiones a través del módulo comercial.

## Gestión de las relaciones con el cliente (CRM)

CRM (Customer Relationship Management) es muy importante en el ERP. Incluye todo lo relacionado con la relación comercial con clientes o posibles clientes.

Busca tener información **centralizada** para optimizar los procesos de gestión.

Hay aplicaciones únicamente destinadas a la gestión CRM. En ERP Odoo existe un módulo independiente de CRM.

Módulo para la gestión de las relaciones con los clientes, gestionando la adecuada atención a los ya existentes, así como, la gestión eficiente de clientes potenciales, oportunidades, peticiones, campañas, etc. Gestiona tareas como la comunicación, asignación, resolución y notificación. Con este módulo nos aseguramos que todos los casos de nuestros clientes obtengan una atención y seguimiento igual de adecuado. Además, extrae información precisa de todos los datos registrados para la ayuda en la toma de decisiones.

- Datos identificativos del contacto.
- Segmentación de clientes en función de múltiples criterios.
- Determinación de clientes reales y potenciales.

- Gestión de llamadas.
- Calendario de encuentros.
- Generación y seguimiento de campañas de marketing.
- Seguimiento de acciones comerciales.
- Enlace con otros documentos y procesos de la aplicación.
- Herramientas de productividad: editor de documentos, sincronización de contactos y calendario, envíos masivos por correo electrónico, mensajería sms o fax, etc.
- Estadísticas diversas.

Los módulos más avanzados de gestión de las relaciones con el cliente pueden incluso incorporar una **Extranet**, para la conexión por parte de clientes (y proveedores) con el sistema de la empresa, y así poder consultar la información a la que ésta les dé acceso.

El módulo de CRM en Odoo funciona creando flujos de venta para un cliente dado. Utiliza diversas herramientas en las relaciones con los clientes que deberemos configurar para adaptarlas a nuestra empresa.

Un flujo de venta está definido por un importe estimado de venta, la probabilidad de realizarla y el cliente relacionado. El flujo de venta pasa por los siguientes estados (pueden ser definidos por el usuario):

- **Nuevo:** Estado del flujo en el momento de su creación.
- **Calificado:** Una vez validado el flujo de venta pasa al estado calificado.
- **Propuesta:** Se realiza un presupuesto basado en ese flujo.
- **Negociación:** El flujo pasa al estado de negociación cuando el presupuesto es aprobado por el cliente y se quieren negociar los términos de la venta.
- **Ganado:** Cuando el presupuesto se materializa en una venta.

## Fabricación (Manufacturing))

Módulo para la gestión de la fabricación. Incluye la creación de lista de materiales, procesar órdenes de fabricación, vender conjunto de productos como un kit o la gestión de productos semiacabados.

## Punto de venta (Point of sale, TPV)

Este módulo incluye varios de los procesos de la empresa como son creación de una orden de compra, una devolución o la modificación de la forma de pago en el momento mismo de realizar la venta, simplificando operaciones tales como el cálculo automático de la devolución de dinero en una venta, o permitir al usuario la creación de la factura de forma automática, entre otras.

## Campañas publicitarias (Mass Mailing Campaigns)

Gestionaremos las campañas publicitarias, utilizando varios métodos de comunicación, creando campañas personalizadas, de forma que aumenten las ventas y se promueva la empresa.

## Comercio electrónico (e-commerce)

Nos suministra las herramientas necesarias para hacer una proyección mayor de los artículos y productos de nuestra empresa, mostrando de forma muy visual las características de los mismos, conectando con los clientes a través de internet. Permite crear filtros y categorías de productos, así como todas sus características, logotipo, etc. Actualmente un módulo imprescindible en nuestro ERP.

## Constructor de sitios web (Website Builder)

Nos permite el diseño de sitios web de forma simple. Nuestra web así podrá integrar herramientas como e-commerce, blog, etc.

## Localización española

Adaptar ERP a las leyes y normas de cada país en:

- Plan General Contable Español
- Módulo de Impuestos, tipos de IVA
- Validación de datos (CIF, NIF, cuentas bancarias)
- Inclusión de datos maestros, datos sobre provincias de España
- Traducción al español

### 3. Operaciones de gestión y consulta de la información

La información contenida en ERP se trata con SGBD (DBMS) que ofrecen programas para acceder y gestionar los datos.

La base de datos del ERP es de gran tamaño. Almacena datos, vistas, otros elementos. Es necesaria una organización entre sus componentes (nomenclatura o normativas que deben seguir los desarrolladores a la hora de modificar el código fuente o esquema: colocar un prefijo para saber a qué modulo pertenecen, establecer una serie de campos como obligatorios)

#### 3.1. Elementos de una base de datos

Los conceptos habituales:

- Tablas
- Campos o atributos
- Filas o registros
- Vista
- Relación
- Consulta
- Procedimiento almacenado: Función o procedimiento almacenado físicamente en una BBDD que hace una tarea simple sobre los datos (disparadores, triggers)
- Formulario: Documento digital para manejar los datos, almacenándolos y procesándolos posteriormente
- Informe: Exposición de los datos presentados de forma fácil de analizar e imprimir

#### 3.2. Administración de las bases de datos

Odoo dispone forma de administrar la base de datos del ERP aunque se puede hacer también desde el gestor de base de datos (como pgAdmin).

Necesario introducir:

- Nombre de conexión
- Servidor
- Puerto
- Base de datos de Mantenimiento (base de datos inicial con la que nos conectamos)
- Nombre de usuario
- Clave

#### 3.3. Consultas de acceso a datos

Para acceder a la información, solicitando un extracto.

Necesario:

- Seleccionar las tablas o vistas sobre las que va a actuar la consulta
- Establecer relación entre tablas o vistas en caso de que la aplicación no la proporcione
- Seleccionar campos a mostrar
- Ejecutar

### 4. Técnicas de optimización de consultas

Optimización: Modificar un sistema para mejorar su eficiencia o el uso de los recursos disponibles.

Al manejar grandes cantidades de datos el resultado puede tomar un tiempo considerable y ser poco óptimo. Como técnicas de optimización mencionamos:

- **Diseño de tablas:** Asegurar que no hay duplicidad de datos y que se aprovecha el almacenamiento
- **Campos:** Ajustar el espacio de los campos al máximo para no desperdiciar
- **Índices:** Búsquedas a velocidad superior pero no debemos indexarlo todo porque ocupan más y tardan en actualizar. Podría estudiarse si deben usarse si es un campo usado como criterio de búsqueda o si es una clave ajena en otra tabla.
- **Optimizar sentencias**
- **Optimizar la base de datos**

## 4.1. Conexión directa con PostgreSQL

```
sudo su postgres
psql
\h # Ayuda
\l # Bases de datos existentes
\c [nombre_bd] # Conectar con la BBDD
\d # Tablas existentes en la BBDD
\d [nombre_tabla] # Descripción de la tabla
VACUUM VERBOSE ANALYZE [tabla] # Limpia y analiza Bases de datos
\q # Salir
```

## 4.2. Sistemas batch inputs. Generación de programas de extracción y procesamiento de datos

**Batch-input** procede del el uso en sistemas SAP de un método usado para transferir grandes cantidades de datos a un sistema ERP. Requieren programas tanto par la generación del archivo como para el procesamiento posterior.

Hay dos forma de hacer un batch-input:

- **Método clásico:** Asíncrono. Se procesan los datos pero se actualizan más tarde. Genera archivo de mensajes o logs para tratar errores a posteriori.
- **Método call transaction:** Método online usado para dar de alta rápidamente pocos registros. Es síncrono, no genera logs y es mucho más rápido pero solo sirve para pocos datos.

El proceso de batch-input tiene dos fases:

- **Fase de generación:** Genera el archivo batch-input con los datos a introducir o modificar
- **Fase de procesamiento:** El archivo batch-input se ejecuta haciéndose efectivas las modificaciones en base de datos

## 5. Extracción de datos en sistemas ERP

La extracción de datos puede hacerse usando diferentes sistemas.- Muchas veces con herramientas ofimáticas que se conectan a la aplicación ERP para obtener información de la BBDD y volcarla en la aplicación.

Hay procesos más complicados (Procesos de Business Intelligence) que:

- transforman y combinan los datos para extraer la información
- la convierten en potentes indicadores
- la muestran en diversos formatos gráficos

Según el origen de los datos y el tipo de información se pueden usar:

- **Consultas e informes:** Datos en una sola base de datos y se extraen a partir e una consulta SQL. La aplicación facilita informes predefinidos aunque se pueden generar personalizados usando la aplicación o herramientas externas como JasperReports

- **Almacenes de datos:** La extracción de datos se hace desde diferentes sistemas y distintas bases de datos, creando almacenes de datos para homogeneizar e integrar la información.
- **Cubos multidimensionales:** Un cubo n-dimensional es un conjunto de datos multidimensionales organizados en ejes y celdas que maneja la información de la base de datos relacional. También hay bases de datos multidimensionales (sin necesidad de almacenar primero en relacionales y luego usar cubos)

En ocasiones la extracción no se hace en tiempo real para evitar disminución en la velocidad de respuesta. Primero se extrae la información y luego es manipulada podría haber alguna leve diferencia entre la información manipulada y el contenido en BBDD.

## 5.1. Importar y exportar datos

La importación y exportación de datos se hace a través de los mecanismos provistos por la aplicación. Muchas veces en formato CSV en el que las columnas se separan por comas (o punto y coma donde la coma sea separador decimal) y las filas por saltos de línea.

La herramienta debe permitir seleccionar los campos a importar y volcar dicha información en el objeto deseado. Se podrán importar en una sola tabla o varias.

La exportación igual: En el objeto a exportar se elige "Exportar" y se genera el archivo que puede abrirse por cualquier aplicación ofimática o, en el caso del CSV, incluso por editor de textos sencillo.

Se puede exportar datos desde los formularios también. Permitiendo más campos a exportar que la vista de árbol.

## 5.2. Copias de seguridad

Deben programarse copias de seguridad de las bases de datos periódicamente.

Odoo da un Gestor de bases de datos al que puede accederse permitiendo realizar copia completa en formato comprimido.

Se podría hacer copia de seguridad y tener entorno de producción y otro de pruebas.

También se pueden hacer a través del gestor PgAdmin.

## 6. Auditoría y control

Debe haber herramientas que permitan hacer un seguimiento de los datos del equipo servidor donde están las aplicaciones.

Datos instantáneos del rendimiento del sistema (procesador, memoria, dispositivos entrada salida) o recogerlos y almacenarlos en ficheros históricos que informen sobre carencias y cuellos de botella del sistema.

### 6.1. Auditoría por ficheros

Odoo tiene su fichero de configuración `odoo.conf` y allí están los parámetros:

- `syslog`: Linux auditoría
  - `logrotate`: Sistema rote los archivos de log
  - `logfile`: Localización de ficheros de log
- En la carpeta del servidor está PostgreSQL y en el fichero de configuración de la base de datos `postgresql.conf` se describe la terminología para los distintos parámetros:
- FILE LOCATIONS: Colocar archivos de configuración en otro lugar
  - ERROR\_REPORTING\_AND\_LOGGING Configurar aspectos sobre creación de ficheros de auditoría con métodos como stderr, csvlog, syslog (eventlog en Windows), así se puede establecer lista de destinos de registros separados por comas; por defecto solo se registrará en stderr.
  - RUNTIME STATISTICS: Estadísticas de uso sobre el sistema para determinar el rendimiento.

### 6.2. Control del rendimiento

Se permiten instalar módulos para recoger datos y mostrar informes de análisis y monitorización del rendimiento en el sistema ERP.

### 6.3. Trazas del sistema

La actividad queda registrada en los logs. Puede querer examinarse las trazas para realizar control de acceso (quien entra, qué comandos ejecuta, qué errores muestran) También para resolver posibles problemas ante fallos de funcionamiento.

El fichero `odoo.log`

- En Linux se guardan en `/var/log`
- En Windows se encuentra en el directorio base de instalación en la carpeta server.

Serán necesarios permisos de root para poder visualizar estos ficheros.



# UD 03 - IMPLANTACIÓN DE SISTEMAS ERP-CRM

## 1. Introducción

Se hace importante hacer un estudio de las necesidades y requisitos para la adquisición de un ERP (Identificación de los procesos más importantes de la empresa y de qué forma pueden llevarse a cabo). Se podrían contratar a consultoras externas que analicen y hagan un informe final con recomendaciones de implantación del software para, con este estudio, tomar la decisión de adquirir una u otra aplicación.

Las fases del **proceso de selección, implantación y puesta en marcha** del ERP se pueden definir como:

- **Selección del ERP:** Basándose en los procesos clave de la empresa, qué tareas se repiten y pueden ser automatizadas, qué necesidades son deseables en el nuevo sistema informático, qué módulos pueden responder a estas necesidades. (Empresa o consultora externa)  
(Por el proveedor del ERP)
- **Implantación:** Cambios y adaptaciones en la aplicación que necesite la empresa.
- **Puerta en marcha:** Instalación del programa en producción y resolución de problemas de implantación
- **Cierre y finalización:** Revisión final del sistema, comprobando su funcionamiento.

Software como servicio: Servicio ofrecido por proveedor en forma de suscripción mensual que incluye equipos servidores, mantenimiento del sistema, hosting y soporte. Para pequeñas empresas que por bajo costo tienen completo e integrado sistema de gestión.

### 1.1. Tipos y necesidades de las empresas

Empresas susceptibles de implantar un ERP son:

- **Pequeña y mediana empresa:** Gestión de clientes, proveedores, productos; Compras, ventas y almacén.
- **Sector servicios:** Gestión por proyectos (Módulo específico control y seguimiento de proyectos)
- **Tiendas y restaurantes:** Venta de productos con terminales de puntos de venta en lectores de código de barras o dispositivo táctil. Interfaz táctil y amigable. Categorías de productos, múltiples pedidos y métodos de pago.
- **Ayuntamientos:** Gestión de proyectos y contabilidad en departamentos, control de compras y stocks, recursos humanos, atención al ciudadano (CRM) con portales del municipio, padrón, gestión de tasas
- **Venta telefónica:** Módulo de CRM importante para la información resultante del contacto telefónico con el cliente.

## 2. Selección del sistema ERP

Análisis previo -> Elección del ERP y de los módulos que mejor se adaptan a los requerimientos.

### 2.1. Análisis inicial

En el análisis inicial se estudia cómo funcionan las áreas de la empresa (Compras, Ventas, Marketing y gestión de relaciones con el cliente, Logística y Recursos humanos).

Se cubren:

1. **Estructura de información o datos maestros:** Datos que necesita la aplicación para trabajar con el sistema
2. **Procesos de negocio:** Estudio de los procesos o tareas de cada área de la empresa y de qué herramientas usan para su comunicación. Verificar si los procesos del ERP se adaptan con los requeridos

3. **Informes necesarios:** Informes dentro de los que incorpore en el ERP o nuevos que se adapten a los requisitos
4. **Traspaso de información:** Analizar la migración de los sistemas de gestión que usa la empresa al nuevo ERP considerando la estructura y característica de los datos a traspasar, campos que necesita el ERP para funcionar, verificar introducción de los datos necesarios para que funcione el ERP.
5. **Planificación de la implantación:** Adecuada gestión de implantación para que sea sistemática y organizada.  
Reflejar cada proceso e información a registrar y centralizar.  
Forma de gestionar la implantación como proyecto dentro de toda la estructura de la empresa.

## 2.2. Módulos a conocer

¿Qué módulos del ERP se pueden usar?

Pueden ser

- **Módulo base:** Se instala con la aplicación. Opciones mínimas para funcionar
- **Módulos precargados:** Cargados durante la instalación del sistema. Están disponibles para ser instalados en cualquier momento
- **Módulos no precargados:** No aparecen en la lista de módulos, necesitan primero ser cargados en la aplicación.
- También hay **módulos especiales de un país** como los archivos de localización española para llevar a cabo tareas comunes de cualquier empresa como ventas, compras, productos, almacén, contabilidad, facturación, plan contable...

Tras el análisis inicial se plantearán qué módulos deben activarse para cubrir todos los aspectos funcionales.

Los más destacables son:

- **Accounting and Finance - Contabilidad y finanzas :**
- **CRM - Gestión de clientes:**
- **Sale - Ventas:**
- **Purchase Management - Compras:**
- **Inventory Management - Inventario:**
- **Manufacturing - Fabricación:**
- **Point of sale - Punto de venta:**
- **Recursos humanos:**
- **Mass Mailing Campaigns - Campañas publicitarias:**
- **E-commerce - Comercio electrónico:**
- **Website Builder - Construcción de sitios web (CMS, Sistema gestor de contenidos):**

## 3. Módulos no precargados

Los módulos se muestran con color relativo al tema del módulo y botón en color gris y claro desactivado con la inscripción instalado.

Los que pueden instalarse aparecen con "Instalar". Simplemente debe instalarse y configurar los parámetros necesarios con la información necesaria.

### 3.1. Respaldo del sistema

Antes de modificar el ERP es conveniente hacer copia de seguridad del sistema por si el nuevo módulo deja el sistema inestable. Gestionar bases de datos > Copia (Backup)

El archivo comprimido se guardará en nuestro equipo y se restaurará, dado el caso, con Restore Database.

### 3.2. Instalación de módulos no protegidos

## Descargarlo en internet y cargarlo en el sistema

Se descarga el módulo desde la web de aplicaciones <http://www.odoo.com/apps>

Se descarga módulo con Download y después se copia descomprimido en el directorio addons de Odoo.

En el modo desarrollador de Odoo se clicka en Actualizar lista de aplicaciones y ya entonces se podrá ver.

## 4. Distribución de módulos

También se pueden descargar módulos no precargados mediante el sistema de control de versiones. Odoo Community Association (OCA) coordina las contribuciones mediante varios repositorios de Github en <https://github.com/OCA>

### 4.1. Sistemas de control de versiones

Git funciona con flujos de trabajo compuestos por tres estados:

- Modificado: Cambios en un archivo
- Preparado: Se le indica a Git que lo hemos modificado
- Confirmado: Se confirman los cambios

### 4.2. Gestionar repositorios

Absurdo describirlo.

## 5. Catálogo de la base de datos

El **catálogo del sistema** es el lugar donde el SGBD del ERP almacena la información.

Está formado por metadatos sobre las estructuras de datos e información interna para el funcionamiento.

Hay dos fuentes:

- `information_schema`: estructuras definidas según ANSI\_SQL, ubicadas en el esquema propio
- `pgcatalog`: *Catalogos propios de PostgreSQL. Estructuras particulares de PostgreSQL que pueden cambiar según la versión usada y se identifican con el prefijo pg\**

El catálogo `pg_database` almacena información sobre las BBDD disponibles.

No suelen modificarse manualmente sino con instrucciones SQL (ej.: CREATE DATABASE inserta fila en `pg_database` y crea BBDD en disco)

### 5.1. Estructura de la base de datos

Con el ERP puede accederse a la estructura de la base de datos desde el menú de "Técnico / Estructura de la base de datos" solo visible entrando en modo desarrollador.

Destacan las pestañas:

- **Modelos**: Modelos de BBDD y crear nuevos
- **Campos**: Campos asociados a los modelos, crear, editar
- **Restricciones del modelo**
- **Relaciones many2many**: Relaciones establecidas, nombre y objeto
- **Modelos referenciables**: Referenciar con nombre los objetos. Ej.: `account.invoice` se referencia con nombre Factura
- **Precisión decimal** de los campos numéricos con los que trabaja la base de datos.

## 6. Implantación en la empresa

En el Proyecto de Implantación de un ERP se incluye:

- La adaptación de la aplicación a los requerimientos

- Formación de usuarios
- Traspaso de datos
- Configuración del programa
- Pruebas de usuarios
- Pruebas definitivas y revisión

#### **Riesgos de implantación e integración de ERP:**

- Finalización fuera del plazo previsto
- Sobrepasar presupuesto asignado
- Funcionamiento no esperado
- Acontecimientos imprevistos que impidan el desarrollo del proyecto

Llevar a cabo un proyecto de implantación ordenado y controlado siempre es recomendable, ya que minimiza los riesgos de implantación del ERP.

Debe hacerse adecuada gestión del riesgo para solucionar posibles problemas. Debe llevarse con éxito en la parte técnica pero la metodología también debe ser aceptada por los usuarios o igualmente supondrá un fracaso.

### **6.1. Consultas necesarias para obtener información**

En la implantación el proveedor de ERP debe:

- Diseño adaptación del programa
- Puesta en marcha
- Soporte

Si el análisis llevado a cabo antes de la selección del ERP fue exhaustivo, gran parte de la información recabada sirve para confeccionar los requerimientos necesarios en la implantación. Aún así recomendable consultar Datos de la empresa, Clientes, Proveedores, Productos, Almacén, información de Compra-Venta (tarifas, formas de pago), Información financiera (plan contable, impuestos).

### **6.2. Creación de objetos en el ERP**

Puede tener que añadirse campos a objetos existentes o crear objetos nuevos. Puede ser que se necesiten bases de datos diferentes, cada BBDD es una empresa distinta.

La modificación de objetos puede hacerse

- Con pgAdmin tabla, columna, clave foránea, regla, evento
- Desde Odoo con Técnico/Estructura de la base de datos / Modelos para introducir nombre del objeto, objeto, descripción de los campos, tipo de los campos, permiso de acceso

### **6.3. Creación de informes y gráficos personalizados**

Diferenciamos entre:

- Informes estadísticos: Informes y gráficos dinámicos según las opciones que seleccionemos para ser mostrados por pantalla
- Documentos imprimibles: Documento PDF a partir de los datos seleccionados. Pueden generarse con el lenguaje de programación de la aplicación, con herramientas ofimáticas para descargar, modificar y subir o con un motor de informes (exportar el objeto desde la aplicación y a partir de ahí construir el informe, caso de JasperReports con iReport)

### **6.4. Traspaso de datos**

Necesario volcar toda la información del sistema antiguo al nuevo ERP. Es muy importante. La información de la empresa es muy valiosa y la mala gestión puede paralizar toda la organización. Debe estudiarse el formato de almacenamiento del software origen y destino para llevar a cabo una buena importación (conseguir emparejamiento entre ambos)

Tareas a realizar:

- Unificar formato y contenido de los datos: Unir información dispersa o distribuida en un único archivo
- Eliminar duplicidad de datos
- Mejorar codificación de la información: Introducir o corregir cosas que falten para conseguir mayor calidad de datos.
- Guardar los datos con el formato de exportación elegido
- Introducción de datos de las tablas secundarias: si se necesitan algunos datos antes de poder hacer la importación
- Realizar la importación

En Odoo consideremos:

- Los ficheros CSV deben separarse por punto y coma
- El separador de texto en el CSV debe ser comillas dobles
- La primera fila del CSV debe contener los nombres de los campos en el idioma dado en las preferencias de la aplicación
- Revisar los datos de las tablas secundarias. (Si el campo no existe en la tabla secundaria... ejemplo importamos empresas con una categoría Cliente y esta no existe en res.partner.category)

## 6.5. Planificación de la implantación

Figuras clave:

- Dirección o Responsables de la empresa: Toma de decisiones en el proyecto. Debe estar plenamente implicada en el mismo
- Jefe de proyecto: De la empresa o agente externo. Valida, verifica y hace interlocutor entre todos los miembros
- Responsable de migración: Conoce bien el sistema antiguo y las necesidades a cubrir con el nuevo
- Equipo de consultoría: Realizan labores de análisis inicial de proceso y requisitos, propuesta de solución, instalación y configuración del sistema, formación de usuarios y programación a medida de los módulos que se necesiten.

Etapas:

- Análisis de procesos y enfoque de solución
- Planificación del proyecto (tiempos y coste)
- Fase de instalación (traspaso de datos, inicio de programación, formación a responsables)
- Fase de consultoría (formación a usuarios e instalación de módulos)
- Fase de pruebas (antiguo en paralelo con el nuevo)
- Puesta en marcha
- Revisión de funcionalidades y realización de ajustes
- Finalización del proyecto

## 7. Configuración del sistema

Establecer parámetros del sistema para que se ajusten a las necesidades de la empresa. Adaptación de informes, consultas y otros objetos.

Los cambios en la configuración se hacen desde la interfaz del cliente, modificando la forma general en qué funciona y las diferentes herramientas de análisis usadas. También se puede cambiar la apariencia, asignar funciones a usuarios, establecer qué operaciones se pueden realizar.

Es importante configurar el derecho de acceso a la información con políticas para los diferentes usuarios.

Evitar que usuarios inexpertos hagan cambios no adecuados en la aplicación, introduzcan incongruencias o actúen de mala fe.

## 7.1. Control de acceso

Hay **usuarios** y **grupos**, pudiendo un usuarios pertenecer a uno o más grupos que determina qué menús puede visualizar y a qué tablas de la base de datos puede acceder.

Lo primero es configurar los grupos (importante que sean representativos). Desde Ajustes -> Usuarios y Compañías. Allí a qué módulos se da acceso a los distintos usuarios.

Para accesos a cada usuario en la ficha de cada uno aparecen varias pestañas y en Permisos de acceso y pulsando sobre el botón editar, modificarlos.

Ejemplo: Grupo **Comercial** con acceso solo a algunos menús de **Empresas** y ninguna nformación contable.

Los usuarios del **Departamento de Ventas** se hacen miembros del grupo **Comercial**:

Se puede crear un **Responsable de Ventas** con acceso a los permisos de **Comercial** y además a las comisiones de venta.

## 7.2. Cambiar la apariencia del sistema

Al modificar la apariencia debe valorarse la necesidad real de hacerlo porque puede suponer tener que formar de nuevo a los usuarios y actualizar la documentación.

Al modificar el componente, puede ser interesante duplicarlo, así se mantendrá el componente original si se necesita volver atrás.

## 8. Puesta en marcha y finalización del proyecto

Se hacen las pruebas definitivas de todos los módulos con dos posibles formas:

- Pruebas de funcionamiento en paralelo: Permite evaluar desajustes y, si todo coinciden, abandonar el antiguo sistema. Provoca entradas de datos duplicadas y coste e nel tiempo.
- Bloqueo del sistema antiguo y puesta en marcha con el ERP del sistema implementado. Si no se ha probado el nuevo puede fallar en la puesta en marcha.

La elección de uno u otro depende de la bondad de las pruebas realizadas: Debe ser una fase exhaustiva y organizada con sumo cuidado. Así, sí podría ponerse en marcha el nuevo directamente.

Pasado un tiempo prudencial de la puesta en marcha se puede dar por finalizada la implantación. Debe hacerse revisión final para ver si:

- Se han alcanzado los objetivos previstos
- El funcionamiento de los módulos es adecuado
- Los usuarios están formados
- Se ha cumplido el presupuesto o hay desviaciones
- No hay errores, sobrecargas, imprevistos en el sistema.

### 8.1. Factores de éxito y fracaso en la implantación de un ERP

#### Factores de éxito

- Buena dirección del proyecto
- Dotación de medios adecuada
- Implicación y compromiso de toda la organización

#### Factores de fracaso

- Falta de liderazgo del equipo directivo: No hay objetivos claros o no existe compromiso de cambio al nuevo sistema

- Resistencia al cambio: Desconfianza en los consultores externos, poca o mala formación, hábitos de trabajo
- Consultores inexpertos sin formación o sin experiencia
- Software ERP poco flexible
- Software ERP con interfaz poco amigable
- Funcionalidad que se atribuyó al ERP pero que este no contempla
- Falta de capacidad y/o recursos técnicos y/o humanos del proyecto

Debe analizarse cuáles son los puntos críticos y solucionarlos poco a poco empezando por los más necesarios. Concienciar a todo el personal de que la implantación no es proceso trivial y que requiere colaboración de todos los que vayan a usarla.

## 8.2. Comenzar con la gestión

Configurado e implantado el sistema deben introducirse datos, crear, configurar todos los módulos.

Según cada empresa se necesitará más o menos parámetros.

### Gestión de compra-venta

Compras proveedores, Clientes ventas

- 1.-Crear clientes /proveedores.
- 2.-Categorías de productos.
- 3.-Crear productos.
- 4.-Crear el stock inicial.
- 5.-Crear una orden de compra - venta.

### Gestión de almacén

Almacén: Localización física de elementos de stocks. Albaranes de entrada, de salida e internos. De entrada para recepción, de salida para salida de productos, internos para movimientos entre almacenes.

- 1.-Crear estructura.
- 2.-Crear categorías de productos.
- 3.-Crear productos.
- 4.-Crear el stock inicial.
- 5.-Establecer reglas de abastecimiento.
- 6.-Comprobar los niveles de stock.

**Gestión de contabilidad:** Información fluye en tiempo real junto con los sistemas de compras. Se abastece de configuración de ejercicios y periodos, diarios, plan contable, impuestos, plazos y tipos de pago y de la actividad diaria del sistema.

- 1.-Configuración.
- 2.-Gestión de facturas.
- 3.-Gestión de asientos.
- 4.-Gestión de la conciliación.
- 5.-Generación de informes.
- 6.-Cierre de periodos fiscales.
- 7.-Cierre del año fiscal.

**Gestión de los recursos humanos:** Creación, modificación y mantenimiento de empleados y contratos, control de asistencia, gestión de nóminas

- 1.-Configuración de asistencia.
- 2.-Configuración de la empresa.
- 3.-Configuración de los contratos.
- 4.-Alta de contratos y empleados.
- 5.-Gestión de la asistencia.
- 6.-Nóminas.
- 7.-Informes.

**Gestión de las relaciones con los clientes:** Según características de la empresa.



- 1.-Configuración.
- 1.1.-Crear iniciativa.
- 1.1.1.-Crear oportunidad.
- 1.1.2.-Convertir a presupuesto.
- 1.2.-Reclamaciones.
- 1.3.-Ayuda.

## 8.3. Comercio electrónico o e-commerce

Es la compra y venta de productos o servicios a través de medios electrónicos. Consiste en compra y venta entre personas y empresas, aunque también en la adquisición de artículos virtuales.

### Ventajas:

- **Mejoras en la distribución:** Costos de distribución o ventas que tienden a cero, entregas de inmediato
- **Comunicaciones comerciales por vía electrónica:** Fidelización de clientes mediante dialogo asincrónico a la conveniencia de ambas partes
- **Beneficios operacionales:** Reducción de errores, tiempo, sobrecostos en tratamiento de información. Mercados y segmentos nuevos. Ventajas en las ventas. Mayor facilidad para entrar en nuevos mercados.
- **Facilidad para fidelizar** clientes

### Características:

- **Ubicuidad:** Siempre disponible. Marketspace, realizar compras en cualquier lugar.
- **Alcance global:** Llega a todo el mundo
- **Estándares universales:** XML, HTML, CSS...
- **Riqueza:** Una sola experiencia de consumo y mensaje de comercialización
- **Interactividad:** Interacción con el usuario
- **Densidad de información:** Mayor prevalencia, precisión y actualidad. Información abundante, económica y precisa.
- **Personalización/adecuación**
- **Tecnología social:** Generación de contenido por parte del usuario en redes sociales

# UD 04 - DESARROLLO DE COMPONENTES

## 1. Técnicas y estándares. Modelo-Vista-Controlador

### Definición de MVC

El patrón Modelo-Vista-Controlador (MVC) divide la aplicación en tres componentes para conseguir que estén desacoplados:

- **Modelo:** Datos de la aplicación (Tablas)
- **Vista:** Interfaz de usuario (XML)
- **Controlador:** Forma en la que la interfaz reacciona a la entrada del usuario (Código)
- El controlador tiene acceso completo a la vista y al modelo
- El modelo tiene acceso limitado a la vista: El modelo envía notificación de que sus datos han sido modificados para que la vista actualice el contenido. El modelo no necesita conocer la vista, pero esta sí que necesita acceder al modelo.
- La vista tiene acceso limitado al controlador: Las dependencias que la vista tiene sobre el controlador deben ser mínimas para que este pueda ser sustituido en cualquier momento.

En el MVC de Odoo:

- Modelo: Tablas de la base de datos
- Vista: Archivos XML que definen la interfaz de usuario del módulo (vistas formulario, vistas tipo árbol...)
- Controlador: Código que define el comportamiento de la aplicación. Objetos creados en Python



## Especificaciones técnicas y funcionales

### Técnicas

- La arquitectura cliente-servidor se permite al cliente hacer llamadas a procedimientos remotos para comunicación (ejecutar código en otra máquina). Esto se hace mediante los protocolos XML-RPC (llamada, argumentos y resultados enviados por HTTP y codificadas con XML) o Net-RCP (más reciente, basado en funciones en Python)
- Odoo funciona sobre un framework OpenObject que permite desarrollo rápido de aplicaciones (RAD) conteniendo:
  - ORM: Mapeo objeto-relacional a objetos Python
  - Arquitectura MVC
  - Diseñador de informes
  - Herramientas de Business Intelligence y cubos multidimensionales
  - Cliente de escritorio o web

### Funcionales

Odoo se compone de un núcleo y varios módulos. Estos componentes además de un nombre comercial tienen un nombre técnico.

- **base:** módulo básico con objetos como empresas (res.partner), direcciones de empresas (res.partner.address), usuarios (res.user), monedas (res.currency)...
- **account:** Gestión contable y financiera
- **product:** Productos y tarifas
- **purchase:** Gestión de compras
- **sale:** Gestión de ventas
- **mrp:** Fabricación y planificación de recursos
- **crm:** Gestión de las relaciones entre clientes y proveedores

En el directorio `addons` del servidor:

- Por cada módulo se corresponde una carpeta con el nombre del módulo en ese directorio
- Hay directorios con subcarpetas como report, wizard, test que contienen archivos del tipo de objeto utilizado (informes, asistentes y accesos directos, datos de prueba...)

Los archivos `_____.py` contienen valores dentro de un diccionario Python ({}):

name: nombre del módulo

version: versión del módulo

description: descripción de módulo

author: autor del módulo

website: sitio web del módulo

license: licencia del módulo (por defecto GPL )

depends: lista de módulos de los que depende el módulo (el módulo base es el más usado porque se definen datos para vistas, informes...)

init\_xml: archivos XML que se cargan con la instalación del módulo

installable: determina si el módulo es instalable o no

## 2. Lenguaje de los sistemas ERP-CRM. Python en Odoo

Los ERP tienen lenguajes de programación modernos, muchos orientados a objetos. Pueden tener lenguajes propietarios como SAP ABAP o Apex.

En el caso de Odoo se utiliza Python (Guido van Rossum, años 80 en Países Bajos; existe la Python Software Foundation (PSF) para poseer propiedad intelectual sobre la licencia y promover el uso del lenguaje; Licencia de código abierto, compatible con GPL)

### Características de Python

- **Sintaxis** sencilla
- **Lenguaje interpretado:** Se ejecutan usando un intérprete. El código fuente se traduce a archivo bytecode con extensión `.pyc` o `.pyo` que se ejecutará en sucesivas ocasiones
- **Tipado dinámico:** El tipo de los datos se determina en tiempo de ejecución
- **Fuertemente tipado:** No se puede usar la variable como si fuera de otro tipo, salvo conversión
- **Multiplataforma**
- **Orientado a objetos**

Tutorial en: <https://tutorialpython.com/>

### 2.1. Empezando con Python

Los archivos pueden ejecutarse con `python programa.py`

El primer programa:

```
print('Hola mundo')
```

Los comentarios en el lenguaje se escriben con `#` para una línea o con `"""` si es de más de una línea

En este lenguaje no hay delimitadores de sentencia como los puntos y coma, ni delimitadores de bloques como las llaves.

## 2.2. Variables

No se declaran, se utilizan directamente.

- Números enteros (Se representan con el tipo int que usa 32 o 64 bits dependiendo del tipo de procesador o el tipo long que permite almacenar números de cualquier precisión, limitados por la memoria disponible en la máquina ya que a bajo nivel se implementa con C por debajo)
- Números reales (float)
- Números complejos
- Cadenas
- Booleanos
- Operadores aritméticos (...división entera //, exponencial \*\*), booleanos (and, or, not), relacionales (== != < > <= >=)

```
# Define dos variables numero1 y numero2
numero1 = int(2)
numero2 = float(2.5)

"""
Y ahora me apetece jugar con printf
"""
# Concatenando strings
print("Querido Alcalde de Alicante " + "que así se llama")
# Imprimiendo numerito
print(numero1)
# Imprimiendo numeritos concatenados al string
print("Este es el numero1: (Parsealo a str o no compila)" + str(numero1))
print("Este es el numero2: (Parsealo a str o no compila) " + str(numero2))
# Con text blocks
print("""
    Oye pues es un poquito curioso esto de Python    Una pena que solo lo vaya a usar un
    día""")
# Con text blocks formateados
print("""
Ojalá saber quien es {0} para saludarlo
""".format("M. Rajoy"))
# Con payloads
ciudad = "Málaga"
print(f"Vaya ciudad más bonita es {ciudad}, aunque muy llena de turistas está")
```

## 2.3. Colecciones

### A. Listas

Son mutables. Se definen con corchetes.

`lista[i]`: Devuelve el elemento  
`lista.pop(i)`: Devuelve el elemento de i y lo borra  
`lista.append(elemento)`: Añade elemento al final  
`lista.insert(i, elemento)`: Inserta en posición i  
`lista.extend(lista2)`: Fusiona lista con lista 2  
`lista.remove(elemento)`: Elimina la primera vez que aparece el elemento  
Pueden ser desempaquetadas (asigno la lista a múltiples variables en una sola instrucción)

```
# Declaramos una lista
myList=["Manzana", "Pera"]
fruta = myList[0]
## La puedo desempaquetar
a,b = myList
print(str(a))
print(str(b))
print(f"Mi fruta preferida es la {fruta}")
print("Pero si quiero la elimino de la lista {0} y chao".format(str(myList)))
myList.pop(0)
print("¿ves? ya no está:"+str(myList))
myList2=["Patata", "Lechuga", "Pimiento"]
```

```
## Elimino
del(myList2[1])
print("Pongamos un {0} y a juntarlo con la otra".format(myList2))
myList.extend(myList2)
print("¿ves? {0}".format(str(myList)))

#¿Y si fuese una tupla? (Inmutable)
myTupla=("Manzana", "Pera")
print(str(myTupla))
```

## B. Tuplas

Las tuplas son inmutables. Se definen con paréntesis.

```
myTupla=("Manzana", "Pera")
```

## C. Diccionarios

Diccionarios antes no tenían orden.

Ahora guardan el orden de inserción de los elementos.

`diccionario.get('key')`: Toma del diccionario esa clave  
`diccionario.pop('key')`: Toma y elimina del diccionario esa clave  
`diccionario.update({'key': 'valor'})`: Actualiza valor de esa clave  
`"key" in diccionario`: devuelve true o false según esté o no en el diccionario como key  
`"definicion" in diccionario`: devuelve true o false según esté o no en el diccionario como valor  
`del diccionario[key]`: Elimina del diccionario  
`diccionario["key"] = "value"`: Añade al diccionario

```
# Inicio un diccionario
diccionario = {'diasDescuento': '20', 'diasIntegro': '30', 'diasDevolucion': '40'}
print(diccionario)
# Cojo una de las claves
print(diccionario.get('diasDescuento'))
# Compruebo si una de las claves está
print('diasDescuento' in diccionario)
# O si uno de los valores está
print('40' in diccionario.values())
# Elimino un valor
del diccionario['diasDevolucion']
print(diccionario)
# Modifico un valor
diccionario.update({'diasDescuento': '25'})
# Añado un valor
diccionario["diasRebajas"] = "12"
print(diccionario)
```

## D. Conjuntos

No admiten duplicados

Se crean con llaves pero como si fuesen una lista

```
mySet = {'rojo', 'amarillo', 'verde'}
# Lo puedo usar para borrar de duplicados de una lista
lista = ['Fernando', 'Fernando', 'Felipe']
print(lista)
# ['Fernando', 'Fernando', 'Felipe']
lista = list(set(lista))
print(lista)
# ['Fernando', 'Felipe']
```

## 2.4. Sentencias del lenguaje

## A. Condicionales

Se hacen con `if`, `elif` y `else`  
Y los operadores lógicos son `and`, `or`...

```
x = 10
if x > 0 and x % 2 == 0:
    print("x es un número positivo par")
elif x > 0 and x % 2 != 0:
    print("x es un número positivo impar")
elif x == 0:
    print("x es igual a cero")
else:
    print("x es un número negativo")
```

---

## Match

```
status = 418
match status:
    case 400:
        print("Bad request")
    case 404:
        print("Not found")
    case 418:
        print("I'm a teapot")
    case _:
        print("Something's wrong with the internet")
```

## B. Bucles

El `while` no tiene misterio (salvo que no hay `i++`)

El `for` es `for i in range(n)`. Aunque si se necesita se puede especificar el paso.

El foreach es `for elemento in mi_lista` o similar

```
contador = 0
print("contador while")
while contador < 5:
    print(contador)
    contador += 1

print("contador for in range")
for i in range(5):
    print(i)

print("contador for in range inverso")
for i in range(10, 0, -1):
    print(i)

print("contador foreach")
mi_lista = [1, 2, 3, 4, 5]
for elemento in mi_lista:
    print(elemento)

print("contador foreach diccionario")
# Bucle for con diccionario
mi_diccionario = {"a": 1, "b": 2, "c": 3}
for clave in mi_diccionario:
    print(clave, mi_diccionario[clave])
```

Sentencias `break`, `continue`, `else` hacen lo esperado..

Y, si el bucle for, termina sin encontrar un `break` podría usarse una clausula `else` para ejecutar algo

```

for n in range(2, 10):
    for x in range(2, n):
        if n % x == 0:
            print(n, 'equals', x, '*', n//x)
            break
    else:
        # loop fell through without finding a factor
        print(n, 'is a prime number')

```

La sentencia `pass` no hace nada. Se puede usar cuando una sentencia es requerida por la sintaxis pero el programa no requiere ninguna acción.

```

while true
    pass

clas MyEmptyClass
    pass

def initlog(*args)
    pass #todo

```

## 2.5. Llamadas a funciones

Se inicia como

```
def nombreFuncion:
```

La indentación es la que permite diferenciar qué está dentro y qué está fuera de la función.

Puede o no tener un `return`. Cuando la función no tiene `return`, devuelve el valor "None"

```

#Declaramos funcion
def sumaNumeritos(a, b):
    print(a+b)

def restaNumeritos(a,b):
    return a - b

def cuadradoNumerito(a):
    return a**2
# Comprobamos que devuelve "None"
print(sumaNumeritos(1,2))
# Comprobamos que devuelve -1
print(restaNumeritos(1,2))
# Comprobamos cómo es la operación de la potencia
print(cuadradoNumerito(3))

```

## 2.6. Clases y objetos

Se definen con la palabra `class` seguida del nombre de la clase, dos puntos (:) y el cuerpo de la clase.

```

class Individuo:
    def __init__(self, nombre, edad):
        self.nombre = nombre
        self.edad = edad

    def dimeEdad(self):
        if self.edad >= 18:
            print("Mayor de edad")
        else:
            print("Menor de edad")

# Crear una instancia de Individuo
individuo = Individuo("Jesús", 23)
individuo.dimeEdad() # Salida: "Mayor de edad"
print(individuo.edad) # Salida: Jesús

```



```

class Persona:
    def __init__(self, nombre, edad):
        self.nombre = nombre
        self.edad = edad

    def saludar(self):
        if self.nombre is not None:
            print(f"Hola, mi nombre es {self.nombre}.")
        else:
            print("Hola, soy una persona sin nombre.")

# Crear una instancia de Persona sin proporcionar nombre
persona = Persona()
persona.saludar() # Salida: "Hola, soy una persona sin nombre."

# Establecer el nombre de la persona después de la creación
persona.nombre = "Juan"
persona.saludar() # Salida: "Hola, mi nombre es Juan."

```

## 2.7. Módulos y paquetes

Un módulo es un archivo que contiene **declaraciones y definiciones de Python**. (O sea, un archivo de Python `.py`) (Archivos)

Un paquete es una colección de módulos relacionados. (Directorios)

Esto ayuda a:

- Hacer el código más mantenible
- Reutilizar código

Se usa con `import nombre_modulo`

O si quisieramos importar algo concreto `from nombre_modulo import Persona`

funciones.py

```

def comenzamosModulos():
    print("Comenzamos los modulos")

```

miCodigo.py

```

import funciones

funciones.comenzamosModulos()

```

Para que Python trate a un directorio como paquete debe crearse un archivo `__init__.py` en dicha carpeta. Para hacer que un módulo pertenezca a un paquete hay que copiar el archivo `nombre_modulo.py` en el directorio del paquete.

Los paquete se importan también con la sentencia `import`

Muchos pragramadores elaboran módulos y los comparten.

Para instalar el módulo simplemente hay que copiar el archivo `.py` en la carpeta en la que se tenga el código fuente. Sin embargo, puede instalarse de forma más automatizada.

## 2.8 Manejo de excepciones

Las excepciones en Python se manejan con `try-except`

```

try:
    def dividir(a, b):
        return a / b
    dividir(1,0)

```

```
except:
    print ("Ha ocurrido un error")

try:
    numerito = 2
    print("Este es el numero1: (Parsealo a str o no funciona)" + numerito)
except:
    print ("Ha ocurrido un error")
```

## 2.9. Librerías de funciones (APIs)

Python proporciona una API estándar cuyo código fuente está abierto y libre. Destacan:

- `os`: Funciones del sistema operativo como creación de archivos y directorios, leer o escribir un fichero, manipular rutas
- `sys`: Acceder a información sobre el intérprete de Python como el prompt del sistema, datos de licencia y cualquier parámetro o variable que tenga que ver con el intérprete
- `datetime`: Módulos para la manipulación de fechas y horas
- `math`: Funciones matemáticas
- `locale`: configuración regional del sistema (moneda, formato de fecha y hora, números, codificación)
- `MySQLdb`: interfaz para acceder a MySQL desde Python

## 2.10. Inserción, modificación y eliminación de datos en objetos (Módulos Odoo)

Primero debe situarse la carpeta `addons` de Odoo en el sistema operativo. (En nuestra instalación de Odoo el archivo `addons.conf` la opción `addons_path`, es la ruta donde por defecto se instalan los módulos)

Es interesante usar una ruta diferente a los módulos propios que se desarrollen para no perderlos (editando el archivo `odoo.conf` y mediante una nueva ruta separada por "," de la otra.

Para crear el módulo se debe en la carpeta del módulo

- **Crear estructura del módulo** colgando de la carpeta `nombre_modulo` y teniendo cuidado con los permisos. Puede usarse el comando `odoo-bin scaffold` o bien `python odoo-bin scaffold`

También puede descargarse la estructura de un módulo de la documentación oficial de Odoo

- **Configurar los datos del módulo** (`__manifest.py__`) y meterlo en el servidor. Activar modo desarrollador y actualizar las aplicaciones.
- **Creación del modelo, meterlo en el servidor y reiniciar el servicio de Odoo** (es necesario cada vez que se modifique un archivo Python)
- **Creación de las vistas** ( archivo.xml), **editar archivo** `views.xml` e introducirlo en el servidor. No es necesario reiniciar el servicio de Odoo pero sí actualizar el módulo o desinstalar y volver a instalar.
- Comprobar el módulo. Meter registros, consultar, modificar, eliminar, realizar búsquedas...

En la estructura del módulo de Odoo se encuentran los archivos:

- `__init__.py`: Donde se inicia el módulo. Se indica qué modelo quiere vincularse al módulo. El modelo es la base de datos. Es necesario para que la carpeta se trate como paquete de Python. Contiene los import de cada archivo del módulo.
- `__terp.py__`: Contiene un diccionario Python con la descripción del módulo, autor, version, módulos de los que depende
- `nombre_modulo.py`: Clase definiendo campos que tendrá. Al crearla se crea el modelo (tabla en BBDD) y el controlador porque se define el comportamiento que tendrá.

- `nombre_modulo_view.xml`: Vista del módulo o del objeto que creará el módulo. Deben tenerse conocimientos de XML o coger vista de otro objeto y a partir de ella crear la del nuevo objeto.

### 3. Entornos de desarrollo y herramientas de desarrollo en sistemas ERP-CRM

Para programar en Python se podría usar como IDE sencillo: Idle. (Otros IDEs Visual Studio Code, Sublime Text) Allí puede abrirse el código con Python, revisarse la sintaxis (Run/Check) o ejecutar el programa (Run/Run module) entre otras opciones.

Puede accederse a extensiones o plugins de Odoo desde diferentes entornos.

### Depuración de un programa

Python incorpora un depurador dentro de su biblioteca de módulos llamado `pdb`

Tiene puntos de ruptura, ejecución paso a paso de código, inspeccionar valores de variables y otras opciones de depuración.

Aunque es mejor hacerlo desde entornos gráficos como desde el IDE IDLE en su módulo de depuración.

### 4. Formularios e informes en sistemas ERP-CRM

- **Formularios:** Interfaz de un módulo. Una vez creado el objeto del módulo `nombre_modulo.py` deben crearse los menús, acciones, vistas (formulario u otro tipo) para interactuar con el objeto. Estos elementos se describen en `nombre_modulo_view.xml`
- **Informes:** Pueden ser estadísticos (listados de datos y gráficos por pantalla) o documentos cuya finalidad es ser impresos.

#### Arquitectura de formularios e informes. Elementos

Se construyen de forma dinámica por la descripción XML de la pantalla del cliente.

La estructura `nombre_modulo_view.xml` es similar a esta:

```
<?xml version="1.0" encoding="utf-8"?>
<odoo>
  <data>
    <record model="object_model_name" id="object_xml_id">
      <field name="field1">value1</field>
      <field name="field2">value2</field>
    </record>
    <record model="object_model_name2" id="object_xml_id2">
      <field name="field1" ref="module.object_xml_id"/>
      <field name="field2" eval="ref('module.object_xml_id')"/>
    </record>
  </data>
</odoo>
```

(Ahora Odoo, antes OpenERP)

Los registros se definen con la etiqueta `record` que indica el inicio y el fin de la descripción del registro. Cada tipo de registro hace referencia a un objeto diferente. Con el atributo model puede definirse si es una acción o una vista.

Dentro del record van los field que definen las características del registro.

Un ejemplo de creación de vista de tipo formulario es:

```
<record model="ir.ui.view" id="view_agenda_form">
  <field name="name">agenda</field>
  <field name="model">agenda</field>
  <field name="type">form</field>
  <field name="priority" eval="5"/>
```

```

        <field name="arch" type="xml">
        <form string="Agenda">
            <field name="nombre" select="1"/>
            <field name="telefono" select="1" />
        </form>
    </field>
</record>

```

El `<field name="name">` hace referencia al nombre de la vista. El `<field name="model">` al objeto del modelo, es decir, la tabla en la base de datos del cual muestra la información. La vista a su vez se forma por dos campos "nombre" y "telefono"

Puede definirse un elemento del menú para acceder al objeto agenda

```

<menuitem name="Agenda" id="menu_agenda_agenda_form"/>
<menuitem name="Agenda" id="menu_agenda_form" parent="agenda.menu_agenda_agenda_form"
action="action_agenda_form"/>

```

Y la acción asociada a ese elemento de menú

```

<record model="ir.actions.act_window" id="action_agenda_form">
    <field name="res_model">agenda</field>
    <field name="domain">[]</field>
</record>

```

En este ejemplo, definimos dos vistas: una vista de formulario para el modelo `product.product` y una vista de árbol para el mismo modelo. En la vista de formulario, definimos un formulario con tres campos: `name`, `description` y `list_price`, agrupados dentro de un grupo. En la vista de árbol, definimos un árbol con tres columnas: `name`, `default_code` y `type`.

Ten en cuenta que estas definiciones de vista XML se proporcionan dentro de archivos `.xml` en los módulos de Odoo. Luego, estos archivos XML son cargados por el sistema de Odoo para construir la interfaz de usuario de la aplicación.

```

<odoo>
    <data>
        <!-- Ejemplo de una vista formulario -->
        <record id="view_form_producto" model="ir.ui.view">
            <field name="name">Formulario de Producto</field>
            <field name="model">product.product</field>
            <field name="arch" type="xml">
                <form string="Producto">
                    <sheet>
                        <group>
                            <field name="name"/>
                            <field name="description"/>
                            <field name="list_price"/>
                        </group>
                    </sheet>
                </form>
            </field>
        </record>

        <!-- Ejemplo de una vista árbol -->
        <record id="view_tree_producto" model="ir.ui.view">
            <field name="name">Árbol de Producto</field>
            <field name="model">product.product</field>
            <field name="arch" type="xml">
                <tree string="Productos">
                    <field name="name"/>
                    <field name="default_code"/>
                    <field name="type"/>
                </tree>
            </field>
        </record>
    </data>
</odoo>

```

## Herramientas para creación de formularios e informes

Se usará básicamente la creación de archivos XML con el nombre de `nombre_modulo_view.xml`

Otra forma para los informes es utilizando Jasper Reports (librería generación informes). Consiste en generar un XML con la descripción del informe a crear, que es tratado para generar un documento en formato de salida (PDF, HTML, XLS...)