|  |  |
| --- | --- |
|  |  |
| Módulo Gestión de Inventario, Tickets y Clientes |  |
|  |  |
|  | 2° DAMÁlvaro Pavón Martínez |
|  |  |

Tabla de contenido

[Descripción del Problema 3](#_Toc191450776)

[Programación de la aplicación 3](#_Toc191450777)

[Lista de Ficheros de Código Fuente y Breve Descripción 4](#_Toc191450778)

[Documentación Técnica Detallada de cada Archivo 5](#_Toc191450779)

[**\_\_manifest\_\_.py** 5](#_Toc191450780)

[**\_\_init\_\_.py** 6](#_Toc191450781)

[**Ir.model.access.csv** 7](#_Toc191450782)

[**Modelo Hardware (hardware.py)** 7](#_Toc191450783)

[**Modelo Ticket (ticket.py)** 8](#_Toc191450784)

[**Modelo ticket line (ticket\_line.py)** 10](#_Toc191450785)

[**Modelo Mantenimiento (mantenimiento.py)** 11](#_Toc191450786)

[**Extensión del módulo Cliente (cliente.py)** 11](#_Toc191450787)

[**Vistas y Menús (Archivos XML)** 12](#_Toc191450788)

[Manual de Usuario 20](#_Toc191450789)

[Instalación y configuración inicial 20](#_Toc191450790)

[Requisitos previos 20](#_Toc191450791)

[Pasos para la instalación 20](#_Toc191450792)

[Navegación en la aplicación 21](#_Toc191450793)

[Inicio de sesión y acceso 21](#_Toc191450794)

[Menú principal y submenús 22](#_Toc191450795)

[Gestión del inventario (hardware) 22](#_Toc191450796)

[Creación de un registro de hardware 22](#_Toc191450797)

[Edición y consulta 23](#_Toc191450798)

[Gestión de tickets 24](#_Toc191450799)

[Registro de incidencias y comprobantes 24](#_Toc191450800)

[Asociaciones en el ticket 25](#_Toc191450801)

[Gestión de mantenimientos 25](#_Toc191450802)

[Registro de mantenimientos 25](#_Toc191450803)

[Consulta y búsqueda 26](#_Toc191450804)

[Gestión de clientes 26](#_Toc191450805)

[Registro y consulta del historial de tickets 26](#_Toc191450806)

[Uso de reportes 27](#_Toc191450807)

[Generación del reporte PDF 27](#_Toc191450808)

[Webgrafía 28](#_Toc191450809)

[Conclusión 29](#_Toc191450810)

[Link de descarga 29](#_Toc191450811)

# Descripción del Problema

El proyecto consiste en desarrollar un módulo para Odoo titulado “Gestión de Inventario, Tickets, Mantenimientos y Clientes”. La aplicación esta diseñada para solucionar varias problematicas que enfrentan las empresas:

* **Control del Inventario:**

Permite registrar y mantener actualizado el stock de productos o equipos (hardware), incluyendo información como nombre, descripción, modelo, precio, marca, imagen y cantidad disponible, y asignar un responsable a cada producto si fuese necesario.

* **Gestion de Tickets:**

Los tickets se generan para entregar un comprobante a los clientes una vez realizado el servicio o la venta. Ademas, se mantiene un registro en el sistema que diferencia los tickets pagados de los pendientes de pago, facilitando el control de la facturación.

* **Asignacion de mantenimientos:**

Permite registrar y asignar mantenimientos o servicios a los productos del inventario, lo que facilita el seguimiento de intervenciones y reparaciones.

* **Registro de clientes:**

Los clientes se registran cuando se genera un ticket, es decir, se crean en el sistema al producirse una venta o servicio. Este registro se utiliza para llevar un historial de clientes que han comprado para dar unn seguimiento a futuras incidencias.

La solución integrada mejora la eficiencia operativa, asegura un control riguroso del stock, facilita la entrega de comprobantes a los clientes y permite mantener un historial detallado de compras y servicios.

Al final de esta documentación encontraras el Link de descarga del modulo.

# Programación de la aplicación

La aplicación se organiza en varios componentes clave:

* **Archivos de Configuración:**
* ***\_\_manifest\_\_.py***: Define la identidad del módulo, dependencias, archivos a cargar y metadatos esenciales (nombre, versión, autor, etc.).
* ***\_\_init\_\_.py***: Importa los submódulos que contienen la lógica y definición de modelos.
* **Modelos y lógica de negocio (Archivos .py):**
* ***Hardware.py***: Modelo hardware para registrar y gestionar los productos del inventario.
* ***Ticket.py***: Modelo ticket para entregar comprobantes a los clientes y para llevar el control de los tickets pagados y pendientes.
* ***Ticket\_line.py***: Gestiona las lineas de producto en cad ticket, controlando y ajustando el stock
* ***Mantenimiento.py***: Modelo Mantenimiento (custom) para registrar servicios realizados en el inventario
* ***Cliente.py***: Extiende el modelo res.partner para registrar a los clientes que han generado un ticket (clientes que han comprado) y llevar su historial
* **Vistas y Menús (Archivos .xml):**
* Se definen vistas (formulario, lista y búsqueda) para cada modelo
* Se configuran menús de navegación para acceder a las areas de Inventario, Tickets, Mantenimientos y Clientes
* Se establecen plantillas y layouts para generación de reportes PDF, que se entregan a los clientes como comprobantes
* **Seguridad**
* Archivo CSV (ir.model.access.csv) definen los permisos de acceso a cada modelo, protegiendo la integridad de los datos

# Lista de Ficheros de Código Fuente y Breve Descripción

* **\_\_manifest\_\_.py:**

Funcionalidad: Configuración e identificación del módulo, definición de dependencias y datos a cargar.

Cometido: Permitir que Odoo reconozca, instale y configure el módulo correctamente

* **\_\_init\_\_.py:**

Funcionalidad: Importa los módulos Python que contienen los modelos y la lógica de negocio.

Cometido: Garantizar la carga inicial de todos los componentes del modulo

* **Hardware.py:**

Funcionalidad: Define el modelo Hardware para registrar productos del inventario

Cometido: Gestionar datos esenciales (nombre, descripción, modelo, precio, marca, imagen, stock y responsable si fuese necesario) y calcular el número de tickets asociados

* **Ticket.py:**

Funcionalidad: Define el modelo Ticket para registrar ventas.

Cometido: Documentar todas las ventas y entregar comprobantes a los clientes y mantener un registro de los tickets pagados y pendientes

* **Ticket\_line.py:**

Funcionalidad: Gestiona las líneas de producto en un ticket.

Cometido: Permitir la selección de productos, definir cantidades, calcular totales y ajustar el stock automáticamente.

* **Mantenimiento.py:**

Funcionalidad: Define el modelo Mantenimiento (custom) para registrar servicios realizados.

Cometido: Permitir la asignación de mantenimientos a uno o varios productos del inventario

* **Cliente.py:**

Funcionalidad: Extiende el modelo res.partner para incorporar una lista de tickets por cliente

Cometido: Registrar a los clientes una vez que generan un ticket, llevando un historial de clientes que han comprado

* **Archivos XML:**

Funcionalidad: Configuran vistas, menús y reportes PDF.

Cometido: Proveer una interfaz de usuario organizada e intuitiva

* **Archivo de seguridad (ir.model.access.csv)**

Funcionalidad: Define los permisos de acceso a cada modelo.

Cometido: Garantizar la integridad y seguridad de los datos

# Documentación Técnica Detallada de cada Archivo

## **\_\_manifest\_\_.py**

* **Propósito**:

Establece la identidad y configuración del modulo

* **Elementos clave:**
* Name, summary, description: Detallan la gestión del inventario, tickets (para comprobantes y gestión de pagos), mantenimientos y clientes.
* Depends: Especifica la dependencia de los módulos base y producto
* Data: Lista archivos XML, CSV (incluyendo ir.model.access.csv) a cargar

Texto

El contenido generado por IA puede ser incorrecto.

## **\_\_init\_\_.py**

* **Propósito:**

Importar todos los modelos y la lógica

* **Elementos clave:**
* Importa: hardware, cliente, ticket\_line, ticket y mantenimiento

Texto

El contenido generado por IA puede ser incorrecto.

## **Ir.model.access.csv**

* **Propósito:**

Define los permisos de acceso a cada modelo

* **Elementos clave:**
* Permite controlar quien puede leer, crear, actualizar o eliminar registros, protegiendo la integridad de los datos

Texto

El contenido generado por IA puede ser incorrecto.

## **Modelo Hardware (hardware.py)**

* **Propósito**:

Gestionar el inventario de productos

* **Campos básicos:**

Nombre, descripción, modelo, precio, marca, imagen, stock, responsable

* **Campo computado:**

Ticket\_count: Calcula incidencias asociadas

* **Validaciones y Acciones:**

Valida precio y ofrece action\_view\_tickets para acceder a los tickets relacionados

Texto

El contenido generado por IA puede ser incorrecto.

## **Modelo Ticket (ticket.py)**

* **Propósito:**

Registrar tickets que se usan para entregar comprobantes y controlar tickets pagados y pendientes

* **Campos clave:**

Titulo, descripción, fechas, estado; relaciones con clientes, contactos, líneas de producto y mantenimientos

* **Campos computados:**

Total\_price y pagado para mostrar el estado visualmente

* **Acciones:**

Toggle\_estado para cambiar el estado y action\_print\_ticket para generar el PDF

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

## **Modelo ticket line (ticket\_line.py)**

* **Propósito:**

Gestionar cada línea de producto en un ticket

* **Calculo automático:**

Line\_total = precio del producto x cantidad

* **Control de Stock:**

Métodos créate, write y unlink ajustan el inventario automáticamente

* **Validaciones:**

Se asegura que la cantidad sea mayor que 0

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

## **Modelo Mantenimiento (mantenimiento.py)**

* **Propósito:**

Registrar servicios realizados sobre los productos

* **Campos**:

Nombre, descripción, costo

* **Relación:**

Hardware\_ids: Asocia mantenimientos a productos

Texto

El contenido generado por IA puede ser incorrecto.

## **Extensión del módulo Cliente (cliente.py)**

* **Propósito:**

Registrar a los clientes que generan un ticket y llevar su historial

* **Elementos clave:**

Campo tickets\_ids y método action\_view\_tickets en res.partner

Texto

El contenido generado por IA puede ser incorrecto.

## **Vistas y Menús (Archivos XML)**

* **Vistas:**

Formularios, listas y búsquedas para cada modelo

* **Menú:**

Menú principal “Gestión de Inventario, Tickets, Mantenimientos y Clientes” y submenús

* **Reportes:**

Report\_layout.xml y report\_ticket.xml definen el formato de los comprobantes en PDF

* ***Cliente\_views.xml***

Texto

El contenido generado por IA puede ser incorrecto.

* Texto

  El contenido generado por IA puede ser incorrecto.Texto

  El contenido generado por IA puede ser incorrecto.***Hardware\_views.xml***
* ***Mantenimiento\_views.xml***

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

* ***Menú\_views.xml***

Texto

El contenido generado por IA puede ser incorrecto.

* ***Report\_layout.xml***

Texto

El contenido generado por IA puede ser incorrecto.

* ***Ticket\_views.xml***

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.Texto

El contenido generado por IA puede ser incorrecto.

* Texto

  El contenido generado por IA puede ser incorrecto.Texto

  El contenido generado por IA puede ser incorrecto.***Report\_ticket.xml***

Texto

El contenido generado por IA puede ser incorrecto.

# Manual de Usuario

El manual de usuario explica de forma practica como instalar, configurar y utilizar la aplicación. Se incluyen instrucciones paso a paso

## Instalación y configuración inicial

### Requisitos previos

* Odoo en versión compatible (Odoo 18 o superior)
* Módulos base y producto activos
* Acceso al directorio addons

### Pasos para la instalación

1. **Copia del módulo:**

Descarga y coloca la carpeta del modulo en el directorio de addons

1. **Actualizar la lista de módulos:**

Selecciona “Actualizar lista de módulos” desde el menú de aplicaciones

1. **Instalación**:

Busca “gestion\_hardware” y pulsa en instalar

1. **Verificación** **inicial**:

Comprueba que el icono y el nombre aparezcan correctamente

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

## Navegación en la aplicación

### Inicio de sesión y acceso

* Inicia sesión con tus credenciales en Odoo
* La aplicación se mostrará en el panel principal

Interfaz de usuario gráfica, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

### Menú principal y submenús

* Menú principal: “Gestión de Inventario, tickets, mantenimientos y clientes
* Submenús:
* Inventario (hardware): gestión y control del stock
* Tickets: registro de incidencias (comprobantes para clientes y control de pagos)
* Mantenimientos: registro y asignación de servicios
* Clientes: registro de clientes (generados al emitir un ticket) y su historial.

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

## Gestión del inventario (hardware)

### Creación de un registro de hardware

1. Acceso a la sección:

En “Inventario”, pulsa “Crear”.

1. Rellenar el formulario:

* Nombre: Ejemplo: “Impresora Láser HP”
* Descripción: “Impresora láser de alta velocidad, ideal para oficinas”
* Modelo: “LaserJet Pro MFP M426”
* Precio: Ej. 250 (valor positivo)
* Marca: “HP”
* Imagen: Sube una imagen
* Stock: Ej. 10 unidades
* Responsable: Selecciona el usuario encargado en caso de que se necesite

1. Guardar registro:

Se actualiza el stock y el campo de incidencias (inicialmente 0)

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

### Edición y consulta

* Visualiza la lista de productos y edita o consulta detalles
* Pulsa “Ver Tickets” para ver incidencias asociadas

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

## Gestión de tickets

### Registro de incidencias y comprobantes

1. Crear un ticket:

En “Tickets”, pulsa “Crear”

1. Completar el formulario:

* Titulo: Ejemplo: “Falla en impresora HP”
* Descripción: “La impresora no imprime y muestra errores”
* Fecha del ticket: selecciona la fecha
* Estado: Inicialmente “pendiente”
* Cliente: Selecciona el cliente (se registra automáticamente al generar el ticket)
* Contactos: Añade contactos adicionales si procede

1. Agregar líneas de producto:

* En “Productos”, pulsa “Agregar un elemento”
* Selecciona el hardware y define la cantidad
* El sistema calcula el total y verifica el stock

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

### Asociaciones en el ticket

* Cambiar estado:

Pulsa “Cambiar Estado” para alternar entre “pendiente” y “pagado”, facilitando el control de tickets entregados y pendientes

* Generar reporte PDF:

Pulsa “Imprimir Ticket” para generar un comprobante en PDF que se entrega al cliente y se registra internamente

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

## Gestión de mantenimientos

### Registro de mantenimientos

1. Acceder a la sección:

En “Mantenimientos”, pulsa “Crear”

1. Rellenar el formulario:

* Nombre: Ejemplo: “Reparación de impresora HP”
* Costo: Ej. 50
* Descripción: “Reemplazo de tóner y mantenimiento general”
* Hardware asociado: Selecciona uno o varios productos (ej. “Impresora Láser HP”)

1. Guardar el registro:

El mantenimiento se vincula a los productos seleccionados

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

### Consulta y búsqueda

* Utiliza la vista lista y la barra de búsqueda para filtrar mantenimientos

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

## Gestión de clientes

### Registro y consulta del historial de tickets

1. Registro automático:

Los clientes se registran automáticamente al generar un ticket (cuando se realiza una compra o servicio)

1. Acceso al módulo de tickets

Usa el submódulo “tickets”

1. Visualización de la ficha:

Selecciona un ticket para ver su ficha, que muestra la información y cliente

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

## Uso de reportes

### Generación del reporte PDF

1. Desde el ticket:

Pulsa “Imprimir ticket” en el formulario del ticket

1. Contenido del reporte:

* El PDF incluirá:
* Encabezado: logotipo, dirección, teléfono y correo
* Detalles: titulo, descripción, fechas, estado y cliente
* Productos: tabla con cada línea (nombre, cantidad, total)
* Mantenimientos: tabla (si existen) con detalles de servicios
* Resumen: total final del ticket

1. Utilidad:

El comprobante PDF sirve para entregar al cliente y para mantener el registro interno, diferenciando tickets pagados y pendientes

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

# Webgrafía

* Documentación oficial de Odoo:
* <https://www.odoo.com/documentation>
* Ejemplos y tutoriales de OpenAcademy:
* <https://github.com/OCA/OpenAcademy>
* Foros de ayuda de Odoo:
* <https://www.odoo.com/forum/help-1>
* Guía de buenas prácticas en desarrollo de módulos de Odoo:
* <https://vauxoo.github.io/odoo/howtos/backend.html>
* Informes personalizados:
* <https://www.odoo.com/documentation/18.0/es/applications/finance/accounting/reporting/customize.html>
* <https://www.odoo.com/documentation/18.0/es/developer/tutorials/pdf_reports.html>
* Etapas y estados de tareas:
* <https://www.odoo.com/documentation/18.0/es/applications/services/project/tasks/task_stages_statuses.html>
* Filtros y agrupaciones personalizadas:
* <https://www.odoo.com/documentation/18.0/es/applications/essentials/search.html>
* Heredar de un módulo de Odoo:
* <https://www.youtube.com/watch?v=1R4jtxj72Lc>

# Conclusión

El módulo “Gestión de inventario, tickets, mantenimientos y clientes” integra de manera eficaz el control del stock, la generación y seguimiento de tickets (que se utilizan para entregar comprobantes a los clientes y llevar un registro de tickets pagados y pendientes), la asignación de mantenimientos y el registro de clientes que han realizado compras.

Gracias a la implementación de campos computados, validaciones automáticas y vistas personalizadas, el sistema optimiza la gestión operativa y facilita la documentación formal de los servicios.

# Link de descarga

* Link modulo:
* <https://github.com/AlvaroPavon/Modulo-Odoo>